

BUKU KERJA PRAKTEK MAHASISWA



WORKSHOP APLIKASI MOBILE

SEMESTER 3

Tim Penyusun:

Arvita Agus Kurniasari, S.ST., M.Tr.Kom.

I Putu Dody Lesmana, ST., MT.

Choirul Huda, S.Kom., M.Kom.

**D4 TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI**

POLITEKNIK NEGERI JEMBER

2021

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI
POLITEKNIK NEGERI JEMBER

**LEMBAR PENGESAHAN
WORKSHOP APLIKASI MOBILE**

Penulis 1

**Arvita Agus Kurniasari, S.ST., M.Tr.Kom.
NIP. -**

Penulis 2

Penulis 3

**I Putu Dody Lesmana, ST., MT.
NIP. 19790921 200501 1 001**

**Choirul Huda, S.Kom., M.Kom.
NIP. -**

**Menyetujui,
Ketua Jurusan**

**Mengetahui,
Koordinator Program Studi**

**Hendra Yufit Riskiawan, S.Kom., M.Cs.
NIP. 19830203 200604 1 003**

**Trismayanti Dwi P., S.Kom., M.Cs.
NIP. 19900227 201803 2 001**

KATA PENGANTAR

Puji Syukur kehadirat Tuhan Yang Maha Esa karena atas limpahan rahmat-Nya sehingga kami dapat menyelesaikan Buku Kerja Praktim Mahasiswa (BKPM) Workshop Aplikasi Mobile. Materi Workshop terdiri dari pengenalan github, pengenalan android studio, layout, activity, fragment, intent, manajemen file, shared preferences, GPS dan akses sensor, JSON, pengenalan retrofit, volley, get, **post** dan project final. BKPM ini disusun berdasarkan metode *Student Center Learning* yaitu menempatkan mahasiswa sebagai pusat kegiatan belajar. BKPM ini terdiri dari Pokok Bahasan, Acara Praktikum, Tempat, Alokasi Waktu, Capaian Pembelajaran Mata Kuliah (CPMK), Indikator, Dasar Teori, Alat dan Bahan, Prosedur Kerja, Hasil dan Pembahasan, Kesimpulan, Rubrik Penilaian.

Kami menyadari masih banyak kekurangan dalam penyusunan BKPM ini. Oleh karena itu, kami sangat mengharapkan kritik dan saran demi perbaikan dan kesempurnaan modul ini. Kami mengucapkan terima kasih kepada berbagai pihak yang telah membantu proses penyelesaian BKPM ini. Semoga BKPM ini dapat bermanfaat bagi kita semua.

Jember, 1 September 2021

Tim Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	ii
KATA PENGANTAR	iii
DAFTAR ISI.....	iv
ACARA 1 : PENGENALAN DAN WORKFLOW PADA GIT	1
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	1
b. Indikator.....	1
c. Dasar Teori.....	1
d. Alat dan Bahan.....	2
e. Prosedur Kerja	2
f. Hasil dan Pembahasan	2
g. Kesimpulan	2
h. Rubrik Penilaian.....	2
ACARA 2 : PENGENALAN DAN WORKFLOW PADA GIT	3
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	3
b. Indikator.....	3
c. Dasar Teori.....	3
d. Alat dan Bahan.....	3
e. Prosedur Kerja	3
f. Hasil dan Pembahasan	11
g. Kesimpulan	11
h. Rubrik Penilaian.....	11
ACARA 3 : PENGENALAN DAN WORKFLOW PADA GIT	12
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	12
b. Indikator.....	12
c. Dasar Teori.....	12
d. Alat dan Bahan.....	12
e. Prosedur Kerja	12
f. Hasil dan Pembahasan	14
g. Kesimpulan	14
h. Rubrik Penilaian.....	14

ACARA 4 : PENGENALAN DAN WORKFLOW PADA GIT	15
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	15
b. Indikator.....	15
c. Dasar Teori.....	15
d. Alat dan Bahan.....	15
e. Prosedur Kerja	15
f. Hasil dan Pembahasan	17
g. Kesimpulan	17
h. Rubrik Penilaian.....	17
ACARA 5 : PENGENALAN DAN INSTALASI ANDROID STUDIO	18
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	18
b. Indikator.....	18
c. Dasar Teori.....	18
d. Alat dan Bahan.....	18
e. Prosedur Kerja	18
f. Hasil dan Pembahasan	27
g. Kesimpulan	27
h. Rubrik Penilaian.....	27
ACARA 6 : PENGENALAN DAN INSTALASI ANDROID STUDIO	28
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	28
b. Indikator.....	28
c. Dasar Teori.....	28
d. Alat dan Bahan.....	28
e. Prosedur Kerja	28
f. Hasil dan Pembahasan	28
g. Kesimpulan	28
h. Rubrik Penilaian.....	29
ACARA 7 : PENGENALAN DAN INSTALASI ANDROID STUDIO	30
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	30
b. Indikator.....	30
c. Dasar Teori.....	30
d. Alat dan Bahan.....	33

e. Prosedur Kerja	33
f. Hasil dan Pembahasan	35
g. Kesimpulan	35
h. Rubrik Penilaian.....	36
ACARA 8 : PENGENALAN DAN INSTALASI ANDROID STUDIO	37
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	37
b. Indikator.....	37
c. Dasar Teori.....	37
d. Alat dan Bahan.....	37
e. Prosedur Kerja	37
f. Hasil dan Pembahasan	37
g. Kesimpulan	37
h. Rubrik Penilaian.....	38
ACARA 9 : LAYOUT, MATERIAL DESIGN DAN SCROLL VIEW	39
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	39
b. Indikator.....	39
c. Dasar Teori.....	39
d. Alat dan Bahan.....	41
e. Prosedur Kerja	41
f. Hasil dan Pembahasan	47
g. Kesimpulan	47
h. Rubrik Penilaian.....	48
ACARA 10 : LAYOUT, MATERIAL DESIGN DAN SCROLL VIEW	49
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	49
b. Indikator.....	49
c. Dasar Teori.....	49
d. Alat dan Bahan.....	50
e. Prosedur Kerja	51
f. Hasil dan Pembahasan	57
g. Kesimpulan	57
h. Rubrik Penilaian.....	57
ACARA 11 : LAYOUT, MATERIAL DESIGN DAN SCROLL VIEW	58

a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	58
b.	Indikator.....	58
c.	Dasar Teori.....	58
d.	Alat dan Bahan.....	59
e.	Prosedur Kerja	59
f.	Hasil dan Pembahasan	64
g.	Kesimpulan	65
h.	Rubrik Penilaian.....	65
	ACARA 12 : LAYOUT, MATERIAL DESIGN DAN SCROLL VIEW	66
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	66
b.	Indikator.....	66
c.	Dasar Teori.....	66
d.	Alat dan Bahan.....	67
e.	Prosedur Kerja	67
f.	Hasil dan Pembahasan	72
g.	Kesimpulan	73
h.	Rubrik Penilaian.....	73
	ACARA 13 : ACTIVITY (RECYCLER VIEW).....	74
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	74
b.	Indikator.....	74
c.	Dasar Teori.....	74
d.	Alat dan Bahan.....	75
e.	Prosedur Kerja	75
f.	Hasil dan Pembahasan	75
g.	Kesimpulan	75
h.	Rubrik Penilaian.....	75
	ACARA 14 : ACTIVITY (RECYCLER VIEW).....	76
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	76
b.	Indikator.....	76
c.	Dasar Teori.....	76
d.	Alat dan Bahan.....	76
e.	Prosedur Kerja	76

f.	Hasil dan Pembahasan	80
g.	Kesimpulan	80
h.	Rubrik Penilaian.....	80
	ACARA 15 : ACTIVITY (RECYCLER VIEW).....	81
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	81
b.	Indikator.....	81
c.	Dasar Teori.....	81
d.	Alat dan Bahan.....	81
e.	Prosedur Kerja	82
f.	Hasil dan Pembahasan	87
g.	Kesimpulan	87
h.	Rubrik Penilaian.....	88
	ACARA 16 : ACTIVITY (RECYCLER VIEW).....	89
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	89
b.	Indikator.....	89
c.	Dasar Teori.....	89
d.	Alat dan Bahan.....	89
e.	Prosedur Kerja	89
f.	Hasil dan Pembahasan	89
g.	Kesimpulan	90
h.	Rubrik Penilaian.....	90
	ACARA 17 : FRAGMENT	91
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	91
b.	Indikator.....	91
c.	Dasar Teori.....	91
d.	Alat dan Bahan.....	97
e.	Prosedur Kerja	97
f.	Hasil dan Pembahasan	97
g.	Kesimpulan	97
h.	Rubrik Penilaian.....	97
	ACARA 18 : FRAGMENT	98
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	98

b.	Indikator	98
c.	Dasar Teori.....	98
d.	Alat dan Bahan.....	98
e.	Prosedur Kerja	98
f.	Hasil dan Pembahasan	105
g.	Kesimpulan	105
h.	Rubrik Penilaian.....	105
	ACARA 19 : FRAGMENT	106
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	106
b.	Indikator.....	106
c.	Dasar Teori.....	106
d.	Alat dan Bahan.....	106
e.	Prosedur Kerja	106
f.	Hasil dan Pembahasan	107
g.	Kesimpulan	107
a.	Rubrik Penilaian.....	107
	ACARA 20 : FRAGMENT	108
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	108
b.	Indikator.....	108
c.	Dasar Teori.....	108
d.	Alat dan Bahan.....	108
e.	Prosedur Kerja	108
f.	Hasil dan Pembahasan	108
g.	Kesimpulan	108
h.	Rubrik Penilaian.....	109
	ACARA 21 : INTENT	110
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	110
b.	Indikator.....	110
c.	Dasar Teori.....	110
d.	Alat dan Bahan.....	111
e.	Prosedur Kerja	111
f.	Hasil dan Pembahasan	111

g. Kesimpulan	112
h. Rubrik Penilaian.....	112
ACARA 22 : INTENT	113
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	113
b. Indikator	113
c. Dasar Teori.....	113
d. Alat dan Bahan.....	113
e. Prosedur Kerja	113
f. Hasil dan Pembahasan	115
g. Kesimpulan	115
h. Rubrik Penilaian.....	115
ACARA 23 : INTENT	116
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	116
b. Indikator.....	116
c. Dasar Teori.....	116
d. Alat dan Bahan.....	116
e. Prosedur Kerja	116
f. Hasil dan Pembahasan	120
g. Kesimpulan	120
h. Rubrik Penilaian.....	120
ACARA 24 : INTENT	121
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	121
b. Indikator.....	121
c. Dasar Teori.....	121
d. Alat dan Bahan.....	121
e. Prosedur Kerja	121
f. Hasil dan Pembahasan	121
g. Kesimpulan	122
h. Rubrik Penilaian.....	122
ACARA 25 : MANAJEMEN FILE PADA ANDROID DAN SQLITE	123
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	123
b. Indikator.....	123

c. Dasar Teori.....	123
d. Alat dan Bahan.....	124
e. Prosedur Kerja	124
f. Hasil dan Pembahasan	124
g. Kesimpulan	124
h. Rubrik Penilaian.....	125
ACARA 26 : MANAJEMEN FILE PADA ANDROID DAN SQLITE	126
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	126
b. Indikator	126
c. Dasar Teori.....	126
d. Alat dan Bahan.....	126
e. Prosedur Kerja	126
f. Hasil dan Pembahasan	135
g. Kesimpulan	135
h. Rubrik Penilaian.....	135
ACARA 27 : MANAJEMEN FILE PADA ANDROID DAN SQLITE	136
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	136
b. Indikator	136
c. Dasar Teori.....	136
d. Alat dan Bahan.....	137
e. Prosedur Kerja	138
f. Hasil dan Pembahasan	138
g. Kesimpulan	138
h. Rubrik Penilaian.....	138
ACARA 28 : MANAJEMEN FILE PADA ANDROID DAN SQLITE	139
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	139
b. Indikator	139
c. Dasar Teori.....	139
d. Alat dan Bahan.....	139
e. Prosedur Kerja	139
b. Hasil dan Pembahasan	158
c. Kesimpulan	158

d. Rubrik Penilaian.....	158
ACARA 29 : SHARED PREFERENCE	159
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	159
b. Indikator.....	159
c. Dasar Teori.....	159
d. Alat dan Bahan.....	159
e. Prosedur Kerja	159
f. Hasil dan Pembahasan	160
g. Kesimpulan	160
h. Rubrik Penilaian.....	160
ACARA 30 : SHARED PREFERENCE	161
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	161
b. Indikator.....	161
c. Dasar Teori.....	161
d. Alat dan Bahan.....	161
e. Prosedur Kerja	161
f. Hasil dan Pembahasan	164
g. Kesimpulan	164
h. Rubrik Penilaian.....	164
ACARA 31 : SHARED PREFERENCE	165
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	165
b. Indikator.....	165
c. Dasar Teori.....	165
d. Alat dan Bahan.....	165
e. Prosedur Kerja	165
f. Hasil dan Pembahasan	172
g. Kesimpulan	172
h. Rubrik Penilaian.....	173
ACARA 32 : SHARED PREFERENCE	174
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	174
b. Indikator.....	174
c. Dasar Teori.....	174

d. Alat dan Bahan.....	174
e. Prosedur Kerja	174
f. Hasil dan Pembahasan	174
g. Kesimpulan	175
h. Rubrik Penilaian.....	175
ACARA 33 : GPS DAN AKSES SENSOR	176
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	176
b. Indikator.....	176
c. Dasar Teori.....	176
d. Alat dan Bahan.....	177
e. Prosedur Kerja	177
f. Hasil dan Pembahasan	182
g. Kesimpulan	182
h. Rubrik Penilaian.....	183
ACARA 34 : GPS DAN AKSES SENSOR	184
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	184
b. Indikator.....	184
c. Dasar Teori.....	184
d. Alat dan Bahan.....	185
e. Prosedur Kerja	185
f. Hasil dan Pembahasan	192
g. Kesimpulan	193
h. Rubrik Penilaian.....	193
ACARA 35 : GPS DAN AKSES SENSOR	194
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	194
b. Indikator.....	194
c. Dasar Teori.....	194
d. Alat dan Bahan.....	195
e. Prosedur Kerja	195
f. Hasil dan Pembahasan	200
g. Kesimpulan	200
h. Rubrik Penilaian.....	200

ACARA 36 : GPS DAN AKSES SENSOR	201
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	201
b. Indikator.....	201
c. Dasar Teori.....	201
d. Alat dan Bahan.....	202
e. Prosedur Kerja	202
f. Hasil dan Pembahasan	210
g. Kesimpulan	210
h. Rubrik Penilaian.....	210
ACARA 37 : PENGENALAN JSON	211
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	211
b. Indikator.....	211
c. Dasar Teori.....	211
d. Alat dan Bahan.....	212
e. Prosedur Kerja	212
f. Hasil dan Pembahasan	215
g. Kesimpulan	215
h. Rubrik Penilaian.....	215
ACARA 38 : PENGENALAN JSON	216
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	216
b. Indikator.....	216
c. Dasar Teori.....	216
d. Alat dan Bahan.....	216
e. Prosedur Kerja	216
f. Hasil dan Pembahasan	224
g. Kesimpulan	224
h. Rubrik Penilaian.....	224
ACARA 39 : PENGENALAN JSON	225
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	225
b. Indikator.....	225
c. Dasar Teori.....	225
d. Alat dan Bahan.....	229

e. Prosedur Kerja	229
f. Hasil dan Pembahasan	241
g. Kesimpulan	241
h. Rubrik Penilaian.....	241
ACARA 40 : PENGENALAN JSON	242
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	242
b. Indikator.....	242
c. Dasar Teori.....	242
d. Alat dan Bahan.....	242
e. Prosedur Kerja	242
f. Hasil dan Pembahasan	243
g. Kesimpulan	244
h. Rubrik Penilaian.....	244
ACARA 41 : PENGENALAN RETROFIT, VOLLEY, GET, POST	245
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	245
b. Indikator.....	245
c. Dasar Teori.....	245
d. Alat dan Bahan.....	246
e. Prosedur Kerja	246
f. Hasil dan Pembahasan	262
g. Kesimpulan	262
h. Rubrik Penilaian.....	262
ACARA 42 : PENGENALAN RETROFIT, VOLLEY, GET, POST	263
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	263
b. Indikator.....	263
c. Dasar Teori.....	263
d. Alat dan Bahan.....	264
e. Prosedur Kerja	264
f. Hasil dan Pembahasan	267
g. Kesimpulan	267
h. Rubrik Penilaian.....	267
ACARA 43 : PENGENALAN RETROFIT, VOLLEY, GET, POST	268

a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	268
b.	Indikator.....	268
c.	Dasar Teori.....	268
d.	Alat dan Bahan.....	268
e.	Prosedur Kerja	268
f.	Hasil dan Pembahasan	281
g.	Kesimpulan	281
h.	Rubrik Penilaian.....	281
	ACARA 44 : PENGENALAN RETROFIT, VOLLEY, GET, POST	282
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	282
b.	Indikator.....	282
c.	Dasar Teori.....	282
d.	Alat dan Bahan.....	282
e.	Prosedur Kerja	282
f.	Hasil dan Pembahasan	290
g.	Kesimpulan	290
h.	Rubrik Penilaian.....	291
	ACARA 45 : PROJECT FINAL.....	292
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	292
b.	Indikator.....	292
c.	Dasar Teori.....	292
d.	Alat dan Bahan.....	295
e.	Prosedur Kerja	295
f.	Hasil dan Pembahasan	295
g.	Kesimpulan	296
h.	Rubrik Penilaian.....	296
	ACARA 46 : PROJECT FINAL.....	297
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	297
b.	Indikator.....	297
c.	Dasar Teori.....	297
d.	Alat dan Bahan.....	298
e.	Prosedur Kerja	298

f. Hasil dan Pembahasan	299
g. Kesimpulan	299
h. Rubrik Penilaian.....	299
ACARA 47 : PROJECT FINAL.....	300
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	300
b. Indikator.....	300
c. Dasar Teori.....	300
d. Alat dan Bahan.....	301
e. Prosedur Kerja	302
f. Hasil dan Pembahasan	302
g. Kesimpulan	302
h. Rubrik Penilaian.....	302
ACARA 48 : PROJECT FINAL.....	303
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	303
b. Indikator.....	303
c. Dasar Teori.....	303
d. Alat dan Bahan.....	303
e. Prosedur Kerja	303
f. Hasil dan Pembahasan	303
g. Kesimpulan	303
h. Rubrik Penilaian.....	304
ACARA 49 : PROJECT FINAL.....	305
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	305
b. Indikator.....	305
c. Dasar Teori.....	305
d. Alat dan Bahan.....	307
e. Prosedur Kerja	307
f. Hasil dan Pembahasan	307
g. Kesimpulan	307
h. Rubrik Penilaian.....	307
ACARA 50 : PROJECT FINAL.....	308
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	308

b.	Indikator	308
c.	Dasar Teori.....	308
d.	Alat dan Bahan.....	309
e.	Prosedur Kerja	309
f.	Hasil dan Pembahasan	309
g.	Kesimpulan	309
h.	Rubrik Penilaian.....	309
	ACARA 51 : PROJECT FINAL.....	310
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	310
b.	Indikator.....	310
c.	Dasar Teori.....	310
d.	Alat dan Bahan.....	310
e.	Prosedur Kerja	310
f.	Hasil dan Pembahasan	310
g.	Kesimpulan	311
h.	Rubrik Penilaian.....	311
	ACARA 52 : PROJECT FINAL.....	312
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	312
b.	Indikator.....	312
c.	Dasar Teori.....	312
d.	Alat dan Bahan.....	312
e.	Prosedur Kerja	312
f.	Hasil dan Pembahasan	312
g.	Kesimpulan	312
h.	Rubrik Penilaian.....	313
	ACARA 53 : PROJECT FINAL.....	314
a.	Capaian Pembelajaran Mata Kuliah (CPMK).....	314
b.	Indikator.....	314
c.	Dasar Teori.....	314
d.	Alat dan Bahan.....	314
e.	Prosedur Kerja	314
f.	Hasil dan Pembahasan	314

g. Kesimpulan	314
h. Rubrik Penilaian.....	314
ACARA 54 : PROJECT FINAL.....	315
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	315
b. Indikator	315
c. Dasar Teori.....	315
d. Alat dan Bahan.....	315
e. Prosedur Kerja	315
f. Hasil dan Pembahasan	315
g. Kesimpulan	315
h. Rubrik Penilaian.....	315
ACARA 55 : PROJECT FINAL.....	316
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	316
b. Indikator.....	316
c. Dasar Teori.....	316
d. Alat dan Bahan.....	316
e. Prosedur Kerja	316
f. Hasil dan Pembahasan	316
g. Kesimpulan	316
h. Rubrik Penilaian.....	316
ACARA 54 : PROJECT FINAL.....	317
a. Capaian Pembelajaran Mata Kuliah (CPMK).....	317
b. Indikator.....	317
c. Dasar Teori.....	317
d. Alat dan Bahan.....	317
e. Prosedur Kerja	317
f. Hasil dan Pembahasan	317
g. Kesimpulan	317
h. Rubrik Penilaian.....	317

ACARA 1 : PENGENALAN DAN WORKFLOW PADA GIT

Pokok Bahasan	: Pengenalan Git
Acara Praktikum/Pertemuan	: Minggu 1 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep git
2. Mahasiswa mampu menerapkan git

b. Indikator

Kemampuan mahasiswa dalam memahami konsep dasar git

c. Dasar Teori

Pengenalan Git

Instalasi Git di Windows memang tidak seperti di Linux yang ketik perintah langsung terinstal. Kita harus men-download dulu, kemudian melakukan ritual *next → next → finish*. Tapi dalam ritual tersebut, ada pilihan yang harus diperhatikan agar perintah git dapat dikenali di CMD.

Terdapat tiga kondisi file dalam Git.

1) Modified

Modified adalah kondisi dimana revisi atau perubahan sudah dilakukan, tetapi belum ditandai dan belum disimpan di *version control*. Contohnya pada gambar di atas, ada tiga file HTML yang dalam kondisi *modified*.

2) Staged

Staged adalah kondisi dimana revisi sudah ditandai, tetapi belum disimpan di *version control*. Untuk mengubah kondisi file dari *modified* ke *staged* gunakan perintah git add *nama_file*. Contoh:

```
git add index.html
```

3) Committed

Committed adalah kondisi dimana revisi sudah disimpan di version control. perintah untuk mengubah kondisi file dari *staged* ke *committed* adalah git commit.

d. Alat dan Bahan

1. Kertas folio
2. Spidol

e. Prosedur Kerja

Mahasiswa memperhatikan penjelasan dosen pengampu mata kuliah Workshop Aplikasi Mobile berbasis Android.

1. Mahasiswa menentukan kelompok dengan 4-5 anggota.
2. Membuat resume secara individu tentang konsep dasar dari pengenalan dan workflow git pada kertas folio.
3. Diskusikan dengan kelompok hasil resume yang telah dikerjakan secara individu, dan salin kedalam PPT.
4. Presentasikan hasil membuat resume di depan kelas setiap kelompok secara bergantian.

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan (definisi, konsep dasar pengenalan dan workflow git).
2. Dokumentasi tugas berupa file ppt.

g. Kesimpulan

Mahasiswa memahami mengenai definisi git, konsep dasar pengenalan dan workflow git.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan dari tugas ditunjang dengan bukti referensi	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan dari tugas	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan tugas	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan tugas	(1)	2	3	4
Jumlah skor					

ACARA 2 : PENGENALAN DAN WORKFLOW PADA GIT

Pokok Bahasan	: Instalasi Git
Acara Praktikum/Pertemuan	: Minggu 1 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu instalasi Git pada masing-masing PC
2. Mahasiswa mampu melakukan konfigurasi

b. Indikator

Keberhasilan mahasiswa dalam melakukan tahap-tahap instalasi git, sampai dengan mampu melakukan konfigurasi awal untuk akun git setiap individu.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kertas A4
2. Spidol
3. Jaringan Internet

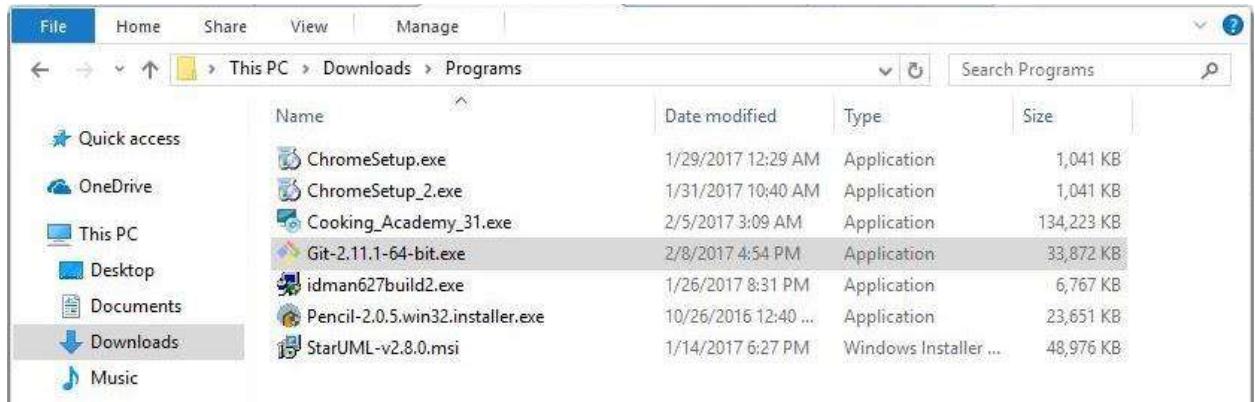
e. Prosedur Kerja

Instalasi Git

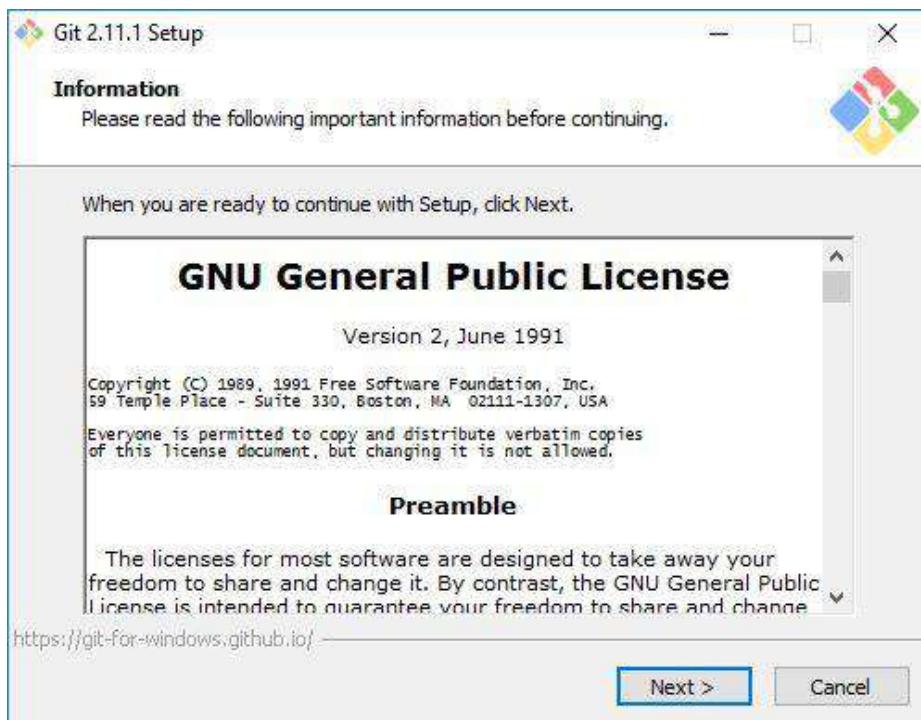
1. Download Git

Silahkan buka website resminya Git (git-scm.com).

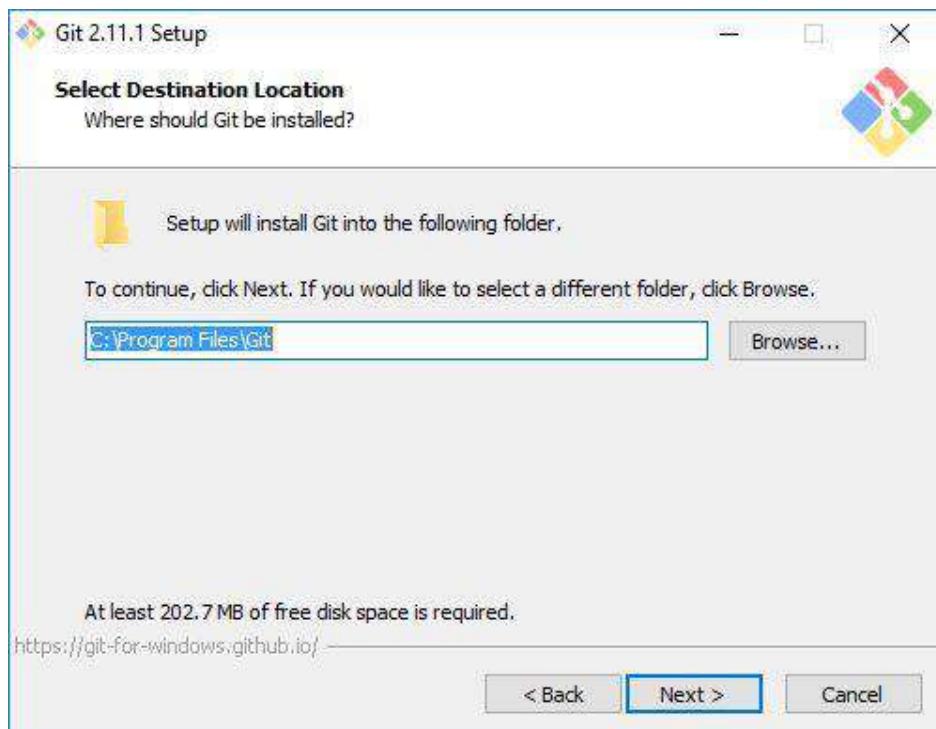
2. Kemudian unduh Git sesuai dengan arsitektur komputer yang digunakan. Kalau menggunakan 64bit, unduh yang 64bit. Begitu juga kalau menggunakan 32bit.
3. Setelah file ter-unduh, silahkan klik 2x file instaler Git yang sudah diunduh.



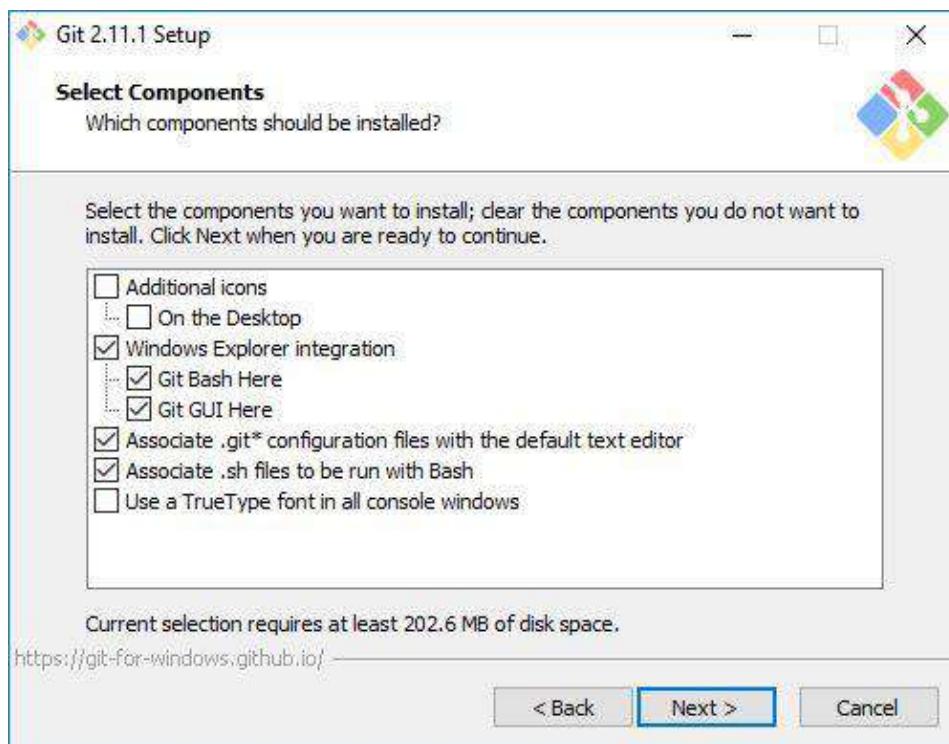
Maka akan muncul infomasi lisensi Git, klik *Next >* untuk melanjutkan.



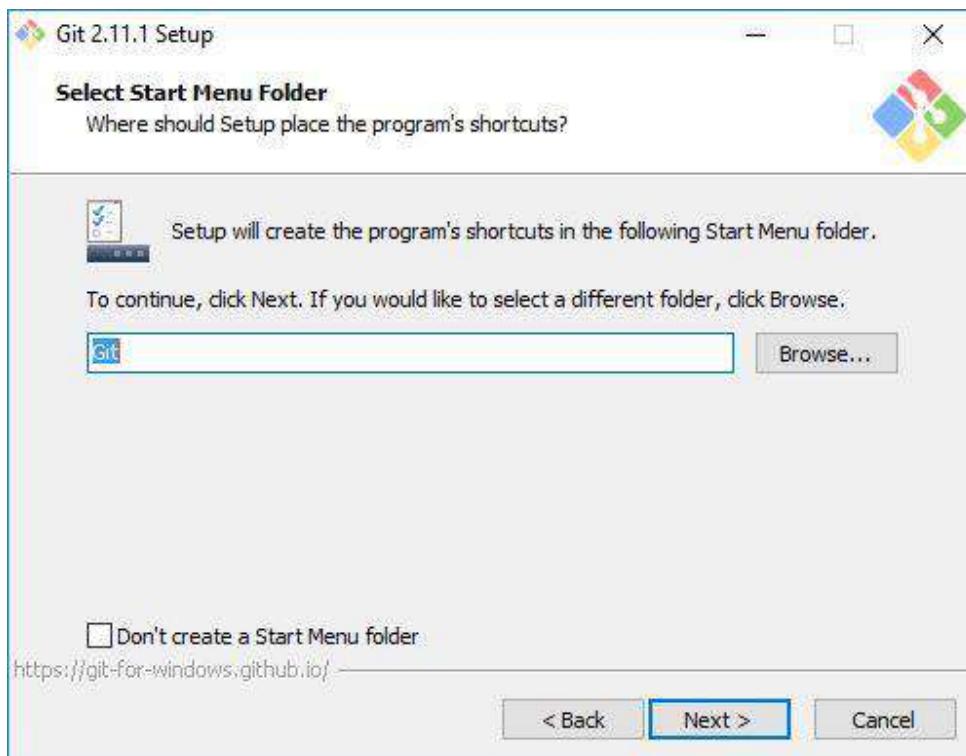
4. Selanjutnya menentukan lokasi instalasi. Biarkan saja apa adanya, kemudian klik *Next >*.



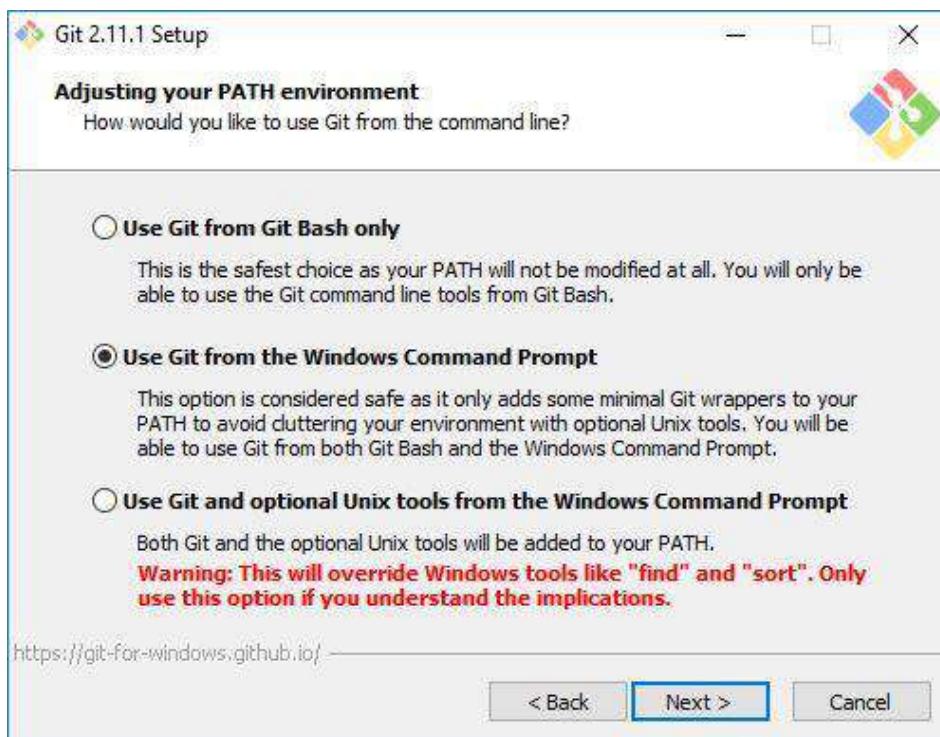
5. Selanjutnya pemilihan komponen, biarkan saja seperti ini kemudian klik *Next >*.



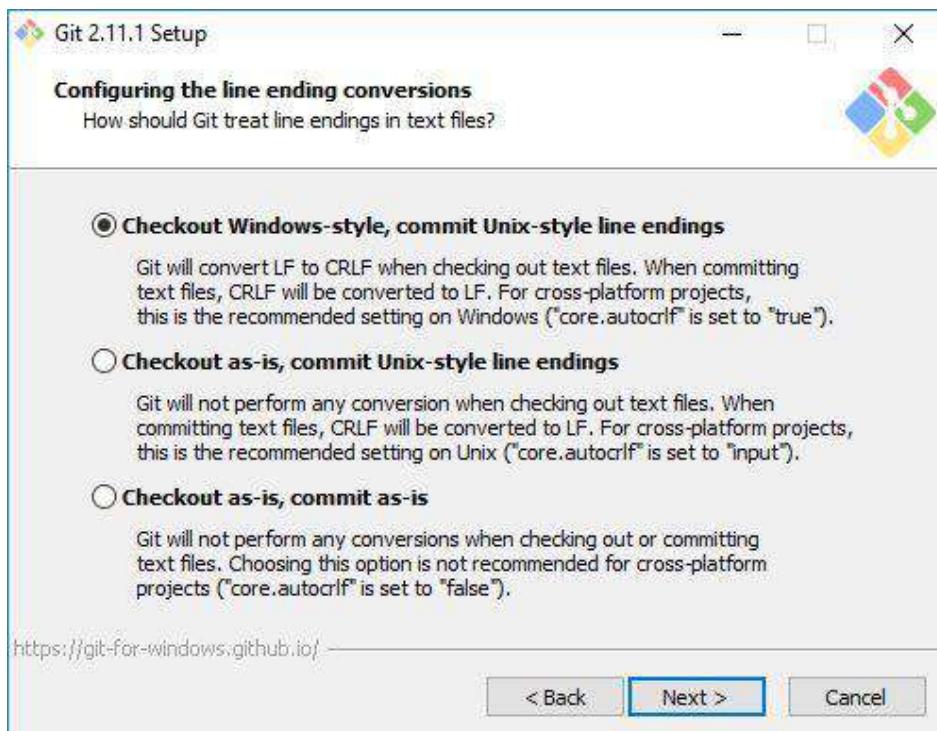
6. Selanjutnya pemilihan direktori start menu, klik *Next >*.



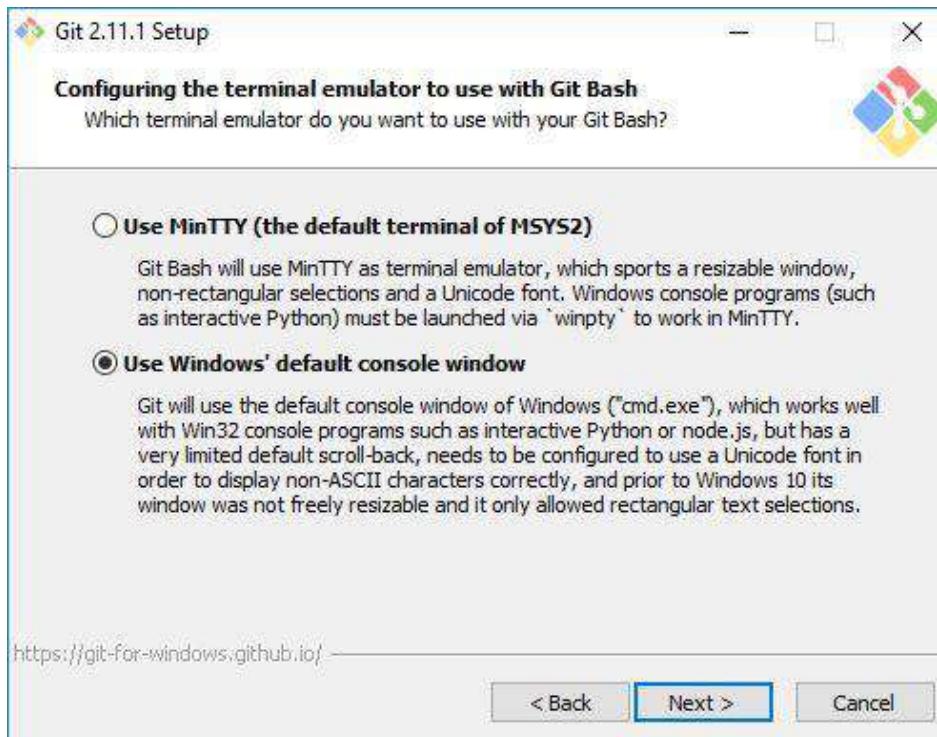
7. Selanjutnya pengaturan *PATH Environment*. Pilih yang tengah agar perintah git dapat di kenali di *Command Prompt* (CMD). Setelah itu klik *Next >*.



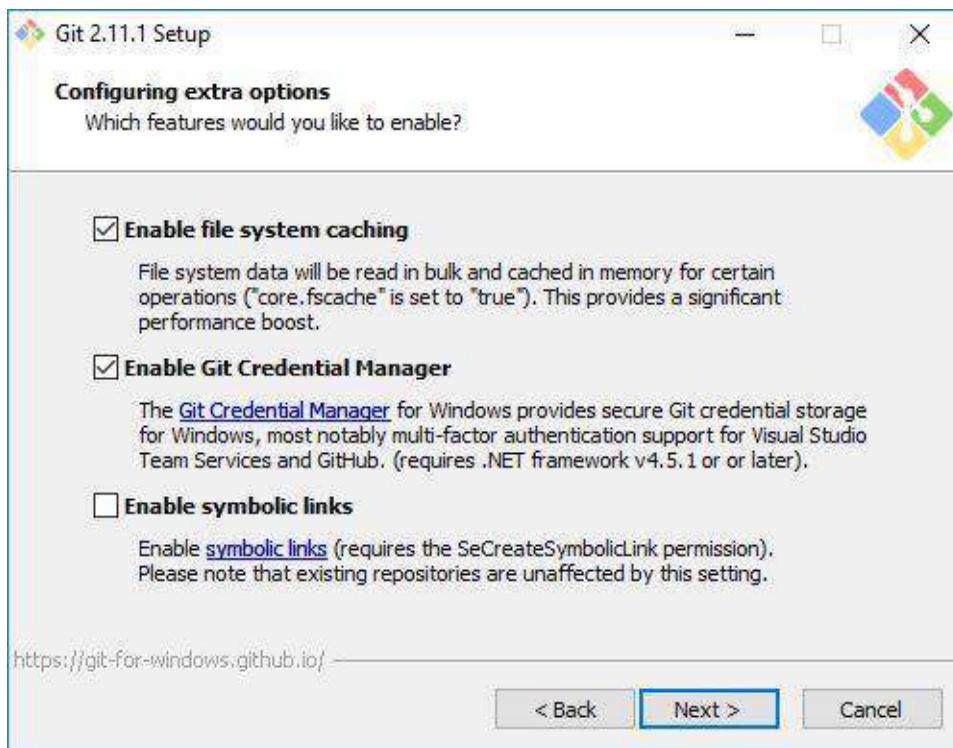
8. Selanjutnya konversi *line ending*. Biarkan saja seperti ini, kemudian klik *Next >*.



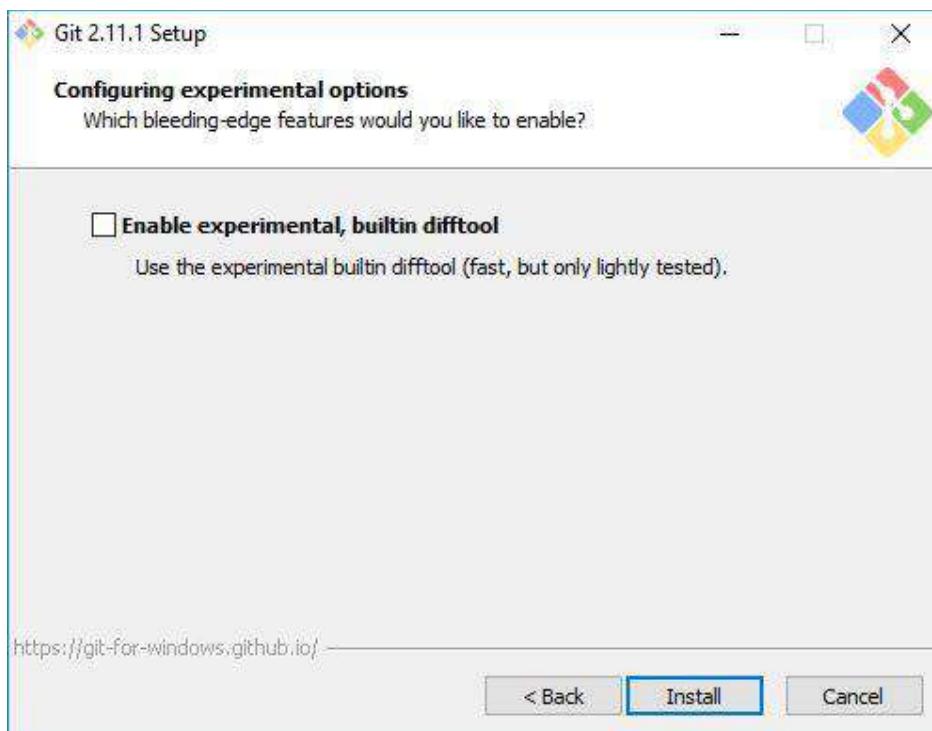
9. Selanjutnya pemilihan emulator terminal. Pilih saja yang bawah, lalu klik *Next >*.



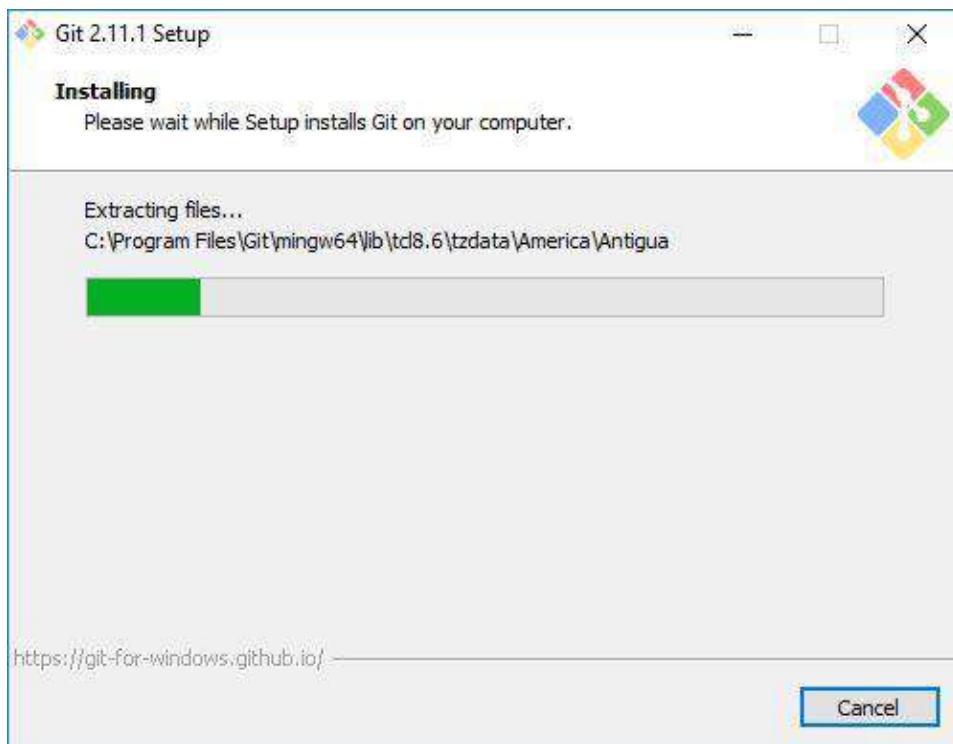
10. Selanjutnya pemilihan opsi ekstra. Klik saja *Next >*.



11. Selanjutnya pemilihan opsi eksperimental, langsung saja klik *Install* untuk instalasi.



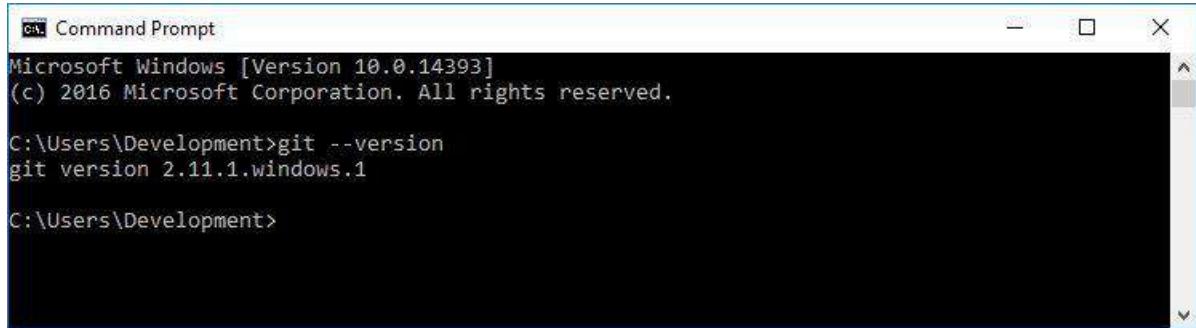
12. Tunggu beberapa saat, instalasi sedang dilakukan.



13. Setelah selesai, kita bisa langsung klik *Finish*.



14. Selamat, Git sudah terinstal di Windows. Untuk mencobanya, silahkan buka CMD atau PowerShell, kemudian ketik perintah git --version.



```
c:\ Command Prompt
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\Development>git --version
git version 2.11.1.windows.1

C:\Users\Development>
```

Konfigurasi Awal

Ada beberapa konfigurasi yang harus dipersiapkan sebelum mulai menggunakan Git, seperti *name* dan *email*.

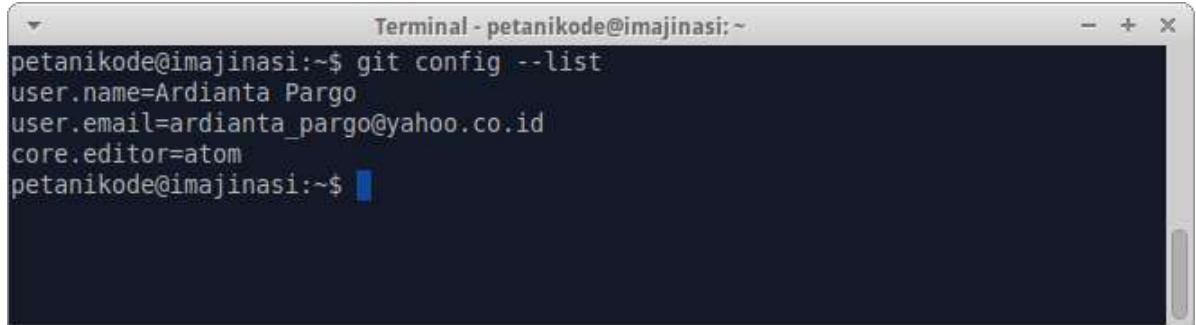
1. Silahkan lakukan konfigurasi dengan perintah berikut ini.

```
git config --global user.name "developer"
git config --global user.email developer@polije.ac.id
```

2. Kemudian periksa konfigurasinya dengan perintah:

```
git config --list
```

Apabila berhasil tampil seperti gambar berikut ini, berarti konfigurasi berhasil.



```
Terminal - petanikode@imajinasi:~
petanikode@imajinasi:~$ git config --list
user.name=Ardianta Pargo
user.email=ardianta_pargo@yahoo.co.id
core.editor=atom
petanikode@imajinasi:~$
```

- Konfigurasi core.editor bersifat opsional, sedangkan *name* dan *email* wajib.

- *Username* dan *email* harus mengikuti akun Github, Gitlab, Bitbucket atau yang lainnya, agar mudah dalam mengintegrasikan.

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan secara individu tahap instalasi git yang di buktikan dengan screenshot pada masing-masing PC pada setiap prosesnya.
2. Dokumentasi hasil konfigurasi berupa akun git pribadi.

g. Kesimpulan

Mahasiswa berhasil instalasi Git pada masing-masing PC serta dilanjutkan melakukan konfigurasi dengan menggunakan akun individu.

h. Rubrik Penilaian

No	Indikator	Skor*			
		1	2	3	4
1	Ketepatan waktu dan ketepatan dalam menjelaskan tahapan instalasi secara berurutan, serta konfigurasi				(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan tahapan instalasi, serta konfigurasi	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan tugas, serta konfigurasi	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan tugas, serta konfigurasi	(1)	2	3	4
Jumlah skor					

ACARA 3 : PENGENALAN DAN WORKFLOW PADA GIT

Pokok Bahasan	: Membuat Revisi Pertama pada Git
Acara Praktikum/Pertemuan	: Minggu 1 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu melakukan revisi pertama
2. Mahasiswa menjelaskan maksud command git revisi pertama

b. Indikator

Keberhasilan mahasiswa dalam melakukan revisi pertama pada git, serta memahami setiap command git revisi pertama.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kertas A4
2. Spidol
3. Jaringan Internet

e. Prosedur Kerja

1. Ubah kondisi tiga file HTML tadi menjadi *staged* dengan perintah git add.

```
git add index.html  
git add about.html  
git add contact.html
```

atau bisa melakukannya seperti ini:

```
git add index.html about.html contact.html
```

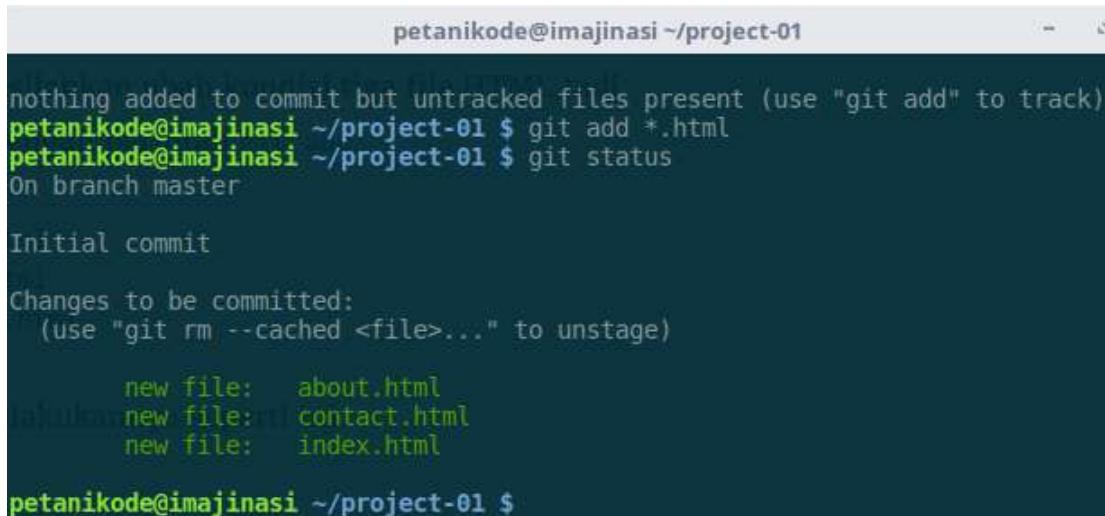
atau:

```
git add *.html
```

atau seperti ini (semua file dan direktori):

```
git add .
```

2. Selanjutnya, cobalah ketik perintah `git status` lagi. Kondisi filenya sekarang akan menjadi *staged*.



```
petanikode@imajinasi ~/project-01
nothing added to commit but untracked files present (use "git add" to track)
petanikode@imajinasi ~/project-01 $ git add *.html
petanikode@imajinasi ~/project-01 $ git status
On branch master
Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

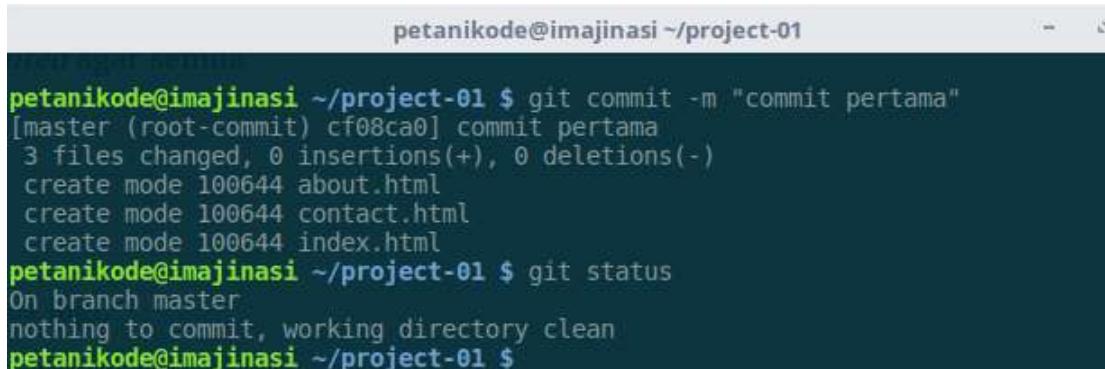
    new file:   about.html
    new file:   contact.html
    new file:   index.html

petanikode@imajinasi ~/project-01 $
```

3. Lalu, ubah kondisi file tersebut ke *committed* agar semua perubahan disimpan oleh Git.

```
git commit -m "Commit pertama"
```

4. Setelah itu, coba cek dengan perintah `git status` lagi.



```
petanikode@imajinasi ~/project-01 $ git commit -m "commit pertama"
[master (root-commit) cf08ca0] commit pertama
 3 files changed, 0 insertions(+), 0 deletions(-)
  create mode 100644 about.html
  create mode 100644 contact.html
  create mode 100644 index.html
petanikode@imajinasi ~/project-01 $ git status
On branch master
nothing to commit, working directory clean
petanikode@imajinasi ~/project-01 $
```

5. Selamat, revisi pertama sudah kita buat. Selanjutnya cobalah untuk membuat revisi kedua.

f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan secara individu tahap revisi pertama pada git yang dibuktikan dengan screenshot pada masing-masing pekerjaannya.
2. Penjelasan pada laporan maksud dan fungsi setiap command yang digunakan .

g. Kesimpulan

Mahasiswa berhasil melakukan revisi pertama dan paham maksud command git yang digunakan pada revisi pertama.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara rinci command git yang digunakan pada revisi pertama	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan command git yang digunakan pada revisi pertama	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan tugas	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan tugas	(1)	2	3	4
Jumlah skor					

ACARA 4 : PENGENALAN DAN WORKFLOW PADA GIT

Pokok Bahasan	: Membuat Revisi Kedua pada Git
Acara Praktikum/Pertemuan	: Minggu 1 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu melakukan revisi kedua
2. Mahasiswa menjelaskan maksud command git revisi kedua

b. Indikator

Keberhasilan mahasiswa dalam melakukan revisi pertama pada git, serta memahami setiap command git revisi kedua.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kertas A4
2. Spidol
3. Jaringan Internet

e. Prosedur Kerja

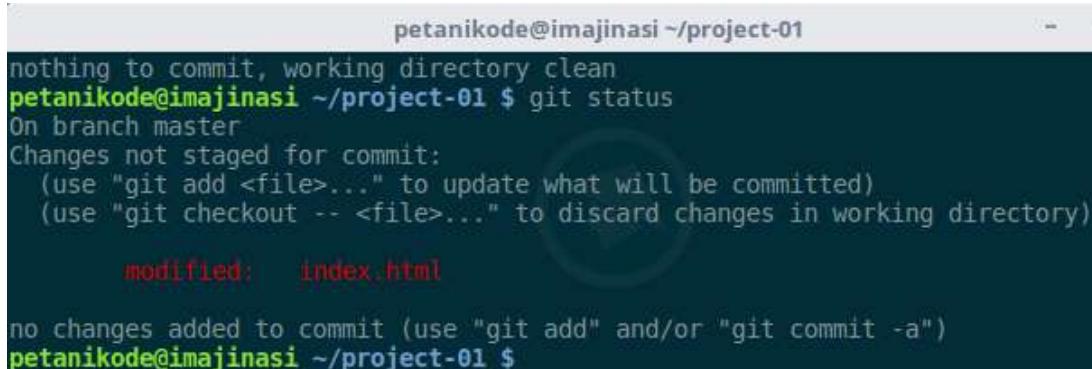
Ceritanya ada perubahan yang akan kita lakukan pada file index.html.

1. Silahkan modifikasi isi file index.html. Sebagai contoh diisikan seperti ini.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Belajar Git - Project 01</title>
  </head>
  <body>
    <p>Hello Semua, Saya sedang belajar Git</p>
```

```
</body>  
</html>
```

2. Setelah itu ketik lagi perintah git status.



```
petanikode@imajinasi ~/project-01  
nothing to commit, working directory clean  
petanikode@imajinasi ~/project-01 $ git status  
On branch master  
Changes not staged for commit:  
  (use "git add <file>..." to update what will be committed)  
  (use "git checkout -- <file>..." to discard changes in working directory)  
  
    modified:   index.html  
  
no changes added to commit (use "git add" and/or "git commit -a")  
petanikode@imajinasi ~/project-01 $
```

Terilhat di sana, file `index.html` sudah dimodifikasi. Kondisinya sekarang berada dalam *modified*.

3. Lakukan *commit* lagi seperti revisi pertama.

```
git add index.html  
git commit -m "ditambahkan isi"
```

- Dengan demikian, revisi kedua sudah disimpan oleh Git.
- Argumen **-m** untuk menambahkan pesan setiap menyimpan revisi.

4. Sekarang Git sudah mencatat dua revisi yang sudah diakukan.

Bisa diibaratkan revisi-revisi ini sebagai *checkpoint* pada Game. Apabila nanti ada kesalahan, kita bisa kembali ke *checkpoint* ini.



f. Hasil dan Pembahasan

1. Dokumentasi tugas berupa laporan secara individu tahap revisi kedua pada git yang dibuktikan dengan screenshot pada masing-masing pekerjaannya.
2. Penjelasan pada laporan maksud dan fungsi setiap command yang digunakan .

g. Kesimpulan

Mahasiswa berhasil melakukan revisi kedua dan paham maksud command git yang digunakan pada revisi pertama.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara rinci command git yang digunakan pada revisi kedua	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan command git yang digunakan pada revisi kedua	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan tugas	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan tugas	(1)	2	3	4
Jumlah skor					

ACARA 5 : PENGENALAN DAN INSTALASI ANDROID STUDIO

Pokok Bahasan	: Instalasi Android Studio
Acara Praktikum/Pertemuan	: Minggu 2 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu melakukan instalasi Android Studio.
2. Mahasiswa mampu setting up gradle pada Android Studio.

b. Indikator

Keberhasilan mahasiswa melakukan instalasi dan setting up gradle pada Android Studio.

c. Dasar Teori

Untuk membuat aplikasi Android, diperlukan software khusus. Salah satu software yang paling populer dan mudah digunakan adalah Android Studio.

Beberapa hal yang perlu diperhatikan sebelum instalasi Android Studio :

- RAM minimal 3GB. Direkomendasikan 8GB RAM.
- Minimal 2 GB penyimpanan yang tersedia. Direkomendasikan 4 GB (500 MB untuk IDE dan 1,5 GB untuk Android SDK dan sistem emulator)
- Resolusi layar minimum 1280 x 800.

d. Alat dan Bahan

1. Laptop minimal RAM 3GB
2. Jaringan Internet

e. Prosedur Kerja

Cara install Android Studio sendiri terdiri dari dua langkah utama, yaitu:

- Setup Android Studio
- Install SDK Android Studio

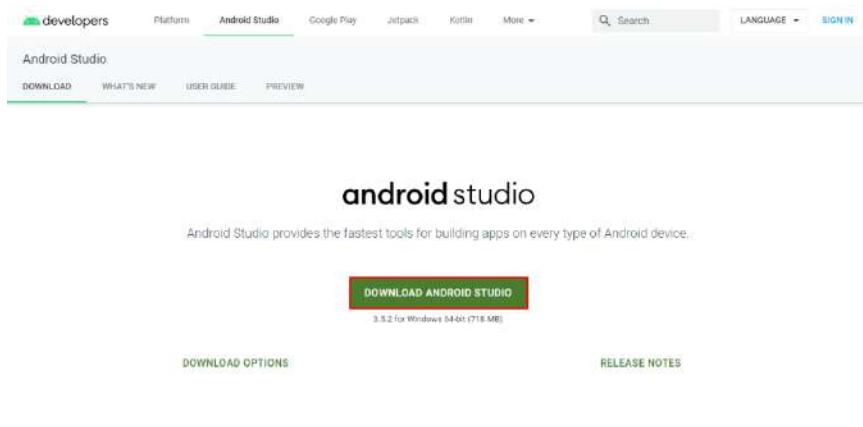
Di bawah ini akan menjelaskan kedua langkah utama tersebut secara lengkap.

1. Setup Android Studio

Berikut ini adalah langkah pertama cara Install Android Studio :

1.1. Download Android Studio

Sebelum melakukan instalasi, hal pertama yang harus dilakukan adalah mengunduh file Android Studio. Buka <https://developer.android.com/studio> untuk download Android Studio



1.2. Install Android Studio

Setelah selesai download file Android Studio, buka file tersebut dan ikuti instruksi instalasi di bawah ini:

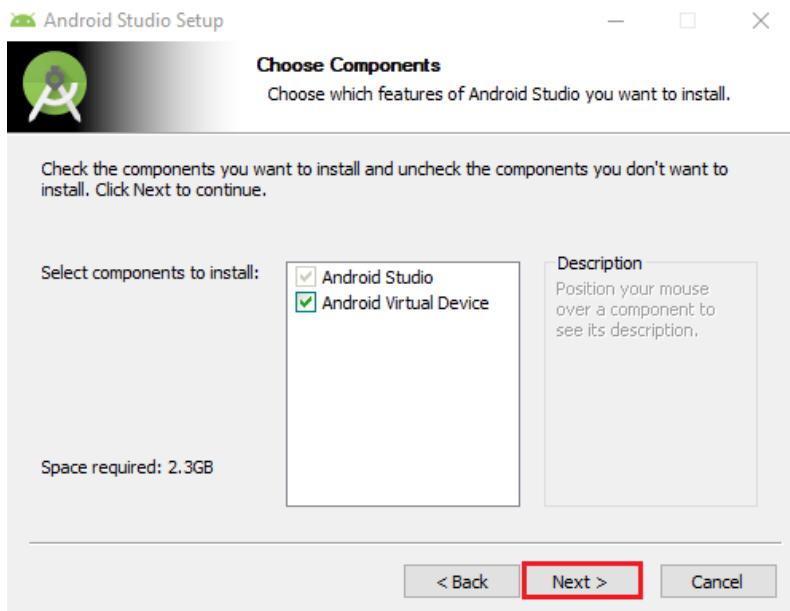
Pertama, akan muncul halaman seperti pada gambar di bawah ini. Klik **Next** untuk melanjutkan ke proses instalasi.



1.3. Pilih Komponen Tambahan

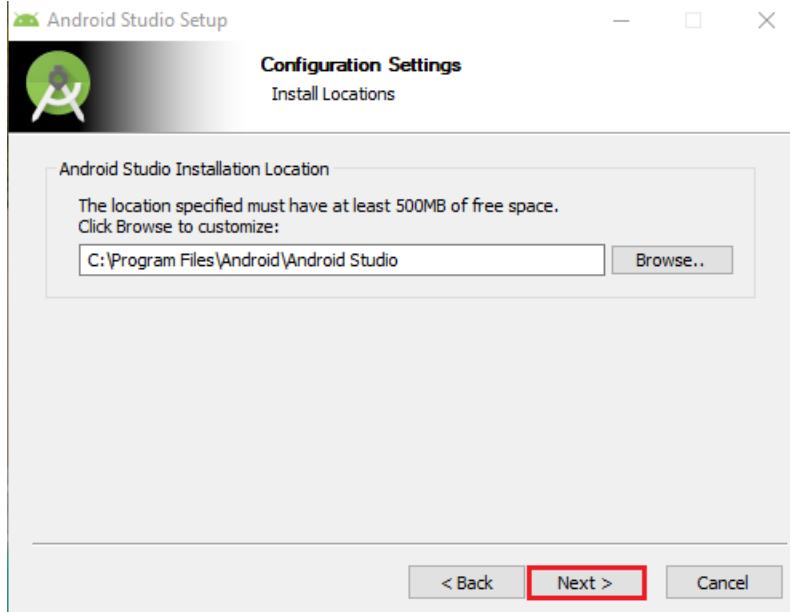
Kemudian pilih komponen tambahan untuk install Android Studio. AVD (Android Virtual Device) ini fungsinya adalah untuk mengkonfigurasi perangkat yang dijalankan dengan

emulator Android. Sesuaikan komponen tambahan yang dipilih seperti pada gambar di bawah ini. Jika sudah klik **Next** untuk melanjutkan instalasi.



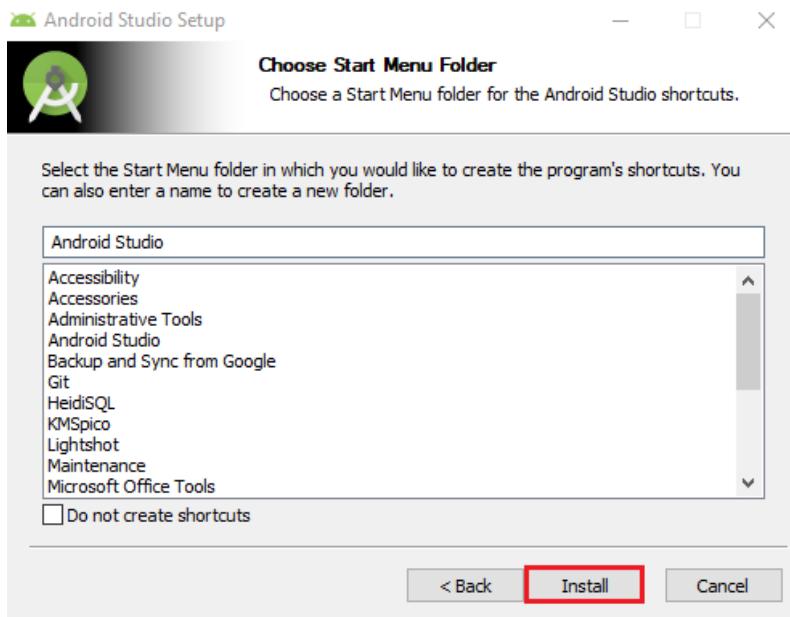
1.4. Tentukan Lokasi Instalasi

Selanjutnya, pilih lokasi untuk install Android Studio pada komputer. Pada langkah menginstall di lokasi C:\Program Files\Android\Android Studio. Setelah menentukan lokasi instalasi Android Studio, klik Next untuk melanjutkan.



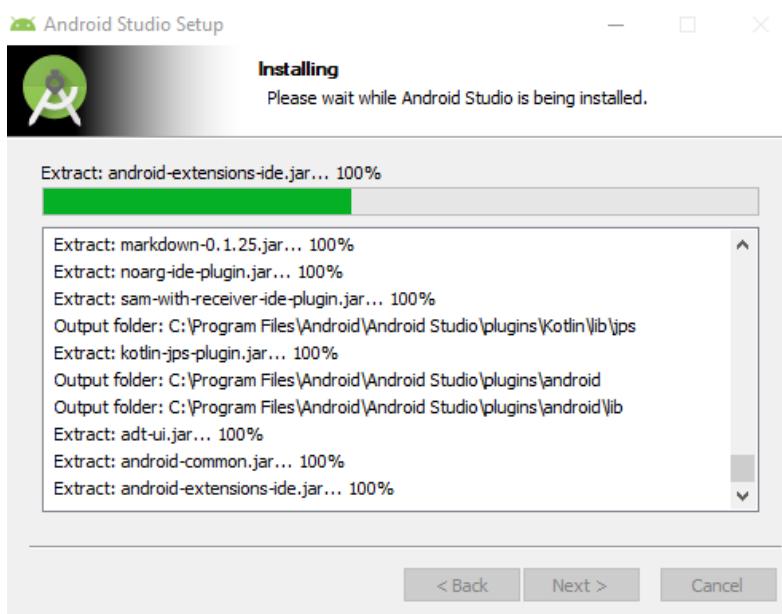
1.5. Tentukan Nama Aplikasi Android Studio

Sebenarnya bebas mengganti nama aplikasi Android Studio yang akan ditampilkan pada Start Menu. Akan tetapi, demi kemudahan saat mencari aplikasi ini, sebaiknya gunakan nama Android Studio saja. Klik Install untuk melanjutkan proses.



1.6. Mulai Proses Instalasi

Setelah menentukan nama aplikasi Android Studio, bisa memulai proses instalasi Android Studio. Gambar di bawah ini adalah proses instalasi Android Studio, tunggu hingga proses Selesai.



2. Install SDK Android Studio

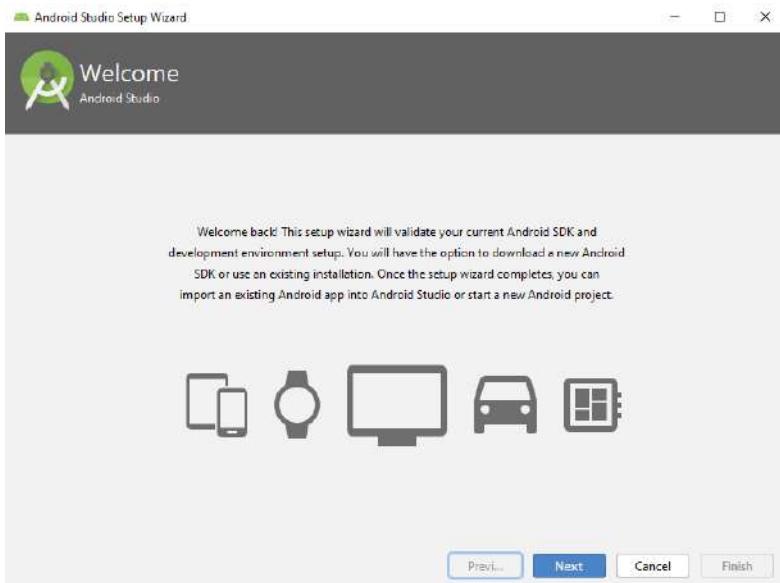
SDK adalah seperangkat alat dan program perangkat lunak yang digunakan oleh pengembang untuk membuat aplikasi untuk platform tertentu.

Sebelum menginstall SDK Android Studio, pastikan perangkat Anda terhubung ke internet. Sebab akan ada proses download untuk komponen-komponen SDK Android Studio. Berikut ini panduan install SDK Android Studio.

2.1. Install SDK Android Studio

Sebelumnya, jika telah berhasil menginstall Android Studio. Buka aplikasi tersebut dan ikuti instruksi instalasi SDK di bawah ini:

Pertama akan muncul halaman seperti pada gambar di bawah ini. Klik **Next** untuk melanjutkan ke proses instalasi.

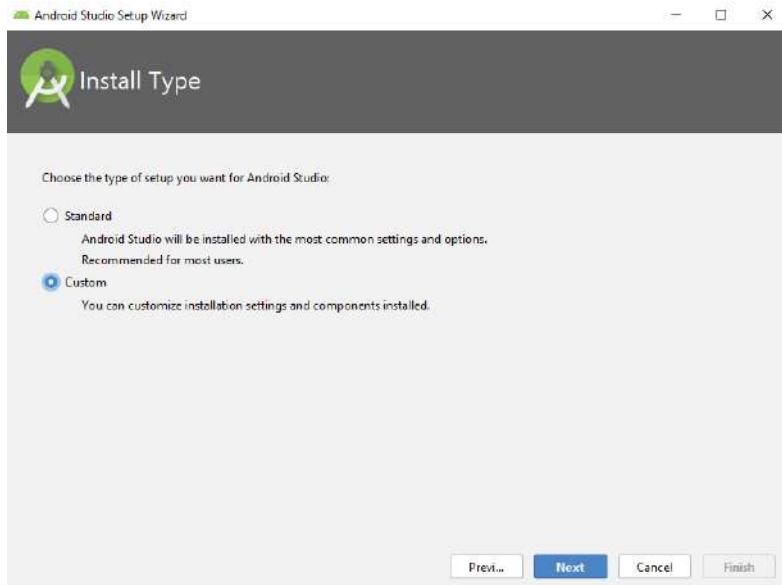


2.2. Pilih Tipe Instalasi

Selanjutnya, pilih tipe instalasi. Anda akan diberikan dua pilihan, yaitu:

- **Standard:** Untuk pilihan standard, Anda akan mendapatkan default pengaturan dan instalasi tambahan dari Android Studio.
- **Custom:** Untuk pilihan custom, Anda bisa memilih pengaturan dan komponen tambahan yang Anda perlukan saja.

Pada tutorial ini kami akan memilih **Custom** agar aplikasi tambahan yang tidak diinginkan tidak terinstall. Klik **Next** untuk melanjutkan instalasi.

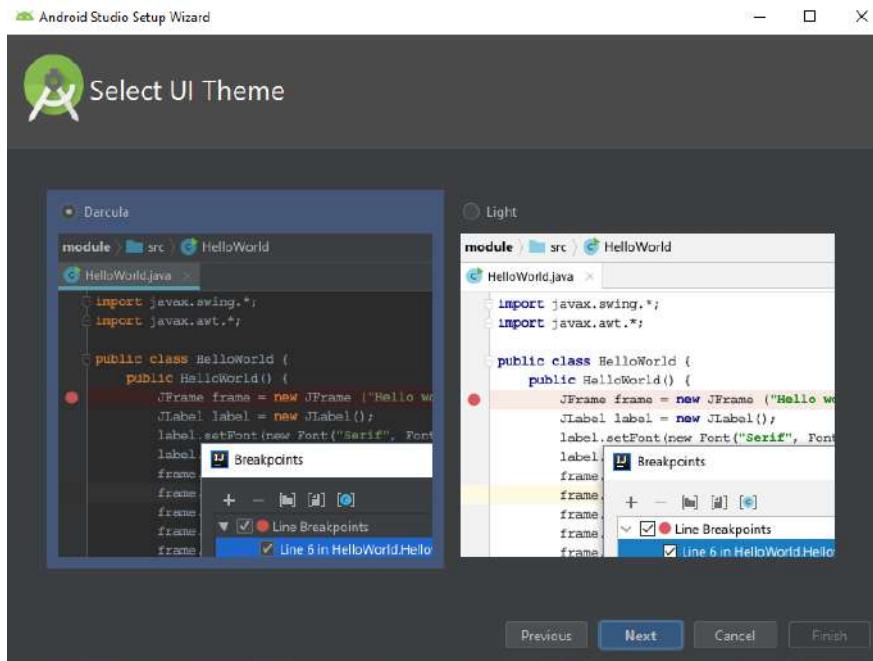


2.3. Pilih Tampilan Android Studio

Kemudian, pilih tema untuk tampilan di dashboard Android Studio. Ada dua pilihan pada tema Android Studio, yaitu:

- **Darcula** : Bertema warna Dark (Hitam)
- **Light** : Bertema warna Light (Putih)

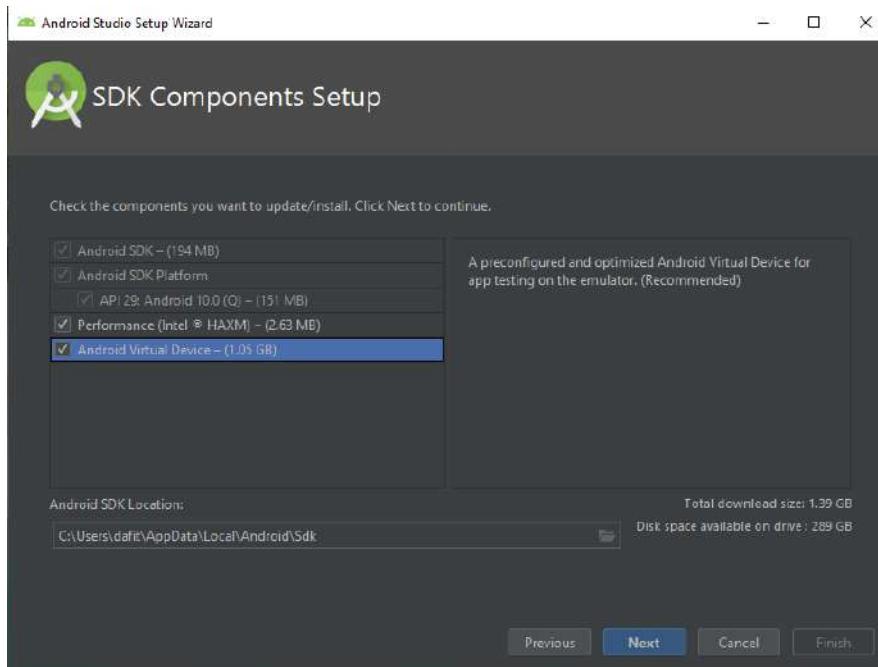
Di tutorial ini kami memilih tema **Darcula**. Lalu klik **Next** untuk melanjutkan.



2.4. Pilih Komponen SDK Tambahan

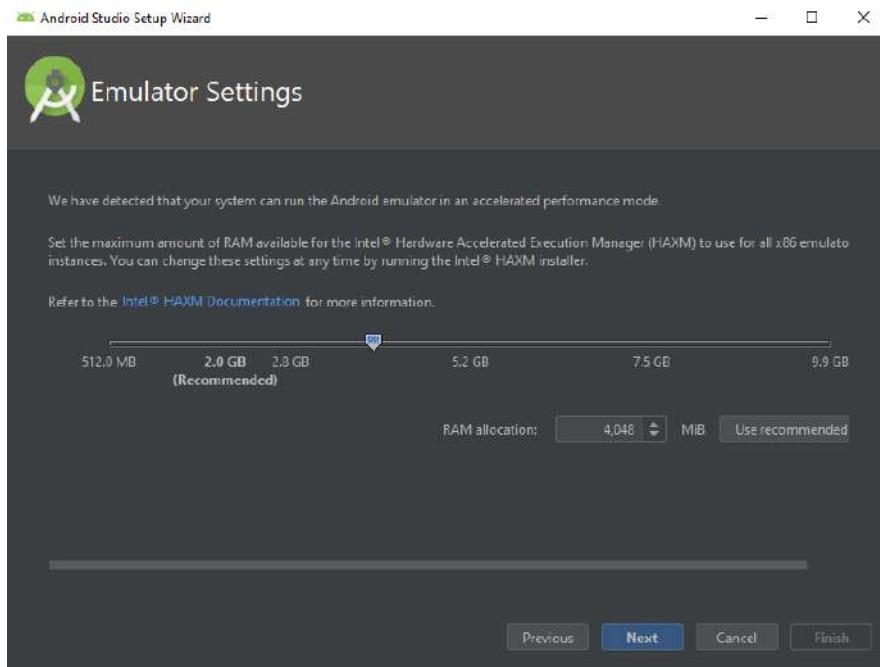
Langkah pemilihan komponen SDK ini hanya akan muncul kalau memilih tipe instalasi **Custom**. Jika memilih tipe instalasi **Standard**, Anda tidak akan mendapatkan pilihan komponen SDK karena semua komponen sudah dipilihkan secara default dari Android Studio.

Sesuaikan komponen tambahan yang dipilih seperti pada gambar di bawah ini. Kemudian klik **Next** untuk melanjutkan instalasi.



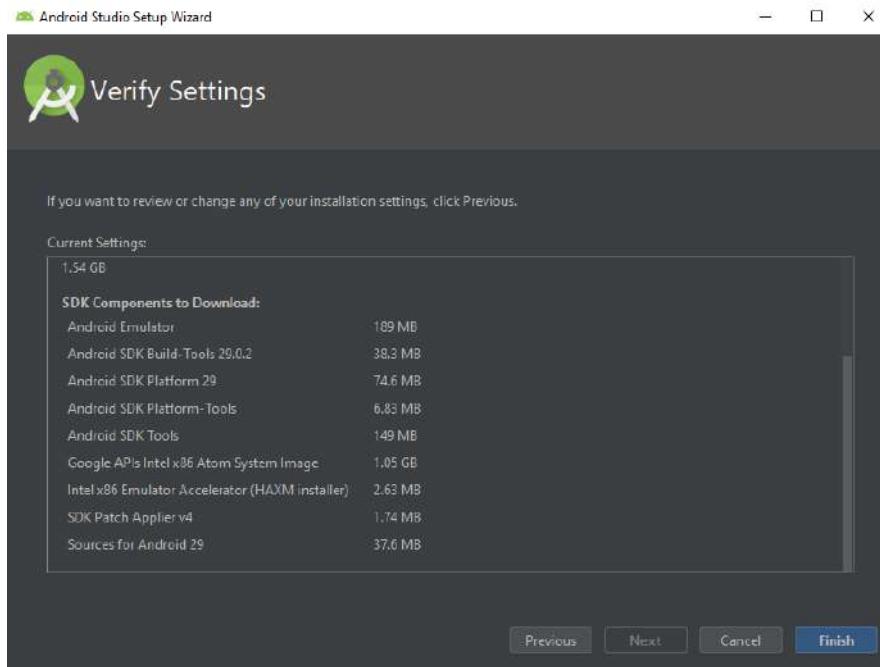
2.5. Tentukan RAM untuk Emulator

Selanjutnya, tentukan RAM. Sebaiknya gunakan RAM minimal 4GB agar tidak memperlambat proses running Emulator ketika menjalankan aplikasi yang telah dibuat. Klik **Next** untuk melanjutkan instalasi.



2.6. Review Pengaturan Instalasi

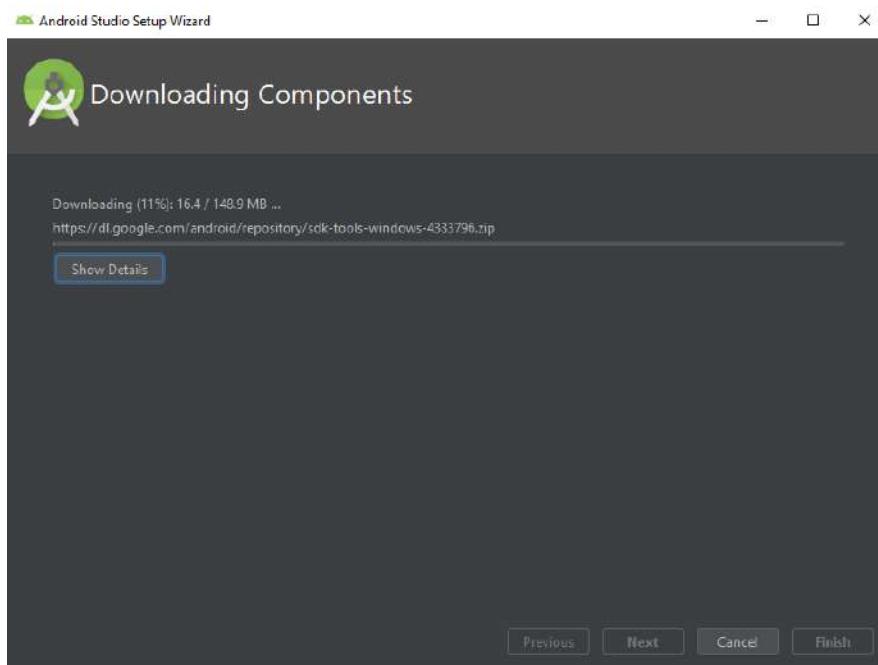
Kemudian akan diberikan informasi mengenai komponen tambahan SDK beserta ukuran filenya. Klik **Finish** untuk memulai proses download komponen-komponen tersebut.



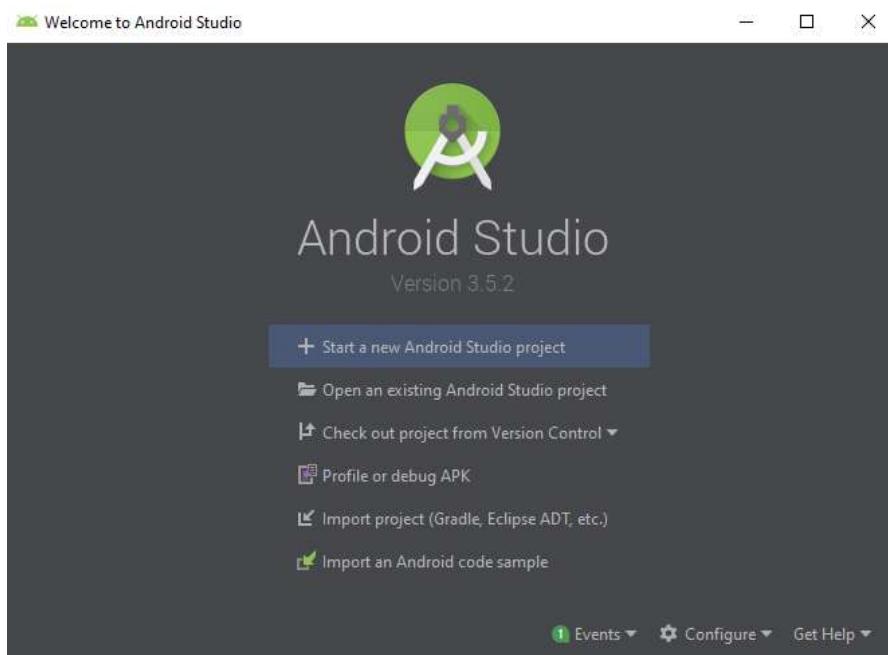
2.7. Proses Download dan Install SDK Android Studio

Setelah proses download komponen SDK dimulai, hanya perlu menunggunya sampai selesai mengunduh.

Proses download dan instalasi file komponen SDK cukup lama karena ukuran filenya yang cukup besar. Oleh karena itu, pastikan koneksi internet stabil agar tidak terjadi error saat proses instalasi berlangsung.



Setelah proses download dan install selesai, akan muncul tampilan seperti gambar di bawah ini.



Selanjutnya bisa mulai untuk membuat Project pertama di Android Studio.

f. Hasil dan Pembahasan

Perangkat lunak Android Studio ter-instal di laptop masing-masing mahasiswa.

g. Kesimpulan

Mahasiswa berhasil instalasi Android Studio.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan menjelaskan detail dalam proses instalasi	1	2	3	(4)
2	Ketepatan waktu dan ketepatan menjelaskan dalam proses instalasi	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat menjelaskan dalam proses instalasi	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan menjelaskan dalam proses instalasi	(1)	2	3	4
Jumlah skor					

ACARA 6 : PENGENALAN DAN INSTALASI ANDROID STUDIO

Pokok Bahasan	: Instalasi Android Studio
Acara Praktikum/Pertemuan	: Minggu 2 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu melakukan instalasi Android Studio.
2. Mahasiswa mampu setting up gradle pada Android Studio.

b. Indikator

Keberhasilan mahasiswa melakukan instalasi dan setting up gradle pada Android Studio.

c. Dasar Teori

-

d. Alat dan Bahan

1. Laptop minimal RAM 3GB
2. Jaringan Internet

e. Prosedur Kerja

Mahasiswa membuat laporan dokumentasi dengan penjelasan sedetail mungkin dalam proses instalasi Android Studio.

f. Hasil dan Pembahasan

Dokumentasi dalam bentuk laporan hasil instalasi Android Studio.

g. Kesimpulan

Mahasiswa berhasil membuat dokumentasi dalam bentuk laporan instalasi Android Studio.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan menjelaskan laporan dokumentasi detail dalam proses instalasi	1	2	3	(4)
2	Ketepatan waktu dan ketepatan menjelaskan laporan dokumentasi dalam proses instalasi	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat menjelaskan laporan dokumentasi dalam proses instalasi	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan menjelaskan laporan dokumentasi dalam proses instalasi	(1)	2	3	4
Jumlah skor					

ACARA 7 : PENGENALAN DAN INSTALASI ANDROID STUDIO

Pokok Bahasan	: Pengenalan Android Studio
Acara Praktikum/Pertemuan	: Minggu 2 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu membuat project pertama di Android Studio.
2. Mahasiswa mampu mengenal beberapa komponen yang ada di Android Studio.

b. Indikator

Keberhasilan mahasiswa membuat project pertama di Android Studio, dan mengenal beberapa komponennya.

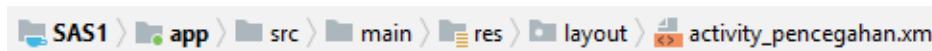
c. Dasar Teori

Tools



Tools merupakan alat-alat yang sering digunakan dalam development seperti copy/paste, build, menjalankan aplikasi, hingga menjalankan emulator.

Navigasi



Membantu melihat struktur dari kedalaman (depth) dan posisi proyek yang sedang kita buka sekarang.

Project Explorer dan Editor



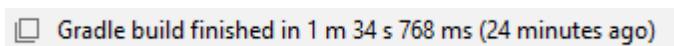
Merupakan bagian utama dari IDE Android Studio di mana kita menuliskan kode. Pada tampilan di atas, sebelah kiri adalah struktur proyek kita dan sebelah kanan adalah editor. Bagian ini akan dibahas lebih detail di poin selanjutnya.

Tool window bar



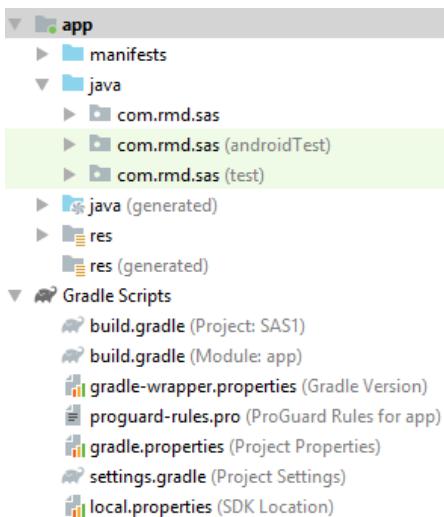
Tools menu yang mengelilingi editor ini merupakan button yang dapat di-expand ataupun untuk menampilkan Tools secara detail dan individual.

Status Bar



Terletak di bagian terbawah Android Studio, berfungsi untuk menampilkan status proyek kita dan pesan peringatan (warning message), apabila ada.

Project Structure

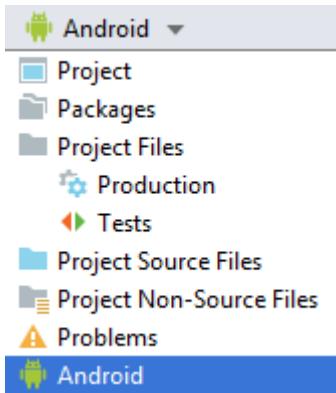


Setiap proyek di Android Studio setidaknya terdiri dari 1 modul atau lebih, dengan source code dan resource-nya. Jenis modul di antaranya:

- Android App Module
- Library Modul
- Google App Engine Module

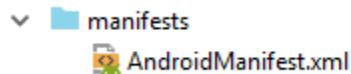
Perhatikan pada Screenshot Project Structure di atas. Root project yang bernama app merupakan Android App Module. Secara default ketika kita membuat proyek baru, Android Studio akan menampilkan struktur yang lebih ringkas dan cepat sesuai dengan kebutuhan

pengembangan Android. Bila ingin melihat struktur proyek dalam bentuk selain str Android, kita dapat mengubahnya melalui tombol dropdown yang terdapat di atas project structure.

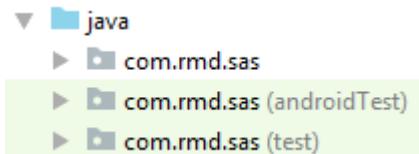


Pada bagian ini kita dapat mengganti tampilan project structure sesuai kebutuhan. Kita bahas lebih detail tentang proyek yang baru saja kita buat.

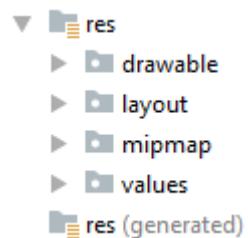
Manifest, Java, Res, dan Gradle



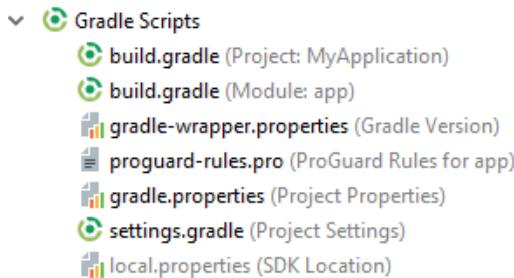
Manifest adalah salah satu berkas yang harus ada di dalam sebuah proyek Android. Manifest akan memberikan beragam informasi penting kepada sistem Android. Sistem perlu mengetahui apa yang akan digunakan oleh aplikasi sebelum dijalankan.



Berisi berkas source code kita yang ditulis dalam bahasa Java, termasuk juga kode Unit Test dan androidTest (Instrumentation Test).



Mengatur resource di dalamnya, yang mana bukan berupa kode, melainkan layout aplikasi, sumber gambar, ikon, hingga style. Di dalam folder res ini juga terdapat sejumlah folder yang sudah diatur dan dikategorikan sesuai kebutuhan.



Secara default Gradle merupakan build tools yang digunakan oleh Android Studio. Fungsinya adalah untuk membantu kita mengkompilasi dan menjalankan source code aplikasi yang kita kembangkan berdasarkan konfigurasi di Gradle. Gradle sendiri juga mendukung manajemen proyek dalam hal penambahan library di luar framework Android.

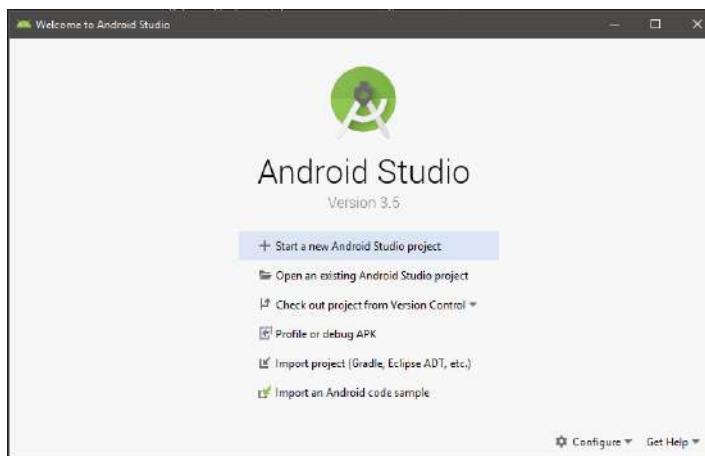
d. Alat dan Bahan

1. Kertas A4
2. Jaringan Internet

e. Prosedur Kerja

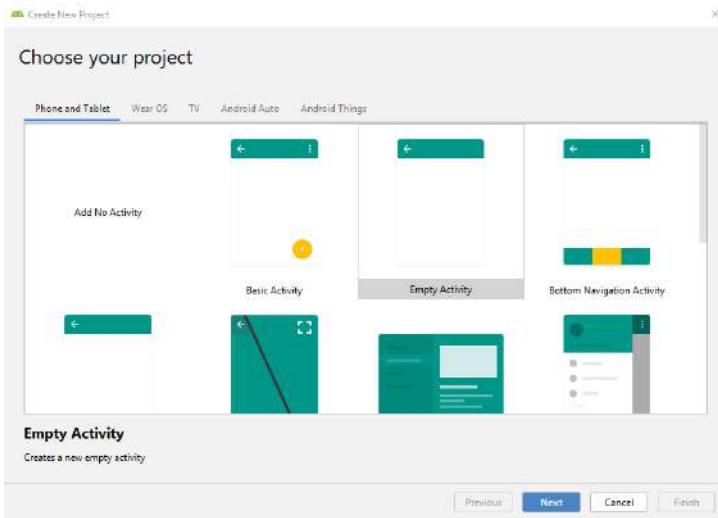
Android Studio

1. Kali pertama menjalankan Android Studio, akan melihat tampilan seperti berikut ini.
2. Untuk mulai proyek baru pilihlah “Start a new Android Project”.

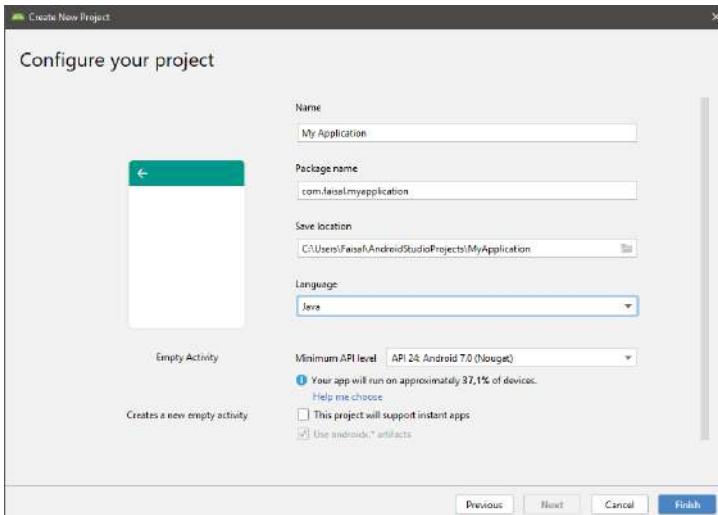


3. Pada halaman Project Wizard ini pengguna diminta untuk memilih jenis activity, pemilihan activity ini menyesuaikan dengan tujuan dari pembuatan project, pilih “empty activity” apabila akan melakukan costum activity.
4. Selain itu adalah target devices, dimana bisa memilih peranti target dari aplikasi yang akan dibuat pada tampilan di bawah terlihat ada jenis phone and tablet, wear OS, TV, Android Auto dan yang terakhir Android Things.

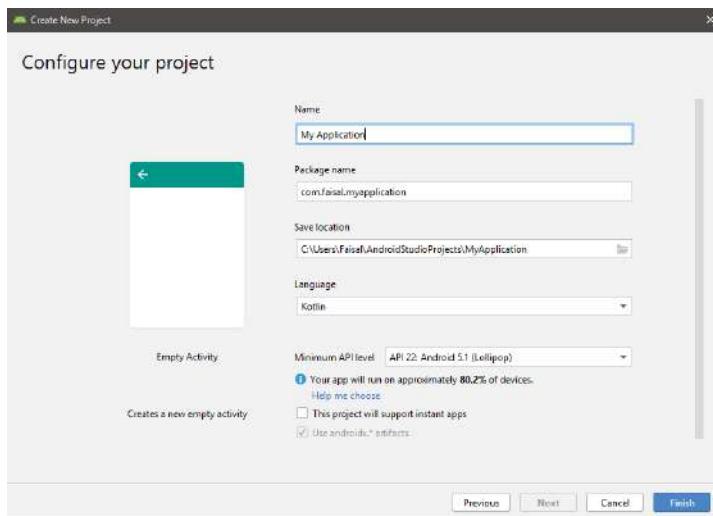
5. Dialog di bawah ini adalah default template. Di dalamnya terdapat beberapa template yang bisa digunakan seperti Empty Activity, Login Activity, Navigation Drawer Activity dan lain-lain.



6. Dalam dialog ini bisa memberi nama aplikasi yang hendak dibuat, dan company domain.
7. Company domain akan digunakan sebagai alat identifikasi ketika aplikasi akan dipublikasikan. Kita juga dapat mengganti lokasi di mana proyek akan disimpan.

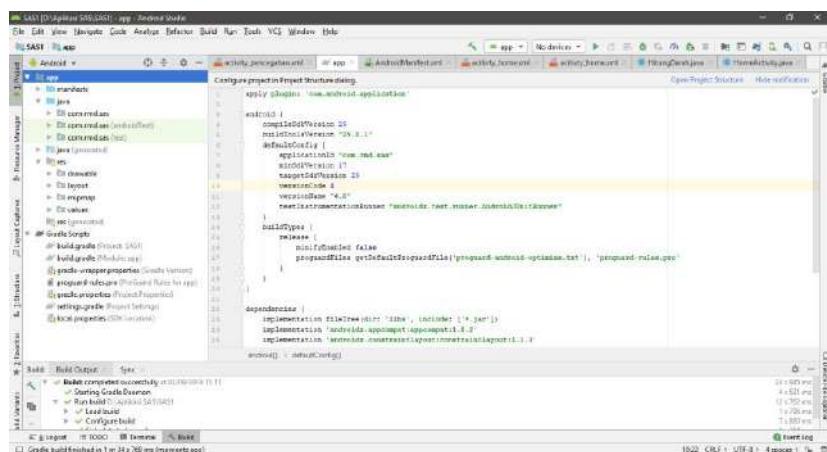


8. Dialog berikut ini adalah penamaan activity yang pertama kali dibuat.
9. Usai memberi nama, tekan Finish.



Antarmuka Android Studio

Membuat proyek pertama kali di Project Wizard, telah selesai. Selanjutnya akan menemui tampilan penuh Android Studio. Untuk meningkatkan produktivitas, mari kita bahas lebih jauh tentang antarmuka (interface) dari Android Studio ini.



Di atas adalah screenshot tampilan penuh IDE Android Studio berbasis IntelliJ IDEA. Mungkin tampilan tersebut akan berbeda dengan tampilan di layar karena perbedaan konfigurasi dan versi Android Studio.

f. Hasil dan Pembahasan

Dokumen project android studio yang telah dibuat.

g. Kesimpulan

Mahasiswa berhasil membuat project pertama pada Android Studio.

h. Rubrik Penilaian

No	Indikator	Skor*				
1	Ketepatan waktu dan ketepatan menjelaskan detail dalam proses pembuatan project	1	2	3	(4)	
2	Ketepatan waktu dan ketepatan menjelaskan dalam proses pembuatan project	1	2	(3)	4	
3	Ketepatan waktu akan tetapi kurang tepat menjelaskan dalam proses pembuatan project	1	(2)	3	4	
4	Keterlambatan pengumpulan tugas dan ketidaktepatan menjelaskan dalam proses pembuatan project	(1)	2	3	4	
Jumlah skor						

ACARA 8 : PENGENALAN DAN INSTALASI ANDROID STUDIO

Pokok Bahasan	: Pengenalan Android Studio
Acara Praktikum/Pertemuan	: Minggu 2 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu membuat project pertama di Android Studio.
2. Mahasiswa mampu mengenal beberapa komponen yang ada di Android Studio.

b. Indikator

Keberhasilan mahasiswa membuat project pertama di Android Studio, dan mengenal beberapa komponennya.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kertas A4
2. Jaringan Internet

e. Prosedur Kerja

Mahasiswa membuat laporan dokumentasi dengan penjelasan dalam pembuatan project pertama pada Android Studio.

f. Hasil dan Pembahasan

Dokumentasi dalam bentuk laporan hasil pembuatan project pertama pada Android Studio.

g. Kesimpulan

Mahasiswa berhasil membuat dokumentasi dalam bentuk laporan pembuatan project pertama pada Android Studio.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan menjelaskan laporan dokumentasi detail dalam pembuatan project	1	2	3	(4)
2	Ketepatan waktu dan ketepatan menjelaskan laporan dokumentasi dalam pembuatan project	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat menjelaskan laporan dokumentasi dalam pembuatan project	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan menjelaskan laporan dokumentasi dalam pembuatan project	(1)	2	3	4
Jumlah skor					

ACARA 9 : LAYOUT, MATERIAL DESIGN DAN SCROLL VIEW

Pokok Bahasan	: Linear Layout dan Relative Layout
Acara Praktikum/Pertemuan	: Minggu 3 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep linear layout.
2. Mahasiswa mampu memahami konsep relative layout.
3. Mahasiswa mampu menerapkan konsep linear layout.
4. Mahasiswa mampu menerapkan konsep relative layout.

b. Indikator

1. Keberhasilan mahasiswa dalam menerapkan linear layout pada studi kasus.
2. Keberhasilan mahasiswa dalam menerapkan relative layout pada studi kasus.

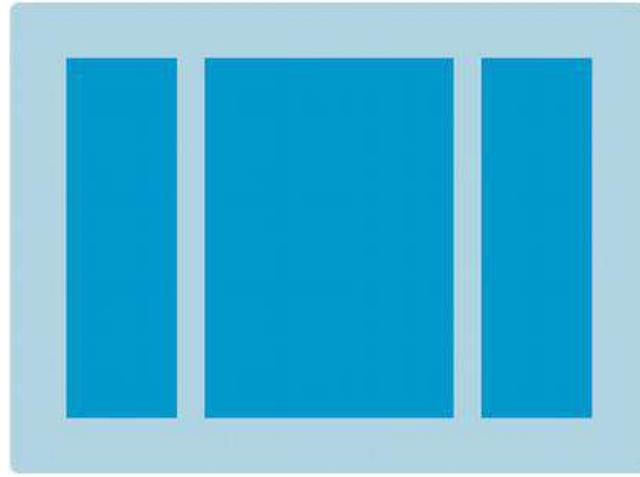
c. Dasar Teori

Layout

Layout merupakan suatu tampilan tata letak di Android untuk mengatur penempatan teks, gambar, ataupun komponen lainnya sehingga tampilan pada aplikasi yang dibuat terlihat rapih dan nyaman untuk dilihat oleh pengguna.

1) Liniear Layout

Linear Layout adalah layout yang menyajarkan semua *child view*-nya dalam satu arah, secara vertikal atau horizontal. Anda bisa menetapkan arah layout dengan atribut **android:orientation**.



Gambar 1. Linier Layout

Semua anak LinearLayout akan ditumpuk satu sama lain, jadi daftar vertikal hanya akan memiliki satu anak per baris, seberapa pun lebarnya, dan daftar horizontal hanya akan setinggi satu baris (tinggi anak yang tertinggi, ditambah pengisi). LinearLayout akan mengikuti *margin* antara anak dan *gravity* (sejajar kanan, tengah, atau kiri) setiap anak.

2) Relative Layout

Relative Layout adalah layout yang penataannya ini adalah penataan yang menempatkan widget-widget didalamnya seperti layer, sehingga sebuah widget dapat berada di atas/di bawah widget lainnya atau dengan kata lain Relative merupakan layout yang penataannya lebih bebas (Relative) sehingga bisa di tata di mana saja.



Gambar 2. Relative Layout

Jadi dapat diartikan bahwa **Relative Layout** adalah desain tampilan pada aplikasi dengan tata letak objek atau komponen secara bebas tanpa aturan sesuai orientasi (horizontal atau vertical) seperti yang diterapkan pada **Linear Layout**.

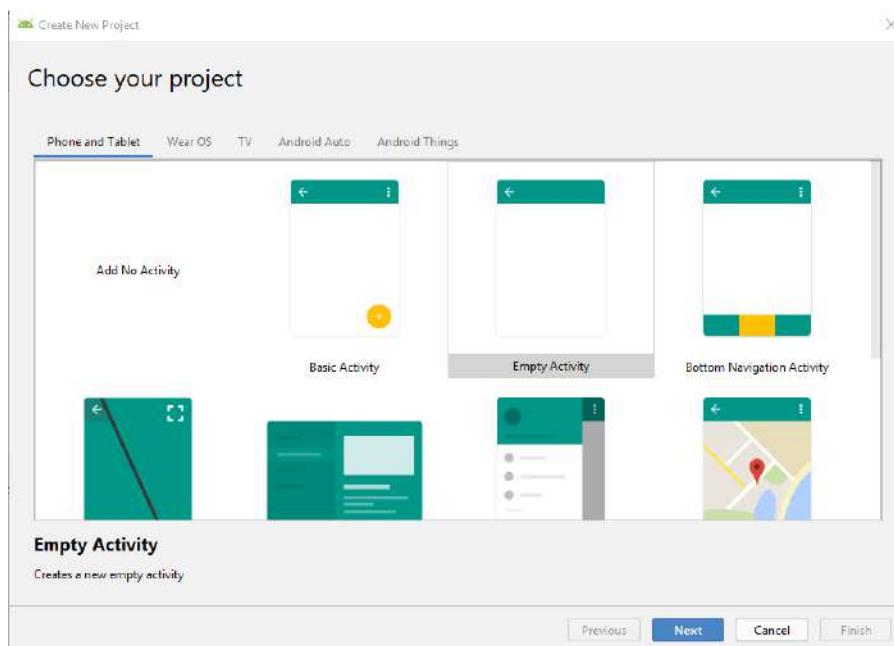
d. Alat dan Bahan

1. Kertas A4
2. Kabel Data USB
3. Jaringan Internet

e. Prosedur Kerja

Linear Layout

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project Linear Layout
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**



5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Implementasikan code dibawah ini :

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:paddingLeft="16dp"
8      android:paddingRight="16dp"
9      android:orientation="vertical"
10     tools:context=".LinierLayout">
11     <EditText
12         android:id="@+id/to"
13         android:layout_width="match_parent"
14         android:layout_height="wrap_content"
15         android:hint="to"/>
16     <EditText
17         android:id="@+id/subject"
18         android:layout_width="match_parent"
19         android:layout_height="wrap_content"
20         android:hint="subject" />
21     <EditText
22         android:layout_width="match_parent"
23         android:layout_height="0dp"
24         android:layout_weight="1"
25         android:gravity="top"
26         android:id="@+id/message"
27         android:hint="message" />
28     <Button
29         android:id="@+id/send"
30         android:layout_width="122dp"
31         android:layout_height="69dp"
32         android:layout_gravity="right"
33         android:text="send"
34         tools:ignore="RtlHardcoded" />
35 </LinearLayout>
```

7. Running hasil code diatas, dan akan menampilkan output seperti berikut:



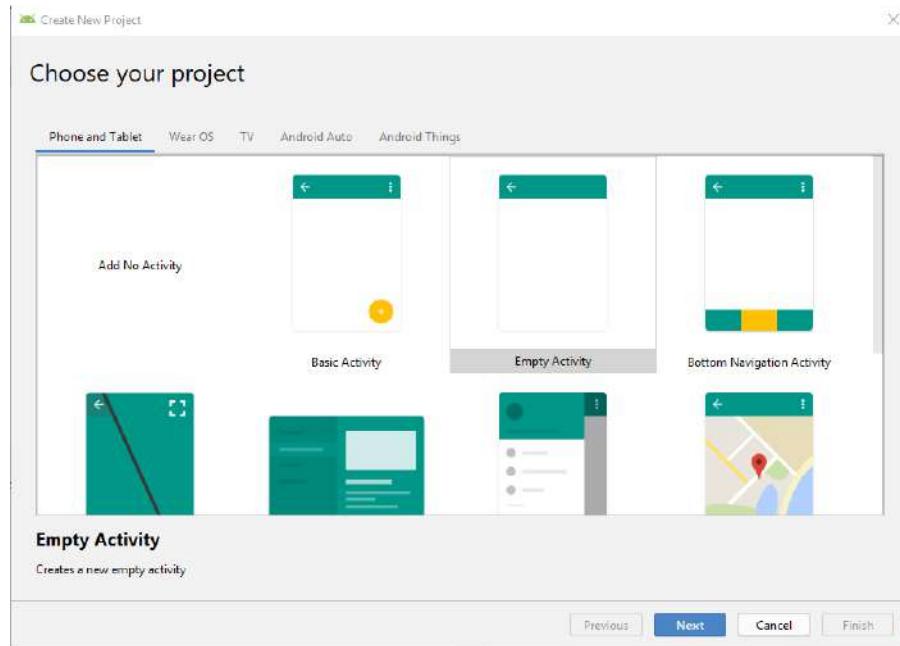
Gambar 3. Hasil Linear Layout

8. Pada **LinierActivity.java** tambahkan code berikut :

```
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.os.Bundle;
6 import android.widget.Button;
7 import android.widget.EditText;
8
9 public class LinierLayout extends AppCompatActivity {
10     EditText to, subject, message;
11     Button send;
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.linier_layout);
17
18         to = findViewById(R.id.to);
19         subject = findViewById(R.id.subject);
20         message = findViewById(R.id.message);
21         send = findViewById(R.id.send);
22     }
}
```

Relative Layout

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project Relative Layout
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**



5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Implementasikan code dibawah ini :

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent" android:layout_height="match_parent">
4     <EditText
5         android:id="@+id/editText1"
6         android:layout_width="wrap_content"
7         android:layout_height="wrap_content"
8         android:layout_marginLeft="100dp"
9         android:layout_marginTop="150dp"
10        android:ems="10"
11        android:hint="Enter Date" />
```

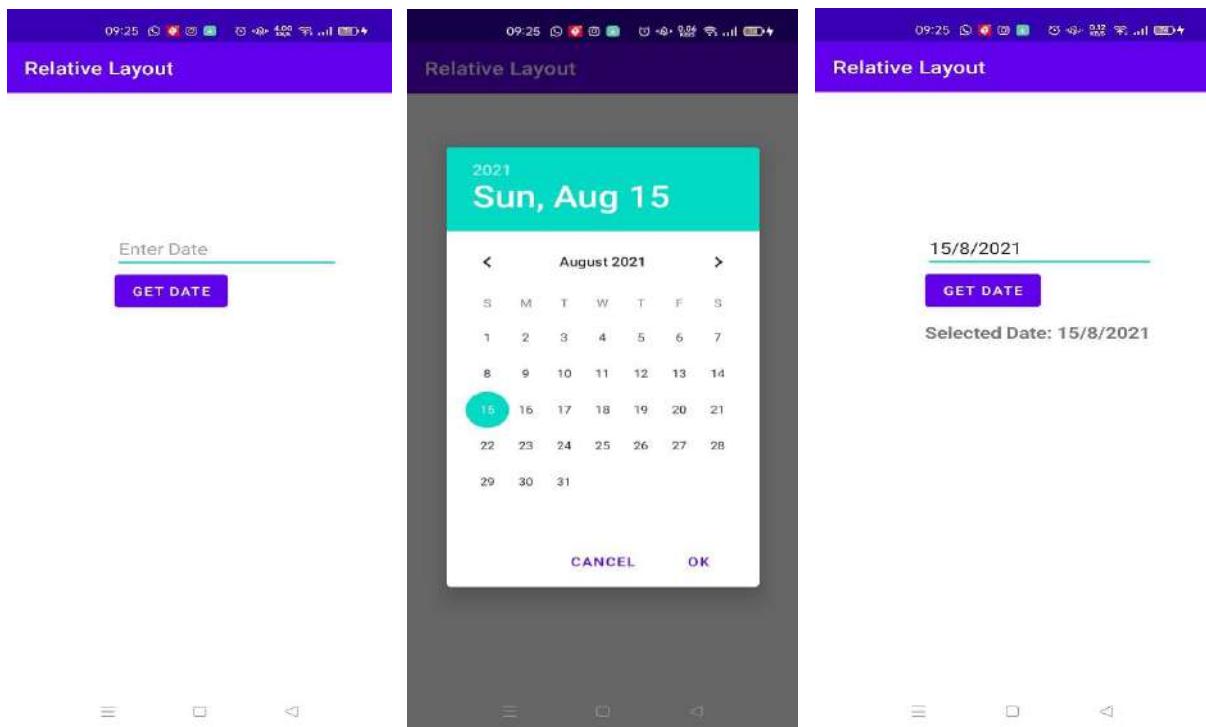
```
12     <Button  
13         android:id="@+id/button1"  
14         android:layout_width="wrap_content"  
15         android:layout_height="wrap_content"  
16         android:layout_below="@+id/editText1"  
17         android:layout_marginLeft="100dp"  
18         android:text="Get Date" />  
19     <TextView  
20         android:id="@+id/textView1"  
21         android:layout_width="wrap_content"  
22         android:layout_height="wrap_content"  
23         android:layout_below="@+id/button1"  
24         android:layout_marginLeft="100dp"  
25         android:layout_marginTop="10dp"  
26             android:textStyle="bold"  
27             android:textSize="18dp"/>  
28     </RelativeLayout  
29 
```

7. Pada **RelativeLayout.java** tambahkan code berikut :

```
3     import androidx.appcompat.app.AppCompatActivity;  
4  
5     import android.app.DatePickerDialog;  
6     import android.os.Bundle;  
7     import android.text.InputType;  
8     import android.view.View;  
9     import android.widget.Button;  
10    import android.widget.DatePicker;  
11    import android.widget.EditText;  
12    import android.widget.TextView;  
13    import java.util.Calendar;  
14  
15    public class RelativeLayout extends AppCompatActivity {  
16        DatePickerDialog picker;  
17        EditText eText;  
18        Button btnGet;  
19        TextView tvw;  
20        @Override  
21    } 
```

```
21.  protected void onCreate(Bundle savedInstanceState) {  
22.      super.onCreate(savedInstanceState);  
23.      setContentView(R.layout.relative_layout);  
24.      tvw=(TextView)findViewById(R.id.textView1);  
25.      eText=(EditText) findViewById(R.id.editText1);  
26.      eText.setInputType(InputType.TYPE_NULL);  
27.      eText.setOnClickListener(new View.OnClickListener() {  
28.          @Override  
29.          public void onClick(View v) {  
30.              final Calendar cldr = Calendar.getInstance();  
31.              int day = cldr.get(Calendar.DAY_OF_MONTH);  
32.              int month = cldr.get(Calendar.MONTH);  
33.              int year = cldr.get(Calendar.YEAR);  
34.              // date picker dialog  
35.              picker = new DatePickerDialog( context RelativeLayout.this,  
36.                  new DatePickerDialog.OnDateSetListener() {  
37.                      @Override  
38.                      public void onDateSet(DatePicker view, int year, int monthOfYear,  
39.                                         int dayOfMonth) {  
40.                             eText.setText(dayOfMonth + "/" + (monthOfYear + 1) + "/" + year);  
41.                         }  
42.                     }, year, month, day);  
43.                     picker.show();  
44.                 }  
45.             );  
46.             btnGet=(Button)findViewById(R.id.button1);  
47.             btnGet.setOnClickListener(new View.OnClickListener() {  
48.                 @Override  
49.                 public void onClick(View v) {  
50.                     tvw.setText("Selected Date: " + eText.getText());  
51.                 }  
52.             });  
53.         }  
54.     }
```

9. Running hasil code diatas, dan akan menampilkan output seperti berikut:



Gambar 4. Hasil Relative Layout

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan linear layout.
2. Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project linear layout yang di buktikan dengan screenshot pada proses percobaan.
3. Project aplikasi dari percobaan relative layout.
4. Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project relative layout yang di buktikan dengan screenshot pada proses percobaan.

g. Kesimpulan

1. Mahasiswa berhasil memahami konsep linear layout, serta berhasil menerapkan konsep linear layout.
2. Mahasiswa mampu memahami konsep relative layout, serta berhasil menerapkan konsep relative layout.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail setiap baris code pada linear layout dan relative layout	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan dengan singkat setiap baris code pada linear layout dan relative layout	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan setiap baris code pada linear layout dan relative layout	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan setiap baris code pada linear layout dan relative layout	(1)	2	3	4
Jumlah skor					

ACARA 10 : LAYOUT, MATERIAL DESIGN DAN SCROLL VIEW

Pokok Bahasan	: Constraint layout dan Frame Layout
Acara Praktikum/Pertemuan	: Minggu 3 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep constraint layout.
2. Mahasiswa mampu memahami konsep frame layout.
3. Mahasiswa mampu menerapkan konsep constraint layout.
4. Mahasiswa mampu menerapkan konsep frame layout.

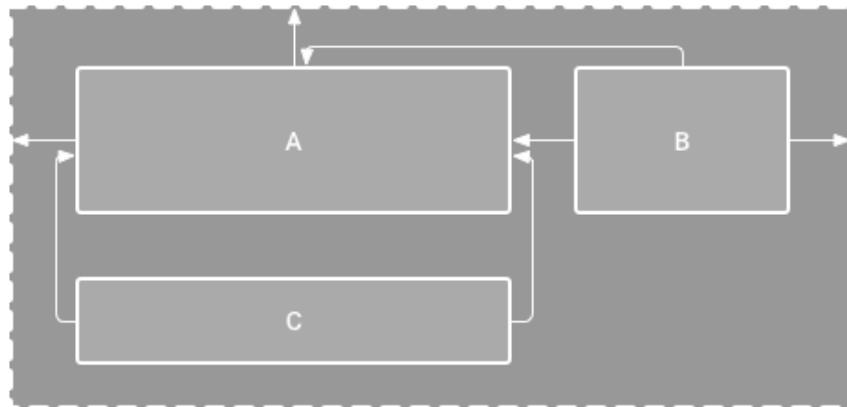
b. Indikator

1. Keberhasilan mahasiswa dalam menerapkan constraint layout pada studi kasus.
2. Keberhasilan mahasiswa dalam menerapkan frame layout pada studi kasus.

c. Dasar Teori

1) Constraint layout

Constraint Layout merupakan layout mirip dengan **Relative Layout**, karena letak View bergantung pada View lain dalam satu Layout ataupun dengan Parent Layoutnya. Contohnya di **Relative Layout** kita bisa meletakkan sebuah View di atas, bawah, atau samping View lain. Kita juga dapat mengatur posisinya berdasarkan Parent Layout menggunakan tag seperti centerVertical, alignParentBottom, dll. Akan tetapi, **Constraint Layout** jauh lebih fleksibel dan lebih mudah digunakan di **Layout Editor**.



Gambar 3. Model Constraint Layout

Pada **Constraint Layout**, setiap View memiliki tali (Constraint) yang menarik tiap sisinya, yang mana tali tersebut bisa kita atur Elastisitas, Margin, dsb. Tali tersebut wajib kita “ikatkan” kepada anchor point atau suatu titik yang dapat berupa sisi dari Parent Layout, View lain, ataupun titik bantu (helper) yang bisa kita buat sendiri.

2) Frame Layout

Frame Layout adalah Layout yang biasanya digunakan untuk membuat objek yang saling bertindihan contohnya yaitu kita membuat button di atas image.

3) Table Layout

Table Layout adalah Layout yang digunakan untuk membangun user interface (tampilan antar muka) aplikasi android dengan berdasarkan Baris dan Kolom. **Table Layout** terdiri dari:

- a) **Row/ baris** pada dasarnya digunakan untuk menyimpan satu jenis record, hanya satu informasi yang dapat disimpan.
- b) **Kolom** adalah sub bagian terbagi dari setiap baris dan satu baris dapat menampung beberapa jenis kolom. Setiap kolom terdiri dari jenis informasi yang berbeda mengenai baris tersebut.

Beberapa tag pada Table Layout :

- a) `TableLayout` : Tag pembuka untuk menggunakan TableLayout
- b) `TableRow` : Tag untuk membuat Baris

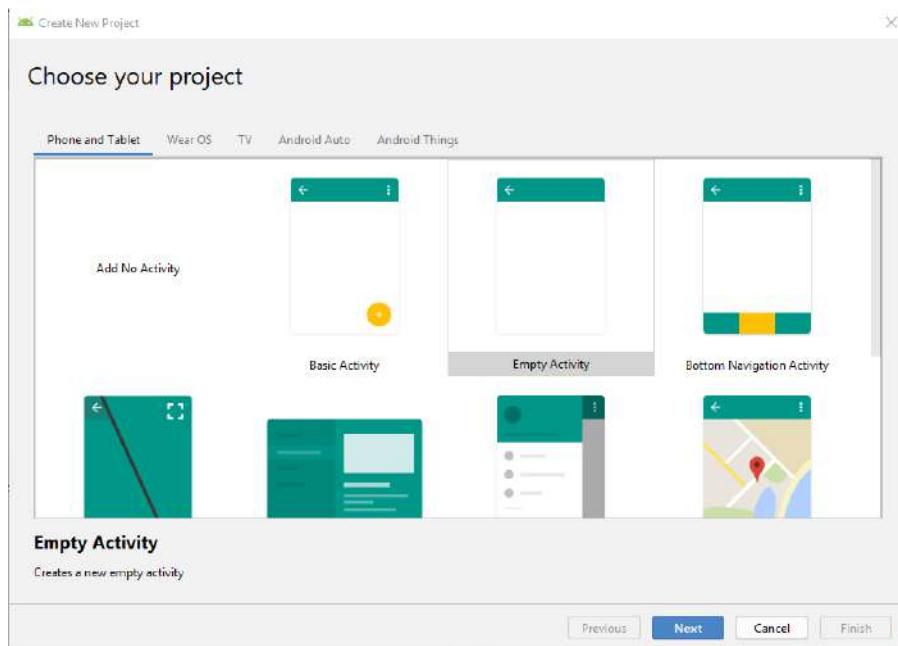
d. Alat dan Bahan

1. Kertas A4
2. Kabel Data USB
3. Jaringan Internet

e. Prosedur Kerja

Constraint Layout

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project Constraint Layout
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**



5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Implementasikan code dibawah ini :

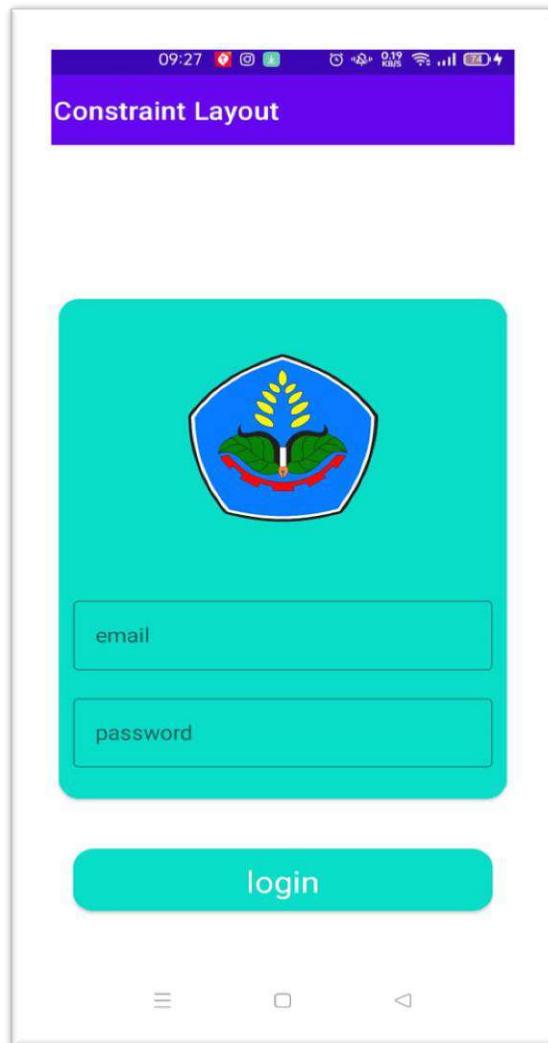
```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".ConstraintLayout">
9      <androidx.cardview.widget.CardView
10         android:layout_width="300dp"
11         android:layout_height="50dp"
12         app:layout_constraintEnd_toEndOf="parent"
13         app:layout_constraintStart_toStartOf="parent"
14         app:layout_constraintBottom_toBottomOf="parent"
15         app:cardCornerRadius="15dp"
16         app:cardMaxElevation="20dp"
17         android:id="@+id/cardView"
18         app:cardBackgroundColor="@color/teal_200"
19         android:layout_marginBottom="50dp">
20             <androidx.constraintlayout.widget.ConstraintLayout
21                 android:layout_width="match_parent"
22                 android:layout_height="match_parent">
23                 <TextView
24                     android:text="login"
25                     android:layout_width="wrap_content"
26                     android:layout_height="wrap_content"
27                     app:layout_constraintTop_toTopOf="parent"
28                     app:layout_constraintStart_toStartOf="parent"
29                     android:layout_marginTop="8dp"
30                     app:layout_constraintEnd_toEndOf="parent"
31                     android:id="@+id/textView"
32                     android:layout_marginBottom="8dp"
33                     app:layout_constraintBottom_toBottomOf="parent"
34                     android:fontFamily="sans-serif"
35                     android:textColor="@android:color/background_light"
36                     android:textSize="24sp"
37                     app:layout_constraintVertical_bias="1.0"/>
38             </androidx.constraintlayout.widget.ConstraintLayout>
39         </androidx.cardview.widget.CardView>
40         <androidx.cardview.widget.CardView
41             android:layout_width="320dp"
42             android:layout_height="400dp"
43             app:layout_constraintStart_toStartOf="parent"
44             app:layout_constraintEnd_toEndOf="parent"
45             app:layout_constraintTop_toTopOf="parent">
```

```
46        android:layout_marginTop="8dp"
47        android:layout_marginBottom="8dp"
48        app:layout_constraintBottom_toTopOf="@+id/cardView"
49        app:cardCornerRadius="15dp"
50        app:cardMaxElevation="20dp"
51        app:layout_constraintVertical_bias="0.781"
52        app:cardBackgroundColor="@color/teal_200"
53        android:id="@+id/cardViews2">
54            <androidx.constraintlayout.widget.ConstraintLayout
55                android:layout_width="match_parent"
56                android:layout_height="match_parent">
57                <ImageView
58                    android:id="@+id/imageView"
59                    android:layout_width="150dp"
60                    android:layout_height="150dp"
61                    android:layout_marginTop="36dp"
62                    android:layout_marginBottom="8dp"
63                    app:layout_constraintBottom_toBottomOf="parent"
64                    app:layout_constraintEnd_toEndOf="parent"
65                    app:layout_constraintStart_toStartOf="parent"
66                    app:layout_constraintTop_toTopOf="parent"
67                    app:layout_constraintVertical_bias="0.0"
68                    app:srcCompat="@drawable/logo" />
69                <com.google.android.material.textfield.TextInputLayout
70                    android:id="@+id/textInputLayout2"
71                    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
72                    android:layout_width="300dp"
73                    android:layout_height="wrap_content"
74                    android:layout_marginTop="50dp"
75                    app:layout_constraintEnd_toEndOf="parent"
76                    app:layout_constraintStart_toStartOf="parent"
77                    app:layout_constraintTop_toBottomOf="@+id/imageView"
78                    app:passwordToggleTint="@color/teal_200">
79                    <com.google.android.material.textfield.TextInputEditText
80                        android:layout_width="match_parent"
81                        android:layout_height="wrap_content"
82                        android:hint="email"
83                        android:inputType="textEmailAddress" />
84                </com.google.android.material.textfield.TextInputLayout>
85                <com.google.android.material.textfield.TextInputLayout
86                    android:id="@+id/textInputLayout3"
87                    style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
88                    android:layout_width="300dp"
89                    android:layout_height="wrap_content"
90                    android:layout_marginTop="8dp"
91                    android:layout_marginBottom="16dp"
```

```
92
93
94
95
96
97
98
99
100
101
102
103
104
105
```

```
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textInputLayout2"
        app:passwordToggleTint="@color/teal_200">
    <com.google.android.material.textfield.TextInputEditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="password"
        android:inputType="textPassword" />
</com.google.android.material.textfield.TextInputLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
</androidx.cardview.widget.CardView>
</androidx.constraintlayout.widget.ConstraintLayout>
```

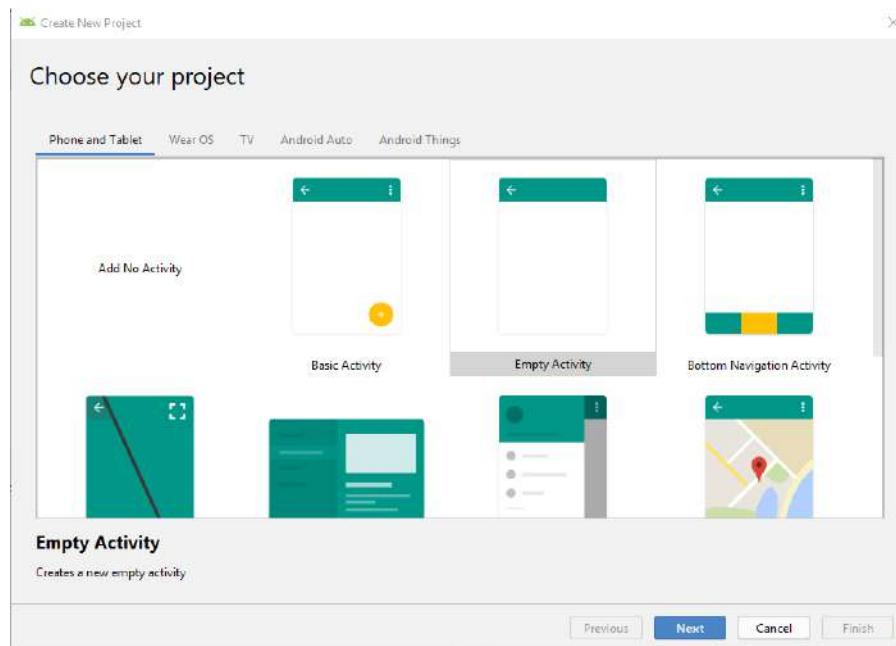
7. Running hasil code diatas, dan akan menampilkan output seperti berikut:



Gambar 5. Hasil Constraint Layout

Frame Layout

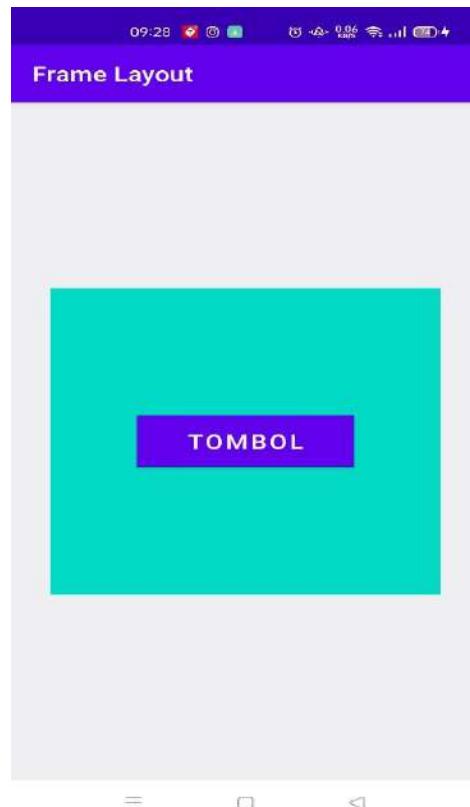
1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project Frame Layout
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**



5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Implementasikan code dibawah ini :

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:background="#111133"
8     tools:context=".FrameLayout">
9     <ImageView
10        android:layout_width="300dp"
11        android:layout_height="300dp"
12        android:layout_gravity="center"
13        android:background="@color/teal_200" />
14     <Button
15        android:layout_width="500px"
16        android:layout_height="150px"
17        android:layout_gravity="center"
18        android:background="@color/teal_200"
19        android:text="Tombol"
20        android:textColor="#ffff"
21        android:textSize="20sp" />
22 </FrameLayout>
```

7. Running hasil code diatas, dan akan menampilkan output seperti berikut:



Gambar 6. Hasil Frame Layout

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan constraint layout.
2. Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project constraint layout yang di buktikan dengan screenshot pada proses percobaan.
3. Project aplikasi dari percobaan frame layout.
4. Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project frame layout yang di buktikan dengan screenshot pada proses percobaan.

g. Kesimpulan

1. Mahasiswa berhasil memahami konsep constraint layout, serta berhasil menerapkan konsep constraint layout.
2. Mahasiswa mampu memahami konsep frame layout, serta berhasil menerapkan konsep frame layout.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail setiap baris code pada constraint layout dan frame layout	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan dengan singkat setiap baris code pada constraint layout dan frame layout	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan setiap baris code pada constraint layout dan frame layout	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan setiap baris code pada constraint layout dan frame layout	(1)	2	3	4
Jumlah skor					

ACARA 11 : LAYOUT, MATERIAL DESIGN DAN SCROLL VIEW

Pokok Bahasan	: Table Layout dan Material Design
Acara Praktikum/Pertemuan	: Minggu 3 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep table layout.
2. Mahasiswa mampu memahami konsep material design.
3. Mahasiswa mampu menerapkan konsep table layout.
4. Mahasiswa mampu menerapkan konsep material design.

b. Indikator

1. Keberhasilan mahasiswa dalam menerapkan table layout pada studi kasus.
2. Keberhasilan mahasiswa dalam menerapkan material design pada studi kasus.

c. Dasar Teori

Material Design

Sebelum mempelajari Material Design pada Android Studio ada baiknya kita mengetahui apa itu Material Design.

“Material Design adalah library desain untuk OS Android yang dikembangkan oleh Google. Dirilis pertama kali untuk OS Android Versi 5 yaitu Lollipop dan diperkenalkan ke public pada acara Google I/O tanggal 25 Juni 2014.”



Contoh Desain Aplikasi Android yang Menggunakan Material Design

Dengan menggunakan Material Design kita bisa membuat tampilan UI / UX (User Interface / User Experience) aplikasi android terlihat lebih nyata, bertekstur, modern, dan profesional.

Ada beberapa komponen Material Design yang bisa kita integrasikan pada aplikasi android yang sedang kita buat, di antaranya :

- Tema Material
- List dan Card
- Shadow
- Animasi
- dan lainnya

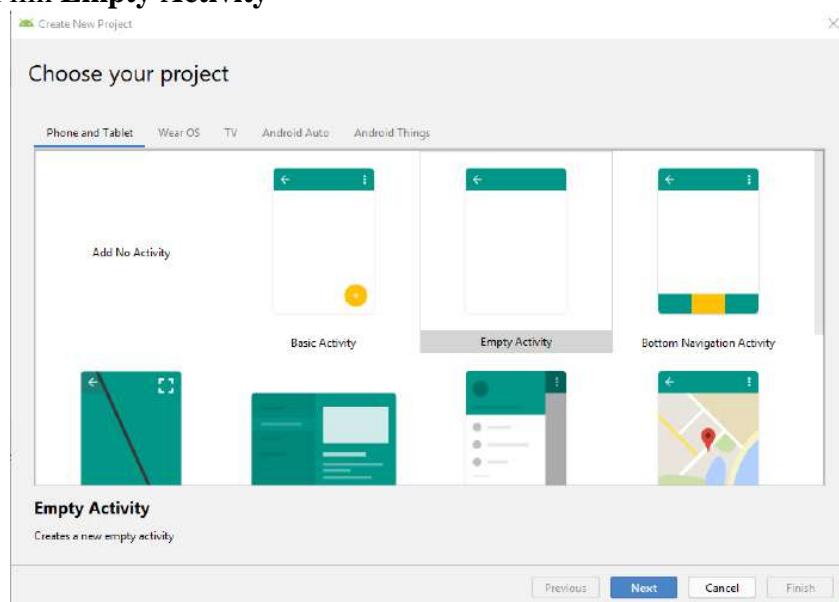
d. Alat dan Bahan

1. Kertas A4
2. Kabel Data USB
3. Jaringan Internet

e. Prosedur Kerja

Tabel Layout

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project Table Layout
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**



5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Implementasikan code dibawah ini :

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="fill_parent"
6      android:layout_height="fill_parent"
7      tools:context=".TableLayout">
8      <TableRow
9          android:paddingTop="10dp"
10         android:gravity="center"
11         android:layout_marginTop="20dp">
12             <TextView
13                 android:id="@+id/status"
14                 android:layout_width="wrap_content"
15                 android:layout_gravity="center"
16                 android:layout_span="2"
17                 android:text="@string/login"
18                 android:textColor="#890000"
19                 android:textSize="25sp"
20                 android:textStyle="bold">
21             </TextView>
22         </TableRow>
23         <TableRow
24             android:layout_marginTop="20dip">
25             <TextView
26                 android:layout_width="wrap_content"
27                 android:text="Username"
28                 android:textSize="20sp"
29                 android:textColor="#000000"
30                 android:layout_marginLeft="20dip">
31             </TextView>
32             <EditText
33                 android:id="@+id/screenName"
34                 android:layout_height="wrap_content"
35                 android:layout_marginLeft="20dip"
36                 android:layout_marginRight="20dip"
37                 android:layout_weight="1">
38             </EditText>
39         </TableRow>
40         <TableRow
41             android:layout_marginTop="20dip">
42             <TextView
43                 android:layout_width="wrap_content"
44                 android:text="Password"
45                 android:textSize="20sp"
46                 android:textColor="#000000"
47                 android:layout_marginLeft="20dip">
48             </TextView>
49             <EditText
```

```
50        android:id="@+id/password"
51        android:layout_height="wrap_content"
52        android:layout_marginLeft="20dip"
53        android:layout_marginRight="20dip"
54        android:layout_weight="1"/>
55    
```

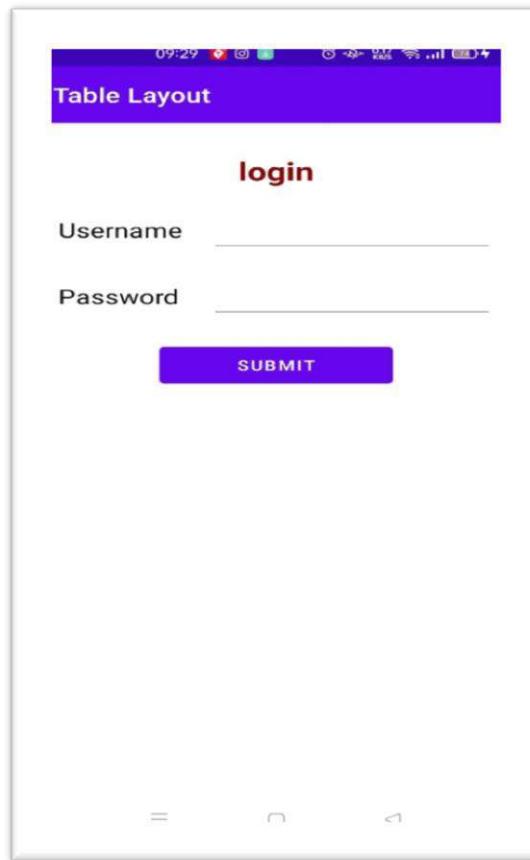
</EditText>

```
56    </TableRow>
57    <TableRow
58        android:gravity="center"
59        android:layout_marginTop="20dip">
60        <Button
61            android:text="Submit"
62            android:clickable="true"
63            android:layout_width="wrap_content"
64            android:layout_height="wrap_content"
65            android:id="@+id/save"
66            android:layout_span="2">
67        
```

</Button>

```
68    </TableRow>
69    </TableLayout>
```

7. Running hasil code diatas, dan akan menampilkan output seperti berikut:

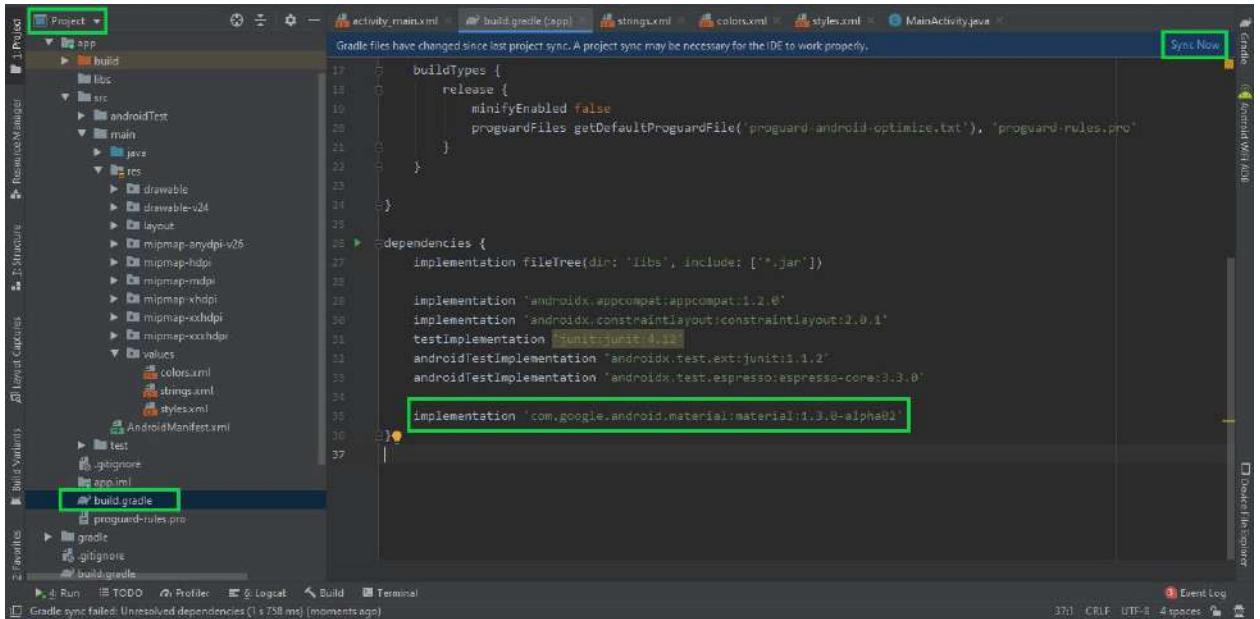


Gambar 7. Hasil Table Layout

Material Design

1. Menambahkan Required Dependency
2. Menambahkan google material design komponen dependency pada **build.gradle file**. Setelah menambahkan dependencies dan jangan lupa click “Sync Now” button pada pojok kanan atas.

implementation ‘com.google.android.material:material:1.3.0-alpha02’



3. Merubah Tema Aplikasi

4. Buka app -> src -> main -> res -> values -> styles.xml jika tidak ada maka buat file **style.xml**, kemudian tambahkan sintak berikut.

```
1 <?xml version="1.0" encoding="utf-8"?>  
2 <resources>  
3     <!-- Base application theme. --&gt;<br/>4     <style name="AppTheme" parent="Theme.MaterialComponents.DayNight.DarkActionBar">  
5         <!-- Customize your theme here --&gt;<br/>6         <item name="colorPrimary">@color/colorPrimary</item>  
7         <item name="colorPrimaryDark">@color/colorPrimaryDark</item>  
8         <item name="colorAccent">@color/colorAccent</item>  
9     </style>  
10 </resources>
```

5. Merubah warna aplikasi

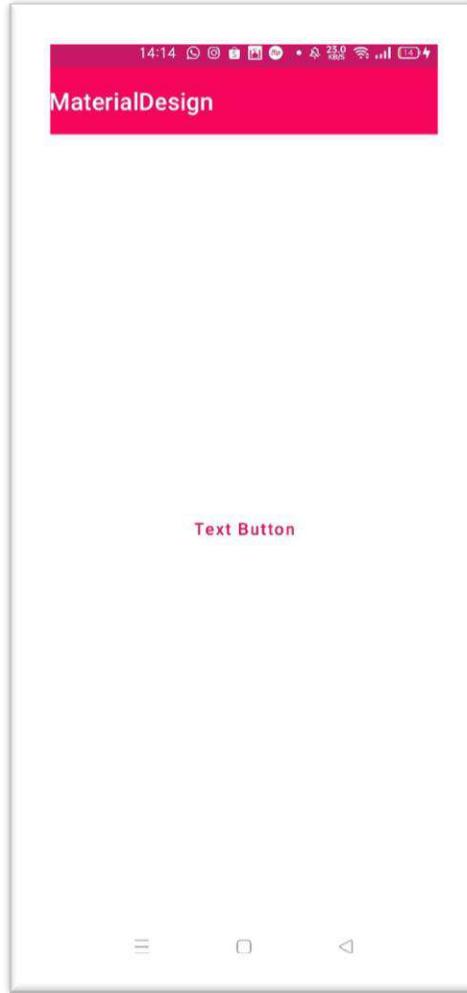
6. Buka app -> src -> main -> res -> value -> colors.xml file dan rubah seperti sintak berikut.

```
1 <?xml version="1.0" encoding="utf-8"?>
2   <resources>
3     <color name="colorPrimary">#F50057</color>
4     <color name="colorPrimaryDark">#C51162</color>
5     <!--> <color name="colorAccent">#FF80AB</color>
6   </resources>
```

7. Menambahkan file materialdesign.xml pada **app -> src -> main -> res -> layout** dengan sintak berikut.

```
1 <?xml version="1.0" encoding="utf-8"?>
2   <androidx.constraintlayout.widget.ConstraintLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     xmlns:tools="http://schemas.android.com/tools"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     tools:context=".MainActivity"
9     tools:ignore="ExtraText">
10
11     <!--style attribute below is to be invoked for text button-->
12     <Button
13       android:id="@+id/text_button"
14       style="@style/Widget.MaterialComponents.Button.TextButton"
15       android:layout_width="wrap_content"
16       android:layout_height="wrap_content"
17       android:text="Text Button"
18       android:textAllCaps="false"
19       app:layout_constraintBottom_toBottomOf="parent"
20       app:layout_constraintEnd_toEndOf="parent"
21       app:layout_constraintStart_toStartOf="parent"
22       app:layout_constraintTop_toTopOf="parent" />
23   </androidx.constraintlayout.widget.ConstraintLayout>
```

8. Maka akan menghasilkan layout seperti di halaman berikutnya:



Gambar 8. Hasil Material Design

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan table layout.
2. Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project table layout yang di buktikan dengan screenshot pada proses percobaan.
3. Project aplikasi dari percobaan material design.
4. Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project material design yang di buktikan dengan screenshot pada proses percobaan.

g. Kesimpulan

1. Mahasiswa berhasil memahami konsep table layout, serta berhasil menerapkan konsep table layout.
2. Mahasiswa mampu memahami konsep material design, serta berhasil menerapkan konsep material design.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail setiap baris code pada table layout dan material design	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan dengan singkat setiap baris code pada table layout dan material design	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan baris code pada table layout dan material design	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan baris code pada table layout dan material design	(1)	2	3	4
Jumlah skor					

ACARA 12 : LAYOUT, MATERIAL DESIGN DAN SCROLL VIEW

Pokok Bahasan	: Scroll View
Acara Praktikum/Pertemuan	: Minggu 3 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep scroll view.
2. Mahasiswa mampu menerapkan konsep scroll view.
3. Mahasiswa mampu menyelesaikan studi kasus.

b. Indikator

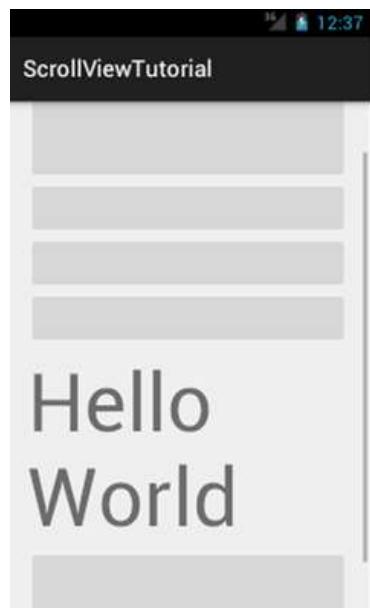
1. Keberhasilan mahasiswa dalam menerapkan scroll view pada studi kasus.
2. Keberhasilan mahasiswa dalam memecahkan studi kasus tambahan.

c. Dasar Teori

ScrollView

ScrollView adalah container untuk tampilan UI pada android. ScrollView digunakan agar user bisa melakukan scroll secara vertikal pada layout yang sudah dibuat jika layout tersebut terlalu besar.

ScrollView sangat penting sekali untuk pengembangan aplikasi android, hampi semua aplikasi yang di kembangkan menggunakan ScrollView. Didalam Android Studio terdapat beberapa jenis ScrollView, yaitu **ScrollView** Vertikal, **HorizontalScrollView** dan **Nested ScrollView** (ScrollView bersarang).



Contoh Desain Aplikasi Android yang Menggunakan Scrol View

d. Alat dan Bahan

1. Kertas A4
2. Kabel Data USB
3. Jaringan Internet

e. Prosedur Kerja

Scroll View

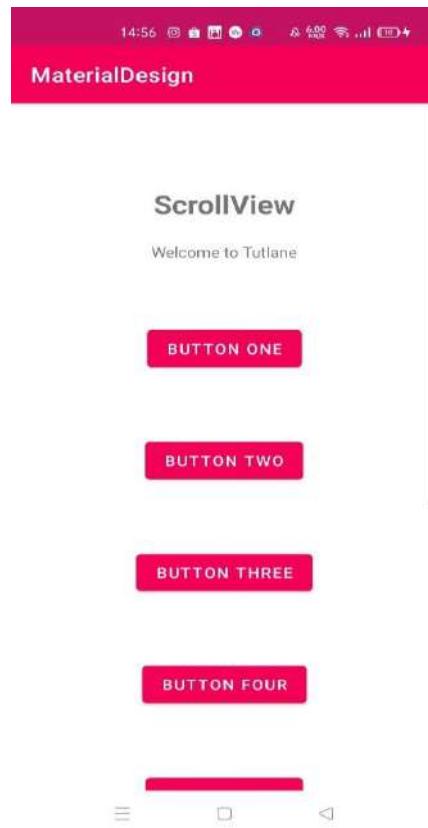
1. Menambahkan file scrollview.xml pada **app → src → main → res → layout** dengan sintak berikut.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="wrap_content"
5     android:fillViewport="false">
6     <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
7         android:orientation="vertical" android:layout_width="match_parent"
8         android:layout_height="match_parent">
9             <TextView android:id="@+id/loginscrn"
10                android:layout_width="wrap_content"
11                android:layout_height="wrap_content"
12                android:layout_marginTop="80dp"
13                android:text="ScrollView"
14                android:textSize="25dp"
15                android:textStyle="bold"
16                android:layout_gravity="center"/>
```

```
16     android:layout_gravity="center"/>
17     <TextView android:id="@+id/fstTxt"
18         android:layout_width="wrap_content"
19         android:layout_height="wrap_content"
20         android:layout_marginTop="20dp"
21         android:text="welcome to Tutlane"
22         android:layout_gravity="center"/>
23     <Button android:layout_width="wrap_content"
24         android:layout_height="wrap_content"
25         android:layout_gravity="center"
26         android:layout_marginTop="60dp"
27
28     <Button android:layout_width="wrap_content"
29         android:layout_height="wrap_content"
30         android:layout_gravity="center"
31         android:layout_marginTop="60dp"
32         android:text="Button Two" />
33     <Button android:layout_width="wrap_content"
34         android:layout_height="wrap_content"
35         android:layout_gravity="center"
36         android:layout_marginTop="60dp"
37         android:text="Button Three" />
38     <Button android:layout_width="wrap_content"
39         android:layout_height="wrap_content"
40         android:layout_gravity="center"
41         android:layout_marginTop="60dp"
42         android:text="Button Four" />
43     <Button android:layout_width="wrap_content"
44         android:layout_height="wrap_content"
45         android:layout_gravity="center"
46         android:layout_marginTop="60dp"
47         android:text="Button Five" />
48     <Button android:layout_width="wrap_content"
49         android:layout_height="wrap_content"
50         android:layout_gravity="center"
51         android:layout_marginTop="60dp"
52         android:text="Button Six" />
53     <Button android:layout_width="wrap_content"
54         android:layout_height="wrap_content"
```

```
55         android:layout_gravity="center"
56         android:layout_marginTop="60dp"
57         android:text="Button Seven" />
58     <Button android:layout_width="wrap_content"
59         android:layout_height="wrap_content"
60         android:layout_gravity="center"
61         android:layout_marginTop="60dp"
62         android:text="Button Eight" />
63     <Button android:layout_width="wrap_content"
64         android:layout_height="wrap_content"
65         android:layout_gravity="center"
66         android:layout_marginTop="60dp"
67         android:text="Button Nine" />
68     </LinearLayout>
69 </ScrollView>
```

Maka akan menghasilkan layout seperti di halaman berikutnya:



Gambar 9. Hasil Scroll View 1

2. Menambahkan file scrollviewhorizontal.xml pada **app → src → main → res → layout** dengan sintak berikut.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <HorizontalScrollView xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="wrap_content"
5     android:fillViewport="true">
6     <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
7         android:orientation="horizontal" android:layout_width="match_parent"
8         android:layout_height="match_parent"
9         android:layout_marginTop="150dp">
10        <Button android:layout_width="wrap_content"
11            android:layout_height="wrap_content"
12            android:text="Button One" />
13        <Button android:layout_width="wrap_content"
14            android:layout_height="wrap_content"
15            android:text="Button Two" />
16        <Button android:layout_width="wrap_content"
17            android:layout_height="wrap_content"
18            android:text="Button Three" />
19        <Button android:layout_width="wrap_content"
20            android:layout_height="wrap_content"
21            android:text="Button Four" />
22        <Button android:layout_width="wrap_content"
23            android:layout_height="wrap_content"
24            android:text="Button Five" />
25        <Button android:layout_width="wrap_content"
26            android:layout_height="wrap_content"
27            android:text="Button Six" />
28        <Button android:layout_width="wrap_content"
29            android:layout_height="wrap_content"
30            android:text="Button Seven" />
31        <Button android:layout_width="wrap_content"
32            android:layout_height="wrap_content"
33            android:text="Button Eight" />
34        <Button android:layout_width="wrap_content"
35            android:layout_height="wrap_content"
36            android:text="Button Nine" />
37    </LinearLayout>
38 </HorizontalScrollView>
```

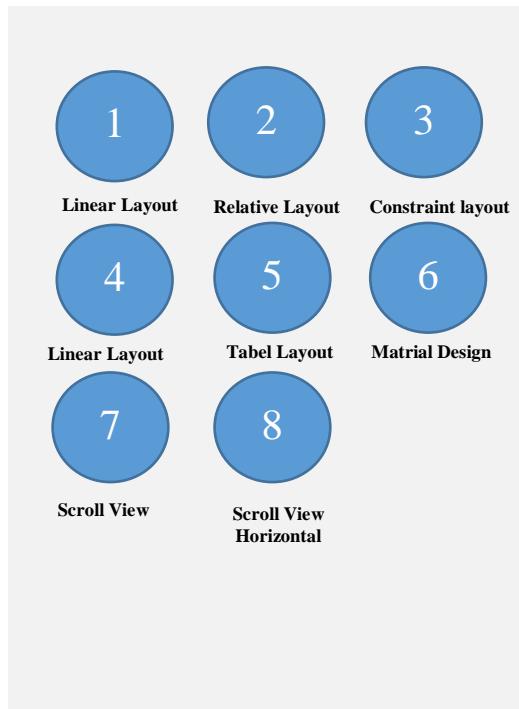
Maka akan menghasilkan layout seperti di halaman berikutnya:



Gambar 10. Hasil Scroll View 2

Tugas

Membuat Layout baru dengan aturan tampilan **minimal** seperti gambar 11 dengan contoh implementasi pada gambar 12 (lebih dari seperti contoh diperbolehkan) :



Gambar 11. Aturan Tampilan



Gamar 12. Contoh Implementasi

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan scroll view.
2. Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project scroll view yang di buktikan dengan screenshot pada proses percobaan.
3. Project aplikasi dari tugas layout, material design dan scroll view.
4. Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project tugas layout, material design dan scroll view yang di buktikan dengan screenshot pada proses percobaan.

g. Kesimpulan

1. Mahasiswa berhasil memahami konsep scroll view, serta berhasil menerapkan konsep scroll view.
2. Mahasiswa mampu menyelesaikan studi kasus dan mampu menjelaskan secara tepat.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menyelesaikan studi kasus dalam bentuk project, serta menjelaskan setiap baris code.	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menyelesaikan studi kasus dalam bentuk project, namun kurang tepat menjelaskan setiap baris code.	1	2	(3)	4
3	Ketepatan waktu dan namun kurang tepat dalam menyelesaikan studi kasus dalam bentuk project serta menjelaskan setiap baris code.	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan setiap baris code.	(1)	2	3	4
Jumlah skor					

ACARA 13 : ACTIVITY (RECYCLER VIEW)

Pokok Bahasan	: Selection Widget
Acara Praktikum/Pertemuan	: Minggu 4 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep dasar selection widged.
2. Mahasiswa mampu membedakan jenis-jenis selection widged.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep dasar selection widged.

c. Dasar Teori

Selection Widget

Selection widget adalah widget yang menyediakan list berupa pilihan-pilihan yang menggunakan framework dari class adapter. Android menyediakan framework data adapter yang menyediakan untuk membantu membangun selection widget dengan data yang berasal baik dari array maupun dari database. Berikut cara menggunakan class adapter :

```
new ArrayAdapter<String>
(this, android.R.layout.simple_list_item_1, items);
```

Beberapa Selection Widgget antara lain :

1) List View

ListView menampilkan daftar item vertical yang dapat di scroll apabila data yang ditampilkan melebihi besarnya layar samalahnya komponen widget lainnya, membuat listView bisa dilakukan dengan men-drag komponen ke dalam layout atau dengan menuliskan source code pada file .xml.

2) Spinner

Spinner menyediakan cara cepat untuk memilih satu nilai dari satu set. Dalam keadaan default, spinner menampilkan nilai yang saat ini dipilih. Spinner yang disentuh akan menampilkan

menu tarik-turun dengan semua nilai lain yang tersedia, dari mana pengguna dapat memilih yang baru.

3) Auto Complete Text View

Editable TextView yang menunjukkan saran penyelesaian secara otomatis saat pengguna mengetik. Daftar saran ditampilkan di menu drop-down tempat pengguna dapat memilih item untuk menggantikan konten kotak edit. Drop-down dapat ditutup kapan saja dengan menekan tombol kembali atau, jika tidak ada item yang dipilih dalam drop-down, dengan menekan tombol tengah enter / dpad.

d. Alat dan Bahan

1. Kertas A4
2. Spidol

e. Prosedur Kerja

1. Mahasiswa membuat rancangan design untuk contoh tampilan listview, spinner, dan autocomplete TextView pada kertas A4.
2. Buat pada PPT dalam bentuk *prototype*
3. Presentasikan didepan kelas

f. Hasil dan Pembahasan

1. Dokumen tugas rancangan design contoh tampilan listview, spinner, dan auto complete text view.
2. Dokumen prototype dalam bntuk PPT.

g. Kesimpulan

Mahasiswa memahami konsep dasar listview, spinner, dan auto complete text view dengan mampu memahami perbedaan dari ketiganya.

h. Rubrik Penilaian

No	Indikator	Skor*			
		1	2	3	4
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail perbedaan listview, spinner, dan auto complete text view				(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan dari tugas	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan tugas	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan tugas	(1)	2	3	4
Jumlah skor					

ACARA 14 : ACTIVITY (RECYCLER VIEW)

Pokok Bahasan	: Implementasi List View
Acara Praktikum/Pertemuan	: Minggu 4 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu meng-implementasi-kan list view.
2. Mahasiswa mampu memahami konsep meng-implementasi-kan list view.

b. Indikator

Keberhasilan mahasiswa dalam meng-implementasikan list view.

c. Dasar Teori

-

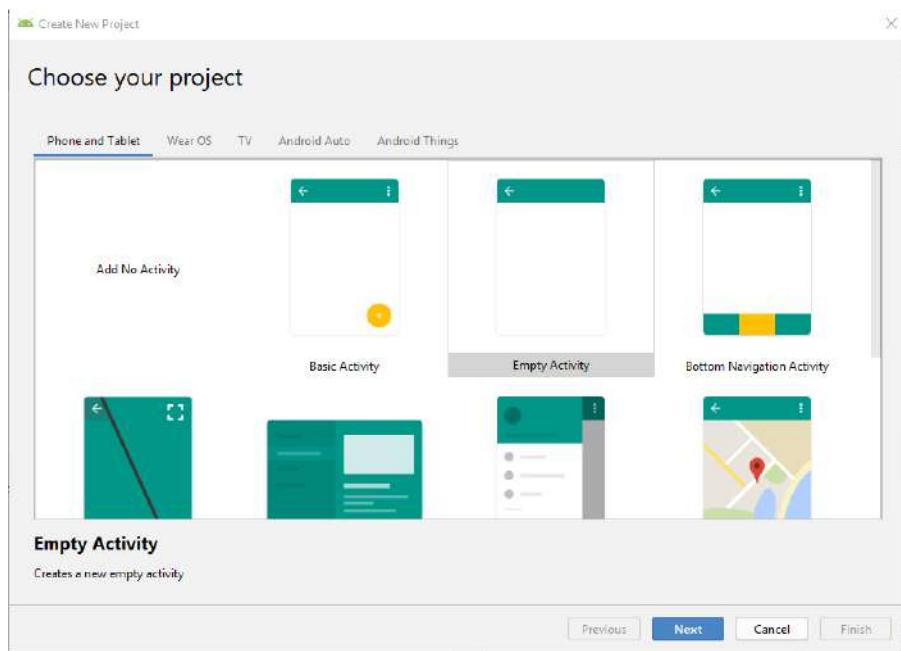
d. Alat dan Bahan

1. Kertas A4
2. Jaringan Internet

e. Prosedur Kerja

List View

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project ListViewActivity pada XMLnya
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**



5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Implementasikan code dibawah ini :

```
<resources>
    <string name="app_name">ExampleListView</string>
    <string-array name="countries_arr">
        <item>Afghanistan</item>
        <item>Albania</item>
        <item>Algeria</item>
        <item>American Samoa</item>
        <item>Andorra</item>
        <item>Angola</item>
        <item>Anguilla</item>
        <item>Antarctica</item>
        <item>Antigua and Barbuda</item>
```

```

<item>Argentina</item>
<item>Armenia</item>
<item>Aruba</item>
<item>Australia</item>
<item>Austria</item>
<item>Azerbaijan</item>
<item>Bahrain</item>
<item>Bangladesh</item>
<item>Barbados</item>
<item>Belarus</item>
<item>Belgium</item>
</string-array>
</resources>

```

7. Setelah itu, masuk ke **string.xml** yang letaknya di folder **res/values/string.xml** lalu implementasikan kode dibawah ini :

```

<?xml version="1.0" encoding="utf-8"?>
<ListView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/listView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="app.com.example.android.examplelistview.MainActivity">

</ListView>

```

disini dicontohkan menampilkan beberapa nama negara di dunia seperti contoh di halaman berikutnya.

8. Setelah **string.xml**, sekarang ke activity yang telah di buat, tambahkan kode berikut :

```

package app.com.example.android.examplelistview;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements AdapterView.OnItemClickListener{
    ListView listView;
    ArrayAdapter<CharSequence> adapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        listView = (ListView)findViewById(R.id.listView);
        adapter = ArrayAdapter.createFromResource(this,R.array.countries_arr,android.R.layout.simple_list_item_1);
        //adapter= new ArrayAdapter<String>(this, android.R.layout.simple_list_item_1);
        listView.setAdapter(adapter);
        listView.setOnItemClickListener(this);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.list, menu);
        return true;
    }

    @Override
    public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
        Toast.makeText(this, adapter.getItem(position), Toast.LENGTH_SHORT).show();
    }
}

```

Penjelasan program pada skrip program adapter :

- **ArrayAdapter.createFromResource(this,R.array.countries_arr,android.R.layout.simple_list_item_1);** itu berarti mengambil nilai array pada **string.xml** dengan nama **countries_arr**, lalu menampilkannya dengan model tampilan **simple_list_item_1**.
- Kemudian pada method **onItemClick()** berarti ketika data pada listview ditekan akan menampilkan data berupa Toast message, lalu memanggil method onItemClick ini pada method onCreate dengan cara **listView.setOnItemClickListener(this);**
- Lalu perlu extended kan class **MainActivity** seperti ini public class **MainActivity extends AppCompatActivity implements AdapterView.OnItemClickListener**

9. Running hasil code diatas dan lakukan uji coba, kemudian perhatikan apa yang terjadi.

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan list view.
2. Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project list view yang dibuktikan dengan screenshot pada proses percobaan.

g. Kesimpulan

Mahasiswa berhasil meng-implementasi-kan list view pada project individu serta memahami konsep meng-implementasi-kan list view.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail setiap baris code pada percobaan listview	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan baris code pada percobaan listview	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat baris code pada percobaan listview	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan baris code pada percobaan listview	(1)	2	3	4
Jumlah skor					

ACARA 15 : ACTIVITY (RECYCLER VIEW)

Pokok Bahasan	: Card View dan Recycler View
Acara Praktikum/Pertemuan	: Minggu 4 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu meng-implementasi-kan card view dan recycler view.
2. Mahasiswa mampu memahami konsep meng-implementasi-kan card view dan recycler view.

b. Indikator

Keberhasilan mahasiswa dalam meng-implementasikan card view dan recycler view.

c. Dasar Teori

Card View dan Recycler View

Terkadang dalam sebuah aplikasi kita ingin menampilkan sebuah set data yang berjumlah besar (ratusan — atau mungkin sampai jutaan). Nah disini kita tentu perlu sebuah view yang mampu menghandle itu. Adapun sebelum RecyclerView ada namanya ListView. Namun ada beberapa kekurangan yang ada pada ListView. Disini muncullah RecyclerView dengan kemampuan yang lebih baik dari ListView (lebih cepat dan lebih efisien — terutama dalam menangani data berjumlah besar). Adapun contoh penggunaan RecyclerView ada pada GMail.

Sedangkan CardView merupakan view paketan dari Material Design — diperkenalkan pada Android Versi Lollipop. Bentuk view ini biasa digunakan untuk pembungkusan data (sepaket data — beberapa view) yang ditandai dengan tampilan view nya yang memiliki drop shadow (elevation) dan corner radius. Adapun CardView ini merupakan subclass dari FrameLayout.

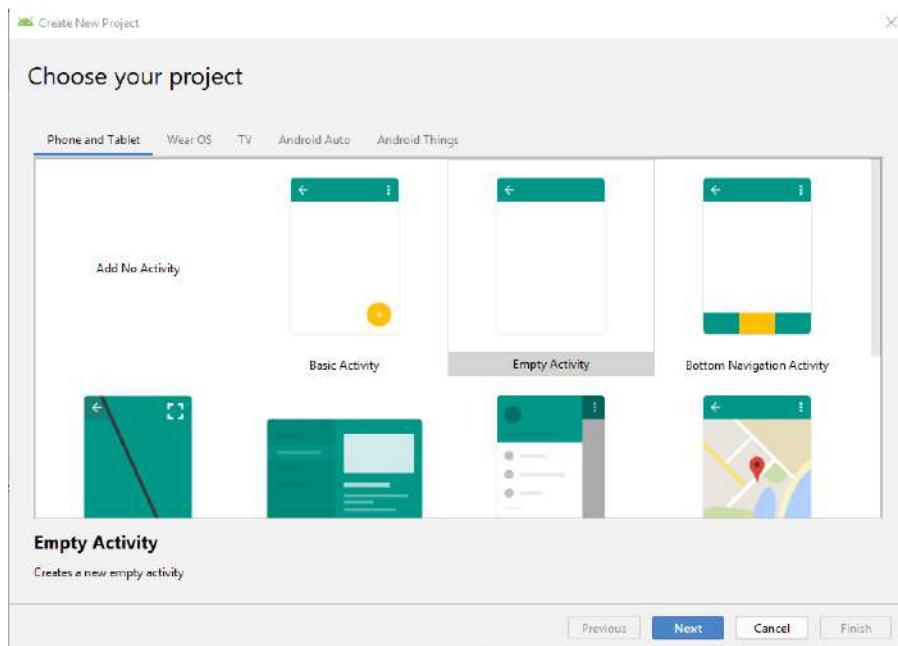
d. Alat dan Bahan

1. Kertas A4
2. Kabel Data USB
3. Jaringan Internet

e. Prosedur Kerja

AutoComplete TextView

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project **Linear Layout**
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**



5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Tambahkan dependencies seperti kode dibawah ini :

```
dependencies {
    // rest of code

    implementation 'com.android.support:recyclerview-v7:28.0.0'
    implementation 'com.android.support:cardview-v7:28.0.0'
}
```

7. Tambahkan recyclerview pada layout

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <android.support.v7.widget.RecyclerView
        android:id="@+id/recycler_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>

</RelativeLayout>
```

8. Membuat model class sesuai dengan data

Contoh terdapat class untuk mahasiswa maka model akan nampak seperti berikut :

```
public class Mahasiswa {

    private String nama;
    private String npm;
    private String nohp;

    public Mahasiswa(String nama, String npm, String nohp) {
        this.nama = nama;
        this.npm = npm;
        this.nohp = nohp;
    }

    public String getNama() {
        return nama;
    }

    public void setNama(String nama) {
        this.nama = nama;
    }
}
```

```

public String getNpm() {
    return npm;
}

public void setNpm(String npm) {
    this.npm = npm;
}

public String getNohp() {
    return nohp;
}

public void setNohp(String nohp) {
    this.nohp = nohp;
}
}

```

9. Membuat layout list_itemnya

Setelah itu mari kita buat layout untuk *view item*-nya. Ini untuk mendeskripsikan apa saja element dalam baris list. Disinilah kita letakkan CardView-nya. Dalam contoh ini kita ingin menampilkan nama, npm dan nohp dalam TextView :

```

<?xml version="1.0" encoding="utf-8"?>

<android.support.v7.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android
    xmlns:card_view="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    card_view:cardCornerRadius="6dp"
    card_view:cardElevation="3dp"
    card_view:cardUseCompatPadding="true">

    <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="5dp"
        android:orientation="vertical"
        android:padding="5dp">

```

```

<!-- TextView Nama Mahasiswa -->
<TextView
    android:id="@+id/txt_nama_mahasiswa"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

<!-- TextView Npm Mahasiswa -->
<TextView
    android:id="@+id/txtnpm_mahasiswa"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

<!-- TextView NoHp Mahasiswa -->
<TextView
    android:id="@+id/txt_nohp_mahasiswa"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

</LinearLayout>

</android.support.v7.widget.CardView>

```

10. Membuat Adapter

```

package net.androku.databindingmvvm;

import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import java.util.ArrayList;
|
public class MahasiswaAdapter extends RecyclerView.Adapter<MahasiswaAdapter.MahasiswaViewHolder> {

    private ArrayList<Mahasiswa> dataList;

    public MahasiswaAdapter(ArrayList<Mahasiswa> dataList) {
        this.dataList = dataList;
    }
}

```

```

@Override
public MahasiswaViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
    LayoutInflator layoutInflater = LayoutInflator.from(parent.getContext());
    View view = layoutInflater.inflate(R.layout.row_mahasiswa, parent, false);
    return new MahasiswaViewHolder(view);
}

@Override
public void onBindViewHolder(MahasiswaViewHolder holder, int position) {
    holder.txtNama.setText(dataList.get(position).getNama());
    holder.txtNpm.setText(dataList.get(position).getNpm());
    holder.txtNoHp.setText(dataList.get(position).getNohp());
}

@Override
public int getItemCount() {
    return (dataList != null) ? dataList.size() : 0;
}

public class MahasiswaViewHolder extends RecyclerView.ViewHolder{
    private TextView txtNama, txtNpm, txtNoHp;

    public MahasiswaViewHolder(View itemView) {
        super(itemView);
        txtNama = (TextView) itemView.findViewById(R.id.txt_nama_mahasiswa);
        txtNpm = (TextView) itemView.findViewById(R.id.txt_npm_mahasiswa);
        txtNoHp = (TextView) itemView.findViewById(R.id.txt_nohp_mahasiswa);
    }
}
}

```

11. Set Adapter ke RecyckerView dalam MainActivity.java

Disini akan ditambahkan dummy data yang akan dimasukkan ke arraylist dan di load keadapter yang telah dibuat sebelumnya.

```

package net.androku.databindigmvvm;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {

    private RecyclerView recyclerView;
    private MahasiswaAdapter adapter;
    private ArrayList<Mahasiswa> mahasiswaArrayList;

```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    addData();

    recyclerView = (RecyclerView) findViewById(R.id.recycler_view);

    adapter = new MahasiswaAdapter(mahasiswaArrayList);

    RecyclerView.LayoutManager layoutManager = new LinearLayoutManager(MainActivity.this);

    recyclerView.setLayoutManager(layoutManager);

    recyclerView.setAdapter(adapter);
}

void addData(){
    mahasiswaArrayList = new ArrayList<>();
    mahasiswaArrayList.add(new Mahasiswa("Dimas Maulana", "1414370309", "123456789"));
    mahasiswaArrayList.add(new Mahasiswa("Fadly Yonk", "1214234560", "987654321"));
    mahasiswaArrayList.add(new Mahasiswa("Ariyandi Nugraha", "1214230345", "987648765"));
    mahasiswaArrayList.add(new Mahasiswa("Aham Siswana", "1214378098", "098758124"));
}

```

Apabila tidak ada error jalankan dan lihat hasilnya seperti apa, bandingkan dengan penggunaan listview.

f. Hasil dan Pembahasan

Project aplikasi dari percobaan implementasi code card view dan recycler view.

g. Kesimpulan

Mahasiswa berhasil meng-implementasi-kan list view pada project individu serta memahami konsep implementasi list view.

h. Rubrik Penilaian

No	Indikator	Skor*				
1	Ketepatan waktu dan ketepatan implementasi pada percobaan card view dan recycler view dengan penjelasan singkat	1	2	3	(4)	
2	Ketepatan waktu dan ketepatan implementasi pada percobaan card view dan recycler view	1	2	(3)	4	
3	Ketepatan waktu akan tetapi kurang tepat implementasi pada percobaan card view dan recycler view	1	(2)	3	4	
4	Keterlambatan pengumpulan tugas dan ketidaktepatan implementasi pada percobaan card view dan recycler view	(1)	2	3	4	
Jumlah skor						

ACARA 16 : ACTIVITY (RECYCLER VIEW)

Pokok Bahasan	: Card View dan Recycler View
Acara Praktikum/Pertemuan	: Minggu 4 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep dasar card view dan recycler view.
2. Mahasiswa mampu memahami setiap baris code dari implementasi card view dan recycler view.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep dasar card view dan recycler view.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kertas A4
2. Jaringan Internet

e. Prosedur Kerja

1. Mahasiswa melakukan analisa code untuk setiap baris code dari implementasi card view dan recycler view.
2. Menjelaskan fungsi dan arti dari setiap baris code.
3. Analisa hasil running dari implementasi project card view dan recycler view.
4. Screenshot output setiap proses dari project card view dan recycler view.
5. Tulisakan dalam bentuk laporan.

f. Hasil dan Pembahasan

Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project card view dan recycler view yang di buktikan dengan screenshot pada proses percobaan.

g. Kesimpulan

Mahasiswa memahami konsep dasar card view dan recycler view dengan mampu memahami setiap baris code dari implementasi card view dan recycler view.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail setiap baris code pada implemenasi card view dan recycler view	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan setiap baris code pada implemenasi card view dan recycler view	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan setiap baris code pada implemenasi card view dan recycler view	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan setiap baris code pada implemenasi card view dan recycler view	(1)	2	3	4
Jumlah skor					

ACARA 17 : FRAGMENT

Pokok Bahasan	: Konsep Dasar Fragment
Acara Praktikum/Pertemuan	: Minggu 5 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep fragment.
2. Mahasiswa mampu menerapkan dan meng-implementasikan fragment pada android.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep fragment, implementasi fragment pada android, serta analisa source code.

c. Dasar Teori

Fragment

Fragment merupakan komponen view yang fleksibel dan dapat disisipkan pada sebuah tampilan dari activity, di samping itu fragment juga merupakan komponen yang memiliki fungsi untuk menampilkan antarmuka ke pengguna melalui activity dengan memiliki layout XML sendiri. Fragment memiliki daur hidup sendiri yang bergantung penuh pada daur hidup activity dimana fragment tersebut ditanamkan, namun tidak tergantung pada activity yang disisipkan. Fragment digunakan untuk memecah komponen pada aplikasi agar bisa lebih fleksibel dan mudah untuk digunakan.

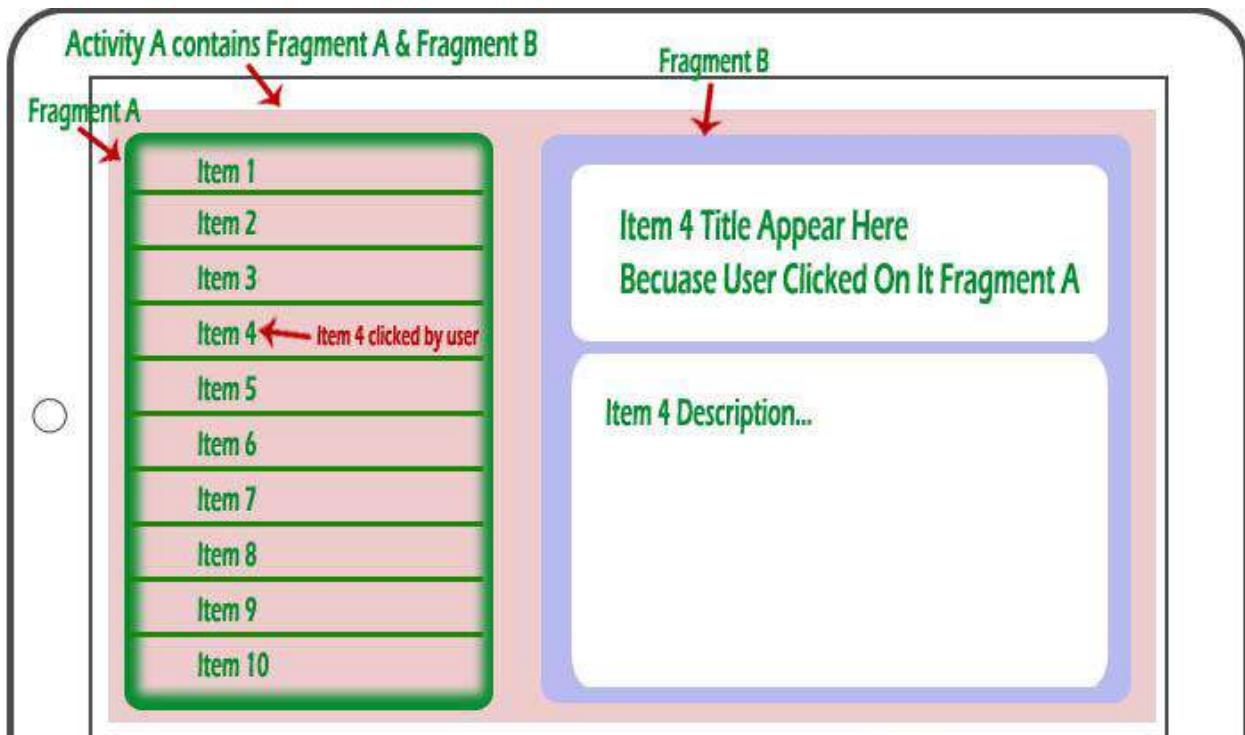
Sebelum Fragment diperkenalkan, kita hanya dapat menampilkan satu activity di layar pada satu waktu tertentu sehingga tidak dapat membagi layar dan mengontrol bagian yang berbeda secara terpisah. Dengan bantuan Fragment, kita dapat membagi layar di berbagai bagian dan mengontrol bagian-bagian yang berbeda secara terpisah. Dengan menggunakan fragment, kita dapat membentuk beberapa fragment dalam satu activity. Fragment memiliki event, layout, dan status mereka sendiri. Hal ini memberikan fleksibilitas dan juga meniadakan limitasi dalam activity tunggal di layar pada suatu waktu. Berikut ini adalah kode untuk fragment dasar dalam XML.

```

<fragment
    android:id="@+id/fragments"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />

```

Fragment digunakan agar komponen tampilan aplikasi menjadi fleksibel dan dapat digunakan kembali (reusable). Satu activity bisa memiliki lebih dari satu fragment. Tidak seperti activity, fragment tidak perlu didaftarkan ke dalam file AndroidManifest.xml. Satu kelas Java dinyatakan sebagai sebuah fragment ketika kelas tersebut meng- extends (inherit) kelas fragment. Melalui Android Support Library, fragment bersifat kompatibel sampai Android api level 10 Gingerbread. Analogi yang mendekati fragment pada platform lain adalah penggunaan komponen iframe pada aplikasi berbasis web.



Di Android fragmen adalah bagian dari aktivitas yang memungkinkan desain aktivitas yang lebih modular. Tidak salah jika kita mengatakan fragmen adalah sejenis sub kegiatan. Ini mewakili perilaku atau bagian dari antarmuka pengguna dalam suatu activity. Kita dapat menggabungkan beberapa fragmen dalam activity tunggal untuk membuat UI multi panel dan menggunakan

kembali fragmen dalam beberapa activity. Kita selalu perlu menyematkan Fragmen dalam suatu aktivitas dan daur hidup fragmen secara langsung dipengaruhi oleh daur hidup aktivitas host.

Dapat membuat fragmen dengan memperluas kelas fragmen atau dengan memasukkan fragmen ke dalam tata letak activity dengan mendeklarasikan fragmen dalam file tata letak activity, sebagai elemen <fragment>. Kami dapat memanipulasi setiap fragmen secara independen, seperti menambah atau menghapusnya.



Saat melakukan transaksi fragmen, kita dapat menambahkan fragmen ke dalam tumpukan belakang yang dikelola oleh activity. *Back-stack* memungkinkan kita untuk membalikkan transaksi fragmen dengan menekan tombol *Back* pada perangkat. Sebagai contoh jika kita mengganti fragmen dan menambahkannya ke dalam *back-stack* maka pada menekan tombol kembali pada perangkat itu akan menampilkan fragmen sebelumnya.

Beberapa poin penting mengenai fragmen dalam android:

- 1) Fragmen ditambahkan dalam versi Android Honeycomb

- 2) Kita dapat menambah, mengganti atau menghapus fragmen dalam sebuah activity saat aktivitas sedang berjalan. Untuk melakukan operasi ini kita membutuhkan Layout (Relative Layout, FrameLayout atau layout lainnya) dalam file xml dan kemudian mengganti layout tersebut dengan fragment yang diperlukan.
- 3) Fragmen memiliki tata letak dan perilakunya sendiri dengan *callback* siklus hidupnya sendiri.
- 4) Fragmen dapat digunakan dalam berbagai aktivitas.
- 5) Kita juga bisa menggabungkan beberapa fragmen dalam satu aktivitas untuk membangun UI multi-bidang.

Kegunaan fragmen di android antara lain:

- 1) Sebelum pengenalan fragment, aplikasi hanya dapat menampilkan satu aktivitas di layar pada satu titik waktu tertentu, jadi tidak dapat membagi layar dan mengontrol bagian yang berbeda secara terpisah. Dengan bantuan fragment, dapat membagi layar di bagian yang berbeda dan mengontrol bagian yang berbeda secara terpisah.
- 2) Dengan menggunakan fragmen, kita dapat menyusun beberapa fragmen dalam satu aktivitas.
- 3) Fragmen memiliki acara, tata letak, dan siklus hidupnya sendiri-sendiri. Ini memberikan fleksibilitas dan juga menghilangkan batasan aktivitas tunggal di layar pada satu waktu.

Kode dasar Fragment dalam XML:

```
<fragment  
    android:id="@+id/fragments"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent" />
```

Membuat Fragment Class pada Android Studio:

- 1) Untuk membuat fragmen, pertama-tama memperluas kelas fragmen
- 2) Lalu mengganti metode siklus hidup kunci untuk memasukkan logika aplikasi kita, mirip dengan yang kita lakukan dengan kelas activity.
- 3) Saat membuat fragmen, harus menggunakan *callback* **onCreateView ()** untuk menentukan tata letak dan menjalankan fragmen.

```

import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.ViewGroup;

public class FirstFragment extends Fragment {
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
Bundle savedInstanceState) {
// Inflate the layout for this fragment
return inflater.inflate(R.layout.fragment_first, container, false);
}
}

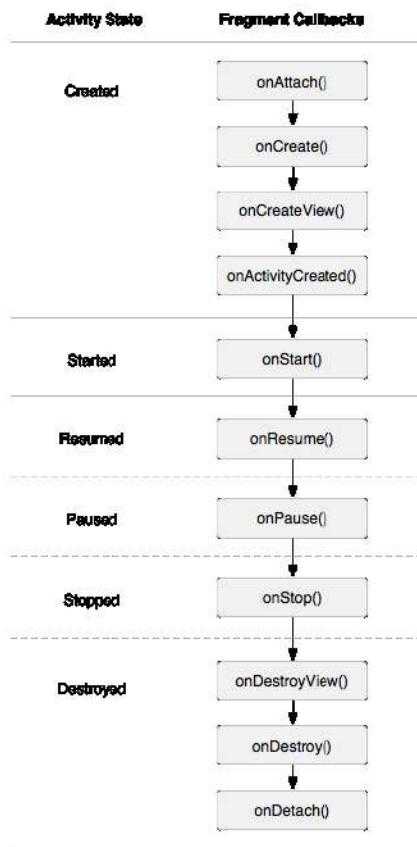
```

- Di sini, parameter inflater adalah LayoutInflater yang digunakan untuk memekarkan tata letak, parameter penampung adalah **ViewGroup** induk (dari tata letak activity) tempat tata letak fragmen akan dimasukkan.
- Parameter **saveInstanceState** adalah **Bundle** yang menyediakan data tentang instance fragment sebelumnya.
- Metode **inflate ()** memiliki tiga argumen:
 - Pertama adalah tata letak sumber daya yang ingin kita perbesar
 - Kedua adalah ViewGroup untuk menjadi induk dari tata letak yang diperbesar. Meneruskan penampung penting agar sistem dapat menerapkan parameter tata letak ke tampilan akar dari tata letak yang digelembungkan, ditentukan oleh tampilan induk yang dituju.
 - Ketiga adalah nilai boolean yang menunjukkan apakah tata letak yang diluaskan harus dilampirkan ke **ViewGroup** (parameter kedua) selama inflasi.

Ada beberapa kelas utama yang terkait dengan Fragmen yaitu:

- 1) FragmentActivity: Basic class yang memuat segala hal mengenai kompatibilitas pada fragment
- 2) Fragment: Class induk dari fragment
- 3) FragmentManager: Kelas untuk berinteraksi dengan objek fragmen di dalam Activity
- 4) FragmentTransaction kelas untuk melakukan serangkaian atomic set of Fragment seperti ganti atau tambah fragmen.

Fragment Life Cycle



Skema di atas menunjukkan *callback* method apa saja yang akan dipanggil di dalam fragment ketika terjadi perubahan pada sebuah activity. Terdapat beberapa state yang perlu diketahui dalam penggunaan fragment ini.

- **Resume**

Fragment bisa dilihat ketika activity sedang berjalan.

- **Paused**

Ketika ada activity lain yang menutupi sebagian dari activity dimana fragment ditambahkan. Yang dimaksud menutupi sebagian adalah ketika activity-nya tidak tertutup sepenuhnya oleh activity lain. Jadi masih ada bagian dari activity yang masih bisa dilihat di layar.

- **Stopped**

Ketika fragment tidak kelihatan di layar. Bisa jadi karena activity dimana fragment itu ditambahkan berhenti atau bahkan fragment itu sendiri sudah dihapus dari activity. Pada

kondisi ini fragment masih hidup dengan semua informasinya. Akan tetapi sudah tidak kelihatan di layar dan akan dihancurkan.

Skema di halaman sebelumnya menunjukkan bahwa perubahan state dari sebuah activity akan mempengaruhi life cycle dari sebuah fragment. Ini karena fragment merupakan komponen view yang bisa ditambahkan (embed) ke dalam activity.

d. Alat dan Bahan

1. Kertas A4
2. Spidol

e. Prosedur Kerja

1. Mahasiswa mencari video refrensi tentang implementasi fragment.
2. Analisa kegunaan dan manfaat fragment dari video yang sudah dilihat.
3. Resume pada kertas A4.

f. Hasil dan Pembahasan

1. Dokumen berupa video refrensi yang telah di unduh dan di analisa.
2. Dokumen tugas resum analisa video dan contoh pada aplikasi yang digunakan.

g. Kesimpulan

Mahasiswa memahami konsep dasar fragment dengan menerapkan contoh pada aplikasi android yang sering digunakan sehari-hari

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail tugas review video	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan dari tugas review video	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan tugas review video	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan tugas review video	(1)	2	3	4
Jumlah skor					

ACARA 18 : FRAGMENT

Pokok Bahasan	: Implementasi Fragment
Acara Praktikum/Pertemuan	: Minggu 5 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep fragment.
2. Mahasiswa mampu menerapkan dan meng-implementasikan fragment pada android.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep fragment, implementasi fragment pada android, serta analisa source code.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kertas A4
2. Kabel Data USB
3. Jaringan Internet

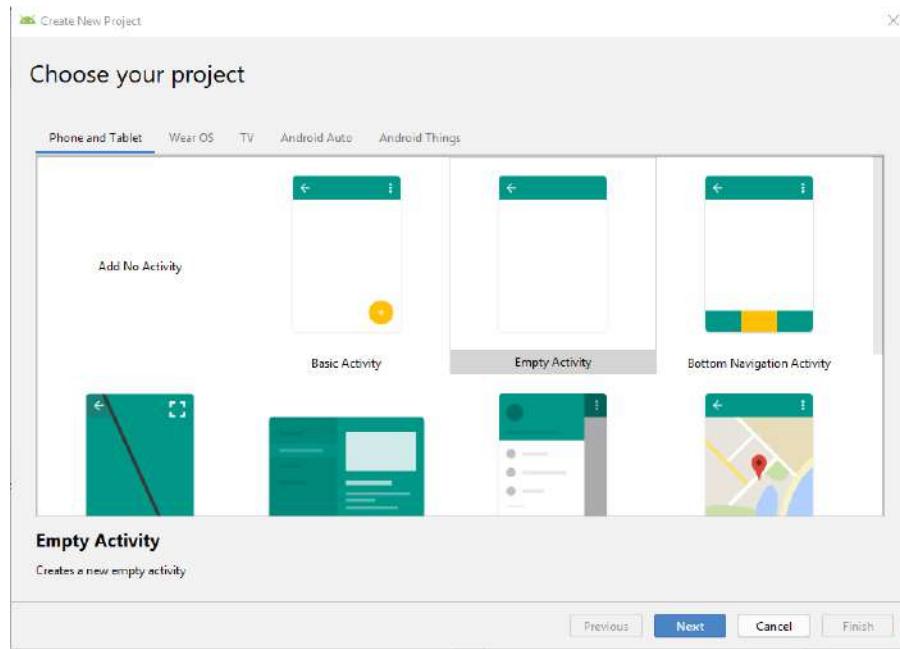
e. Prosedur Kerja

- Percobaan di acara 18 praktikum minggu 5 pertemuan 2 terdapat contoh dua fragmen dan memuatnya dengan klik tombol.
- Menampilkan dua tombol dan *FrameLayout* di acivity dan melakukan acivity **setOnClickListener** di kedua Tombol.
- Pada klik tombol pertama mengganti fragmen pertama dan pada klik tombol kedua mengganti fragmen kedua dengan tata letak (*FrameLayout*).
- Di kedua fragmen menampilkan *TextView* dan tombol, ketika klik tombol akan menampilkan nama fragmen dengan bantuan *Toast*.

1. Buka **Android Studio**

2. Buat project baru, dan berikan nama untuk project **FragmentExample**

3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15.
4. Pilih **Empty Activity**



5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Buka **res → layout →activity_main.xml** (or) **main.xml** dan tambakan code berikut:

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:orientation="vertical"
8      tools:context=".MainActivity">
9      <Button
10         android:id="@+id/firstFragment"
11         android:layout_marginTop="10dp"
12         android:layout_width="match_parent"
13         android:layout_height="wrap_content"
14         android:text=" First Fragment"
15         android:textColor="@color/white"
16         android:textSize="20sp" />

```

```
17     <Button
18         android:id="@+id/secondFragment"
19         android:layout_width="match_parent"
20         android:layout_height="wrap_content"
21         android:text="Second Fragment"
22         android:layout_marginTop="10dp"
23         android:textColor="@color/white"
24         android:textSize="20sp"/>
25     <FrameLayout
26         android:id="@+id/frameLayout"
27         android:layout_width="match_parent"
28         android:layout_height="match_parent"
29         android:layout_marginTop="10dp" />
30     </LinearLayout>
```

7. Buka src → package → MainActivity.java

- Pada langkah ini membuka **MainActivity** dan menambahkan kode untuk memulai tombol.
- Setelah itu melakukan activity **setOnClickListener** di kedua tombol.
- Pada klik tombol pertama mengganti fragmen pertama dan pada klik tombol kedua mengganti fragmen kedua dengan tata letak (*FrameLayout*).
- Untuk mengganti fragment dengan *FrameLayout* :
 - Pertama-tama membuat Manajer Fragmen
 - Kemudian memulai transaksi menggunakan Transaksi Fragmen
 - Terakhir mengganti fragmen dengan tata letak yaitu *FrameLayout*.

```
3     import androidx.appcompat.app.AppCompatActivity;
4     import android.os.Bundle;
5     import android.view.View;
6     import android.widget.Button;
7     import android.app.FragmentTransaction;
8     import android.app.Fragment;
9     import android.app.FragmentManager;
```

```
10
11     public class MainActivity extends AppCompatActivity {
12         Button btnFirstFragment , btnSecondFragment ;
13
14     @Override
15     protected void onCreate(Bundle savedInstanceState) {
16         super.onCreate(savedInstanceState);
17         setContentView(R.layout.activity_main);
18         btnFirstFragment = (Button) findViewById(R.id.firstFragment);
19         btnSecondFragment = (Button) findViewById(R.id.secondFragment);
20         btnFirstFragment.setOnClickListener(new View.OnClickListener() {
21             @Override
22             public void onClick(View v) {
23                 // Load First Fragment
24                 loadFragment(new FirstFragment());
25             }
26         });
27         btnSecondFragment.setOnClickListener(new View.OnClickListener() {
28             @Override
29             public void onClick(View v) {
30                 // load First Fragment
31                 loadFragment(new SecondFragment());
32             }
33         });
34     }
35
36     private void loadFragment(Fragment fragment) {
37         // create a FragmentManager
38         FragmentManager fm = getFragmentManager();
39         //create a FragmentTransaction to begin the transaction and replace the Fragment
40         FragmentTransaction fragmentTransaction = fm.beginTransaction();
41         //replace the FrameLayout with new Fragment
42         fragmentTransaction.replace(R.id.frameLayout, fragment);
43         fragmentTransaction.commit(); //save the changes
44     }
}
```

8. Sekarang membutuhkan 2 fragmen dan 2 layout xml. Jadi buat dua fragmen dengan cara:
 - Klik kanan pada folder paket
 - Buat kelas dan beri nama sebagai **FirstFragment** dan **SecondFragment**
 - Tambahkan kode berikut masing-masing.

FirstFragment.class

- Pertama memekarkan tata letak dan mendapatkan referensi **Button**.

- Setelah itu melakukan activity **setOnClickListener** pada **Button** sehingga setiap kali pengguna klik tombol, pesan "**Fragmen Pertama**" ditampilkan di layar dengan menggunakan *Toast*.

```

1  package com.example.fragment;
2  import android.os.Bundle;
3  import android.view.LayoutInflater;
4  import android.view.View;
5  import android.view.ViewGroup;
6  import android.widget.Button;
7  import android.widget.Toast;
8  import android.app.Fragment;
9  import androidx.annotation.Nullable;
10 public class FirstFragment extends Fragment {
11     View view;
12     Button firstButton ;
13     @Override
14     @Override
15     public View onCreateView(LayoutInflater inflater, @Nullable ViewGroup container,
16                             Bundle savedInstanceState) {
17         //inflate the layout for this fragment
18         view = inflater.inflate(R.layout.fragment_first, container, attachToRoot: false);
19         //get the reference of Button
20         firstButton = view.findViewById(R.id.firstButton);
21         //perform setOnClickListener on secondButton
22         firstButton.setOnClickListener(new View.OnClickListener() {
23             @Override
24             public void onClick(View v) {
25                 //display a message by using a Toast
26                 Toast.makeText(getActivity(), text: "First Fragment", Toast.LENGTH_LONG).show();
27             }
28         });
29     }
30 }
```

SecondFragment.class

- Pertama memekarkan tata letak dan mendapatkan referensi **Button**.
- Setelah itu melakukan activity **setOnClickListener** pada **Button**, sehingga setiap kali pengguna klik tombol pesan "**Fragmen Kedua**" ditampilkan di layar dengan menggunakan *Toast*.

```
1 package com.example.fragment;
2 import android.os.Bundle;
3 import android.view.LayoutInflater;
4 import android.view.View;
5 import android.view.ViewGroup;
6 import android.widget.Button;
7 import android.widget.Toast;
8 import androidx.annotation.Nullable;
9 import android.app.Fragment;
10 public class SecondFragment extends Fragment {
11     View view;
12     Button SecondButton ;
13     @Override
14     public View onCreateView(LayoutInflater inflater, @Nullable ViewGroup container,
15                             Bundle savedInstanceState) {
16         //inflate the layout for this fragment
17         view = inflater.inflate(R.layout.fragment_second, container, false);
18         //get the reference of Button
19         SecondButton = view.findViewById(R.id.secondButton);
20         //perform setOnClickListener on secondButton
21         SecondButton.setOnClickListener(new View.OnClickListener() {
22             @Override
23             public void onClick(View v) {
24                 //display a message by using a Toast
25                 Toast.makeText(getActivity(), "Second Fragment", Toast.LENGTH_LONG).show();
26             }
27         });
28         return view;
29     }
30 }
```

9. Sekarang buat 2 xml layout

- Klik kanan pada res / layout → New → Layout Resource File
- Beri nama sebagai **fragment_first** dan **fragment_second**
- Tambahkan kode berikut di file masing-masing.

fragment_first.xml

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent"
5     xmlns:tools="http://schemas.android.com/tools"
6     tools:context=".FirstFragment">
```

```
7     <TextView
8         android:layout_width="wrap_content"
9         android:layout_height="wrap_content"
10        android:layout_centerHorizontal="true"
11        android:layout_marginTop="100dp"
12        android:text="this is First Fragment"
13        android:textColor="@color/black"
14        android:textSize="25sp"/>
15
16    <Button
17        android:id="@+id/firstButton"
18        android:layout_width="match_parent"
19        android:layout_height="wrap_content"
20        android:layout_centerInParent="true"
21        android:layout_marginLeft="20dp"
22        android:text="First Fragment"
23        android:layout_marginRight="20dp"
24        android:background="@color/teal_200"
25        android:textColor="@color/white"
26        android:textSize="20sp"
27        android:textStyle="bold"/>
28
29    </RelativeLayout>
```

10. Buka res → values → colors.xml

Pada langkah ini ditentukan warna yang digunakan dalam file xml.

```
1     <?xml version="1.0" encoding="utf-8"?>
2     <resources>
3         <color name="purple_200">#FFBB86FC</color>
4         <color name="purple_500">#FF6200EE</color>
5         <color name="purple_700">#FF3700B3</color>
6         <color name="teal_200">#FF03DAC5</color>
7         <color name="teal_700">#FF018786</color>
8         <color name="button_background_color">#925</color>
9         <color name="green">#0f0</color>
10        <color name="black">#000</color>
11        <color name="white">#fff</color>
12    </resources>
```

11. Open AndroidManifest.xml

- Langkah ini akan menampilkan file **Android Manifest** yang tidak melakukan apa-apa karena hanya membutuhkan satu **Activity** yaitu **MainActivity** yang sudah ditentukan di dalamnya.
- Dalam proyek membuat dua fragmen tetapi kami tidak perlu mendefinisikan fragmen dalam manifes karena fragmen adalah bagian dari acivity.

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      package="com.example.fragment">
4          <application>
5              android:allowBackup="true"
6              android:icon="@mipmap/ic_launcher"
7              android:label="Fragment"
8              android:roundIcon="@mipmap/ic_launcher_round"
9              android:supportsRtl="true"
10             android:theme="@style/Theme.Fragment">
11                 <activity android:name=".MainActivity">
12                     <intent-filter>
13                         <action android:name="android.intent.action.MAIN" />
14
15                         <category android:name="android.intent.category.LAUNCHER" />
16                     </intent-filter>
17                 </activity>
18             </application>
19         </manifest>

```

f. Hasil dan Pembahasan

Project aplikasi dari percobaan implementasi code fragment.

g. Kesimpulan

Mahasiswa berhasil meng-implementasi-kan fragment pada project individu serta memahami konsep implementasi fragment.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan implementasi pada percobaan fragment dengan penjelasan singkat	1	2	3	(4)
2	Ketepatan waktu dan ketepatan implementasi pada percobaan fragment	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat implementasi pada percobaan fragment	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan implementasi pada percobaan fragment	(1)	2	3	4
Jumlah skor					

ACARA 19 : FRAGMENT

Pokok Bahasan	: Implementasi Fragment
Acara Praktikum/Pertemuan	: Minggu 5 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep fragment.
2. Mahasiswa mampu menerapkan dan meng-implementasikan fragment pada android.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep fragment, implementasi fragment pada android, serta analisa source code.

c. Dasar Teori

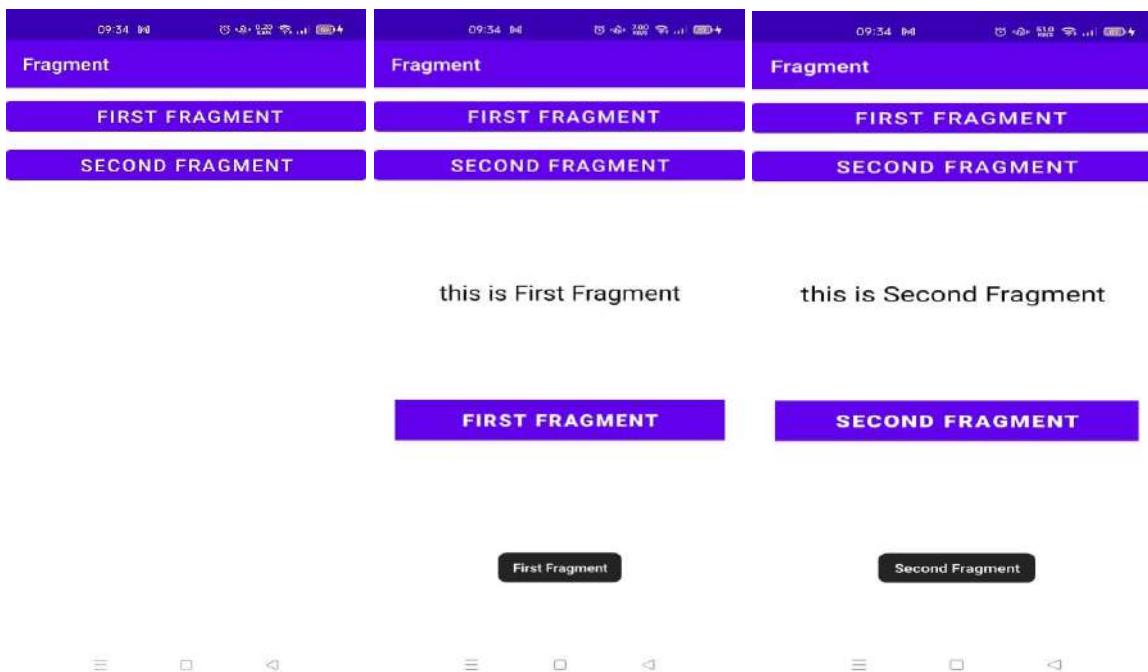
-

d. Alat dan Bahan

1. Kertas A4
2. Jaringan Internet

e. Prosedur Kerja

1. Mahasiswa melanjutkan implementasi code pada acara 18 praktikum minggu 5 pertemuan 2 sampai dengan selesai
2. Setelah selesai menuliskan code, running aplikasi dan lakukan percobaan.
3. Aplikasi memiliki dua tombol dengan nama **Fragmen Pertama** dan **Fragmen Kedua**.
4. Klik tombol pertama menunjukkan **Fragmen Pertama**
5. Klik tombol kedua menampilkan **Fragmen Kedua**
6. Kedua tombol tersebut sebenarnya menggantikan tata letak (*FrameLayout*).
7. Hasil running akan menampilkan seperti dibawah :



Gambar 1. Hasil running impelmentasi fragment

f. Hasil dan Pembahasan

Project aplikasi dari percobaan implementasi code fragment.

g. Kesimpulan

Mahasiswa berhasil meng-implementasi-kan fragment pada project individu serta memahami konsep implementasi fragment.

a. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan implementasi pada percobaan fragment dengan penjelasan singkat	1	2	3	(4)
2	Ketepatan waktu dan ketepatan implementasi pada percobaan fragment	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat implementasi pada percobaan fragment	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan implementasi pada percobaan fragment	(1)	2	3	4
Jumlah skor					

ACARA 20 : FRAGMENT

Pokok Bahasan	: Analisa Hasil Implementasi Fragment
Acara Praktikum/Pertemuan	: Minggu 5 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep fragment.
2. Mahasiswa mampu menerapkan dan meng-implementasikan fragment pada android.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep fragment, implementasi fragment pada android, serta analisa source code.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kertas A4
2. Jaringan Internet

e. Prosedur Kerja

1. Mahasiswa melakukan analisa code untuk setiap baris code dari implementasi fragment.
2. Menjelaskan fungsi dan arti dari setiap baris code.
3. Analisa hasil running dari implementasi project fragment.
4. Screenshot output setiap proses dari project fragment.
5. Tulisakan dalam bentuk laporan.

f. Hasil dan Pembahasan

Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project fragment yang di buktikan dengan screenshot pada proses percobaan.

g. Kesimpulan

Mahasiswa memahami konsep dasar fragment dengan mampu memahami setiap baris code dari implementasi fragment.

h. Rubrik Penilaian

No	Indikator	Skor*			
		1	2	3	4
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail setiap baris code pada implementasi fragment	1	2	3	4
2	Ketepatan waktu dan ketepatan dalam menjelaskan setiap baris code pada implementasi fragment	1	2	3	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan setiap baris code pada implementasi fragment	1	2	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan setiap baris code pada implementasi fragment	1	2	3	4
Jumlah skor					

ACARA 21 : INTENT

Pokok Bahasan	: Implementasi Intent
Acara Praktikum/Pertemuan	: Minggu 6 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep intent.
2. Mahasiswa mampu menerapkan dan meng-implementasikan inten pada android.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep intent, implementasi intent pada android, serta analisa source code.

c. Dasar Teori

Intent

Intent adalah sebuah kelas dalam programming android yang berfungsi untuk perpindahan halaman dan juga mekanisme untuk melakukan sebuah action dan komunikasi antar komponen aplikasi misal activity, services, dan broadcast receiver. Intent juga merupakan suatu objek yang terdapat dalam suatu activity dimana objek tersebut dapat komunikasi dengan activity yang lain, baik activity pada fungsi internal android misal seperti memanggil activity dalam satu package atau beda package yang masih berada dalam satu project.

Terdapat tiga penggunaan umum intent dalam aplikasi android yaitu:

- Memindahkan satu activity ke activity lain dengan atau tidak membawa data.
- Menjalankan background service, misalnya melakukan sinkronisasi ke server dan menjalankan proses berulang (periodic/scheduler task).
- Mengirimkan obyek broadcast ke aplikasi yang membutuhkan. Misal, ketika aplikasi membutuhkan proses menjalankan sebuah background service setiap kali aplikasi selesai melakukan booting. Aplikasi harus bisa menerima obyek broadcast yang dikirimkan oleh sistem Android untuk event booting tersebut.

Intent merupakan objek tipe **android.content.Intent**. Melalui metode **startActivity()** yang digunakan untuk memulai sebuah activity lain. Intent dibagi menjadi 2, yaitu :

- 1) **Explicit Intent** berfungsi untuk mengaktifkan komponen-komponen dalam satu aplikasi yang sama. Misalnya seperti : Berpindah Activity. Tipe Intent ini digunakan untuk menjalankan komponen dari dalam sebuah aplikasi. Explicit intent bekerja dengan menggunakan nama kelas yang dituju misal :
com.dicoding.activity.DetailActivity. umumnya intent ini digunakan untuk mengaktifkan komponen pada satu aplikasi.
- 2) **Implicit Intent** berfungsi untuk memanggil fungsi activity yang sudah ada di fungsi internal android seperti Dial Number, Open Browser dan lainnya. Tipe intent ini tidak memerlukan detail nama kelas yang ingin diaktifkan. Model ini memungkinkan komponen dari aplikasi lain bisa merespon request intent yang dijalankan. Penggunaan tipe intent ini umumnya diperuntukan untuk menjalankan fitur/fungsi dari komponen aplikasi lain. Contohnya ketika kita membutuhkan fitur untuk mengambil foto. Daripada membuat sendiri fungsi kamera, lebih baik kita menyerahkan proses tersebut pada aplikasi kamera bawaan dari peranti atau aplikasi kamera lain yang telah terinstal sebelumnya di peranti. Hal yang sama misalnya ketika kita membutuhkan fungsi berbagi konten. Kita bisa memanfaatkan intent untuk menampilkan aplikasi mana saja yang bisa menangani fitur tersebut. Implementasi implicit intent ini akan sangat memudahkan bagi pengembang agar tetap fokus pada proses bisnis inti dari aplikasi yang dikembangkan.

d. Alat dan Bahan

1. Kertas A4
2. Jaringan Internet

e. Prosedur Kerja

1. Mahasiswa mencari video refrensi tentang implementasi intent.
2. Analisa kegunaan dan manfaat intent dari video yang sudah dilihat.
3. Resume pada kertas A4.

f. Hasil dan Pembahasan

1. Dokumen berupa video refrensi yang telah di unduh dan di analisa.
2. Dokumen tugas resum analisa video dan contoh pada aplikasi yang digunakan.

g. Kesimpulan

Mahasiswa memahami konsep dasar intent dengan menerapkan contoh pada aplikasi android yang sering digunakan sehari-hari

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail tugas review video	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan dari tugas review video	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan tugas review video	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan tugas review video	(1)	2	3	4
Jumlah skor					

ACARA 22 : INTENT

Pokok Bahasan	: Implementasi Intent Implicit
Acara Praktikum/Pertemuan	: Minggu 6 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep intent.
2. Mahasiswa mampu menerapkan dan meng-implementasikan inten pada android.
3. Mahasiswa mampu membedakan penggunaan intent implicit dan intent explicit.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep intent, implementasi intent pada android, perbedaan intent implicit dan intent explicit serta analisa source code.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kertas A4
2. Kabel Data USB
3. Jaringan Internet

e. Prosedur Kerja

Intent Implicit

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project **ImpIntentApp**
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**
5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Menggunakan **Constraint Layout**, buat satu komponen edit text dan button seperti pada source code pada **activity_main.xml** di bawah ini

```

<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_height="match_parent"
    android:layout_width="match_parent"
    tools:context="com.example.impintentapp.MainActivity">
    <EditText
        android:id="@+id/editText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginEnd="8dp"
        android:layout_marginStart="8dp"
        android:layout_marginTop="60dp"
        android:ems="10"
        android:text="http://"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.575"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        android:inputType="textUri" />
    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginRight="8dp"
        android:layout_marginLeft="156dp"
        android:layout_marginTop="172dp"
        android:text="@string/visit"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText"
        android:layout_marginEnd="8dp"
        android:layout_marginStart="156dp" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

7. Tambahkan baris source code di bawah ini ke dalam class **MainActivity**.

```

public class MainActivity extends AppCompatActivity {
    Button button;
    EditText editText;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        button = findViewById(R.id.button);
        editText = findViewById(R.id.editText);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String url=editText.getText().toString();
                Intent intent=new Intent(Intent.ACTION_VIEW, Uri.parse(url));
                startActivity(intent);
            }
        });
    }
}

```

8. Terakhir pada file **AndroidManifest.xml** tambahkan tag intent-filter seperti pada gambar berikut.

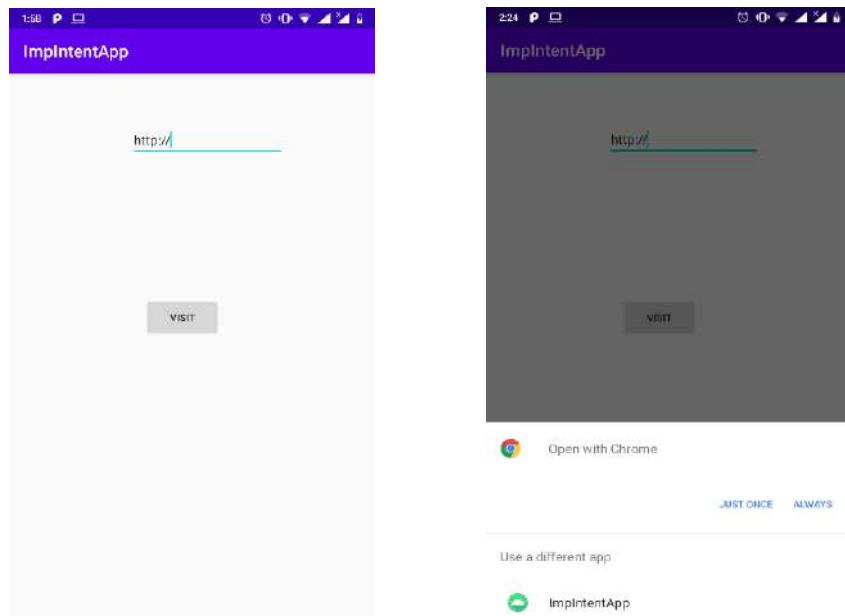
```

<intent-filter tools:ignore="AppLinkUrlError">
    <action android:name="android.intent.action.VIEW" />

    <category android:name="android.intent.category.DEFAULT" />
    <data android:scheme="http" />
</intent-filter>

```

9. Coba jalankan dan perhatikan apa yang terjadi apabila edit text diinputkan alamat url kemudian button ‘visit’ ditekan.



f. Hasil dan Pembahasan

Project aplikasi dari percobaan implementasi code intent implicit.

g. Kesimpulan

Mahasiswa berhasil meng-implementasi-kan intent implicit pada project individu serta memahami konsep implementasi intent implicit.

h. Rubrik Penilaian

No	Indikator	Skor*				
1	Ketepatan waktu dan ketepatan implementasi pada percobaan intent implicit dengan penjelasan singkat	1	2	3	4	
2	Ketepatan waktu dan ketepatan implementasi pada percobaan intent implicit	1	2	3	4	
3	Ketepatan waktu akan tetapi kurang tepat implementasi pada percobaan intent implicit	1	2	3	4	
4	Keterlambatan pengumpulan tugas dan ketidaktepatan implementasi pada percobaan intent implicit	1	2	3	4	
Jumlah skor						

ACARA 23 : INTENT

Pokok Bahasan	: Implementasi Intent Explicit
Acara Praktikum/Pertemuan	: Minggu 6 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep intent.
2. Mahasiswa mampu menerapkan dan meng-implementasikan inten pada android.
3. Mahasiswa mampu membedakan penggunaan intent implicit dan intent explicit.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep intent, implementasi intent pada android, perbedaan intent implicit dan intent explicit serta analisa source code.

c. Dasar Teori

-

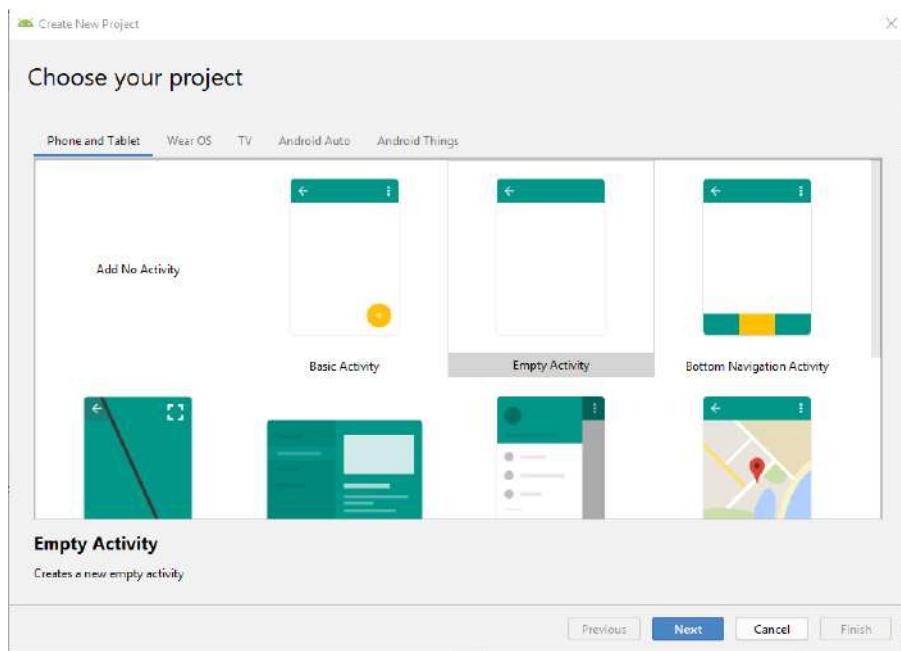
d. Alat dan Bahan

1. Kertas A4
2. Kabel Data USB
3. Jaringan Internet

e. Prosedur Kerja

Intent Explicit

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project **ExpIntentApp**
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**



5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Ketikan source code dalam **activity_main.xml** seperti di bawah ini

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.expintentapp.MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/who_are_you" />

    <TextView
        android:text="@string/i_am"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="29dp"
        android:id="@+id/textView2"
        android:layout_below="@+id/textView"
        android:layout_alignParentStart="true" />
```

```

<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:inputType="textPersonName"
    android:ems="10"
    android:id="@+id/edt_nama"
    android:layout_alignBaseline="@+id/textView2"
    android:layout_alignBottom="@+id/textView2"
    android:layout_alignParentEnd="true"
    android:layout_toEndOf="@+id/textView2" />

<Button
    android:text="SEND"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/edt_nama"
    android:layout_marginTop="26dp"
    android:id="@+id	btn_send"
    android:layout_alignParentEnd="true"
    android:layout_alignParentStart="true" />
</RelativeLayout>

```

7. Kemudian tambahkan 1 activity baru pada project, dengan nama **ActivitySecond**, dan ketikan source code pada **activity_second.xml** seperti berikut ini

```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_2"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.expintentapp.SecondActivity">

    <TextView
        android:text="TextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_alignParentStart="true"
        android:textSize="20sp"
        android:layout_marginStart="12dp"
        android:textStyle="bold"
        android:id="@+id/txtHello" />

    <TextView
        android:text="@string/this_is_second_activity"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/txtHello"
        android:layout_alignStart="@+id/txtHello"
        android:id="@+id/textView3" />
</RelativeLayout>

```

8. Pada class **MainActivity**, tambahkan source code berikut ini

```

EditText name;
Button btnSend;

private String KEY_NAME = "NAMA";

```

9. Masih di **MainActivity.java**, tambahkan baris code di bawah ini pada method **onCreate**

```

name = (EditText) findViewById(R.id.edt_nama);
btnSend = (Button) findViewById(R.id.btn_send);

btnSend.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        try {
            String nama = name.getText().toString();
            if (nama != ""){
                Intent i = new Intent(MainActivity.this, SecondActivity.class);
                i.putExtra(KEY_NAME, nama);
                startActivity(i);

            } else {
                Toast.makeText(getApplicationContext(), "YOU NEED TO FILL YOUR
NAME", Toast.LENGTH_SHORT);
            }
        } catch (Exception e){
            e.printStackTrace();
            Toast.makeText(getApplicationContext(), "ERROR, TRY AGAIN
!", Toast.LENGTH_SHORT);
        }
    }
});

```

10. Lalu pada activity berikutnya, tambahkan source code berikut ini untuk mengambil data dari activity sebelumnya

```

TextView txtHello;
private String nama;
private String KEY_NAME = "NAMA";

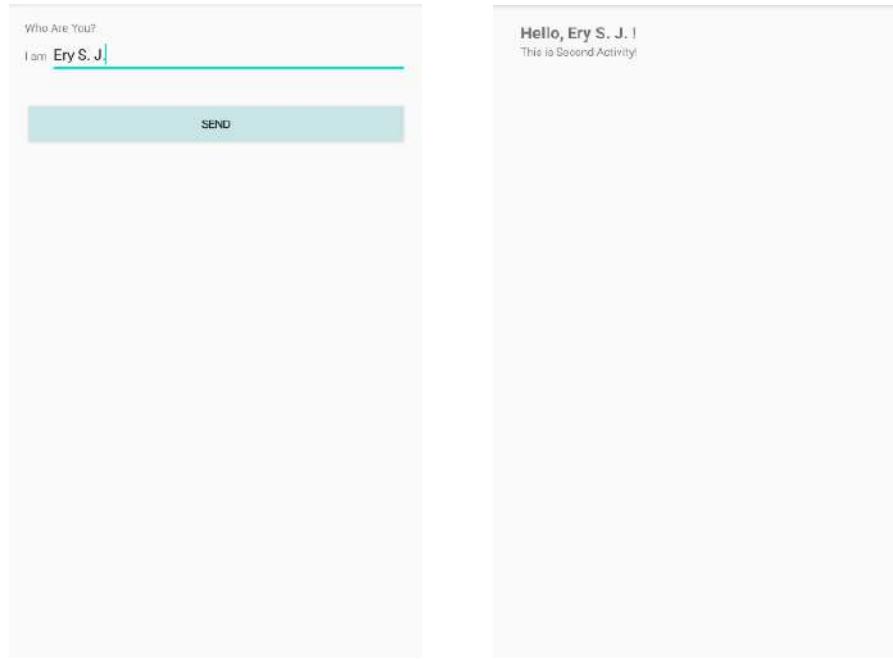
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_second);

    txtHello = (TextView) findViewById(R.id.txtHello);

    Bundle extras = getIntent().getExtras();
    nama = extras.getString(KEY_NAME);
    txtHello.setText("Hello, " + nama + " !");
}

```

11. Build dan run project tersebut, lalu perhatikan hasilnya.



f. Hasil dan Pembahasan

Project aplikasi dari percobaan implementasi code intent explicit.

g. Kesimpulan

Mahasiswa berhasil meng-implementasi-kan intent explicit pada project individu serta memahami konsep implementasi intent explicit.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan implementasi pada percobaan intent explicit dengan penjelasan singkat	1	2	3	(4)
2	Ketepatan waktu dan ketepatan implementasi pada percobaan intent explicit	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat implementasi pada percobaan intent explicit	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan implementasi pada percobaan intent explicit	(1)	2	3	4
Jumlah skor					

ACARA 24 : INTENT

Pokok Bahasan	: Analisa Code Intent Implicit dan Intent Explicit
Acara Praktikum/Pertemuan	: Minggu 6 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep intent.
2. Mahasiswa mampu menerapkan dan meng-implementasikan inten pada android.
3. Mahasiswa mampu membedakan penggunaan intent implicit dan intent explicit.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep intent, implementasi intent pada android, perbedaan intent implicit dan intent explicit serta analisa source code.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kertas A4
2. Jaringan Internet

e. Prosedur Kerja

1. Mahasiswa melakukan analisa code untuk setiap baris code dari implementasi intent implicit dan intent explicit.
2. Menjelaskan fungsi dan arti dari setiap baris code.
3. Analisa hasil running dari implementasi project intent implicit dan intent explicit.
4. Screenshot output setiap proses dari project intent implicit dan intent explicit.
5. Tulisakan dalam bentuk laporan.

f. Hasil dan Pembahasan

Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project intent implicit dan intent explicit yang di buktikan dengan screenshot pada proses percobaan.

g. Kesimpulan

Mahasiswa memahami konsep dasar intent implicit dan intent explicit dengan mampu memahami setiap baris code dari implementasi intent implicit dan intent explicit.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail setiap baris code pada implemenasi intent implicit dan intent explicit	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan setiap baris code pada implemenasi intent implicit dan intent explicit	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan setiap baris code pada implemenasi intent implicit dan intent explicit	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan setiap baris code pada implemenasi intent implicit dan intent explicit	(1)	2	3	4
Jumlah skor					

ACARA 25 : MANAJEMEN FILE PADA ANDROID DAN SQLITE

Pokok Bahasan	: Konsep Manajemen File pada Android
Acara Praktikum/Pertemuan	: Minggu 7 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep manajemen file.
2. Mahasiswa mampu menerapkan dan meng-implementasikan manajemen file pada android.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep manajemen file, implementasi manajemen file pada android, serta analisa source code.

c. Dasar Teori

Manajemen File

Read dan Write file pada Android akan diperlukan jika ingin untuk membaca data dari SD Card, menyimpan data aplikasi berupa text file. Data disimpan dalam file yang ditentukan oleh pengguna itu sendiri dan pengguna dapat mengakses file ini. File-file ini hanya dapat diakses hingga aplikasi ditutup atau yg telah memasang kartu SD pada perangkat.

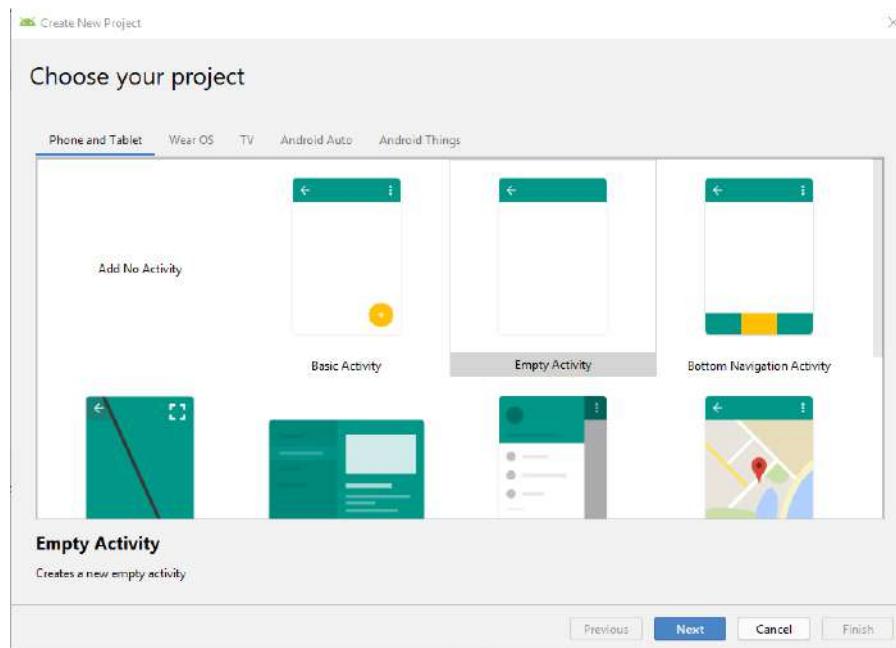
- `getExternalStorageDirectory()` – Cara lama untuk mengakses penyimpanan eksternal di API Level kurang dari 7. Ini mutlak sekarang dan tidak disarankan. Ini secara langsung mendapatkan referensi ke direktori root penyimpanan eksternal atau kartu SD.
- `getExternalFilesDir(String type)` – Ini adalah cara yang disarankan untuk memungkinkan kami membuat file pribadi khusus untuk aplikasi dan file dihapus saat aplikasi dicopot. Contohnya adalah data pribadi aplikasi.
- `getExternalStoragePublicDirectory()` – Ini adalah cara yang disarankan saat ini yang memungkinkan kami menyimpan file untuk umum dan tidak terhapus dengan pencopotan aplikasi. Contoh gambar yang diklik oleh kamera ada bahkan kami menghapus aplikasi kamera.

d. Alat dan Bahan

1. Media Penyimpanan
2. Kertas A4
3. Jaringan Internet

e. Prosedur Kerja

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project ManagementFile
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**



5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Buat file **MainActivity.java** dan **Main2Activity.java**

f. Hasil dan Pembahasan

Folder project aplikasi dari implementasi code manajemen file.

g. Kesimpulan

Mahasiswa berhasil memahami konsep manajemen file serta membuat folder project aplikasi dari untuk implementasi code manajemen file.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Pemahaman konsep dasar manajemen file dengan penjelasan detail	1	2	3	(4)
2	Pemahaman konsep dasar manajemen file dengan penjelasan singkat	1	2	(3)	4
3	Pemahaman konsep dasar manajemen file dengan penjelasan yang kurang	1	(2)	3	4
4	Tidak pemaham konsep dasar manajemen file dengan penjelasan singkat	(1)	2	3	4
Jumlah skor					

ACARA 26 : MANAJEMEN FILE PADA ANDROID DAN SQLITE

Pokok Bahasan	: Implementasi Manajemen File pada Android
Acara Praktikum/Pertemuan	: Minggu 7 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep manajemen file.
2. Mahasiswa mampu menerapkan dan meng-implementasikan manajemen file pada android.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep manajemen file, implementasi manajemen file pada android, serta analisa source code.

c. Dasar Teori

-

d. Alat dan Bahan

1. Media Penyimpanan
2. Kertas A4
3. Jaringan Internet

e. Prosedur Kerja

Manajemen File

1. Mengubah source di bawah pada **MainActivity.java**

```
3   import androidx.appcompat.app.AppCompatActivity;
4   import androidx.core.app.ActivityCompat;
5   import android.Manifest;
6   import android.content.Intent;
7   import android.os.Bundle;
8   import android.os.Environment;
9   import android.view.View;
10  import android.widget.EditText;
11  import android.widget.Toast;
12  import java.io.File;
13  import java.io.FileOutputStream;
14  import java.io.IOException;
```

```

15
16     public class MainActivity extends AppCompatActivity {
17         EditText editText;
18         private int STORAGE_PERMISSION_CODE = 23;
19
20         @Override
21         protected void onCreate(Bundle savedInstanceState) {
22             super.onCreate(savedInstanceState);
23             setContentView(R.layout.activity_main);
24             editText = (EditText) findViewById(R.id.editText2);
25         }
26
27         public void next(View view) {
28             Intent intent = new Intent( packageContext: MainActivity.this, Main2Activity.class );
29             startActivity(intent);
30         }
31
32         public void savePublic(View view) {
33             //Permission to access external storage
34             ActivityCompat.requestPermissions( activity: this, new String[]{Manifest.permission.READ_EXTERNAL_STORAGE}, STORAGE_PERMISSION_CODE );
35             String info = editText.getText().toString();
36             File folder = Environment.getExternalStoragePublicDirectory(Environment.DIRECTORY_DOWNLOADS); // Folder Name
37             File myFile = new File(folder, child: "myData1.txt"); // Filename
38             writeData(myFile, info);
39             editText.setText("");
40         }
41
42         public void savePrivate(View view) {
43             String info = editText.getText().toString();
44             File folder = getExternalFilesDir( type: "arvita"); // Folder Name
45             File myFile = new File(folder, child: "myData2.txt"); // Filename
46             writeData(myFile, info);
47             editText.setText("");
48         }
49
50         private void writeData(File myFile, String data) {
51             FileOutputStream fileOutputStream = null;
52             try {
53                 System.out.println("TES");
54                 fileOutputStream = new FileOutputStream(myFile);
55                 fileOutputStream.write(data.getBytes());
56                 Toast.makeText( .context: this, (text: "Done" + myFile.getAbsolutePath(), Toast.LENGTH_SHORT).show();
57             } catch (Exception e) {
58                 e.printStackTrace();
59             } finally {
60                 if (fileOutputStream != null) {
61                     try {
62                         fileOutputStream.close();
63                     } catch (IOException e) {
64                         e.printStackTrace();
65                     }
66                 }
67             }
68         }
69     }
70 }
```

2. Menambahkan source di bawah pada **Main2Activity.java**

```
3 import android.content.Intent;
4 import android.os.Bundle;
5 import android.os.Environment;
6 import android.view.View;
7 import android.widget.TextView;
8 import androidx.appcompat.app.AppCompatActivity;
9 import java.io.File;
10 import java.io.FileInputStream;
11 import java.io.IOException;
12
13 public class Main2Activity extends AppCompatActivity {
14     TextView showText;
15
16     @Override
17     protected void onCreate(Bundle savedInstanceState) {
18         super.onCreate(savedInstanceState);
19         setContentView(R.layout.activity_main2);
20         showText = (TextView) findViewById(R.id.getText);
21     }
22
23     public void back(View view) {
24         Intent intent = new Intent(packageContext: Main2Activity.this, MainActivity.class);
25         startActivity(intent);
26     }
27
28     public void getPublic(View view) {
29         File folder = Environment.getExternalStoragePublicDirectory(Environment.DIRECTORY_DOWNLOADS); // Folder Name
30         File myFile = new File(folder, child: "myData1.txt"); // Filename
31         String text = getdata(myFile);
32         if (text != null) {
33             showText.setText(text);
34         } else {
35             showText.setText("No Data");
36         }
37     }
38
39     public void getPrivate(View view) {
40         File folder = getExternalFilesDir( type: "private"); // Folder Name
41         File myFile = new File(folder, child: "myData2.txt"); // Filename
42         String text = getdata(myFile);
43         if (text != null) {
44             showText.setText(text);
45         } else {
46             showText.setText("No Data");
47         }
48     }
49
50     private String getdata(File myfile) {
51         FileInputStream fileInputStream = null;
52         try {
53             fileInputStream = new FileInputStream(myfile);
54             int i = -1;
55             StringBuffer buffer = new StringBuffer();
```

```
56     }
57     while ((i = fileInputStream.read()) != -1) {
58         buffer.append((char) i);
59     }
60     return buffer.toString();
61 } catch (Exception e) {
62     e.printStackTrace();
63 } finally {
64     if (fileInputStream != null) {
65         try {
66             fileInputStream.close();
67         } catch (IOException e) {
68             e.printStackTrace();
69         }
70     }
71 }
72 return null;
73 }
74 }
```

3. Mengubah source di bawah pada **activity_main.xml**

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:id="@+id/activity_main"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:paddingBottom="@dimen/activity_vertical_margin"
8     android:paddingLeft="@dimen/activity_horizontal_margin"
9     android:paddingRight="@dimen/activity_horizontal_margin"
10    android:paddingTop="@dimen/activity_vertical_margin"
11    tools:context="com.example.storage.MainActivity">
12    <TextView
13        android:id="@+id/textView"
14        android:layout_width="match_parent"
15        android:layout_height="wrap_content"
16        android:layout_alignParentTop="true"
17        android:layout_marginTop="46dp"
18        android:gravity="center"
19        android:text="@string/add_text"
20        android:textSize="18sp"
21        android:textStyle="bold|italic" />
```

```
22     <Button
23         android:id="@+id/button4"
24         android:layout_width="wrap_content"
25         android:layout_height="wrap_content"
26         android:layout_alignLeft="@+id/button"
27         android:layout_alignParentBottom="true"
28         android:layout_alignStart="@+id/button"
29         android:layout_marginBottom="52dp"
30         android:layout_marginLeft="96dp"
31         android:layout_marginStart="96dp"
32         android:onClick="next"
33         android:text="@string/click_to_view" />
34     <Button
35         android:id="@+id/button2"
36         android:layout_width="wrap_content"
37         android:layout_height="wrap_content"
38         android:layout_above="@+id/button4"
39         android:layout_alignParentEnd="true"
40         android:layout_alignParentRight="true"
41         android:layout_marginBottom="22dp"
42         android:layout_marginEnd="32dp"
43         android:layout_marginRight="32dp"
44         android:onClick="savePublic"
45         android:text="@string/save_as_public" />
46     <Button
47         android:id="@+id/button"
48         android:layout_width="wrap_content"
49         android:layout_height="wrap_content"
50         android:layout_alignBaseline="@+id/button2"
51         android:layout_alignBottom="@+id/button2"
52         android:layout_alignParentLeft="true"
53         android:layout_alignParentStart="true"
54         android:layout_marginLeft="24dp"
55         android:layout_marginStart="24dp"
56         android:onClick="savePrivate"
57         android:text="@string/save_as_private" />
```

```
58     <EditText  
59         android:id="@+id/editText2"  
60         android:layout_width="match_parent"  
61         android:layout_height="wrap_content"  
62         android:layout_alignParentLeft="true"  
63         android:layout_alignParentStart="true"  
64         android:layout_below="@+id/textView"  
65         android:layout_marginTop="16dp"  
66         android:ems="10"  
67         android:gravity="center_vertical|center"  
68         android:inputType="textMultiLine" />  
69     </RelativeLayout>
```

4. Menambahkan source di bawah pada res → value → dimens.xml

```
1  <resources>  
2      <!-- Default screen margins, per the Android Design guidelines. --&gt;<br/>3      <dimen name="activity_horizontal_margin">16dp</dimen>  
4      <dimen name="activity_vertical_margin">16dp</dimen>  
5  </resources>  
6
```

5. Merubah source di bawah pada res → value → color.xml

```
1  <?xml version="1.0" encoding="utf-8"?>  
2  <resources>  
3      <color name="colorPrimary">#3F51B5</color>  
4      <color name="colorPrimaryDark">#303F9F</color>  
5      <color name="colorAccent">#FF4081</color>  
6  </resources>  
7
```

6. Menambahkan source di bawah pada activity_main2.xml

```
1  <?xml version="1.0" encoding="utf-8"?>  
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
3      xmlns:tools="http://schemas.android.com/tools"  
4      android:id="@+id/activity_main2"  
5      android:layout_width="match_parent"  
6      android:layout_height="match_parent"  
7      android:paddingBottom="@dimen/activity_vertical_margin"  
8      android:paddingLeft="@dimen/activity_horizontal_margin"  
9      android:paddingRight="@dimen/activity_horizontal_margin"  
10     android:paddingTop="@dimen/activity_vertical_margin"  
11     tools:context="com.example.storage.MainActivity">
```

```
12    <TextView  
13        android:id="@+id/getText"  
14        android:layout_width="match_parent"  
15        android:layout_height="wrap_content"  
16        android:layout_alignParentLeft="true"  
17        android:layout_alignParentStart="true"  
18        android:layout_alignParentTop="true"  
19        android:layout_marginTop="33dp"  
20        android:gravity="center"  
21        android:text=""  
22        android:textSize="18sp"  
23        android:textStyle="bold|italic" />  
24    <Button  
25        android:id="@+id/button3"  
26        android:layout_width="wrap_content"  
27        android:layout_height="wrap_content"  
28        android:layout_above="@+id/button5"  
29        android:layout_alignParentEnd="true"  
30        android:layout_alignParentRight="true"  
31        android:onClick="getPublic"  
32        android:text="@string/show_public_data" />  
33    <Button  
34        android:id="@+id/button2"  
35        android:layout_width="wrap_content"  
36        android:layout_height="wrap_content"  
37        android:layout_alignParentLeft="true"  
38        android:layout_alignParentStart="true"  
39        android:layout_below="@+id/getText"  
40        android:layout_marginTop="178dp"  
41        android:onClick="getPrivate"  
42        android:text="@string/show_private_data"  
43        tools:ignore="UnknownId" />  
44    <Button  
45        android:id="@+id/button5"  
46        android:layout_width="wrap_content"  
47        android:layout_height="wrap_content"  
48        android:layout_below="@+id/button2"  
49        android:layout_centerHorizontal="true"  
50        android:layout_marginTop="59dp"  
51        android:onClick="back"  
52        android:text="@string/back" />  
53    </RelativeLayout>
```

7. Menambahkan source di bawah pada res → value → string.xml

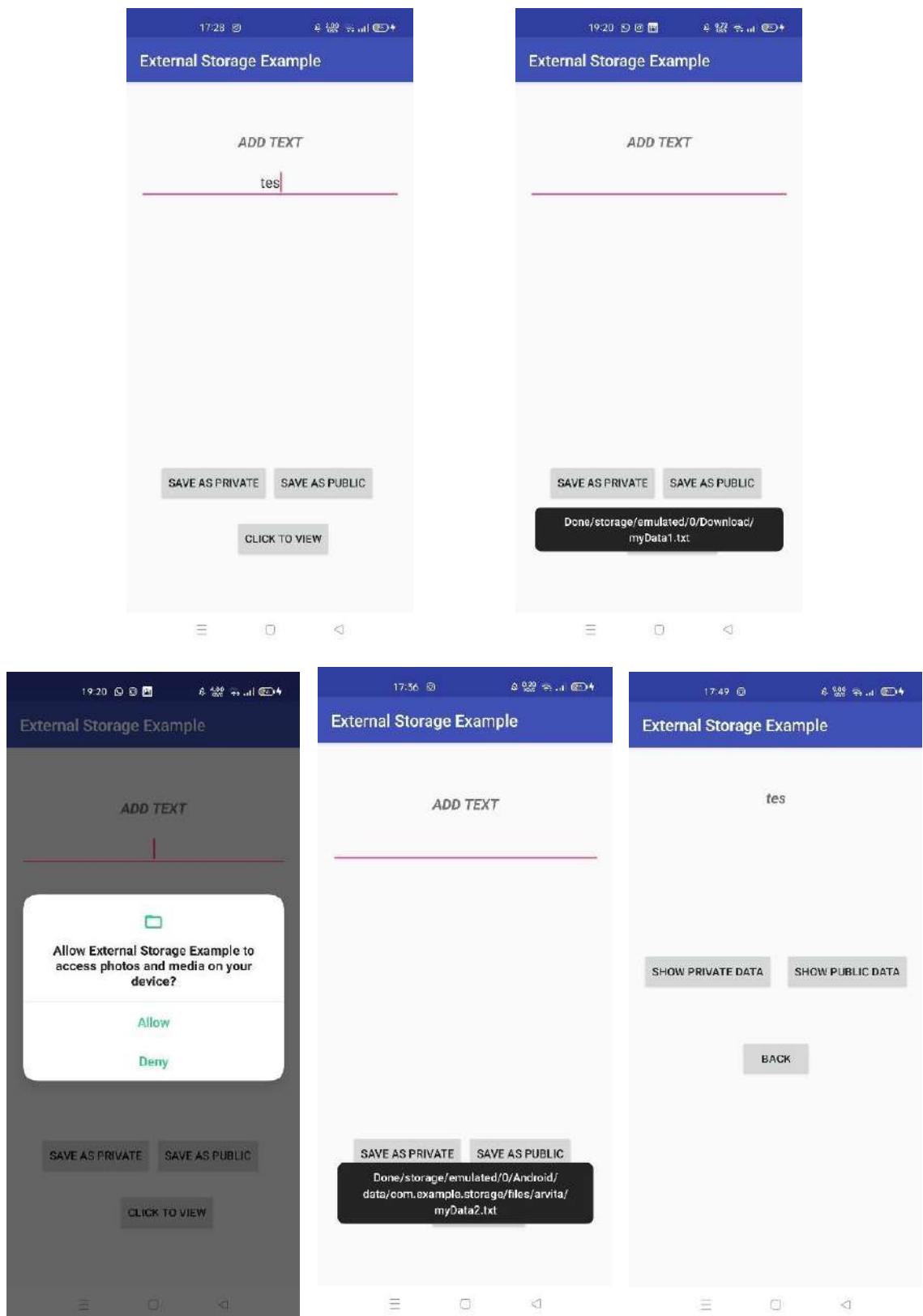
```
1  <resources>
2      <string name="app_name">External Storage Example</string>
3      <string name="back">Back</string>
4      <string name="show_private_data">SHOW PRIVATE DATA</string>
5      <string name="show_public_data">SHOW PUBLIC DATA</string>
6      <string name="add_text">ADD TEXT</string>
7      <string name="click_to_view">Click To View</string>
8      <string name="save_as_private">SAVE AS PRIVATE</string>
9      <string name="save_as_public">SAVE AS PUBLIC</string>
10
11 </resources>
12
```

8. Menambahkan source di bawah pada AndroidManifest.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      package="com.example.storage">
4          <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
5          <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
6
7          <application
8              android:allowBackup="true"
9              android:icon="@mipmap/ic_launcher"
10             android:label="External Storage Example"
11             android:roundIcon="@mipmap/ic_launcher_round"
12             android:supportsRtl="true"
13             android:theme="@style/AppTheme">
14                 <activity android:name=".MainActivity">
15                     <intent-filter>
16                         <action android:name="android.intent.action.MAIN" />
17
18                         <category android:name="android.intent.category.LAUNCHER" />
19                     </intent-filter>
20                 </activity>
21                 <activity android:name=".Main2Activity"></activity>
22             </application>
23
24 </manifest>
```

9. Build dan run project tersebut, lalu perhatikan hasilnya.

Hasil run project akan menghasilkan layout seperti di halaman berikutnya:



Gambar 1. Menghasilkan Layout Manajemen File

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan implementasi code manajement file.
2. Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project manajement file yang di buktikan dengan screenshot pada proses percobaan.

g. Kesimpulan

Mahasiswa berhasil meng-implementasi-kan manajement file pada project individu serta memahami konsep implementasi manajement file.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail setiap baris code pada project manajement file	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan dengan singkat setiap baris code pada project manajement file	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan project manajement file	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan project manajement file	(1)	2	3	4
Jumlah skor					

ACARA 27 : MANAJEMEN FILE PADA ANDROID DAN SQLITE

Pokok Bahasan	: Konsep SQLite
Acara Praktikum/Pertemuan	: Minggu 7 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep SQLite.
2. Mahasiswa mampu menerapkan dan meng-implementasikan SQLite pada android.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep SQLite, implementasi SQLite pada android, serta analisa source code.

c. Dasar Teori

SQLite

SQLite itu merupakan sebuah Database yang bersifat ACID-compliant dan memiliki ukuran pustaka kode yang relatif kecil, ditulis dalam bahasa C. SQLite merupakan proyek yang bersifat public domain yang dikerjakan oleh D. Richard Hipp. SQLite adalah sebuah open source database yang telah ada cukup lama, cukup stabil, dan sangat terkenal pada perangkat kecil, termasuk Android. Android menyediakan database relasional yang ringan untuk setiap aplikasi menggunakan SQLite. Aplikasi dapat mengambil keuntungan dari itu untuk mengatur relational database engine untuk menyimpan data secara aman dan efisien.

1) Teknik Penyimpanan pada Android

Dalam sistem android memiliki beberapa teknik untuk melakukan penyimpanan data. Teknik yang umum di gunakan adalah sebagai berikut :

- ***Shared Preferences***, yaitu menyimpan data beberapa nilai (value) dalam bentuk groups key yang dikenal dengan preferences.
- ***Files***, yaitu menyimpan data dalam bentuk file, dapat berupa menulis ke file atau membaca dari file.
- ***SQLite Database***, yaitu menyimpan data dalam bentuk databases.

- **Content Providers**, yaitu menyimpan data dalam bentuk content providers service.

Pada bagian ini akan fokus membahas bagian databases. Tidak ada database yang otomatis disediakan oleh android, jika menggunakan SQLite harus meng-create database sendiri, mendefinisikan tabelnya, serta index dan datanya. Untuk membuat dan membuka database yang paling baik adalah menggunakan libraries **Important android.database.sqlite.SQLiteOpenHelper** yang menyediakan 3 (tiga) metode yaitu :

- **Constructor**

Menyediakan representasi versi dari database dan skema database yang kita gunakan.

- **onCreate()**

Menyediakan SQLiteDatabase object yang kita gunakan akan definisi tabel dan inisialisasi data.

- **onUpgrade()**

Menyediakan fasilitas konversi database dari database versi yang lama ke database.

2) Relasi antara database SQLite dengan Android

Untuk Android, SQLite dijadikan satu di dalam Android runtime, sehingga setiap aplikasi Android dapat membuat basis data SQLite. Karena SQLite menggunakan antarmuka SQL, cukup mudah untuk digunakan orang – orang dengan pengalaman lain yang berbasis databases. Terdapat beberapa alasan mengapa SQLite sangat cocok untuk pengembangan aplikasi Android, yaitu :

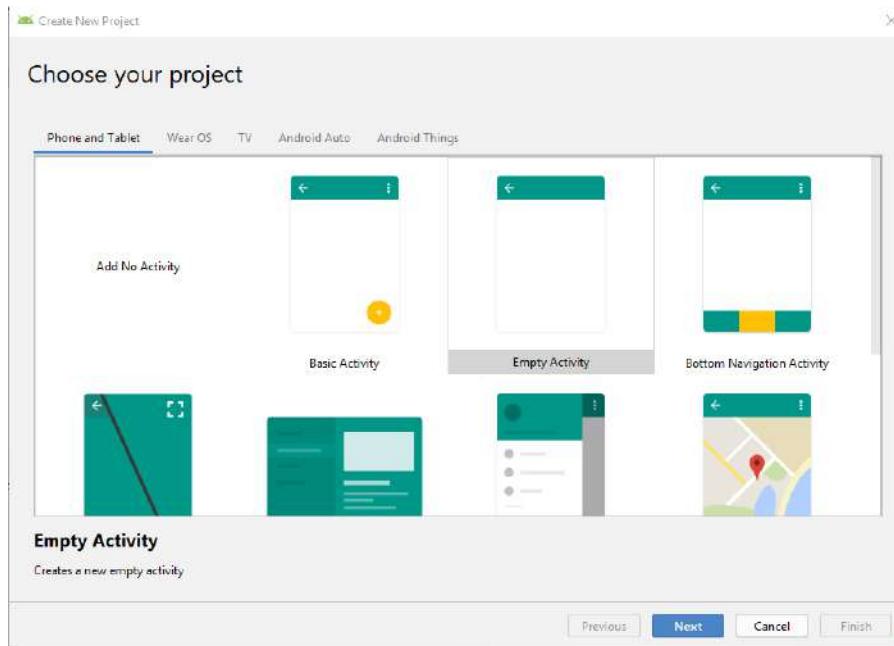
- Database dengan konfigurasi nol. Artinya tidak ada konfigurasi database untuk para developer. Ini membuatnya relatif mudah digunakan.
- Tidak memiliki server.
- Tidak ada proses database SQLite yang berjalan. Pada dasarnya satu set libraries menyediakan fungsionalitas database.
- Single-file database. Ini membuat keamanan database secara langsung. Open source. Hal ini membuat developer mudah dalam pengembangan aplikasi.

d. Alat dan Bahan

1. Media Penyimpanan
2. Kertas A4
3. Jaringan Internet

e. Prosedur Kerja

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project ImplementasiSQLite
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**



5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Buat file **MainActivity.java** dan **Main2Activity.java**

f. Hasil dan Pembahasan

Folder project aplikasi dari percobaan implementasi code SQLite.

g. Kesimpulan

Mahasiswa berhasil memahami konsep SQLite serta membuat folder project aplikasi dari untuk implementasi code SQLite.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Pemahaman konsep dasar SQLite dengan penjelasan detail	1	2	3	(4)
2	Pemahaman konsep dasar SQLite dengan penjelasan singkat	1	2	(3)	4
3	Pemahaman konsep dasar SQLite dengan penjelasan yang kurang	1	(2)	3	4
4	Tidak pemaham konsep dasar SQLite dengan penjelasan singkat	(1)	2	3	4
Jumlah skor					

ACARA 28 : MANAJEMEN FILE PADA ANDROID DAN SQLITE

Pokok Bahasan	: Implementasi SQLite
Acara Praktikum/Pertemuan	: Minggu 7 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep SQLite.
2. Mahasiswa mampu menerapkan dan meng-implementasikan SQLite pada android.

b. Indikator

Keberhasilan mahasiswa dalam memahami konsep SQLite, implementasi SQLite pada android, serta analisa source code.

c. Dasar Teori

-

d. Alat dan Bahan

1. Media Penyimpanan
2. Kertas A4
3. Jaringan Internet

e. Prosedur Kerja

1. Buat file **DataHelper.java**

DataHelper.java adalah proses pembuatan Database SQLiteSimpan dengan Nama DataHelper.java

```
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;

public class DataHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "biодатадирі.db";
    private static final int DATABASE_VERSION = 1;
    public DataHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
        // TODO Auto-generated constructor stub
    }
}
```

```

@Override
public void onCreate(SQLiteDatabase db) {
    // TODO Auto-generated method stub
    String sql = "create table biodata(no integer primary key, nama text null, tgl text null, jk text null, alamat text null);";
    Log.d( tag, "Data", msg: "onCreate: " + sql);
    db.execSQL(sql);
    sql = "INSERT INTO biodata (no, nama, tgl, jk, alamat) VALUES ('1', 'Darsiwan', '1996-07-12', 'Laki-laki','Indramayu');";
    db.execSQL(sql);
}

@Override
public void onUpgrade(SQLiteDatabase arg0, int arg1, int arg2) {
    // TODO Auto-generated method stub
}

```

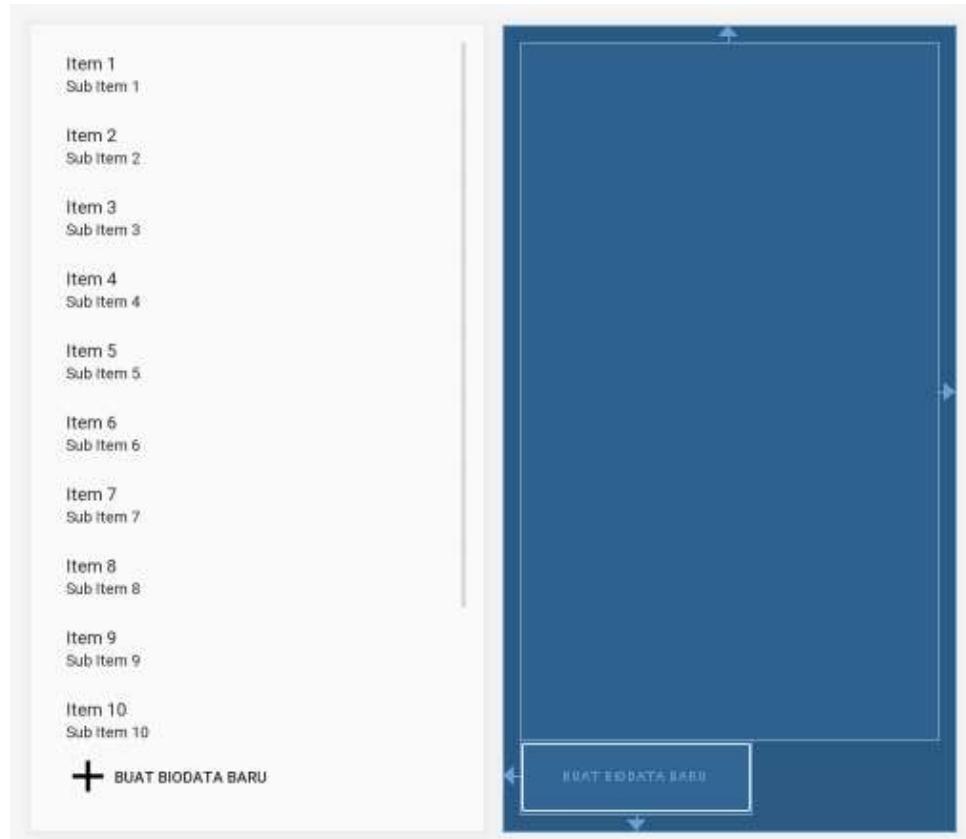
2. Buat layout xml pada **activity_main.xml**

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    tools:context="com.example.darsiwan.crudsqliite.MainActivity">

    <ListView
        android:id="@+id/listView1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentEnd="true"
        android:layout_above="@+id/button2">
    </ListView>
    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        style="?android:attr/borderlessButtonStyle"
        android:drawableLeft="@drawable/icon_add"
        android:text="Buat Biodata Baru"
        android:layout_alignParentBottom="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true" />
</RelativeLayout>

```



Gambar 2. Layout activity_main

3. Tuliskan source di bawah pada **MainActivity.java**

```
package com.example.darsiwani.crudsqliite;

import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.Menu;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ListView;
```

```

public class MainActivity extends AppCompatActivity {
    String[] daftar;
    ListView ListView01;
    Menu menu;
    protected Cursor cursor;
    DataHelper dbcenter;
    public static MainActivity ma;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        Button btn=(Button)findViewById(R.id.button2);

        btn.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                Intent inte = new Intent( mContext: MainActivity.this, BuatBiodata.class);
                startActivity(inte);
            }
        });
    }

    ma = this;
    dbcenter = new DataHelper( context: this);
    RefreshList();
}

public void RefreshList(){
    SQLiteDatabase db = dbcenter.getReadableDatabase();
    cursor = db.rawQuery( sql: "SELECT * FROM biodata", selectionArgs: null);
    daftar = new String[cursor.getCount()];
    cursor.moveToFirst();
    for (int cc=0; cc < cursor.getCount(); cc++){
        cursor.moveToPosition(cc);
        daftar[cc] = cursor.getString( 1).toString();
    }
    ListView01 = (ListView)findViewById(R.id.listView1);
    ListView01.setAdapter(new ArrayAdapter( context: this, android.R.layout.simple_list_item_1, daftar));
    ListView01.setSelected(true);
    ListView01.setOnItemClickListener(new OnItemClickListener() {
}

```

```

        public void onItemClick(AdapterView arg0, View arg1, int arg2, long arg3) {
            final String selection = daftar[arg2]; //.getItemAtPosition(arg2).toString();
            final CharSequence[] dialogitem = {"Lihat Biodata", "Update Biodata", "Hapus Biodata"};
            AlertDialog.Builder builder = new AlertDialog.Builder( context: MainActivity.this);
            builder.setTitle("Pilihan");
            builder.setItems(dialogitem, new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int item) {
                    switch(item){
                        case 0 :
                            Intent i = new Intent(getApplicationContext(), LihatBiodata.class);
                            i.putExtra( name: "nama", selection);
                            startActivity(i);
                            break;
                        case 1 :
                            Intent in = new Intent(getApplicationContext(), UpdateBiodata.class);
                            in.putExtra( name: "nama", selection);
                            startActivity(in);
                            break;
                        case 2 :
                            SQLiteDatabase db = dbcenter.getWritableDatabase();
                            db.execSQL("delete from biodata where nama = '"+selection+"'");
                            RefreshList();
                            break;
                    }
                }
            });
            builder.create().show();
        });
    });
    ((ArrayAdapter)ListView01.getAdapter()).notifyDataSetInvalidated();
}
}

```

4. Buat 3 **Empty Activity** baru, buat nama *Layout* sebagai berikut :

- activity_buat_biodata.xml
- activity_lihat_biodata.xml
- activity_update_biodata.xml

Dan Activity Name :

- BuatBiodata.java
- LihatBiodata.java
- UpdateBiodata.java

5. Setelah ke 3 Activity tadi di buat tuliskan code pada layout xml dengan nama masing-masing:

- **activity_buat_biodata.xml**

```
|<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
|    xmlns:tools="http://schemas.android.com/tools"
|    android:layout_width="match_parent"
|    android:layout_height="match_parent"
|    android:paddingBottom="16dp"
|    android:paddingLeft="16dp"
|    android:paddingRight="16dp"
|    android:paddingTop="16dp"
|    tools:context=".BuatBiodata" >
|
|    <EditText
|        android:id="@+id/editText1"
|        android:layout_width="match_parent"
|        android:layout_height="wrap_content"
|        android:layout_alignLeft="@+id/textView1"
|        android:layout_below="@+id/textView1"
|        android:inputType="number"
|        android:maxLength="10">
|
|        <requestFocus />
|    </EditText>
|
|    <TextView
|        android:id="@+id/textView1"
|        android:layout_width="wrap_content"
|        android:layout_height="wrap_content"
|        android:layout_alignParentLeft="true"
|        android:layout_alignParentTop="true"
|        android:text="Nomor" />
|
|    <TextView
|        android:id="@+id/textView2"
|        android:layout_width="wrap_content"
|        android:layout_height="wrap_content"
|        android:layout_alignLeft="@+id/editText1"
|        android:layout_below="@+id/editText1"
|        android:layout_marginTop="10dp"
|        android:text="Nama" />
```

```
<EditText
    android:id="@+id/editText2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView2"
    android:layout_below="@+id/textView2"
    android:inputType="text"
    android:maxLength="20"/>

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText2"
    android:layout_below="@+id/editText2"
    android:layout_marginTop="10dp"
    android:text="Tanggal Lahir" />

<EditText
    android:id="@+id/editText3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView3"
    android:layout_below="@+id/textView3"
    android:inputType="date"/>

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText3"
    android:layout_below="@+id/editText3"
    android:layout_marginTop="10dp"
    android:text="Jenis Kelamin" />

<EditText
    android:id="@+id/editText4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView4"
    android:layout_below="@+id/textView4"
    android:inputType="text"
    android:maxLength="1"
    android:hint="L atau P"/>

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText4"
    android:layout_below="@+id/editText4"
    android:layout_marginTop="10dp"
    android:text="Alamat" />

<EditText
    android:id="@+id/editText5"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView5"
    android:layout_below="@+id/textView5"
    android:maxLength="100"
    android:inputType="text"/>
```

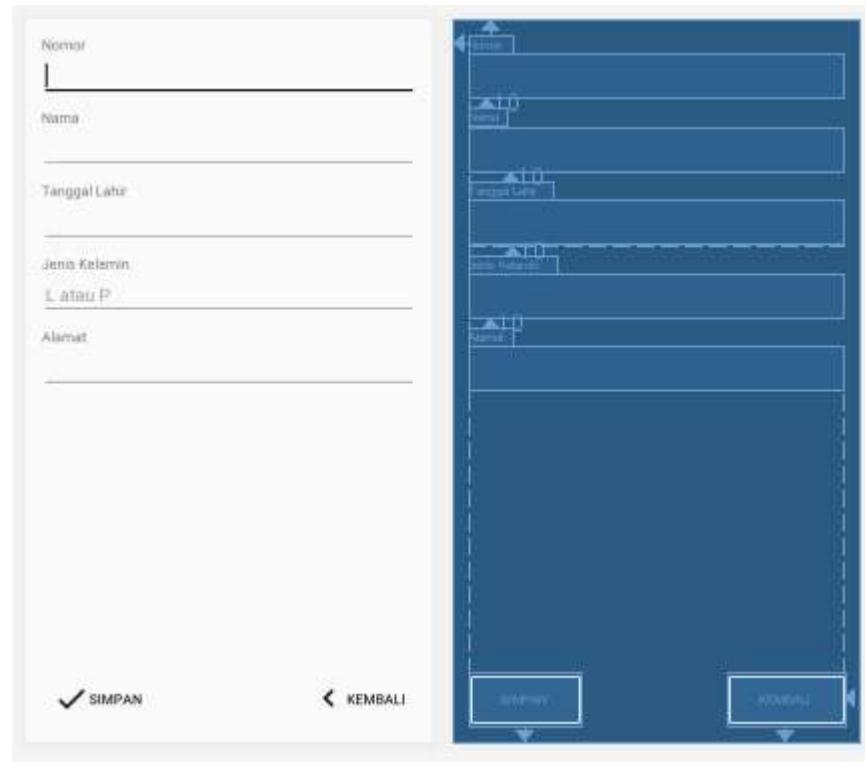
```

<Button
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText5"
    android:layout_alignParentBottom="true"
    style="?android:attr/borderlessButtonStyle"
    android:drawableLeft="@drawable/ic_done"
    android:text="Simpan" />

<Button
    android:id="@+id/button2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText5"
    android:layout_alignRight="@+id/editText5"
    android:layout_alignEnd="@+id/editText5"
    android:layout_alignBottom="true"
    android:layout_alignParentBottom="true"
    android:style="?android:attr/borderlessButtonStyle"
    android:drawableLeft="@drawable/ic_arrow"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_alignRight="@+id/editText5"
    android:layout_alignEnd="@+id/editText5" />

}</RelativeLayout>

```



Gambar 3. Layout activity_buat_biodata

▪ activity_lihat_biodata.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    tools:context=".LihatBiodata" >

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_alignParentTop="true"
        android:layout_marginRight="104dp"
        android:layout_marginTop="20dp"
        android:text="TextView" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignRight="@+id/textView1"
        android:layout_below="@+id/textView1"
        android:layout_marginTop="20dp"
        android:text="TextView" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignLeft="@+id/textView2"
        android:layout_below="@+id/textView2"
        android:layout_marginTop="20dp"
        android:text="TextView" />

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignLeft="@+id/textView3"
        android:layout_below="@+id/textView3"
        android:layout_marginTop="20dp"
        android:text="TextView" />

    <TextView
        android:id="@+id/textView5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignRight="@+id/textView4"
        android:layout_below="@+id/textView4"
        android:layout_marginTop="20dp"
        android:text="TextView" />

    <TextView
        android:id="@+id/TextView05"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBaseline="@+id/textView5"
        android:layout_alignBottom="@+id/textView5"
```

```
    android:layout_alignLeft="@+id/TextView03"
    android:text="Alamat" />

<TextView
    android:id="@+id/TextView03"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/textView4"
    android:layout_alignBottom="@+id/textView4"
    android:layout_alignLeft="@+id/TextView04"
    android:text="Jenis Kelamin" />

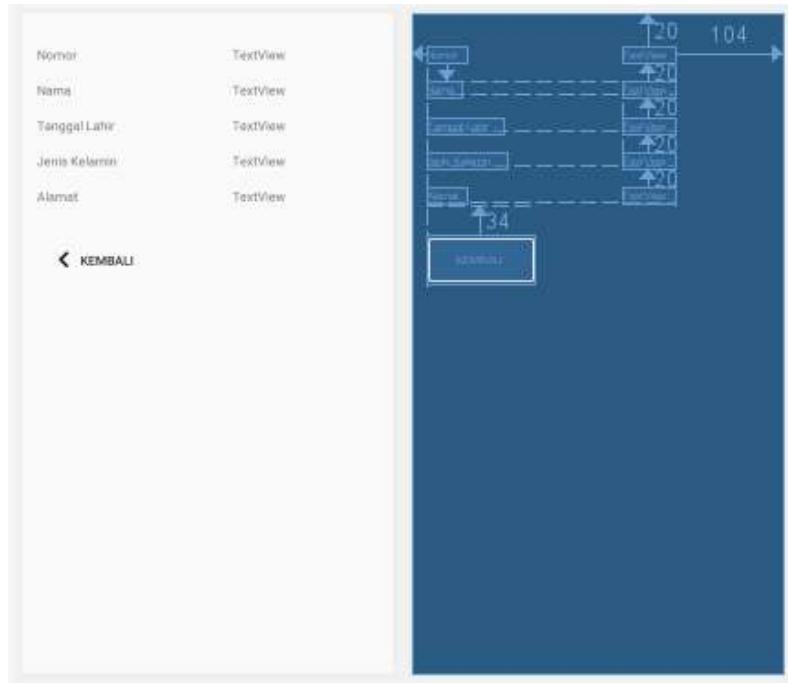
<TextView
    android:id="@+id/TextView04"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/textView3"
    android:layout_alignBottom="@+id/textView3"
    android:layout_alignLeft="@+id/TextView02"
    android:text="Tanggal Lahir" />

<TextView
    android:id="@+id/TextView02"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/textView2"
    android:layout_alignBottom="@+id/textView2"
    android:layout_alignLeft="@+id/TextView01"
    android:text="Nama" />

<TextView
    android:id="@+id/TextView01"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/textView2"
    android:layout_alignParentLeft="true"
    android:text="Nomor" />

<Button
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/TextView05"
    android:layout_below="@+id/TextView05"
    android:layout_marginTop="34dp"
    android:text="Kembali"
    style="?android:attr/borderlessButtonStyle"
    android:drawableLeft="@drawable/ic_arrow"/>

</RelativeLayout>
```



Gambar 4. Layout activity_lihat_biodata

▪ activity_update_biodata.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    tools:context=".UpdateBiodata" >

    <EditText
        android:id="@+id/editText1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignLeft="@+id/textView1"
        android:layout_below="@+id/textView1"
        android:inputType="number"
        android:maxLength="10">

        <requestFocus />
    </EditText>

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:text="Nomor" />

    <TextView
```

```
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignLeft="@+id/editText1"
        android:layout_below="@+id/editText1"
        android:layout_marginTop="10dp"
        android:text="Nama" />

<EditText
    android:id="@+id/editText2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView2"
    android:layout_below="@+id/textView2"
    android:inputType="text"
    android:maxLength="20"/>

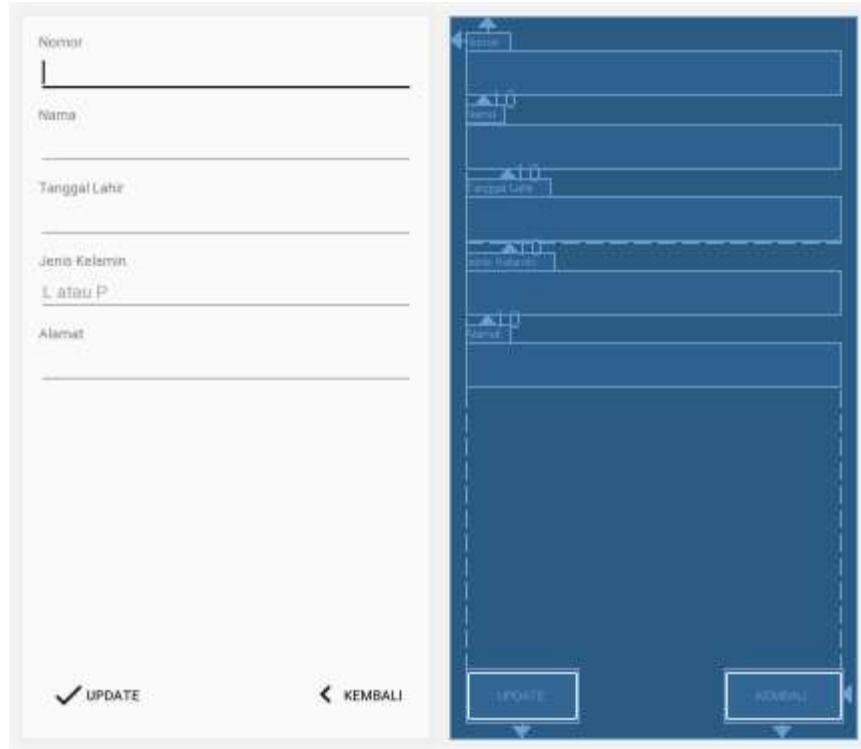
<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText2"
    android:layout_below="@+id/editText2"
    android:layout_marginTop="10dp"
    android:text="Tanggal Lahir" />

<EditText
    android:id="@+id/editText3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView3"
    android:layout_below="@+id/textView3"
    android:inputType="date"/>

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText3"
    android:layout_below="@+id/editText3"
    android:layout_marginTop="10dp"
    android:text="Jenis Kelamin" />

<EditText
    android:id="@+id/editText4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView4"
    android:layout_below="@+id/textView4"
    android:inputType="text"
    android:maxLength="1"
    android:hint="L atau P"/>
```

```
<TextView  
    android:id="@+id/textView5"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignLeft="@+id/editText4"  
    android:layout_below="@+id/editText4"  
    android:layout_marginTop="10dp"  
    android:text="Alamat"  
    android:inputType="text"  
    android:maxLength="100"/>  
  
<EditText  
    android:id="@+id/editText5"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_alignLeft="@+id/textView5"  
    android:layout_below="@+id/textView5" />  
  
<Button  
    android:id="@+id/button1"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_alignLeft="@+id/editText5"  
    android:layout_alignParentBottom="true"  
    style="?android:attr/borderlessButtonStyle"  
    android:drawableLeft="@drawable/ic_done"  
    android:text="Update" />  
  
<Button  
    android:id="@+id/button2"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Kembali"  
    style="?android:attr/borderlessButtonStyle"  
    android:drawableLeft="@drawable/ic_arrow"  
    android:layout_alignParentBottom="true"  
    android:layout_alignRight="@+id/editText5"  
    android:layout_alignEnd="@+id/editText5"  
    />  
/>
```



Gambar 5. Layout activity_update_biodata

6. Tuliskan baris kode java class di bawah dengan nama masing-masing acivity :

▪ **BuatBiodata.java**

```
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class BuatBiodata extends AppCompatActivity {
    protected Cursor cursor;
    DataHelper dbHelper;
    Button ton1, ton2;
    EditText text1, text2, text3, text4, text5;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_buat_biodata);
```

```

        dbHelper = new DataHelper( context: this );
        text1 = (EditText) findViewById(R.id.editText1);
        text2 = (EditText) findViewById(R.id.editText2);
        text3 = (EditText) findViewById(R.id.editText3);
        text4 = (EditText) findViewById(R.id.editText4);
        text5 = (EditText) findViewById(R.id.editText5);
        ton1 = (Button) findViewById(R.id.button1);
        ton2 = (Button) findViewById(R.id.button2);

        ton1.setOnClickListener((arg0) -> {
            // TODO Auto-generated method stub
            SQLiteDatabase db = dbHelper.getWritableDatabase();
            db.execSQL("insert into biodata(no, nama, tgl, jk, alamat) values('" +
                    text1.getText().toString() + "','" +
                    text2.getText().toString() + "','" +
                    text3.getText().toString() + "','" +
                    text4.getText().toString() + "','" +
                    text5.getText().toString() + "')");
            Toast.makeText(getApplicationContext(), text: "Berhasil", Toast.LENGTH_LONG).show();
            MainActivity.ma.RefreshList();
            finish();
        });
        ton2.setOnClickListener((arg0) -> {
            // TODO Auto-generated method stub
            finish();
        });
    }
}
}

```

■ LihatBiodata.java

```

import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class LihatBiodata extends AppCompatActivity {
    protected Cursor cursor;
    DataHelper dbHelper;
    Button ton2;
    TextView text1, text2, text3, text4, text5;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_lihat_biodata);
    }
}

```

```

        dbHelper = new DataHelper(context: this);
        text1 = (TextView) findViewById(R.id.textView1);
        text2 = (TextView) findViewById(R.id.textView2);
        text3 = (TextView) findViewById(R.id.textView3);
        text4 = (TextView) findViewById(R.id.textView4);
        text5 = (TextView) findViewById(R.id.textView5);
        SQLiteDatabase db = dbHelper.getReadableDatabase();
        cursor = db.rawQuery( sql: "SELECT * FROM biodata WHERE nama = '" +
            getIntent().getStringExtra( name: "nama") + "'", selectionArgs: null);
        cursor.moveToFirst();
        if (cursor.getCount()>0)
        {
            cursor.moveToPosition( i: 0);
            text1.setText(cursor.getString( i: 0).toString());
            text2.setText(cursor.getString( i: 1).toString());
            text3.setText(cursor.getString( i: 2).toString());
            text4.setText(cursor.getString( i: 3).toString());
            text5.setText(cursor.getString( i: 4).toString());
        }
        ton2 = (Button) findViewById(R.id.button1);
        ton2.setOnClickListener((arg0) -> {
            // TODO Auto-generated method stub
            finish();
        });
    }
}

```

▪ UpdateBiodata.java

```

import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class UpdateBiodata extends AppCompatActivity {
    protected Cursor cursor;
    DataHelper dbHelper;
    Button ton1, ton2;
    EditText text1, text2, text3, text4, text5;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_update_biodata);

```

```

        dbHelper = new DataHelper( context: this);
        text1 = (EditText) findViewById(R.id.editText1);
        text2 = (EditText) findViewById(R.id.editText2);
        text3 = (EditText) findViewById(R.id.editText3);
        text4 = (EditText) findViewById(R.id.editText4);
        text5 = (EditText) findViewById(R.id.editText5);
        SQLiteDatabase db = dbHelper.getReadableDatabase();
        cursor = db.rawQuery( sql: "SELECT * FROM biodata WHERE nama = '" +
        getIntent().getStringExtra( name: "nama") + "'", selectionArgs: null);
        cursor.moveToFirst();
        if (cursor.getCount()>0)
        {
            cursor.moveToPosition(0);
            text1.setText(cursor.getString( i: 0).toString());
            text2.setText(cursor.getString( i: 1).toString());
            text3.setText(cursor.getString( i: 2).toString());
            text4.setText(cursor.getString( i: 3).toString());
            text5.setText(cursor.getString( i: 4).toString());
        }
        ton1 = (Button) findViewById(R.id.button1);
        ton2 = (Button) findViewById(R.id.button2);
        // daftarkan even onClick pada btnSimpan
        ton1.setOnClickListener((arg0) → {
            // TODO Auto-generated method stub
            SQLiteDatabase db = dbHelper.getWritableDatabase();
            db.execSQL("update biodata set nama='"+
            text2.getText().toString() +"', tgl='"+ +
            text3.getText().toString()+"', jk='"+ +
            text4.getText().toString() +"', alamat='"+ +
            text5.getText().toString() + "' where no='"+ +
            text1.getText().toString()+"'");
            Toast.makeText(getApplicationContext(), text: "Berhasil", Toast.LENGTH_LONG).show();
            MainActivity.ma.RefreshList();
            finish();
        });
        ton2.setOnClickListener((arg0) → {
            // TODO Auto-generated method stub
            finish();
        });
    });

}

```

7. Tambahan : edit manifest → **AndroidManifest.xml**

```
<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="CrudSqlite"
    android:supportsRtl="true"
    android:theme="@style/AppTheme">
    <activity android:name=".MainActivity">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity android:name=".BuatBiodata"
        android:label="Form Biodata Baru">
    </activity>
    <activity android:name=".LihatBiodata"
        android:label="Lihat Biodata">
    </activity>
    <activity android:name=".UpdateBiodata"
        android:label="Update Biodata">
    </activity>
</application>
```

8. Edit res → values → **color.xml**

```
<resources>
    <color name="colorPrimary">#3F51B5</color>
    <color name="colorPrimaryDark">#303F9F</color>
    <color name="colorAccent">#000</color>
</resources>
```

9. Terakhir edit res → values → **styles.xml**

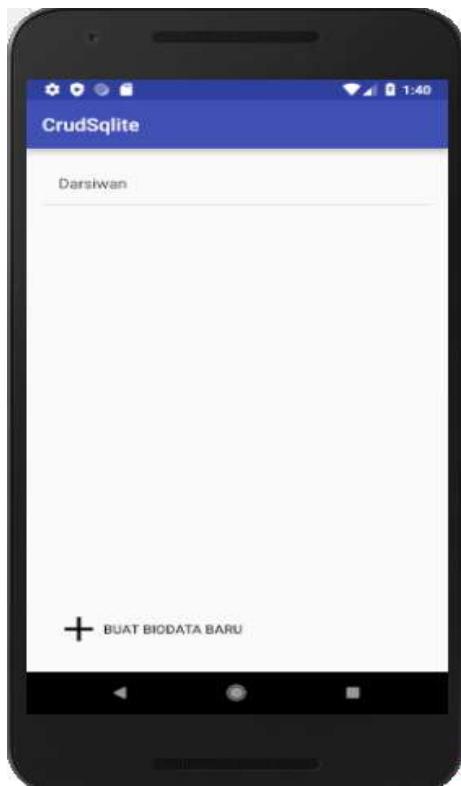
```
<resources>

    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>

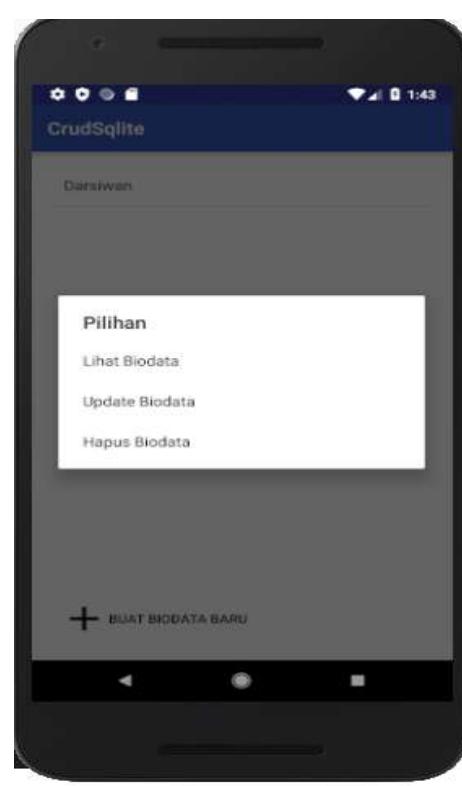
</resources>
```

10. Build dan run project tersebut, lalu perhatikan hasilnya.

Hasil run project akan menghasilkan layout seperti di halaman berikutnya:



Gambar 6. Hasil Menu Utama



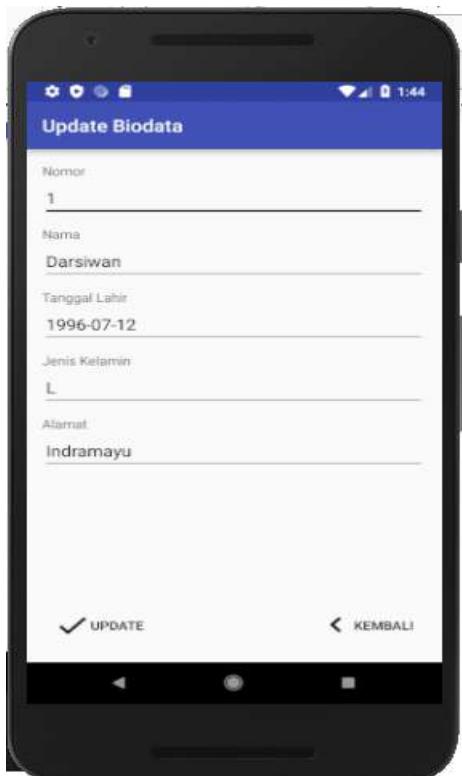
Gambar 7. Hasil untuk Pilihan



Gambar 8. Hasil Buat Biodata



Gambar 9. Hasil Lihat Biodata



Gambar 10. Hasil Buat Biodata

b. Hasil dan Pembahasan

1. Project aplikasi dari percobaan implementasi code SQLite.
2. Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project SQLite yang di buktikan dengan screenshot pada proses percobaan.

c. Kesimpulan

Mahasiswa berhasil meng-implementasi-kan SQLite pada project individu serta memahami konsep implementasi SQLite.

d. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail setiap baris code pada project SQLite	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan dengan singkat setiap baris code pada project SQLite	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan project SQLite	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan project SQLite	(1)	2	3	4
Jumlah skor					

ACARA 29 : SHARED PREFERENCE

Pokok Bahasan	: Konsep Shared Preference
Acara Praktikum/Pertemuan	: Minggu 9 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep shared preference.
2. Mahasiswa mampu membuat design sederhana untuk form register dan form login.

b. Indikator

Keberhasilan dalam memahami konsep shared preference, serta membuat design sederhana form register dan form login.

c. Dasar Teori

Shared preference

Pada Android, terdapat beberapa opsi penyimpanan (*storage*). Salah satunya adalah *Shared Preferences*. Karakter dari penyimpanan (*storage*) ini adalah *key-value storage*, sehingga hanya bisa menyimpan data dengan tipe sederhana seperti *String*, *Double* atau *float*, *Integer*, *Boolean*, dan *Long*. *Shared Preferences* menyimpan data melalui penggunaan pasangan nama / nilai. Misalnya, tentukan nama untuk data yang ingin disimpan, dan kemudian keduanya beserta nilainya akan disimpan secara otomatis ke dalam file XML. Oleh karena itu, penyimpanan ini tergolong sangat minimalis dan sangat cocok untuk penyimpanan data yang sedikit.

d. Alat dan Bahan

1. Kertas A4
2. Spidol

e. Prosedur Kerja

1. Praktikum ini akan menggunakan *Shared Preferences* untuk pembuatan aplikasi *login* dan *register* sederhana.
2. Aplikasi ini terdapat *login*, *register*, dan *logout*.

3. Memanfaatkan *Shared Preferences* meskipun halaman tersebut sudah ditutup, dapat mencegah aplikasi untuk *logout* kecuali pengguna memutuskan untuk *logout*.
4. Sehingga pengguna tidak perlu melakukan *login* berulang kali setiap membuka aplikasinya.
5. Mahasiswa membuat design sederhana untuk form register dan form login pada kertas A4.

f. Hasil dan Pembahasan

Dokumen tugas design sederhana form register dan form login, dilengkapi dengan penjelasan fungsi seriap item yang terdapat pada form tersebut.

g. Kesimpulan

Mahasiswa berhasil dalam memahami konsep shared preference, serta membuat design sederhana form register dan form login.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Pemahaman konsep dasar shared preference dengan penjelasan detail, dan design sederhana form register dan form login	1	2	3	(4)
2	Pemahaman konsep dasar manajemen file dengan penjelasan singkat dan design sederhana form register dan form login	1	2	(3)	4
3	Pemahaman konsep dasar manajemen file dengan penjelasan yang kurang, dan design sederhana form register dan form login	1	(2)	3	4
4	Tidak pemaham konsep dasar manajemen file dengan penjelasan singkat, dan design sederhana form register dan form login	(1)	2	3	4
Jumlah skor					

ACARA 30 : SHARED PREFERENCE

Pokok Bahasan	: Konsep Shared Preference
Acara Praktikum/Pertemuan	: Minggu 9 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep shared preference.
2. Mahasiswa mampu membuat code pada file preference.

b. Indikator

Keberhasilan dalam memahami konsep shared preference, serta implementasi code pada file preference.

c. Dasar Teori

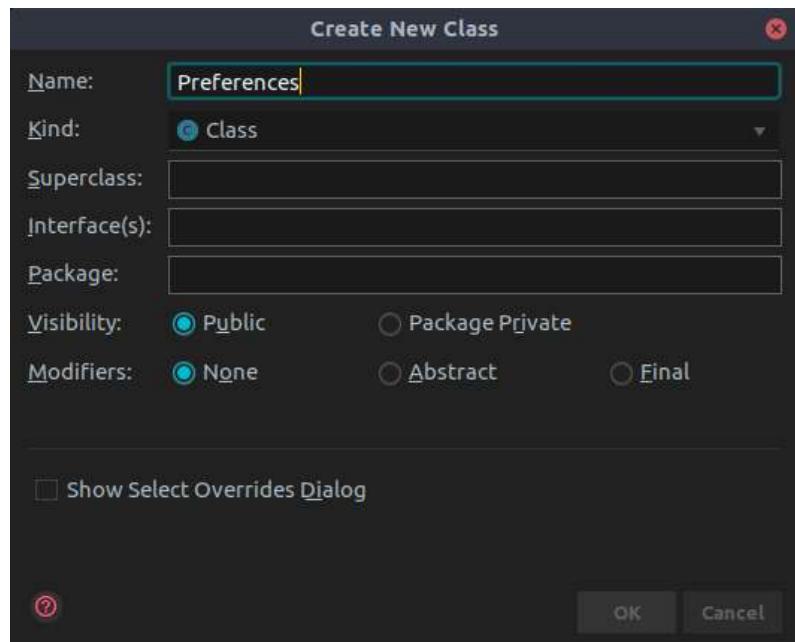
-

d. Alat dan Bahan

1. Kertas A4
2. Kabel Data USB
3. Jaringan Internet

e. Prosedur Kerja

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project **UsingPreferences**
3. Selanjutnya harus menentukan **Target Android Device** nya , pilih API 15
4. Pilih **Empty Activity**
5. Selanjutnya pada bagian **Customize The Activity** biarkan saja default
6. Dibutuhkan file dengan ekstensi **.java** baru sebagai tempat untuk *Shared Preferences* yang akan digunakan nantinya.
7. Klik **File → New → Java Class**, ketikan ‘**Preferences**’ pada form ‘Name:’ seperti pada gambar di bawah. Lalu klik tombol ‘Ok’



8. Berikut ini adalah *source code* dari class ‘Preferences’.

```
import android.content.Context;
import android.content.SharedPreferences;
import android.preference.PreferenceManager;

public class Preferences {

    /** Pendeklarasian key-data berupa String, untuk sebagai wadah penyimpanan data.
     * Jadi setiap data mempunyai key yang berbeda satu sama lain */
    static final String KEY_USER_TEREGISTER = "user", KEY_PASS_TEREGISTER = "pass";
    static final String KEY_USERNAME_SEDANG_LOGIN = "Username_logged_in";
    static final String KEY_STATUS_SEDANG_LOGIN = "Status_logged_in";

    /** Pendeklarasian Shared Preferences yang berdasarkan parameter context */
    private static SharedPreferences getSharedPreference(Context context){
        return PreferenceManager.getDefaultSharedPreferences(context);
    }

    /** Deklarasi Edit Preferences dan mengubah data
     * yang memiliki key isi KEY_USER_TEREGISTER dengan parameter username */
    public static void setRegisteredUser(Context context, String username){
        SharedPreferences.Editor editor = getSharedPreference(context).edit();
        editor.putString(KEY_USER_TEREGISTER, username);
        editor.apply();
    }
    /** Mengembalikan nilai dari key KEY_USER_TEREGISTER berupa String */
    public static String getRegisteredUser(Context context){
        return getSharedPreference(context).getString(KEY_USER_TEREGISTER, "");
    }
}
```

```

    /**
     * Deklarasi Edit Preferences dan mengubah data
     * yang memiliki key KEY_PASS_TERREGISTER dengan parameter password */
    public static void setRegisteredPass(Context context, String password){
        SharedPreferences.Editor editor = getSharedPreference(context).edit();
        editor.putString(KEY_PASS_TERREGISTER, password);
        editor.apply();
    }

    /**
     * Mengembalikan nilai dari key KEY_PASS_TERREGISTER berupa String */
    public static String getRegisteredPass(Context context){
        return getSharedPreference(context).getString(KEY_PASS_TERREGISTER,"");
    }

    /**
     * Deklarasi Edit Preferences dan mengubah data
     * yang memiliki key KEY_USERNAME_SEDANG_LOGIN dengan parameter username */
    public static void setLoggedInUser(Context context, String username){
        SharedPreferences.Editor editor = getSharedPreference(context).edit();
        editor.putString(KEY_USERNAME_SEDANG_LOGIN, username);
        editor.apply();
    }

    /**
     * Mengembalikan nilai dari key KEY_USERNAME_SEDANG_LOGIN berupa String */
    public static String getLoggedInUser(Context context){
        return getSharedPreference(context).getString(KEY_USERNAME_SEDANG_LOGIN,"");
    }

    /**
     * Deklarasi Edit Preferences dan mengubah data
     * yang memiliki key KEY_STATUS_SEDANG_LOGIN dengan parameter status */
    public static void setLoggedInStatus(Context context, boolean status){
        SharedPreferences.Editor editor = getSharedPreference(context).edit();
        editor.putBoolean(KEY_STATUS_SEDANG_LOGIN,status);
        editor.apply();
    }

    /**
     * Mengembalikan nilai dari key KEY_STATUS_SEDANG_LOGIN berupa boolean */
    public static boolean getLoggedInStatus(Context context){
        return getSharedPreference(context).getBoolean(KEY_STATUS_SEDANG_LOGIN,false);
    }

    /**
     * Deklarasi Edit Preferences dan menghapus data, sehingga menjadikannya bernilai default
     * khusus data yang memiliki key KEY_USERNAME_SEDANG_LOGIN dan KEY_STATUS_SEDANG_LOGIN */
    public static void clearLoggedInUser (Context context){
        SharedPreferences.Editor editor = getSharedPreference(context).edit();
        editor.remove(KEY_USERNAME_SEDANG_LOGIN);
        editor.remove(KEY_STATUS_SEDANG_LOGIN);
        editor.apply();
    }
}

```

9. Dalam *class* ‘**Preferences**’, ditambahkan *mutator* ‘**SharedPreferences**’ yang berguna untuk pendeklarasian ‘**SharedPreferences**’ di *activity* yang berbeda-beda dan setiap kali menggunakan ‘**SharedPreferences**’ harus melalui *mutator* ini.

```
public static SharedPreferences getSharedPreference(Context
        context) {
    return PreferenceManager.getDefaultSharedPreferences(context);
}
```

10. Lalu deklarasi-kan *key* yang akan kita gunakan nanti saat kita ingin menambah atau mengedit data.

```
public class Preferences {
    static final String KEY_USER_TEREGISTER =
        "user",KEY_PASS_TEREGISTER ="pass";
    static final String KEY_USERNAME_SEDANG_LOGIN =
        "Username_logged_in";
    static final String KEY_STATUS_SEDANG_LOGIN =
        "Status_logged_in";
}
```

11. Tambahkan *getter* *setter* untuk semua *key* yang sudah dideklarasikan, dan buat semua *getter* dan *setter* memiliki *public static void* serta berparameter ‘Context’. Khusus *setter*, tambahkan parameter ‘String’ atau ‘boolean’ untuk *setter* status.

f. Hasil dan Pembahasan

Project aplikasi dari percobaan implementasi code pada file preferences.

g. Kesimpulan

Mahasiswa berhasil dalam memahami konsep shared preference, serta implementasi code pada file preference.

h. Rubrik Penilaian

No	Indikator	Skor*				
1	Pemahaman konsep dasar implementasi code pada file preference dengan penjelasan detail	1	2	3	(4)	
2	Pemahaman konsep dasar implementasi code pada file preference dengan penjelasan singkat	1	2	(3)	4	
3	Pemahaman konsep dasar implementasi code pada file preference dengan penjelasan yang kurang	1	(2)	3	4	
4	Tidak pemaham konsep dasar implementasi code pada file preference dengan penjelasan singkat	(1)	2	3	4	
Jumlah skor						

ACARA 31 : SHARED PREFERENCE

Pokok Bahasan	: Shared Preference pada Register
Acara Praktikum/Pertemuan	: Minggu 9 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami shared preference pada register dan login.
2. Mahasiswa mampu implementasi code shared preference untuk register dan login pada android.

b. Indikator

Keberhasilan dalam memahami shared preference pada register dan login, serta implementasi code shared preference untuk register dan login pada android.

c. Dasar Teori

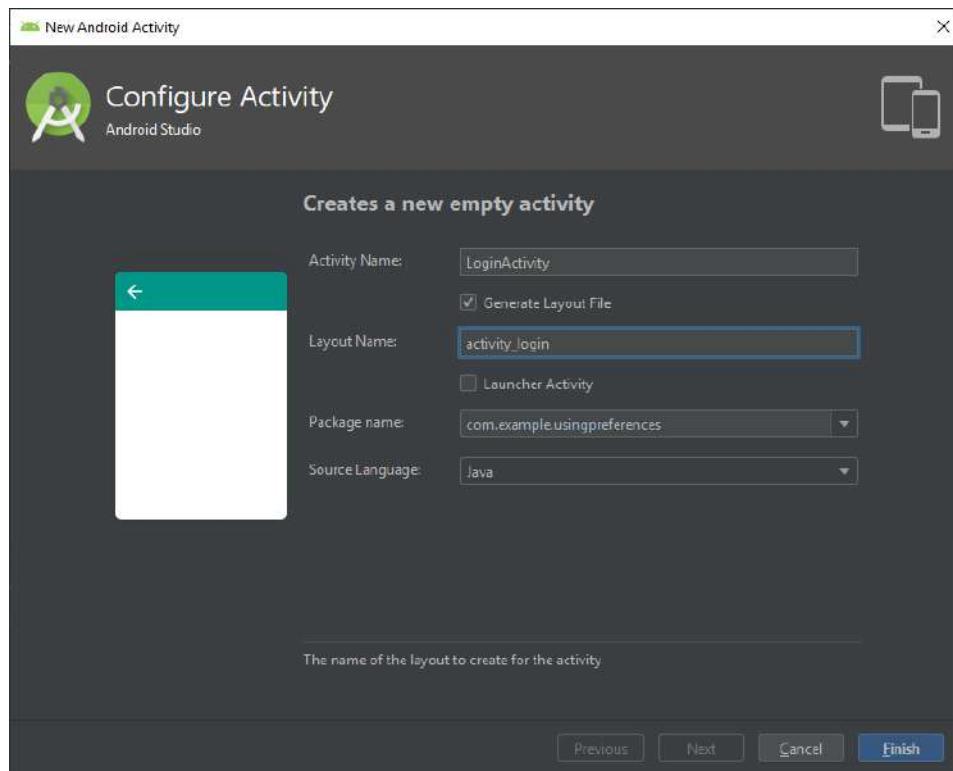
-

d. Alat dan Bahan

1. Kertas A4
2. Kabel Data USB
3. Jaringan Internet

e. Prosedur Kerja

1. Buat form Login dan Register pada projec preference
 - klik **File** → **New** → **Activity** → **Empty Activity** lalu akan muncul jendela baru seperti gambar di bawah ini.
 - Gantilah nama activity pada form ‘Activity Name:’ dengan ‘**LoginActivity**’ dan ‘**RegisterActivity**’ [tanpa spasi]. Lalu klik tombol ‘Finish’.



2. Tambahkan baris source code di bawah ini ke dalam class **MainActivity**.

```
3 import androidx.appcompat.app.AppCompatActivity;
4 import android.content.Intent;
5 import android.os.Bundle;
6 import android.view.View;
7 import android.widget.TextView;
8 public class MainActivity extends AppCompatActivity {
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        setContentView(R.layout.activity_main);
13
14        /* Deklarasi dan Menginisialisasi variable nama dengan Label Nama dari Layout MainActivity */
15        TextView nama = findViewById(R.id.tv_namaMain);
16
17        /* Men-set Label Nama dengan data User sedang login dari Preferences */
18        nama.setText(Preferences.getLoggedInUser(getApplicationContext()));
19
20        /* Men-set Status dan User yang sedang login menjadi default atau kosong di
21         * Data Preferences. Kemudian menuju ke LoginActivity*/
22        findViewById(R.id.button_logoutMain).setOnClickListener(v) ->{
23            //Menghapus Status login dan kembali ke Login Activity
24            Preferences.clearLoggedInUser(getApplicationContext());
25            startActivity(new Intent(getApplicationContext(), LoginActivity.class));
26            finish();
27        };
28    }
29 }
30 }
```

3. Tambahkan baris source code di bawah ini ke dalam class **LoginActivity**.

```
3   import android.content.Intent;
4   import android.os.Bundle;
5   import android.text.TextUtils;
6   import android.view.KeyEvent;
7   import android.view.View;
8   import android.view.inputmethod.EditorInfo;
9   import android.widget.EditText;
10  import android.widget.TextView;
11
12  import androidx.appcompat.app.AppCompatActivity;
13
14  public class LoginActivity extends AppCompatActivity {
15
16      private EditText mViewUser, mViewPassword;
17
18      @Override
19      protected void onCreate(Bundle savedInstanceState) {
20          super.onCreate(savedInstanceState);
21          setContentView(R.layout.activity_login);
22
23          /* Menginisialisasi variable dengan Form User dan Form Password dari Layout LoginActivity */
24          mViewUser=findViewById(R.id.et_emailSignin);
25          mViewPassword =findViewById(R.id.et_passwordSignin);
26          /* Menjalankan Method razia() Jika tombol SignIn di keyboard di sentuh */
27          mViewPassword.setOnEditorActionListener((v, actionId, event) -> {
28              if (actionId == EditorInfo.IME_ACTION_DONE || actionId == EditorInfo.IME_NULL) {
29                  razia();
30                  return true;
31              }
32              return false;
33          });
34
35
36          /* Menjalankan Method razia() jika merasakan tombol SignIn disentuh */
37          findViewById(R.id.button_signinSignin).setOnClickListener((v) -> {
38              razia();
39          });
39
40          /* Ke RegisterActivity jika merasakan tombol Signup disentuh */
41          findViewById(R.id.button_signupSignin).setOnClickListener((v) -> {
42              startActivity(new Intent(getApplicationContext(),RegisterActivity.class));
43          });
44
45
46      }
47
48      /** ke MainActivity jika data Status Login dari Data Preferences bernilai true */
49      @Override
50      protected void onStart() {
51          super.onStart();
52          if (Preferences.getLoggedInStatus(getApplicationContext())){
53              startActivity(new Intent(getApplicationContext(),MainActivity.class));
54              finish();
55          }
56      }
57
58  }
```

```

61     }
62 }
63
64 /**
65  * Men-check inputan User dan Password dan Memberikan akses ke MainActivity */
66 private void razia(){
67     /* Mereset semua Error dan fokus menjadi default */
68     mViewUser.setError(null);
69     mViewPassword.setError(null);
70     View fokus = null;
71     boolean cancel = false;
72
73     /* Mengambil text dari form User dan form Password dengan variable baru bertipe String*/
74     String user = mViewUser.getText().toString();
75     String password = mViewPassword.getText().toString();
76
77     /* Jika form user kosong atau TIDAK memenuhi kriteria di Method cekUser() maka, Set error
78      * di Form User dengan menset variable fokus dan error di Viewnya juga cancel menjadi true*/
79     if (TextUtils.isEmpty(user)){
80         mViewUser.setError("This field is required");
81         fokus = mViewUser;
82         cancel = true;
83     }else if(!cekUser(user)){
84         mViewUser.setError("This Username is not found");
85         fokus = mViewUser;
86         cancel = true;
87     }
88
89     /* Sama syarat percabangannya dengan User seperti di atas. Bedanya ini untuk Form Password*/
90     if (TextUtils.isEmpty(password)){
91         mViewPassword.setError("This field is required");
92         fokus = mViewPassword;
93         cancel = true;
94     }else if (!cekPassword(password)){
95         mViewPassword.setError("This password is incorrect");
96         fokus = mViewPassword;
97         cancel = true;
98     }
99
100    /* Jika cancel true, variable fokus mendapatkan fokus */
101    if (cancel) fokus.requestFocus();
102    else masuk();
103}
104
105 /**
106  * Menuju ke MainActivity dan Set User dan Status sedang login, di Preferences */
107 private void masuk(){
108     Preferences.setLoggedInUser(getApplicationContext(),
109         Preferences.getRegisteredUser(getApplicationContext()));
110     Preferences.setLoggedInStatus(getApplicationContext(), status: true);
111     startActivity(new Intent(getApplicationContext(),MainActivity.class));finish();
112 }

```

```

112 	/** True jika parameter password sama dengan data password yang terdaftar dari Preferences */
113 	private boolean cekPassword(String password){
114 		return password.equals(Preferences.getRegisteredPass(getApplicationContext()));
115 	}
116
117 	/** True jika parameter user sama dengan data user yang terdaftar dari Preferences */
118 	private boolean cekUser(String user){
119 		return user.equals(Preferences.getRegisteredUser(getApplicationContext()));
120 	}
121 }

```

4. Code XML terdiri dari **activity_login**, **activity_register**, dan **activity_main**. Pertama tambahkan baris source code di bawah ini ke dalam **activity_login**.

```

1 	<?xml version="1.0" encoding="utf-8"?>
2 	<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3 		xmlns:tools="http://schemas.android.com/tools"
4 		android:layout_width="match_parent"
5 		android:layout_height="match_parent"
6 		android:gravity="center"
7 		android:padding="18dp"
8 		android:orientation="vertical"
9 		tools:context=".LoginActivity">
10 	<EditText
11 		android:id="@+id/et_emailSignin"
12 		android:layout_width="match_parent"
13 		android:layout_height="wrap_content"
14 		android:hint="Username"
15 		android:inputType="textEmailAddress"
16 		android:maxLines="1"
17 		android:singleLine="true"
18 		android:imeOptions="actionNext"/>
19 	<EditText
20 		android:id="@+id/et_passwordSignin"
21 		android:layout_width="match_parent"
22 		android:layout_height="wrap_content"
23 		android:layout_marginBottom="30dp"
24 		android:hint="Password"
25 		android:inputType="textPassword"
26 		android:maxLines="1"
27 		android:singleLine="true"
28
29 		android:imeActionId="6"
30 		android:imeActionLabel="Sign In"
31 		android:imeOptions="actionDone" />
32 	<LinearLayout
33 		android:layout_width="match_parent"
34 		android:layout_height="wrap_content">

```

```
34    <Button
35        android:id="@+id/button_signupSignin"
36        android:layout_width="0dp"
37        android:layout_height="wrap_content"
38        android:text="Sign Up"
39        android:layout_weight="1"/>
40    <Button
41        android:id="@+id/button_signinSignin"
42        android:layout_width="0dp"
43        android:layout_height="wrap_content"
44        android:text="Sign In"
45        android:layout_weight="1"/>
46    </LinearLayout>
47 </LinearLayout>
```

5. Tambahkan baris source code di bawah ini ke dalam **activity_register**

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      android:gravity="center"
7      android:padding="18dp"
8      android:orientation="vertical"
9      tools:context=".RegisterActivity">
10     <EditText
11         android:id="@+id/et_emailSignup"
12         android:layout_width="match_parent"
13         android:layout_height="wrap_content"
14         android:hint="Username"
15         android:inputType="textEmailAddress"
16         android:maxLines="1"
17         android:singleLine="true"
18         android:imeOptions="actionNext"/>
19     <EditText
20         android:id="@+id/et_passwordSignup"
21         android:layout_width="match_parent"
22         android:layout_height="wrap_content"
23         android:hint="Password"
24         android:inputType="textPassword"
25         android:maxLines="1"
26         android:singleLine="true"
27         android:imeOptions="actionNext"/>
```

```
26     <EditText
27         android:id="@+id/et_passwordSignup2"
28         android:layout_width="match_parent"
29         android:layout_height="wrap_content"
30         android:layout_marginBottom="30dp"
31         android:hint="Confirm Password"
32         android:inputType="textPassword"
33         android:maxLines="1"
34         android:singleLine="true"
35         android:imeActionLabel="Sign Up"
36         android:imeOptions="actionDone" />
37
38     <Button
39         android:id="@+id/button_signupSignup"
40         android:layout_width="match_parent"
41         android:layout_height="wrap_content"
42         android:text="Sign Up" />
43
44 </LinearLayout>
```

6. Tambahkan baris source code di bawah ini ke dalam activity_main

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      android:padding="30dp"
7      tools:context=".MainActivity">
8
9     <LinearLayout
10        android:layout_width="match_parent"
11        android:layout_height="wrap_content"
12        android:orientation="vertical">
13
14         <TextView
15             android:layout_width="wrap_content"
16             android:layout_height="wrap_content"
17             android:textSize="20sp"
18             android:text="Masuk sebagai" />
19
20         <TextView
21             android:id="@+id/tv_namaMain"
22             android:layout_width="wrap_content"
23             android:layout_height="wrap_content"
24             android:textSize="30sp"
25             tools:text="Namaku adalah" />
26
27     </LinearLayout>
28
29     <Button
30         android:id="@+id/button_logoutMain"
31         android:layout_width="match_parent"
32         android:layout_height="wrap_content"
33         android:layout_alignParentBottom="true"
34         android:layout_marginTop="10dp"
35         android:text="Log Out" />
36
37 </RelativeLayout>
```

7. Perhatikan *code* berikut pada tombol ‘Login’.

```
SharedPreferences.Editor editor =
getSharedPreference(context).edit();
editor.putBoolean(KEY_STATUS_SEDANG_LOGIN,status); editor.apply();
```

Akan tetapi apabila sudah membuat class Preferences, hanya perlu menambahkan code berikut.

```
Preferences.setLoggedInStatus(getApplicationContext(),true);
```

8. Terakhir, jangan lupa untuk mengubah code pada file ‘**AndroidManifest.xml**’ untuk mendeklarasikan activity baru yang telah dibuat sebelumnya, dengan ‘**LoginActivity**’ sebagai activity utamanya. Berikut code dari **AndroidManifest.xml**.

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      package="com.example.usingpreferences">
4
5      <application
6          android:allowBackup="true"
7          android:icon="@mipmap/ic_launcher"
8          android:label="UsingPreferences"
9          android:roundIcon="@mipmap/ic_launcher_round"
10         android:supportsRtl="true"
11         android:theme="@style/AppTheme">
12             <activity android:name="com.example.usingpreferences.LoginActivity">
13                 <intent-filter>
14                     <action android:name="android.intent.action.MAIN" />
15
16                     <category android:name="android.intent.category.LAUNCHER" />
17                 </intent-filter>
18             </activity>
19             <activity android:name="com.example.usingpreferences.MainActivity"/>
20             <activity android:name="com.example.usingpreferences.RegisterActivity"/>
21         </application>
22
23     </manifest>
```

9. Build dan run project tersebut, lalu perhatikan hasilnya.

f. Hasil dan Pembahasan

Project aplikasi dari percobaan implementasi code shared preference pada register dan login.

g. Kesimpulan

Mahasiswa berhasil dalam memahami konsep shared preference pada register dan login, serta implementasi code shared preference pada register dan login.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Pemahaman konsep dasar implementasi code shared preference pada register dan login dengan penjelasan detail	1	2	3	(4)
2	Pemahaman konsep dasar implementasi code shared preference pada register dan login dengan penjelasan singkat	1	2	(3)	4
3	Pemahaman konsep dasar implementasi code shared preference pada register dan login dengan penjelasan yang kurang	1	(2)	3	4
4	Tidak pemaham konsep dasar implementasi code shared preference pada register dan login dengan penjelasan singkat	(1)	2	3	4
Jumlah skor					

ACARA 32 : SHARED PREFERENCE

Pokok Bahasan	: Shared Preference pada Login
Acara Praktikum/Pertemuan	: Minggu 9 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami shared preference pada register dan login.
2. Mahasiswa mampu implementasi code shared preference untuk register dan login pada android.

b. Indikator

Keberhasilan dalam memahami shared preference pada register dan login, serta implementasi code shared preference untuk register dan login pada android.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kertas A4
2. Kabel Data USB
3. Jaringan Internet

e. Prosedur Kerja

1. Mahasiswa melakukan analisa code untuk setiap baris code dari implementasi shared preferences register dan login.
2. Menjelaskan fungsi dan arti dari setiap baris code shared preferences register dan login.
3. Analisa hasil running dari implementasi project shared preferences register dan login.
4. Screenshot output setiap proses dari project shared preferences register dan login.
5. Tulisakan dalam bentuk laporan.

f. Hasil dan Pembahasan

Dokumentasi tugas berupa laporan secara individu penjelasan setiap code pada project shared preferences register dan login yang di buktikan dengan screenshot pada proses percobaan.

g. Kesimpulan

Mahasiswa memahami konsep dasar shared preferences register dan login dengan mampu memahami setiap baris code dari implementasi shared preferences register dan login.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menjelaskan secara detail setiap baris code pada implemenasi shared preferences register dan login	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menjelaskan setiap baris code pada implemenasi shared preferences register dan login	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menjelaskan setiap baris code pada implemenasi shared preferences register dan login	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menjelaskan setiap baris code pada implemenasi shared preferences register dan login	(1)	2	3	4
Jumlah skor					

ACARA 33 : GPS DAN AKSES SENSOR

Pokok Bahasan	: Menampilkan Maps
Acara Praktikum/Pertemuan	: Minggu 10 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep location based service.
2. Mahasiswa mampu menerapkan location based service pada android.
3. Mahasiswa mampu menampilkan maps.
4. Mahasiswa mampu mendapatkan maps API key, menampilkan zoom control, menampilkan navigasi ke lokasi khusus.

b. Indikator

1. Keberhasilan mahasiswa menampilkan maps.
2. Keberhasilan mahasiswa mendapatkan maps API key.
3. Keberhasilan mahasiswa menampilkan zoom control.
4. Keberhasilan mahasiswa menampilkan navigasi ke lokasi khusus.

c. Dasar Teori

Location Based Service

Location based Service (LBS), yang dalam Bahasa Indonesianya Layanan Berbasis Lokasi adalah suatu sistem yang menyediakan layanan yang mempunyai lokasi geografis sehingga dapat dilihat pada peta. Salah satu sumber data geografis yang mudah untuk didapatkan adalah berupa koordinat lokasi yang diambil dari *smartphone*.

Smartphone Android mempunyai dua macam jenis lokasi, *coarse-location* dan *fine-location*. Lokasi *coarse* / kasar adalah lokasi user yang ditentukan lewat BTS terdekat, mempunyai presisi yang tidak akurat, namun hemat baterai. Sedangkan *fine-location* / lokasi halus, adalah lokasi user yang didapatkan melalui GPS yang sekarang sudah umum ada di perangkat Android. *Fine-location* mempunyai presisi yang lebih tinggi, namun boros baterai.

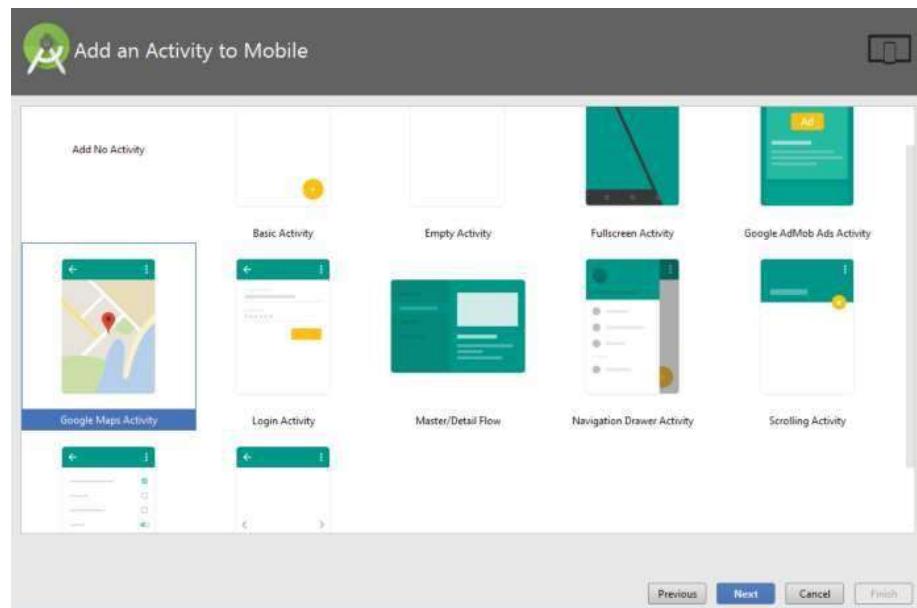
d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

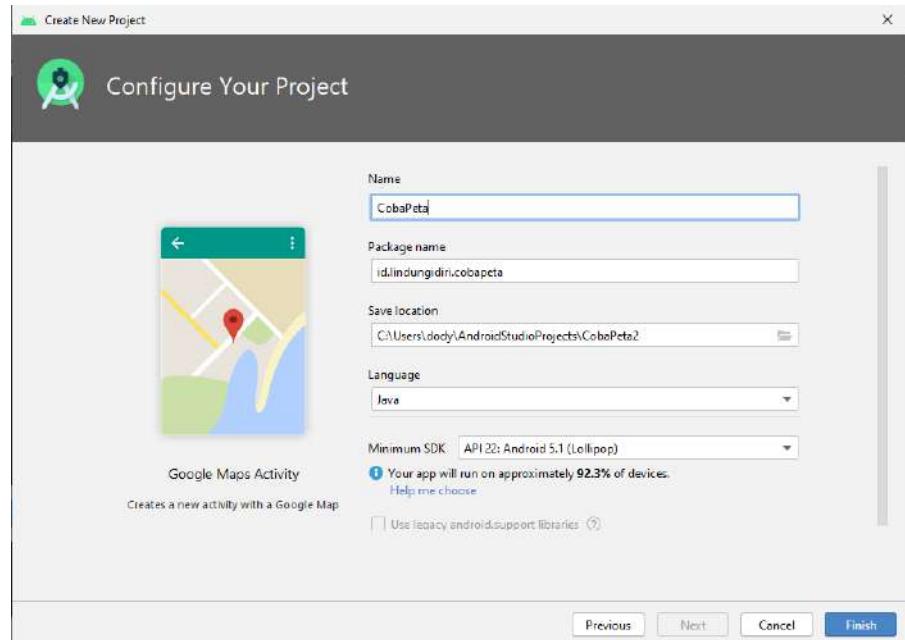
e. Prosedur Kerja

Location Based Service (LBS) : implementasi LBS pada smartphone.

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama untuk project LBS
3. Dari Create New Project Wizard, pilih Google Maps Activity seperti yang ditunjukkan pada Gambar 1, kemudian pilih Next
4. Halaman berikutnya tampil Gambar 2 yang menunjukkan nama proyek, nama paket, bahasa Java yang digunakan, dan minimum SDK yang di dukung



Gambar 1. Membuat activity untuk menampilkan maps



Gambar 2. Konfigurasi template maps activity Android Studio

Menampilkan Maps

Google Maps adalah salah satu dari banyak aplikasi yang dibundel dengan platform android. Selain hanya menggunakan aplikasi Peta, juga dapat menanamkannya ke aplikasi dan membuatnya melakukan beberapa hal yang sangat keren. Bagian ini menjelaskan cara menggunakan Google Maps di aplikasi android.

A. Mendapatkan Maps API Key

Setelah project default dari “**Google Maps Activity**” dibuat, lakukan pengecekan file “**google_maps_api.xml**” di folder **Android→App → res → values**. Dalam file ini ada 3 informasi yang harus diperhatikan yaitu :

1. Package name, yang merupakan nama package dari aplikasi yang dibuat
2. SHA-1 certificate fingerprint, yang merupakan kode fingerprint SHA-1
3. string name="google_maps_key", yang berisi Google Maps API Key

Secara default “**google_maps_key**” masih kosong (bertuliskan YOUR_KEY_HERE), silakan anda membuat Maps API Key melalui langkah-langkah sebagai berikut :

1. Buka URL <https://console.cloud.google.com/>
2. Login menggunakan akun Google pribadi
3. Buat Proyek dengan nama sesuai keinginan

4. Pilih proyek yang baru dibuat
5. Buka Navigation **Menu → APIs & Services → Library**
6. Enable Library berikut: a) Maps SDK for Android; b) Directions API; c) Geocoding API; d) Geolocation API; e) Maps Places API; f) Maps JavaScript API
7. Buka Navigation **Menu → APIs & Services → Credentials**
8. Klik Create Credentials untuk membuat API Key
9. Untuk meningkatkan security dapat ditambahkan restriction dengan memasukkan aplikasi pengembang (contoh Android App), nama paket, dan SHA-1 dari file “google_maps_api.xml”
10. Copy API Key dan isikan pada file “google_maps_api.xml”
11. Jalankan proyek android dan jika berhasil maka dihasilkan tampilan Gambar 3.
12. Jika terdapat kesalahan silakan cek Logcat untuk melihat warning atau error.



Gambar 3. Menampilkan peta pada aplikasi

13. Menampilkan Google Maps di aplikasi, pertama-tama memerlukan izin **ACCESS_FINE_LOCATION** dalam file manifes. Ini dibuat untuk secara otomatis ketika memilih untuk mengatur Google Maps Activity. Jika membuka **AndroidManifest.xml**, maka dapat melihat garis.

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
```

14. Untuk menguji aplikasi pada emulator android, pastikan untuk membuat emulator dengan versi SDK yang menyertakan Google Play Services sebagai target yang dipilih.

B. Menampilkan Zoom Control

Subbab A menunjukkan bagaimana dapat menampilkan Google Maps di aplikasi android, dan dapat menggeser peta ke lokasi yang diinginkan dan memperbarui saat itu juga. Namun, tidak ada cara untuk menggunakan emulator untuk memperbesar atau memperkecil dari lokasi tertentu (pada perangkat android nyata dapat mencubit peta untuk memperbesarnya). Pada subbab B ini dapat memungkinkan pengguna memperbesar atau memperkecil peta menggunakan kontrol zoom bawaan.

1. Buka kembali proyek yang dibuat di bagian sebelumnya
2. Tambahkan pernyataan tebal berikut ke **activity_maps.xml**
3. Pastikan untuk mengganti semua instance com.jfdimarzio dengan referensi ke paket yang digunakan dalam aplikasi Anda:

```
<?xml version="1.0" encoding="utf-8"?>
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:map="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/map"
    android:name="com.google.android.gms.maps.SupportMapFragment"
    map:uiZoomControls="true"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MapsActivity" />
```

4. Tekan Shift + F9 untuk men-debug aplikasi pada emulator android.
5. Amati kontrol zoom bawaan yang muncul di bagian bawah peta ketika mengklik dan menarik peta (lihat Gambar 4). Klik ikon minus (-) untuk memperkecil peta, dan ikon plus (+) untuk memperbesar peta.



Gambar 4. Menampilkan zoom control

Untuk menampilkan kontrol zoom bawaan, harus menambahkan parameter ke activity_maps.xml yang membuat uiZoomControls menjadi true:

```
map:uiZoomControls="true"
```

C. Navigasi ke Lokasi Khusus

Secara default, Google Maps menampilkan peta Australia saat pertama kali dimuat. Namun, dapat mengatur Google Maps untuk menampilkan lokasi tertentu. Untuk melakukannya, dapat menggunakan metode moveCamera () dari kelas GoogleMap. Try It Out berikut ini menunjukkan bagaimana secara programatik menghidupkan Google Maps ke lokasi tertentu.

1. Menggunakan proyek yang dibuat di bagian sebelumnya, ubah pernyataan tebal berikut ke file MapsActivity.java:

```
@Override  
public void onMapReady(GoogleMap googleMap) {  
    mMap = googleMap;  
  
    // Add a marker in Sydney and move the camera  
    LatLng boston = new LatLng(42.3601, -71.0589);  
    mMap.addMarker(new MarkerOptions().position(boston).title("Boston, Mass"));  
    mMap.moveCamera(CameraUpdateFactory.newLatLng(boston));  
}
```

2. Tekan Shift + F9 untuk men-debug aplikasi pada emulator android. Ketika peta dimuat, amati bahwa sekarang menjiwai ke lokasi tertentu di Boston, Massachusetts (lihat Gambar 5).

Dalam kode sebelumnya, objek LatLng dibuat dan diatur ke koordinat baru 42,3601 N, 71,0589 W (koordinat Boston). Perhatikan bahwa untuk mewakili W (barat) atau S (selatan) menggunakan nilai negatif koordinat.



Gambar 5. Menampilkan lokasi Boston

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan implementasi code untuk menampilkan maps dengan mendapatkan maps API key, menampilkan zoom control, menampilkan navigasi ke lokasi khusus.
2. Dokumentasi hasil output aplikasi dalam bentuk laporan dilengkapi dengan analisa code pada implementasi source code.

g. Kesimpulan

Mahasiswa menampilkan maps dengan mendapatkan maps api key, menampilkan zoom control, menampilkan navigasi ke lokasi khusus

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara detail setiap baris code untuk menampilkan maps	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara singkat setiap baris code untuk menampilkan maps	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam implementasi code serta menjelaskan setiap baris code untuk menampilkan maps	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam implementasi code serta menjelaskan setiap baris code untuk menampilkan maps	(1)	2	3	4
Jumlah skor					

ACARA 34 : GPS DAN AKSES SENSOR

Pokok Bahasan	: Mendapatkan Dan Monitoring Lokasi
Acara Praktikum/Pertemuan	: Minggu 10 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep location based service.
2. Mahasiswa mampu menerapkan location based service pada android.
3. Mahasiswa mampu mendapatkan lokasi dengan menentukan lokasi fisik pengguna.
4. Mahasiswa mampu memonitoring lokasi untuk memantau lokasi tertentu.

b. Indikator

1. Keberhasilan mahasiswa mendapatkan lokasi dengan menentukan lokasi fisik pengguna.
2. Keberhasilan mahasiswa mampu memonitoring lokasi untuk memantau lokasi tertentu.

c. Dasar Teori

Saat ini, perangkat seluler umumnya dilengkapi dengan penerima GPS. Karena banyaknya satelit yang mengorbit bumi, kita dapat menggunakan penerima GPS untuk menemukan lokasi dengan mudah. Namun, GPS membutuhkan langit yang cerah untuk bekerja dan karenanya tidak selalu berfungsi di dalam ruangan atau tempat satelit tidak dapat menembus (seperti terowongan melalui gunung).

Cara efektif lain untuk menemukan posisi adalah melalui triangulasi menara seluler. Ketika ponsel dihidupkan, ponsel itu terus-menerus bersentuhan dengan BTS di sekitarnya. Dengan mengetahui identitas menara sel, dimungkinkan untuk menerjemahkan informasi ini menjadi lokasi fisik melalui penggunaan berbagai basis data yang mengandung identitas menara sel dan lokasi geografisnya yang sebenarnya. Keuntungan dari triangulasi menara sel adalah ia bekerja di dalam ruangan, tanpa perlu mendapatkan informasi dari satelit. Namun, itu tidak setepat GPS karena akurasinya tergantung pada cakupan sinyal yang tumpang tindih, yang sedikit bervariasi. Triangulasi menara seluler bekerja paling baik di daerah padat penduduk di mana menara seluler terletak dekat.

Metode ketiga untuk menemukan posisi adalah mengandalkan triangulasi Wi-Fi. Alih alih menyambungkan ke menara seluler, perangkat terhubung ke jaringan Wi-Fi dan memeriksa penyedia layanan terhadap basis data untuk menentukan lokasi yang dilayani oleh penyedia. Dari tiga metode yang dijelaskan di sini, triangulasi Wi-Fi adalah yang paling tidak akurat.

d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

e. Prosedur Kerja

Mendapatkan lokasi

Pada platform Android, SDK menyediakan kelas LocationManager untuk membantu perangkat untuk menentukan lokasi fisik pengguna. Berikut ini menunjukkan bagaimana ini dilakukan dalam kode.

1. Menggunakan proyek yang sama yang dibuat di bagian sebelumnya, tambahkan pernyataan tebal berikut ke file MapsActivity.java:

```
package com.example.locbasserv;

import androidx.core.app.ActivityCompat;
import androidx.fragment.app.FragmentActivity;

import android.Manifest;
import android.content.Context;
import android.content.pm.PackageManager;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.widget.Toast;

import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;

public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {

    final private int REQUEST_PERMISSION_ACCESS = 123;
    boolean permissionGranted = false;
    private GoogleMap mMap;
    LocationManager lm;
    LocationListener locationListener;
```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_maps);
    // Obtain the SupportMapFragment and get notified when the map is ready
    // to be used.
    SupportMapFragment mapFragment = (SupportMapFragment)
        getSupportFragmentManager().findFragmentById(R.id.map);
    mapFragment.getMapAsync(this);
}

@Override
public void onPause() {
    super.onPause();
    //---remove the location listener---
    if (ActivityCompat.checkSelfPermission(this,
        Manifest.permission.ACCESS_COARSE_LOCATION)
        != PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this,
            new String[]{Manifest.permission.ACCESS_COARSE_LOCATION},
            REQUEST_COARSE_ACCESS);
        return;
    } else {
        permissionGranted = true;
    }
    if(permissionGranted) {
        lm.removeUpdates(locationListener);
    }
}

/** Manipulates the map once available. ...*/
@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;
    lm = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
    locationListener = new MyLocationListener();

    if (ActivityCompat.checkSelfPermission(this,
        android.Manifest.permission.ACCESS_FINE_LOCATION)
        != PackageManager.PERMISSION_GRANTED
        && ActivityCompat.checkSelfPermission(this,
            android.Manifest.permission.ACCESS_COARSE_LOCATION)
        != PackageManager.PERMISSION_GRANTED)
    {
        ActivityCompat.requestPermissions(this, new
            String[]{android.Manifest.permission.ACCESS_COARSE_LOCATION},
            REQUEST_COARSE_ACCESS);
        return;
    } else {
        permissionGranted = true;
    }
    if(permissionGranted) {
        lm.requestLocationUpdates(LocationManager.GPS_PROVIDER,
            0, 0, locationListener);
    }
}

```

```

@Override
public void onRequestPermissionsResult(int requestCode, String[] permissions,
                                       int[] grantResults) {
    switch (requestCode) {
        case REQUEST_COARSE_ACCESS:
            if (grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                permissionGranted = true;
            } else {
                permissionGranted = false;
            }
            break;
        default:
            super.onRequestPermissionsResult(requestCode, permissions,
                                             grantResults);
    }
}

private class MyLocationListener implements LocationListener {
    public void onLocationChanged(Location loc) {
        if(loc != null) {
            Toast.makeText(getApplicationContext(), "Location changed : Lat: \"" +
                    "+ loc.getLatitude() + "\n" + " Lng: \n" +
                    + loc.getLongitude(), Toast.LENGTH_SHORT).show();
            LatLng p = new LatLng((int) (loc.getLatitude()),
                                  (int) (loc.getLongitude()));
            mMap.moveCamera(CameraUpdateFactory.newLatLng(p));
            mMap.animateCamera(CameraUpdateFactory.zoomTo(7));
        }
    }
    public void onProviderDisabled(String provider) {}
    public void onProviderEnabled(String provider) {}
    public void onStatusChanged(String provider, int status, Bundle extras) {}
}
}

```

2. Tambahkan baris tebal berikut ke file **AndroidManifest.xml**. Pastikan untuk mengganti semua instance dari com.jfdimarzio dengan referensi ke paket yang digunakan dalam aplikasi:

```

<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"
/>

```

3. Tekan Shift + F9 untuk men-debug aplikasi pada emulator android.
4. Untuk mensimulasikan data GPS yang diterima oleh emulator android, dapat menggunakan alat Kontrol Lokasi di sisi kanan emulator.
5. Perhatikan bahwa peta pada emulator sekarang menjiwai ke lokasi lain (lihat Gambar 6), ini membuktikan bahwa aplikasi telah menerima data GPS.
 - Di Android, layanan berbasis lokasi disediakan oleh kelas LocationManager, yang terletak di paket android.location.

- Menggunakan kelas LocationManager, aplikasi dapat memperoleh pembaruan berkala dari lokasi geografis perangkat, serta mengaktifkan maksud saat memasuki kedekatan lokasi tertentu.
6. Dalam file MapsActivity.java
- Pertama-tama memeriksa izin untuk menggunakan Lokasi Kursus
 - Kemudian mendapatkan referensi ke kelas LocationManager menggunakan metode getSystemService (). Lakukan ini dalam metode onCreate () dari LBSActivity:

```
//---use the LocationManager class to obtain locations data--  
lm = (LocationManager) getSystemService(Context.LOCATION_SERVICE);  
locationListener = new MyLocationListener();
```



Gambar 6. Mendapatkan lokasi tertentu

7. Selanjutnya, membuat turunan dari kelas MyLocationListener, yang ditentukan nanti di kelas. Kelas MyLocationListener mengimplementasikan kelas abstrak LocationListener.

Diperlukan mengganti empat metode dalam implementasi ini:

- **onLocationChanged (Lokasi lokasi)** : Panggil ketika lokasi telah berubah
- **onProviderDisabled (Penyedia string)** : Dipanggil saat penyedia dinonaktifkan oleh pengguna
- **onProviderEnabled (Penyedia string)** : Panggil ketika penyedia diaktifkan oleh pengguna
- **onStatusChanged (Penyedia string, status int, Bundel ekstra)** : Panggil ketika status penyedia berubah

8. Dalam contoh, biasanya lebih tertarik dengan apa yang terjadi ketika lokasi berubah, jadi menulis kode di metode **onLocationChanged ()**. Khususnya, ketika lokasi berubah, dapat menampilkan dialog kecil di layar yang menunjukkan informasi lokasi baru: lintang dan bujur. Menampilkan dialog ini menggunakan kelas *Toast*:

```
public void onLocationChanged(Location loc) {  
    if(loc != null) {  
        Toast.makeText(getApplicationContext(), "Location changed : Lat: \"'" +  
            "+ loc.getLatitude() + "\n" + " \\"' Lng: '\n'" +  
            + loc.getLongitude(), Toast.LENGTH_SHORT).show();  
        LatLng p = new LatLng((int) (loc.getLatitude()),  
            (int) (loc.getLongitude()));  
        mMap.moveCamera(CameraUpdateFactory.newLatLng(p));  
        mMap.animateCamera(CameraUpdateFactory.zoomTo(7));  
    }  
}
```

9. Dalam metode sebelumnya, juga menavigasi peta ke lokasi yang telah diterima. Untuk diberi tahu setiap kali ada perubahan lokasi, harus mendaftarkan permintaan untuk perubahan lokasi sehingga program dapat diberi tahu secara berkala. Melakukan ini melalui metode **requestLocation-Updates ()** :

```
if(permissionGranted) {  
    lm.requestLocationUpdates(LocationManager.GPS_PROVIDER,  
        0,0, locationListener);  
}
```

Metode requestLocationUpdates () mengambil empat argumen:

- **provider** : Nama penyedia yang didaftarkan. Dalam hal ini, menggunakan GPS untuk mendapatkan data lokasi geografi.

- **minTime** : Interval waktu minimum untuk pemberitahuan, dalam milidetik. 0 menunjukkan bahwa developer ingin terus diberi tahu tentang perubahan lokasi.
- **minDistance** : Interval jarak minimum untuk notifikasi, dalam meter. 0 menunjukkan bahwa developer ingin terus diberi tahu tentang perubahan lokasi.
- **listener** : Objek yang metode `onLocationChanged ()` akan dipanggil untuk setiap pembaruan lokasi

10. Dalam metode **onPause ()**, menghapus pendengar saat aktivitas dihancurkan atau masuk ke latar belakang (sehingga aplikasi tidak lagi mendengarkan perubahan lokasi, sehingga menghemat baterai perangkat). Dapat melakukannya menggunakan metode **removeUpdates ()**:

```
Override
public void onPause() {
    super.onPause();
    //---remove the location listener---
    if (ActivityCompat.checkSelfPermission(this,
        Manifest.permission.ACCESS_COARSE_LOCATION)
        != PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this,
            new String[]{Manifest.permission.ACCESS_COARSE_LOCATION},
            REQUEST_COARSE_ACCESS);
        return;
    } else {
        permissionGranted = true;
    } if(permissionGranted) {
        lm.removeUpdates(locationListener);}}
```

Dapat menggabungkan penyedia lokasi GPS dengan penyedia lokasi jaringan dalam aplikasi:

```
@Override
public void onResume() {
    super.onResume();
    //---request for location updates---
    lm.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0,
        locationListener);
    //---request for location updates---
    lm.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, 0, 0,
        locationListener);
}
```

- Namun, perlu diketahui bahwa hal itu akan menyebabkan aplikasi menerima dua set koordinat yang berbeda, karena penyedia GPS dan penyedia NETWORK akan mencoba untuk mendapatkan lokasi menggunakan metode mereka sendiri (GPS versus Wi-Fi dan triangulasi ID sel).

- Karenanya, penting untuk memantau status kedua penyedia di perangkat dan menggunakan yang sesuai. Untuk memeriksa status kedua penyedia dengan menerapkan tiga metode berikut (ditunjukkan dalam huruf tebal) dari kelas **MyLocationListener**:

```

private class MyLocationListener implements LocationListener {
    public void onLocationChanged(Location loc) {
        if(loc != null) {
            Toast.makeText(getApplicationContext(), "Location changed : Lat: \""
                    "+ loc.getLatitude() + "\n" + " Lng: \n"
                    + loc.getLongitude(), Toast.LENGTH_SHORT).show();
            LatLng p = new LatLng((int) (loc.getLatitude()),
                    (int) (loc.getLongitude()));
            mMap.moveCamera(CameraUpdateFactory.newLatLng(p));
            mMap.animateCamera(CameraUpdateFactory.zoomTo(7));
        }
    }
    public void onProviderDisabled(String provider)
    {
        Toast.makeText(getApplicationContext(), provider + " Disabled",
                Toast.LENGTH_SHORT).show();
    }
    public void onProviderEnabled(String provider)
    {
        Toast.makeText(getApplicationContext(), provider + " Enabled",
                Toast.LENGTH_SHORT).show();
    }
    public void onStatusChanged(String provider, int status, Bundle extras)
    {
        String statusString = "";
        switch (status) {
            case LocationProvider.AVAILABLE:
                statusString = "Available";
            case LocationProvider.OUT_OF_SERVICE:
                statusString = "Out of Service";
            case LocationProvider.TEMPORARILY_UNAVAILABLE:
                statusString = "Temporarily Unavailable";
        }
        Toast.makeText(getApplicationContext(), provider + " " + statusString,
                Toast.LENGTH_SHORT).show();
    }
}

```

Monitoring lokasi

Salah satu fitur yang sangat keren dari kelas LocationManager adalah kemampuannya untuk memantau lokasi tertentu. Ini dicapai dengan menggunakan metode addProximityAlert () .

1. Implementasikan kode berikut untuk menunjukkan cara memonitor lokasi tertentu sehingga jika pengguna berada dalam radius lima meter dari lokasi itu, aplikasi akan meluncurkan niat untuk meluncurkan browser web:

```
import android.app.PendingIntent;
import android.content.Intent;
import android.net.Uri;
    //---use the LocationManager class to obtain locations data---
lm = (LocationManager) getSystemService(Context.LOCATION_SERVICE);

    //---PendingIntent to launch activity if the user is within
    // some locations--
PendingIntent pendingIntent = PendingIntent.getActivity(this, 0, new
Intent(android.content.Intent.ACTION_VIEW, Uri.parse("http://www.amazon.com")), 0);
lm.addProximityAlert(37.422006, -122.084095, 5, -1, pendingIntent);
```

- Metode addProximityAlert () membutuhkan lima argumen:
 - Lintang
 - Garis bujur
 - Radius (dalam meter)
 - Kedaluwarsa (durasi untuk kedekatan peringatan ini valid, setelah itu dihapus; –1 tanpa kedaluwarsa)
 - Intent yang tertunda
- Perhatikan bahwa jika layar perangkat Android mati, kedekatan juga diperiksa setiap empat menit untuk menjaga masa pakai baterai perangkat.

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan implementasi code untuk mendapatkan lokasi dengan menentukan lokasi fisik pengguna memonitoring lokasi tertentu.
2. Dokumentasi hasil output aplikasi dalam bentuk laporan dilengkapi dengan analisa code pada implementasi source code.

g. Kesimpulan

Mahasiswa mendapatkan lokasi dengan menentukan lokasi fisik pengguna dan mampu memonitoring lokasi untuk memantau lokasi tertentu.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara detail setiap baris code untuk mendapatkan lokasi	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara singkat setiap baris code untuk mendapatkan lokasi	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam implementasi code serta menjelaskan setiap baris code untuk mendapatkan lokasi	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam implementasi code serta menjelaskan setiap baris code untuk mendapatkan lokasi	(1)	2	3	4
Jumlah skor					

ACARA 35 : GPS DAN AKSES SENSOR

Pokok Bahasan	: Pengaturan Android Manifest, Asset dan Activity
Acara Praktikum/Pertemuan	: Minggu 10 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami sensor hardware pada android.
2. Mahasiswa mampu mengakses dan menggunakan perangkat sensor yang tertanam di dalam *smartphone* android.
3. Mahasiswa mampu melakukan pengaturan android manifest, menambahkan asset dan *activity* yang perlu dibuat.

b. Indikator

1. Keberhasilan mahasiswa mengakses dan menggunakan perangkat sensor yang tertanam di dalam *smartphone* android.
2. Keberhasilan mahasiswa melakukan pengaturan android manifest, menambahkan asset dan *activity* yang perlu dibuat.

c. Dasar Teori

Akses Sensor

Akses sensor mengular bagaimana cara mengakses suatu sensor yang tertanam di dalam *smartphone*. Sensor merupakan alat yang terpasang pada mobile *device* untuk mengukur, menganalisis, dan memantau kondisi lingkungan. Sensor juga dapat memberikan aksi dari perubahan kondisi tersebut.

Sensor dikelompokkan menjadi 3 jenis yaitu Sensor Gerak (*Motion Sensor*), Sensor Lingkungan (*Environment Sensor*), serta Sensor Posisi (*Position Sensor*). Sensor gerak memantau perubahan kondisi menggunakan sepanjang 3 sumbu X, Y, dan Z (3 variabel). Contoh sensor gerak antara lain *Accelerometer*, *Gyroscope*, dan *Rotation sensor*.

Sensor lingkungan berfungsi untuk mengukur berbagai parameter lingkungan sekitar ponsel, seperti suhu dan tekanan udara, pencahayaan, dan kelembapan. Contoh sensor lingkungan antara

lain *barometers* (tekanan udara), *photometers* (pencahayaan), dan *thermometers* (suhu). Nilai data yang diambil pada sensor lingkungan hanya pada array ke-1 (1 variabel).

Sensor posisi berfungsi untuk mengukur seberapa dekat ponsel dengan objek sekitar. Sensor kategori ini ialah *proximity* (cm), *orientation* (derajat) dan *magnetic* (μT /mikroTesla).

d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

e. Prosedur Kerja

Bagian mempraktikkan implementasi sensor pada *smartphone*. Terdiri dari pengaturan Android Manifest, menambahkan asset, *activity* yang perlu dibuat.

Android Manifest

- Agar sensor dapat bekerja secara maksimal, maka diperlukan *permission* WAKE_LOCK.
- *Permission*, ini bertujuan agar layar dan CPU selalu terjaga sehingga *smartphone* dapat maksimal memantau perubahan kondisi lingkungan.
- Pengaturan lainnya yaitu pada *provider* bertujuan agar sistem mampu membaca file mp3 yang digunakan untuk memutar suara ketika terjadi perubahan nilai sensor.

1. Tambahkan kode Android Manifest dibawah ini pada proyek LBS yang sudah dibuat agar aplikasi berjalan lancar.

```
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3   package="com.choirulhuda.sensors">
4
5     <uses-permission android:name="android.permission.WAKE_LOCK" />
6 
```

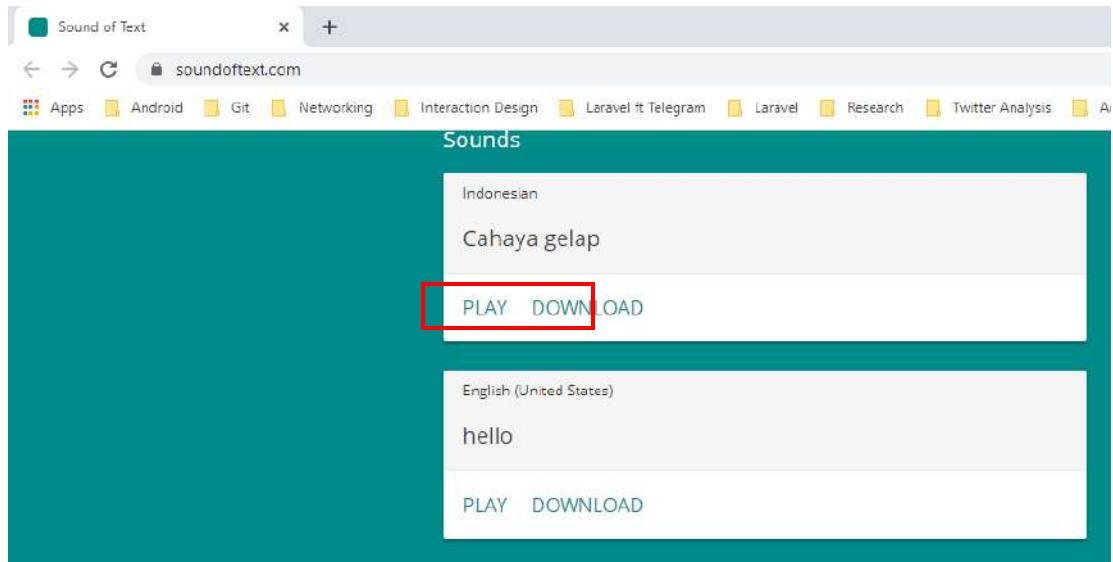
2. Tambah kode provider pada bagian paling bawah

```
27   <provider
28     android:name="android.support.v4.content.FileProvider"
29     android:authorities="${applicationId}.fileprovider"
30     android:exported="false"
31     android:grantUriPermissions="true">
32       <meta-data
33         android:name="android.support.FILE_PROVIDER_PATHS"
34         android:resource="@xml/file_paths" />
35     </provider>
36   </application>
37
38 </manifest>
```

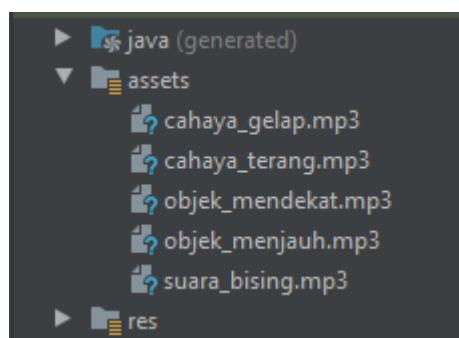
Menambahkan Asset

- Aset yang digunakan yaitu file **mp3** dalam bentuk bahasa.
- File ini diperoleh dengan cara mengakses tautan <https://soundoftext.com/> kemudian tuliskan **cahaya gelap**.

1. Unduh file tersebut seperti pada gambar di bawah ini.



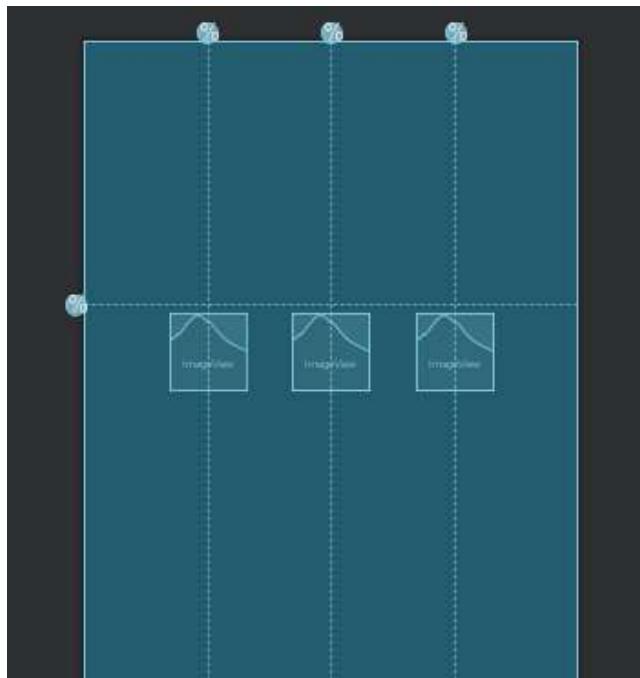
2. Lanjutkan dengan mengunduh file dengan suara **cahaya gelap, objek menjauh, objek mendekat**.
3. Simpan semua file tersebut dengan nama sesuai suara. Misalkan pada cahaya gelap, simpan dengan nama **cahaya_gelap.mp3**
4. Tambahkan folder asset pada proyek android dengan cara klik kanan pada **app → New → Folder → Assets Folder**.
5. Masukkan semua file mp3 kedalam folder tersebut dengan cara *copy paste*. Hasilnya terlihat pada gambar dibawah ini.



Main Activity

Main Activity digunakan sebagai activity utama untuk menampilkan 3 tombol yang mewakili aksi menampilkan semua sensor di dalam *smartphone*, Sensor Cahaya (*Light Sensor*), dan Sensor Kedekatan (*Proximity Sensor*).

1. Perhatikan UI beserta kode XML pada activity_main.xml dibawah ini. 3 tombol pada UI berupa ImageView yang akan diberikan aksi **OnClick**.



2. Agar lebih mudah menampilkan UI, tambahkan kode XML dibawah ini pada proyek android.

```
2 <?xml version="1.0" encoding="utf-8"?>
3 <android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     xmlns:tools="http://schemas.android.com/tools"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     tools:context=".MainActivity">
9
10    <ImageView
11        android:id="@+id/img_list_all_sensor"
12        android:layout_width="wrap_content"
13        android:layout_height="wrap_content"
14        android:layout_marginTop="8dp"
15        app:layout_constraintEnd_toStartOf="@+id/guideline"
16        app:layout_constraintStart_toStartOf="@+id/guideline"
17        app:layout_constraintTop_toTopOf="@+id/guideline4"
18        app:srcCompat="@drawable/ic_format_list" />
```

```
18
19 <android.support.constraint.Guideline
20     android:id="@+id/guideline"
21     android:layout_width="wrap_content"
22     android:layout_height="wrap_content"
23     android:orientation="vertical"
24     app:layout_constraintGuide_percent="0.25" />
25
26 <android.support.constraint.Guideline
27     android:id="@+id/guideline2"
28     android:layout_width="wrap_content"
29     android:layout_height="wrap_content"
30         android:orientation="vertical"
31         app:layout_constraintGuide_percent="0.50121653" />
32
33 <android.support.constraint.Guideline
34     android:id="@+id/guideline3"
35     android:layout_width="wrap_content"
36     android:layout_height="wrap_content"
37     android:orientation="vertical"
38     app:layout_constraintGuide_percent="0.75" />
39
40 <android.support.constraint.Guideline
41     android:id="@+id/guideline4"
42     android:layout_width="wrap_content"
43     android:layout_height="wrap_content"
44     android:orientation="horizontal"
45     app:layout_constraintGuide_percent="0.3" />
46
47 <ImageView
48     android:id="@+id/img_light_sensor"
49     android:layout_width="wrap_content"
50     android:layout_height="wrap_content"
51     android:layout_marginStart="8dp"
52     android:layout_marginTop="8dp"
53     android:layout_marginEnd="8dp"
54     app:layout_constraintEnd_toStartOf="@+id/guideline2"
55     app:layout_constraintStart_toStartOf="@+id/guideline2"
56     app:layout_constraintTop_toTopOf="@+id/guideline4"
57     app:srcCompat="@drawable/ic_highlight" />
58
59 <ImageView
60     android:id="@+id/img_proximity_sensor"
61     android:layout_width="wrap_content"
62     android:layout_height="wrap_content"
63     android:layout_marginStart="8dp"
64     android:layout_marginTop="8dp"
65     android:layout_marginEnd="8dp"
66     app:layout_constraintEnd_toStartOf="@+id/guideline3"
67     app:layout_constraintStart_toStartOf="@+id/guideline3"
68     app:layout_constraintTop_toTopOf="@+id/guideline4"
69     app:srcCompat="@drawable/ic_proximity" />
70
71 </android.support.constraint.ConstraintLayout>
```

3. Implementasikan OnClickListener pada Main Activity dan berikan aksi pada masing-masing tombol yang sudah dibuat seperti dibawah ini.
- Namun sebelum itu, buatlah 3 activity baru terdiri dari :
 - **AllSensorActivity**
 - **LightSensorActivity**
 - **ProximitySensorActivity**
 - Berikan intent pada masing-masing tombol yang mengarah pada ketiga activity baru tersebut, seperti kode dibawah ini:

```
9  public class MainActivity extends AppCompatActivity implements View.OnClickListener {
10
11     private ImageView imgAllSensor, imgProximitySensor, imgLightSensor;
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.activity_main);
17
18         imgAllSensor = findViewById(R.id.img_list_all_sensor);
19         imgLightSensor = findViewById(R.id.img_light_sensor);
20         imgProximitySensor = findViewById(R.id.img_proximity_sensor);
21         imgAllSensor.setOnClickListener(this);
22         imgLightSensor.setOnClickListener(this);
23         imgProximitySensor.setOnClickListener(this);
24     }
25
26     @Override
27     public void onClick(View view) {
28         int id = view.getId();
29         switch (id) {
30
31             case R.id.img_list_all_sensor:
32                 Intent intentAll = new Intent(getApplicationContext(), AllSensorActivity.class);
33                 startActivity(intentAll);
34                 break;
35
36             case R.id.img_light_sensor:
37                 Intent intentLight = new Intent(getApplicationContext(), LightSensorActivity.class);
38                 startActivity(intentLight);
39                 break;
40
41             case R.id.img_proximity_sensor:
42                 Intent intent = new Intent(getApplicationContext(), ProximitySensorActivity.class);
43                 startActivity(intent);
44                 break;
45
46         }
47     }
48 }
49 }
```

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan implementasi code untuk mengakses dan menggunakan perangkat sensor yang tertanam di dalam *smartphone* android.
2. Dokumentasi hasil output aplikasi dalam bentuk laporan dilengkapi dengan analisa code pada implementasi source code.

g. Kesimpulan

Mahasiswa mengakses dan menggunakan perangkat sensor yang tertanam di dalam smartphone android dengan melakukan pengaturan android manifest, menambahkan asset dan activity yang perlu dibuat.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara detail setiap baris code untuk pengaturan android manifest, menambahkan asset dan <i>activity</i> yang dibuat	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara singkat setiap baris code untuk pengaturan android manifest, menambahkan asset dan <i>activity</i> yang dibuat	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam implementasi code serta menjelaskan setiap baris code untuk pengaturan android manifest, menambahkan asset dan <i>activity</i> yang dibuat	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam implementasi code serta menjelaskan setiap baris code untuk pengaturan android manifest, menambahkan asset dan <i>activity</i> yang dibuat	(1)	2	3	4
Jumlah skor					

ACARA 36 : GPS DAN AKSES SENSOR

Pokok Bahasan	: Menangkap Perubahan Nilai dari Suatu Sensor
Acara Praktikum/Pertemuan	: Minggu 10 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami sensor hardware pada android.
2. Mahasiswa mampu mengakses dan menggunakan perangkat sensor yang tertanam di dalam *smartphone* android.
3. Mahasiswa mampu menampilkan semua sensor.
4. Mahasiswa mampu menangkap perubahan nilai dengan implementasi sensor light dan sensor proximity.

b. Indikator

1. Keberhasilan mahasiswa menampilkan semua sensor.
2. Keberhasilan mahasiswa menangkap perubahan nilai dengan implementasi sensor light dan sensor proximity.

c. Dasar Teori

Menampilkan Semua Sensor

Activity ini digunakan untuk menampilkan semua sensor yang tersedia di dalam *smartphone*. UI berupa ListView yang memiliki data berupa Nama Sensor, Tipe, Power (Tenaga yang digunakan), serta Nilai Range Maksimal yang dapat ditangkap oleh sensor.

Menangkap Perubahan Nilai

Perubahan pada nilai dapat ditangkap melalui parameter **sensorEvent** yang diperoleh dari fungsi **onSensorChanged** ketika mengimplementasikan **SensorEventListener**. Pada bagian ini, sensor yang akan diimplementasikan yaitu sensor **light** dan sensor **proximity**.

▪ **Light Sensor**

Light sensor digunakan untuk mendeteksi tingkat pencahayaan pada lingkungan sekitar. Sensor ini hanya menggunakan 1 variabel array pada parameter **sensorEvent**. Agar dapat

mengetahui perbedaan kondisi pencahayaan, maka UI pada sensor light akan ditampilkan dalam bentuk grafik.

- **Proximity sensor**

Proximity sensor digunakan untuk mendeteksi kedekatan dengan objek terdekat. Sensor ini digunakan untuk memematikan layar ketika menerima penggilan agar dapat menhemat data baterai

Kedua sensor ini hanya menggunakan 1 variabel array pada parameter sensorEvent. Namun, ketika menggunakan sensor gerak, maka sistem harus menggunakan 3 variabel array.

d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

e. Prosedur Kerja

Buka kembali proyek LBS yang sudah dibuat pada acara sebelumnya, dan tambahkan beberapa kode.

Menampilkan Semua Sensor

1. Berikut ini kode program dari **activity_all_sensor.xml**

```
2 <android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     tools:context=".AllSensorActivity">
8
9     <ListView
10         android:id="@+id/lv_sensors"
11         android:layout_width="0dp"
12         android:layout_height="0dp"
13         android:layout_marginStart="8dp"
14         android:layout_marginTop="8dp"
15         android:layout_marginEnd="8dp"
16         android:layout_marginBottom="8dp"
17         app:layout_constraintBottom_toBottomOf="parent"
18         app:layout_constraintEnd_toEndOf="parent"
19         app:layout_constraintStart_toStartOf="parent"
20         app:layout_constraintTop_toTopOf="parent" />
21     </android.support.constraint.ConstraintLayout>
```

- Masing-masing **item** pada listview harus dibuat XML terpisah, karena akan diulang terus-menerus.

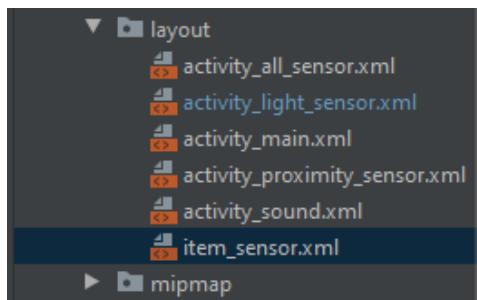
- Oleh sebab itu, perlu dibuat XML baru dengan nama **item_sensor.xml** dengan rootnya berupa LinearLayout didalam nya berisi **TextView**.
2. Tambahkan XML pada proyek android dan tambahkan kode dibawah ini:

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:orientation="vertical" android:layout_width="match_parent"
4      android:layout_height="match_parent">
5      <TextView
6          android:id="@+id/txt_item_sensor"
7          android:layout_width="wrap_content"
8          android:layout_height="wrap_content" />
9
10 </LinearLayout>

```

3. Tampilan pada folder layout pada proyek android



4. Tambahkan kode dibawah ini pada **AllSensorActivity.java**, agar bisa menampilkan daftar sensor yang tersedia

```

17  public class AllSensorActivity extends AppCompatActivity {
18
19     @Override
20     protected void onCreate(Bundle savedInstanceState) {
21         super.onCreate(savedInstanceState);
22         setContentView(R.layout.activity_all_sensor);
23         SensorManager sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
24         List<Sensor> sensorList = sensorManager.getSensorList(Sensor.TYPE_ALL);
25         ListView listView = findViewById(R.id.lv_sensors);
26         listView.setAdapter(new MySensorAdapter(this, R.layout.item_sensor, sensorList));
27     }
28
29     class MySensorAdapter extends ArrayAdapter<Sensor> {
30
31         private int txtResourceID;
32
33         public MySensorAdapter( Context context, int resource, List<Sensor> objects) {
34             super(context, resource, objects);
35             this.txtResourceID = resource;
36         }
37
38         public class ViewHolder {
39             private TextView itemView;
40         }

```

```
41.     public View getView(int position, View convertView, ViewGroup parent) {
42.         int idSensor = position + 1;
43.
44.         ViewHolder viewHolder = null;
45.         if (convertView == null) {
46.             convertView = LayoutInflater.from(this.getContext())
47.                 .inflate(txtResourceID, parent, attachToRoot: false);
48.             viewHolder = new ViewHolder();
49.             viewHolder.itemView = convertView.findViewById(R.id.txt_item_sensor);
50.             convertView.setTag(viewHolder);
51.         } else {
52.             viewHolder = (ViewHolder) convertView.getTag();
53.         }
54.
55.         Sensor item = getItem(position);
56.
57.         if (item != null) {
58.             viewHolder.itemView.setText("IdSensor: " + item.getIdSensor() + "\nName: " + item.getName() + "\nType: " + item.getType()
59.             + "\nPower: " + item.getPower() + "mA\nMax Range: " + item.getMaximumRange());
60.         }
61.
62.         return convertView;
63.     }
64. }
65. }
```

Menangkap Perubahan Nilai

Penerapan sensor Proximity dan Light, akan dibahas pada subbab di bawah ini.

A. Light Sensor

1. Tambahkan sintak dibawah ini pada build.gradle android dan jalankan **sync now** :

```
implementation 'com.jjoe64:graphview:4.2.2'
```

2. Buatlah UI pada XML seperti pada gambar dibawah ini, perhatikan kodenya dan tambahkan pada proyek android.

```
2. <android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3.     xmlns:app="http://schemas.android.com/apk/res-auto"
4.     xmlns:tools="http://schemas.android.com/tools"
5.     android:layout_width="match_parent"
6.     android:layout_height="match_parent"
7.     tools:context=".LightSensorActivity">
8.
9.     <TextView
10.         android:id="@+id/txt_brightness_info"
11.         android:layout_width="wrap_content"
12.         android:layout_height="wrap_content"
13.         android:layout_marginStart="8dp"
14.         android:layout_marginTop="8dp"
15.         android:layout_marginEnd="8dp"
16.         android:fontFamily="monospace"
17.         android:text="Brightness Level"
18.         android:textSize="30sp"
19.         android:textStyle="bold"
20.         app:layout_constraintEnd_toEndOf="parent"
21.         app:layout_constraintStart_toStartOf="parent"
22.         app:layout_constraintTop_toTopOf="@+id/guideline5" />
23. 
```

```
24     <com.jjoe64.graphview.GraphView
25         android:id="@+id/graph_light"
26         android:layout_width="0dp"
27         android:layout_height="0dp"
28         android:layout_marginStart="8dp"
29         android:layout_marginTop="16dp"
30         android:layout_marginEnd="8dp"
31         android:layout_marginBottom="32dp"
32         app:layout_constraintBottom_toBottomOf="parent"
33         app:layout_constraintEnd_toEndOf="parent"
34         app:layout_constraintStart_toStartOf="parent"
35         app:layout_constraintTop_toBottomOf="@+id/txt_brightness_info" />
36
37     <android.support.constraint.Guideline
38         android:id="@+id/guideline5"
39         android:layout_width="wrap_content"
40         android:layout_height="wrap_content"
41         android:orientation="horizontal"
42         app:layout_constraintGuide_percent="0.12" />
43
44 </android.support.constraint.ConstraintLayout>
```



3. Buat fungsi di dalam **LightSensorActivity.java** untuk menginisiasi grafik dan menampilkan garis grafiknya, seperti pada kode di bawah ini:

```

    public GraphView initGraph(int id, String title) {
        GraphView graph = (GraphView) findViewById(id);
        graph.getViewport().setXAxisBoundsManual(true);
        graph.getViewport().setMinX(0);
        graph.getViewport().setMaxX(5);
        graph.getGridLabelRenderer().setLabelVerticalWidth(100);
        graph.setTitle(title);
        graph.getGridLabelRenderer().setHorizontalLabelsVisible(false);
        graph.getLegendRenderer().setVisible(true);
        graph.getLegendRenderer().setAlign(LegendRenderer.LegendAlign.TOP);
        return graph;
    }

    public LineGraphSeries<DataPoint> initSeries(int color, String title) {
        LineGraphSeries<DataPoint> series;
        series = new LineGraphSeries<>();
        series.setDrawDataPoints(true);
        series.setDrawBackground(true);
        series.setColor(color);
        series.setTitle(title);
        series.setBackgroundColor(Color.argb(alpha: 100, red: 204, green: 119, blue: 119));
        return series;
    }
}

```

4. Implementasikan SensorEventLisneter pada Activity, kemudian *generate* fungsi yang wajib dibuat seperti **onSensorChanged()** dan **onAccuracyChanged()**. Perhatikan kode dibawah ini:

```

22  public class LightSensorActivity extends AppCompatActivity implements SensorEventListener {
23
24      private SensorManager sensorManager;
25      private Sensor lightSensor;
26      private TextView txtBrightnessInfo;
27      private MediaPlayer mPlayer;
28      private GraphView mGraphLight;
29      private LineGraphSeries<DataPoint> mSeriesLight;
30      private double graphLastAccelXValue = 5d;
31
32      @Override
33      protected void onCreate(Bundle savedInstanceState) {
34          super.onCreate(savedInstanceState);
35          setContentView(R.layout.activity_light_sensor);
36          txtBrightnessInfo = findViewById(R.id.txt_brightness_info);
37
38          sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
39          lightSensor = sensorManager.getDefaultSensor(Sensor.TYPE_LIGHT);
40          if (lightSensor == null) {
41              txtBrightnessInfo.setText("Sensor tidak tersedia");
42          } else {
43              sensorManager.registerListener(listener: this, lightSensor, SensorManager.SENSOR_DELAY_NORMAL);
44          }
45
46          mGraphLight = initGraph(R.id.graph_light, title: "Sensor of Light");
47          mSeriesLight = initSeries(Color.RED, title: "Lux");
48          mGraphLight.addSeries(mSeriesLight);
49
50      }
}

```

5. Perhatikan kode dibawah ini. Bagian menjabarkan bagaimana mengambil suatu nilai pada tipe sensor tertentu dan menampilkan hasilnya dalam bentuk grafik.

```

52     @Override
53     public void onSensorChanged(SensorEvent sensorEvent) {
54         int sensorType = sensorEvent.sensor.getType();
55
56         switch (sensorType) {
57             case Sensor.TYPE_LIGHT:
58                 txtBrightnessInfo.setText(getResources().getString(R.string.label_brightness, sensorEvent.values[0]));
59                 //graph view
60                 graphLastAccelXValue += 0.15d;
61                 mSeriesLight.appendData(new DataPoint(graphLastAccelXValue, sensorEvent.values[0]),
62                                         scrollToEnd: true, maxDataPoints: 33);
63
64             if (sensorEvent.values[0] == 0) {
65                 mPlayer = new MediaPlayer();
66
67                 try {
68                     AssetFileDescriptor as = this.getAssets().openFd(fileName: "cahaya_gelap.mp3");
69                     mPlayer.setDataSource(as.getFileDescriptor(), as.getStartOffset(), as.getLength());
70                     as.close();
71                     mPlayer.prepare();
72                     mPlayer.start();
73                 } catch (IOException e) {
74                     e.printStackTrace();
75                 }
76             } else if (sensorEvent.values[0] > 150){
77                 mPlayer = new MediaPlayer();
78             }
79         }
80     }
81
82     try {
83         AssetFileDescriptor as = this.getAssets().openFd(fileName: "cahaya_terang.mp3");
84         mPlayer.setDataSource(as.getFileDescriptor(), as.getStartOffset(), as.getLength());
85         as.close();
86         mPlayer.prepare();
87         mPlayer.start();
88     } catch (IOException e) {
89         e.printStackTrace();
90     }
91     break;
92 }
93
94     @Override
95     public void onAccuracyChanged(Sensor sensor, int i) {
96
97 }
98
99     @Override
100    public void onBackPressed() {
101        super.onBackPressed();
102        sensorManager.unregisterListener(listener: this, lightSensor);
103    }

```

6. Berikut string pada **label_brightness** agar dapat menampilkan nilai yang terus berubah

```
<string name="label_brightness">Level : %1$.2f lux</string>
```

```

78
79     try {
80         AssetFileDescriptor as = this.getAssets().openFd(fileName: "cahaya_gelap.mp3");
81         mPlayer.setDataSource(as.getFileDescriptor(), as.getStartOffset(), as.getLength());
82         as.close();
83         mPlayer.prepare();
84         mPlayer.start();
85     } catch (IOException e) {
86         e.printStackTrace();
87     }
88     break;
89 }
90
91     @Override
92     public void onAccuracyChanged(Sensor sensor, int i) {
93
94 }
95
96     @Override
97     public void onBackPressed() {
98        super.onBackPressed();
99        sensorManager.unregisterListener(listener: this, lightSensor);
100    }

```

7. Build dan run project tersebut, lalu perhatikan hasilnya.

Proximity sensor

- Pertama, buat UI XML dengan satu **TextView** seperti dibawah ini:

```
2 <android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     tools:context=".ProximitySensorActivity">
8
9     <TextView
10         android:id="@+id/txt_information"
11         android:layout_width="wrap_content"
12         android:layout_height="wrap_content"
13         android:layout_marginStart="8dp"
14         android:layout_marginTop="8dp"
15         android:layout_marginEnd="8dp"
16         android:layout_marginBottom="8dp"
17         android:fontFamily="monospace"
18         android:text="TextView"
19         android:textSize="30sp"
20         android:textStyle="bold"
21         app:layout_constraintBottom_toBottomOf="parent"
22         app:layout_constraintEnd_toEndOf="parent"
23         app:layout_constraintStart_toStartOf="parent"
24         app:layout_constraintTop_toTopOf="parent" />
25 </android.support.constraint.ConstraintLayout>
```

- Buatlah label pada string.xml untuk menampilkan perubahan nilai secara terus dengan mudah, seperti string pada *light sensor*.

```
<string name="label_proximity">Objek : %1$.2f cm</string>
```

- Implementasikan SensorEventListener pada activity ProximitySensor, kemudian override method **onSensorChanged()** dan **onAccuracyChanged()**.
- Panggil Sensor Service menggunakan variabel Sensor Manager, lanjutkan dengan memberikan nilai TYPE_PROXIMITY pada variabel Sensor.
- Pada method **onBackPressed()**, matikan sensor dengan memanggilan **unregisterListener** agar dapat menghemat daya baterai.
- Selengkapnya, tulislah kode berikut ini pada proyek android.

```
23  public class ProximitySensorActivity extends AppCompatActivity implements SensorEventListener {
24
25      private TextView txtInfo;
26      private SensorManager sensorManager;
27      private Sensor sensorProximity;
28      private MediaPlayer mPlayer;
29
30
31      @Override
32      protected void onCreate(Bundle savedInstanceState) {
33          super.onCreate(savedInstanceState);
34          setContentView(R.layout.activity_proximity_sensor);
35
36          txtInfo = findViewById(R.id.txt_information);
37          txtInfo.setText("Loading...");
38
39          sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
40          sensorProximity = sensorManager.getDefaultSensor(Sensor.TYPE_PROXIMITY);
41
42
43          if (sensorProximity == null) {
44              txtInfo.setText("Sensor tidak tersedia");
45          } else {
46              sensorManager.registerListener(this, sensorProximity, SensorManager.SENSOR_DELAY_NORMAL);
47          }
48
49      }
50
51      @Override
52      public void onBackPressed() {
53          super.onBackPressed();
54          sensorManager.unregisterListener(this, sensorProximity);
55      }
56
57      @Override
58      public void onSensorChanged(SensorEvent sensorEvent) {
59          int sensorType = sensorEvent.sensor.getType();
60
61          switch (sensorType) {
62              case Sensor.TYPE_PROXIMITY:
63                  txtInfo.setText(getResources().getString(R.string.label_proximity, sensorEvent.values[0]));
64
65                  if (sensorEvent.values[0] == 0) {
66                      mPlayer = new MediaPlayer();
67
68                      try {
69                          AssetFileDescriptor as = this.getAssets().openFd("objek_mendekat.mp3");
70                          mPlayer.setDataSource(as.getFileDescriptor(), as.getStartOffset(), as.getLength());
71                          as.close();
72                          mPlayer.prepare();
73                          mPlayer.start();
74                      } catch (IOException e) {
75                          e.printStackTrace();
76                      }
77                  } else {
78                      mPlayer = new MediaPlayer();
79                  }
80
81          }
82      }
83
84      @Override
85      public void onAccuracyChanged(Sensor sensor, int accuracy) {
86
87      }
88  }
```

```

79
80     }
81     try {
82         AssetFileDescriptor as = this.getAssets().openFd("objek_menjauh.mp3");
83         mPlayer.setDataSource(as.getFileDescriptor(), as.getStartOffset(), as.getLength());
84         as.close();
85         mPlayer.prepare();
86         mPlayer.start();
87     } catch (IOException e) {
88         e.printStackTrace();
89     }
90     break;
91 }
92 }
93
94 @Override
95 public void onAccuracyChanged(Sensor sensor, int i) {
96
97 }
98 }
99

```

f. Hasil dan Pembahasan

- Project aplikasi dari percobaan implementasi code untuk menampilkan semua sensor dengan menangkap perubahan nilai dengan implementasi sensor light dan sensor proximity.
- Dokumentasi hasil output aplikasi dalam bentuk laporan dilengkapi dengan analisa code pada implementasi source code.

g. Kesimpulan

Mahasiswa menampilkan semua sensor dengan menangkap perubahan nilai dengan implementasi sensor light dan sensor proximity.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara detail setiap baris code untuk menampilkan semua sensor	1	2	3	4
2	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara singkat setiap baris code untuk menampilkan semua sensor	1	2	3	4
3	Ketepatan waktu akan tetapi kurang tepat dalam implementasi code serta menjelaskan setiap baris code untuk menampilkan semua sensor	1	2	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam implementasi code serta menjelaskan setiap baris code untuk menampilkan semua sensor perlu dibuat	1	2	3	4
Jumlah skor					

ACARA 37 : PENGENALAN JSON

Pokok Bahasan	: Pengenalan JSON
Acara Praktikum/Pertemuan	: Minggu 11 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep JSON.
2. Mahasiswa mampu menerapkan JSON.

b. Indikator

Keberhasilan mahasiswa memahami konsep JSON dan menerapkan JSON pada android.

c. Dasar Teori

JSON



JSON adalah singkatan dari JavaScript Object Notation. Ini adalah format minimal yang dapat dibaca dan untuk menyusun data. JSON adalah transmisi data independen antara server dan aplikasi. Merupakan alternatif terbaik untuk XML. Ini mewakili data dalam format teks sehingga dapat dengan mudah di urai.

Android telah menyediakan kelas yang berbeda untuk memanipulasi data JSON. Mereka adalah JSONArray, JSONStringer, JSONObject dan JSONTokenizer. File JSON memiliki banyak komponen seperti Array ([], Objects ({ }), Key dan Value.

JSON digunakan pada android untuk keperluan aplikasi android itu sendiri, terkadang suatu aplikasi android membutuhkan sebuah data yang berasal dari suatu server untuk ditampilkan dalam aplikasi android tersebut. Dan untuk mengambil data yang terdapat dari server kita dapat menggunakan API (Application Programming Interface).

API (Application Programming Interface) adalah sekumpulan perintah, fungsi, protokol, dan objek yang dapat digunakan oleh seorang programmer untuk membuat aplikasi perangkat lunak (aplikasi) berinteraksi dengan sistem eksternal (server).

d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

e. Prosedur Kerja

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama JsonMovieDB
3. Menambahkan dependency pada **build.gradle** seperti berikut kemudian sync

```
1      implementation 'com.google.android.material:material:1.2.0'
2      implementation 'com.amitshekhar.android:android-networking:1.0.2'
3      implementation 'com.ramotion.cardslider:card-slider:0.3.1'
4      implementation 'com.github.florent37:diagonallayout:1.0.9'
5      implementation 'com.flaviofaria:kenburnsview:1.0.7'
6      implementation 'com.github.ivbaranov:materialfavoritebutton:0.1.5'
7      implementation 'com.github.bumptech.glide:glide:4.11.0'
8      annotationProcessor 'com.github.bumptech.glide:compiler:4.11.0'
9 }
```

4. Mengubah file **string.xml** seperti berikut

```
1 <resources>
2     <string name="app_name">JSON and API</string>
3 </resources>
```

5. Mengubah file **activity_layout.xml** seperti berikut

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:orientation="vertical"
8     tools:context="com.example.myapplication.MainActivity">
```

```
9      <TextView
10     android:layout_width="160dp"
11     android:layout_height="wrap_content"
12     android:paddingTop="100dp"
13     android:textSize="50dp"
14     android:layout_gravity="center_horizontal"
15     android:textAlignment="center"
16     android:text="JSON" />
17
18     <LinearLayout
19       android:layout_width="match_parent"
20       android:layout_height="wrap_content"
21       android:gravity="center"
22       android:paddingTop="50dp"
23       android:orientation="horizontal">
24       <LinearLayout
25         android:layout_width="wrap_content"
26         android:layout_height="wrap_content"
27         android:gravity="center"
28         android:orientation="vertical">
29           <ImageView
30             android:id="@+id/json"
31             android:layout_width="120dp"
32             android:layout_height="120dp"
33             android:layout_gravity="center"
34             android:layout_margin="16dp"
35             android:layout_weight="1"
36             android:src="@drawable/ic_baseline_looks_one_24" />
37
38           <LinearLayout
39             android:layout_width="wrap_content"
40             android:layout_height="wrap_content"
41             android:gravity="center"
42             android:orientation="vertical">
43             <ImageView
44               android:id="@+id/json_api"
45               android:layout_width="120dp"
46               android:layout_height="120dp"
47               android:layout_gravity="center"
48               android:layout_margin="16dp"
49               android:layout_weight="1"
50               android:src="@drawable/ic_baseline_looks_two_24" />
51           </LinearLayout>
52
53           <LinearLayout
54             android:layout_width="match_parent"
55             android:layout_height="wrap_content"
```

```
55     android:gravity="center"
56     android:paddingTop="50dp"
57     android:orientation="horizontal">
58     <LinearLayout
59         android:layout_width="wrap_content"
60         android:layout_height="wrap_content"
61         android:gravity="center"
62         android:orientation="vertical">
63         <ImageView
64             android:id="@+id/moviedb"
65             android:layout_width="120dp"
66             android:layout_height="120dp"
67             android:layout_gravity="center"
68             android:layout_margin="16dp"
69             android:layout_weight="1"
70             android:src="@drawable/ic_baseline_looks_3_24" />
71     </LinearLayout>
72     </LinearLayout>
73 </LinearLayout>
74
```

6. Menambahkan **color.xml** seperti berikut

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <resources>
3      <color name="colorPrimary">#212d3b</color>
4      <color name="colorPrimaryDark">#161d27</color>
5      <color name="colorRating">#FBC02D</color>
6      <color name="colorAccent">#52a2f7</color>
7      <color name="colorDark">#414449</color>
```

7. Mengubah file **MainActivity.java** seperti berikut

```
3  import androidx.appcompat.app.AppCompatActivity;
4
5  import android.content.Intent;
6  import android.os.Bundle;
7  import android.view.View;
8  import android.widget.ImageView;
9
10 public class MainActivity extends AppCompatActivity {
11
12     @Override
13     protected void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView(R.layout.activity_main);
16         ImageView showJson = (ImageView) findViewById(R.id.json);
17         ImageView showJsonAPI = (ImageView) findViewById(R.id.json_api);
18         ImageView showMovieDB = (ImageView) findViewById(R.id.moviedb);
```

```

19
20     showJson.setOnClickListener(new View.OnClickListener() {
21         @Override
22         public void onClick(View view) {
23             Intent intent = new Intent( mContext: MainActivity.this, JsonParseActivity.class);
24             startActivity(intent);
25         }
26     });
27     showJsonAPI.setOnClickListener(new View.OnClickListener() {
28         @Override
29         public void onClick(View view) {
30             Intent intent = new Intent( mContext: MainActivity.this, JsonAPIActivity.class);
31             startActivity(intent);
32         }
33     });
34     showMovieDB.setOnClickListener(new View.OnClickListener() {
35         @Override
36         public void onClick(View view) {
37             Intent intent = new Intent( mContext: MainActivity.this, MovieDBActivity.class);
38             startActivity(intent);
39         }
40     });
41 }
42 }

```

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan implementasi code pengenalan JSON.
2. Dokumentasi hasil output aplikasi dalam bentuk laporan dilengkapi dengan analisa code pada implementasi source code.

g. Kesimpulan

Mahasiswa memahami konsep JSON dan menerapkan JSON pada android.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara detail setiap baris code JSON	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara singkat setiap baris code JSON	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam implementasi code serta menjelaskan setiap baris code JSON	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam implementasi code serta menjelaskan setiap baris code JSON	(1)	2	3	4
Jumlah skor					

ACARA 38 : PENGENALAN JSON

Pokok Bahasan	: Pengambilan API
Acara Praktikum/Pertemuan	: Minggu 11 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep JSON
2. Mahasiswa mampu menerapkan JSON Parse dan JSON API
3. Mahasiswa mampu menguji url API melalui postman

b. Indikator

Keberhasilan mahasiswa menerapkan JSON Parse dan JSON API serta menguji url API melalui postman.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

e. Prosedur Kerja

JSON PARSE

1. Buka kembali proyek JsonMovieDB yang dibuat di bagian sebelumnya
2. Menambahkan file **json_parse.xml** dengan sintak seperti berikut

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="fill_parent"
4      android:layout_height="fill_parent"
5      android:orientation="vertical" >
6      <ListView
7          android:id="@+id/user_list"
8          android:layout_width="fill_parent"
9          android:layout_height="wrap_content"
10         android:dividerHeight="1dp" />
11     </LinearLayout>
```

3. Menambahkan file **list_row.xml** dengan sintak seperti berikut.

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="fill_parent"
4      android:layout_height="wrap_content"
5      android:orientation="horizontal"
6      android:padding="5dip" >
7      <TextView
8          android:id="@+id/name"
9          android:layout_width="wrap_content"
10         android:layout_height="wrap_content"
11         android:textStyle="bold"
12         android:textSize="17dp" />
13      <TextView
14          android:id="@+id/designation"
15          android:layout_width="wrap_content"
16          android:layout_height="wrap_content"
17          android:layout_below="@+id/name"
18          android:layout_marginTop="7dp"
19          android:textColor="#343434"
20          android:textSize="14dp" />
21      <TextView
22          android:id="@+id/location"
23          android:layout_width="wrap_content"
24          android:layout_height="wrap_content"
25          android:layout_alignBaseline="@+id/designation"
26          android:layout_alignBottom="@+id/designation"
27          android:layout_alignParentRight="true"
28          android:textColor="#343434"
29          android:textSize="14dp" />
30    </RelativeLayout>
31
```

4. Menambahkan file **JsonParseActivity.java** dengan sintak seperti berikut

```
3  import android.os.Bundle;
4  import android.util.Log;
5  import android.widget.ListAdapter;
6  import android.widget.ListView;
7  import android.widget.SimpleAdapter;
8  import androidx.appcompat.app.AppCompatActivity;
9  import org.json.JSONArray;
10 import org.json.JSONException;
11 import org.json.JSONObject;
12 import java.util.ArrayList;
13 import java.util.HashMap;
14
15 public class JsonParseActivity extends AppCompatActivity {
16
```

```

17     @Override
18     protected void onCreate(Bundle savedInstanceState) {
19         super.onCreate(savedInstanceState);
20         setContentView(R.layout.json_parse);
21         getActionBar().setTitle("JSON");
22         String jsonStr = getListData();
23         try {
24             ArrayList<HashMap<String, String>> userList = new ArrayList<>();
25             ListView lv = (ListView) findViewById(R.id.user_list);
26             JSONObject jObj = new JSONObject(jsonStr);
27             JSONArray jsonArry = jObj.getJSONArray( name: "users" );
28             for (int i = 0; i < jsonArry.length(); i++) {
29                 HashMap<String, String> user = new HashMap<>();
30                 JSONObject obj = jsonArry.getJSONObject(i);
31                 user.put("name", obj.getString( name: "name"));
32                 user.put("designation", obj.getString( name: "designation"));
33                 user.put("location", obj.getString( name: "location"));
34                 userList.add(user);
35             }
36             ListAdapter adapter = new SimpleAdapter( context: JsonParseActivity.this, userList,
37                     R.layout.list_row, new String[]{ "name", "designation", "location" },
38                     new int[]{ R.id.name, R.id.designation, R.id.location });
39             lv.setAdapter(adapter);
40         } catch (JSONException ex) {
41             Log.e( tag: "JsonParser Example", msg: "unexpected JSON exception", ex );
42         }
43     }
44
45     @
46     private String getListData() {
47         String jsonStr = "{ \"users\":[" +
48             "{\"name\":\"Monkey D. Luffy\", \"designation\":\"Leader\", \"location\":\"East Blue\"} " +
49             ",{\"name\":\"Roronoa Zoro\", \"designation\":\"Vice Leader\", \"location\":\"East Blue\"} " +
50             ",{\"name\":\"Nami\", \"designation\":\"Navigator\", \"location\":\"East Blue\"}] } ";
51         return jsonStr;
52     }
53 }
```

JSON API

- Menambahkan file **json_api.xml** dengan sintak seperti berikut

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="fill_parent"
4      android:layout_height="fill_parent"
5      android:orientation="vertical" >
6      <ListView
7          android:id="@+id/user_list"
8          android:layout_width="fill_parent"
9          android:layout_height="wrap_content"
10         android:dividerHeight="1dp" />
11  </LinearLayout>
```

2. Menambahkan file **list_item.xml** dengan sintak seperti berikut

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="fill_parent"
4      android:layout_height="wrap_content"
5      android:orientation="vertical">
6      <TextView
7          android:id="@+id/name"
8          android:layout_width="fill_parent"
9          android:layout_height="wrap_content"
10         android:paddingBottom="2dip"
11         android:paddingTop="6dip"
12         android:textColor="@color/colorPrimaryDark"
13         android:textSize="16sp"
14         android:textStyle="bold" />
15      <TextView
16          android:id="@+id/email"
17          android:layout_width="fill_parent"
18          android:layout_height="wrap_content"
19          android:paddingBottom="2dip"
20          android:textColor="@color/colorAccent" />
21      <TextView
22          android:id="@+id/mobile"
23          android:layout_width="wrap_content"
24          android:layout_height="wrap_content"
25          android:textColor="#5d5d5d"
26          android:textStyle="bold" />
27  </LinearLayout>
```

3. Menambahkan file **HttpHandle.java** dengan sintak seperti berikut

```
1  package com.example.myapplication;
2
3  import android.util.Log;
4  import java.io.BufferedReader;
5  import java.io.IOException;
6  import java.io.InputStream;
7  import java.io.InputStreamReader;
8  import java.net.HttpURLConnection;
9  import java.net.MalformedURLException;
10 import java.net.ProtocolException;
11 import java.net.URL;
12
13
14 public class HttpHandler {
15     private static final String TAG = HttpHandler.class.getSimpleName();
16     public HttpHandler() {
17     }
18
19     public String makeServiceCall(String reqUrl) {
20         String response = null;
```

```
21     try {
22         URL url = new URL(reqUrl);
23         HttpURLConnection conn = (HttpURLConnection) url.openConnection();
24         conn.setRequestMethod("GET");
25         // read the response
26         InputStream in = new BufferedInputStream(conn.getInputStream());
27         response = convertStreamToString(in);
28     } catch (MalformedURLException e) {
29         Log.e(TAG, msg: "MalformedURLException: " + e.getMessage());
30     } catch (ProtocolException e) {
31         Log.e(TAG, msg: "ProtocolException: " + e.getMessage());
32     } catch (IOException e) {
33         Log.e(TAG, msg: "IOException: " + e.getMessage());
34     } catch (Exception e) {
35         Log.e(TAG, msg: "Exception: " + e.getMessage());
36     }
37     return response;
38 }
39
40 @
41     private String convertStreamToString(InputStream is) {
42         BufferedReader reader = new BufferedReader(new InputStreamReader(is));
43         StringBuilder sb = new StringBuilder();
44         String line;
45         try {
46             while ((line = reader.readLine()) != null) {
47                 sb.append(line).append('\n');
48             }
49         } catch (IOException e) {
50             e.printStackTrace();
51         } finally {
52             try {
53                 is.close();
54             } catch (IOException e) {
55                 e.printStackTrace();
56             }
57         }
58         return sb.toString();
59     }
60 }
```

4. Menambahkan file **JsonAPIActivity.java** dengan sintak seperti berikut

```
3   import android.app.AlertDialog;
4   import android.os.AsyncTask;
5   import android.os.Bundle;
6   import android.util.Log;
7   import android.widget.AdapterView;
8   import android.widget.ListView;
9   import android.widget.SimpleAdapter;
10  import android.widget.Toast;
11  import androidx.appcompat.app.AppCompatActivity;
12  import org.json.JSONArray;
13  import org.json.JSONException;
14  import org.json.JSONObject;
15  import java.util.ArrayList;
16  import java.util.HashMap;
17
18  public class JsonAPIActivity extends AppCompatActivity {
19
20      private String TAG = MainActivity.class.getSimpleName();
21
22      private ProgressDialog pDialog;
23      private ListView lv;
24
25      // URL to get contacts JSON
26      private static String url = "https://api.androidhive.info/contacts/";
27
28      ArrayList<HashMap<String, String>> contactList;
29
30      @Override
31      protected void onCreate(Bundle savedInstanceState) {
32          super.onCreate(savedInstanceState);
33          setContentView(R.layout.json_api);
34
35          contactList = new ArrayList<>();
36
37          lv = (ListView) findViewById(R.id.listview);
38
39          new GetContacts().execute();
40      }
41
42      /**
43       * Async task class to get json by making HTTP call
44       */
45      private class GetContacts extends AsyncTask<Void, Void, Void> {
46
47          @Override
48          protected void onPreExecute() {
49              super.onPreExecute();
50              // Showing progress dialog
51              pDialog = new ProgressDialog(JsonAPIActivity.this);
52              pDialog.setMessage("Please wait...");
53              pDialog.setCancelable(false);
54              pDialog.show();
55
56          }
57
58          @Override
59          protected Void doInBackground(Void... arg0) {
```

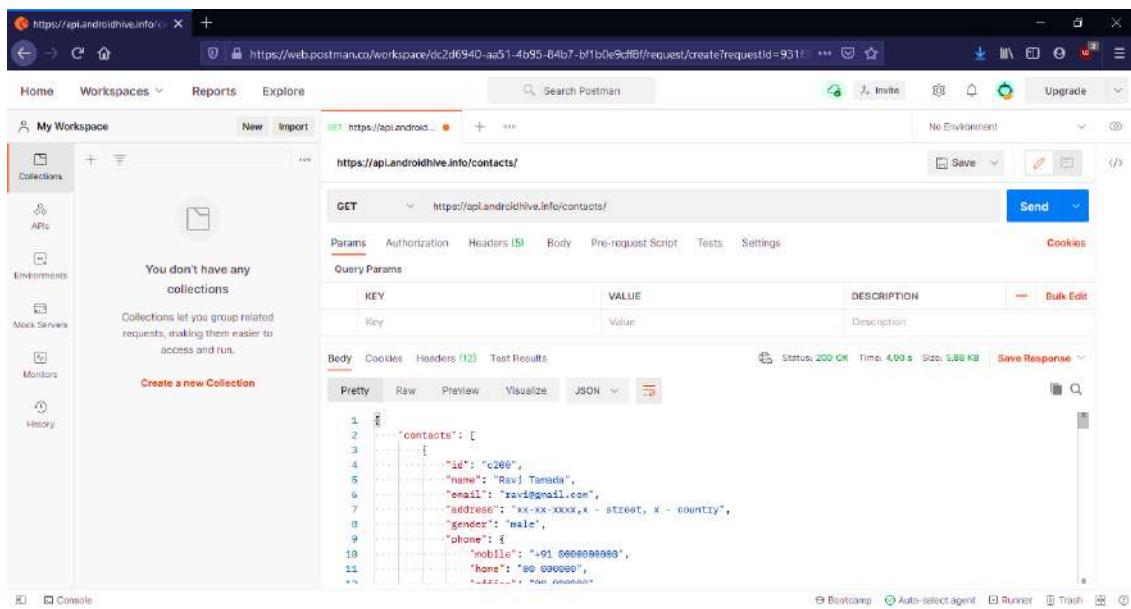
```

60          HttpHandler sh = new HttpHandler();
61
62          // Making a request to url and getting response
63          String jsonStr = sh.makeServiceCall(url);
64
65          Log.e(TAG, msg: "Response from url: " + jsonStr);
66
67          if (jsonStr != null) {
68              try {
69                  JSONObject jsonObj = new JSONObject(jsonStr);
70
71                  // Getting JSON Array node
72                  JSONArray contacts = jsonObj.getJSONArray( name: "contacts");
73
74                  // looping through All Contacts
75                  for (int i = 0; i < contacts.length(); i++) {
76                      JSONObject c = contacts.getJSONObject(i);
77
78                      String id = c.getString( name: "id");
79                      String name = c.getString( name: "name");
80                      String email = c.getString( name: "email");
81                      String address = c.getString( name: "address");
82                      String gender = c.getString( name: "gender");
83
84                      // Phone node is JSON Object
85                      JSONObject phone = c.getJSONObject("phone");
86                      String mobile = phone.getString( name: "mobile");
87                      String home = phone.getString( name: "home");
88                      String office = phone.getString( name: "office");
89
90                      // tmp hash map for single contact
91                      HashMap<String, String> contact = new HashMap<>();
92
93                      // adding each child node to HashMap key => value
94                      contact.put("id", id);
95                      contact.put("name", name);
96                      contact.put("email", email);
97                      contact.put("mobile", mobile);
98
99                      // adding contact to contact list
100                     contactList.add(contact);
101                 }
102             } catch (final JSONException e) {
103                 Log.e(TAG, msg: "Json parsing error: " + e.getMessage());
104                 runOnUiThread(new Runnable() {
105                     @Override
106                     public void run() {
107                         Toast.makeText(getApplicationContext(),
108                             text: "Json parsing error: " + e.getMessage(),
109                             Toast.LENGTH_LONG)
110                             .show();
111                     }
112                 });
113             }

```

```
114             }
115         } else {
116             Log.e(TAG, "Couldn't get json from server.");
117             runOnUiThread(new Runnable() {
118                 @Override
119                 public void run() {
120                     Toast.makeText(getApplicationContext(),
121                         Toast.LENGTH_LONG)
122                         .show();
123                 }
124             });
125         }
126     }
127     return null;
128 }
129
130
131     @Override
132     protected void onPostExecute(Void result) {
133         super.onPostExecute(result);
134         // Dismiss the progress dialog
135         if (pDialog.isShowing())
136             pDialog.dismiss();
137         /**
138          * Updating parsed JSON data into ListView
139         */
140        ListAdapter adapter = new SimpleAdapter(
141             context JsonAPIActivity.this, contactList,
142             R.layout.list_item, new String[]{"name", "email",
143             "mobile"}, new int[]{R.id.name,
144             R.id.email, R.id.mobile});
145         lv.setAdapter(adapter);
146     }
147 }
148 }
```

5. Menguji url API melalui postman <https://web.postman.co/home>
<https://api.androidhive.info/contacts/>



f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan implementasi code JSON Parse dan JSON API, dengan berhasil menguji url API melalui postman.
2. Dokumentasi hasil output aplikasi dalam bentuk laporan dilengkapi dengan analisa code pada implementasi source code.
3. Dokumentasi hasil pengujian url API melalui postman.

g. Kesimpulan

Mahasiswa menerapkan JSON Parse dan JSON API serta menguji url API melalui postman.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara detail setiap baris code, dan pengujian melalui postman	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara singkat setiap baris code, dan pengujian melalui postman	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam implementasi code serta menjelaskan setiap baris code, dan pengujian melalui postman	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam implementasi code serta menjelaskan setiap baris code, dan pengujian melalui postman	(1)	2	3	4
Jumlah skor					

ACARA 39 : PENGENALAN JSON

Pokok Bahasan	: Implementasi JSON API MovieDB
Acara Praktikum/Pertemuan	: Minggu 11 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep MovieDB.
2. Mahasiswa mampu menerapkan JSON API movieDB pada android.

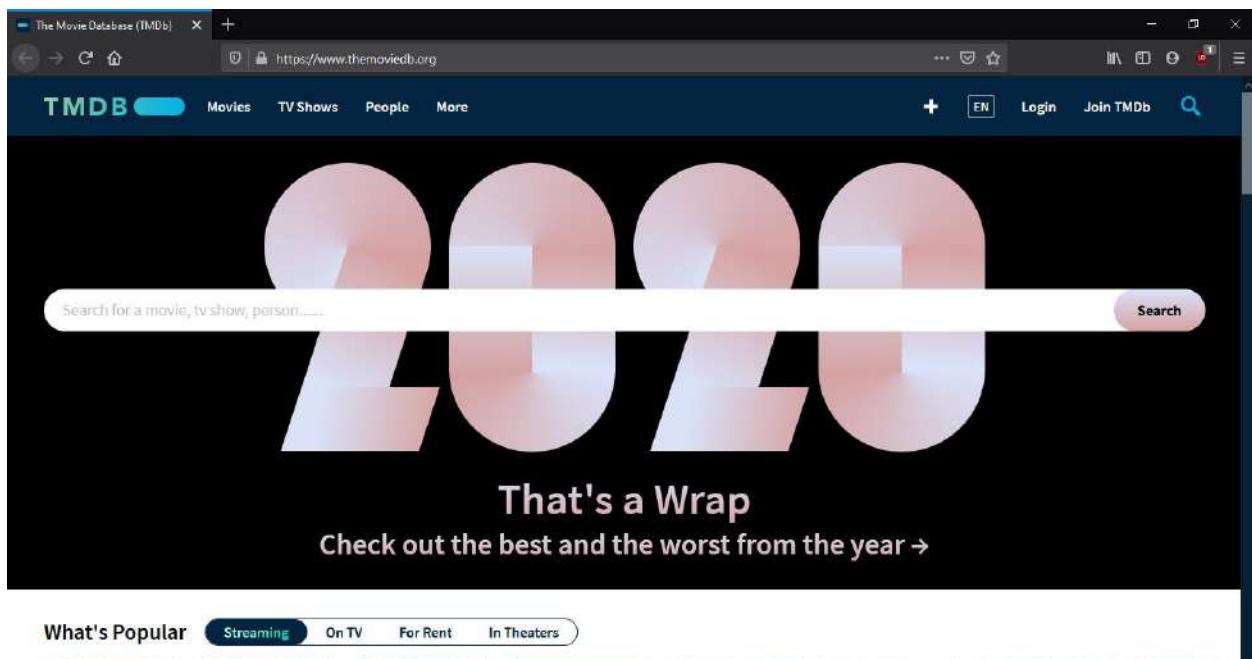
b. Indikator

Keberhasilan mahasiswa menerapkan JSON API movieDB pada android.

c. Dasar Teori

MovieDB

Sebuah website bernama The Moview DB, website penyedia informasi tentang film dan juga menyediakan API secara gratis dengan syarat harus register terlebih duluh.



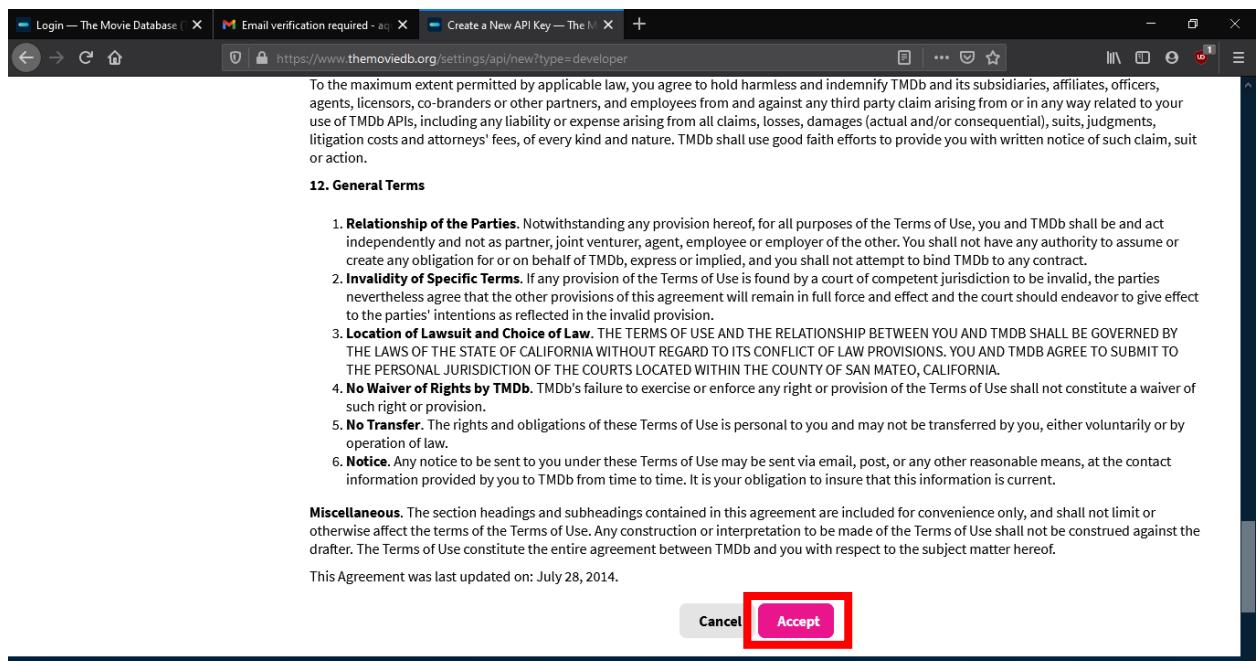
Untuk mendapatkan API, masuk ke website tersebut, lalu pilih Sign Up. Selanjutnya masuk ke akun, Untuk mengakses API-nya diperlukan API key, terlebih dahulu harus buat akun. Setelah itu

akses ke menu settings di profile, kemudian generate API key di menu API, jika sudah maka akan tampil data seperti berikut.

The image consists of two screenshots of the Movie Database (TMDB) website. The top screenshot shows the user's profile page. At the top right, there is a sidebar menu with options: 'View profile', 'Discussions', 'Lists', 'Ratings', 'Watchlist', 'Edit Profile', 'Logout', and 'Settings'. The 'Settings' button is highlighted with a red box. Below the sidebar, there is a 'Stats' section showing 'Total Edits' and 'Total Ratings', both at 0. To the right, there is a 'Rating Overview' chart and a note stating 'Most Watched Genres' with the message 'You haven't logged any movies or TV shows.' The bottom screenshot shows the 'Account Settings' page under the 'Settings' section. On the left, there is a sidebar with options: 'Edit Profile', 'Account Settings' (which is selected and highlighted with a red box), 'Streaming Services', 'Notification Settings', 'Blocked Users', 'Import List', 'Sharing Settings', 'Social Settings', 'Sessions', and 'API'. The 'API' option is also highlighted with a red box. The main content area shows various account settings like 'Default Language' (English (en-US)), 'Fallback Language' (None (Don't Fallback)), 'Country' (Indonesia), 'Timezone - Auto detect?' (checked, Asia - Bangkok), 'Include Adult Items in Search?' (No), and 'Filter Profanity?' (Yes). A 'Save' button is located at the bottom of this section.

The screenshot shows the TMDB API Settings page. On the left, there's a sidebar with a pink header labeled 'Settings' containing links like 'Edit Profile', 'Account Settings', and 'Request an API Key'. The main content area has tabs for 'API', 'Overview', and 'Create'. Under the 'API' tab, there's a section titled 'Documentation' with a link to developers.themoviedb.org, and a 'Support' section with a link to support forums. A prominent red box highlights the 'Request an API Key' button, which is described as a link to generate a new API key.

This screenshot shows the 'Create a New API Key' page. It features a sidebar with the same 'Settings' menu as the previous page. The main area asks 'What type of API key do you wish to register?' and lists two options: 'Developer' and 'Professional'. The 'Developer' option is highlighted with a red box and includes a bulleted list: 'You are an individual', 'Your project is still in development', 'Your project is non profit', and 'Your project is ad supported'. The 'Professional' option is also listed with its own bullet points: 'You represent a company', 'Your project is for profit (not ad supported)', and 'You are an OEM or hardware vendor'.



API Overview Create

Type of Use
Mobile Application

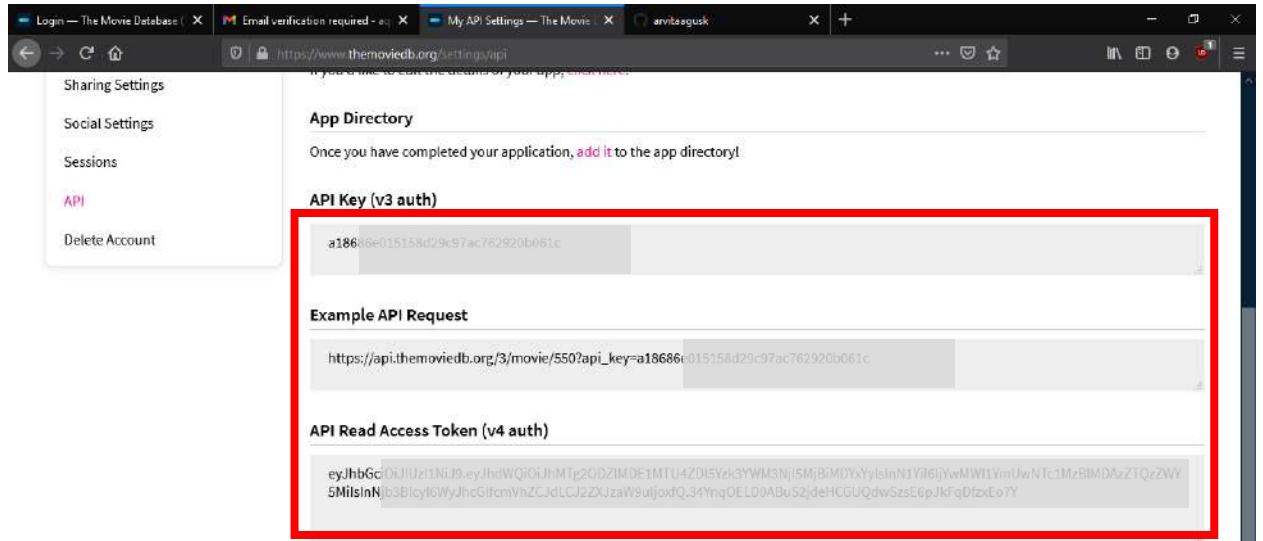
Application Name
MovieDB

Application URL

Application Summary

First Name Last Name

*Application URL dapat menggunakan url github project



d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

e. Prosedur Kerja

JSON API MovieDB

1. Buka kembali proyek JsonMovieDB yang dibuat di bagian sebelumnya
2. Menambahkan file **moviedb.xml** dengan sintak seperti berikut

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      android:background="@color/colorPrimaryDark"
7      android:fitsSystemWindows="true">
8      <com.github.florent37.diagonallayout.Diagonallayout
9          android:id="@+id/diagonallayout"
10         android:layout_width="match_parent"
11         android:layout_height="250dp"
12         app:diagonal_angle="10"
13         app:diagonal_direction="left"
14         app:diagonal_position="bottom">
15             <LinearLayout
16                 android:layout_width="match_parent"
17                 android:layout_height="match_parent"
18                 android:background="@drawable/design_header"
19                 android:orientation="vertical"
20                 android:scaleType="fitXY" />

```

```
21     </com.github.florent37.diagonallayout.DiagonalLayout>
22     <include layout="@layout/toolbar_main" />
23     <androidx.coordinatorlayout.widget.CoordinatorLayout
24         android:layout_width="match_parent"
25         android:layout_height="match_parent"
26         android:layout_below="@+id/toolbar">
27         <FrameLayout
28             android:id="@+id/frameLayout"
29             android:layout_width="match_parent"
30             android:layout_height="match_parent" />
31     </androidx.coordinatorlayout.widget.CoordinatorLayout>
32 </RelativeLayout>
```

3. Menambahkan file toolbar_main.xml dengan sintak seperti berikut

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.appcompat.widget.Toolbar xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      android:id="@+id/toolbar"
5      android:layout_width="match_parent"
6      android:layout_height="?actionBarSize">
7      <RelativeLayout
8          android:layout_width="match_parent"
9          android:layout_height="wrap_content"
10         android:orientation="horizontal">
11         <TextView
12             android:layout_width="wrap_content"
13             android:layout_height="wrap_content"
14             android:layout_centerVertical="true"
15             android:text="Hello, 'Arvita' Welcome to our Application"
16             android:textColor="@android:color/white"
17             android:textSize="16sp" />
18     </RelativeLayout>
19 </androidx.appcompat.widget.Toolbar>
```

4. Menambahkan file fragmen_film.xml dengan sintak seperti berikut

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.core.widget.NestedScrollView xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent">
6      <LinearLayout
7          android:layout_width="match_parent"
8          android:layout_height="match_parent"
9          android:layout_marginTop="10dp"
10         android:orientation="vertical">
```

```
11     <TextView  
12         android:layout_width="wrap_content"  
13         android:layout_height="wrap_content"  
14         android:layout_gravity="start|center"  
15         android:layout_marginStart="5dp"  
16         android:layout_marginTop="10dp"  
17         android:layout_marginBottom="5dp"  
18         android:text="Rekomendasi Film"  
19         android:textColor="@android:color/white"  
20         android:textSize="16sp"  
21         android:textStyle="bold"  
22         android:layout_marginLeft="5dp" />  
23     <androidx.recyclerview.widget.RecyclerView  
24         android:id="@+id/rvFilmRecommend"  
25         android:layout_width="match_parent"  
26         android:layout_height="match_parent"  
27         tools:listitem="@layout/list_item_film" />  
28     </LinearLayout>  
29 </androidx.core.widget.NestedScrollView>
```

5. Menambahkan file list_item_film.xml dengan sintak seperti berikut

```
1     <?xml version="1.0" encoding="utf-8"?>  
2     <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
3         xmlns:app="http://schemas.android.com/apk/res-auto"  
4         android:layout_width="match_parent"  
5         android:layout_height="wrap_content">  
6         <FrameLayout  
7             android:layout_width="match_parent"  
8             android:layout_height="wrap_content">  
9             <RelativeLayout  
10                android:layout_width="match_parent"  
11                android:layout_height="match_parent"  
12                android:layout_gravity="bottom"  
13                android:orientation="vertical"  
14                android:paddingStart="5dp"  
15                android:paddingEnd="5dp">  
16                <androidx.cardview.widget.CardView  
17                    android:id="@+id/cvFilm"  
18                    android:layout_width="match_parent"  
19                    android:layout_height="wrap_content"  
20                    android:layout_margin="5dp"  
21                    app:cardBackgroundColor="@color/colorPrimary"  
22                    app:cardCornerRadius="10dp"  
23                    app:cardElevation="3dp">  
24                    <RelativeLayout  
25                        android:layout_width="match_parent"  
26                        android:layout_height="120dp"  
27                        android:layout_marginStart="105dp"
```

```
28         android:layout_marginLeft="105dp">
29     <LinearLayout
30         android:layout_width="match_parent"
31         android:layout_height="wrap_content"
32         android:layout_centerInParent="true"
33         android:orientation="vertical">
34         <TextView
35             android:id="@+id/tvTitle"
36             android:layout_width="wrap_content"
37             android:layout_height="wrap_content"
38             android:layout_marginBottom="5dp"
39             android:textColor="@android:color/white"
40             android:textSize="18sp" />
41         <TextView
42             android:id="@+id/tvDesc"
43             android:layout_width="wrap_content"
44             android:layout_height="wrap_content"
45             android:ellipsize="end"
46             android:maxLines="2"
47             android:textAlignment="viewStart"
48             android:textColor="@android:color/white"
49             android:textSize="12sp"
50             android:gravity="start" />
51         <RatingBar
52             android:id="@+id/ratingBar"
53             style="@style/Widget.AppCompat.RatingBar.Small"
54             android:layout_width="wrap_content"
55             android:layout_height="wrap_content"
56             android:progressTint="@color/colorRating" />
57     </LinearLayout>
58 </RelativeLayout>
59     <TextView
60         android:id="@+id/tvRealeseDate"
61         android:layout_width="wrap_content"
62         android:layout_height="wrap_content"
63         android:layout_gravity="bottom|end"
64         android:layout_marginEnd="10dp"
65         android:layout_marginBottom="10dp"
66         android:background="@drawable/bg_rounded"
67         android:backgroundTint="@color/colorAccent"
68         android:padding="5dp"
69         android:textColor="@android:color/white"
70         android:textSize="8sp"
71         android:textStyle="bold"
72         android:layout_marginRight="10dp" />
73 </androidx.cardview.widget.CardView>
```

```
74        </RelativeLayout>
75        <LinearLayout
76            android:layout_width="wrap_content"
77            android:layout_height="wrap_content"
78            android:layout_gravity="start|center"
79            android:layout_marginStart="20dp"
80            android:layout_marginBottom="10dp"
81            android:gravity="center"
82            android:layout_marginLeft="20dp">
83            <ImageView
84                android:id="@+id/imgPhoto"
85                android:layout_width="90dp"
86                android:layout_height="140dp"
87                android:layout_marginTop="4dp"
88                android:layout_marginEnd="10dp"
89                android:scaleType="fitCenter"
90                android:layout_marginRight="10dp" />
91        </LinearLayout>
92    </FrameLayout>
93</RelativeLayout>
```

6. Menambahkan file MovieDBAcitivity.java dengan sintak seperti berikut :

```
1  package com.example.myapplication;
2
3  import android.app.Activity;
4  import android.os.Build;
5  import android.os.Bundle;
6  import android.view.View;
7  import android.view.Window;
8  import android.view.WindowManager;
9  import androidx.appcompat.app.AppCompatActivity;
10
11 public class MovieDBActivity extends AppCompatActivity {
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.movie_db);
17
18         if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
19             getWindow().getDecorView().setSystemUiVisibility(View.SYSTEM_UI_FLAG_LAYOUT_STABLE |
20                     View.SYSTEM_UI_FLAG_LAYOUT_FULLSCREEN);
21         }
22         if (Build.VERSION.SDK_INT >= 21) {
23             setWindowFlag(activity: this, WindowManager.LayoutParams.FLAG_TRANSLUCENT_STATUS, on: false);
24             getWindow().setStatusBarColor(getResources().getColor(R.color.colorPrimary));
25         }
26         getSupportFragmentManager().beginTransaction().replace(R.id.frameLayout, new FragmentMovie()).commit();
27     }
28
29     @
30     public static void setWindowFlag(Activity activity, final int bits, boolean on) {
31         Window window = activity.getWindow();
```

```
51     WindowManager.LayoutParams winParams = window.getAttributes();
52     if (on) {
53         winParams.flags |= bits;
54     } else {
55         winParams.flags &= ~bits;
56     }
57     window.setAttributes(winParams);
58 }
59 }
```

7. Menambahkan file MovieAdapter.java dengan sintak seperti berikut

```
3   import android.content.Context;
4   import android.view.LayoutInflater;
5   import android.view.View;
6   import android.view.ViewGroup;
7   import android.widget.ImageView;
8   import android.widget.RatingBar;
9   import android.widget.TextView;
10  import androidx.cardview.widget.CardView;
11  import androidx.recyclerview.widget.RecyclerView;
12  import com.bumptech.glide.Glide;
13  import com.bumptech.glide.load.resource.bitmap.RoundedCorners;
14  import com.bumptech.glide.request.RequestOptions;
15
16  import java.util.List;
17
18  public class MovieAdapter extends RecyclerView.Adapter<MovieAdapter.ViewHolder> {
19
20      private List<ModelMovie> items;
21      private MovieAdapter.onSelectData onSelectData;
22      private Context mContext;
23      private double Rating;
24
25      public interface onSelectData {
26          void onSelected(ModelMovie modelMovie);
27      }
28
29      public MovieAdapter(Context context, List<ModelMovie> items, MovieAdapter.onSelectData xSelectData) {
30          this.mContext = context;
31          this.items = items;
32          this.onSelectData = xSelectData;
33      }
34
35      @Override
36      public MovieAdapter.ViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
37          View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.list_item_film, parent, false);
38          return new MovieAdapter.ViewHolder(view);
39      }
40
41      @Override
42      public void onBindViewHolder(MovieAdapter.ViewHolder holder, int position) {
43          final ModelMovie data = items.get(position);
44      }
45  }
```

```

45     Rating = data.getVoteAverage();
46     holder.tvTitle.setText(data.getTitle());
47     holder.tvRealeseDate.setText(data.getReleaseDate());
48     holder.tvDesc.setText(data.getOverview());
49
50     float newValue = (float)Rating;
51     holder.ratingBar.setNumStars(5);
52     holder.ratingBar.setStepSize((float) 0.5);
53     holder.ratingBar.setRating(newValue / 2);
54
55     Glide.with(mContext).RequestManager
56         .load(string: ApiEndpoint.URLIMAGE + data.getPosterPath()) RequestBuilder<Drawable>
57         .apply(new RequestOptions()
58             .placeholder(R.drawable.ic_image)
59             .transform(new RoundedCorners( roundingRadius: 16)))
60         .into(holder.imgPhoto);
61
62     holder.cvFilm.setOnClickListener(new View.OnClickListener() {
63         @Override
64         public void onClick(View v) { onSelectData.onSelected(data); }
65     });
66
67
68     @Override
69     public int getItemCount() { return items.size(); }
70
71     //Class Holder
72     class ViewHolder extends RecyclerView.ViewHolder {
73         public CardView cvFilm;
74         public ImageView imgPhoto;
75         public TextView tvTitle;
76         public TextView tvRealeseDate;
77         public TextView tvDesc;
78         public RatingBar ratingBar;
79         public ViewHolder(View itemView) {
80             super(itemView);
81             cvFilm = itemView.findViewById(R.id.cvFilm);
82             imgPhoto = itemView.findViewById(R.id.imgPhoto);
83             tvTitle = itemView.findViewById(R.id.tvTitle);
84             tvRealeseDate = itemView.findViewById(R.id.tvRealeseDate);
85             tvDesc = itemView.findViewById(R.id.tvDesc);
86             ratingBar = itemView.findViewById(R.id.ratingBar);
87         }
88     }
89 }

```

8. Menambahkan file ModelMovie.java dengan sintak seperti berikut

```
1 package com.example.myapplication;
2
3 import java.io.Serializable;
4
5 public class ModelMovie implements Serializable {
6
7     private int Id;
8     private String Title;
9     private double VoteAverage;
10    private String Overview;
11    private String ReleaseDate;
12    private String PosterPath;
13    private String BackdropPath;
14    private String Popularity;
15
16    public ModelMovie() {
17    }
18
19    public int getId() { return Id; }
20
21    public void setId(int id) { Id = id; }
22
23    public String getTitle() { return Title; }
24
25    public void setTitle(String title) { Title = title; }
26
27    public double getVoteAverage() { return VoteAverage; }
28
29    public void setVoteAverage(double voteAverage) { VoteAverage = voteAverage; }
30
31    public String getOverview() { return Overview; }
32
33    public void setOverview(String overview) { Overview = overview; }
34
35    public String getReleaseDate() { return ReleaseDate; }
36
37    public void setReleaseDate(String releaseDate) { ReleaseDate = releaseDate; }
38
39    public String getPosterPath() { return PosterPath; }
40
41    public void setPosterPath(String posterPath) { PosterPath = posterPath; }
42
43    public String getBackdropPath() { return BackdropPath; }
44
45    public void setBackdropPath(String backdropPath) { BackdropPath = backdropPath; }
46
47    public String getPopularity() { return Popularity; }
48
49    public void setPopularity(String popularity) { Popularity = popularity; }
50
51 }
```

9. Menambahkan file FragmentMovie.java dengan sintak seperti berikut

```
3   import android.app.AlertDialog;
4   import android.os.Bundle;
5   import android.view.LayoutInflater;
6   import android.view.View;
7   import android.view.ViewGroup;
8   import android.widget.Toast;
9   import androidx.fragment.app.Fragment;
10  import androidx.recyclerview.widget.LinearLayoutManager;
11  import androidx.recyclerview.widget.RecyclerView;
12  import com.androidnetworking.AndroidNetworking;
13  import com.androidnetworking.common.Priority;
14  import com.androidnetworking.error.ANError;
15  import com.androidnetworking.interfaces.JSONObjectRequestListener;
16  import org.json.JSONArray;
17  import org.json.JSONException;
18  import org.json.JSONObject;
19  import java.text.ParseException;
20  import java.text.SimpleDateFormat;
21  import java.util.ArrayList;
22  import java.util.List;
23
24  public class FragmentMovie extends Fragment implements MovieAdapter.onSelectData {
25
26      private RecyclerView rvFilmRecommend;
27      private MovieAdapter movieAdapter;
28      private ProgressDialog progressDialog;
29      private List<ModelMovie> moviePopular = new ArrayList<>();
30
31      public FragmentMovie() {
32      }
33
34      @Override
35      @Override
36      public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {
37          View rootView = inflater.inflate(R.layout.fragment_film, container, attachToRoot: false);
38
39          progressDialog = new ProgressDialog(getActivity());
40          progressDialog.setTitle("Mohon Tunggu");
41          progressDialog.setCancelable(false);
42          progressDialog.setMessage("Sedang menampilkan data");
43
44          rvFilmRecommend = rootView.findViewById(R.id.rvFilmRecommend);
45          rvFilmRecommend.setLayoutManager(new LinearLayoutManager(getActivity()));
46          rvFilmRecommend.setHasFixedSize(true);
47
48          getMovieHorizontal();
49          getMovie();
50
51          return rootView;
52      }
53
54      private void setSearchMovie(String query) {
55          progressDialog.show();
56      }
57  }
```

```

55         AndroidNetworking.get(ApiEndpoint.BASEURL + ApiEndpoint.SEARCH_MOVIE
56             + ApiEndpoint.APIKEY + ApiEndpoint.LANGUAGE + ApiEndpoint.QUERY + query)
57             .setPriority(Priority.HIGH)
58             .build()
59             .getAsJSONObject(new JSONObjectRequestListener() {
60                 @Override
61                 e| public void onResponse(JSONObject response) {
62                     try {
63                         progressDialog.dismiss();
64                         moviePopular = new ArrayList<>();
65                         JSONArray jsonArray = response.getJSONArray( name: "results");
66                         for (int i = 0; i < jsonArray.length(); i++) {
67                             JSONObject jsonObject = jsonArray.getJSONObject(i);
68                             ModelMovie dataApi = new ModelMovie();
69                             SimpleDateFormat formatter = new SimpleDateFormat( pattern: "EEE, d MMM yyyy");
70                             SimpleDateFormat dateFormat = new SimpleDateFormat( pattern: "yyyy-mm-dd");
71                             String datePost = jsonObject.getString( name: "release_date");
72
73                             dataApi.setId(jsonObject.getInt( name: "id"));
74                             dataApi.setTitle(jsonObject.getString( name: "title"));
75                             dataApi.setVoteAverage(jsonObject.getDouble( name: "vote_average"));
76                             dataApi.setOverview(jsonObject.getString( name: "overview"));
77                             dataApi.setReleaseDate(formatter.format(dateFormat.parse(datePost)));
78                             dataApi.setPosterPath(jsonObject.getString( name: "poster_path"));
79                             dataApi.setBackdropPath(jsonObject.getString( name: "backdrop_path"));
80                             dataApi.setPopularity(jsonObject.getString( name: "popularity"));
81                             moviePopular.add(dataApi);
82                             showMovie();
83                         }
84                     } catch (JSONException | ParseException e) {
85                         e.printStackTrace();
86                         Toast.makeText(getApplicationContext(), text "Gagal menampilkan data!", Toast.LENGTH_SHORT).show();
87                     }
88                 }
89             }
90             @Override
91             e| public void onError(ANError anError) {
92                 progressDialog.dismiss();
93                 Toast.makeText(getApplicationContext(), text "Tidak ada jaringan internet!", Toast.LENGTH_SHORT).show();
94             }
95         });
96     }
97
98     private void getMovieHorizontal() {
99         progressDialog.show();
100        AndroidNetworking.get(ApiEndpoint.BASEURL + ApiEndpoint.MOVIE_PLAYNOW + ApiEndpoint.APIKEY + ApiEndpoint.LANGUAGE)
101            .setPriority(Priority.HIGH)
102            .build()
103            .getAsJSONObject(new JSONObjectRequestListener() {
104                @Override
105                e| public void onResponse(JSONObject response) {
106                    try {
107                        progressDialog.dismiss();
108                        JSONArray jsonArray = response.getJSONArray( name: "results");
109                        for (int i = 0; i < jsonArray.length(); i++) {
110                            JSONObject jsonObject = jsonArray.getJSONObject(i);
111                            ModelMovie dataApi = new ModelMovie();
112                            SimpleDateFormat formatter = new SimpleDateFormat( pattern: "EEE, d MMM yyyy");
113                            SimpleDateFormat dateFormat = new SimpleDateFormat( pattern: "yyyy-mm-dd");
114                            String datePost = jsonObject.getString( name: "release_date");
115
116                            dataApi.setId(jsonObject.getInt( name: "id"));
117                            dataApi.setTitle(jsonObject.getString( name: "title"));
118                            dataApi.setVoteAverage(jsonObject.getDouble( name: "vote_average"));
119                            dataApi.setOverview(jsonObject.getString( name: "overview"));

```

```

128                     dataApi.setReleaseDate(formatter.format(dateFormat.parse(datePost)));
129                     dataApi.setPosterPath(jsonObject.getString( name: "poster_path"));
130                     dataApi.setBackdropPath(jsonObject.getString( name: "backdrop_path"));
131                     dataApi.setPopularity(jsonObject.getString( name: "popularity"));
132                 }
133             } catch (JSONException | ParseException e) {
134                 e.printStackTrace();
135                 Toast.makeText(getApplicationContext(), text: "Gagal menampilkan data!", Toast.LENGTH_SHORT).show();
136             }
137         }
138     }
139     private void getMovie() {
140         progressDialog.show();
141         AndroidNetworking.get(ApiEndpoint.BASEURL + ApiEndpoint.MOVIE_POPULAR + ApiEndpoint.APIKEY + ApiEndpoint.LANGUAGE)
142             .setPriority(Priority.HIGH)
143             .build()
144             .getAsJSONObject(new JSONObjectRequestListener() {
145                 @Override
146                 public void onResponse(JSONObject response) {
147                     try {
148                         progressDialog.dismiss();
149                         moviePopular = new ArrayList<>();
150                         JSONArray jsonArray = response.getJSONArray( name: "results");
151                         for (int i = 0; i < jsonArray.length(); i++) {
152                             JSONObject jsonObject = jsonArray.getJSONObject(i);
153                             ModelMovie dataApi = new ModelMovie();
154                             SimpleDateFormat formatter = new SimpleDateFormat( pattern: "EEE, d MMM yyyy");
155                             SimpleDateFormat dateFormat = new SimpleDateFormat( pattern: "yyyy-mm-dd");
156                             String datePost = jsonObject.getString( name: "release_date");
157                             dataApi.setId(jsonObject.getInt( name: "id"));
158                             dataApi.setTitle(jsonObject.getString( name: "title"));
159                             dataApi.setVoteAverage(jsonObject.getDouble( name: "vote_average"));
160                             dataApi.setOverview(jsonObject.getString( name: "overview"));
161                             dataApi.setReleaseDate(formatter.format(dateFormat.parse(datePost)));
162                             dataApi.setPosterPath(jsonObject.getString( name: "poster_path"));
163                             dataApi.setBackdropPath(jsonObject.getString( name: "backdrop_path"));
164                             dataApi.setPopularity(jsonObject.getString( name: "popularity"));
165                             moviePopular.add(dataApi);
166                         showMovie();
167                     }
168                 } catch (JSONException | ParseException e) {
169                     e.printStackTrace();
170                     Toast.makeText(getApplicationContext(), text: "Gagal menampilkan data!", Toast.LENGTH_SHORT).show();
171                 }
172             }
173         }
174         @Override
175         public void onError(ANError anError) {
176             progressDialog.dismiss();
177             Toast.makeText(getApplicationContext(), text: "Tidak ada jaringan internet!", Toast.LENGTH_SHORT).show();
178         }
179     });

```

```
180     }
181
182     private void showMovie() {
183         movieAdapter = new MovieAdapter(getActivity(), moviePopular, xSelectData: this);
184         rvFilmRecommend.setAdapter(movieAdapter);
185         movieAdapter.notifyDataSetChanged();
186     }
187
188     @Override
189     public void onSelected(ModelMovie modelMovie) {
190     }
191 }
```

10. Menambahkan file ApiEndpoint.java dengan sintak seperti berikut

```
3  public class ApiEndpoint {
4
5      public static String BASEURL = "https://api.themoviedb.org/3/";
6      public static String APIKEY = "api_key=a18686e015158d29c97ac762920b061c";
7      public static String LANGUAGE = "&language=en-US";
8      public static String SEARCH_MOVIE = "search/movie?";
9      public static String QUERY = "&query=";
10     public static String MOVIE_PLAYNOW = "movie/now_playing?";
11     public static String MOVIE_POPULAR = "discover/movie?";
12     public static String URLIMAGE = "https://image.tmdb.org/t/p/w780/";
13 }
```

11. Menambahkan sintak AndroidManifest.xml seperti berikut

```
4      <uses-permission android:name="android.permission.INTERNET" />
5      <uses-permission android:name="android.permission.WAKE_LOCK" />
6      <uses-permission android:name="android.permission.VIBRATE" />
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21     <activity android:name=".JsonParseActivity">
22         </activity>
23     <activity android:name=".JsonAPIActivity">
24         </activity>
25     <activity android:name=".MovieDBActivity">
26         </activity>
```

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan implementasi code JSON API movieDB.
2. Dokumentasi hasil implementasi code dalam bentuk laporan.

g. Kesimpulan

Mahasiswa menerapkan JSON API movieDB pada android.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam implementasi code disertai menjelaskan setiap baris code JSON API movieDB	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam implementasi code JSON API movieDB	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam implementasi code JSON API movieDB	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam implementasi code JSON API movieDB	(1)	2	3	4
Jumlah skor					

ACARA 40 : PENGENALAN JSON

Pokok Bahasan	: Pengujian JSON API MovieDB melalui Postman
Acara Praktikum/Pertemuan	: Minggu 11 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep MovieDB.
2. Mahasiswa mampu menguji hasil implemntasi JSON API movieDB melalui postman.

b. Indikator

Keberhasilan mahasiswa dalam menguji hasil implemntasi JSON API movieDB melalui postman.

c. Dasar Teori

Postman

Postman adalah suatu aplikasi yang digunakan untuk pengembangan API. Aplikasi ini berkerja dengan cara untuk menguji coba API yang akan dikirim dan diterima dari server. Dengan begitu, pengembang aplikasi dapat mengetahui struktur data yang akan digunakan pada aplikasi yang tengah dibuat sehingga mampu mempercepat proses pengembangan aplikasi.

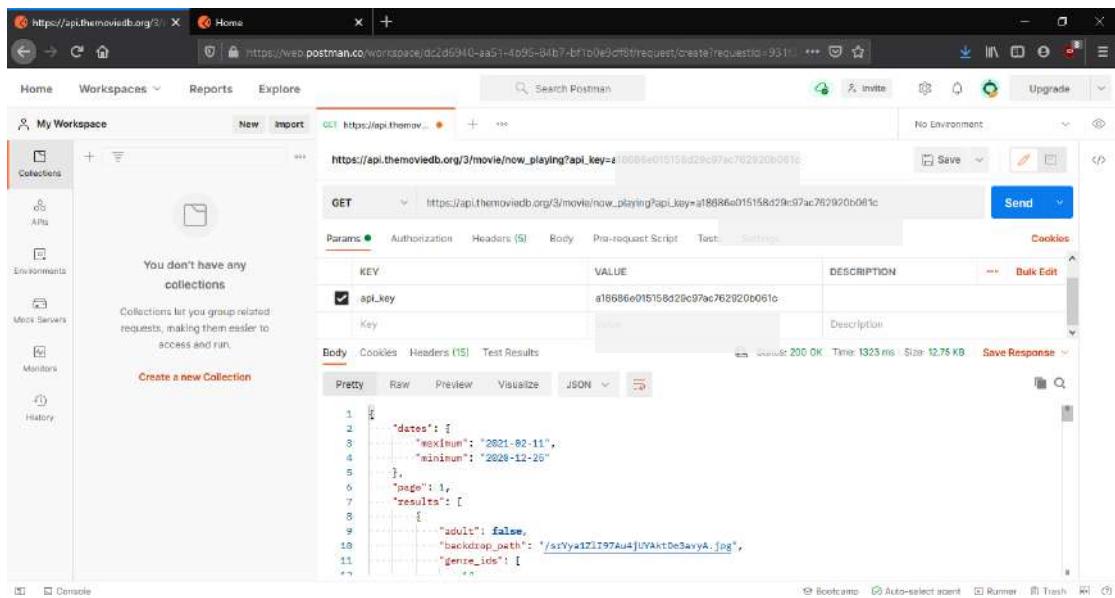
d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

e. Prosedur Kerja

1. Menguji url API melalui postman
2. Buka halaman <https://web.postman.co/home>
3. Copykan link berikut pada postman

https://api.themoviedb.org/3/movie/now_playing?api_key=a18686e015158d29c97ac762920b061c



4. Hasil running aplikasi di bawah ini:

The screenshot displays a web application interface with three main sections labeled 1, 2, and 3. Sections 1 and 2 show JSON and JSON+API data for characters, while section 3 shows JSON and API data for movies. The right side of the interface shows movie recommendations.

Section 1 (JSON):

Name	Role	Team	Email	Phone
Monkey D. Luffy	Leader	East Blue	ravig@gmail.com	+91 0000000000
Roronoa Zoro	Vice Leader	East Blue	johnny_depp@gmail.com	+91 0000000000
Nami	Navigator	East Blue	leonardo_dicaprio@gmail.com	+91 0000000000

Section 2 (JSON and API):

Name	Role	Team	Email	Phone
Ravi Tamada		East Blue	ravig@gmail.com	+91 0000000000
Johnny Depp		East Blue	johnny_depp@gmail.com	+91 0000000000
Leonardo Dicaprio		East Blue	leonardo_dicaprio@gmail.com	+91 0000000000
John Wayne			john_wayne@gmail.com	+91 0000000000
Angelina Jolie			angelina_jolie@gmail.com	+91 0000000000
Dido			dido@gmail.com	+91 0000000000
Adele			adele@gmail.com	+91 0000000000
Hugh Jackman			hugh_jackman@gmail.com	+91 0000000000
Will Smith			will_smith@gmail.com	+91 0000000000
Clint Eastwood				

Section 3 (JSON and API):

Title	Description	Rating
Wonder Woman 1984	Wonder Woman comes into conflict with the Soviet Union during the Cold War in the 1980s while Steve Trevor is captured.	★★★★★
Outside the Wire	In the near future, a drone pilot is sent into a deadly militarized zone and must ...	★★★★★
Soul	Joe Gardner is a middle school teacher with a love for jazz music. After a success...	★★★★★
Skylines	When a virus threatens to turn the now...	★★★★★

f. Hasil dan Pembahasan

- Project aplikasi dari percobaan implementasi code JSON API movieDB.
- Dokumentasi hasil output aplikasi dalam bentuk laporan dilengkapi dengan analisa code pada implementasi source code.
- Dokumentasi hasil pengujian url API melalui postman.

g. Kesimpulan

Mahasiswa menerapkan JSON API movieDB serta menguji url API melalui postman.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara detail setiap baris code, dan pengujian melalui postman	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara singkat setiap baris code, dan pengujian melalui postman	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam implementasi code serta menjelaskan setiap baris code, dan pengujian melalui postman	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam implementasi code serta menjelaskan setiap baris code, dan pengujian melalui postman	(1)	2	3	4
Jumlah skor					

ACARA 41 : PENGENALAN RETROFIT, VOLLEY, GET, POST

Pokok Bahasan	: Pengenalan Retrofit
Acara Praktikum/Pertemuan	: Minggu 12 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

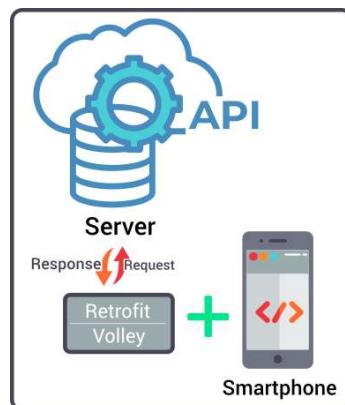
1. Mahasiswa mampu memahami konsep retrofit.
2. Mahasiswa mampu menerapkan retrofit.

b. Indikator

Keberhasilan mahasiswa memahami serta menerapkan konsep retrofit, dengan implementasi source code pada kasus tertentu.

c. Dasar Teori

Retrofit dan Volley merupakan suatu *library* yang digunakan sebagai media komunikasi antara *client* (*smartphone*) dengan server berbasis *web service*. Perhatikan gambar berikut ini, kedua *library* seolah-olah menjadi penghubung antara *smartphone* dengan server. Smartphone bertugas melakukan *request* kepada server. Setelah itu, server akan memberikan *response* data sesuai yang diminta (*request*).



Kedua *library* ini menyediakan 4 metode HTTP Request yang umum digunakan, antara lain:

- GET, digunakan untuk membaca data (*read only*) pada resource.
- POST, digunakan untuk mengirimkan atau membuat *resource* baru pada *server*.
- PUT, digunakan untuk memperbaharui data pada *resource*.

- DELETE, digunakan untuk menghapus data pada *resource*.

Retrofit mampu menangani *request* dan *response* dengan mudah seperti pemanggilan fungsi didalam bahasa java, dimana volley harus menambahkan kode ekstra agar memperoleh hasil yang serupa. Ketika ingin menampilkan gambar dari suatu tautan, retrofit tidak melakukan secara langsung. Sistem harus menambahkan *library* lain seperti **Picasso** agar gambar tampil dilayar android. Berbeda halnya dengan volley yang sudah mendukung *image loading*, sehingga tidak perlu menambahkan *library* lainnya.

Retrofit

Retrofit merupakan *library* yang dibuat untuk memudahkan komunikasi berbasis *web service* pada Java dan Android. *Library* ini disebut juga sebagai REST (**R**epresentational **S**tate **T**ransfer) *Client* pada Java dan Android. Komunikasi yang digunakan berupa pertukaran data berbentuk JSON (*JavaScript Object Notation*) atau XML (*Extensible Markup Language*). Retrofit menggunakan *library* OkHttp agar dapat melakukan *request* HTTP.

Pada praktikum kali ini, format pertukaran data menggunakan JSON karena lebih cepat serta ringan. JSON juga lebih umum digunakan oleh para pengembang aplikasi baik berbasis *web* maupun *mobile*.

Langkah pertama yaitu tambahkan *permission* INTERNET pada Manifest agar aplikasi dapat terhubung ke server meskipun server terletak pada *localhost*. Selanjutkan, tambahkan *library* **retrofit** dan **Gson** pada gradle nya. **Gson** digunakan untuk membaca data *response* dari server berupa API agar mudah digunakan. Selain itu, retrofit juga menggunakan JAVA 8 yang dapat diambil di *official website* android. Tambahkan kode dibawah ini pada build.gradle

```
compileOptions {  
    sourceCompatibility JavaVersion.VERSION_1_8  
    targetCompatibility JavaVersion.VERSION_1_8  
}
```

d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

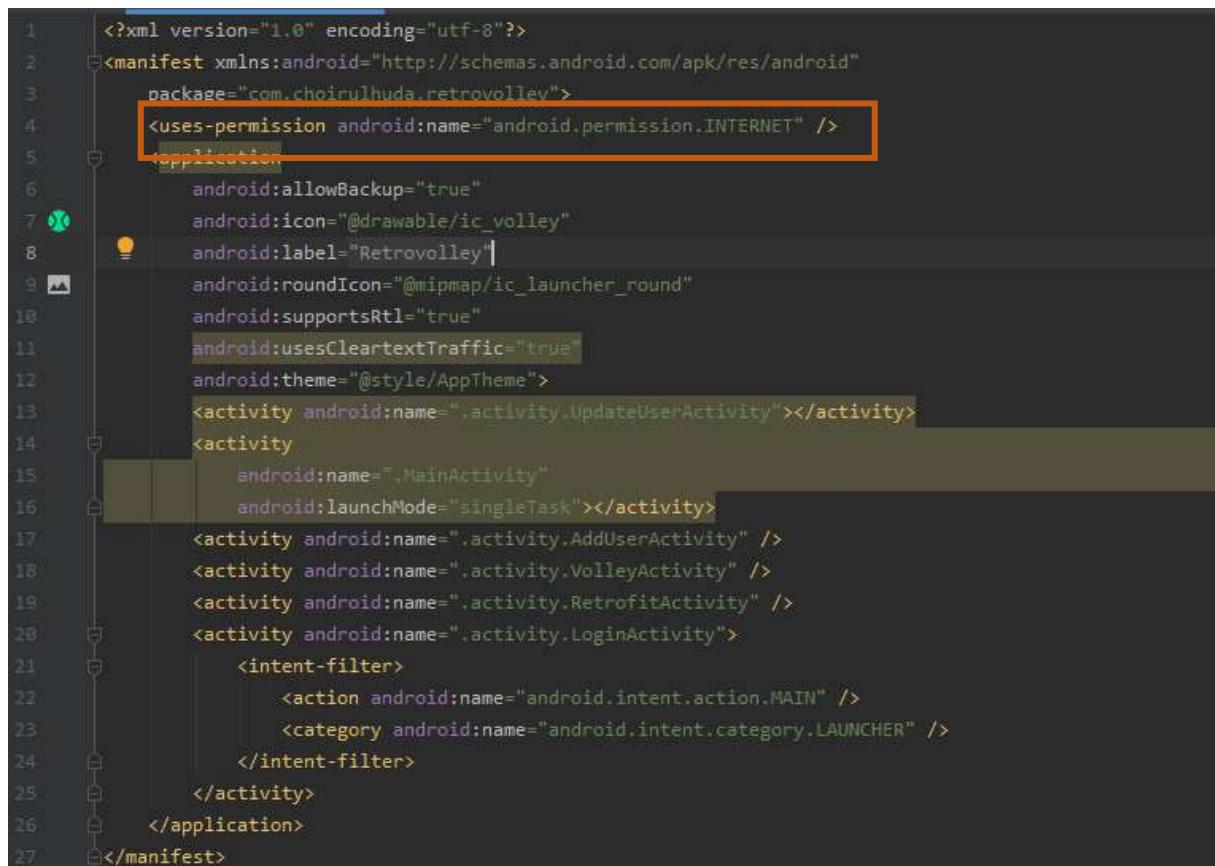
e. Prosedur Kerja

- Praktikum ini akan mengimplementasikan instalasi Retrofit dan Volley pada aplikasi android.
- Instalasi meliputi library **retrofit** dan **converter-gson**.

- Library converter-gson digunakan untuk mengkonversi response dari server yang berbentuk JSON agar mudah dibaca oleh android dan mudah diolah ketika menuliskan kode program.

Retrofit

1. Buka **Android Studio**
2. Buat project baru, dan berikan nama **Retrovolley**.
3. Sesuaikan API, disarankan menggunakan minimal API 21 karena saat ini versinya semakin tinggi.
4. Tambahkan *permission* INTERNET pada manifest android. Hal ini diperlukan meskipun web server berada pada *local host* di laptop masing-masing.
 - Permission ini menandakan bahwa aplikasi android dapat melakukan koneksi internet.



```

1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      package="com.choirulhuda.retrovolley">
4          <uses-permission android:name="android.permission.INTERNET" />
5          <application>
6              android:allowBackup="true"
7              android:icon="@drawable/ic_volley"
8              android:label="Retrovolley"
9              android:roundIcon="@mipmap/ic_launcher_round"
10             android:supportsRtl="true"
11             android:usesCleartextTraffic="true"
12             android:theme="@style/AppTheme">
13                 <activity android:name=".activity.UpdateUserActivity"></activity>
14                 <activity
15                     android:name=".MainActivity"
16                     android:launchMode="singleTask"></activity>
17                 <activity android:name=".activity.AddUserActivity" />
18                 <activity android:name=".activity.VolleyActivity" />
19                 <activity android:name=".activity.RetrofitActivity" />
20                 <activity android:name=".activity.LoginActivity">
21                     <intent-filter>
22                         <action android:name="android.intent.action.MAIN" />
23                         <category android:name="android.intent.category.LAUNCHER" />
24                     </intent-filter>
25                 </activity>
26             </application>
27         </manifest>

```

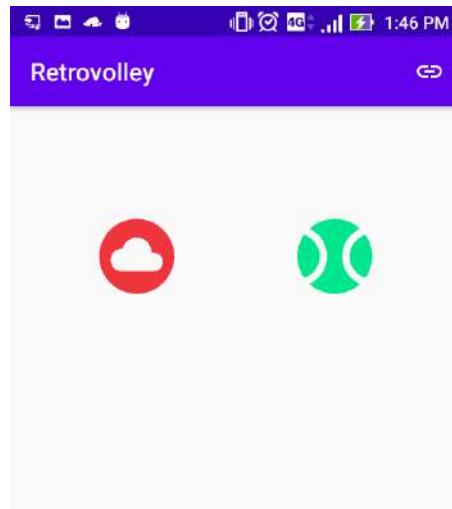
5. Selanjutnya, tambahkan dua *library* dibawah ini agar dapat mengimplementasikan retrofit pada proyek android.

```

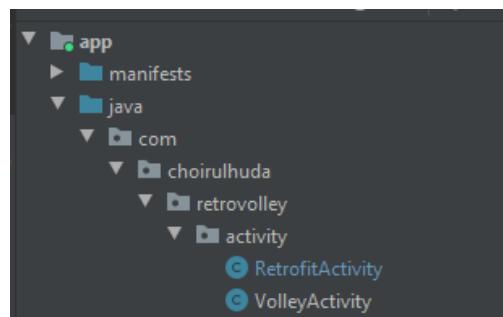
implementation 'com.squareup.retrofit2:retrofit:2.9.0'
implementation 'com.squareup.retrofit2:converter-gson:2.9.0'

```

6. Kemudian klik **sync now** di pojok kanan atas, agar *library* tersimpan pada proyek android.
7. Tambahkan 2 tombol pada Main Activity seperti dibawah ini.
 - Icon ditambahkan dari dalam Android Studio dengan cara klik kanan pada folder **drawable** → **New** → **Vector Asset**.
 - Pilih ikon yang diinginkan dan beri nama ikon pertama **ic_retrofit**.
 - Selanjutnya tambahkan ikon baru dan beri nama **ic_volley**.
8. Munculkan kedua ikon pada layout **activity_main** dengan cara menambahkan komponen **ImageView** satu persatu mulai dari **ic_retrofit** dilanjutkan pada **ic_volley**.
 - Jika benar, maka akan muncul gambar dibawah ini.



9. Buatlah 2 activity baru untuk menangani Retrofit dan Volley dengan cara klik **File** → **New** → **Empty Activity**.
 - Pertama beri nama **RetrofitActivity** dan yang kedua **VolleyActivity**.
10. Masukkan kedua activity tersebut kedalam package baru agar lebih rapi, dengan cara klik kanan pada folder **retrovolley** → **New** → **Package** dan beri nama **activity**.
 - *Drag and drop* **RetrofitActivity** dan **VolleyActivity** pada package **activity**. Jika benar akan terlihat seperti ini:



11. Kembali pada **activity_main.xml**.

- Berikan aksi pada kedua ikon tersebut melalui file XML seperti dibawah ini.
- Ketik secara manual pada **ic_volley** dengan nama **actionVolley** dan *create* aksinya.
- Pada **ic_retrofit** beri nama **actionRetrofit**.

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".MainActivity">
9      <ImageView
10         android:id="@+id/img_retrofit"
11         android:layout_width="wrap_content"
12         android:layout_height="wrap_content"
13         app:layout_constraintEnd_toStartOf="@+id/guideline"
14         app:layout_constraintHorizontal_bias="0.604"
15         app:layout_constraintStart_toStartOf="parent"
16         app:layout_constraintTop_toTopOf="@+id/guideline2"
17         app:srcCompat="@drawable/ic_retrofit"
18         android:onClick="actionRetrofit"/>
19      <androidx.constraintlayout.widget.Guideline
20          android:id="@+id/guideline"
21          android:layout_width="wrap_content"
22          android:layout_height="wrap_content"
23          android:orientation="vertical"
24          app:layout_constraintGuide_percent="0.5" />
25      <androidx.constraintlayout.widget.Guideline
26          android:id="@+id/guideline2"
27          android:layout_width="wrap_content"
28          android:orientation="horizontal"
29          app:layout_constraintGuide_percent="0.15" />
30      <ImageView
31          android:id="@+id/img_volley"
32          android:layout_width="wrap_content"
33          android:layout_height="wrap_content"
34          app:layout_constraintEnd_toEndOf="parent"
35          app:layout_constraintHorizontal_bias="0.406"
36          app:layout_constraintStart_toStartOf="@+id/guideline"
37          app:layout_constraintTop_toTopOf="@+id/img_retrofit"
38          app:srcCompat="@drawable/ic_volley"
39          android:onClick="actionVolley"/>
40    </androidx.constraintlayout.widget.ConstraintLayout>
```

12. Tuliskan **intent** pada masing-masing aksi di dalam **Main Activity** seperti dibawah ini

```
30  public class MainActivity extends AppCompatActivity {
31
32      private final String TAG = getClass().getSimpleName();
33      private SharedPreferences pref;
34      private int flagExit = 0;
35      private User user;
36
37      @Override
38      protected void onCreate(Bundle savedInstanceState) {
39          super.onCreate(savedInstanceState);
40          setContentView(R.layout.activity_main);
41
42          pref = getSharedPreferences(GlobalVariable.PREFERENCE_NAME, MODE_PRIVATE);
43
44          Bundle extras = getIntent().getExtras();
45          if (extras != null) {
46              setTitle(extras.getString(GlobalVariable.CURRENT_USERNAME));
47          } else
48              setTitle("Home");
49      }
50
51      @Override
52      public boolean onCreateOptionsMenu(Menu menu) {
53          getMenuInflater().inflate(R.menu.main, menu);
54          return true;
55      }
56      @Override
57      public boolean onOptionsItemSelected(MenuItem item) {
58          int id = item.getItemId();
59          switch (id) {
60              case R.id.action_url:
61                  View urlView = getLayoutInflater().inflate(R.layout.prompt_url, null);
62                  EditText edtBaseURL = urlView.findViewById(R.id.edt_base_url);
63                  String globalURL = pref.getString(GlobalVariable.BASE_URL, null);
64
65                  if (globalURL != null) {
66                      edtBaseURL.setText(globalURL);
67                  }
68                  new AlertDialog.Builder( context: this)
69                      .setTitle("Base URL")
70                      .setView(urlView)
71                      .setPositiveButton( text: "OK", new DialogInterface.OnClickListener() {
72                          @Override
73                          public void onClick(DialogInterface dialog, int which) {
74                              String globalURL = edtBaseURL.getText().toString();
75                              SharedPreferences.Editor prefEditor = pref.edit();
76                              prefEditor.putString(GlobalVariable.BASE_URL, globalURL);
77                              prefEditor.apply();
78                          }
79                      })
80                      .setNegativeButton( text: "Batal", new DialogInterface.OnClickListener() {
81                          @Override
```

```

82.     public void onClick(DialogInterface dialog, int which) { dialog.dismiss(); }
83.     }).show();
84.     break;
85.   }
86.   return super.onOptionsItemSelected(item);
87. }
88.
89. @Override
90. public void onBackPressed() {
91.   flagExit++;
92.   if (flagExit == 2) {
93.     super.onBackPressed();
94.   } else {
95.     Toast.makeText( context: this, text: "Tekan lagi untuk keluar", Toast.LENGTH_SHORT).show();
96.   }
97. }
98.
99. public void actionRetrofit(View view) {
100.   Intent retrofit = new Intent( packageContext: this, RetrofitActivity.class);
101.   startActivity(retrofit);
102. }
103. public void actionVolley(View view) {
104.   Intent volley = new Intent( packageContext: this, VolleyActivity.class);
105.   startActivity(volley);
106. }
107. }
108.
109. }

```

13. Agar aplikasi lebih menarik, maka proses input **BASE URL** ditambahkan melalui ikon yang terletak pada **ActionBar**.
14. Tambahkan ikon **link** melalui **vector asset** seperti yang sudah dijelaskan diatas.
15. Buatlah direktori baru pada **res** dengan cara klik kanan pada folder **res**→**New**→ **Directory** beri nama **menu**.
16. Kemudian, buat xml baru dengan nama **main.xml** dengan cara klik kanan pada folder **menu**→**New**→**Menu Resource File** beri nama main.
17. Tuliskan kode berikut ini pada **retrofit.xml**, agar terbentuk mancul ikon add:

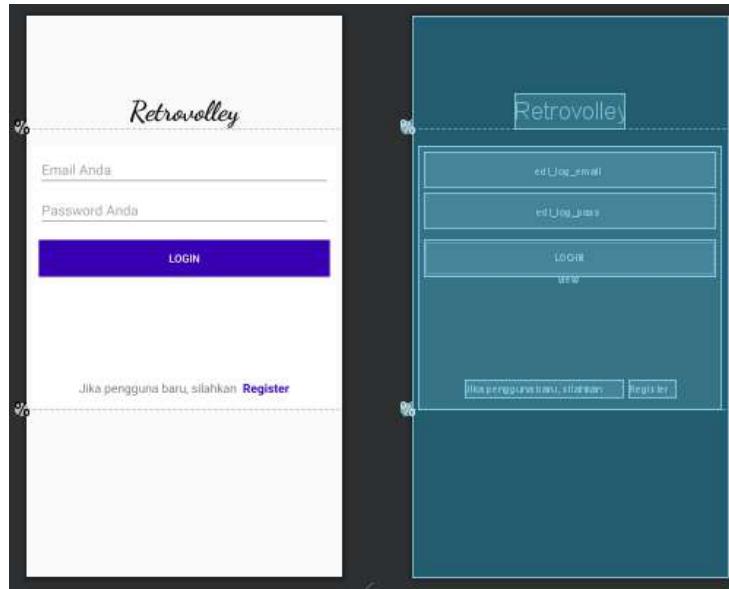
```

1  <?xml version="1.0" encoding="utf-8"?>
2  <menu xmlns:android="http://schemas.android.com/apk/res/android"
3    xmlns:app="http://schemas.android.com/apk/res-auto">
4
5    <item
6      android:id="@+id/action_url"
7      android:orderInCategory="100"
8      android:icon="@drawable/ic_link"
9      android:title="Base URL"
10     app:showAsAction="ifRoom" />
11
12 </menu>
13

```

Membuat Halaman Login

- Fitur login merupakan langkah awal yang disediakan pada hampir seluruh aplikasi.
- Ditambah lagi, fitur ini diterapkan saat membuat proyek akhir dengan sistem operasi android.



1. Pertama, buatlah halaman login yang berisi **Edittext Email** dan **Password**, serta **Button Login** seperti gambar diatas. Adapan kode yang terbentuk seperti dibawah ini:

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      android:background="@drawable/bg_login"
9      tools:context=".activity.LoginActivity">
10     <View
11         android:id="@+id/view"
12         android:layout_width="0dp"
13         android:layout_height="0dp"
14         android:layout_marginStart="8dp"
15         android:layout_marginTop="24dp"
16         android:layout_marginEnd="8dp"
17         android:background="@android:color/white"
18         app:layout_constraintBottom_toTopOf="@+id/guideline6"
19         app:layout_constraintEnd_toEndOf="parent"
20         app:layout_constraintHorizontal_bias="0.0"
21         app:layout_constraintStart_toStartOf="parent"
22         app:layout_constraintTop_toTopOf="@+id/guideline5"
23         app:layout_constraintVertical_bias="0.0" />
24     <TextView
25         android:id="@+id/textView"
26         android:layout_width="wrap_content"
27         android:layout_height="wrap_content"
```

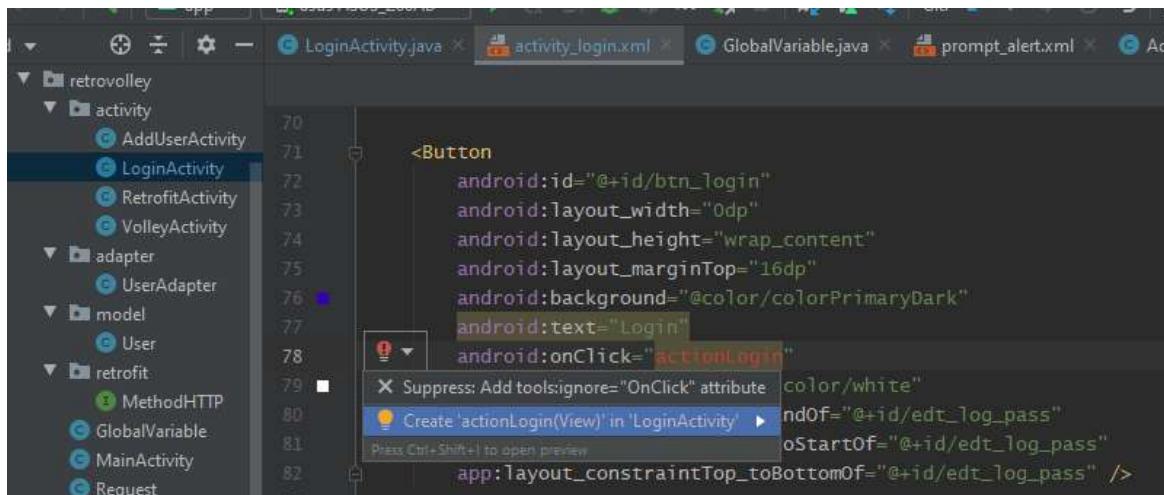
```
28         android:fontFamily="cursive"
29         android:text="Retrovolley"
30         android:textAllCaps="false"
31         android:textColor="@android:color/white"
32         android:textSize="36sp"
33         android:textStyle="bold"
34         app:layout_constraintBottom_toTopOf="@+id/guideline5"
35         app:layout_constraintEnd_toEndOf="parent"
36         app:layout_constraintStart_toStartOf="parent" />
37     <androidx.constraintlayout.widget.Guideline
38         android:id="@+id/guideline5"
39         android:layout_width="wrap_content"
40         android:layout_height="wrap_content"
41         android:orientation="horizontal"
42         app:layout_constraintGuide_percent="0.2" />
43     <EditText
44         android:id="@+id/edt_log_email"
45         android:layout_width="0dp"
46         android:layout_height="wrap_content"
47         android:layout_marginStart="8dp"
48         android:layout_marginTop="8dp"
49         android:layout_marginEnd="8dp"
50         android:ems="10"
51         android:hint="Email Anda"
52         android:inputType="textEmailAddress"
53         app:layout_constraintEnd_toEndOf="@+id/view"
54         app:layout_constraintStart_toStartOf="@+id/view"
55         app:layout_constraintTop_toTopOf="@+id/view" />
56     <EditText
57         android:id="@+id/edt_log_pass"
58         android:layout_width="0dp"
59         android:layout_height="wrap_content"
60         android:layout_marginTop="8dp"
61         android:ems="10"
62         android:hint="Password Anda"
63         android:inputType="textPassword"
64         app:layout_constraintEnd_toEndOf="@+id/edt_log_email"
65         app:layout_constraintStart_toStartOf="@+id/edt_log_email"
66         app:layout_constraintTop_toBottomOf="@+id/edt_log_email" />
67     <Button
68         android:id="@+id/btn_login"
69         android:layout_width="0dp"
70         android:layout_height="wrap_content"
71         android:layout_marginTop="16dp"
72         android:background="@color/colorPrimaryDark"
73         android:text="Login"
74         android:onClick="actionLogin"
75         android:textColor="@android:color/white"
76         app:layout_constraintEnd_toEndOf="@+id/edt_log_pass"
77         app:layout_constraintStart_toStartOf="@+id/edt_log_pass"
78         app:layout_constraintTop_toBottomOf="@+id/edt_log_pass" />
```

```

79     <TextView
80         android:id="@+id/textView2"
81         android:layout_width="wrap_content"
82         android:layout_height="wrap_content"
83         android:layout_marginBottom="16dp"
84         android:text="jika pengguna baru, silahkan"
85         android:textSize="16sp"
86         app:layout_constraintBottom_toBottomOf="@+id/view"
87         app:layout_constraintEnd_toEndOf="@+id/btn_login"
88         app:layout_constraintHorizontal_bias="0.3"
89         app:layout_constraintStart_toStartOf="@+id/btn_login" />
90     <TextView
91         android:id="@+id/txt_register"
92         android:layout_width="wrap_content"
93         android:layout_height="wrap_content"
94         android:layout_marginStart="8dp"
95         android:onClick="actionRegister"
96         android:text="Register"
97         android:textColor="@color/colorPrimaryDark"
98         android:textSize="16sp"
99         android:textStyle="bold"
100        app:layout_constraintBottom_toBottomOf="@+id/textView2"
101        app:layout_constraintStart_toEndOf="@+id/textView2"
102        app:layout_constraintTop_toTopOf="@+id/textView2" />
103    <androidx.constraintlayout.widget.Guideline
104        android:id="@+id/guideline6"
105        android:layout_width="wrap_content"
106        android:layout_height="wrap_content"
107        android:orientation="horizontal"
108        app:layout_constraintGuide_percent="0.7" />
109    </androidx.constraintlayout.widget.ConstraintLayout>

```

2. Tambahkan **Textview Register** untuk melakukan registrasi apabila pengguna ingin melakukan penambahan secara langsung.
3. Selanjutnya beri aksi **onClick** langsung melalui **activity_login.xml** pada tombol **Submit** dengan cara menuliskan **actionLogin** pada kode tombol login. Perhatikan gambar dibawah ini:



4. Selanjutnya, inisiasi edittext email dan password pada menu **oncreate** didalam LoginActivity seperti gambar dibawah ini:

```

41 public class LoginActivity extends AppCompatActivity {
42
43     private final String TAG = getClass().getSimpleName();
44     private SharedPreferences pref;
45     private EditText edtUserEmail, edtUserPassword;
46
47     @Override
48     protected void onCreate(Bundle savedInstanceState) {
49         super.onCreate(savedInstanceState);
50         setContentView(R.layout.activity_login);
51
52         edtUserEmail = findViewById(R.id.edt_log_email);
53         edtUserPassword = findViewById(R.id.edt_log_pass);
54
55         pref = getSharedPreferences(GlobalVariable.PREFERENCE_NAME, MODE_PRIVATE);
56     }
57
58     @Override
59     public boolean onCreateOptionsMenu(Menu menu) {
60         getMenuInflater().inflate(R.menu.main, menu);
61         return true;
62     }
63
64     @Override
65     public boolean onOptionsItemSelected(@NonNull MenuItem item) {
66         int id = item.getItemId();
67         switch (id) {
68             case R.id.action_url:
69                 View urlView = getLayoutInflater().inflate(R.layout.prompt_url, null);
70                 EditText edtBaseURL = urlView.findViewById(R.id.edt_base_url);
71                 String globalURL = pref.getString(GlobalVariable.BASE_URL, null);
72
73                 if (globalURL != null) {
74                     edtBaseURL.setText(globalURL);
75                 }
76                 new AlertDialog.Builder(context)
77                     .setTitle("Base URL")
78                     .setView(urlView)
79                     .setPositiveButton("OK", new DialogInterface.OnClickListener() {
80                         @Override
81                         public void onClick(DialogInterface dialog, int which) {
82                             String globalURL = edtBaseURL.getText().toString();
83                             SharedPreferences.Editor prefEditor = pref.edit();
84                             prefEditor.putString(GlobalVariable.BASE_URL, globalURL);
85                             prefEditor.apply();
86                         }
87                     }).setNegativeButton("Batal", new DialogInterface.OnClickListener() {
88                         @Override
89                         public void onClick(DialogInterface dialog, int which) { dialog.dismiss(); }
90                     }).show();
91
92

```

```

93         break;
94     }
95     return super.onOptionsItemSelected(item);
96 }
97
98 public void actionRegister(View view) {
99     Intent intent = new Intent( packageContext: this, AddUserActivity.class);
100    intent.putExtra(GlobalVariable.TYPE_CONN, GlobalVariable.RETROFIT);
101    startActivity(intent);
102 }

```

5. Berikutnya, verifikasi input pada edittext useremail dan password ketika pengguna menekan tombol **Login**. Periksa apakah pengguna telah mengisi kolom ataukah belum seperti kode dibawah ini pada LoginActivity.

```

104     public void actionLogin(View view) {
105         boolean isValid = false;
106
107         if (edtUserEmail.getText().toString().isEmpty()){
108             edtUserEmail.setError("Tidak boleh kosong");
109             edtUserEmail.requestFocus();
110             isValid = false;
111         } else
112             isValid = true;
113
114         if (edtUserPassword.getText().toString().isEmpty()){
115             edtUserPassword.setError("Tidak boleh kosong");
116             edtUserPassword.requestFocus();
117             isValid = false;
118         } else
119             isValid = true;
120
121         if (isValid) {
122             User user = new User();
123             user.setUser_email(edtUserEmail.getText().toString());
124             user.setUser_password(edtUserPassword.getText().toString());
125
126             loginUsingRetrofit(user.getUser_email(), user.getUser_password());
127
128             //loginUsingVolley(user.getUser_email(), user.getUser_password());
129         }
130     }

```

6. Buatlah method untuk menjalankan fungsi **loginUsingRetrofit** pada LoginActivity seperti dibawah ini:

```

207     public void loginUsingRetrofit(String email, String password) {
208         ProgressDialog proDialog = new ProgressDialog( context: this);
209         proDialog.setTitle("Retrovolley");
210         proDialog.setMessage("Silahkan tunggu");
211         proDialog.show();
212
213         String globalURL = pref.getString(GlobalVariable.BASE_URL, s1: null);
214
215         Retrofit.Builder builder = new Retrofit.Builder()
216             .baseUrl(globalURL)
217             .addConverterFactory(GsonConverterFactory.create());
218
219         Retrofit retrofit = builder.build();
220         MethodHTTP client = retrofit.create(MethodHTTP.class);
221         Call<UserResponse> call = client.login(email, password);
222
223         call.enqueue(new Callback<UserResponse>() {
224             @Override
225             public void onResponse(Call<UserResponse> call, Response<UserResponse> response) {
226                 proDialog.dismiss();
227                 if (response.body() != null ){
228                     if (response.body().getCode() == 200) {
229                         User loggedUser = response.body().getUser_list().get(0);
230                         Intent intent = new Intent( packageContext: LoginActivity.this,
231                                         MainActivity.class);
232                         intent.putExtra(GlobalVariable.CURRENT_USERNAME,
233                                         loggedUser.getUser_fullname());
234                         startActivity(intent);
235                         startActivity(intent);
236                         finish();
237                     } else if (response.body().getCode() == 401){
238                         new AlertDialog.Builder( context: LoginActivity.this)
239                             .setTitle("Peringatan!")
240                             .setMessage(response.body().getStatus())
241                             .setPositiveButton( text: "OK", new DialogInterface.OnClickListener() {
242                                 @Override
243                                 public void onClick(DialogInterface dialogInterface, int i) {
244                                     edtUserPassword.setText("");
245                                     dialogInterface.dismiss();
246                                 }
247                             }).show();
248                     } else {
249                         //code 400
250                         new AlertDialog.Builder( context: LoginActivity.this)
251                             .setTitle("Peringatan!")
252                             .setMessage(response.body().getStatus())
253                             .setPositiveButton( text: "OK", new DialogInterface.OnClickListener() {
254                                 @Override
255                                 public void onClick(DialogInterface dialogInterface, int i) {
256                                     Intent intent = new Intent( packageContext: LoginActivity.this,
257                                         AddUserActivity.class);
258                                     intent.putExtra(GlobalVariable.TYPE_CONN,
259                                         GlobalVariable.RETROFIT);

```

```

259.         startActivity(intent);
260.         finish();
261.     }
262.     .setNegativeButton( text "Batal",
263.         new DialogInterface.OnClickListener() {
264.             @Override
265.             public void onClick(DialogInterface dialogInterface, int i) {
266.                 dialogInterface.dismiss();
267.             }
268.         }).show();
269.     }
270. } else {
271.     Toast.makeText( context: LoginActivity.this, text: "Status : Error!",
272.                     Toast.LENGTH_SHORT).show();
273. }
274.
275. Log.e(TAG, msg: "Error : "+response.message());
276. }
277. @Override
278. public void onFailure(Call<UserResponse> call, Throwable t) {
279.     proDialog.dismiss();
280.     Log.d(TAG, t.getMessage());
281.     Toast.makeText( context: LoginActivity.this, text: "Error : "+t.toString(),
282.                     Toast.LENGTH_SHORT).show();
283. }
284. });
285. }

```

Membuat Halaman Registrasi

- Pertama buat activity baru dengan nama **AddUserActivity**, dan buatlah 3 **EditText** sebagai input data Nama Lengkap, Email, dan Password. ID tidak perlu ditambahkan karena sudah *auto increment* oleh DBMS secara langsung. Tambahkan **Button** untuk *submit* data pada web server. Perhatikan **activity_add_user.xml** dibawah ini:

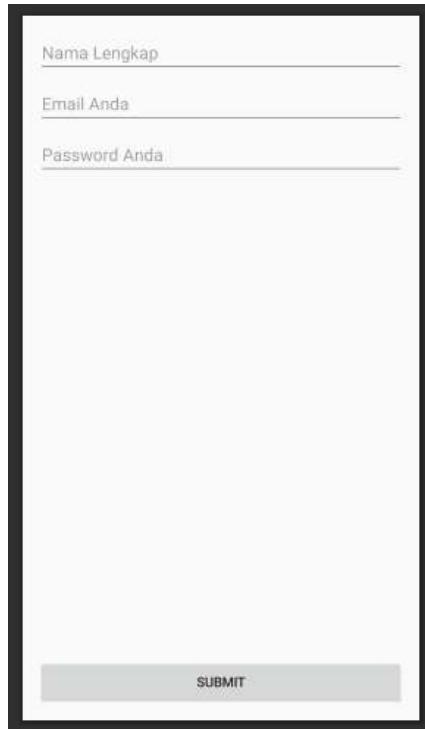
```

1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".activity.AddUserActivity">
9      <EditText
10         android:id="@+id/edt_fullname"
11         android:layout_width="0dp"
12         android:layout_height="wrap_content"
13         android:layout_marginStart="16dp"
14         android:layout_marginTop="16dp"
15         android:layout_marginEnd="16dp"
16         android:ems="10"

```

```
17    android:hint="Nama Lengkap"
18    android:inputType="textPersonName"
19    app:layout_constraintEnd_toEndOf="parent"
20    app:layout_constraintStart_toStartOf="parent"
21    app:layout_constraintTop_toBottomOf="@+id/txt_title_library" />
22  <EditText
23    android:id="@+id/edt_email"
24    android:layout_width="0dp"
25    android:layout_height="wrap_content"
26    android:layout_marginTop="8dp"
27    android:ems="10"
28    android:hint="Email Anda"
29    android:inputType="textEmailAddress"
30    app:layout_constraintEnd_toEndOf="@+id/edt_fullname"
31    app:layout_constraintStart_toStartOf="@+id/edt_fullname"
32    app:layout_constraintTop_toBottomOf="@+id/edt_fullname" />
33  <EditText
34    android:id="@+id/edt_password"
35    android:layout_width="0dp"
36    android:layout_height="wrap_content"
37    android:layout_marginTop="8dp"
38    android:ems="10"
39    android:hint="Password Anda"
40    android:inputType="textPassword"
41    app:layout_constraintEnd_toEndOf="@+id/edt_fullname"
42    app:layout_constraintStart_toStartOf="@+id/edt_fullname"
43    app:layout_constraintTop_toBottomOf="@+id/edt_email" />
44  <Button
45    android:id="@+id/btn_submit"
46    android:layout_width="0dp"
47    android:layout_height="wrap_content"
48    android:layout_marginBottom="16dp"
49    android:text="Submit"
50    android:onClick="actionSubmit"
51    app:layout_constraintBottom_toBottomOf="parent"
52    app:layout_constraintEnd_toEndOf="@+id/edt_password"
53    app:layout_constraintStart_toStartOf="@+id/edt_password" />
54  <TextView
55    android:id="@+id/txt_title_library"
56    android:layout_width="wrap_content"
57    android:layout_height="wrap_content"
58    android:layout_marginTop="8dp"
59    android:text="Sent using Volley"
60    android:textSize="18sp"
61    android:textStyle="bold"
62    app:layout_constraintEnd_toEndOf="parent"
63    app:layout_constraintStart_toStartOf="parent"
64    app:layout_constraintTop_toTopOf="parent" />
65  </androidx.constraintlayout.widget.ConstraintLayout>
```

2. Hasil XML yang ditampilkan menjadi seperti ini:



3. Melengkapi kode AddUserActivity seperti dibawah ini:

```
33  public class AddUserActivity extends AppCompatActivity {
34
35      private EditText edtFullscreen, edtEmail, edtPassword;
36      private TextView txtTitleLibrary;
37      private Button btnSubmit;
38      private String typeConn = "retrofit";
39      private SharedPreferences pref;
40
41      private final String TAG = getClass().getSimpleName();
42
43      @Override
44      protected void onCreate(Bundle savedInstanceState) {
45          super.onCreate(savedInstanceState);
46          setContentView(R.layout.activity_add_user);
47          edtFullscreen = findViewById(R.id.edt_fullname);
48          edtEmail = findViewById(R.id.edt_email);
49          edtPassword = findViewById(R.id.edt_password);
50          btnSubmit = findViewById(R.id.btn_submit);
51          txtTitleLibrary = findViewById(R.id.txt_title_library);
52
53          setTitle("Tambah User");
54
55          pref = getSharedPreferences(GlobalVariable.PREFERENCE_NAME, MODE_PRIVATE);
56
57          Bundle extras = getIntent().getExtras();
58          if (extras != null) {
59              typeConn = extras.getString(GlobalVariable.TYPE_CONN, defaultValue: "Undefined");
60              if (typeConn.equalsIgnoreCase(GlobalVariable.RETROFIT)){
61                  txtTitleLibrary.setText("Send Using Retrofit");
```

```
62         }
63     } else {
64         txtTitleLibrary.setText("Send using Volley");
65     }
66 }
67 @Override
68 public void onBackPressed() {
69     super.onBackPressed();
70     finish();
71 }
72
73 public void actionSubmit(View view) {
74     boolean isValid = false;
75
76     if (edtfullname.getText().toString().isEmpty()) {
77         edtfullname.setError("Tidak boleh kosong");
78         edtfullname.requestFocus();
79         isValid = false;
80     } else {
81         isValid = true;
82     }
83
84     if (edtemail.getText().toString().isEmpty()) {
85         edtemail.setError("Tidak boleh kosong");
86         edtemail.requestFocus();
87         isValid = false;
88     } else {
89         isValid = true;
90     }
91
92     if (edtpassword.getText().toString().isEmpty()) {
93         edtpassword.setError("Tidak boleh kosong");
94         edtpassword.requestFocus();
95         isValid = false;
96     } else {
97         isValid = true;
98     }
99
100    if (isValid) {
101        User user = new User();
102        user.setUser_fullname(edtfullname.getText().toString());
103        user.setUser_email(edtemail.getText().toString());
104        user.setUser_password(edtpassword.getText().toString());
105        if (typeConn.equalsIgnoreCase(GlobalVariable.RETROFIT))
106            submitByRetrofit(user);
107        else submitByVolley(user);
108    }
109 }
```

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan implementasi code retrofit dan UI-nya, halaman register, halaman login.
2. Dokumentasi hasil output aplikasi dalam bentuk laporan dilengkapi dengan analisa code pada implementasi source code.

g. Kesimpulan

Mahasiswa memahami serta menerapkan konsep retrofit, dengan implementasi soure code pada kasus tertentu.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara detail setiap baris code retrofit	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara singkat setiap baris code retrofit	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam implementasi code serta menjelaskan setiap baris code retrofit	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam implementasi code serta menjelaskan setiap baris code retrofit	(1)	2	3	4
Jumlah skor					

ACARA 42 : PENGENALAN RETROFIT, VOLLEY, GET, POST

Pokok Bahasan	: Pengenalan Volley
Acara Praktikum/Pertemuan	: Minggu 12 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep volley.
2. Mahasiswa mampu menerapkan volley.

b. Indikator

Keberhasilan mahasiswa memahami serta menerapkan konsep retrofit, dengan implementasi soure code pada kasus tertentu.

c. Dasar Teori

Volley

Volley adalah kerangka kerja http yang menawarkan penjadwalan permintaan, penanganan permintaan secara bersamaan, cache respons berbasis disk dan memori, prioritas permintaan, dan kemampuan pembatalan permintaan di luar kotak.

Volley bisa get dan post jenis data apa pun menggunakan volley framework. Volley menawarkan penanganan respons bawaan untuk menangani string, gambar, dan respons JSON. Jika aplikasi perlu menangani jenis respons selain string, gambar, dan JSON, maka harus dituliskan permintaan khusus dengan memperluas Request class dan menerapkan metode parseNetworkResponse () dan metode deliverResponse () .

Volley bagus untuk mentransmisikan data aplikasi sederhana. Jika fitur aplikasi memerlukan unduhan besar, lebih baik menggunakan DownloadManager karena volley menyimpan cache untuk semua respons.

Menambahkan library voli ke proyek aplikasi dengan cara memasukkan dependensi dalam file gradle seperti "compile 'com.android.volley:volley:1.0.0'" . Karena permintaan dan respons http dikirimkan melalui jaringan, aplikasi yang menggunakan volley memerlukan izin untuk mengakses internet. Anda perlu meminta izin android.permission.INTERNET dengan menambahkannya ke file manifes.

d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

e. Prosedur Kerja

1. Buka kembali proyek **Retrovolley** yang dibuat di bagian sebelumnya
2. Tambahkan library *olley* sehingga jumlah total library yang ditambahkan menjadi 3 pada **build.gradle**.

Gambar dibawah ini menunjukkan library yang telah ditambahkan:

```
implementation 'com.squareup.retrofit2:retrofit:2.9.0'
implementation 'com.squareup.retrofit2:converter-gson:2.9.0'
implementation 'com.android.volley:volley:1.1.1'
```

3. Buatlah method untuk menjalankan fungsi **loginUsingVolley** pada loginActivity seperti dibawah ini:

```
132     public void loginUsingVolley(String email, String password){
133         ProgressDialog proDialog = new ProgressDialog( context: this );
134         proDialog.setTitle("Retrovolley");
135         proDialog.setMessage("Silahkan tunggu");
136         proDialog.show();
137
138         Gson gson = new Gson();
139         String URL = pref.getString(GlobalVariable.BASE_URL, s1: null) +
140             "/olley/Login.php?email="+email+"&password="+password;
141
142         JsonObjectRequest request = new JsonObjectRequest(Request.Method.GET, URL, jsonRequest: null,
143             new com.android.volley.Response.Listener<JSONObject>() {
144
145             @Override
146             public void onResponse(JSONObject response) {
147                 proDialog.dismiss();
148                 UserResponse userResponse = gson.fromJson(response.toString(),
149                     UserResponse.class);
150                 if (userResponse.getCode() == 200) {
151                     User loggedUser = userResponse.getUser_list().get(0);
152                     Intent intent = new Intent( packageContext: LoginActivity.this,
153                         MainActivity.class);
154                     intent.putExtra(GlobalVariable.CURRENT_USERNAME,
155                         loggedUser.getUser_fullname());
156                     startActivity(intent);
157                 }
158             }
159         });
160         RequestQueue queue = Volley.newRequestQueue(this);
161         queue.add(request);
162     }
```

```

156                     finish();
157             } else if (userResponse.getCode() == 401) {
158                 new AlertDialog.Builder( context: LoginActivity.this )
159                     .setTitle("Peringatan!")
160                     .setMessage(userResponse.getStatus())
161                     .setPositiveButton( text: "OK",
162                         new DialogInterface.OnClickListener() {
163                             @Override
164                             public void onClick(DialogInterface dialogInterface, int i) {
165                                 edtUserPassword.setText("");
166                                 dialogInterface.dismiss();
167                             }
168                         }).show();
169             } else {
170                 new AlertDialog.Builder( context: LoginActivity.this )
171                     .setTitle("Peringatan!")
172                     .setMessage(userResponse.getStatus())
173                     .setPositiveButton( text: "OK",
174                         new DialogInterface.OnClickListener() {
175                             @Override
176                             public void onClick(DialogInterface dialogInterface, int i) {
177                                 Intent intent = new Intent(
178                                     packageContext: LoginActivity.this,
179                                     AddUserActivity.class);
180                                 intent.putExtra(GlobalVariable.TYPE_CONN,
181                                     GlobalVariable.RETROFIT);
182                                 startActivity(intent);
183                                 finish();
184                             }
185                         }).setNegativeButton( text: "Batal",
186                             new DialogInterface.OnClickListener() {
187                                 @Override
188                                 public void onClick(DialogInterface dialogInterface, int i) {
189                                     dialogInterface.dismiss();
190                                 }
191                         }).show();
192                     }
193             }
194         }, new com.android.volley.Response.ErrorListener() {
195             @Override
196             public void onErrorResponse(VolleyError error) {
197                 proDialog.dismiss();
198                 Toast.makeText(getApplicationContext(), text: "Login Error : " + error.getMessage(),
199                     Toast.LENGTH_SHORT).show();
200                 Log.e(TAG, msg: "Error : " + error.getMessage());
201             }
202         });
203         int socketTimeout = 5000; //delay 5 detik
204         RetryPolicy policy = new DefaultRetryPolicy(socketTimeout,
205             DefaultRetryPolicy.DEFAULT_MAX_RETRIES,
206             DefaultRetryPolicy.DEFAULT_BACKOFF_MULT);
207     }

```

```

208     request.setRetryPolicy(policy);
209     RequestQueue requestQueue = Volley.newRequestQueue( context: LoginActivity.this);
210     requestQueue.add(request);
211 }
212

```

4. Buatlah fungsi untuk melakukan submit data pada submitByVolley pada AddUserActivity seperti berikut:

```

162     public void submitByVolley(User user){
163         Gson gson = new Gson();
164         String URL = pref.getString(GlobalVariable.BASE_URL, null)
165             + "/volley/User_Registration.php";
166
167         ProgressDialog proDialog = new ProgressDialog( context: this);
168         proDialog.setTitle("Volley");
169         proDialog.setMessage("Sedang disubmit");
170         proDialog.show();
171
172         String userRequest = gson.toJson(user);
173         RequestQueue requestQueue = Volley.newRequestQueue(getApplicationContext());
174         JsonObjectRequest request = new JsonObjectRequest(com.android.volley.Request.Method.POST,
175             URL, jsonRequest null,
176             new com.android.volley.Response.Listener<JSONObject>() {
177                 @Override
178                 public void onResponse(JSONObject response) {
179                     proDialog.dismiss();
180                     if (response != null) {
181                         Request requestFormat = gson.fromJson(response.toString(), Request.class);
182                         if (requestFormat.getCode() == 201) {
183                             Toast.makeText(getApplicationContext(),
184                                 text: "Response : " +requestFormat.getStatus(),
185                                 Toast.LENGTH_SHORT).show();
186                             finish();
187                         } else if (requestFormat.getCode() == 406) {
188                             Toast.makeText(getApplicationContext(),
189                                 text: "Response : "+requestFormat.getStatus(),
190                                 Toast.LENGTH_SHORT).show();
191                         } else {
192                             Toast.makeText(getApplicationContext(),
193                                 text: "Response : "+requestFormat.getStatus(),
194                                 Toast.LENGTH_SHORT).show();
195                             finish();
196                         }
197                     }
198                 },
199                 new com.android.volley.Response.ErrorListener() {
200                     @Override
201                     public void onErrorResponse(VolleyError error) {
202                         proDialog.dismiss();
203                         Log.e(TAG, msg: "Error POST Volley : "+error.getMessage());
204                     }
205                 } );

```

```

211     };
212     requestQueue.add(request);
213     requestQueue.start();
214 }
215 }
```

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan implementasi code volley dan UI-nya.
2. Dokumentasi hasil output aplikasi dalam bentuk laporan dilengkapi dengan analisa code pada implementasi source code.

g. Kesimpulan

Mahasiswa memahami serta menerapkan konsep volley, dengan implementasi soure code pada kasus tertentu.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara detail setiap baris code volley	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam implementasi code serta menjelaskan secara singkat setiap baris code volley	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam implementasi code serta menjelaskan setiap baris code volley	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam implementasi code serta menjelaskan setiap baris code volley	(1)	2	3	4
Jumlah skor					

ACARA 43 : PENGENALAN RETROFIT, VOLLEY, GET, POST

Pokok Bahasan	: Getting Data (Get)
Acara Praktikum/Pertemuan	: Minggu 12 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu membuat program menyiapkan API.
2. Mahasiswa mampu membuat program untuk parsing data ke android.
3. Mahasiswa mampu membuat program untuk menampilkan data dalam bentuk list view yang sudah di custom.

b. Indikator

1. Keberhasilan mahasiswa menyiapkan API untuk membuat program parsing data ke android.
2. Keberhasilan mahasiswa membuat program untuk menampilkan data dalam bentuk list view yang sudah di custom.

c. Dasar Teori

Getting data

Get merupakan salah satu dari 4 metode HTTP Request yang umum digunakan. Metode ini digunakan untuk membaca data (*read only*) pada *resource*. Pada praktikum ini akan mengimplementasikan Get menggunakan Retrofit dan Volley menggunakan proyek **Retrovolley** yang telah dibuat pada pertemuan sebelumnya.

d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

e. Prosedur Kerja

Membuat Koneksi Retrovit dan Volley

1. Model dibuat menyesuaikan dengan *response* yang diberikan oleh server, oleh sebab itu perlu mengetahui format *response*. Adapun format respon yang dibutuhkan sebagai berikut :

```

1  {
2      "code": 200,
3      "status": "Successful",
4      "user_list": [
5          {
6              "id": "3",
7              "user_email": "huda@gmail.com",
8              "user_password": "1234",
9              "user_fullname": "Choirul Huda"
10         },
11     ]

```

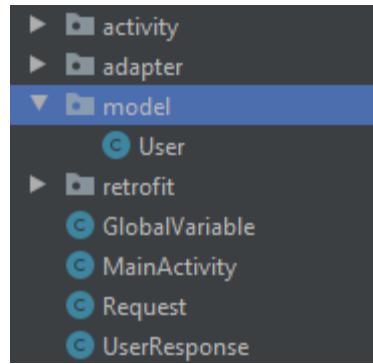
2. Pertama, buat kelas **User** yang berisi variabel **id**, **user_email**, **user_password**, dan **user_fullname** sekaligus *generate getter* dan *setter* nya seperti gambar dibawah ini.

```

3  public class User {
4      private int id;
5      private String user_email;
6      private String user_password;
7      private String user_fullname;
8
9      public int getId() { return id; }
10
11     public void setId(int id) { this.id = id; }
12
13     public String getUser_fullname() { return user_fullname; }
14
15     public void setUser_fullname(String user_fullname) { this.user_fullname = user_fullname; }
16
17     public String getUser_email() { return user_email; }
18
19     public void setUser_email(String user_email) { this.user_email = user_email; }
20
21     public String getUser_password() { return user_password; }
22
23     public void setUser_password(String user_password) { this.user_password = user_password; }
24
25     @Override
26     public String toString() {
27         return "User(" +
28             "id=" + id +
29             ", user_fullname='" + user_fullname + '\'' +
30             ", user_email='" + user_email + '\'' +
31             ", user_password='" + user_password + '\'' +
32             ')';
33     }
34 }

```

3. Kelas tersebut ke dalam *package retrofit* seperti dibawah ini



4. Selanjutnya, buat kelas untuk menampung *response* dan *request* dari metode HTTP. Data yang akan diambil berupa **code**, **status** dan daftar **user**, sehingga bentuk *response* harus menyesuaikan seperti dibawah ini. *Generate* juga *getter* dan *setter* nya :

```
8  public class UserResponse {  
9  
10     private int code;  
11     private String status;  
12     private List<User> user_list;  
13  
14     public int getCode() { return code; }  
15  
16     public void setCode(int code) { this.code = code; }  
17  
18     public String getStatus() { return status; }  
19  
20     public void setStatus(String status) { this.status = status; }  
21  
22     public List<User> getUser_list() { return user_list; }  
23  
24     public void setUser_list(List<User> user_list) { this.user_list = user_list; }  
25  
26 }
```

5. Format data yang dikembalikan dari *request* hanya berupa **code** dan **status** saja, sehingga bentuknya sebagai berikut:

```
3  public class Request {  
4      private int code;  
5      private String status;  
6  
7      public int getCode() { return code; }  
8  
9      public void setCode(int code) { this.code = code; }  
10  
11     public String getStatus() { return status; }  
12  
13     public void setStatus(String status) { this.status = status; }  
14  
15 }
```

6. Buat kelas interface **MethodHTTP** untuk menyimpan jenis *Method HTTP Request* beserta URL dari masing-masing metode. Metode yang digunakan yaitu **GET** dan **POST**, disertai URL terakhir yaitu **User_Registration.php**. Secara lengkap seperti dibawah ini

```
18 public interface MethodHTTP {  
19  
20     //get all user  
21     @GET("volley/User_Registration.php")  
22     Call<UserResponse> getUser();  
23  
24     //get specific using username and password  
25     @GET("volley/Login.php?")  
26     Call<UserResponse> login(@Query("email") String email, @Query("password") String password);  
27  
28     //get specific using id  
29     @GET("volley/User.php?")  
30     Call<UserResponse> getUserByID(@Query("id") int id);  
31  
32     //submit user registration  
33     @POST("volley/User_Registration.php")  
34     Call<Request> sendUser(@Body User user);  
35  
36     //update user  
37     @POST("volley/User.php")  
38     Call<Request> updateUser(@Body User user);  
39  
40     @DELETE("volley/User.php?")  
41     Call<Request> deleteUser(@Query("id") int id);  
42 }
```

7. Selanjutnya, buat kelas untuk menampung *globalVariable* seperti berikut :

```
5  public class GlobalVariable {  
6      private Context context;  
7      public static String BASE_URL = "http://192.168.43.51";  
8      public static String TYPE_CONN = "typeConnection";  
9      public static String RETROFIT = "retrofit";  
10     public static String VOLLEY = "volley";  
11     public static String PREFERENCE_NAME = "Retrovolley";  
12     public static String CURRENT_USER_ID = "currentUserID";  
13     public static String CURRENT_USERNAME = "currentUsername";  
14     public static String CURRENT_USER_EMAIL = "currentUserEmail";  
15     public static String CURRENT_USER_PASSWORD = "currentUserPassword";  
16     public static String UPDATE_USER = "updateUser";  
17     public static String DELETE_USER = "deleteUser";  
18     public GlobalVariable() {  
19     }  
20     public GlobalVariable(Context context) { this.context = context; }  
21 }
```

Membuat Adapter Model

Adapter dibuat agar tampilan dapat disesuaikan dengan kebutuhan. Dalam hal ini, user akan ditampilkan dalam bentuk **list view** seperti pada gambar dibawah ini. Tiap item list view berisi data **Nama user, email, dan ID user**.



- Pertama, buat UI dalam bentuk XML berupa LinearLayout seperti dibawah ini. Simpan Layout tersebut dengan nama **item_user.xml**.

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="match_parent"
4      android:layout_height="wrap_content"
5      android:orientation="vertical"
6      android:paddingLeft="4dp"
7      android:paddingTop="6dp"
8      android:paddingRight="4dp"
9      android:paddingBottom="6dp">
10     <TextView
11         android:id="@+id/txt_user_fullname"
12         android:layout_width="match_parent"
13         android:layout_height="wrap_content"
14         android:text="Name"
15         android:textColor="@android:color/black"
16         android:textSize="16sp"
17         android:textStyle="bold" />
18     <TextView
19         android:id="@+id/txt_user_email"
```

```

19         android:id="@+id/txt_user_email"
20         android:layout_width="match_parent"
21         android:layout_height="wrap_content"
22         android:text="Email"
23         android:textColor="@android:color/black" />
24     <TextView
25         android:id="@+id/txt_id"
26         android:layout_width="match_parent"
27         android:layout_height="wrap_content"
28         android:text="123"
29         android:textColor="@android:color/black" />
30     </LinearLayout>

```

2. Buat UI dalam bentuk seperti dibawah ini. Simpan Layout tersebut dengan nama **cardview_user.xml**.

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.cardview.widget.CardView
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      android:layout_width="match_parent"
6      android:layout_height="wrap_content"
7      app:cardCornerRadius="4dp">
8      <LinearLayout
9          android:layout_width="match_parent"
10         android:layout_height="match_parent"
11         android:orientation="vertical"
12         android:padding="6dp">
13         <TextView
14             android:id="@+id/txt_fullname2"
15             android:layout_width="match_parent"
16             android:layout_height="wrap_content"
17             android:text="Chirul Huda" />
18         <TextView
19             android:id="@+id/txt_id2"
20             android:layout_width="match_parent"
21             android:layout_height="wrap_content"
22             android:text="ID_5" />
23         <TextView
24             android:id="@+id/txt_user_email2"
25             android:layout_width="match_parent"
26             android:layout_height="wrap_content"
27             android:text="huda@gmail.com" />
28     </LinearLayout>
29 </androidx.cardview.widget.CardView>

```

3. Agar data pada item list view dapat ditampilkan sesuai kebutuhan, maka diperlukan **adapter** yang mengacu pada nama **model** dalam hal ini yaitu **user** sehingga nama adapternya yaitu **UserAdapter**. Buatlah kelas UserAdapter sesuai kode berikut ini:

```

18  public class UserAdapter extends ArrayAdapter<User> {
19
20      public UserAdapter(Context context, List<User> objects) { super(context, resource: 0, objects); }
21
22
23
24      @NonNull
25      @Override
26      public View getView(int position, @Nullable View convertView, @NonNull ViewGroup parent) {
27          User user = getItem(position);
28          if (convertView == null) {
29              convertView = LayoutInflater.from(getContext()).
30                  inflate(R.layout.item_user, parent, attachToRoot: false);
31          }
32
33          TextView txtFullname = convertView.findViewById(R.id.txt_user_fullname);
34          TextView txtEmail = convertView.findViewById(R.id.txt_user_email);
35          TextView txtID = convertView.findViewById(R.id.txt_id);
36          txtFullname.setText(user.getUser_fullname());
37          txtEmail.setText(user.getUser_email());
38          txtID.setText(String.valueOf(user.getId()));
39
40          return convertView;
41      }
42  }

```

- Perhatikan layout dan ID pada masing-masing textView nya!

Implementasi GET pada Retrofit

- Pada **RetrofitActivity.java**, tambahkan baris code di bawah ini pada method **onCreate**.

```

38  public class RetrofitActivity extends AppCompatActivity {
39
40      private final String TAG = getClass().getSimpleName();
41      private ListView lvUser;
42      private List<User> listUser = new ArrayList<>();
43      private SharedPreferences pref;
44
45      @Override
46      protected void onCreate(Bundle savedInstanceState) {
47          super.onCreate(savedInstanceState);
48          setContentView(R.layout.activity.Retrofit);
49          lvUser = findViewById(R.id.lv_user);
50
51          pref = getSharedPreferences(GlobalVariable.PREFERENCE_NAME, MODE_PRIVATE);
52
53          //Method GET
54          getUserFromAPI();
55
56          setTitle("Retrofit");
57      }

```

- Agar aplikasi lebih menarik, maka proses **intent** dari **RetrofitActivity** ke **AddUserActivity** ditambahkan melalui ikon yang terletak pada **ActionBar**.
- Tambahkan ikon **add** melalui **vector asset** seperti yang sudah dijelaskan diatas.

- Buatlah direktori baru pada **res** dengan cara klik kanan pada folder **res** → **New** → **Directory** beri nama **menu**.
- Kemudian, buat xml baru dengan nama **retrofit.xml** dengan cara klik kanan pada folder **menu** → **New** → **Menu Resource File** beri nama retrofit.
- Tuliskan kode berikut ini pada retrofit.xml, agar terbentuk mancul ikon add:

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <menu xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto">
4      <!-- . . . -->
5      <item
6          android:id="@+id/action_add"
7          android:orderInCategory="100"
8          android:icon="@drawable/ic_add"
9          android:title="Add"
10         app:showAsAction="ifRoom" />
11
12     </menu>
13
14
15
16
17

```

- Buatlah fungsi **getUserFromAPI()** di dalam RetrofitActivity untuk menampilkan data dari server. Masih didalam **RetrofitActivity**, panggil fungsi **onCreateOptionsMenu** dan **onOptionsItemSelected** tuliskan kode dibawah ini untuk mengaktifkan menu ActionBar. Supaya lebih mudah, pahami keseluruhan kode dibawah ini:

```

59
60     @Override
61     public boolean onCreateOptionsMenu(Menu menu) {
62         getMenuInflater().inflate(R.menu.retrofit, menu);
63         return true;
64     }
65
66     @Override
67     public boolean onOptionsItemSelected(@NonNull MenuItem item) {
68         int id = item.getItemId();
69         switch (id) {
70             case R.id.action_add:
71                 Intent intent = new Intent( mContext, AddUserActivity.class );
72                 intent.putExtra( GlobalVariable.TYPE_CONN, GlobalVariable.RETROFIT );
73                 startActivity( intent );
74                 break;
75             /*case R.id.action_refresh:
76                 getUserFromAPI();
77                 Toast.makeText(this, "Refresh", Toast.LENGTH_SHORT).show();
78                 break;*/
79         }
80         return super.onOptionsItemSelected(item);
81     }
82
83     public void actionRefresh(View view) { getUserFromAPI(); }
84
85     public void actionClose(View view) { this.finish(); }
86
87     private void getUserFromAPI() {

```

```

91     ProgressDialog proDialog = new ProgressDialog( context: this);
92     proDialog.setTitle("Retrofit");
93     proDialog.setMessage("Silahkan tunggu");
94     proDialog.show();
95
96     String globalURL = pref.getString(GlobalVariable.BASE_URL, s1: null);
97
98     Retrofit.Builder builder = new Retrofit.Builder()
99         .baseUrl(globalURL)
100        .addConverterFactory(GsonConverterFactory.create());
101
102     Retrofit retrofit = builder.build();
103     MethodHTTP client = retrofit.create(MethodHTTP.class);
104     Call<UserResponse> call = client.getUser();
105
106     call.enqueue(new Callback<UserResponse>() {
107         @Override
108         public void onResponse(Call<UserResponse> call, Response<UserResponse> response) {
109             proDialog.dismiss();
110             if (response.body().getUser_list() != null) {
111                 listUser = response.body().getUser_list();
112                 UserAdapter userAdapter = new UserAdapter(getApplicationContext(), listUser);
113                 lvUser.setAdapter(userAdapter);
114                 lvUser.setOnItemClickListener(new AdapterView.OnItemClickListener() {
115                     @Override
116                     public void onItemClick(AdapterView<?> adapterView, View view, int i,
117                         long l) {
118                         Intent intent = new Intent( packageContext: RetrofitActivity.this,
119                             UpdateUserActivity.class);
120                         intent.putExtra(GlobalVariable.CURRENT_USER_ID,
121                             listUser.get(i).getId());
122                         intent.putExtra(GlobalVariable.CURRENT_USER_EMAIL,
123                             listUser.get(i).getUser_email());
124                         intent.putExtra(GlobalVariable.CURRENT_USERNAME,
125                             listUser.get(i).getUser_fullname());
126                         intent.putExtra(GlobalVariable.CURRENT_USER_PASSWORD,
127                             listUser.get(i).getUser_password());
128                         ;
129                         startActivity(intent);
130                     }
131                 });
132             } else {
133                 Toast.makeText(getApplicationContext(), text: "Data Kosong!", Toast.LENGTH_SHORT).show();
134             }
135         }
136     });
137     @Override
138     public void onFailure(Call<UserResponse> call, Throwable t) {
139         Log.d(TAG, t.toString());
140     }
141 });

```

```
142     }
143 
144     public void tambahanRetrofit() {
145         /*HttpLoggingInterceptor interceptor = new HttpLoggingInterceptor();
146         interceptor.setLevel(HttpLoggingInterceptor.Level.BODY);
147         OkHttpClient client = new OkHttpClient.Builder().addInterceptor(interceptor).build();*/
148 
149         Gson gson = new GsonBuilder()
150             .setLenient()
151             .create();
152     }
153 
154 }
```

8. Buat UI dalam bentuk seperti dibawah ini. Simpan Layout tersebut dengan nama **activity_retofit.xml**.

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".activity.RetrofitActivity">
9      <Button
10         android:id="@+id/btn_close"
11         android:layout_width="0dp"
12         android:layout_height="wrap_content"
13         android:layout_marginStart="8dp"
14         android:layout_marginEnd="8dp"
15         android:layout_marginBottom="16dp"
16         android:onClick="actionClose"
17         android:text="close"
18         app:layout_constraintBottom_toBottomOf="parent"
19         app:layout_constraintEnd_toStartOf="@+id/guideline3"
20         app:layout_constraintStart_toStartOf="parent" />
21      <androidx.constraintlayout.widget.Guideline
22          android:id="@+id/guideline3"
23          android:layout_width="wrap_content"
24          android:layout_height="wrap_content"
25          android:orientation="vertical"
26          app:layout_constraintGuide_percent="0.5" />
27      <Button
```

```

28     android:id="@+id/btn_refresh"
29     android:layout_width="0dp"
30     android:layout_height="wrap_content"
31     android:layout_marginStart="8dp"
32     android:layout_marginEnd="8dp"
33     android:onClick="actionRefresh"
34     android:text="Refresh"
35     app:layout_constraintEnd_toEndOf="parent"
36     app:layout_constraintStart_toStartOf="@+id/guideline3"
37     app:layout_constraintTop_toTopOf="@+id/btn_close"/>
38   <ListView
39     android:id="@+id/lv_user"
40     android:layout_width="0dp"
41     android:layout_height="0dp"
42     android:layout_marginStart="8dp"
43     android:layout_marginTop="8dp"
44     android:layout_marginEnd="8dp"
45     android:layout_marginBottom="16dp"
46     app:layout_constraintBottom_toTopOf="@+id/btn_close"
47     app:layout_constraintEnd_toEndOf="parent"
48     app:layout_constraintStart_toStartOf="parent"
49     app:layout_constraintTop_toTopOf="parent" />
50 </androidx.constraintlayout.widget.ConstraintLayout>

```

Implementasi GET pada Volley

- Buatlah sebuah method di dalam **VolleyActivity** untuk melakukan request data dari server dengan metode HTTP GET seperti dibawah ini.

```

65     private void getUserFromAPI() {
66         Gson gson = new Gson();
67         String URL = pref.getString(GlobalVariable.BASE_URL, null)
68             + "/volley/User_Registration.php";
69
70         ProgressDialog proDialog = new ProgressDialog( context: this);
71         proDialog.setTitle("Volley");
72         proDialog.setMessage("Silahkan tunggu");
73         proDialog.show();
74         JsonObjectRequest request = new JsonObjectRequest(Request.Method.GET, URL,
75             jsonRequest: null,
76             new Response.Listener<JSONObject>() {
77                 @Override
78                 public void onResponse(JSONObject response) {
79                     proDialog.dismiss();
80                     if (response != null) {
81                         UserResponse userResponse = gson.fromJson(response.toString(),
82                             UserResponse.class);
83                         if (userResponse.getCode() == 200) {
84                             UserAdapter adapter = new UserAdapter(getApplicationContext(),
85                             userResponse.getUser_list());
86                             lvUserVolley.setAdapter(adapter);
87                             lvUserVolley.setOnItemClickListener(
88                                 new AdapterView.OnItemClickListener() {

```

```

89                                     @Override
90                                     public void onItemClick(AdapterView<?> adapterView, View view,
91                                         int i, long l) {
92                                         Toast.makeText(getApplicationContext(),
93                                         userResponse.getUser_list().get(i).getUser_fullname(),
94                                         Toast.LENGTH_SHORT).show();
95                                     }
96                                 });
97                             });
98                         });
99                     },
100                 new Response.ErrorListener() {
101                     @Override
102                     public void onErrorResponse(VolleyError error) {
103                         proDialog.dismiss();
104                         Toast.makeText(getApplicationContext(), text: "Volley Error : " + error.getMessage(),
105                                         Toast.LENGTH_SHORT).show();
106                         Log.e(TAG, msg: "Error : " + error.getMessage());
107                     }
108                 });
109             }
110             RequestQueue requestQueue = Volley.newRequestQueue(getApplicationContext());
111             requestQueue.add(request);
112         }

```

2. Panggil fungsi tersebut pada **onCreate** seperti pada kode di bawah ini:

```

32     public class VolleyActivity extends AppCompatActivity {
33         private final String TAG = getClass().getSimpleName();
34         private ListView lvUserVolley;
35         private SharedPreferences pref;
36
37         @Override
38         protected void onCreate(Bundle savedInstanceState) {
39             super.onCreate(savedInstanceState);
40             setContentView(R.layout.activity_volley);
41             lvUserVolley = findViewById(R.id.lv_user_volley);
42             pref = getSharedPreferences(GlobalVariable.PREFERENCE_NAME, MODE_PRIVATE);
43             setTitle("Volley");
44             getUserFromAPI();
45         }
46         @Override
47         public boolean onCreateOptionsMenu(Menu menu) {
48             getMenuInflater().inflate(R.menu.retrofit, menu);
49             return super.onCreateOptionsMenu(menu);
50         }
51         @Override
52         public boolean onOptionsItemSelected(@NonNull MenuItem item) {
53             switch (item.getItemId()) {
54                 case R.id.action_add:
55                     Intent intent = new Intent( packageContext: this, AddUserActivity.class);
56                     intent.putExtra(GlobalVariable.TYPE_CONN, GlobalVariable.VOLLEY);
57                     startActivity(intent);
58                     break;
59             }
60         }

```

3. Panggil fungsi `getUserFromAPI()` pada `actionRefresh`, agar dapat dipanggil ketika ada data baru. Pada `actionClose` beri perintah untuk menutup **VolleyActivity**, seperti pada kode dibawah ini:

```
111     public void actionRefresh(View view) { getUserFromAPI(); }
114
115     public void actionClose(View view) { finish(); }
118 }
```

4. Buat UI dalam bentuk seperti dibawah ini. Simpan Layout tersebut dengan nama `activity_volley.xml`.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     xmlns:tools="http://schemas.android.com/tools"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     tools:context=".activity.VolleyActivity">
9     <androidx.constraintlayout.widget.Guideline
10        android:id="@+id/guideline4"
11        android:layout_width="wrap_content"
12        android:layout_height="wrap_content"
13        android:orientation="vertical"
14        app:layout_constraintGuide_percent="0.50121653" />
15     <Button
16        android:id="@+id/btn_volley_close"
17        android:layout_width="0dp"
18        android:layout_height="wrap_content"
19        android:layout_marginStart="8dp"
20        android:layout_marginEnd="8dp"
21        android:layout_marginBottom="16dp"
22        android:text="Close"
23        android:onClick="actionClose"
24        app:layout_constraintBottom_toBottomOf="parent"
25        app:layout_constraintEnd_toStartOf="@+id/guideline4"
26        app:layout_constraintStart_toStartOf="parent" />
27     <Button
28        android:id="@+id/btn_volley_refresh"
29        android:layout_width="0dp"
30        android:layout_height="wrap_content"
31        android:layout_marginStart="8dp"
32        android:layout_marginEnd="8dp"
33        android:text="Refresh"
34        android:onClick="actionRefresh"
35        app:layout_constraintBottom_toTopOf="@+id/btn_volley_close"
36        app:layout_constraintEnd_toEndOf="parent"
37        app:layout_constraintStart_toStartOf="@+id/guideline4"
38        app:layout_constraintTop_toTopOf="@+id/btn_volley_close" />
```

```

39     <ListView
40         android:id="@+id/lv_user_volley"
41         android:layout_width="0dp"
42         android:layout_height="0dp"
43         android:layout_marginStart="8dp"
44         android:layout_marginTop="8dp"
45         android:layout_marginEnd="8dp"
46         android:layout_marginBottom="16dp"
47         app:layout_constraintBottom_toTopOf="@+id/btn_volley_close"
48         app:layout_constraintEnd_toEndOf="parent"
49         app:layout_constraintStart_toStartOf="parent"
50         app:layout_constraintTop_toTopOf="parent" />
51     </androidx.constraintlayout.widget.ConstraintLayout>

```

f. Hasil dan Pembahasan

1. API untuk parsing data.
2. Project aplikasi dari percobaan implementasi code program untuk parsing data ke android dan menampilkan data dalam bentuk list view yang sudah di custom.
3. Dokumentasi hasil output aplikasi dalam bentuk laporan dilengkapi dengan analisa code pada implementasi source code.

g. Kesimpulan

1. Mahasiswa membuat program menyiapkan API.
2. Mahasiswa membuat program untuk menampilkan data dalam bentuk list view yang sudah di custom.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam menyiapkan API, implementasi code untuk menampilkan data dalam bentuk list view, serta menjelaskan secara detail setiap baris code	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam menyiapkan API, implementasi code untuk menampilkan data dalam bentuk list view, serta menjelaskan singkat setiap baris code	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam menyiapkan API, implementasi code untuk menampilkan data dalam bentuk list view, serta penjelasannya	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam menyiapkan API, implementasi code untuk menampilkan data dalam bentuk list view, tanpa penjelasan	(1)	2	3	4
Jumlah skor					

ACARA 44 : PENGENALAN RETROFIT, VOLLEY, GET, POST

Pokok Bahasan	: Sending Data (Post)
Acara Praktikum/Pertemuan	: Minggu 12 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu memahami konsep sending data (post).
2. Mahasiswa mampu menerapkan sending data (post) pada android.

b. Indikator

Keberhasilan mahasiswa memahami konsep sending data (post) dan menerapkan pada android.

c. Dasar Teori

Sending data

Sending menggunakan metode POST yang digunakan untuk mengirimkan atau membuat resource baru pada server. POST merupakan salah satu metode pada HTTP salah satu dari 4 metode HTTP Request yang umum diimplementasikan. Pada praktikum ini akan mengimplementasikan POST menggunakan Retrofit dan Volley menggunakan proyek **Retrovolley** yang telah dibuat pada pertemuan sebelumnya.

d. Alat dan Bahan

1. Kabel Data USB
2. Jaringan Internet

e. Prosedur Kerja

- POST merupakan metode HTTP digunakan untuk mengirimkan data pada web server.
 - Kemudian, web server akan menyimpan data ada DBMS sehingga bisa ditampilkan hasilnya menggunakan metode GET yang sudah dijelaskan diatas.
1. Kembali pada **AddUserActivity**.
 2. Buatlah fungsi untuk melakukan submit data metode POST seperti dibawah ini:

```

111     public void submitByRetrofit(User user){
112         ProgressDialog proDialog = new ProgressDialog( context: this);
113         proDialog.setTitle("Retrofit");
114         proDialog.setMessage("Sedang disubmit");
115         proDialog.show();
116
117         String globalURL = pref.getString(GlobalVariable.BASE_URL, s1: null);
118
119         Retrofit.Builder builder = new Retrofit.Builder()
120             .baseUrl(globalURL)
121             .addConverterFactory(GsonConverterFactory.create());
122
123         Retrofit retrofit = builder.build();
124         MethodHTTP client = retrofit.create(MethodHTTP.class);
125         Call<Request> call = client.sendUser(user);
126
127         call.enqueue(new Callback<Request>() {
128             @Override
129             public void onResponse(Call<Request> call, Response<Request> response) {
130                 proDialog.dismiss();
131                 if (response.body() != null) {
132                     if (response.body().getCode() == 201) {
133                         Toast.makeText(getApplicationContext(),
134                             text: "Response : "+response.body().getStatus(),
135                             Toast.LENGTH_SHORT).show();
136                         finish();
137                     } else if (response.body().getCode() == 406) {
138                         Toast.makeText(getApplicationContext(),
139                         Toast.makeText(getApplicationContext(),
140                             text: "Response : "+response.body().getStatus(),
141                             Toast.LENGTH_SHORT).show();
142                         edtEmail.requestFocus();
143                     } else {
144                         Toast.makeText(getApplicationContext(),
145                             text: "Response : "+response.body().getStatus(),
146                             Toast.LENGTH_SHORT).show();
147                         finish();
148                     }
149                 } else {
150                     Toast.makeText(getApplicationContext(),
151                         text: "Data Kosong!",Toast.LENGTH_SHORT).show();
152                 }
153                 Log.e(TAG, msg: "Error : "+response.message());
154             }
155             @Override
156             public void onFailure(Call<Request> call, Throwable t) {
157                 proDialog.dismiss();
158                 Log.e(TAG, msg: "Error POST Retrofit : "+t.getMessage());
159             });
160         });

```

Implementasi Post pada Volley

POST diimplementasikan dengan cara mengirimkan objek user pada webserver menggunakan AddUserActivity. Apabila sebelumnya menggunakan *library* retrofit, maka kali ini akan ditambahkan *library* Volley.

1. Buatlah fungsi untuk melakukan submit data pada submitbyVolley pada AddUserActivity seperti berikut:

```
162     public void submitByVolley(User user){
163         Gson gson = new Gson();
164         String URL = pref.getString(GlobalVariable.BASE_URL, null)
165             + "/volley/User_Registration.php";
166
167         ProgressDialog proDialog = new ProgressDialog(context: this);
168         proDialog.setTitle("Volley");
169         proDialog.setMessage("Sedang disubmit");
170         proDialog.show();
171
172         String userRequest = gson.toJson(user);
173         RequestQueue requestQueue = Volley.newRequestQueue(getApplicationContext());
174         JsonObjectRequest request = new JsonObjectRequest(com.android.volley.Request.Method.POST,
175             URL, jsonRequest: null,
176             new com.android.volley.Response.Listener<JSONObject>() {
177                 @Override
178                 public void onResponse(JSONObject response) {
179                     proDialog.dismiss();
180                     if (response != null) {
181                         Request requestFormat = gson.fromJson(response.toString(), Request.class);
182                         if (requestFormat.getCode() == 201) {
183                             Toast.makeText(getApplicationContext(),
184                                 text: "Response : " + requestFormat.getStatus(),
185                                 Toast.LENGTH_SHORT).show();
186                             finish();
187                         } else if (requestFormat.getCode() == 406) {
188                             Toast.makeText(getApplicationContext(),
189                                 text: "Response : " + requestFormat.getStatus(),
190                                 Toast.LENGTH_SHORT).show();
191                         } else {
192                             Toast.makeText(getApplicationContext(),
193                                 text: "Response : " + requestFormat.getStatus(),
194                                 Toast.LENGTH_SHORT).show();
195                             finish();
196                         }
197                     }
198                 },
199             new com.android.volley.Response.ErrorListener() {
```

```

209
210     @Override
211     public void onErrorResponse(VolleyError error) {
212         proDialog.dismiss();
213         Log.e(TAG, "Error POST Volley : "+error.getMessage());
214     }
215 }

216     @Override
217     public byte[] getBody() {
218         //return super.getBody();
219         return userRequest.getBytes();
220     }
221 }

222     requestQueue.add(request);
223     requestQueue.start();
224 }

225 }

```

Implementasi Update dan Delete

- Pertama buat activity baru dengan nama **UpdateUserActivity**
 - Buatlah 3 **EditText** sebagai input data Nama Lengkap, Email, dan Password.
 - ID tidak perlu ditambahkan karena sudah *auto increment* oleh DBMS secara langsung.
 - Tambahkan **Button** untuk *update* dan *delete* data pada web server. Perhatikan **activity_update_user.xml** dibawah ini:

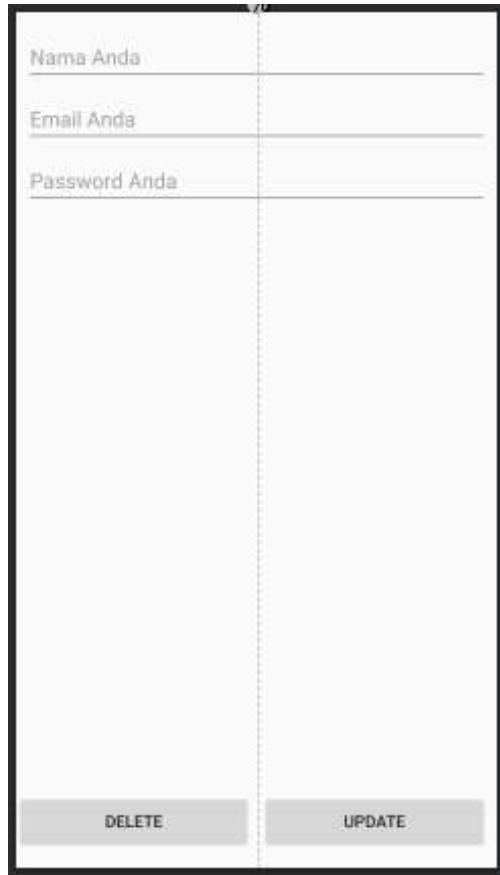
```

1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".activity.UpdateUserActivity">
9      <EditText
10         android:id="@+id/edt_username"
11         android:layout_width="0dp"
12         android:layout_height="wrap_content"
13         android:layout_marginStart="8dp"
14         android:layout_marginTop="16dp"
15         android:layout_marginEnd="8dp"
16         android:ems="18"
17         android:hint="Nama Anda"
18         android:inputType="textPersonName"
19         app:layout_constraintEnd_toEndOf="parent"
20         app:layout_constraintStart_toStartOf="parent"
21         app:layout_constraintTop_toTopOf="parent" />

```

```
49    android:layout_marginEnd="8dp"
50    android:layout_marginBottom="16dp"
51    android:onClick="actionUpdate"
52    android:text="Update"
53    app:layout_constraintBottom_toBottomOf="parent"
54    app:layout_constraintEnd_toEndOf="parent"
55    app:layout_constraintStart_toStartOf="@+id/guideline8" />
56    <androidx.constraintlayout.widget.Guideline
57        android:id="@+id/guideline8"
58        android:layout_width="wrap_content"
59        android:layout_height="wrap_content"
60        android:orientation="vertical"
61        app:layout_constraintGuide_percent="0.5" />
62    <Button
63        android:id="@+id/btn_delete"
64        android:layout_width="0dp"
65        android:layout_height="wrap_content"
66        android:layout_marginEnd="4dp"
67        android:text="Delete"
68        android:onClick="actionDelete"
69        app:layout_constraintBottom_toBottomOf="@+id/btn_update"
70        app:layout_constraintEnd_toStartOf="@+id/guideline8"
71        app:layout_constraintStart_toStartOf="parent"
72        app:layout_constraintTop_toTopOf="@+id/btn_update" />
73    </androidx.constraintlayout.widget.ConstraintLayout>
74    <EditText
75        android:id="@+id/edt_user_email"
76        android:layout_width="0dp"
77        android:layout_height="wrap_content"
78        android:layout_marginTop="8dp"
79        android:ems="10"
80        android:hint="Email Anda"
81        android:inputType="textEmailAddress"
82        app:layout_constraintEnd_toEndOf="@+id/edt_username"
83        app:layout_constraintStart_toStartOf="@+id/edt_username"
84        app:layout_constraintTop_toBottomOf="@+id/edt_username" />
85    <EditText
86        android:id="@+id/edt_password_update"
87        android:layout_width="0dp"
88        android:layout_height="wrap_content"
89        android:layout_marginTop="8dp"
90        android:ems="10"
91        android:hint="Password Anda"
92        android:inputType="textPassword"
93        app:layout_constraintEnd_toEndOf="@+id/edt_user_email"
94        app:layout_constraintStart_toStartOf="@+id/edt_user_email"
95        app:layout_constraintTop_toBottomOf="@+id/edt_user_email" />
96    <Button
97        android:id="@+id/btn_update"
98        android:layout_width="0dp"
99        android:layout_height="wrap_content"
100       android:layout_marginStart="4dp"
```

2. Hasil XML yang ditampilkan menjadi seperti ini:



3. Melengkapi kode UpdateUserActivity seperti dibawah ini:

```
26 public class UpdateUserActivity extends AppCompatActivity {
27     private User user;
28     private SharedPreferences pref;
29     private EditText edtUsername, edtEmail, edtPassword;
30     private final String TAG = getClass().getSimpleName();
31     @Override
32     protected void onCreate(Bundle savedInstanceState) {
33         super.onCreate(savedInstanceState);
34         setContentView(R.layout.activity_update_user);
35         edtUsername = findViewById(R.id.edt_username);
36         edtEmail = findViewById(R.id.edt_user_email);
37         edtPassword = findViewById(R.id.edt_password_update);
38         setTitle("Update Data User");
39         Bundle extras = getIntent().getExtras();
40         if (extras != null) {
41             user = new User();
42             user.setId(extras.getInt(GlobalVariable.CURRENT_USER_ID));
43
44             edtUsername.setText(extras.getString(GlobalVariable.CURRENT_USERNAME));
45             edtEmail.setText(extras.getString(GlobalVariable.CURRENT_USER_EMAIL));
46             edtPassword.setText(extras.getString(GlobalVariable.CURRENT_USER_PASSWORD));
```

```

47     }
48     pref = getSharedPreferences(GlobalVariable.PREFERENCE_NAME, MODE_PRIVATE);
49 }
50
51 public void actionUpdate(View view) {
52     boolean isValidated = false;
53     if (edtUsername.getText().toString().isEmpty()) {
54         edtUsername.setError("Tidak boleh kosong");
55         edtUsername.requestFocus();
56         isValidated = false;
57     } else
58         isValidated = true;
59     if (edtEmail.getText().toString().isEmpty()) {
60         edtEmail.setError("Tidak boleh kosong");
61         edtEmail.requestFocus();
62         isValidated = false;
63     } else
64         isValidated = true;
65     if (edtPassword.getText().toString().isEmpty()) {
66         edtPassword.setError("Tidak boleh kosong");
67         edtPassword.requestFocus();
68         isValidated = false;
69     } else
70         isValidated = true;
71     if (isValidated) {
72         user.setUser_fullname(edtUsername.getText().toString());
73         user.setUser_email(edtEmail.getText().toString());
74         user.setUser_password(edtPassword.getText().toString());
75         ProgressDialog proDialog = new ProgressDialog(context: this);
76         proDialog.setTitle("Retrofit");
77         proDialog.setMessage("Sedang disubmit");
78         proDialog.show();
79         String globalURL = pref.getString(GlobalVariable.BASE_URL, s1: null);
80
81         Retrofit.Builder builder = new Retrofit.Builder()
82             .baseUrl(globalURL)
83             .addConverterFactory(GsonConverterFactory.create());
84         Retrofit retrofit = builder.build();
85         MethodHTTP client = retrofit.create(MethodHTTP.class);
86         Call<Request> call = client.updateUser(user);
87         call.enqueue(new Callback<Request>() {
88             @Override
89             public void onResponse(Call<Request> call, Response<Request> response) {
90                 proDialog.dismiss();
91                 if (response.body() != null) {
92                     if (response.body().getCode() == 200) {
93                         Toast.makeText(getApplicationContext(),
94                             text: "Response : "+response.body().getStatus(),
95                             Toast.LENGTH_SHORT).show();
96                         finish();
97                     } else {

```

```

98         Toast.makeText(getApplicationContext(),
99             text: "Response : "+response.body().getStatus(),
100            Toast.LENGTH_SHORT).show();
101        finish();
102    }
103}
104Log.e(TAG, msg: "Error : "+response.message());
105}
106
107@Override
108public void onFailure(Call<Request> call, Throwable t) {
109    proDialog.dismiss();
110    Log.e(TAG, msg: "Error Update: "+t.getMessage());
111}
112});
113}
114}
115
116public void actionDelete(View view) {
117    int currentUserID = pref.getInt(GlobalVariable.CURRENT_USER_ID, 0);
118    if (currentUserID == user.getId()){
119        new AlertDialog.Builder( context: this)
120            .setTitle("Peringatan!")
121            .setMessage("Tidak dapat menghapus pengguna yang sedang login!")
122            .setPositiveButton( text: "OK", new DialogInterface.OnClickListener() {
123                @Override
124                public void onClick(DialogInterface dialogInterface, int i) {
125                    dialogInterface.dismiss();
126                }
127            }).show();
128    } else {
129        new AlertDialog.Builder( context: this)
130            .setTitle("Peringatan!")
131            .setMessage("Yakin hapus data?")
132            .setPositiveButton( text: "OK", new DialogInterface.OnClickListener() {
133                @Override
134                public void onClick(DialogInterface dialogInterface, int i) {
135                    deleteUserByID(user.getId());
136                }
137            }).show();
138            setNegativeButton( text: "Batal", new DialogInterface.OnClickListener() {
139                @Override
140                public void onClick(DialogInterface dialogInterface, int i) {
141                    dialogInterface.dismiss();
142                }
143            }).show();
144    }
145}

```

```

147     public void deleteUserByID (int id) {
148         ProgressDialog proDialog = new ProgressDialog( context: this);
149         proDialog.setTitle("Retrofit");
150         proDialog.setMessage("Silahkan tunggu");
151         proDialog.show();
152         String globalURL = pref.getString(GlobalVariable.BASE_URL, s1: null);
153         Retrofit.Builder builder = new Retrofit.Builder()
154             .baseUrl(globalURL)
155             .addConverterFactory(GsonConverterFactory.create());
156         Retrofit retrofit = builder.build();
157         MethodHTTP client = retrofit.create(MethodHTTP.class);
158         Call<Request> call = client.deleteUser(id);
159         call.enqueue(new Callback<Request>() {
160             @Override
161             public void onResponse(Call<Request> call, Response<Request> response) {
162                 proDialog.dismiss();
163                 if (response.body() != null) {
164                     if (response.body().getCode() == 200) {
165                         Toast.makeText(getApplicationContext(),
166                             text: "Response : "+response.body().getStatus(),
167                             Toast.LENGTH_SHORT).show();
168                         finish();
169                     } else {
170                         Toast.makeText(getApplicationContext(),
171                             text: "Response : "+response.body().getStatus(),
172                             Toast.LENGTH_SHORT).show();
173                         finish();
174                     }
175                 }
176                 Log.e(TAG, msg: "Error : "+response.message());
177             }
178
179             @Override
180             public void onFailure(Call<Request> call, Throwable t) { Log.d(TAG, t.toString()); }
181         });
182     }
183 }
184
185 }
```

4. Build dan Run proyek tersebut, kemudian analisa hasilnya.

f. Hasil dan Pembahasan

1. Project aplikasi dari percobaan implementasi code program sending data (post).
2. Dokumentasi hasil output aplikasi dalam bentuk laporan dilengkapi dengan analisa code pada implementasi source code.

g. Kesimpulan

Mahasiswa memahami konsep sending data (post) dan menerapkan pada android.

h. Rubrik Penilaian

No	Indikator	Skor*			
1	Ketepatan waktu dan ketepatan dalam implementasi code program sending data (post), serta menjelaskan secara detail setiap baris code	1	2	3	(4)
2	Ketepatan waktu dan ketepatan dalam implementasi code program sending data (post), serta penjelasan singkat setiap baris code	1	2	(3)	4
3	Ketepatan waktu akan tetapi kurang tepat dalam implementasi code program sending data (post), serta penjelasan setiap baris code	1	(2)	3	4
4	Keterlambatan pengumpulan tugas dan ketidaktepatan dalam implementasi code program sending data (post), serta tanpa penjelasan	(1)	2	3	4
Jumlah skor					

ACARA 45 : PROJECT FINAL

Pokok Bahasan	: Laporan Studi Kasus
Acara Praktikum/Pertemuan	: Minggu 13 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu menemukan studi kasus yang akan diselesaikan.
2. Mahasiswa mampu mengumpulkan informasi.
3. Mahasiswa mampu menemukan masalah inti dari studi kasus.
4. Mahasiswa mampu menemukan ide untuk solusi dari permasalahan.

b. Indikator

1. Keberhasilan menemukan studi kasus yang akan diselesaikan.
2. Keberhasilan mahasiswa mengumpulkan informasi dan menemukan masalah inti dari studi kasus.
3. Keberhasilan menemukan ide untuk solusi dari permasalahan.

c. Dasar Teori

Design Thinking

Design Thinking adalah proses berulang dimana kita berusaha memahami pengguna, menantang asumsi, dan mendefinisikan kembali masalah dalam upaya mengidentifikasi strategi dan solusi alternatif yang mungkin tidak langsung terlihat dengan tingkat awal pemahaman kita. Pada saat yang sama, Design Thinking menyediakan pendekatan berbasis solusi untuk menyelesaikan masalah. Ini adalah cara berpikir dan bekerja serta kumpulan metode langsung.

Design Thinking berputar di sekitar minat yang mendalam dalam mengembangkan pemahaman dari orang-orang yang menjadi tujuan perancangan produk atau layanan. Hal ini membantu kita mengamati dan mengembangkan empati dengan target pengguna. Design Thinking membantu kita dalam proses bertanya: mempertanyakan masalah, mempertanyakan asumsi, dan mempertanyakan keterkaitannya.

Design Thinking sangat berguna dalam mengatasi masalah-masalah yang tidak jelas atau tidak dikenal, dengan melakukan reframing masalah dengan cara-cara yang berpusat pada manusia, menciptakan banyak ide dalam brainstorming, dan mengadopsi pendekatan langsung dalam pembuatan prototype dan testing. Design Thinking juga melibatkan eksperimen yang sedang berjalan: membuat sketsa, membuat prototype, testing, dan mencoba berbagai konsep dan ide.

Tahapan dalam Proses Design Thinking

Empathise



Tahap pertama dari proses Design Thinking adalah untuk mendapatkan pemahaman empatik tentang masalah yang dicoba untuk diselesaikan. Ini melibatkan para ahli konsultasi untuk mencari tahu lebih banyak tentang bidang yang menjadi perhatian melalui pengamatan, keterlibatan, dan empati dengan orang-orang untuk memahami pengalaman dan motivasi mereka sehingga memperoleh pemahaman pribadi yang lebih jelas tentang masalah yang terlibat. Empati sangat penting untuk proses desain yang berpusat pada manusia seperti Design Thinking, dan empati memungkinkan pemikir desain untuk menggesampingkan asumsi mereka sendiri tentang dunia untuk mendapatkan wawasan tentang pengguna dan kebutuhan mereka.

Define

Selama tahap Define, kita mengumpulkan informasi yang telah kita buat dan kumpulkan selama tahap Empathise. Disinilah kita akan menganalisis pengamatan dan mensistesisnya untuk menentukan masalah inti yang telah diidentifikasi. Kita harus berusaha menidentifikasi masalah sebagai pernyataan masalah dengan cara yang berpusat pada manusia.



Sebagai ilustrasi, alih-alih mengidentifikasi masalah sebagai keinginan atau kebutuhan perusahaan seperti, “Kita perlu meningkatkan pangsa pasar produk makanan diantara remaja perempuan sebesar 5%,” cara yang lebih baik untuk mendefinisikan masalah adalah jadilah, “Gadis remaja perlu makan makanan bergizi agar dapat berkembang, menjadi sehat dan tumbuh.”

Tahap Define akan membantu para desainer dalam sebuah tim untuk mengumpulkan ide-ide hebat untuk membangun fitur, fungsi, dan elemen lain yang akan memungkinkan mereka untuk menyelesaikan masalah atau, paling tidak, memungkinkan pengguna untuk menyelesaikan masalah sendiri dengan tingkat kesulitan minimal.

Ideate



Selama tahap ketiga dari proses Design Thinking, desainer siap untuk mulai menghasilkan ide. Kita telah tumbuh untuk memahami pengguna dan kebutuhan mereka di tahap Empathize, dan kita telah menganalisis dan mensistesis pengamatan Anda di tahap Define, dan berakhir dengan pernyataan masalah yang berpusat pada manusia. Dengan latar belakang

yang kuat, kita dan anggota tim dapat mulai “berpikir di luar kotak” untuk mengidentifikasi solusi baru untuk pernyataan masalah yang dibuat, dan kita dapat mulai mencari cara alternatif untuk melihat masalah.

Ada ratusan teknik Ideation seperti Brainstorm, Brainwrite, Worst Possible Idea, dan SCRAMPER. Sesi Brainstorm dan Worst Possible Idea biasanya digunakan untuk merangsang pemikiran bebas dan untuk memperluas ruang masalah. Penting untuk mendapatkan sebanyak mungkin ide atau solusi masalah. Kita harus memilih beberapa teknik Ideation lainnya pada akhir fase Ideation untuk membantu kita menyelidiki dan menguji ide-ide kita sehingga kita dapat menemukan cara terbaik untuk memecahkan masalah atau menyediakan elemen-elemen yang diperlukan untuk menghindarinya.

<https://sis.binus.ac.id/2020/03/17/design-thinking-pengertian-tahapan-dan-contoh-penerapannya/>

d. Alat dan Bahan

1. Kertas A4
2. Kertas Folio
3. Sticky Note
4. Spidol

e. Prosedur Kerja

1. Mahasiswa berdiskusi dengan kelompok untuk mencari studi kasus dilingkungan sekitar dan menggali informasi serta temukan inti masalah dari setudi kasus.
2. Dari studi kasus yang sudah ditemukan setiap kelompok menganalisa permasalahan inti yang terdapat dalam studi kasus yang ditemukan.
3. Setiap kelompok menyusun pertanyaan untuk pengumpulan data.
4. Setiap kelompok mempresentasikan didepan kelas.
5. Kelompok lain mengoreksi dan mengomentari dari hasil ide yang dikemukakan.

f. Hasil dan Pembahasan

1. Dokumen hasil diskusi kelompok berupa beberapa studi kasus yang diajukan setiap anggota kelompok beserta alasannya sampai dengan hasil pilihan studi kasus yang dipilih, serta beberapa pertanyaan yang dipersiapkan untuk pengumpulan data.
2. Dokumen laporan untuk disusun dalam laporan akhir.

g. Kesimpulan

Mahasiswa menemukan studi kasus yang akan diselesaikan dari hasil diskusi dalam 1 kelompok, dan susunan pertanyaan yang dipersiapkan untuk pengumpulan data

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- [1] Kelengkapan (25%)
- [2] Kesesuaian (25%)
- [3] Kedalaman materi (25%)
- [4] Kreativitas (25%)

ACARA 46 : PROJECT FINAL

Pokok Bahasan	: Laporan Hasil Observasi
Acara Praktikum/Pertemuan	: Minggu 13 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu menemukan studi kasus yang akan diselesaikan.
2. Mahasiswa mampu mengumpulkan informasi.
3. Mahasiswa mampu menemukan masalah inti dari studi kasus.
4. Mahasiswa mampu menemukan ide untuk solusi dari permasalahan.

b. Indikator

1. Keberhasilan menemukan studi kasus yang akan diselesaikan.
2. Keberhasilan mahasiswa mengumpulkan informasi dan menemukan masalah inti dari studi kasus.
3. Keberhasilan menemukan ide untuk solusi dari permasalahan.

c. Dasar Teori

Proses Bisnis

Definisi Proses Bisnis menurut ahli:

- 1) **Weske, 2007:** Proses bisnis adalah serangkaian instrumen untuk mengorganisir suatu kegiatan dan untuk meningkatkan pemahaman atas keterkaitan suatu kegiatan.
- 2) **Sparx System, 2004:** adalah sekumpulan kegiatan atau aktifitas yang dirancang untuk menghasilkan suatu keluaran tertentu bagi pelanggan tertentu.
- 3) **Hammer dan Champy dalam Weske, 2007:** proses bisnis adalah sekumpulan kegiatan yang mengambil salah satu atau banyak masukan dan menciptakan sebuah keluaran yang berguna bagi pelanggan.

Contoh :

- Penerimaan Karyawan oleh HRD
- Penjualan Toko oleh Kasir

- Pendaftaran Mahasiswa oleh Staff Administrasi
- Peminjaman Buku oleh Mahasiswa

Detail Contoh:

- Proses bisnis peminjaman buku dapat diuraikan sebagai berikut:
 - Memasuki Perpustakaan
 - Menujukkan ID Member
 - Memilih Buku
 - Membawa buku ke pustakawan
 - Memasukkan ID Member dan kode buku
 - Melakukan transaksi peminjaman

Teknik Pengumpulan data

- 1) **Wawancara** : merupakan suatu teknik pengumpulan data dengan jalan mengadakan komunikasi dengan sumber data. Komunikasi tersebut dilakukan dengan dialog (Tanya jawab) secara lisan, baik langsung maupun tidak langsung
- 2) **Observasi** : pengamatan langsung pada lingkungannya dan atau pengamatan langsung suatu kegiatan yang akan dibuat sistemnya.
- 3) **Kuesioner dan Literatur** : Teknik yang dilakukan dengan memberikan kuesioner terhadap user yang telah dipilih sedangkan literatur dengan melakukan pencarian terhadap berbagai sumber tertulis, baik berupa buku-buku, arsip, majalah, artikel, dan jurnal, atau dokumen-dokumen yang relevan dengan permasalahan yang dikaji. Sehingga informasi yang didapat dari kuesioner dan studi kepustakaan ini dijadikan rujukan untuk memperkuat system yang akan dibuat.

d. Alat dan Bahan

1. Kertas A4
2. Kertas Folio
3. Sticky Note
4. Spidol

e. Prosedur Kerja

Seluruh anggota kelompok melakukan pengambilan data dan menggali informasi kepada object, dalam bentuk : (1) observasi; (2) wawancara; (3) kuisioner.

1. Diskusikan dengan masing-masing kelompok hasil pengumpulan data.
2. Uraikan setiap informasi yang didapatkan dengan bahasa hasil analisa.
3. Uraikan proses bisnis dari object studi kasus.
4. Temukan beberapa masalah yang *urgent* dan butuh penyelesaian.
5. Tentukan masalah yang akan diberikan solusi dari hasil diskusi analisa masalah.

f. Hasil dan Pembahasan

1. Dokumen hasil pengumpulan data proses bisnis, permasalahan yang terdapat dalam object serta solusi yang diberikan terhadap masalah yang *urgent*.
2. Dokumen laporan untuk disusun dalam laporan akhir.

g. Kesimpulan

Mahasiswa menemukan studi kasus yang akan diselesaikan dari hasil diskusi dalam 1 kelompok.

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- | | |
|----------------------|-------|
| [1] Kelengkapan | (25%) |
| [2] Kesesuaian | (25%) |
| [3] Kedalaman materi | (25%) |
| [4] Kreativitas | (25%) |

ACARA 47 : PROJECT FINAL

Pokok Bahasan	: Laporan Prototype
Acara Praktikum/Pertemuan	: Minggu 13 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

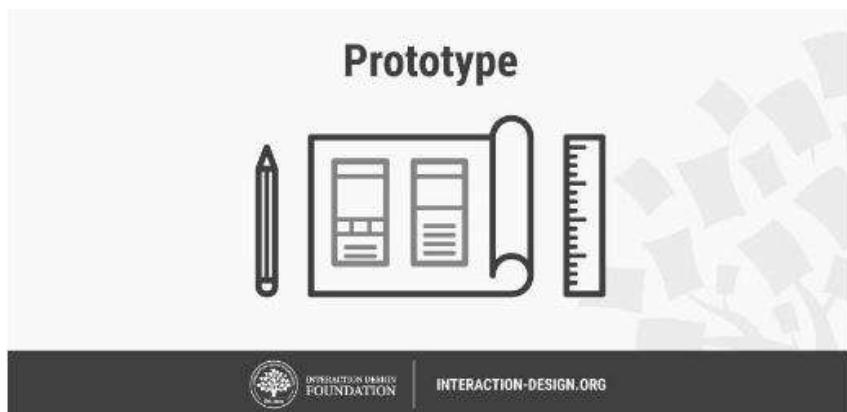
1. Mahasiswa mampu merancang prototype untuk project yang akan dikerjakan.
2. Mahasiswa mampu melakukan validate terhadap prototype.

b. Indikator

Keberhasilan mahasiswa merancang prototype untuk project dan melakukan validate.

c. Dasar Teori

Prototype



Tim desain akan menghasilkan sejumlah versi produk yang murah dan diperkecil atau fitur spesifik yang ditemukan dalam produk, sehingga mereka dapat menyelidiki solusi masalah yang dihasilkan pada tahap sebelumnya. Prototype dapat dibagikan dan diuji dalam tim itu sendiri, di departemen lain, atau pada sekelompok kecil orang diluar tim desain. Ini adalah fase eksperimental, dan tujuannya adalah untuk mengidentifikasi solusi terbaik untuk setiap masalah yang diidentifikasi selama tiga tahap pertama. Solusi diimplementasikan dalam prototype, dan satu per satu, mereka diselidiki dan diterima, diperbaiki dan diperiksa ulang, dan ditolak berdasarkan pengalaman pengguna.

Pada akhir tahap ini, tim desain akan memiliki gagasan yang lebih baik tentang kendala yang melekat pada produk dan masalah yang ada, dan memiliki pandangan yang lebih jelas tentang bagaimana pengguna yang sebenarnya akan berperilaku, berpikir, dan rasakan ketika berinteraksi dengan bagian akhir produk.

Test



Desainer menguji produk lengkap secara ketat menggunakan solusi terbaik yang diidentifikasi selama fase prototyping. Ini adalah tahap akhir dari design thinking, tetapi dalam proses berulang, hasil yang dihasilkan selama fase testing sering digunakan untuk mendefinikan kembali satu atau lebih masalah dan menginformasi pemahaman pengguna, kondisi penggunaan, bagaimana orang berpikir, berperilaku, dan merasakan, dan berempati. Bahkan selama fase ini, perubahan dan penyempurnaan dilakukan untuk menyingkirkan solusi masalah dan memperoleh pemahaman sedalam mungkin terhadap produk dan penggunanya.

<https://sis.binus.ac.id/2020/03/17/design-thinking-pengertian-tahapan-dan-contoh-penerapannya/>

d. Alat dan Bahan

1. Kertas A4
2. Kertas Folio
3. Sticky Note
4. Spidol
5. Penggaris

e. Prosedur Kerja

1. Berdiskusi dengan masing-masing kelompok untuk merancang prototype project.
2. Membuat sketsa prototype untuk setiap form project yang akan dikerjakan.
3. Implementasikan sketsa ke dalam software untuk menghasilkan prototype yang mudah dipahami.

f. Hasil dan Pembahasan

1. Dokumen hasil pembuatan prototype project akhir menggunakan software yang telah dipilih.
2. Dokumen laporan untuk disusun dalam laporan akhir.

g. Kesimpulan

Mahasiswa dalam 1 kelompok menyelesaikan prototype project akhir yang telah dikerjakan.

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- | | |
|----------------------|-------|
| [1] Kelengkapan | (25%) |
| [2] Kesesuaian | (25%) |
| [3] Kedalaman materi | (25%) |
| [4] Kreativitas | (25%) |

ACARA 48 : PROJECT FINAL

Pokok Bahasan	: Laporan Validasi Prototype
Acara Praktikum/Pertemuan	: Minggu 13 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu melakukan validasi prototype project yang telah dikerjakan.
2. Mahasiswa mampu memperbaiki hasil validasi prototype project.

b. Indikator

Keberhasilan mahasiswa melakukan validasi prototype project dan memperbaikinya.

c. Dasar Teori

-

d. Alat dan Bahan

1. Kertas A4
2. Kertas Folio
3. Sticky Note
4. Spidol

e. Prosedur Kerja

1. Diskusi dengan anggota kelompok untuk melakukan validasi prototype.
2. Melakukan validasi kepada object studi kasus.
3. Diskusi hasil validasi dan memperbaiki prototype jika terdapat ketidak sesuaian dengan keinginan object.
4. Mempresentasikan didepan kelas hasil prototype dan hasil perbaikannya.

f. Hasil dan Pembahasan

Dokumentasi hasil validasi prototype dan hasil perbaikan prototype untuk disusun dalam laporan akhir.

g. Kesimpulan

Mahasiswa berhasil melakukan validasi dan perbaikan prototype.

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- [1] Kelengkapan (25%)
- [2] Kesesuaian (25%)
- [3] Kedalaman materi (25%)
- [4] Kreativitas (25%)

ACARA 49 : PROJECT FINAL

Pokok Bahasan	: Laporan Usecase Diagram
Acara Praktikum/Pertemuan	: Minggu 14 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu membuat usecase secara kelompok dari studi kasus project akhir.
2. Mahasiswa mampu merancang database secara kelompok dari studi kasus project akhir.

b. Indikator

Keberhasilan mahasiswa membuat usecase, merancang dan menyusun database secara berkelompok dari studi kasus project akhir.

c. Dasar Teori

UML Diagram : Use Case Diagram

Use case diagram merupakan diagram yang menggambarkan hubungan antara aktor dengan sistem. Use case diagram bisa mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Use case diagram juga bisa digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan bisa juga mempresentasikan sebuah interaksi aktor dengan sistem. Komponen tersebut kemudian menjelaskan komunikasi antara aktor, dengan sistem yang ada. Dengan demikian, use case dapat dipresentasikan dengan urutan yang sederhana, dan akan mudah dipahami oleh para konsumen. Manfaat dari use case sendiri adalah untuk memudahkan komunikasi dengan menggunakan domain expert dan juga end user, memberikan kepastian pemahaman yang pas tentang requirement atau juga kebutuhan sebuah sistem.

Use case diagram mempunyai 3 komponen ,yaitu :

Sistem

Menyatakan batasan sistem dalam relasi dengan aktor-aktor yang menggunakan (di luar sistem) dan fitur-fitur yang harus disediakan (dalam sistem).

Aktor

Aktor adalah segala hal diluar sistem yang akan menggunakan sistem tersebut untuk melakukan sesuatu. Bisa merupakan manusia, sistem, atau device yang memiliki peranan dalam keberhasilan operasi dari sistem.

Use Case

Use Case sendiri adalah gambaran fungsional dari sebuah sistem. Dengan demikian, antara konsumen dan juga pengguna pada sistem tersebut, akan mengerti atau paham mengenai fungsi sistem yang tengah dibangun.

Use case diagram juga mempunyai beberapa relasi, yaitu:

Association

Teknik mengidentifikasi interaksi yang dilakukan oleh actor tertentu dengan use case tertentu. Hal ini digambarkan dengan garis antara aktor terhadap use case tersebut.

Generalization

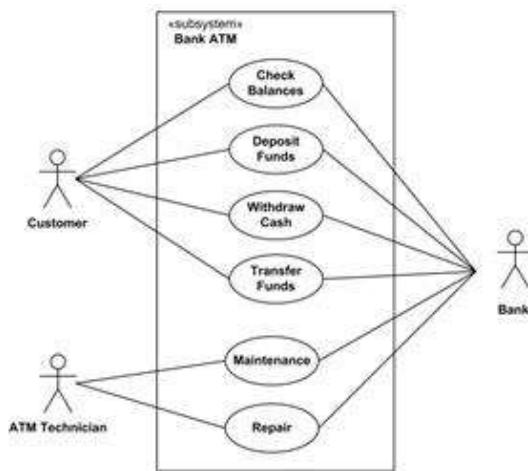
Mendefinisikan relasi antara dua aktor atau dua use case yang mana salah satunya meng-inherit dan menambahkan atau override sifat dari yang lainnya.

Dependency

Dependency ini terbagi menjadi 2 macam, yaitu include dan juga extend.

- Include : Berfungsi untuk mengidentifikasi hubungan antara 2 use case, dimana use case yang satu akan memanggil use case yang lainnya.
- Extend : Apabila pemanggilan, memerlukan kondisi tertentu maka akan berlaku dependensi.

Contoh Use Case Diagram :



An example of use case diagram for Bank ATM subsystem - top level use cases.

<https://socs.binus.ac.id/2019/11/26/uml-diagram-use-case-diagram/>

d. Alat dan Bahan

1. Kertas A4
2. Kertas Folio
3. Sticky Note
4. Spidol
5. Penggaris

e. Prosedur Kerja

Diskusikan dengan kelompok untuk membuat use case project akhir, dan mulai menyusun kedalam laporan akhir.

f. Hasil dan Pembahasan

Dokumentasi use case diagram untuk disusun dalam laporan akhir.

g. Kesimpulan

Mahasiswa membuat use case diagram untuk disusun kedalam project akhir.

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- | | |
|----------------------|-------|
| [1] Kelengkapan | (25%) |
| [2] Kesesuaian | (25%) |
| [3] Kedalaman materi | (25%) |
| [4] Kreativitas | (25%) |

ACARA 50 : PROJECT FINAL

Pokok Bahasan	: Laporan Database
Acara Praktikum/Pertemuan	: Minggu 14 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu membuat usecase secara kelompok dari studi kasus project akhir.
2. Mahasiswa mampu merancang database secara kelompok dari studi kasus project akhir.

b. Indikator

Keberhasilan mahasiswa membuat usecase, merancang dan menyusun database secara berkelompok dari studi kasus project akhir.

c. Dasar Teori

Deinisi Database

Database atau yang dalam bahasa indonesianya dikatakan sebagai basis data/basisdata adalah sekumpulan data yang berupa informasi yang saling berhubungan antara satu dengan lainnya yang disimpan didalam komputer secara sistematis sehingga dapat diakses untuk digunakan kembali.

Pengguna dari database sendiri terbagi menjadi 2 jenis, yaitu database administrator dan database user. Database administrator atau biasa disingkat menjadi DBA berwenang dalam pengawasan database. Selain itu, DBA juga memiliki fungsi untuk mendefinisikan pola struktur database serta memodifikasi pola dan organisasi fisik. Sementara database user adalah pengguna dari database tersebut.

<https://sis.binus.ac.id/2018/12/18/database-secara-umum/>

Conceptual Design Database

Conceptual Design Database adalah tahapan pertama dalam desain database. Tujuan dari tahapan ini adalah untuk merancang database yang independen dari database software dan detail fisik. Tahapan ini akan menghasilkan, conceptual data model yang menjelaskan entitas data, atribut, hubungan antara tabel, dan constraints di suatu database. Design ini bersifat deskriptif dan naratif.

Semua elemen dari data yang dibutuhkan dalam suatu transaksi database harus dijabarkan di modelnya, dan semua elemen data yang ada harus digunakan setidaknya satu kali di suatu transaksi database itu. Conceptual database memiliki 4 tahapan yaitu:

1. Analisis dan persyaratan data
2. Normalisasi dan perancangan hubungan antar entitas
3. Verifikasi data model
4. Distributed database design

<https://sis.binus.ac.id/2020/07/07/database-design-conceptual-design/>

d. Alat dan Bahan

1. Kertas A4
2. Kertas Folio
3. Sticky Note
4. Spidol
5. Penggaris

e. Prosedur Kerja

Diskusikan dengan kelompok untuk membuat perancangan database project akhir dan mengimplementasikan hasil perancangan, dan mulai menyusun kedalam laporan akhir.

f. Hasil dan Pembahasan

Dokumentasi perancangan dan implementasi database untuk disusun dalam laporan akhir.

g. Kesimpulan

Mahasiswa membuat perancangan dan implementasi database untuk disusun kedalam project akhir.

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- | | |
|----------------------|-------|
| [1] Kelengkapan | (25%) |
| [2] Kesesuaian | (25%) |
| [3] Kedalaman materi | (25%) |
| [4] Kreativitas | (25%) |

ACARA 51 : PROJECT FINAL

Pokok Bahasan	: Progres Implementasi Project
Acara Praktikum/Pertemuan	: Minggu 14 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu meng-implementasi-kan aplikasi berbasis android dari kasus tertentu.
2. Mahasiswa mampu menerapkan konsep dasar android yang telah dipelajari.
3. Mahasiswa mampu melaporkan pencapaian progress dari project yang dikerjakan.
4. Mahasiswa mampu mempresentasikan progress masing-masing kelompok.

b. Indikator

Keberhasilan mahasiswa dalam kelompok meng-implementasi-kan aplikasi berbasis android dari kasus tertentu.

c. Dasar Teori

-

d. Alat dan Bahan

1. Laptop
2. Jaringan Internet
3. Kabel Data USB

e. Prosedur Kerja

1. Diskusikan dengan kelompok untuk membuat implementasi aplikasi berbasis android dari kasus tertentu.
2. Push setiap kegiatan dalam membangun aplikasi pada akun GitHub.
3. Invite seluruh anggota kelompok dan dosen pengampu pada GitHub.
4. Melakukan laporan pencapaian dari project yang dikerjakan.
5. Menyampaikan target untuk monitoring selanjutnya.
6. Presentasi progress.

f. Hasil dan Pembahasan

Dokumentasi implementasi project akhir dalam membangun aplikasi pada akun GitHub.

g. Kesimpulan

Mahasiswa setiap kelompok dalam pencapaian progress dari project yang dikerjakan yang di push dalam GitHub.

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- [1] Kelengkapan (25%)
- [2] Kesesuaian (25%)
- [3] Kedalaman materi (25%)
- [4] Kreativitas (25%)

ACARA 52 : PROJECT FINAL

Pokok Bahasan	: Dokumentasi Project
Acara Praktikum/Pertemuan	: Minggu 14 / 4
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

1. Mahasiswa mampu menyusun laporan akhir untuk project akhir.
2. Mahasiswa mampu mendokumentasikan aplikasi dalam bentuk video.

b. Indikator

Keberhasilan mahasiswa setiap kelompok membuat dokumentasi final project berupa laporan dan video.

c. Dasar Teori

-

d. Alat dan Bahan

1. Laptop
2. Jaringan Internet
3. Kabel Data USB
4. Perangkat keras pendukung pengambilan video
5. Perangkat lunak pendukung pengeditan video

e. Prosedur Kerja

Diskusikan dengan kelompok untuk membuat sebuah video dokumentasi pada aplikasi yang telah dibuat, dan dokumentasi dalam bentuk laporan akhir dan manual book.

f. Hasil dan Pembahasan

1. Video dokumentasi penggunaan aplikasi dengan durasi maksimal 5 menit.
2. Laporan akhir dan manual book.

g. Kesimpulan

Mahasiswa setiap kelompok membuat dokumentasi final project berupa video, laporan dan manual book.

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- [1] Kelengkapan (25%)
- [2] Kesesuaian (25%)
- [3] Kedalaman materi (25%)
- [4] Kreativitas (25%)

ACARA 53 : PROJECT FINAL

Pokok Bahasan	: Presentasi Project Akhir
Acara Praktikum/Pertemuan	: Minggu 15 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

Mahasiswa mampu mempresentasikan hasil projectnya.

b. Indikator

Keberhasilan mahasiswa menjelaskan hasil final project, dan penguasaan materi.

c. Dasar Teori

-

d. Alat dan Bahan

1. Laptop
2. LCD proyektor

e. Prosedur Kerja

1. Secara bergantian presentasikan hasil penggeraan project, setiap sesi 30 menit.
2. Lanjutkan dengan diskusi dengan dosen penguji.

f. Hasil dan Pembahasan

Setiap masukan dari dosen penguji menjadi saran perbaikan project yang dikerjakan.

g. Kesimpulan

Presentasi adalah proses pemaparan hasil penyelesaian final project.

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- | | |
|----------------------|-------|
| [1] Kelengkapan | (25%) |
| [2] Kesesuaian | (25%) |
| [3] Kedalaman materi | (25%) |
| [4] Kreativitas | (25%) |

ACARA 54 : PROJECT FINAL

Pokok Bahasan	: Presentasi Project Akhir
Acara Praktikum/Pertemuan	: Minggu 15 / 2
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

Mahasiswa mampu mempresentasikan hasil projectnya.

b. Indikator

Keberhasilan mahasiswa menjelaskan hasil final project, dan penguasaan materi.

c. Dasar Teori

-

d. Alat dan Bahan

1. Laptop
2. LCD proyektor

e. Prosedur Kerja

1. Secara bergantian presentasikan hasil penggeraan project, setiap sesi 30 menit.
2. Lanjutkan dengan diskusi dengan dosen penguji.

f. Hasil dan Pembahasan

Setiap masukan dari dosen penguji menjadi saran perbaikan project yang dikerjakan.

g. Kesimpulan

Presentasi adalah proses pemaparan hasil penyelesaian final project.

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- | | |
|----------------------|-------|
| [1] Kelengkapan | (25%) |
| [2] Kesesuaian | (25%) |
| [3] Kedalaman materi | (25%) |
| [4] Kreativitas | (25%) |

ACARA 55 : PROJECT FINAL

Pokok Bahasan	: Presentasi Project Akhir
Acara Praktikum/Pertemuan	: Minggu 15 / 3
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

Mahasiswa mampu mempresentasikan hasil projectnya.

b. Indikator

Keberhasilan mahasiswa menjelaskan hasil final project, dan penguasaan materi.

c. Dasar Teori

-

d. Alat dan Bahan

1. Laptop
2. LCD proyektor

e. Prosedur Kerja

1. Secara bergantian presentasikan hasil penggeraan project, setiap sesi 30 menit.
2. Lanjutkan dengan diskusi dengan dosen penguji.

f. Hasil dan Pembahasan

Setiap masukan dari dosen penguji menjadi saran perbaikan project yang dikerjakan.

g. Kesimpulan

Presentasi adalah proses pemaparan hasil penyelesaian final project.

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- | | |
|----------------------|-------|
| [1] Kelengkapan | (25%) |
| [2] Kesesuaian | (25%) |
| [3] Kedalaman materi | (25%) |
| [4] Kreativitas | (25%) |

ACARA 54 : PROJECT FINAL

Pokok Bahasan	: Presentasi Project Akhir
Acara Praktikum/Pertemuan	: Minggu 15 / 1
Tempat	: Daring / Luring Politeknik Negeri Jember
Alokasi Waktu	: 100 menit

a. Capaian Pembelajaran Mata Kuliah (CPMK)

Mahasiswa mampu mempresentasikan hasil projectnya.

b. Indikator

Keberhasilan mahasiswa menjelaskan hasil final project, dan penguasaan materi.

c. Dasar Teori

-

d. Alat dan Bahan

1. Laptop
2. LCD proyektor

e. Prosedur Kerja

1. Secara bergantian presentasikan hasil penggerjaan project, setiap sesi 30 menit.
2. Lanjutkan dengan diskusi dengan dosen penguji.

f. Hasil dan Pembahasan

Setiap masukan dari dosen penguji menjadi saran perbaikan project yang dikerjakan.

g. Kesimpulan

Presentasi adalah proses pemaparan hasil penyelesaian final project.

h. Rubrik Penilaian

Penilaian laporan hasil diskusi:

- | | |
|----------------------|-------|
| [1] Kelengkapan | (25%) |
| [2] Kesesuaian | (25%) |
| [3] Kedalaman materi | (25%) |
| [4] Kreativitas | (25%) |