TUTORIAL PEMBUATANPROTOTIPE PREDIKSI KETINGGIAN AIR (PKA) DAN AUGMENTED REALITY BERBASIS IOT VERSI 2



Penulis:

Lalita Chandiany Adiputri Mohamad Nurkamal Fauzan Noviana Riza

TUTORIAL PEMBUATAN PROTOTIPE PREDIKSI KETINGGIAN AIR (PKA) DAN AUGMENTED REALITY BERBASIS IOT VERSI 2

TUTORIAL PEMBUATAN PROTOTIPE PREDIKSI KETINGGIAN AIR (PKA) DAN AUGMENTED REALITY BERBASIS IOT VERSI 2

Lalita Chandiany Adiputri Mohamad Nurkamal Fauzan Noviana Riza Informatics Research Center



Kreatif Industri Nusantara

Judul:

Tutorial Pembuatan Protipe Prediksi Ketinggian Air (PKA) Dan Augmented Reality Berbasis IoT Versi 2

Penulis:

Lalita Chandiany Adiputri Mohamad Nurkamal Fauzan Noviana Riza

ISBN: 978-623-7898-60-3

Editor:

Rolly Maulana Awangga

Penyunting:

Roni Habibi

Desain sampul dan Tata letak:

Lalita Chandiany Adiputri

Penerbit:

Kreatif Industri Nusantara

Redaksi:

Jl. Ligar Nyawang No. 2 Bandung 40191 Tel. 022 2045-8529

Email: awangga@kreatif.co.id

Distributor:

Informatics Research Center Jl. Sariasih No. 54 Bandung 40151 Email: irc@poltekpos.ac.id

Cetakan Pertama, 2020

Hak cipta dilindungi undang-undang Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara apapun tanpa ijin tertulis dari penerbit

'Jika Kamu tidak dapat menahan lelahnya belajar, Maka kamu harus sanggup menahan perihnya Kebodohan.' Imam Syafi'i

CO	Γ	ГО	IDI	IT			C
	IVI	ıĸ	IMI	JI	()	н	Τ,

LALITA CHANDIANY ADIPUTRI, MOHAMAD NURKAMAL FAUZAN, NOVIANA RIZA

Informatics Research Center., Politeknik Pos Indonesia, Bandung, Indonesia

CONTENTS IN BRIEF

1	Sekilas Tentang Prototipe dan Augmented Reality		1
2	Software, Komponen, dan Bahasa Pemrograman Yang Digunakar	1	į
3	Pembuatan Prototipe	5	7
4	Pembuatan Aplikasi Monitoring Air	11	7
5	Pembuatan Augmented Reality Prototipe PKA	15	7

DAFTAR ISI

Daftar Gan	nbar		xiii
Daftar Tabe	el		xxiii
Foreword			xxvii
Kata Penga	ıntar		xxix
Acknowled	lgments		xxxi
Acronyms			xxxiii
Glossary			xxxv
List of Syn	nbols		xxxvii
Introductio	n		xxxix
1 Seki	ilas Tent	tang Prototipe dan Augmented Reality	1
1.1	Sejaral	h <i>Prototype</i>	1
1.2	Tujuan	n Prototype	2
1.3	Augme	ented Reality	2
	1.3.1	Pengertian Augmented Reality	2
	1.3.2	Cara Kerja Augmented Reality	2
			iv

1.3.3	Jenis T	eknologi	AU	emented	Reali	tν
1.0.0	JUILD I	CINITOTOSI	110 %	, III CI II CU	ILCUIT	, y

3

2	Soft	ware,Ko	omponen,dan Bahasa Pemrograman Yang Digu	ınakan (
	2.1	Softwa	re Yang Digunakan	5
		2.1.1	Arduino IDE	5
		2.1.2	Bagian-Bagian Arduino IDE	6
		2.1.3	Unity	8
		2.1.4	Fitur Dalam Unity	9
		2.1.5	Bot Telegram	15
		2.1.6	Antares	16
		2.1.7	Software Membuat 3D Objek	17
		2.1.8	Vuforia	23
		2.1.9	Software Perakitan Komponen	24
	2.2	Kompo	onen Yang Digunakan	31
	2.3	Bahasa	a pemrograman Yang Digunakan	49
		2.3.1	Pengertian Bahasa pemrograman C	49
		2.3.2	Kelebihan dan Kekurangan Bahasa C	50
		2.3.3	Sketch Arduino	50
3	Pem	buatan	Prototipe	57
	3.1	Memb	uat Kerangka Prototipe	57
	3.2	Memb	uat Skema Elektronik Komponen	61
	3.3	Impler	mentasi Coding	68
		3.3.1	Implementasi Coding Pada Arduino IDE	69
		3.3.2	Cara <i>Import Library</i>	71
		3.3.3	Membuat Program Prototipe PKA	74
		3.3.4	Menghubungkan Prototipe PKA Pada Antares	82
		3.3.5	Menambahkan Code Notifikasi Telegram Prototipe	PKA 107
4	Pem	buatan	Aplikasi Monitoring Air	117
	4.1	Pembu	atan Aplikasi Monitoring PKA	118
		4.1.1	Pembuatan Splash Screen	118
		4.1.2	Pembuatan Halaman Menu	124
		4.1.3	Pembuatan Menu About	138
		4.1.4	Pembuatan Menu Ketinggian Air	141
		4.1.5	Pembuatan Menu Keterangan Air	148
		4.1.6	Pembuatan Menu Don't Panic	149
	4.2	Tampi	lan Aplikasi Monitoring	151

				DAFTAR ISI	хi
5	Pem	buatan .	Augmented Reality Prototipe PKA		157
	5.1	Install	Unity 3D		158
	5.2	Memb	uat Augmented Reality PKA		163
		5.2.1	Membuat Database Marker		163
		5.2.2	Membuat Augmented Reality PKA Pada Unit	.y	170
Dafta	ar Pust	aka			201

DAFTAR GAMBAR

1.1	Marker	3
2.1	Arduino IDE	6
2.2	Tampilan Arduino IDE	7
2.3	Software Unity	8
2.4	Tampilan Halaman Asset Store	10
2.5	Tampilan Console Pada Unity	11
2.6	Tampilan Build Setting Pada Unity	11
2.7	Tampilan Player Setting Pada Unity	12
2.8	Tampilan Resolution and Presentation Pada Unity	13
2.9	Tampilan Splash Image Pada Unity	14
2.10	Tampilan XR Settings Pada Unity	15
2.11	Logo Telegram	15
2.12	Logo Antares	16
		xiii

xiv	DAFTAR GAMBAR

2.13	Tampilan Antares Antares	16
2.14	Logo Firebase	17
2.15	sketchup Make	18
2.16	Tampilan sketchup Make	18
2.17	DesignWorkshop Lite	18
2.18	Tampilan DesignWorkshop Lite	19
2.19	Blender	19
2.20	Tampilan Blender	20
2.21	Free CAD	20
2.22	Tampilan Free CAD	21
2.23	TinkerCAD	21
2.24	Tampilan TinkerCAD	22
2.25	OpenSCAD	22
2.26	Tampilan OpenSCAD	23
2.27	Logo Vuforia	23
2.28	Tampilan Frizting	24
2.29	Tampilan Frizting	25
2.30	Tampilan Frizting	25
2.31	Tampilan LiveWire	26
2.32	Tampilan Electronics Workbench (EWB)	26
2.33	Tampilan Electronics Workbench (EWB)	27
2.34	Tampilan Electronics Workbench (EWB)	27
2.35	Tampilan NI Multisim	28
2.36	Tampilan NI Multisim	28
2.37	Tampilan NI Multisim	29
2.38	Tampilan Autodesk Circuits	29
2.39	Tampilan Autodesk Circuits	29
2.40	Tampilan Autodesk Circuits	30

		DAF IAR GAIVIDAR	ΑV
2.41	Tampilan Proteus ISIS		30
2.42	Tampilan Proteus ISIS		31
2.43	Tampilan Proteus ISIS		31
2.44	NodeMCU		32
2.45	Spesifikasi NodeMCU		33
2.46	Skema Posisi Pin NodeMCU		33
2.47	Gambar Sensor Ultrasonic US 100		34
2.48	Prinsip Kerja Sensor Ultrasonic		35
2.49	Rumus perhitungan Jarak Sensor Ultrasonic		36
2.50	Pin Konfigurasi		37
2.51	Skema diagram sensor ultrasonic US 100		37
2.52	Buzzer		38
2.53	Konfigurasi Pin Buzzer		38
2.54	Keterangan Konfigurasi Pin Buzzer		39
2.55	Diagram model 2D Buzzer		40
2.56	Led		40
2.57	Simbol Led		40
2.58	resistor		41
2.59	Macam-Macam Resistor Sesuai Warna		42
2.60	Resistor Yang Digunakan		43
2.61	Kabel Jumper		43
2.62	Kabel Jumper female		44
2.63	Kabel Jumper Male		44
2.64	Kabel Jumper Male-female		45
2.65	Breadboard Mini		45
2.66	Breadboard Medium		46
2.67	breadboard Large		46
2.68	Kardus		46

xvi	DAFTAR GAMBAR

2.69	Kertas Warna	47
2.70	Lem Kayu	47
2.71	Wadah atau Ember	48
2.72	Kayu	48
2.73	Cat Kayu	49
2.74	Contoh Void Setup	51
2.75	Gambar Contoh Fungsi Setup dan Loop	51
3.1	Tampilan website QR Code Generator	58
3.2	Memilih type QR code	59
3.3	Membuat QR code	59
3.4	Mengunduh QR code	60
3.5	Hasil unduh QR code	60
3.6	Tampilan Keseluruhan Prototipe	61
3.7	Tampilan Keseluruhan Prototipe	61
3.8	Tampilan Awal Frizting	62
3.9	Ubah Ukuran Breadboad	62
3.10	Memasukan NodeMCU	63
3.11	Sambungkan Vin ke Breadboard	63
3.12	Mengubah Warna Wire	64
3.13	Sambungkan GND ke Breadboard	64
3.14	Menambahkan Komponen Sensor Ultrasonic	65
3.15	Menambahkan Komponen Led, Resistor dan Buzzer	65
3.16	Rangkaian Alat	66
3.17	Tampilan Prototipe	67
3.18	Tampilan Arduino IDE Blink	69
3.19	Tampilan Di Serial Monitor	70
3.20	Memilih Menu Manage Libraries	71
3.21	Tampilan Sketch Manager	72
	2.70 2.71 2.72 2.73 2.74 2.75 3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8 3.9 3.10 3.11 3.12 3.13 3.14 3.15 3.16 3.17 3.18 3.19 3.20	2.70 Lem Kayu 2.71 Wadah atau Ember 2.72 Kayu 2.73 Cat Kayu 2.74 Contoh Void Setup 2.75 Gambar Contoh Fungsi Setup dan Loop 3.1 Tampilan website QR Code Generator 3.2 Memilih type QR code 3.3 Membuat QR code 3.4 Mengunduh QR code 3.5 Hasil unduh QR code 3.6 Tampilan Keseluruhan Prototipe 3.7 Tampilan Keseluruhan Prototipe 3.8 Tampilan Keseluruhan Prototipe 3.9 Ubah Ukuran Breadboad 3.10 Memasukan NodeMCU 3.11 Sambungkan Vin ke Breadboard 3.12 Mengubah Warna Wire 3.13 Sambungkan GND ke Breadboard 3.14 Menambahkan Komponen Sensor Ultrasonic 3.15 Menambahkan Komponen Led, Resistor dan Buzzer 3.16 Rangkaian Alat 3.17 Tampilan Prototipe 3.18 Tampilan Di Serial Monitor 3.20 Memilih Menu Manage Libraries

		DAFTAR GAMBAR	xvii
3.22	Tampilan Library Yang Terinstall		72
3.23	Tampilan Library ADD ZIP		73
3.24	Pilih File ZIP		73
3.25	Tampilan Library Berhasil Ditambahkan		74
3.26	Tampilan serial monitor		76
3.27	Halaman Awal Antares		82
3.28	Tampilan Registrasi Antares		83
3.29	Verifikasi email		83
3.30	Tampilan Awal Antares		84
3.31	Halaman Overview		84
3.32	Halaman application		85
3.33	Halaman Widgets		85
3.34	Halaman Documentation		86
3.35	Halaman Documentation		86
3.36	Halaman Documentation Tutorial		87
3.37	Halaman DocumentationTutorial		87
3.38	Halaman Account		88
3.39	Halaman End User		88
3.40	Download Library		89
3.41	Download Library di Github		89
3.42	Download Library		90
3.43	Tampilan di Github		90
3.44	Tampilan content		91
3.45	Tampilan Using Boards Manager		91
3.46	Tampilan Arduino IDE		92
3.47	Tampilan Arduino IDE		92
3.48	Tampilan Preferences		93
3.49	Tampilan Arduino IDE Boards Mnager		93

xviii	DAFTAR GAMBAR	
3.50	Tampilan Boards Manager	94
3.51	Tampilan Boards Manager Antares	94
3.52	Tampilan Arduino IDE Include Library	95
3.53	Tampilan Library Manager	95
3.54	Tampilan Library Manager Antares	96
3.55	Tampilan Arduino IDE Manage Library	96
3.56	Tampilan Arduino IDE Library Antares	97
3.57	Tampilan Awal Antares	98
3.58	Tampilan Application Antares	98
3.59	Tampilan Membuat Application Antares	98
3.60	Tampilan Application Yang Dibuat	99
3.61	Tampilan Membuat Device Antares	99
3.62	Tampilan Membuat Device Antares	100
3.63	Tampilan Device Di Application Antares	100
3.64	Tampilan Halaman Device Antares	100
3.65	Tampilan Antares	102
3.66	Tampilan Users Antares	102

105

106

106

107

107

108

108

109

109

110

110

Upload Arduino IDE

Tampilan Application

Tampilan Data Sensor Ultrasonic

Tampilan Project

Tampilan Telegram

Tampilan Bot Telegram

Tampilan Bot Telegram

Tampilan Bot Telegram

Tampilan ID Bot Telegram

Tampilan ID Bot Telegram

Tampilan ID Bot Telegram

3.67

3.68

3.69

3.70

3.71

3.72

3.73

3.74

3.75

3.76

3.77

		DAFTAR GAMBAR	XIX
3.78	Tampilan Bot Monitoring Di Telegram		111
3.79	Tampilan Download CTbot		111
3.80	Tampilan Download Arduino Json		111
3.81	Tampilan Notifikasi DI Telegram		116
4.1	Tampilan Awal Android Studio		119
4.2	Tampilan Memilih Activity		119
4.3	Membuat Activity splash Activity		120
4.4	Konfigurasi Project		121
4.5	Konfigurasi Project		121
4.6	Proses Gradle		121
4.7	Build Gradle		122
4.8	Workspace		122
4.9	Drawable		123
4.10	Library Slider		124
4.11	Activity Baru		125
4.12	Tampilan Slider		126
4.13	Library Cardview		127
4.14	Library Antares		141
4.15	Cara Masuk ke Folder Libs		141
4.16	Tampilan Folder Libs		142
4.17	Tampilan Aplikasi DI Emulator		152
4.18	Splash Screen		152
4.19	Tampilan Menu Utama		153
4.20	Tampilan Menu About		153
4.21	Tampilan Ketinggian Air		154
4.22	Tampilan Menu Keterangan Air		154
4.23	Tampilan Menu Don't Panic		155
4.24	Tampilan Notifikasi Telegram		155

CO, CIVIDA CI C
R GAMBAR

5.1	Tampilan Halaman Download Unity Hub	158
5.2	Unity Hub. Exe	158
5.3	Tampilan Awal Unity Hub	158
5.4	Sign In Unity Hub	159
5.5	Sign In Unity Hub	159
5.6	Tampilan Sesudah Login Unity Hub	160
5.7	Manage License	160
5.8	Tampilan Manage License	161
5.9	Tampilan License Activation	161
5.10	Tampilan Menu Install	162
5.11	Tampilan Menu Install	162
5.12	Tampilan Memilih Modul	163
5.13	Tampilan vuforia	163
5.14	Tampilan Registrasi Vuforia	164
5.15	Tampilan Login Vuforia	164
5.16	Tampilan Halaman Awal Vuforia	165
5.17	Development Key Vuforia	165
5.18	Development Key Vuforia	166
5.19	License Key Vuforia	166
5.20	Database Vuforia	167
5.21	Create Database Vuforia	167
5.22	Database Vuforia	168
5.23	Menambahkan Target Vuforia	168
5.24	Add Target Vuforia	169
5.25	Marker atau Target Vuforia	169
5.26	Marker atau Target Yang Ditambahkan	170
5.27	Tampilan Unity Hub	170
5.28	Tampilan Create New Project	171

		DAFTAR GAMBAR	ххі
5.29	Templates 3D		171
5.30	Project Name dan Location		171
5.31	Project Name dan Location		172
5.32	Mengatur XR Setting		172
5.33	Mengatur XR Setting		173
5.34	Inspector Player Setting		173
5.35	Inspector Player Setting		174
5.36	Download Database Marker		174
5.37	Download Database Marker		175
5.38	Download SDK Vuforia Engine		175
5.39	Download SDK Vuforia Engine		176
5.40	Import Package		176
5.41	Import Package SDK Vuforia		177
5.42	Import Package Database		177
5.43	Import Package Database		178
5.44	Delete Main Camera		178
5.45	Menambahkan AR Camera		179
5.46	Import AR Camera		179
5.47	Menambahkan Image Target		179
5.48	Image Target Marker		180
5.49	Membuat Folder		180
5.50	Folder Yang Sudah Dibuat		181
5.51	Menambahkan 3D Objek ke dalam Folder		181
5.52	Menambahkan 3D Objek ke dalam Image Target		181
5.53	Menambahkan Material		182
5.54	Menambahkan Warna Material		182
5.55	Menambahkan Warna Material		183
5.56	Menambahkan Warna Material Pada 3D Objek		183

xxii	DAFTAR GAMBAR			
5.57	Mengatur Ukuran 3D Objek	183		
5.58	Mengatur Posisi 3D objek	184		
5.59	Posisi 3D Objek Rumah	184		
5.60	Menambahkan 3D Objek Cube	184		
5.61	Mengatur Ukuran dan Posisi 3D Cube	185		
5.62	Posisi dan Ukuran 3D Cube	185		
5.63	Asset Store	186		
5.64	Maximize Asset Store	186		
5.65	Mendownload Texture	186		
5.66	Download Texture	187		
5.67	Import Texture	187		
5.68	Folder Texture	188		
5.69	Folder Texture	188		
5.70	Memberi Warna 3D Cube	189		
5.71	3D Object Cube Diberi Texture	189		
5.72	Membuat Folder Pohon, Jembatan dan Railing	190		
5.73	3D Object Jembatan	190		
5.74	3D Object Pohon	191		
5.75	3D Object Railing	191		
5.76	Menambahkan 3D Object Railing	192		
5.77	Menambahkan 3D Object Jembatan	192		
5.78	Menambahkan 3D Object Pohon	193		
5.79	Menambahkan Audio Source	193		
5.80	Menambahkan Audio	194		
5.81	Edit Script	194		
5.82	Membuat Script C	198		

Menambahkan Script dan Mengatur Speed Rotate

Augmented Reality PKA

199

200

5.83

5.84

DAFTAR TABEL

Listings

src/void.c	52
src/boolean.c	52
src/char.c	52
src/unchar.c	52
src/byte.c	52
src/int.c	53
src/unint.c	53
src/long.c	53
src/unlong.c	53
src/short.c	53
src/float.c	53
src/double.c	54
src/variabel1.c	54
src/variabel2.c	54
src/hello.c	70
src/sensor.c	74
src/led.c	76
src/buzzer.c	77

XXVI LISTINGS

src/gabung.c	77
src/regresi.c	80
src/nodemcuantares.c	101
src/nodemcuantares2.c	101
src/nodemcuantares3.c	102
src/nodemcuantares4.c	103
src/nodemcuantares5.c	103
src/nodemcuantares6.c	103
src/nodemcuantares7.c	104
src/bot.c	112
src/bot2.c	112
src/bot3.c	112
src/bot4.c	112
src/bot5.c	112
src/bot6.c	113
src/botgabung.c	113
src/splash.java	123
src/tampilansplash.xml	124
src/slider.java	125
src/tampilanslider.xml	126
src/menu.java	127
src/homefull.java	128
src/tampilanmenu.xml	130
src/menufull.xml	134
src/about.java	138
src/aboutfull.xml	139
src/manifest.java	142
src/monitoring.java	142
src/tampilanmon.xml	145
src/air.java	148
src/tampilanair.xml	148
src/panic.java	149
src/tampilanpanic.xml	150
src/sound.cs	195
src/rotate.cs	199

FOREWORD

Sepatah kata dari Kaprodi, Pembimbing dan Mahasiswa

KATA PENGANTAR

Buku tutorial pembuatan prototipe PKA dan augmented reality berbasis IoT versi 2 ini diciptakan bagi yang ingin membuat sebuah prototipe dengan menggunakan *microcontroller* NodeMCU, monitoring melalui android serta adanya notifikasi melalui telegram dan augmented reality sebagai sebuah penjelasan mengenai prototipe.

Lalita Chandiany. A, M. Nurkamal .F, Noviana Riza

Bandung, Jawa Barat

Juli, 2020

ACKNOWLEDGMENTS

Terima kasih atas semua masukan dari para dosen pembimbing agar bisa membuat buku ini lebih baik dan lebih mudah dimengerti.

Terima kasih ini juga ditujukan khusus untuk team IRC yang telah fokus untuk belajar dan memahami bagaimana buku ini mendampingi proses Tugas Akhir.

L. C. A.

ACRONYMS

IoT Internet of Thing

IDE Integrated Development Environment

AR Augmented Reality

GLOSSARY

Prototype

Merupakan purwarupa model kerja dasar dari pengembangan sebuah program (software) atau perangkat lunak.

Arduino IDE

Merupakan (*Integrated Development Environment*) software yang di gunakan untuk memprogram di arduino.

Augmented Reality

Merupakan teknologi yang menggabungkan benda maya dua dimensi dan ataupun tiga dimensi ke dalam sebuah lingkungan nyata tiga dimensi lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata.

SYMBOLS

INTRODUCTION

Informatics Research Center Bandung, Jawa Barat, Indonesia

Indonesia sekarang sudah memasuki era industri 4.0 atau revolusi 4.0 dimana otomatisasi sistem produksi dengan memanfaatkan teknologi. Di dalam pabrik mulai menggunakan teknologi baru seperti IoT (internet of things).

SEKILAS TENTANG PROTOTIPE DAN AUGMENTED REALITY

1.1 Sejarah Prototype

Pada tahun 1960-an Herbert Voelcker, 1970: Voelcker mengembangkan alat dasar matematika yang dengan jelas menggambarkan tiga aspek dimensi dan menghasilkan teori-teori awal teori algoritma dan matematika untuk pemodelan solid.Pada tahun 1987, Carl Deckard, membentuk tim peneliti dari University of Texas. datang dengan ide yang revolusioner yang baik. *Prototype* adalah salah satu pendekatan dalam rekayasa perangkat lunak yang secara langsung mendemonstrasikan bagaimana sebuah perangkat lunak atau komponen-komponen perangkat lunak akan bekerja dalam lingkungannya sebelum tahapan konstruksi aktual dilakukan (Howard, 1997). [1]

Selain itu juga prototipe adalah tahapan yang ditujukan untuk mentransformasi sifat-sifat abstrak dari sebuah ide menjadi lebih berwujud. Tahapan ini tidak hanya berupa proses visualisasi ide tetapi juga proses pembangunan ide. Secara umum, Prototype memiliki dua kategori: low-fidelity dan high-fidelity. Proses prototyping yang digunakan di dalam Design Thinking adalah *low-fidelity* atau *Rapid Prototyping*. Proses ini menekankan kepada pembuatan proses pembuatan yang cepat, mudah, murah dan basic . [1]

1.2 Tujuan *Prototype*

prototype bertujuan untuk contoh atau model awal yang dibangun untuk menguji sebuah konsep atau proses atau aksi sebagai sesuatu yang digandakan atau dipelajarinya. Pengertian prototipe tidak selalu merujuk pada ukuran, artinya prototipe tidak selalu harus berukuran sama dengan produk yang akan dibuat. Prototipe bisa berukuran lebih kecil atau lebih besar dibanding dengan produk yang akan dibuat asalkan aksi atau proses yang akan terjadi sebenarnya.

Prototype dapat memberikan ide bagi pembuat dan pemakai potensial tentang cara sistem berfungsi dalam bentuk lengkapnya. Tujuannya adalah mengembangkan model menjadi sistem final. Artinya sistem akan dikembangkan lebih cepat daripada metode tradisional dan biayanya menjadi lebih murah. Selain hal tersebut pembuatan prototipe untuk perbaikan atau penyempurnaan rancangan.

1.3 Augmented Reality

1.3.1 Pengertian Augmented Reality

Augmented reality (AR) merupakan sebuah teknologi yang menggabungkan benda maya dua dimensi dan ataupun tiga dimensi ke dalam sebuah lingkungan nyata tiga dimensi lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata. Dimana teknologi yang memperluas dunia fisik dengan cara menambahkan lapisan informasi digital ke dalamnya. AR muncul dalam tampilan langsung dari lingkungan yang ada dan menambahkan suara, video, dan grafik ke dalamnya. Sehingga, AR adalah kenampakan lingkungan fisik dunia nyata, dibarengi dengan gambar yang dihasilkan komputer sehingga mengubah persepsi realitas.

1.3.2 Cara Kerja Augmented Reality

AR dapat ditampilkan pada berbagai perangkat seperti kacamata, layar, ponsel, dan sebagainya. Agar perangkat berfungsi dengan baik, sejumlah data tertentu dalam bentuk video, gambar, animasi, dan model 3D perlu digunakan.

Sehingga, orang bisa melihat hasilnya dalam cahaya buatan dan alami. AR menggunakan teknologi SLAM (Simultaneous Localization and Mapping), sensor, dan pengukur kedalaman. Misalnya, mengumpulkan data sensor untuk menghitung jarak dari lokasi sensor ke objek.

Berikut adalah beberapa komponen AR:

Kamera dan Sensor.

Kamera dan sensor digunakan untuk mengumpulkan informasi kolaborasi pengguna dan mengirimkannya untuk diproses. Kamera pada gadget memiliki kemampuan untuk memeriksa lingkungan dan dengan data tersebut, akan mampu menemukan barang fisik dan menghasilkan model 3D.

2. Proyeksi.

Komponen ini mengacu pada proyektor yang lebih kecil dari yang biasa ada pada headset AR, yang mengambil informasi dari sensor dan memproyeksikan konten yang terkomputerisasi ke permukaan untuk dilihat.Sebenarnya, pemanfaatan proyeksi di AR belum sepenuhnya dirancang untuk dapat digunakan dalam barang atau layanan komersial.

3. Refleksi

Beberapa gadget AR memiliki cermin untuk membantu mata manusia melihat gambar virtual.Beberapa darinya memiliki variasi cermin kecil yang ditekuk dan beberapa lagi memiliki cermin sisi ganda untuk memantulkan cahaya ke kamera dan mata pengguna.

Tujuan dari cara refleksi tersebut adalah untuk memainkan pengaturan gambar yang tepat.

1.3.3 Jenis Teknologi *AUgmented Reality*

Ada beberapa kategori teknologi augmented reality, masing-masing memiliki perbedaan dan kegunaan dalam pengaplikasiannya sebagai berikut :

1. Marker-Based Augmented Reality



Gambar 1.1 Marker

Marker-based AR menggunakan kamera dan beberapa jenis penanda visual, seperti kode QR/2D. Teknologi ini akan menghasilkan output hanya ketika mark er di scan.Aplikasi marker-based menggunakan kamera pada perangkat untuk membedakan marker dari objek dunia nyata lainnya.

Pola sederhana seperti kode QR digunakan sebagai marker karena dapat dengan mudah dikenali dan tidak memerlukan banyak effort untuk membaca.Posisi dan

4 SEKILAS TENTANG PROTOTIPE DAN AUGMENTED REALITY

orientasi juga dihitung, di mana beberapa jenis konten atau informasi kemudian dibebani banyak marker.

2. Markerless Augmented Reality

Sebagai salah satu aplikasi AR yang diimplementasikan secara luas, markerless augmented reality menggunakan GPS, kompas digital, pengukur kecepatan, atau akselerometer yang tertanam dalam perangkat untuk menyediakan data berdasarkan lokasi.

Teknologi markerless augmented reality adalah ketersediaan fitur pendeteksian lokasi pada smartphone.Ini paling umum digunakan untuk memetakan arah, menemukan bisnis terdekat, dan aplikasi seluler berbasis lokasi lainnya.

3. Projection Based Augmented Reality

Projection based AR bekerja dengan cara memproyeksikan cahaya buatan ke permukaan riil. Aplikasi teknologi ini memungkinkan interaksi manusia dengan mengirimkan cahaya ke permukaan riil dan kemudian merasakan interaksi manusia (sentuhan) dari cahaya yang diproyeksikan.

4. Superimposition Based Augmented Reality

Superimposition based AR mampu mengganti sebagian atau seluruh tampilan asli dari suatu objek dengan pandangan yang baru dan ditambah dari objek yang sama. Pendeteksian objek memainkan peran penting karena aplikasi tidak dapat menggantikan tampilan asli dengan augmented jika tidak dapat menentukan apa objek itu.

Contoh yang dihadapi konsumen akan augmented reality berbasis superimposisi dapat ditemukan dalam katalog furnitur augmented reality IKEA.Dengan mengunduh aplikasi dan memindai halaman yang dipilih dalam katalog cetak atau digital mereka, pengguna dapat menempatkan furnitur IKEA virtual di rumah mereka sendiri dengan bantuan augmented reality.

SOFTWARE,KOMPONEN,DAN BAHASA PEMROGRAMAN YANG DIGUNAKAN

Dalam membuat *prototype* PKA dan *augmented reality* ini dibutuhkan *software* , komponen dan bahasa pemrograman yang sesuai dengan kebutuhan. Agar pada saat implementasi tidak kebingungan dan menghasilkan *output* yang memuaskan. Berikut *software* , komponen dan bahasa pemrograman yang dibutuhkan.

2.1 Software Yang Digunakan

2.1.1 Arduino IDE

Arduino IDE (*Integrated Development Environment*) adalah software yang digunakan untuk memprogram di arduino, dengan kata lain Arduino IDE sebagai media untuk memprogram board Arduino. Arduino IDE bisa di download secara gratis di website resmi Arduino IDE. [2]

Arduino IDE ini berguna sebagai text editor untuk membuat, mengedit, dan juga memvalidasi kode program. bisa juga digunakan untuk meng-upload ke board Arduino. Kode program yang digunakan pada Arduino disebut dengan istilah Arduino sketch atau disebut juga source code arduino, dengan ekstensi file source code .ino

Arduino IDE dibuat dari bahasa pemrograman JAVA. Arduino IDE juga dilengkapi dengan library C/C++ yang biasa disebut Wiring yang membuat operasi input dan output menjadi lebih mudah. Arduino IDE ini dikembangkan dari software Processing yang dirombak menjadi Arduino IDE khusus untuk pemrograman dengan Arduino. [2]

Dalam *software* Arduino IDE ini dapat memprogram kode tidak hanya *microcontroller* arduino saja, tetapi pada *software* ini dapat digunakan untuk memprogram *microcontroller* yang lain salah satu contohnya yaitu Nodemcu.

2.1.2 Bagian-Bagian Arduino IDE



Gambar 2.1 Arduino IDE

Editor *Programming* pada umumnya memiliki fitur untuk *cut / paste* dan untuk *find / replace teks*, demikian juga pada Arduino IDE. Pada bagian keterangan aplikasi memberikan pesan balik saat menyimpan dan mengekspor serta sebagai tempat menampilkan kesalahan. console log menampilkan teks log dari aktivitas Arduino IDE, termasuk pesan kesalahan yang lengkap dan informasi lainnya. Pojok kanan bawah menampilkan *port* serial yang digunakan. Tombol *toolbar* terdapat ikon tombol pintas untuk memverifikasi dan meng-upload program, membuat, membuka, dan menyimpan *sketch*, dan membuka monitor serial. Untuk lebih jelasnya perhatikan gambar dibawah ini:



Gambar 2.2 Tampilan Arduino IDE

- 1. **Verify** pada versi sebelumnya dikenal dengan istilah Compile. Sebelum aplikasi di-upload ke board Arduino, biasakan untuk memverifikasi terlebih dahulu sketch yang dibuat. Jika ada kesalahan pada sketch, nanti akan muncul error. Proses Verify / Compile mengubah sketch ke binary code untuk di-upload ke mikrokontroller.
- 2. **Upload** tombol ini berfungsi untuk mengupload sketch ke board Arduino. Walaupun kita tidak mengklik tombol verify, maka sketch akan di-compile, kemudian langsung di upload ke board. Berbeda dengan tombol verify yang hanya berfungsi untuk memverifikasi source code saja.
- 3. New Sketch Membuka window dan membuat sketch baru

- 4. **Open Sketch** Membuka sketch yang sudah pernah dibuat. Sketch yang dibuat dengan IDE Arduino akan disimpan dengan ekstensi file .ino
- 5. Save Sketch menyimpan sketch, tapi tidak disertai dengan mengcompile.
- 6. **Serial Monitor** Membuka interface untuk komunikasi serial, nanti akan kita diskusikan lebih lanjut pada bagian selanjutnya.
- 7. **Keterangan Aplikasi** pesan-pesan yang dilakukan aplikasi akan muncul di sini, misal *Compiling* dan *Done Uploading* ketika kita *mengcompile* dan mengupload sketch ke board Arduino
- 8. **console log** Pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang sketch akan muncul pada bagian ini. Misal, ketika aplikasi mengcompile atau ketika ada kesalahan pada sketch yang kita buat, maka informasi error dan baris akan diinformasikan di bagian ini.
- 9. **Baris Sketch** bagian ini akan menunjukkan posisi baris kursor yang sedang aktif pada sketch.
- 10. **Informasi Board dan Port** Bagian ini menginformasikan port yang dipakai oleh board Arduino.

Software Arduino IDE ini akan digunakan sebagai pembuatan program untuk *prototype* PKA . Setelah memprogram hal yang perlu dilakukan yaitu meng*upload* skecth kepada NodeMCU.

2.1.3 Unity



Gambar 2.3 Software Unity

Unity merupakan suatu aplikasi yang digunakan untuk mengembangkan game multi platform yang didesain untuk mudah digunakan. Editor pada Unity dibuat dengan user interface yang sederhana. Grafis pada unity dibuat dengan grafis tingkat tinggi untuk OpenGL dan directX. Unity mendukung semua format file, terutamanya format umum seperti semua format dari art applications. Unity cocok dengan versi

64-bit dan dapat beroperasi pada Mac OS x dan windows dan dapat menghasilkan game untuk Mac, Windows, Wii, iPhone, iPad dan Android.

Unity ini adalah sebuah aplikasi berbasis MultiPlatform, Apa itu MultiPlatform? MultiPlatform merupakan aplikasi yang dapat beroperasi di banyak sistem operasi dan sanggup mempublish ke banyak format tipe file, misalnya: exe, apk,dan lainlain. Unity mempunyai beberapapa kelebihan dan kekurangan seperti berikut:

1. Kelebihan

- (a) Gratis, tidak perlu mengeluarkan biaya. Cukup dengan Unity versi Free (Personal Edition), maka sudah bisa membuat karya game.
- (b) Cross-platform, artinya project game dapat dirilis ke dalam berbagai platform terkenal.
- (c) UI yang ditawarkan cukup user-friendly, sehingga cocok untuk pemula.
- (d) Dokumentasi atau tutorial yang banyak dan gratis untuk dipelajari resmi dari Unity.
- (e) Memiliki assets store, di mana kita dapat mencari assets 2D / 3D gratis maupun berbayar.
- (f) Mempunyai text editor bernama Monodevelop. Mirip seperti visual code, Monodevelop ini dapat digunakan untuk coding game dan sudah terintegrasi langsung ke Unity Engine.
- (g) Proses desain level game menjadi lebih mudah. Unity menawarkan banyak alat untuk desain level game, seperti Terrain Editor, Responsive UI Editor, dan masih banyak lagi.

2. Kekurangan

- (a) Butuh penyesuaian yang cukup kompleks pada saat ingin mengembangkan game 2D tanpa bantuan plugin.
- (b) Perlu penyesuaian mengikuti gaya component based.
- (c) Cache memory yang digunakan lumayan banyak.

2.1.4 Fitur Dalam Unity

1. Rendering

Graphics engine yang digunakan adalah Direct3D (Windows, Xbox 360), Open GL (Mac, Windows, Linux, PS3), OpenGL ES (Android, iOS), dan proprietary APIs (Wii). Ada kemampuan untuk bump mapping, reflection mapping, parallax mapping, screen space ambient occlusion (SSAO), dynamic shadows using shadow maps, render-to-texture and full-screen post-processing effects.

Unity dapat mengambil format desain dari 3ds Max, Maya, Softimage, Blender, modo, ZBrush, Cinema 4D, Cheetah3D, Adobe Photoshop, Adobe Fireworks and Allegorithmic Substance. Asset tersebut dapat ditambahkan ke game project dan diatur melalui graphical user interface Unity. Scripting

Script game engine dibuat dengan Mono 2.6, sebuah implementasi open-source dari .NET Framework. Programmer dapat menggunakan UnityScript (bahasa terkustomisasi yang terinspirasi dari syntax ECMAScript, dalam bentuk JavaScript), C, atau Boo (terinspirasi dari syntax bahasa pemrograman phyton). Dimulai dengan dirilisnya versi 3.0, Unity menyertakan versi MonoDevelop yang terkustomisasi untuk debug script.

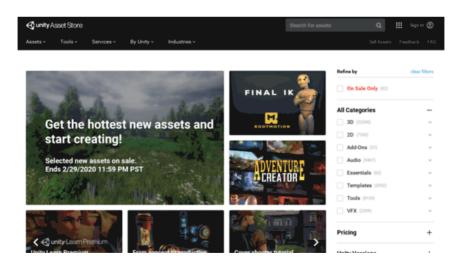
2. Asset Tracking

Unity menyertakan Server Unity Asset sebuah solusi terkontrol untuk developer game asset dan script. Server tersebut menggunakan PostgreSQL sebagai backend, sistem audio dibuat menggunakan FMOD library (dengan kemampuan untuk memutar Ogg Vorbis compressed audio), video playback menggunakan Theora codec, engine daratan dan vegetasi (dimana mensuport tree bill-boarding, Occlusion Culling dengan Umbra), built-in lightmapping dan global illumination dengan Beast, multiplayer networking menggunakan RakNet, dan navigasi mesh pencari jalur built-in.

3. Platforms

Unity support pengembangan ke berbagai platform. Didalam project, developer memiliki kontrol untuk mengirim ke perangkat mobile, web browser, desktop, and console. Unity juga mengizinkan spesifikasi kompresi textur dan pengaturan resolusi di setiap platform yang didukung.

4. Asset Store

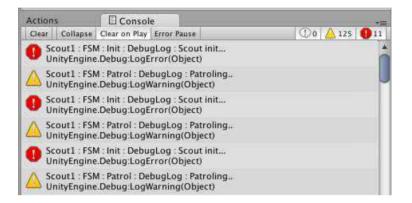


Gambar 2.4 Tampilan Halaman Asset Store

Asset store adalah tempat dimana asset gratis dan berbayar yang disediakan oleh Unity. Kita dapat menggunakan asset-asset tersebut untuk kebutuhan pengembangan game kita. Unity juga menyediakan sample project untuk digunakan sebagai awalan dalam mengembangkan game, misal kita ingin membuat game

FPS maupun TPS, kemudian import ke dalam Unity untuk dikembangkan lebih lanjut atau hanya untuk sekedar belajar membuat game 3D maupun 2D. Selain itu juga bisa menggunakan Asset Store sebagai market untuk menjual asset yang kita buat.

5. Console



Gambar 2.5 Tampilan Console Pada Unity

Console adalah tempat yang menampilkan pesan error dalam project, kita juga dapat melihat script error mana yang terjadi dan dapat diperbaiki script error tersebut. Dan pesan error yang ditampilkan sering terjadi saat membuat script yang tidak dikenali oleh sistem atau kesalahan dalam pembuatan script.

6. **Build Settings**



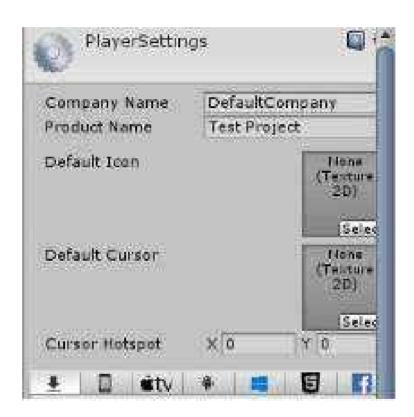
Gambar 2.6 Tampilan Build Setting Pada Unity

Build Settings adalah tempat untuk mengekspor game yang telah dibuat. Pada Scenes In Build juga dapat mengatur scene mana saja yang kita tampilkan di game nantinya. Kita juga dapat memilih platform yang diinginkan, misal kita ingin mengubah ke Platform Android, pilih Android lalu pilih Switch Platform. Jika sudah memilih platform yang diinginkan, kita dapat memilih Build atau Build And Run. Pastikan juga sistem atau script yang kamu buat tidak terdapat error.

7. Player Settings

Player Settings adalah tempat konfigurasi lanjutan dari Build Settings, disini ada 6 poin untuk mengatur game yang dibuat sebelum di ekspor, yaitu:

(a) Player Setting



Gambar 2.7 Tampilan Player Setting Pada Unity

PlayerSettings adalah tempat dimana dapat mengisi Company Name, Product Name, Default Icon, Default Cursor (opsional). Yang mana, - Company Name: nama perusahaan atau nama studio kita atau nama developer - Product Name: nama game atau aplikasi yang dibuat - Default Icon: gambar icon dari aplikasi atau game

(b) Resolution and Presentation

Resolution and presentation adalah tempat dimana kita mengatur resolusi (ukuran) pada platform maupun device yang dipilih. Misal,ingin mengatur

resolusi menjadi penuh (full) harus mencentang Default Is Full Screen, atau dapat mengatur resolusi khusus untuk menyesuaikan dengan tampilan layar pada *screen*. Seperti berikut:



Gambar 2.8 Tampilan Resolution and Presentation Pada Unity

(c) Icon

Icon adalah tempat dimana untuk mengatur gambar utama yang menjadikan sebuah logo di aplikasi atau game.

(d) Splash Image

Splash image adalah tempat dimana untuk mengatur tampilan awal sebelum memulainya game (masuk ke menu utama), Splash Image dapat dikustomisasi dengan logo perusahaan atau studio kita.



Gambar 2.9 Tampilan Splash Image Pada Unity

(e) Other Settings

Other Settings adalah tempat dimana kita mengatur mekanisme game agar game kita lebih optimal dan sesuai target platform atau device yang dituju.

- Bundle Identifier: untuk mengidentifikasikan aplikasi atau game agar memiliki pengenal bundel yang berbeda - Version: untuk mengatur versi aplikasi yang akan di publish - Optimization: untuk mengoptimasi game.

(f) XR Settings

XR settings adalah tempat dimana kita dapat mengatur atau membuat game Virtual Reality atau Augmented Reality dengan cara mencentangnya.

XR Settings	
Virtual Reality Supported 🔲	
Vuforia Augmented Realit	

Gambar 2.10 Tampilan XR Settings Pada Unity

2.1.5 Bot Telegram



Gambar 2.11 Logo Telegram

Seiring Messenger Telegram yang mulai di*install* banyak orang dan dipergunakan untuk percakapan sehari-hari. Memang Telegram belum sepopuler Whatsapp, BBM, maupun Line. Namun, bisa jadi suatu saat akan menjadi suatu messenger yang potensial mendapatkan hati di kalangan masyarakat maya. Menurut Cokrojoyo kelebihan dari Telegram ini adalah adanya landasan untuk menggunakan *Application Programming Interface*(API) untuk masyarakat luas. Salah satu API yang disediakan adalah fitur bot. Bot Telegram adalah bot yang saat ini mulai populer dipergunakan.

Keunggulan pertama dari aplikasi Telegram ini adalah fleksibel. Artinya Anda bisa membuat fitur-fitur tambahan yang disertakan dalam aplikasi Telegram ini. Misalkan Anda ingin membuat polling. Anda bisa membuatnya tanpa sendiri dan menambahkan di Telegram Anda tanpa harus menunggu ada update fitur dari Developernya. Sehingga ini membuat Telegram menjadi lebih lebih fleksibel jika dibanding aplikasi

sejenis seperti WhatsApp ataupun LineMessenger. Selain itu Keunggulan ke dua dari Telegram adalah pesan berbasis awan (cloud-based message). Artinya dengan Telegram Anda bisa berkomunikasi dengan siapapun (yang juga punya akun Telegram) tanpa batasan device/ gadget.

Bot telegram digunakan untuk mengirimkan notifikasi kepada user. Notifikasi ini akan dikirimkan kepada user jika jarak ketinggian air sudah mencapai jarak tertinggi. Pesan yang dikirimkan melalui telegram yaitu jarak ketinggian air dan prediksi waktu berapa lama air mencapai pada volume tertentu.

2.1.6 Antares



Gambar 2.12 Logo Antares

Antares merupakan sebuah *platform IoT* lokal milik Telkom yang telah mendapat pengakuan dari dunia internasional yang dikembangkan oleh departemen media dan digital Telkom MDD. Antares menjadi jembatan dalam solusi IoT yang mendukung berbagai macam protokol yang umum digunakan untuk solusi IoT seperti MQTT, HTTP, websocket, dan CoAP disamping format data JSON dan XML. Selain itu, untuk memudahkan pengembang perangkat lunak dan keras disediakan pula *library* untuk Android dan *microcontroller* berbasis Arduino.



Gambar 2.13 Tampilan Antares Antares

Antares ini digunakan sebagai data base, dimana jarak ketinggian air yang dibaca oleh sensor ultrasonic akan dikirimkan ke antares secara realtime. Setelah data itu disimpan nantinya data tersebut akan ditampilkan pada aplikasi monitoring yang bertujuan agar user dapat mengetahui jarak ketinggian air.

Selain antares masih banyak yang dapat kita gunakan untuk menyimpan data secara online salah satunya yaitu Firebase realtime database yang merupakan database yang di-host di cloud. Data disimpan sebagai JSON dan disinkronkan secara realtime ke setiap klien yang terhubung. Ketika Anda membuat aplikasi lintas platform dengan Android SDK, iOS SDK, dan JavaScript SDK, semua klien Anda akan berbagi sebuah instance Realtime Database dan menerima update data terbaru secara otomatis.



Gambar 2.14 Logo Firebase

2.1.7 Software Membuat 3D Objek

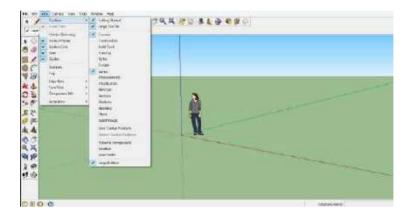
3D objek ini dibuat untuk sebuah desain prototipe. Selain sebagai desain prototipe, 3D objek ini akan digunakan untuk augmented reality. Untuk membuat sebuah desain ada beberapa software yaitu:

1. sketchup Make

SketchUp adalah sebuah perangkat lunak desain grafis yang dikembangkan oleh Trimble. Pendesain grafis ini dapat digunakan untuk membuat berbagai jenis model, dan model yang dibuat dapat diletakkan di Google Earth atau dipamerkan di 3D Warehouse. Tersedia dua versi SketchUp, yaitu SketchUp Make (gratis) dan SketchUp Pro (harga: USD 590.00).



Gambar 2.15 sketchup Make



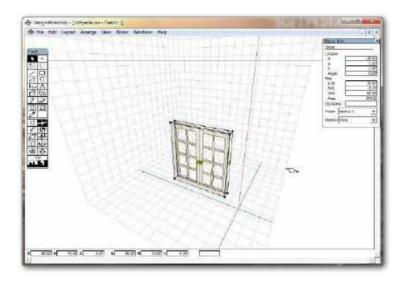
Gambar 2.16 Tampilan sketchup Make

2. DesignWorkshop Lite



Gambar 2.17 DesignWorkshop Lite

DesignWorkshop Lite merupakan paket aplikasi untuk Desain Interior yang menyediakan semua yang kamu butuhkan untuk membangun model desain dan visualisasi arsitektur 3D, lanskap, atau desain tata ruang. Dengan DesignWorkshop Lite kamu dapat melihat model siap pakai dari berbagai ukuran seperti dari website free 3D building models, DXF, DesignWorkshop, atau file dengan format 3DMF.



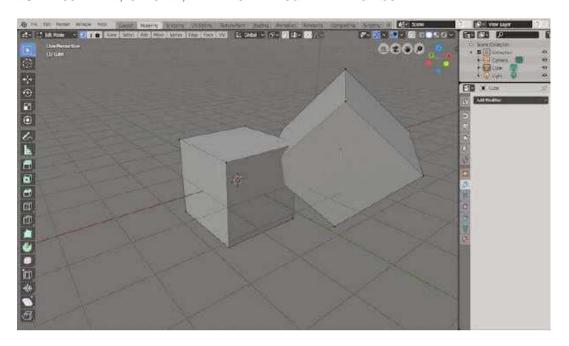
Gambar 2.18 Tampilan DesignWorkshop Lite

3. Blender



Gambar 2.19 Blender

Blender merupakan aplikasi pemodelan komplit dan komprehensif untuk menciptakan, mengubah dan mengedit model 3D dengan mudah. Transformasi model menjadi karakter yang menakjubkan menjadi lebih mudah dari sebelumnya dengan menggunakan Blender. Perangkat lunak ini digunakan untuk membuat film animasi, efek visual, model cetak 3D, aplikasi 3D interaktif dan permainan video. Blender memiliki beberapapa fitur termasuk pemodelan 3D, penteksturan, penyunting gambar bitmap, penulangan, simulasi cairan dan asap, simulasi partikel, animasi, penyunting video, pemahat digital, dan rendering.



Gambar 2.20 Tampilan Blender

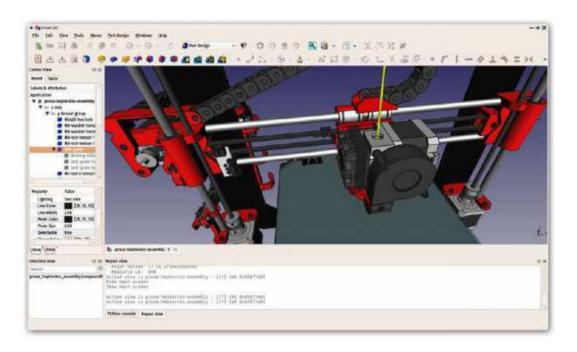
4. Free CAD



Gambar 2.21 Free CAD

FreeCAD adalah software 3D modeling serba-gratis dengan banyak fitur yang bisa disesuaikan, diubah, dan ditambah.FreeCAD bisa digunakan untuk membuat sketsa objek 2 dimensi, seperti bentuk Box, Sphere, Cone, Cylinder, dan Torus; lengkap dengan Tools modifikasi seperti Thickness, Cut, Fillet, Extrude,

Fuse, Boolean.Bisa juga digunakan untuk membuat simulasi gerakan robot, menciptakan model game 3D, model animasi, model arsitektur, model industrial, hingga menampilkan dan menyesuaikan G code.



Gambar 2.22 Tampilan Free CAD

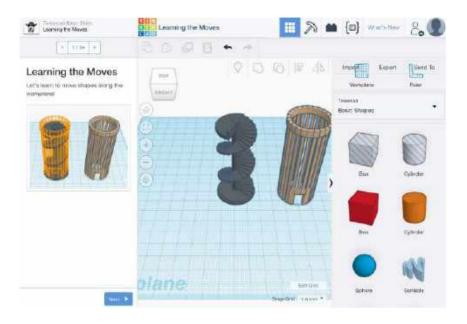
5. TinkerCAD



Gambar 2.23 TinkerCAD

TinkerCAD bisa digunakan untuk membuat 3D model dari dasar menggunakan basic modeling operations, tapi dengan tambahan fungsi untuk menggabungkan atau memisahkan bentuk model yang sedang dibuat.

Setelah menguasai teknik dasar 3D modeling seperti ini, maka pengguna TinkerCAD akan mampu menciptakan model 3D yang mengesankan seperti layaknya RaspberryPi.



Gambar 2.24 Tampilan TinkerCAD

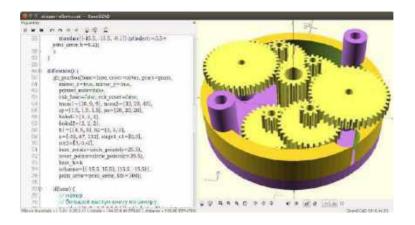
6. OpenSCAD



Gambar 2.25 OpenSCAD

OpenSCAD adalah aplikasi software 3D modeling gratis, tapi bukanlah software yang menarik, dengan cara penggunaan yang berbeda dari software lain. Software OpenSCAD merupakan kompilator 3D berdasarkan bahasa deskripsi tekstual.

Dalam kalimat yang sederhana, model 3D dibuat menggunakan kode. Baik kode yang dirangkai dari dasar ataupun menggunakan kode yang telah digunakan untuk membuat proyek serupa.



Gambar 2.26 Tampilan OpenSCAD

2.1.8 Vuforia



Gambar 2.27 Logo Vuforia

Vuforia merupakan SDK yang dikembangkan oleh Qualcomm untuk membantu pengembang dalam menciptakan aplikasi atau game yang memiliki teknologi Augmented Reality. Aplikasi maupun game yang dibuat dengan teknologi ini akan terlihat lebih interaktif dan hidup.

Dengan adanya Vuforia SDK memudahkan dan mempercepat pengembang dalam membuat aplikasi yang mempunyai teknologi Augmented Reality karena library dan fungsi-fungsi intinya sudah dibuat oleh Qualcomm sehingga pengembang mengembangkan aplikasi menarik menggunakan SDK yang tersedia.

SDK ini memiliki berbagai fitur menarik seperti memindai objek, memindai teks, mengenali bingkai penanda, tombol virtual, mengidentifikasi permukaan objek secara pintar, memindai dengan berbasis awan, mengenali target gambar, mengenali target benda silinder, dan mengenali objek target yang telah ditetapkan.

Vuforia SDK mendukung pengembangan aplikasi untuk perangkat yang berbasis iOS dan Android. Selain itu, Unity Game Engine telah didukung Vuforia SDK sehingga lebih mempermudah untuk mengembangkan game Augmented Reality.

2.1.9 Software Perakitan Komponen

Komponen elektronika adalah komponen yang paling kompleks dan sensitif, tidak jarang terjadi kerusakan apabila komponen yang digunakan tidak sesuai semestinya, misalkan kelebihan daya, arus, short, dll.

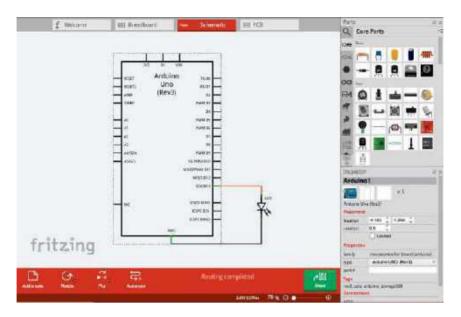
Maka dari itu software simulasi dapat membantu proses merancang rangkaian elektronika sebelum diimplementasikan langsung pada komponen.Untuk membuat suatu rangkaian atau skema kita bisa menggunakan beberapapa software seperti :

1. Frizting

Frizting merupakan aplikasi berbasis free software yang dapat diunduh secara gratis. Aplikasi ini berfungsi untuk membuat sebuah skema rangkaian elektronika secara nyata (real) Karena komponen yang berada pada rangkaian tersebut digambarkan dengan sangat mirip dengan komponen aslinya. Aplikasi ini tidak hanya dilengkapi dengan skema rangkaian yang sangat baik, banyak fitur lainnya yang dapat kita coba seperti PCB layout dan programming Arduino langsung pada aplikasi ini.



Gambar 2.28 Tampilan Frizting



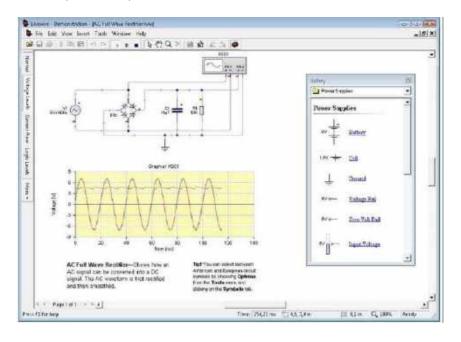
Gambar 2.29 Tampilan Frizting



Gambar 2.30 Tampilan Frizting

2. LiveWire

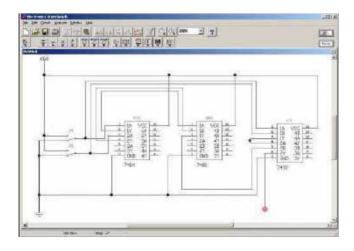
LiveWire merupakan aplikasi simulasi rangkaian elektronika yang mudah digunakan. Dalam aplikasi ini disediakan komponen komponen dasar seperti resistor, kapasitor, op amp, dll. Dengan begitu kita dapat merangkai skema yang akan kita buat (misalkan rangkaian RLC) lalu mensimulasikannya, sehingga dapat melihat arus yang mengalir pada rangkaian tersebut.



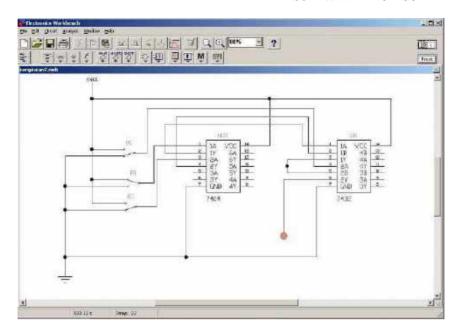
Gambar 2.31 Tampilan LiveWire

3. Electronics Workbench (EWB)

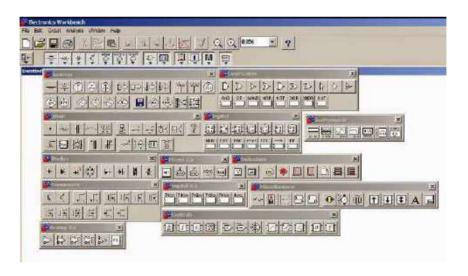
Electronics Workbench atau yang sering dikenal dengan EWB merupakan aplikasi yang tidak jauh berbeda dengan LiveWire. Aplikasi ini sangat mudah untuk digunakan, ditambah dengan beberapapa feature yang lebih lengkap seperti adanya simulasi osiloskop, spektrum analizer, dan *function generator*.



Gambar 2.32 Tampilan *Electronics Workbench* (EWB)



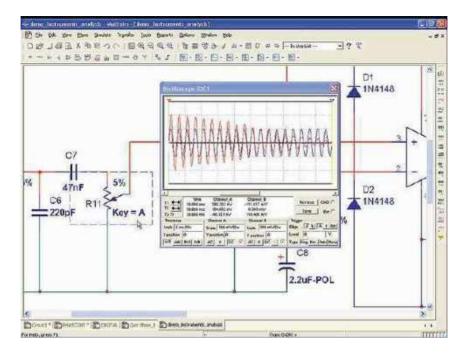
Gambar 2.33 Tampilan Electronics Workbench (EWB)



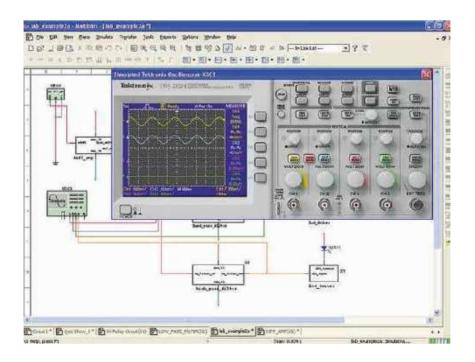
Gambar 2.34 Tampilan *Electronics Workbench* (EWB)

4. NI Multisim

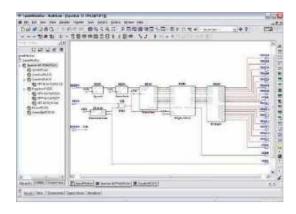
Aplikasi ini dilengkapi dengan komponen-komponen yang memiliki fungsi yang sama dengan aslinya, begitupun dengan perangkat elektronika seperti osiloskop, function generator, spektrum analizer, dll.



Gambar 2.35 Tampilan NI Multisim



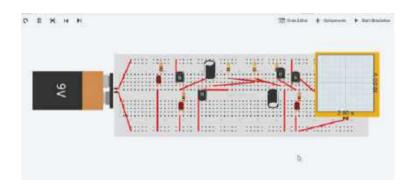
Gambar 2.36 Tampilan NI Multisim



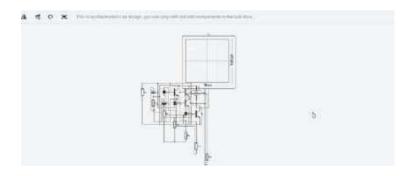
Gambar 2.37 Tampilan NI Multisim

5. Autodesk Circuits

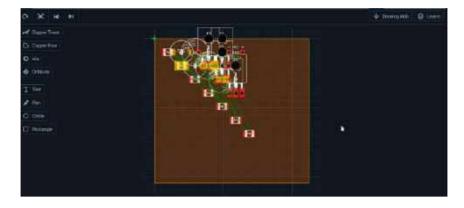
Autodesk Circuits merupakan aplikasi berbasis online. Fitur yang dimiliki oleh aplikasi ini beragam diantaranya Electronics Lab, PCB Design, Library Manager, Circuit Scribe, dan Eagel. Akan tetapi komponen yang berada pada librarynya terbatas.



Gambar 2.38 Tampilan Autodesk Circuits



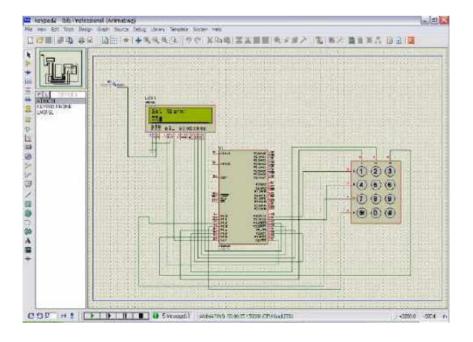
Gambar 2.39 Tampilan Autodesk Circuits



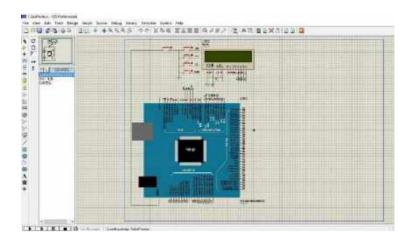
Gambar 2.40 Tampilan Autodesk Circuits

6. Proteus ISIS

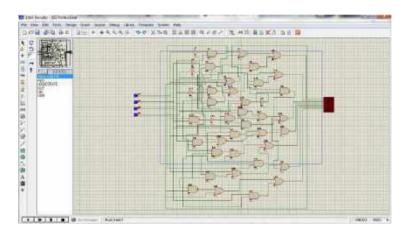
Proteus ISIS merupakan aplikasi yang dapat mensimulasikan semua rangkaian yang akan dibuat, aplikasi ini dilengkapi dengan text editor yang bisa digunakan untuk programming mikrokontroler. Kemudian dapat melakukan PCB layout pada rangkaian yang sudah dibuat, sehingga setelah membuat rangkaian untuk disimulasikan dan ternyata berhasil, kita dapat langsung merancangnya untuk di print pada layout PCB.



Gambar 2.41 Tampilan Proteus ISIS



Gambar 2.42 Tampilan Proteus ISIS



Gambar 2.43 Tampilan Proteus ISIS

Dengan adanya penjelasan tentang software pembuatan skema komponen, kita dapat memilih software untuk pembuatan komponen sesuai yang kita inginkan. Tetapi kita juga harus memilih software yang akan digunakan sesuai kebutuhan. Dalam pembuatan prototipe ini *software* yang digunakan yaitu Frizting.

2.2 Komponen Yang Digunakan

Dalam membuat *prototype* ini dibutuhkan beberapapa komponen elektronik maupun komponen non elektronik untuk membuat *prototype* PKA. Berikut komponen yang dibutuhkan :

1. NodeMCU

NodeMCU adalah ESP8266 (khususnya seri ESP-12, termasuk ESP-12E) maka fitur fitur yang dimiliki NodeMCU akan kurang lebih sama ESP-12 (juga ESP-12E untuk NodeMCU v.2 dan v.3) kecuali NodeMCU telah dibungkus oleh API

sendiri yang dibangun berdasarkan bahasa pemrograman eLua, yang kurang lebih cukup mirip dengan javascript. Selain dapat diprogram menggunakan bahasa LUA dapat juga diprogram menggunakan bahasa C menggunakan arduino IDE. [3] beberapapa fitur di dalamnya antara lain:

- (a) 0 Port GPIO dari D0 D10
- (b) Fungsionalitas PWM
- (c) Antarmuka I2C dan SPI
- (d) Antarmuka 1 Wire
- (e) ADC

Sebelum digunakan Board ini harus di Flash terlebih dahulu agar support terhadap tool yang akan digunakan. Jika menggunakan Arduino IDE menggunakan firmware yang cocok yaitu firmware keluaran dari AiThinker yang support AT Command. Untuk penggunaan tool loader Firmware yang digunakan adalah firmware NodeMCU.



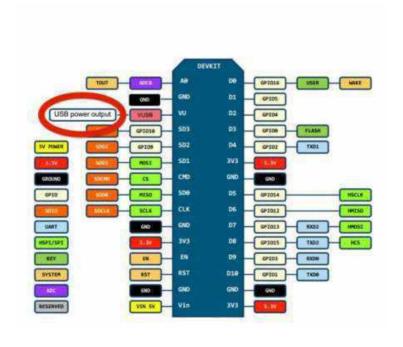
Gambar 2.44 NodeMCU

Gambar diatas merupakan sebuah NodeMCU yang akan digunakan sebagai *microcontroller* untuk pembuatan *prototype* prediksi ketinggian air (PKA) untuk pendeteksi banjir peringatan dini dengan notifikasi melalui bot telegram .

2. Spesifikasi Datasheet NodeMCU

SPESIFIKASI	NODEMCU V3		
Mikrokontroller	ESP8266		
Ukuran Board	57 mmx 30 mm		
Tegangan Input	3.3 ~ 5V		
GPIO	13 PIN		
Kanal PWM	10 Kanal		
10 bit ADC Pin	1 Pin		
Flash Memory	4 MB 40/26/24 MHz		
Clock Speed			
WiFi	4EEE 802.11 b/g/n		
Frekuensi	2.4 GHz - 22.5 Ghz		
USB Port	Micro USB		
Card Reader	Tidak Ada		
USB to Serial Converter	CH340G		

Gambar 2.45 Spesifikasi NodeMCU



Gambar 2.46 Skema Posisi Pin NodeMCU

3. Sensor Ultrasonic US 100

Sensor US-100 adalah versi peningkatan dari US-020 Ultrasonic Sensor pada kelasnya HC-SR04 memiliki performa lebih bagus dibanding US-020) yang sudah dilengkapi dengan fitur kompensasi temperatur. Di kelasnya (ultrasonic range sensor with temperature compensation) US-100 merupakan modul sensor jarak terbaik.

Modul jarak ultrasonik bekerja dengan sistem sonar seperti yang digunakan pada kapal selam, yaitu dengan melepaskan sinyal dalam bentuk gelombang ultrasonik (gelombang suara dengan frekuensi sangat tinggi di luar jangkauan pendengaran telinga manusia) dan mengukur waktu hingga gelombang tersebut dipantulkan. Dengan mengetahui kecepatan suara di udara, kita dapat mengubah besaran waktu ini menjadi jarak dengan rumus:

jarak = (selisih waktu * kecepatan suara diudara) / 2.

Fitur kompensasi suhu ini sangatlah penting untuk meningkatkan akurasi sensor sejenis ini berhubung kecepatan rambat suara di udara sangat terpengaruh oleh suhu/temperatur. Suara adalah sejenis energi kinetis. Molekul pada suhu tinggi memiliki tingkat energi lebih tinggi yang membuat mereka bergetar (vibrate) lebih cepat. Karena molekul ini bergetar lebih cepat, gelombang suara yang melewatinya dapat merambat dengan kecepatan lebih tinggi. Kecepatan rambat suara di udara pada suhu ruang (25C) sekitar 346 meter per detik, sementara pada suhu beku (0C), kecepatannya menurun menjadi 331 meter per detik. Untuk setiap derajat celcius kenaikan suhu, kecepatan rambatannya bertambah 60 cm per detik.

Distance Sensor US-100 ini mengukur suhu lingkungan (ambience temperature) dengan sensor suhu terpadu sehingga dapat mengkompensasi perbedaan suhu, menghasilkan pengukuran jarak yang sangat akurat. Berikut Gambar Sensor US 100:



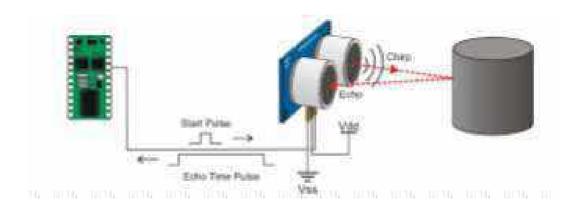
Gambar 2.47 Gambar Sensor Ultrasonic US 100

Sensor ultrasonic menggunakan daya + 5V yang diatur melalui pin Vcc and Ground dari sensor. Arus yang dikonsumsi oleh sensor kurang dari 15mA dan karenanya dapat langsung ditenagai oleh pin 5V on board (Jika tersedia). Trigger dan Echo pin keduanya adalah pin I / O dan karenanya mereka dapat dihubungkan ke pin I / O dari mikrokontroler. Untuk memulai pengukuran, pin pemicu harus dibuat tinggi untuk 10uS dan kemudian dimatikan.

Tindakan ini akan memicu gelombang ultrasonik pada frekuensi 40Hz dari pemancar dan penerima akan menunggu gelombang kembali. Setelah gelombang dikembalikan setelah dipantulkan oleh objek apa pun, pin Echo menjadi tinggi untuk jumlah waktu tertentu yang akan sama dengan waktu yang dibutuhkan gelombang untuk kembali ke sensor . [4]

Jumlah waktu selama pin Echo tetap tinggi diukur oleh MCU / MPU karena memberikan informasi tentang waktu yang dibutuhkan untuk gelombang untuk kembali ke Sensor.

4. Prinsip kerja Sensor ultrasonic



Gambar 2.48 Prinsip Kerja Sensor Ultrasonic

Seperti yang ditunjukkan di atas, sensor memiliki modul 4 pin, yang pin namanya masing-masing adalah Vcc, Trigger, Echo dan Ground. Sensor ini adalah sensor yang sangat populer digunakan dalam banyak aplikasi di mana mengukur jarak atau objek penginderaan diperlukan. Modul sensor Ultrasonik adalah cara mudah untuk mengukur jarak dari benda. Modul ini memiliki banyak aplikasi seperti sensor parkir, hambatan dan sistem pemantauan medan, pengukuran jarak industri, dll. Sistem ini memiliki stabilitas kinerja dan akurasi tinggi mulai dari 2 cm hingga 450cm. Modul ini memiliki dua mata seperti proyek di bagian depan yang membentuk pemancar dan Penerima Ultrasonik. Sensornya bekerja dengan rumus :

$Jarak = Kecepatan \times Waktu$

Gambar 2.49 Rumus perhitungan Jarak Sensor Ultrasonic

Pemancar ultrasonik mentransmisikan gelombang ultrasonik, gelombang ini bergerak di udara dan ketika ia keberatan dengan bahan apa pun itu dipantulkan kembali ke sensor gelombang. Untuk menghitung jarak menggunakan rumus di atas, harus mengetahui kecepatan dan waktu. Karena menggunakan gelombang Ultrasonik, kecepatan universal gelombang AS pada kondisi ruangan yang 330m / s. Sirkuit inbuilt pada modul akan menghitung waktu yang dibutuhkan untuk gelombang US untuk kembali dan menyalakan pin gema tinggi untuk jumlah waktu yang sama, dengan cara ini kita juga dapat mengetahui waktu yang dibutuhkan. Sekarang cukup hitung jaraknya menggunakan mikrokontroler atau mikroprosesor.

5. Spesifikasi(*Data Sheet*)Sensor Ultrasonik US 100

Spesifikasi atau Datasheet dari sensor ultrasonic US 100 sebagai berikut :

(a) Tegangan input: 5V DC

(b) Quiescent current: kurang dari 2mA

(c) output: 5V tinggi

(d) Level output: pada akhir 0V

(e) Sudut induksi: tidak lebih dari 15 derajat

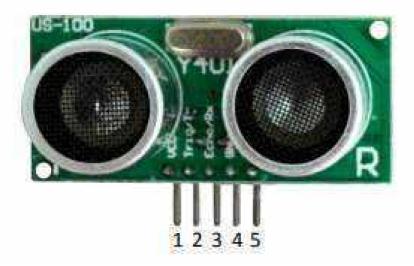
(f) Jarak deteksi: 2 cm-450cm

(g) Presisi: hingga 1mm

(h) Dimensi: 4.4cm x 2.6cm x 1.4cm

(i) Berat: 43g

Adapun pin konfigurasi dari sensor ultrasonic Us 100 yaitu :



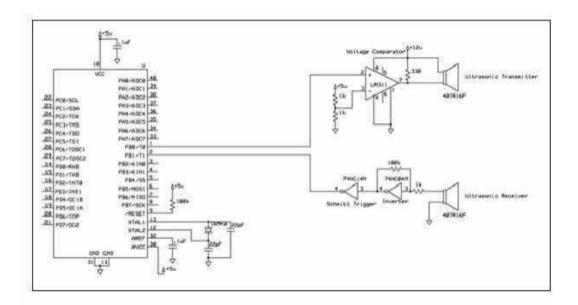
Gambar 2.50 Pin Konfigurasi

(a) VCC: 5V DC

(b) Trig: trigger input(c) Echo: pulse output(d) CND: ground

(d) GND: ground(e) GND: ground

Skema diagram sensor ultrasonic US 100 yaitu :



Gambar 2.51 Skema diagram sensor ultrasonic US 100

6. Buzzer

Buzzer merupakan sebuah komponen elektronika yang masuk dalam keluarga transduser, yang dimana dapat mengubah sinyal listrik menjadi getaran suara. Nama lain dari komponen ini disebut dengan *beeper* .



Gambar 2.52 Buzzer

Dalam kehidupan sehari hari, umumnya digunakan untuk rangkaian alarm pada jam, bel rumah, perangkat peringatan bahaya, dan lain sebagainya. Jenis jenis yang sering ditemukan dipasaran yaitu tipe piezoelectric. Dikarenakan tipe ini memiliki kelebihan seperti harganya yang relatif murah, mudah diaplikasikan ke dalam rangkaian elektronika.

7. Cara Kerja Buzzer



Gambar 2.53 Konfigurasi Pin Buzzer

Pada saat ada aliran catu daya atau tegangan listrik yang mengalir ke rangkaian yang menggunakan piezoelectric, maka akan terjadi pergerakan mekanis pada

piezoelectric tersebut. Yang dimana gerakan tersebut mengubah energi listrik menjadi energi suara yang dapat didengar oleh telinga manusia. Piezoelectric menghasilkan frekuensi di range kisaran antara 1 5 kHz hingga 100 kHz yang diaplikasikan ke Ultrasound. Tegangan operasional piezoelectric pada umumnya yaitu berkisar antara 3Vdc hingga 12 Vdc. Adapun konfigurasi pin buzzer yaitu sebagai berikut:

Pin Number	Pin Name	Description
1	Positive	Identified by (+) symbol or longer terminal lead. Can be powered by 6V DC
2	Negative	Identified by Short terminal lead. Typically connected to the ground of the circuit

Gambar 2.54 Keterangan Konfigurasi Pin Buzzer

8. Spesifikasi(DataSheet) Buzzer

Spesifikasi atau datasheet dari buzzer meliputi:

(a) Nilai Tegangan: 6V DC

(b) Tegangan Pengoperasian: 4-8V DC

(c) Nilai current: ¡30mA

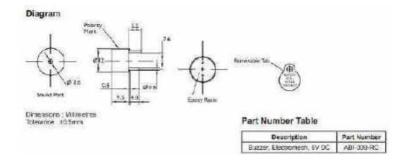
(d) Tipe Suara: Bunyi Kontinu

(e) Frekuensi resonansi: 2300 Hz

(f) Kecil dan package rapih

Buzzer adalah komponen kecil namun efisien untuk menambahkan fitur suara ke proyek / sistem. Ini adalah struktur 2-pin yang sangat kecil dan kompak sehingga dapat dengan mudah digunakan pada Breadboard dan bahkan pada PCB yang menjadikannya komponen yang banyak digunakan dalam sebagian besar aplikasi elektronik.

Buzzer ini dapat digunakan hanya dengan menyalakannya menggunakan catu daya DC mulai dari 4V hingga 9V. Baterai 9V sederhana juga dapat digunakan, tetapi disarankan untuk menggunakan catu daya + 5V atau + 6V yang teregulasi. Buzzer biasanya dikaitkan dengan sirkuit *switching* untuk menghidupkan atau mematikan buzzer pada waktu yang diperlukan dan membutuhkan interval. Adapun diagram buzzer atau model 2D buzzer seperti pada gambar dibawah:



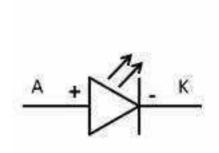
Gambar 2.55 Diagram model 2D Buzzer

9. **Led**

LED (*Light Emitting Diode*) adalah komponen elektronika yang dapat memancarkan cahaya monokromatik ketika diberikan tegangan maju. LED merupakan keluarga Dioda yang terbuat dari bahan semikonduktor. Warna-warna Cahaya yang dipancarkan oleh LED tergantung pada jenis bahan semikonduktor yang dipergunakannya. LED juga dapat memancarkan sinar inframerah yang tidak tampak oleh mata seperti yang sering kita jumpai pada Remote Control TV ataupun Remote Control perangkat elektronik lainnya.



Gambar 2.56 Led



Gambar 2.57 Simbol Led

10. Cara Kerja Led

LED hanya akan memancarkan cahaya apabila dialiri tegangan maju (bias forward) dari Anoda menuju ke Katoda. LED terdiri dari sebuah chip semikonduktor yang di doping sehingga menciptakan junction P dan N. Yang dimaksud dengan proses doping dalam semikonduktor adalah proses untuk menambahkan ketidakmurnian (impurity) pada semi

konduktor yang murni sehingga menghasilkan karakteristik kelistrikan yang diinginkan. Ketika LED dialiri tegangan maju atau bias forward yaitu dari Anoda (P) menuju ke Katoda (K), Kelebihan Elektron pada N-Type material akan berpindah ke wilayah yang kelebihan Hole (lubang) yaitu wilayah yang bermuatan positif (P-Type material). Saat Elektron berjumpa dengan Hole akan melepaskan photon dan memancarkan cahaya monokromatik (satu warna).

11. Resistor



Gambar 2.58 resistor

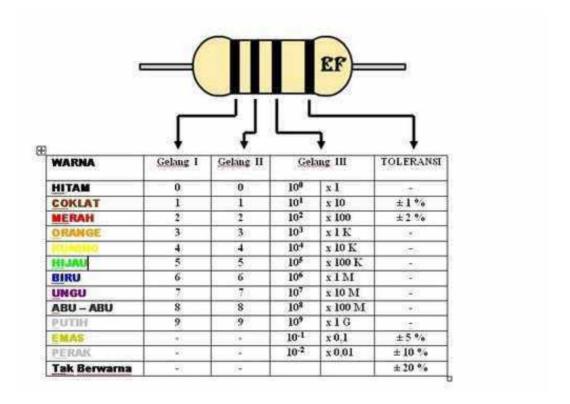
Resistor merupakan komponen elektronika yang tidak memiliki kutub sehingga dapat dipasang bolak balik yang tidak akan menimbulkan masalah pada peralatan elektronika. Resistor atau hambatan salah satu komponen elektronika yang memiliki nilai hambatan tertentu, dimana hambatan ini akan menghambat arus listrik yang mengalir melaluinya.

Satuan Resistor adalah Ohm (simbol:) yang merupakan satuan SI untuk resistansi listrik. Dalam sejarah, kata ohm itu diambil dari nama salah seorang fisikawan hebat asal German bernama George Simon Ohm. Beliau juga yang mencetuskan keberadaan hukum ohm yang masih berlaku hingga sekarang.

12. Fungsi Resistor

Resistor berfungsi sebagai penghambat arus listrik. Jika ditinjau secara mikroskopik, unsur-unsur penyusun resistor memiliki sedikit sekali elektron bebas. Akibat-nya pergerakan elektronya menjadi sangat lambat. Sehingga arus yang terukur pada multimeter akan menunjukan angka yang lebih rendah jika dibandingkan rangkaian listrik tanpa resistor.

13. Macam-Macam Resistor Sesuai Warna



Gambar 2.59 Macam-Macam Resistor Sesuai Warna

Dari gambar diatas kita dapat menyesuaikan penggunaan resistor untuk kebutuhan kita.

14. Resistor Yang Digunakan

Resistor yang digunakan untuk pembuatan *prototype* prediksi ketinggian air (PKA) untuk mendeteksi banjir peringatan dini ini yaitu resistor 300 OHM 1/2WATT Carbon Film Resistor seperti pada gambar dibawah ini :



Gambar 2.60 Resistor Yang Digunakan

Resistor 300 OHM 1/2WATT Carbon Film digunakan untuk menjadi hambatan lampu led, agar pada saat lampu led menyala tidak terlalu terang dan tidak menyebabkan sakit mata pada saat dilihat. Resistor resistor 300 OHM 1/2WATT Carbon Film ini cocok digunakan pada prototipe ini karena cocok sebagai hambatan lampu led karena hanya 300 OHM. Sehingga hambatan yang diberikan oleh resistor ini kepada led tidak terlalu besar.

15. Kabel Jumper



Gambar 2.61 Kabel Jumper

Kabel Jumper digunakan untuk menghubungkan komponen satu dengan komponen lainya. Selain itu kabel jumper ini juga untuk menghubungkan kompo-

nen dengan *microcontroller*. Ada beberapapa jenis kabel jumper yaitu sebagai berikut :



Gambar 2.62 Kabel Jumper female



Gambar 2.63 Kabel Jumper Male

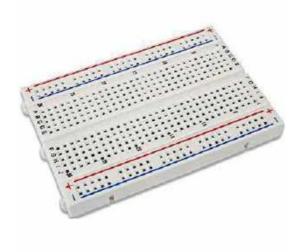


Gambar 2.64 Kabel Jumper Male-female

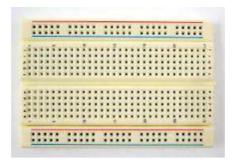
16. BreadBoard

Breadboard adalah board yang digunakan untuk membuat rangkaian elektronik sementara dengan tujuan uji coba atau prototipe tanpa harus menyolder. Dengan memanfaatkan breadboard, komponen-komponen elektronik yang dipakai tidak akan rusak dan dapat digunakan kembali untuk membuat rangkaian yang lain. Breadboard umumnya terbuat dari plastik dengan banyak lubang-lubang di atasnya. Lubang-lubang pada breadboard diatur sedemikian rupa membentuk pola sesuai dengan pola jaringan koneksi di dalamnya.

Breadboard yang tersedia di pasaran umumnya terbagi atas 3 ukuran: mini breadboard, medium breadboard atau large breadboard. Mini breadboard memiliki 170 titik koneksi (bisa juga lebih). Kemudian medium breaboard memiliki 400 titik koneksi. Dan large breadboard memiliki 830 titik koneksi.



Gambar 2.65 Breadboard Mini



Gambar 2.66 Breadboard Medium



Gambar 2.67 breadboard Large

17. Kardus



Gambar 2.68 Kardus

Kardus akan digunakan untuk membuat miniatur rumah , pohon , dan jembatan. Dengan menggunakan kardus miniatur akan lebih kokoh dan tidak gampang loyo serta patah.

18. Kertas Warna



Gambar 2.69 Kertas Warna

Kertas warna digunakan untuk memberikan warna kepada kardus yang sudah dibentuk menjadi rumah, jembata, dan pohon. Jika tidak ingin menggunakan kertas warna kita dapat memberi warna menggunakan cat air , tetapi akan lebih sulit pada saat mengimplementasikan.

19. Lem Kayu



Gambar 2.70 Lem Kayu

Lem kayu berfungsi untuk merekatkan suatu komponen dengan komponen lainya. Pada pembuatan prototipe ini yang akan di rekatkan yaitu kardus yang sudah dibentuk menjadi rumah, jembatan dan pohon. Kardus tersebut akan direkatkan pada dudukan kayu.

20. Ember atau Wadah



Gambar 2.71 Wadah atau Ember

Wadah atau ember ini akan digunakan untuk menampung air pada prototipe PKA. Ember ini akan kaitkan pada bagian bolong tengah kayu yang sudah dibentuk persegi. Wadah atau ember ini dikaitkan dengan cara merekatkannya dengan paku.

21. **Kayu**



Gambar 2.72 Kayu

Kayu pada pembuatan prototipe ini digunakan untuk membuat sebuah dudukan miniatur. Kayu tersebut dibentuk persegi empat seperti meja yang memiliki

4 buah kaki (penyangga). Kemudian untuk bagian tengah akan diberi bolong yang berfungsi untuk air sebagai simulasi.

22. Cat Kayu



Gambar 2.73 Cat Kayu

Cat kayu merupakan sebuah cairan lapisan untuk kayu yang memiliki berbagai jenis warna. Cat kayu pada pembuatan prototipe ini digunakan untuk memberikan warna pada kayu yang telah dibentuk persegi empat yang berfungsi sebagai dudukan miniatur.

2.3 Bahasa pemrograman Yang Digunakan

2.3.1 Pengertian Bahasa pemrograman C

Bahasa C merupakan perkembangan dari bahasa pemrograman c yang diciptakan oleh Brian W. Kerninghan dan Dennis M. Ritchie lalu di kembangkan oleh Bjarne Stroustrup dari Laboratorium Bell, ATT, pada tahun 1983. C cukup kompatibel dengan bahasa pendahulunya C. Pada mulanya C disebut a better C . Nama C sendiri diberikan oleh Rick Mascitti pada tahun 1983, yang berasal dari operator increment pada bahasa C. Keistimewaan yang sangat berarti dari C ini adalah karena bahasa ini mendukung Pemrograman Berorientasi Objek (OOP /Object Oriented Programming) .[5]

Bahasa pemrograman C ini dapat digunakan untuk memprogram sebuah robot, Untuk memprogram menggunakan bahasa C ini dapat menggunakan IDE (Integrated Development Environment) salah satu contohnya yaitu IDE Arduino.

2.3.2 Kelebihan dan Kekurangan Bahasa C

Dalam bahasa pemrograman C ini memiliki kelebihan yaitu sebagai berikut :

- 1. Bahasa C tersedia hampir di semua jenis computer.
- 2. Kode bahasa C sifatnya adalah portable dan fleksibel untuk semua jenis computer.
- 3. Bahasa C hanya menyediakan sedikit kata-kata kunci, hanya terdapat 32 kata kunci.
- 4. Proses executable program bahasa C lebih cepat
- 5. Dukungan pustaka yang banyak.
- 6. C adalah bahasa yang terstruktur
- 7. Bahasa C termasuk bahasa tingkat menengah

Selain memiliki kelebihan bahasa pemrograman C juga mempunyai kekurangannya yaitu :

- 1. Banyaknya Operator serta fleksibilitas penulisan program kadang-kadang membingungkan pemakai.
- 2. Bagi pemula pada umumnya akan kesulitan menggunakan pointer.

2.3.3 Sketch Arduino

Pada arduino bahasa pemrograman yang digunakan adalah bahasa C/C++. Program pada Arduino terbagi menjadi tiga bagian utama yaitu Structure, Values (berisi variabel dan konstanta) dan yang terakhir function. [6] Struktur kode pada arduino yaitu berisi fungsi setup() dan loop() sebagai berikut :

1. **Setup()**

fungsi ini dipanggil pertama kali ketika menjalankan *sketch*. digunakan sebagai tempat *inisialisasi variabel*, *pin mode*, penggunaan *library* dan lainnya. fungsi ini dijalankan sekali ketika *board* dinyalakan atau di *reset*. Berikut contoh dari void setup:

2. **loop()**

Setelah membuat fungsi setup() sebagai tempat *inisialisai variabel* dan menetapkan nilai maka selanjutnya fungsi loop() seperti namanya fungsi ini akan melakukan perulangan berturu-turut, memungkina program untuk mengubah dan menanggapi. digunakan untuk mengontrol *board* Arduino. Berikut contoh dari void loop:

```
int buttonPin = 3;

void setup()
{
    Serial.begin(9600);
    pinMode(buttonPin, INPUT);
}

void loop()
{
    // ...
}
```

Gambar 2.74 Contoh Void Setup

```
sketch_ianOla 5

int ledPin = 13;

void setup()
{
   pinNodu(ledPin, bUTPUT);
}

void loop()
{
   digitalWrite(ledPin, LOW);
}
```

Gambar 2.75 Gambar Contoh Fungsi Setup dan Loop

Pada gambar 2.4 merupakan suatu contoh fungsi void setup dan void loop pada pemrograman bahasa C menggunakan Arduino IDE.

3. Values

Berisi variabel atau konstanta sesuai dengan type data yang didukung oleh Arduino.

4. Function

Segmentasi kode ke fungsi memungkinkan programmer untuk membuat potonganpotongan modular kode yang melakukan tugas yang terdefinisi dan kemudian kembali ke asal kode dari mana fungsi itu dipanggil. Umumnya menggunakan fungsi adalah ketika salah satu kebutuhan untuk melakukan tindakan yang sama beberapapa kali dalam sebuah program.

5. Tipe Data Pada Arduino

Tipe data merupakan kelompok data berdasarkan jenis-jenis tertentu. Tipe data banyak dijumpai dalam berbagai bahasa pemrograman. Begitu juga dalam pemrograman Arduino. Tipe-tipe data yang digunakan dalam pemrograman Arduino antara lain adalah Void, Boolean, Char, Unsigned Char, Byte, Int, Unsigned Int, Word, Long, Unsigned Long, Short, Float, dan Double.

(a) Void

Kata kunci void digunakan dalam deklarasi fungsi. Kata kunci ini menandakan bahwa fungsi tersebut tidak mengembalikan informasi ke fungsi yang dipanggil. Seperti soure code dibawah ini :

```
Void Loop ( ) {
2 // kode program
3 }
```

(b) Boolean

Boolean menangani satu dari dua nilai yaitu, benar (true) atau salah (false). Setiap variabel Boolean menempati satu byte memori. Seperti soure code dibawah ini:

```
boolean val = false ;
boolean state = true ;
```

(c) Char

Tipe data Char adalah tipe data yang mengambil satu byte memori yang menyimpan suatu nilai karakter. Karakter harfiah ditulis dalam kutip tunggal, misalnya A dan untuk multi karakter digunakan tanda kutip ganda, seperti ABC.

Namun demikian, karakter-karakter tersebut disimpan sebagai angka. Hal ini dapat dilihat dengan jelas pada Tabel ASCII. Hal ini berarti bahwa melakukan operasi aritmatika pada karakter, di mana nilai karakter ASCII digunakan. Sebagai contoh, A + 1 memiliki nilai 66, karena nilai ASCII dari huruf kapital A adalah 65.

```
Char chr_r = r
Char chr_e = 97;
```

(d) Unsigned char

Unsigned char merupakan tipe data tak bertanda yang mencakup satu byte memori. Tipe data ini meng-encode bilangan dari 0 sampai 255.

```
Unsigned Char chr_y = 231;
```

(e) Byte

Byte menyimpan bilangan tak bertanda 8 bit, dari 0 sampai 255.

```
byte a = 75;
```

(f) INT

Integer adalah tipe data utama untuk penyimpanan bilangan. int menyimpan nilai 16-bit (2-byte). Rentang nilainya adalah kisaran -32.768 hingga 32.767 Ukuran int berbeda-beda setiap papan arduino. Sebagai contoh untuk Arduino Due, int menyimpan nilai 32-bit (4-byte). Rentang nilainya dari -2,147,483,648 sampai 2,147,483,647.

```
int counter = 92;
```

(g) Unsigned int

Unsigned int sama dengan int dalam cara mereka menyimpan nilai 2 byte. Unsigned int hanya menyimpan nilai positif, menghasilkan rentang dari 0 hingga 65.535. Arduino Due menyimpan nilai 4 byte (32-bit), mulai dari 0 hingga 4.294.967.295.

```
Unsigned int counter = 90;
```

(h) Long

Variabel dengan tipe Long merupakan penyimpanan bilangan dengan ukuran diperluas, dan menyimpan 32 bit (4 byte), dari -2,147,483,648 sampai 2,147,483,647.

```
Long velocity = 172346 ;
```

(i) Unsigned long

Unsigned Long merupakan tipe Long tak bertanda yang diperluas untuk penyimpanan bilangan dan menyimpan 32 bit (4 byte). Tidak seperti long standar, unsigned long tidak akan menyimpan angka negatif, rentangnya dari 0 hingga 4.294.967.295.

```
Unsigned Long velocity = 121016 ;
```

(j) Short

Shot adalah tipe data 16-bit. Pada semua Arduino (berbasis ATMega dan ARM), short menyimpan nilai 16-bit (2-byte). Rentang nilainya antara 32.768 hingga 32.767.

```
short val = 19 ;
```

(k) Float

Tipe data untuk bilangan floating-point adalah bilangan yang memiliki titik desimal. Angka floating-point sering digunakan untuk memperkirakan nilai analog dan kontinu karena memiliki resolusi lebih besar daripada bilangan bulat.

Bilangan floating-point dapat memiliki nilai maksimal 3,4028235E + 38 dan nilai minimal -3,4028235E + 38. Float disimpan sebagai informasi 32 bit (4 byte).

```
float num = 1.523 ;
```

(1) **Double**

Pada Arduino Uno dan arduino berbasis ATMEGA lainnya, bilangan floating-point presisi ganda menempati empat byte. Artinya, implementasi ganda persis sama dengan float, tanpa perolehan presisi. Pada Arduino Due, ganda memiliki presisi 8-byte (64 bit).

```
double num = 45.253; s
```

6. Variabel

Variabel dalam bahasa pemrograman C, dimana bahasa C ini digunakan dalam Arduino, memiliki suatu properti yang disebut dengan cakupan (*scope*). Suatu cakupan merupakan wilayah dari program dan ada tiga tempat dimana variabel dapat dideklarasikan. Ketiga tempat tersebut adalah sebagai berikut:

- (a) Di dalam fungsi atau blok, variabel ini disebut dengan variabel lokal (*local variabel*).
- (b) Di dalam definisi parameter fungsi, yang disebut dengan parameter formal (formal parameters).
- (c) Di luar semua fungsi, variabel ini disebut dengan variabel global (global variabel).

7. Variabel Local (Local variabel)

Sebagaimana disebutkan di atas, bahwa variabel lokal merupakan variabel yang dideklarasikan di dalam suatu fungsi atau blok. Variabel lokal ini hanya dapat digunakan oleh pernyataan (statement) yang berada di dalam fungsi atau blok kode. Berikut ini adalah contoh penggunaan dari variabel lokal.

```
Void setup () {

Void loop () {

int a , b ;

int c ; Local variable declaration

a = 0;

b = 0; actual initialization

c = 10;

}
```

8. Variabel Global (Global variabel)

Variabel global (*Global variabel*)didefinisikan di luar semua fungsi, biasanya pada bagian atas dari program. Variabel global akan menyimpan nilainya sepanjang program dijalankan (*life-time*).

```
Int A, R;
float j = 0; Global variable declaration

Void setup () {
```

```
7
8 Void loop () {
9    int x , y ;
10    int z ; Local variable declaration
11    x = 0;
12    y = 0; actual initialization
13    z = 10;
14 }
```

Variabel global dapat diakses oleh semua fungsi apapun. Artinya variabel global akan tersedia terus untuk digunakan di seluruh program setelah dideklarasikan.Berikut ini adalah contoh penggunaan dari variabel global dalam pemrograman Arduino.

PEMBUATAN PROTOTIPE

Pada bab ini akan membahas tentang pembuatan prototipe. Dimana pada pembuatan prototipe ini akan dibagi menjadi beberapa bagian mulai dari membuat rangka prototipe, membuat skema elektronik komponen, dan mengimplementasikan *coding*. Dalam tahapan pembuatan *prototype* ini tidak boleh ada salah satu *step* yang terlewatkan karena akan terjadi hal yang fatal.

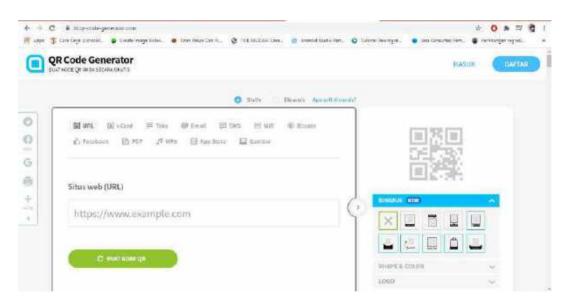
3.1 Membuat Kerangka Prototipe

Rangka atau model prototipe sangat penting karena desain rangka ini akan mempresentasikan tujuan dari prototipe tersebut. Jika membuat suatu kerangka atau model yang asal maka tidak akan mempresentasikan tujuan dari dibuatnya suatu prototipe itu. Untuk membuat rangka prototipe yaitu menggunakan bahan - bahan yang telah disebutkan pada bab sebelumnya . Berikut langkah - langkah pembuatannya :

Siapkan kayu , kemudian bentuk persegi empat seperti meja dengan ukuran 60 x 60 cm . Setelah dibentuk seperti meja pada bagian tengah - tengahnya di buat lubang dengan diameter 40 cm.

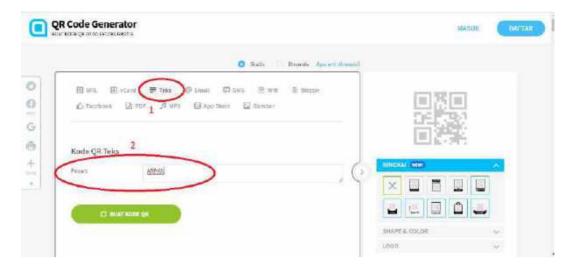
- 2. Kemudian setelah kayu sudah terbentuk persegi empat langkah selanjutnya membuat rumah rumahan , pohon dan *safety railing*. Buat sebanyak masing masing 8 buah.
- 3. Setelah selesai membuat rumah rumahan , pohon dan *safety railing* dari kardus , selanjutnya memberinya warna dengan cara menempelkan kertas warna pada kardus yang sudah dibentuk. Apabila tidak ingin menggunakan kertas warna dapat diganti dengan cat air dan untuk warnanya dapat sesuai keinginan.
- 4. Langkah selanjutnya yaitu memberi warna biru pada bagian tengah meja. warna biru pada bagian tengah ini akan dianalogikan sebagai sungai, kemudian untuk bagian pinggirnya akan diberikan *safety railing* yang telah dibuat sebelum.
- 5. Apabila sudah membuat semua item yang dibutuhkan selanjutnya yaitu menyimpanya di atas meja dan jangan lupa untuk direkatkan menggunakan lem kayu agar tidak lepas.
- 6. Selanjutnya membuat sebuah barcode yang akan ditempelkan di meja untuk augmented reality prototipe PKA. Untuk membuat *barcode* kita dapat membuatnya di situs onlie yaitu https://id.qr-code-generator.com/.

Untuk membuat sebuah QR code kita dapat membuatnya di situs online yaitu https://id.qr.-code-generator.com/ berikut langkah - langkahnya :



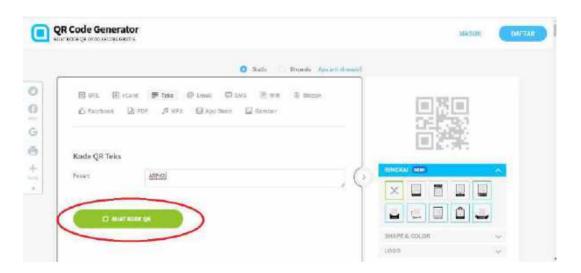
Gambar 3.1 Tampilan website QR Code Generator

Setelah muncul tampilan tersebut kita dapat memilih *type* QR code beberapa diantaranya yaitu URL, Vcard, teks, email, SMS, WIFI. Dalam pembuatan QR code untuk augmented reality ini menggunakan *type* teks dengan cara klik pada bagian teks kemudian masukan yang diinginkan seperti berikut:



Gambar 3.2 Memilih type QR code

Apabila telah mengikuti langkah sebelumnya maka kita buat QR code tersebut dengan cara mengklik buat kode QR.



Gambar 3.3 Membuat QR code

Setelah QR code terbuat kita dapat memilih bingkai barcode sesuai keinginan kemudian untuk mendapatkan hasilnya dengan cara klik unduh.



Gambar 3.4 Mengunduh QR code

Jika sudah di Download maka hasil barcodenya seperti berikut :



Gambar 3.5 Hasil unduh QR code

Kemudian jika telah semuanya selesai print barcode tersebut dengan ukuran 10x10xm dan tempelkan pada meja yang telah dibuat sebelumnya. maka hasilnya akan seperti berikut :



Gambar 3.6 Tampilan Keseluruhan Prototipe



Gambar 3.7 Tampilan Keseluruhan Prototipe

3.2 Membuat Skema Elektronik Komponen

Tahapan selanjutnya adalah membuat atau merakit komponen untuk prototipe. Tetapi ada satu hal sebelum kita masuk ke proses perakitan yaitu terlebih dahulu harus membuat sebuah rangkaian atau skema komponen. Dibuatnya suatu rangkaian atau skema ini untuk mempermudah pada saat proses perakitan komponen. Sehingga rangkaian komponen tersebut sebagai acuan.

Untuk membuat skema komponen prototipe prediksi ketinggian air (PKA) untuk mendeteksi banjir peringatan dini ini menggunakan software frizting. Dikarenakan software frizting ini sangat mudah dipahami. Selain itu komponen yang tersedia cukup banyak sehingga tidak perlu lagi untuk mengimport komponen dan salah satu keuntungan menggunakan frizting ini yaitu software ini free.

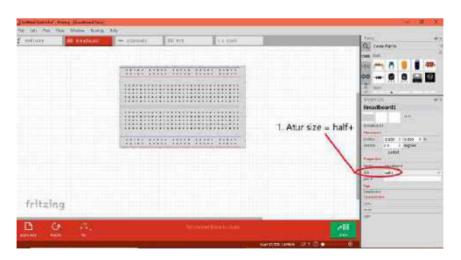
Berikut langkah-langkah untuk membuat skema prototipe prediksi ketinggian air (PKA) untuk mendeteksi banjir peringatan dini menggunakan software frizting :

1. Buka software frizting kemudian pilih breadboard. Seperti pada gambar dibawah ini :



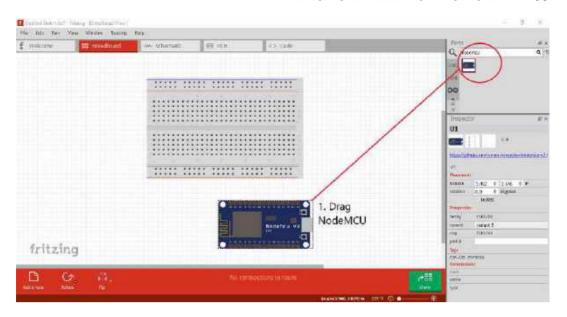
Gambar 3.8 Tampilan Awal Frizting

2. Atur ukuran atau *size breadboard* menjadi ukuran kecil karena pada pembuatan skema ini tidak memerlukan space yang besar.Dengan cara ubah ukuran menjadi *half*+ pada bagian *size* seperti berikut:



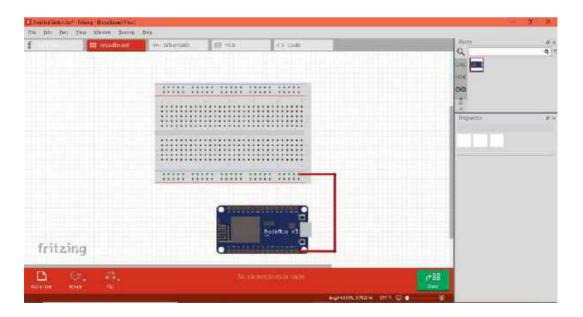
Gambar 3.9 Ubah Ukuran Breadboad

3. Cari komponen NodeMCU pada bagian *search parts* kemudian drag NodeMCU ke dekat *breadboard*.



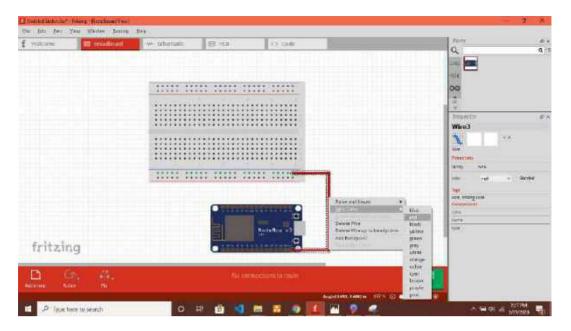
Gambar 3.10 Memasukan NodeMCU

4. Setelah itu sambungkan Vin pada bagian positif (+) *breadboard* dengan cara klik dua kali pada bagian Vin maka akan muncul *wire* dan arahkan ke bagian positif *breadboard*.



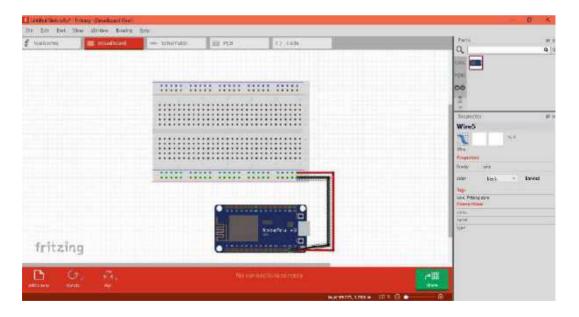
Gambar 3.11 Sambungkan Vin ke *Breadboard*

5. Untuk mengubah warna wire untuk menjadi merah dengan cara klik kanan pada wire. Kemudian pilih *wire colour* dan pilih warna merah atau *red*.



Gambar 3.12 Mengubah Warna Wire

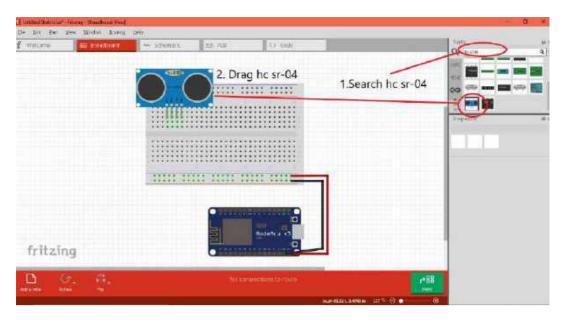
6. Sambungkan GND pada bagian negatif (-) *breadboard* dengan cara klik dua kali pada bagian Vin maka akan muncul *wire* dan arahkan ke bagian negatif *breadboard*.



Gambar 3.13 Sambungkan GND ke *Breadboard*

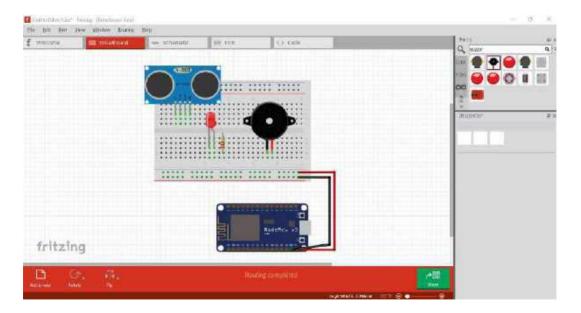
Untuk mengubah warna wirenya sama seperti langkah sebelumnya.

7. Masukan komponen sensor ultrasonic dengan cara ketik hc sr-04 pada bagian *search parts*. Setelah itu drag sensor ke *breadboard*



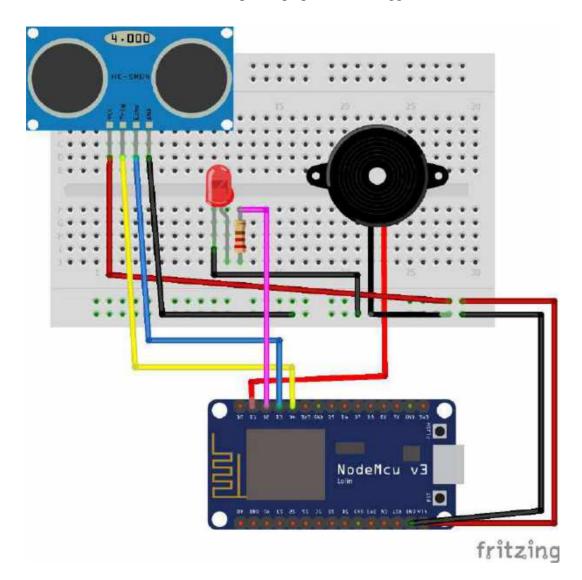
Gambar 3.14 Menambahkan Komponen Sensor Ultrasonic

8. Tambahkan komponen led, resistor, dan buzzer dan drag ke dalam *breadboard* seperti gambar dibawah :



Gambar 3.15 Menambahkan Komponen Led, Resistor dan Buzzer

Berikut hasil akhir dari skema prototipe prediksi ketinggian air (PKA):



Gambar 3.16 Rangkaian Alat

Agar mempermudah pada saat proses perakitan komponen maka harus mengikuti gambar rangkaian atau skema yang telah dibuat sebelumnya. Untuk penjelasannya sebagai berikut :

- 1. Sambungkan Vin NodeMCU menggunakan kabel jumper pada *breadboard* bagi an positif (+). Vin merupakan Power atau VCC dengan memiliki daya 5V. Jadi jika tidak ada 5V dalam sebuah *microcontroller* dapat digantikan dengan vin.
- 2. Sambungkan GND (*ground*) NodeMCU menggunakan kabel jumper pada *bread-board* bagian *negatife* (-). GND merupakan sistem pengamanan pada instalasi listrik dimana jika terjadi kebocoran listrik .

- 3. pin vcc yang berada di sensor ultrasonic US 100 hubungkan pada *breadboard* bagian positif (+) menggunakan kabel jumper yang sebelumnya sudah di sambungkan dengan vin pada NodeMCU. Sehingga jika kita colokan di bagian positif pada *breadboard* otomatis akan sudah teraliri vin.
- 4. pin GND yang berada di sensor ultrasonic US 100 hubungkan pada *breadboard* bagian negatife (-) menggunakan kabel jumper yang sebelumnya sudah di sambungkan dengan GND pada NodeMCU. Sehingga jika kita colokan di bagian negatif pada *breadboard* otomatis akan sudah teraliri GND.
- 5. pin echo pada sensor ultrasonic US 100 hubungkan pada D3 NodeMCU menggunakan kabel jumper.
- 6. pin trig pada sensor ultrasonic US 100 hubungkan pada D4 NodeMCU menggunakan kabel jumper.
- 7. Letakan atau colokan led pada *breadboard*. Pada bagian positif (+) led dihubungkan dengan resistor dan pin D2. Kemudian untuk bagan negatif (-) dihubungkan ke bagian GND.
- 8. Letakan atau colokan buzzer pada *breadboard*. Pada bagian Positif (+) hubungkan dengan D1. Kemudian untuk bagian negatif (-) dihubungkan ke bagian GND.

Setelah Kita mengikuti langkah-langkah tersebut maka simpan kompen pada rangka prototipe. Komponen yang sudah dirangkai di letakan di bagian belakang agar terlihat lebih rapi seperti berikut ini:



Gambar 3.17 Tampilan Prototipe

3.3 Implementasi Coding

implementasi menurut KBBI (Kamus Besar Bahasa Indonesia) yaitu pelaksanaan / penerapan. Sedangkan pengertian umum adalah suatu tindakan atau pelaksana rencana yang telah disusun secara cermat dan rinci (matang).Kata implementasi sendiri berasal dari bahasa Inggris to implement artinya mengimplementasikan. Tak hanya sekedar aktivitas, implementasi merupakan suatu kegiatan yang direncanakan serta dilaksanakan dengan serius juga mengacu pada norma-norma tertentu guna mencapai tujuan kegiatan.[7]

Sedangkan Coding adalah menulis sekumpulan code sesuai syntax (aturan penulis an) tergantung bahasa pemrograman yang dipakai (C,python, php, ruby, java, atau yang lainya) dengan bantuan text editor seperti sublime text, atom, notepad, dll. Dengan coding, kita memberikan daftar instruksi kepada komputer sesuai tujuan kita membuat. Contoh coding yaitu kita melakukan penulisan code untuk membuat website, aplikasi android, dan lain sebagainya.

Dalam mengimplementasikan kodingan untuk membuat prototipe prediksi ketinggian air (PKA) menggunakan bahasa pemrograman C . Bahasa pemrograman C ini telah dijelaskan pada bab sebelumnya. Jika tidak ingin menggunakan bahasa C untuk salah satu alternatifnya yaitu menggunakan bahasa pemrograman python.

Untuk menulis *coding* atau untuk membuat programnya menggunakan sebuah *software* IDE yaitu *software* Arduino IDE. Maka terlebih dahulu harus menginstall *software* Arduino IDE. Untuk mengunduh *software* Arduino IDE dengan cara mengunjungi https://www.arduino.cc/en/main/software. Kemudian unduh sesuai versi OS yang digunakan pada laptop. Setelah mengunduhnya kita harus *install* apk arduino IDE untuk cara *install*nya sudah dijelaskan pada buku versi ke 1.

3.3.1 Implementasi Coding Pada Arduino IDE

Setelah berhasil menginstall *software* Arduino IDE , selanjutnya membuat program untuk prototipe PKA. Tetapi sebelum masuk ke dalam proses memprogram sebaiknya mempelajari terlebih dahulu bahasa pemrograman C pada Arduino IDE seperti berikut ini :

 Hal pertama kali yaitu buka software Arduino IDE, kemudian pilih File - Examples - Basic - Blink . Maka akan muncul pada jendela yang berisi source sebagai berikut:

```
Blink | Arduino 1.6.5
                                                               X
File Edit Sketch Tools Help
  the documentation at http://www.arduing.oc
  This example code is in the public domain.
  modified 8 May 2014
  by Scott Fitsgerald
// the setup function runs once when you press reset or power the board
void setup() (
  // initialize digital pin 13 as an output.
  pinMode (13, 001907);
// the loop function runs over and over again forever
void loop() (
  digitalWrite (13, HIGH): // turn the LED on (HIGH is the voltage level)
  delay(1000);
                           // wait for a second
  digitalWrite(13, LOW): // turn the LED off by making the voltage LOW
  delay(1000);
                           // wait for a second
```

Gambar 3.18 Tampilan Arduino IDE Blink

Program ini memerintahkan Arduino untuk memberi sinyal digital bergantian dari 0 V dan 5V pada port 13.Port 13 Arduino Uno, terdapat LED orange kecil yang tertanam dalam board sehingga untuk mengetes apakah arduino kita berfungsi atau tidak, cukup program port 13 untuk mengirimkan sinyal digital dan lihat hasilnya secara visual pada LED tersebut.

2. Sebelum mengupload program kepada board, pilih port usb terlebih dahulu dimana Arduino Uno terbaca. Pilih Tools ¿ Port untuk menyambungkan port.

- 3. Setelah koneksi ke port serial sudah benar, klik tombol *Upload* pada Arduino IDE. dan tunggu beberapa saat hingga muncul pesan *Done Uploading* pada status bar. Jika berhasil, maka LED orange arduino Uno akan berkedip-kedip dengan delay tertentu.
- 4. Setelah Blink selanjutnya akan membuat "Hello world" menggunakan Arduino IDE yang nantinya akan ditampilkan di serial monitor. Berikut code pemrogramannya:

```
// Test Program Dasar Serial Monitor Menggunakan Arduino
// by Fungky King
void setup()
{
Serial.begin(9600);
}

void loop()
{
Serial.print("Hello World");
delay(2000);
}
```

Setelah program di atas diketikkan pada software Arduino IDE, maka tahap selanjutnya lakukan proses *Upload* ke dalam board Arduinonya tunggu sampai proses selesai sampai ada tanda *Done Uploading*. Untuk melihat hasilnya klik menu Tools kemudian Serial Monitor maka hasilnya akan muncul seperti gambar berikut.



Gambar 3.19 Tampilan Di Serial Monitor

Penjelasan tentang fungsi-fungsi dari tiap-tiap bagian program di atas adalah sebagai berikut: [8]

1. Void Setup() adalah fungsi yang dijalankan secara otomatis pertama kali oleh board Arduino, dimana Semua kode program yang ada dalam void setup akan dibaca sekali oleh Arduino. Biasanya isinya berupa kode perintah untuk menentukan fungsi pada sebuah pin atau deklarasi INPUT/OUTPUT.

- 2. begin() digunakan untuk mengatur baudrate / kecepatan transmisi data. Beberapa pilihan kecepatan komunikasi data yang dapat digunakan pada board arduino adalah 300, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 38400, 57600 atau 115200. Pengaturan baudrate dilakukan pada bagian setup().
- 3. Void loop() melakukan proses dimana semua kode yang ada disini akan dibaca berulang kali (terus menerus) oleh Arduino.
- 4. Perintah Serial.print digunakan untuk menampilkan data ke serial monitor. Data yang ditampilkan dapat berupa karakter, bytes, atau angka.
- 5. delay(2000) merupakan pernyataan untuk melakukan penundaan selama 2000 milidetik atau 2 detik dari proses akhir pembacaan program sebelumnya dimana Arduino akan mengulang proses pembacaan program dari awal kembali.

3.3.2 Cara Import Library

Library adalah kumpulan code yang biasanya terkumpul dalam sebuah namespace/ module/ package (tergantung anda menggunakannya di bahasa pemrograman apa) yang dapat di import/ reuse ke program lain. Untuk mengimport library ada tiga cara yaitu menggunakan *manager library*, mengimport library .zip hasil Download dan Instalasi Manual.

1. Import Library Menggunakan Manager

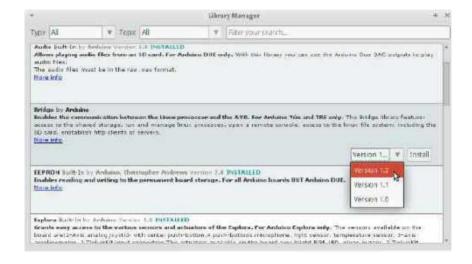
Untuk menginstal sebuah Library baru ke dalam IDE Arduino,dapat menggunakan Library Manager (tersedia sejak Arduino IDE versi 1.6.2). Buka Arduino IDE dan klik ke menu Sketch dan kemudian Include library ¿ Manage Libraries





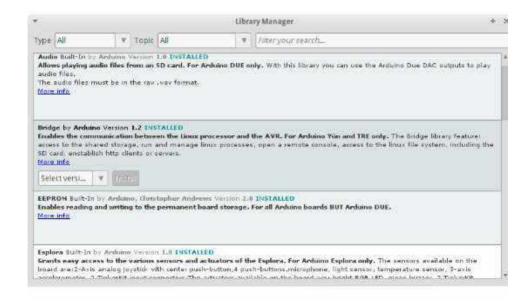
Gambar 3.20 Memilih Menu Manage Libraries

Kemudian Sketch Manager akan terbuka dan akan menemukan daftar Library yang sudah terpasang ataupun Library baru yang siap untuk di instalasi. Dalam contoh ini kita akan menginstal library Bridge. Gulir daftar untuk menemukannya, lalu pilih versi Library yang ingin Anda instal. (beberapa Library terkadang hanya ada satu versi Library yang tersedia).



Gambar 3.21 Tampilan Sketch Manager

Langkah selanjutnya adalah klik instal dan tunggu IDE untuk menginstal library baru. Setelah selesai, sebuah tag Installed akan muncul di sebelah Library Bridge. Setelah terinstal, sekarang kita dapat menemukan Library baru yang tersedia di menu Include Library.

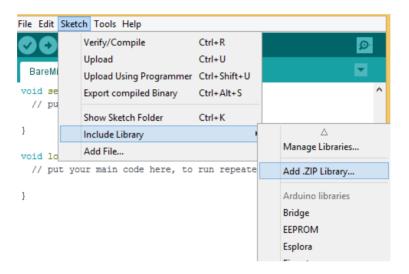


Gambar 3.22 Tampilan Library Yang Terinstall

2. Mengimport Library .zip hasil Download.

Terkadang Library hasil buatan seseorang banyak dan sering yang didistribusikan sebagai file ZIP atau folder sehingga dapat kita unduh, di GitHub misalnya. Nama folder adalah nama Library. Di dalam folder tersebut akan ada file .cpp, file .h , Folder Contoh Sketch, dan file lainnya yang dibutuhkan oleh Library. Dimulai dengan versi 1.0.5, Kita bisa menginstal library pihak ke-3 di IDE. Jangan unzip Library yang telah didownload, biarkan seperti apa adanya.

Buka aplikasi Arduinonya, lalu Masuk ke menu SKETCH, pilih INCLUDE LIBRARY, pilih ADD. ZIP Library, seperti gambar berikut:



Gambar 3.23 Tampilan Library ADD ZIP

Kemudian akan diminta untuk memilih Library yang ingin ditambahkan. klik add .ZIP library, lalu cari file ZIP yang sudah diDownload.



Gambar 3.24 Pilih File ZIP

Jika berhasil, aplikasi Arduino kamu akan muncul keterangan seperti dibawah ini:



Gambar 3.25 Tampilan Library Berhasil Ditambahkan

3. Instalasi Manual

Bila ingin menambahkan Library secara manual, perlu mendownload sebagai file ZIP, lalu extract file tersebut. (Klik Kanan pada File ZIP lalu pilih extract to nama file), setelah folder hasil extrakan tersedia, selanjutnya kita harus memasukannya ke dalam folder di C:User-NamaPC-Documents-Arduino-libraries.

3.3.3 Membuat Program Prototipe PKA

Pembahasan mengenai contoh *coding* pada arduino membuat kita mengetahui cara bagaimana menggunakan Arduino IDE, *import library*, dan struktur pemrogramannya. Selanjutnya akan memprogram komponen untuk *prototype* PKA ini. Dengan cara memprogram serta testing komponen satu persatu. Mengapa kita harus memprogram satu persatu terlebih dahulu tidak langsung? karena jika adanya error pada saat proses penggabungan komponen akan mengetahui komponen manakah yang error.kemudian jika sudah sesuai yang diharapkan digabungkan menjadi kesatuan yang utuh.

1. Pertama yang akan di program adalah sensor ultrasonic yang berfungsi membaca jarak ketinggian air. Berikut kodinganya:

```
#define triggerPin D4
#define echoPin D3

void setup() {
    Serial.begin (9600);
    pinMode(triggerPin, OUTPUT);
    pinMode(echoPin, INPUT);

void loop() {
    long duration, jarak;
    digitalWrite(triggerPin, LOW);
    delayMicroseconds(2);
```

```
digitalWrite(triggerPin, HIGH);
    delayMicroseconds (10);
14
    digitalWrite(triggerPin, LOW);
15
    duration = pulseIn(echoPin, HIGH);
    jarak = (duration/2) / 29.1;
17
    Serial.println("jarak:");
18
    Serial.print(jarak);
    Serial.println(" cm");
20
    delay (1000);
21
22 }
```

Penjelasan coding di atas yaitu:

(a) define triggerPin D4 define echoPin D3

Script tersebut untuk menginisiasi pin yang akan digunakan. Pin D4 digunakan untuk pin trigger dan D3 digunakan untuk pin echo.

(b) Serial.begin (9600); ,pinMode(triggerPin, OUTPUT); ,pinMode(echoPin, INPUT);

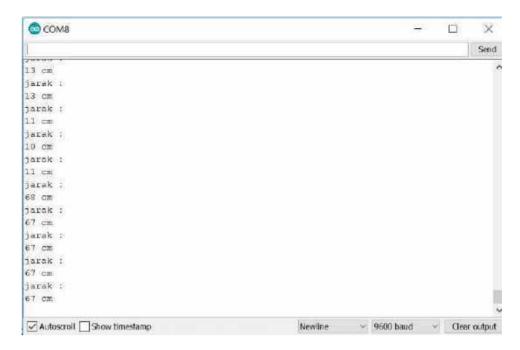
Script di atas berada di fungsi void setup(). Serial.begin() digunakan untuk memulai serial. selanjutnya pinMode digunakan untuk menentukan fungsi dari pin yang akan digunakan (triggerPin dan echoPin). triggerPin berfungsi sebagai pin output sedangkan echoPin sebagai pin input.

(c) long duration, jarak;

Script di atas digunakan untuk mendeklarasikan variabel duration dan jarak. Duration dan jarak merupakan variabel bertipe long. Long merupakan tipe data serupa dengan int tetapi memiliki range yang lebih panjang.

- (d) digitalWrite(triggerPin, LOW);, delayMicroseconds(2);, digitalWrite(triggerPin, HIGH);, delayMicroseconds(10); digitalWrite(triggerPin, LOW);
 Script tersebut menjelaskan bahwa pin trigger menembakkan pulsa sinyal ultrasonik selama 10 microsecond.
- (e) duration = pulseIn(echoPin, HIGH); Selanjutnya ketika gelombang ultrasonik itu ditangkap kembali setiap pulsanya oleh pin echo maka waktu tangkap tersebut diubah menjadi variabel duration.
- (f) jarak = (duration/2) / 29.1;Dengan persamaan di atas nilai waktu duration akan dikonversi menjadi nilai jarak dalam bentuk cm.
- (g) Serial.println("jarak:");, Serial.print(jarak);, Serial.println("cm");, delay(1000); Selanjutnya nilai jarak yang didapatkan akan ditampilkan di serial monitor setiap 1 detik (delay(1000)).

Untuk melihat hasil pembacaan sensor lewat serial monitor pada arduino IDE. Jika tidak ada error maka akan seperti gambar berikut :



Gambar 3.26 Tampilan serial monitor

2. Komponen kedua yang akan di program yaitu led yang berfungsi sebagai indikator.Berikut Kodingannya:

```
const int ledPin = D2;
void setup() {
pinMode(2, OUTPUT);

void loop() {
digitalWrite(2, LOW);
delay(1000);
digitalWrite(2, HIGH);
delay(2000);
}
```

Penjelasan coding di atas yaitu:

- (a) const int ledPin = D2;Script tersebut untuk menginisiasi pin yang akan digunakan.
- (b) pinMode(2, OUTPUT);Menetapkan pin GPIO 2/LED sebagai OUTPUT.
- (c) digitalWrite(2, LOW);
 Perintah untuk mematikan Lampu atau memberikan nilai LOW(0) pada pin GPIO 2.
- (d) delay(1000); Matikan lampu selama 1 detik.
- (e) digitalWrite(2, HIGH); Perintah memberikan nilai HIGH atau menyalakan Lampu.

- (f) delay(2000);
 - Nyalakan lampu selama 2 detik.
- 3. Komponen ketiga yang akan di program yaitu buzzer yang berfungsi sebagai indikator.Berikut Kodingannya:

```
const int ledPin = D1;
void setup() {
pinMode(1, OUTPUT);
void loop() {
digitalWrite(1, LOW);
delay(1000);
digitalWrite(1, HIGH);
delay(2000);
}
```

Penjelasan coding di atas yaitu:

- (a) const int ledPin = D1;
 - Script tersebut untuk menginisiasi pin yang akan digunakan.
- (b) pinMode(1, OUTPUT);

Menetapkan pin GPIO 1/Buzzer sebagai OUTPUT.

(c) digitalWrite(2, LOW);

Perintah untuk mematikan Buzzer atau memberikan nilai LOW(0) pada pin GPIO 1.

(d) delay(1000);

Matikan buzzer selama 1 detik.

(e) digitalWrite(1, HIGH);

Perintah memberikan nilai HIGH atau menyalakan buzzer.

(f) delay(2000);

Nyalakan buzzer selama 2 detik.

4. Langkah selanjutnya yaitu menggabungkan sensor, led , buzzer terlebih dahulu sebelum ditambahkan metode regresi linear sederhana. Berikut kodingan dari komponen yang digabungkan :

```
// defines pins numbers

const int trigPin = 2; //D4

const int echoPin = 0; //D3

const int ledPin = 4; //D2

const int buzzerPin = 5; //D1

// defines variables
long duration;
int distance;
int safetyDistance;
```

```
void setup() {
18 pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input
pinMode(buzzerPin, OUTPUT);
  pinMode(ledPin , OUTPUT);
  Serial.begin(115200); // Starts the serial communication
23
  void loop() {
25
26
  // Clears the trigPin
28 digitalWrite (trigPin, LOW);
  delayMicroseconds (2);
31 // Sets the trigPin on HIGH state for 10 micro seconds
32 digitalWrite(trigPin, HIGH);
delay Microseconds (10);
  digitalWrite (trigPin, LOW);
  // Reads the echoPin, returns the sound wave travel time in
      microseconds
  duration = pulseIn (echoPin, HIGH);
 // Calculating the distance
distance = duration *0.034/2:
41 // Prints the distance on the Serial Monitor
42 // Serial.print("Distance: ");
43 // Serial.println(distance);
  // delay (2000);
  safetyDistance = distance;
  if (safetyDistance <= 5){
    digitalWrite(buzzerPin, HIGH);
    digitalWrite(ledPin, HIGH);
49
  }
51
  else {
    digitalWrite (buzzerPin, LOW);
52
53
    digitalWrite(ledPin, LOW);
55
56 // Prints the distance on the Serial Monitor
Serial.print("Distance: ");
58 Serial. println (distance);
59 Serial.print("pulseIn");
60 Serial.print("duration");
61 }
```

Penjelasan coding di atas yaitu:

(a) const int trigPin = 2; //D4 , const int echoPin = 0; //D3 , const int ledPin = 4; //D2, const int buzzerPin = 5; //D1

- Script tersebut untuk menginisiasi pin yang akan digunakan. Pin D4 digunakan untuk pin trigger,D3 digunakan untuk pin echo, D2 digunakan untuk pin led, dan D1 digunakan untuk buzzerpin.
- (b) // defines variables , long duration; , int distance;, int safetyDistance; Script di atas merupakan sebuah pendefinisian variabel.
- (c) pinMode(trigPin, OUTPUT);, pinMode(echoPin, INPUT); , pinMode(buzzerPin, OUTPUT); , pinMode(ledPin, OUTPUT);, Serial.begin(115200);
 Pada script tersebut merupakan sebuah bagian void setup yang berfungsi sebagai output dan input dari komponen. Selain itu pada script tersebut adanya baudrate yang digunakan 115200 yang berfungsi untuk komunikasi serial.
- (d) digitalWrite(trigPin, LOW); , delayMicroseconds(2); Script di atas merupakan sebuah perintah untuk mematikan (*Low* pada trigpin disensor ultrasonic.
- (e) digitalWrite(trigPin, HIGH , delayMicroseconds(10); , digitalWrite(trigPin, LOW);
 - Pada bagian script di atas merupakan sebuah perintah untuk trigpin pada sensor ultrasonic dan akan adanya jeda sebesar 10 detik.
- (f) safetyDistance = distance;// if (safetyDistance ¡= 5)// digitalWrite(buzzerPin, HIGH);// digitalWrite(ledPin, HIGH);// // else// digitalWrite(buzzerPin, LOW);// digitalWrite(ledPin, LOW);// Script tersebut merupakan sebuah logika dari sensor ultrasonic. Dimana jika jarak yang dibaca oleh sensor ultrasonic adalah kurang dari sama dengan 5 maka buzzer dan led akan menyala dan jika tidak membaca jarak kurang sama dengan 5 buzzer dan led mati.
- (g) Serial.print("Distance: "); , Serial.println(distance); , Serial.print("pulseIn "); , Serial.print("duration "); ,Bagian script tersebut merupakan sebuah komunikasi serial yang nantinya

jarak yang dibaca oleh sensor akan ditampilkan di serial monitor.

Jika sensor ultrasonic, led dan buzzer sudah digabungkan, maka langkah selanjutnya yaitu menambahkan sebuah metode regresi linear sederhana. Regresi adalah suatu metode analisis statistik yang digunakan untuk melihat pengaruh antara dua atau lebih banyak variabel. Hubungan variabel tersebut bersifat fungsional yang diwujudkan dalam suatu model matematis. Pada analisis regresi, variabel dibedakan menjadi dua bagian, yaitu variabel respons (response variable) atau biasa juga disebut variabel bergantung (dependent variable), dan variabel explanatory atau biasa disebut penduga (predictor variable) atau disebut juga variabel bebas (independent variable). Regresi terbagi menjadi beberapa jenis, yaitu regresi sederhana (linier sederhana dan nonlinier sederhana) dan regresi berganda (linier berganda atau nonlinier berganda).

Regresi linear sederhana adalah suatu perhitungan statistik yang menggunakan satu buah variabel bebas dan satu buah variabel terikat. Variabel bebas biasanya dilambangkan dengan simbol X dan variabel terikat dilambangkan dengan simbol Y. Untuk nilai prediksi dari variabel terikat di lambangkan dengan simbol . Untuk mendapatkan nilai prediksi dari variabel terikat terdapat beberapa langkah yang harus dilakukan. Berikut kodingan yang telah ditambahkan metode regresi linear sederhana:

```
// defines pins numbers
4 const int trigPin = 2;
                            //D4
  const int echoPin = 0;
                           //D3
  const int ledPin = 4; //D2
  const int buzzerPin = 5; //D1
  // defines variables
  long duration;
  int distance;
  int safety Distance;
14
  float regresi_a;
  float regresi_b;
18
  void setup() {
19
20
21
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input
  pinMode(buzzerPin, OUTPUT);
  pinMode(ledPin , OUTPUT);
  Serial.begin(115200); // Starts the serial communication
27
  }
28
  void loop() {
29
30
  // Clears the trigPin
31
  digitalWrite(trigPin, LOW);
  delayMicroseconds (2);
34
  // Sets the trigPin on HIGH state for 10 micro seconds
35
  digitalWrite(trigPin, HIGH);
  delayMicroseconds (10);
  digitalWrite(trigPin, LOW);
39
  // Reads the echoPin, returns the sound wave travel time in
40
      microseconds
  duration = pulseIn (echoPin, HIGH);
  // Calculating the distance
43
44 distance = duration *0.034/2;
  // Prints the distance on the Serial Monitor
46 // Serial.print("Distance: ");
```

```
// Serial . println (distance);
  // delay (2000);
   safetyDistance = distance;
  if (safetyDistance <= 5){
     digitalWrite(buzzerPin, HIGH);
52
     digitalWrite(ledPin, HIGH);
53
  }
54
   else {
55
     digital Write (buzzerPin, LOW);
56
     digitalWrite (ledPin, LOW);
57
58
59
  // Prints the distance on the Serial Monitor
  Serial.print("Distance: ");
   Serial.println(distance);
   Serial.print("pulseIn");
   Serial.print("duration");
65
  void hitungParameterRegresi(float &a, float &b)
67
     float X = 0;
68
     float Y = 0;
69
     float X2 = 0;
70
     float Y2 = 0;
     float XY = 0;
     int jumlahData = sizeof(data) / sizeof(Data);
74
75
     for (int i = 1; i < jumlahData; i++)
76
       float x = abs(data[i]. distance - data[i - 1]. distance) / data[i].
78
       selisihWaktu;
       float y = data[i]. selisihWaktu;
79
80
       X += x;
81
       Y += y;
82
83
       X2 += (x * x);
       Y2 += (y * y);
       XY += x * y;
85
     }
86
87
     a = ((Y * X2) - (X * XY)) / ((jumlahData * X2) - (X * X));
     b = ((jumlahData * XY) - (X * Y)) / ((jumlahData * X2) - (X * X));
89
90
     Serial . print ("X=");
91
     Serial. println(X);
92
     Serial. print("Y=");
93
     Serial.println(Y);
94
     Serial.print("X2=");
95
     Serial.println(X2);
     Serial.print("Y2=");
97
     Serial.println(Y2);
98
     Serial.print("XY=");
99
     Serial. println(XY);
100
     Serial.print("a=");
101
```

Pada *coding* di atas hanya ditambahkan beberapa yaitu pada baris ke 35 ,36,66 sampai dengan 104. Berikut penjelasannya :

- script float regresi a dan float regresi b merupakan sebuah pendefinisian variabel dimana a merupakan sebuah nilai konstanta dan nilai b merupakan koefisien variabel x .
- 2. untuk script dari 66 sampai 89 merupakan sebuah perhitungan regresi linearnya kemudian untuk script 91 sampai dengan 104 merupakan sebuah perintah untuk menampilkan nilai a dan nilai b pada serial monitor.

3.3.4 Menghubungkan Prototipe PKA Pada Antares

Langkah selanjutnya yaitu menghubungkan alat atau semua komponen prototipe PKA kepada Antares. Fungsi dihubungkan kepada antares yaitu agar data jarak ketinggian air yang dibaca oleh sensor ultrasonic dapat disimpan di antares . Sebelum memprogram ada langkah-langkah yang harus dilakukan untuk tahapan antares. Tahapan ini sudah dijelaskan pada buku versi untuk lebih detailnya berikut penjelasanya :

1. Registrasi di Antares menggunakan gmail. Untuk membuka halaman antares cukup ketikan https://antares.id/id/index.html pada browser maka akan muncul tampilan seperti gambar berikut.



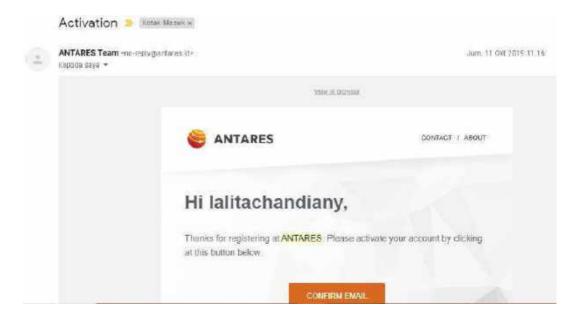
Gambar 3.27 Halaman Awal Antares

jika halaman tersebut muncul, pilih Signup kemudian masukan email yang aktif.

Re	gister	
	your account	
Ω	Lalita	
@	xxxxxxxx@gmail.com	
白	*******	
Ġ)		
+62	XXXXXXXXXXXX	
9	100000000000	

Gambar 3.28 Tampilan Registrasi Antares

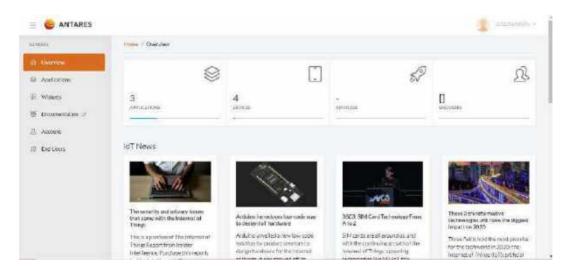
Masukan nama , alamat email (alamat email aktif), passworld, no telepon dan alamat. Alamat email harus aktif karena ini berguna untuk *verifikasi account* dan apabila ada info mengenai antares akan dikirmkan melalui email.



Gambar 3.29 Verifikasi email

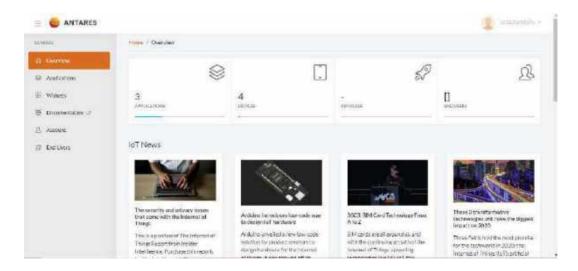
Setelah melakukan registrasi di antares maka akan adanya email *verifikasi* pada email yang sebelumnya email tersebut di daftarkan. Untuk mengkonfirmasi *verifikasi* dari pihak antares cukup klik *confirm*

Kemudian login di antares dengan mamasukan email yang telah didaftarkan dan memasukan passworld yang sebelumnya telah dibuat.



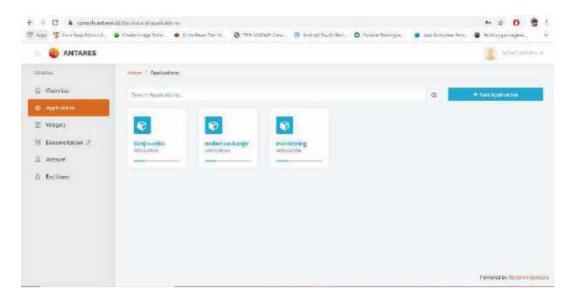
Gambar 3.30 Tampilan Awal Antares

Gambar di atas merupakan halaman awal antares ketika kita sudah login.



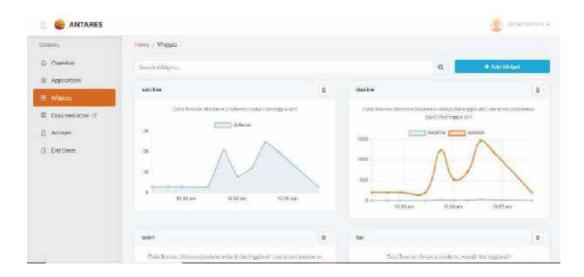
Gambar 3.31 Halaman Overview

Halaman *Overview* adalah halaman awal yang ditampilkan pada saat setelah login. Halaman ini menampilkan sebuah berita mengenai IoT (*Internet of Thing*). Selain mengenai berita IoT halaman ini juga menampikan project yang sudah kita buat. Adanya *application*, *device*, *API useage dan End User*.



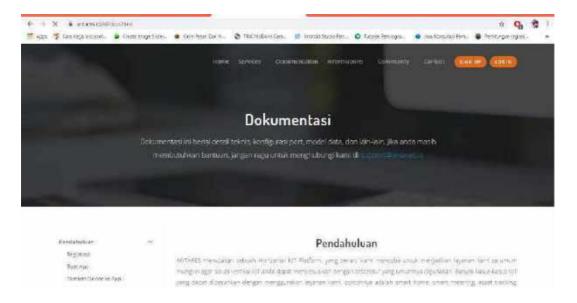
Gambar 3.32 Halaman application

Pada halaman *application* menampilkan *project* yang telah dibuat. Kemudian pada halaman ini kita dapat membuat sebuah project baru.

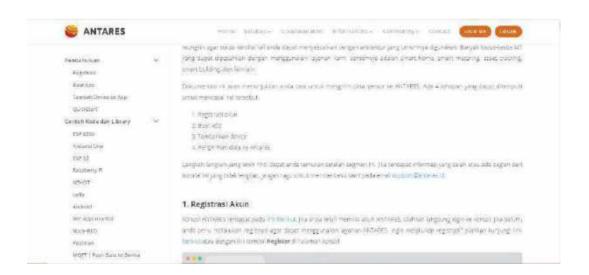


Gambar 3.33 Halaman Widgets

Halaman widget merupakan sebuah halaman yang menampilkan sebuah grafik yang telah dibuat sebelumnya. Selain menampilkan sebuah grafik dalam halaman ini dapat membuat sebuah grafik baru.



Gambar 3.34 Halaman *Documentation*

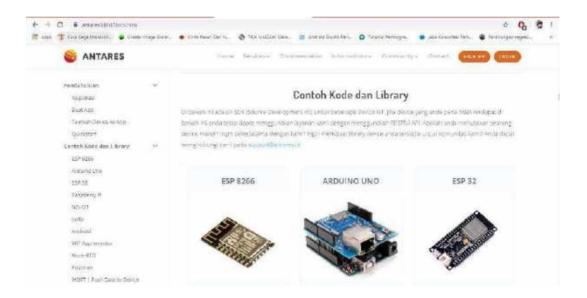


Gambar 3.35 Halaman Documentation

Halaman *Documentation* ini merupakan sebuah halaman yang menampilkan tutorial atau penjelasan mengenai penggunaan antares.

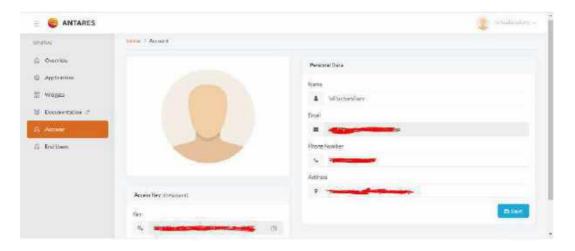
```
6- - C W antensation/document
ANTARES
                                                                                                                                                                                                                                                              Home Senters Description Statements Committy Zenics (Howell Comm
                                         Detalahalaan
                                                     Espidino
                                                     Burnace
                                                     Carron Decine to Pp.D.
                                                                                                                                                                                                                                          And when Property statement and property and
                                                                                                                                                                                                                                              cut structy t
Serial Sepur(11500);
setters.infilding(free);
setters.infilding(free);
setters.infilding(free);
                                                     157.92
                                                     Republican IN
                                                     MENT
                                                     tofu
                                                                                                                                                                                                                                                  enteres and "temperature", temp);
uncares one; "menuity", homy;
                                                      ME sponessio
                                                                                                                                                                                                                                                              previocal(projecthwe, devicebase);
                                                      Market 80
                                                      evel dutil first test (199)
```

Gambar 3.36 Halaman Documentation Tutorial



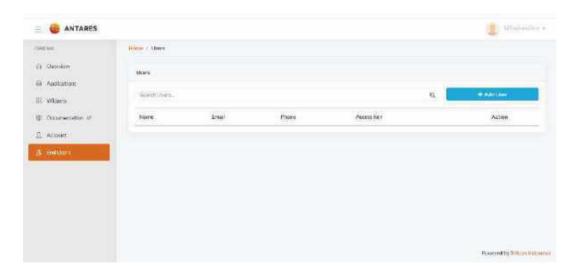
Gambar 3.37 Halaman Documentation Tutorial

Pada halaman *documentation* juga terdapat sebuah tutorial mengenai komponen yang dimana menggunakan antares sebagai media penyimpanan datanya. Kemudian pada halaman ini juga menampilkan sebuah contoh kode yang bisa kita gunakan.



Gambar 3.38 Halaman Account

Halaman *account* ini menampilkan suatu informasi pemilik *account* seperti foto profile, *access key*, *name*, *email*, *phone number*, *address*.



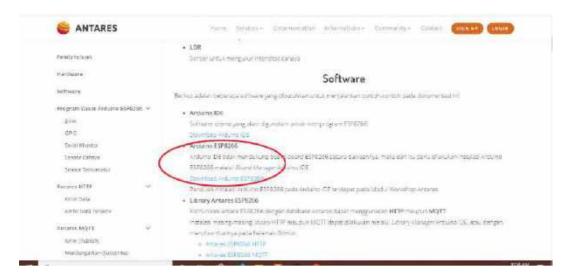
Gambar 3.39 Halaman End User

Halaman *End User* berisi mengenai informasi pembuatan user baru dengan cara memasukan *name*, *email*, *phone*, *access key and action*

2. Memasukan Library antares ke coding

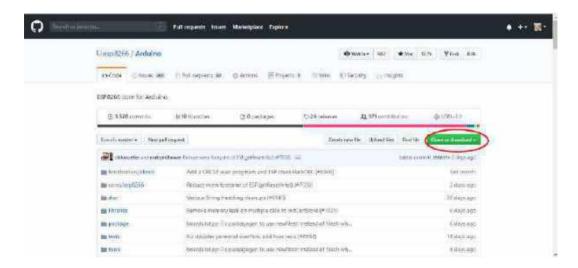
Sebelum memprogram dan membuat logika tentang antares terlebih dahulu harus mendownload *library* yang akan digunakan dengan cara berikut :

Setelah login ke antares pilih menu *documentation* kemudian pilih esp8266 *go to tutorial*.



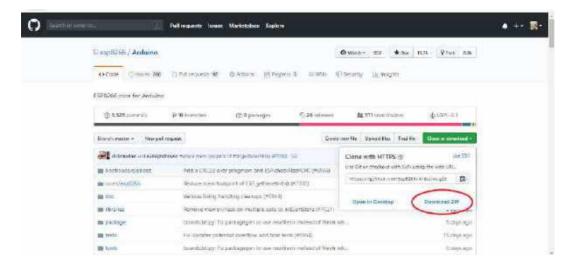
Gambar 3.40 Download Library

Setelah kita memilih *go to tutorial* maka akan muncul tampilan seperti gambar di atas. Lalu pilih atau klik *Download* arduino esp8266.



Gambar 3.41 Download Library di Github

Ketika kita sudah klik *Download* arduino esp8266. Maka Akan muncul tampilan seperti di atas, dimana tampilan tersebut langsung masuk ke repository di github. Setelah masuk ke github kemudian kita klik menu *clone or Download*.

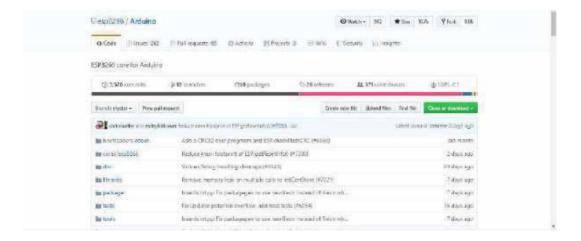


Gambar 3.42 Download Library

Ketika menu *clone or Download* diklik maka akan muncul dua pilihan seperti pada gambar di atas. Untuk Download *library* maka pilih *Download ZIP*.

Jika kita ingin kita *import library* menggunakan board manager, maka langkahlangkahnya seperti berikut :

Setelah login ke antares pilih menu *documentation* kemudian pilih esp8266 *go to tutorial*.



Gambar 3.43 Tampilan di Github

Setelah kita memilih *go to tutorial* maka akan muncul tampilan seperti gambar di atas. Lalu scroll ke bawah hingga menemukan tulisan *contents*.



Gambar 3.44 Tampilan content

Kemudian setelah *scroll* ke bawah hingga menemukan tulisan *content maka* pilih atau klik *using boards manager*.



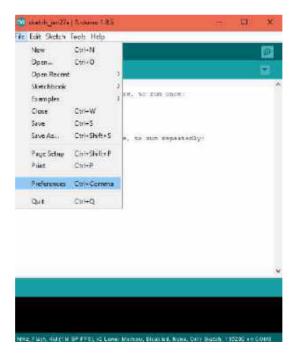
Gambar 3.45 Tampilan Using Boards Manager

Ketika klik *using boards manager* maka tampilan yang akan ditampilkan seperti gambar di atas. Kemudian kita copy alamat link yang tertera.



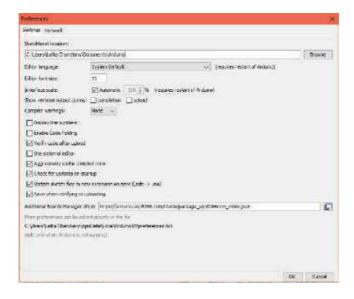
Gambar 3.46 Tampilan Arduino IDE

Setelah mengcopy link dari github,Kita terlebih dahulu membuka software arduino IDE.



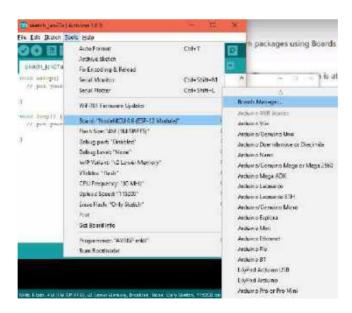
Gambar 3.47 Tampilan Arduino IDE

Ketika arduino IDE telah dibuka. Maka kita pilih File klik preferences seperti pada gambar di atas.



Gambar 3.48 Tampilan Preferences

Setelah memilih preferences maka tampilannya yang muncul seperti pada gambar tersebut. Kemudian paste alamat url di bagian *Additional Boards Manager URLs* yang sebelumnya di copy dari github dan klik ok.



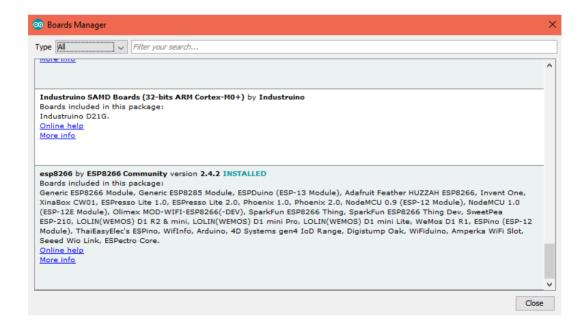
Gambar 3.49 Tampilan Arduino IDE Boards Mnager

Kemudian jika sudah salin alamat url maka selanjutnya pilih Tools lalu Boards kemudian board manager seperti gambar di atas.



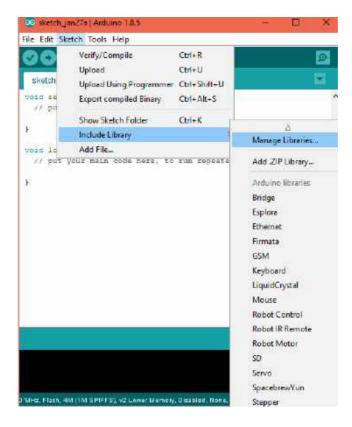
Gambar 3.50 Tampilan Boards Manager

Jika kita telah klik boards manager maka akan muncul tampilan boards manager.



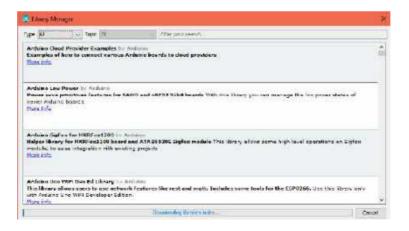
Gambar 3.51 Tampilan Boards Manager Antares

Ketika tampilan boards manager telah muncul kita scroll ke yang paling bawah. Kemudian install esp8266 dan versinya disesuaikan.



Gambar 3.52 Tampilan Arduino IDE Include Library

Setelah menginstal esp8266 tahapan selanjutnya kita pilih sketch lalu Include Library kemudian Manage Library. Seperti pada gambar di atas.



Gambar 3.53 Tampilan Library Manager

Jika kita telah klik manage libraries maka akan muncul tampilan seperti gambar di atas.



Gambar 3.54 Tampilan Library Manager Antares

Setelah muncul tampilan manage libraries kemudian ketikan antares pada bagian *search*. Ketika telah memasukan kata antares maka akan muncul *library* antares paling atas lalu install.



Gambar 3.55 Tampilan Arduino IDE Manage Library

Jika telah menginstall semua library antares yang dibutuhkan, maka hal yang harus dilakukan selanjutnya yaitu menginclude library dengan cara pilih *sketch* lalu *include library* dan pilih Antares ESP8266 HTTP.



Gambar 3.56 Tampilan Arduino IDE Library Antares

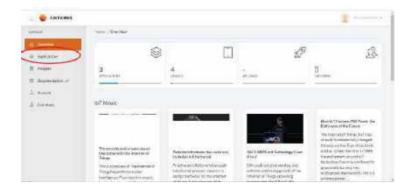
Jika kita sudah mengklik library antares yang telah di Download maka akan tertambahkan *code*nya seperti pada gambar di atas.

3. Membuat Project Di Antares

Sebelum membuat sebuah program yang isinya gabungan dari beberapa komponen untuk membuat *prototype* ini , diperlukan terlebih dahulu membuat sebuah *project* atau *application* di antares yang bertujuan untuk menyimpan data yang dibaca oleh sensor jarak. Berikut langkah-langkahnya:

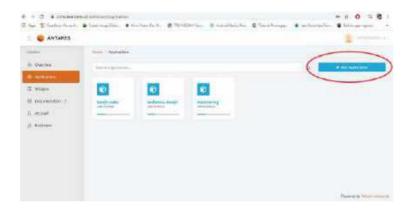
Pertama yang harus dilakukan yaitu masuk ke halaman antares dengan cara mengunjungi website antares yaitu antares.id.

Kemudian login dengan gmail dan passworld yang sudah terdaftar sebelumnya.



Gambar 3.57 Tampilan Awal Antares

Setelah login akan muncul tampilan awal antares yang disediakan dengan beberapa menu. Untuk membuat sebuah *project* baru atau *application* pilih menu *application*.



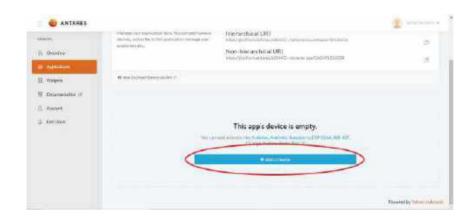
Gambar 3.58 Tampilan Application Antares

Ketika kita mengklik atau pilih menu *application*, maka kita akan dihadapkan dengan tampilan seperti pada gambar di atas. Setelah itu klik *add aplication* untuk membuat sebuah *project* baru.



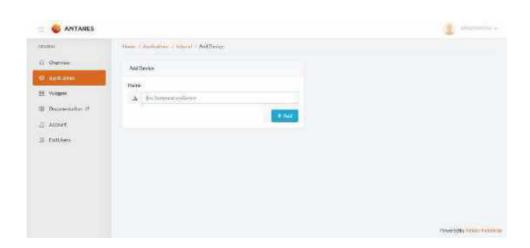
Gambar 3.59 Tampilan Membuat Application Antares

Setelah meng klik *add application*, maka di tampilan selanjutnya harus menggisikan *application name* dan *application ID* dengan nama yang bebas. Disini diberikan sebuah *application name* dan *application IDnya* monitoring. Karena nantinya data ini akan digunakan sebagai monitoring ketinggian air. Kemudian setelah mengisi semua klik *add*.



Gambar 3.60 Tampilan Application Yang Dibuat

Kemudian klik *Add Device* untuk membuat sebuah *device* baru seperti gambar di atas.



Gambar 3.61 Tampilan Membuat Device Antares

Gambar di atas merupakan sebuah tampilan untuk membuat device baru.



Gambar 3.62 Tampilan Membuat Device Antares

Setelah itu masukan nama *devicename* disini menggunakan nama "tinggi sensor" atau dapat disesuaikan dengan kebutuhan kemudian klik *add*.



Gambar 3.63 Tampilan Device Di Application Antares

Sesudah membuat *device* pada *application* maka tampilannya seperti pada gambar di atas. Untuk mengetahui isi apa saja yang ada *device*, maka klik nama *devicenya*.



Gambar 3.64 Tampilan Halaman Device Antares

Pada gambar tersebut merupakan sebuah tampilan *device* tinggi sensor. Dimana pada *device* ini datanya masih kosong karena belum disambungkan deng project.

4. Memprogram Komponen Dengan Mengkoneksikan Pada Antares

Setelah kita membuat sebuah *project* atau *application* di antares, maka sekarang membuat sebuah program yang datanya bisa disimpan diantares. Dimana data tersebut dapat terkoneksi dengan antares. Dengan cara menambahkan code seperti berikut:

(a) Pertama kita harus mengimport library dan menginisiasi pin yang digunakan , dengan cara menambahkan *code* berikut :

```
#include <AntaresESP8266HTTP.h>

const int trigPin = 2; //D4

const int echoPin = 0; //D3

const int ledPin = 4; //D2

const int buzzerPin = 5; //D1
```

(b) Kedua setelah kita menginisiasi pin yang digunakan yaitu mendefinisikan variabel, dengan menambahkan *code* berikut :

```
// defines pins numbers

# define ACCESSKEY "xxx"

# define WIFISSID "xxx"

# define PASSWORD "123 lalita"

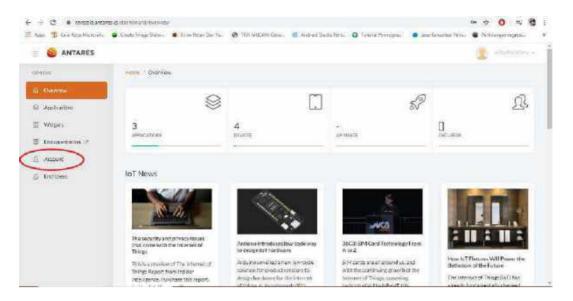
# define projectName "nama project diantares"

# define deviceName "nama device diantares"

AntaresESP8266HTTP antares (ACCESSKEY);

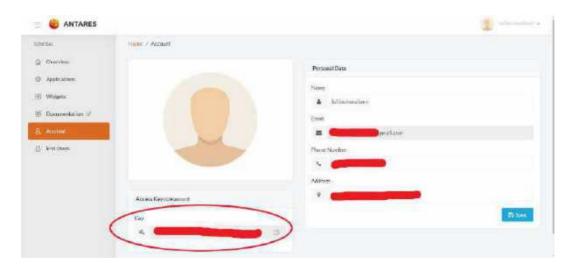
// defines variables
int duration;
int distance;
int safetyDistance;
```

Pada saat pendefinisian variabel harus memasukan *access key* ,WIFI SSID (nama wifi yang akan dikoneksikan), *passworld* (kata sandi dari wifi SSID) wifi ini digunakan untuk koneksi nodemcu. Sedangkan untuk *access key* didapatkan dari antares dengan cara sebagai berikut:



Gambar 3.65 Tampilan Antares

Pertama *login* antares ketika muncul halaman utama pilih *users*.



Gambar 3.66 Tampilan Users Antares

Setelah masuk ke halaman *users copy key* tersebut, kemudian di *paste* di bagian pendefinisian variabel.

(c) Ketika telah mendefinisikan variabel, saatnya masuk bagian void setup, dimana void setup ini dieksekusi pertamakali dan hanya sekali dieksekusi hanya pada saat program dijalankan. Selain itu juga pada bagian void setup ini mengatur *inputan* dan *output* berikut codenya:

```
void setup() {
    Serial.begin(115200);
    antares.setDebug(true);
    antares.wifiConnection(WIFISSID, PASSWORD);
```

```
pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
pinMode(echoPin, INPUT); // Sets the echoPin as an Input
pinMode(buzzerPin, OUTPUT);
pinMode(ledPin, OUTPUT);
```

(d) selanjutnya membuat void loop. Berikut code looping sensor ultrasonic, buzzer dan led.

```
void loop() {
3 // Clears the trigPin
4 digitalWrite (trigPin, LOW);
5 delayMicroseconds(2);
  // Sets the trigPin on HIGH state for 10 micro seconds
8 digitalWrite(trigPin, HIGH);
9 delayMicroseconds(10);
digitalWrite(trigPin, LOW);
12 // Reads the echoPin, returns the sound wave travel time in
      microseconds
 duration = pulseIn (echoPin, HIGH);
13
14
15 // Calculating the distance
distance = duration *0.034/2;
18
20 safetyDistance = distance;
 if (safetyDistance <= 10){
    digitalWrite(buzzerPin, HIGH);
22
    digitalWrite(ledPin, HIGH);
25 else {
    digitalWrite (buzzerPin, LOW);
    digitalWrite (ledPin, LOW);
 }
```

(e) kemudian membuat code untuk ke menghubungkan antaresnya sebagai berikut :

```
antares.add("distance", distance);
antares.add("duration", duration);
antares.send(projectName, deviceName);
```

(f) Terakhir yaitu membuat sebuah code untuk menampilkannya di serial monitor dengan cara membuat code berikut ini :

```
// Prints the distance on the Serial Monitor
Serial.print("Distance: ");
Serial.println(distance);
Serial.print("pulseIn ");
Serial.print(duration);
}
```

Jika kita gabungkan code dari awal yaitu mengimport library sampedengan menampilkan di serial monitor akan seperti berikut ini :

```
#include <AntaresESP8266HTTP.h>
3 const int trigPin = 2;
4 const int echoPin = 0;
                           I/D3
5 const int ledPin = 4; //D2
6 const int buzzerPin = 5; //D1
8 // defines pins numbers
9 #define ACCESSKEY "xxx"
10 #define WIFISSID "xxx"
# define PASSWORD "xxx"
#define projectName "nama project"
14 #define deviceName "nama device"
AntaresESP8266HTTP antares (ACCESSKEY);
 // defines variables
int duration;
19 int distance;
  int safety Distance;
20
  void setup() {
    Serial.begin(115200);
    antares.setDebug(true);
24
    antares.wifiConnection(WIFISSID, PASSWORD);
25
    pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
    pinMode(echoPin, INPUT); // Sets the echoPin as an Input
27
    pinMode(buzzerPin, OUTPUT);
28
    pinMode(ledPin , OUTPUT);
29
30
31
32
  void loop() {
33
34
  // Clears the trigPin
  digitalWrite (trigPin, LOW);
  delayMicroseconds (2);
  // Sets the trigPin on HIGH state for 10 micro seconds
  digitalWrite (trigPin, HIGH);
delay Microseconds (10);
 digitalWrite (trigPin, LOW);
43
  // Reads the echoPin, returns the sound wave travel time in
      microseconds
  duration = pulseIn(echoPin, HIGH);
45
  // Calculating the distance
  distance = duration *0.034/2;
 safetyDistance = distance;
if (safetyDistance <= 10){
    digitalWrite(buzzerPin, HIGH);
52
    digitalWrite(ledPin, HIGH);
```

```
54 }
  else {
    digitalWrite(buzzerPin, LOW);
    digitalWrite(ledPin, LOW);
57
58
59
    antares.add("distance", distance);
61
    antares.add("duration", duration);
62
    antares.send(projectName, deviceName);
63
65 // Prints the distance on the Serial Monitor
66 Serial.print("Distance: ");
67 Serial. println (distance);
68 Serial.print("pulseIn");
69 Serial. print (duration);
```

Setelah semuanya komponen di program dan sudah dikoneksi dengan antares saatnya mengupload file. Tetapi lebih baik sebelum mengupload harus melakukan verifikasi coding . Verifikasi ini bertujuan untuk mengecek apakah ada error atau tidaknya. Cara untuk memverifikasi dengan cara klik centang pada Arduino IDE.



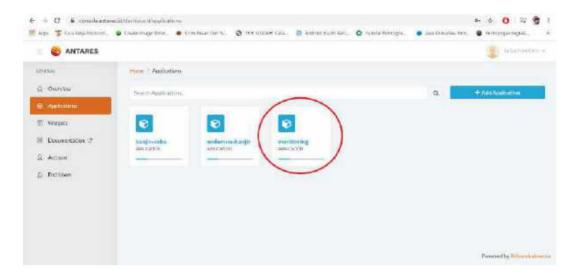
Gambar 3.67 Upload Arduino IDE

Kemudian setelah melakukan verifikasi dan apabila tidak ada error maka langsung untuk mengupload code yang sebelumnya telah dibuat klik gambar panah pada arduino IDE seperti pada gambar di atas. Lalu jika tidak ada error lakukan *upload sketch*

Jika proses *upload* sudah selesai , maka akan adanya tulisan *done upload* di bagian pojok kanan bawah.

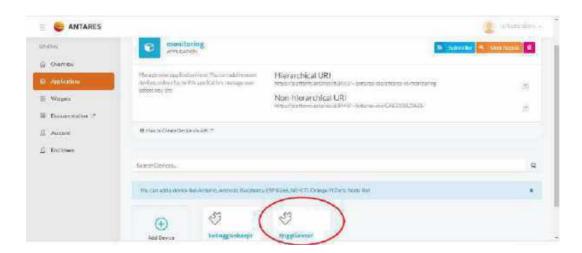
Setelah upload maka semua komponen akan menjalankan perintahnya sesuai code yang telah dibuat sebelumnya. Kemudian untuk melihat data yang dibaca oleh sensor ultrasonik kemudian dikirimkan ke antares sebagai berikut:

Hal pertama yang harus dilakukan yaitu *login* antares, kemudian pilih menu *application*.



Gambar 3.68 Tampilan Application

Maka akan tampil halaman seperti di atas kemudian pilih *project* yang telah kita buat serta dikoneksikan dengan *project prototype* ini yaitu pilih yang bernama monitoring.



Gambar 3.69 Tampilan *Project*

Setelah memilih *project* kemudian memilih *device* yang telah dikoneksikan sebelumnya yaitu nama *devicennya* tinggi sensor.



Gambar 3.70 Tampilan Data Sensor Ultrasonic

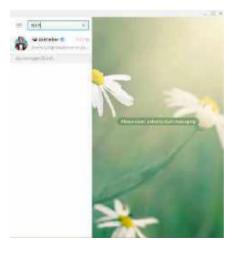
Pada gambar di atas merupakan data ketinggian air yang dibaca oleh sensor ultrasonic yang dikirimkan ke antares secara *realtime*.

3.3.5 Menambahkan Code Notifikasi Telegram Prototipe PKA

Pembahasan mengenai telegram telah dibahas pada bab *software* yang digunakan . Pada prototipe PKA ini adanya sebuah notifikasi melalui bot telegram. Notifikasi ini bertujuan untuk sebuah peringatan bila akan terjadinya banjir. Dalam notifikasi tersebut terdapat sebuah warning mengenai ketinggian air dan berapa lama waktu yang dibutuhkan untuk mencapai volume tertentu.

Untuk membuat sebuah notifikasi ini hanya menambahkan beberapa *code* pemrograman pada *code* pemrograman sebelumnya. Sebelum memprogramnya terlebih dulu harus membuat sebuah *bot* ditelegram. Dengan cara berikut :

1. Buka aplikasi telegram dan ketikan BotFather.



Gambar 3.71 Tampilan Telegram

2. Lalu buka BotFather, ketik START.



Gambar 3.72 Tampilan Bot Telegram

3. Ketik /newbot , selanjutnya akan diminta memberikan nama bot dan username bot. Jika sudah sesuai akan muncul Token, seperti yang dilingkari dibawah ini. Simpan Token tersebut.

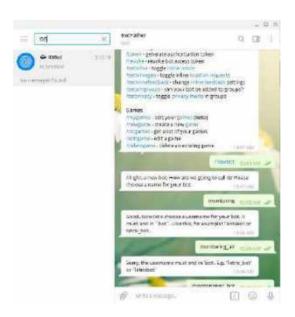


Gambar 3.73 Tampilan Bot Telegram



Gambar 3.74 Tampilan Bot Telegram

4. selain BotFather, harus mengetahui idnya yaitu cari ID Bot.



Gambar 3.75 Tampilan ID Bot Telegram

5. Klik /start , lalu ketik /getid. Kemudian akan muncul id telegram seperti yang dibawah ini:



Gambar 3.76 Tampilan ID Bot Telegram



Gambar 3.77 Tampilan ID Bot Telegram

Jika sudah membuat bot dan juga sudah mengetahui Token dan id telegram. Selanjutnya yang tidak kalah penting perlu masuk ke Bot yang sudah buat dan klik /start. Untuk testing.



Gambar 3.78 Tampilan Bot Monitoring Di Telegram

Jika semua langkah sudah dilakukan, selanjutnya menDownload sebuah library telegram. Pada tahapan ini perlu menDownload library ctbot pada https://www.dropbox.com/s/zyx7i3o/CTBot.zip dan https://www.dropbox.com/s/g9ycw3o3uv7cg z7/ArduinoJson.zip untuk library arduino json. Jika tidak ingin menDownload melalui situs tersebut dapat menDownload di situs lain atau github.



Gambar 3.79 Tampilan Download CTbot



Gambar 3.80 Tampilan Download Arduino Json

Tahapan selanjutnya yaitu menambahkan *code* pemrogramanya dengan cara berikut

1. Pertama *import* library yang telah diDownload. Sehingga akan muncul secara otomatis code berikut :

```
#include <CTBot.h>
```

2. Kemudian menginisiasi sebagai berikut kodenya.

```
CTBot myBot;

String token = "xxx";

const int id = xxx;
```

3. Selanjutnya mendefinisikan variabel.

```
int duration;
int distance;
int safetyDistance;
boolean kirimTBot = true;
String jarak = "";

String jsonString = "{}";
```

4. Membuat sebuah inputan maupun outputan di bagian void setup.

```
void setup() {
    Serial.begin(115200);
    myBot.wifiConnect(WIFISSID, PASSWORD);
5
    myBot.setTelegramToken(token);
6
    delay (1000);
7
8
    antares.setDebug(true);
9
    antares.wifiConnection(WIFISSID, PASSWORD);
10
11
    pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
    pinMode(echoPin, INPUT); // Sets the echoPin as an Input
13
    pinMode(buzzerPin, OUTPUT);
14
    pinMode(ledPin , OUTPUT);
15
17
```

5. Membuat Json nya agar dapat mengirimkan notifikasi melalui telegram.

```
isiJson("distance", distance);
isiJson("duration", duration);

// Serial.println(jsonString);
antares.isi(jsonString);
antares.send(projectName, deviceName);
```

6. Membuat void json nya dengan code sebagai berikut :

```
void isiJson(String key, int value) {
   DynamicJsonBuffer jsonBuffer;
   JsonObject& object = jsonBuffer.parseObject(jsonString);
   object[key] = value;
   String newInsert;
   object.printTo(newInsert);
   jsonString = newInsert;
}
```

7. Jika *code*nya digabungkan dari awal sampai menggunakan bot telegram yaitu sebagai berikut :

```
ng#include <CTBot.h>
2 #include <ArduinoJson.h>
3 #include <AntaresESP8266HTTP.h>
4 #include "data.h"
  #define WIFISSID "xxxxxx"
7 #define PASSWORD "xxxxx"
9 #define ACCESSKEY "xxxxxxxxx"
10 #define projectName "nodemcu-banjir"
#define deviceName "ketinggia-air"
  AntaresESP8266HTTP antares (ACCESSKEY);
14
15
16 CTBot myBot;
String token = "xxxxxxxxxxxxxxxxx;";
  const int id = xxxxxxxxxxxxxx;
 // defines variables
int duration;
22 int distance;
23 int safetyDistance;
24 boolean kirimTBot = true;
  String jarak = "";
26
  String | isonString = "{}";
27
28
29 // defines pins numbers
30 const int trigPin = 2;
                          //D4
const int echoPin = 0; //D3
32 const int ledPin = 4; //D2
  const int buzzerPin = 5; //D1
33
35 float regresi_a;
36 float regresi_b;
37
  void setup() {
39
40
    Serial . begin (115200);
41
    myBot.wifiConnect(WIFISSID, PASSWORD);
42
    myBot.setTelegramToken(token);
43
    delay (1000);
```

```
antares.setDebug(true);
    antares.wifiConnection(WIFISSID, PASSWORD);
47
48
    pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
49
    pinMode(echoPin, INPUT); // Sets the echoPin as an Input
50
    pinMode(buzzerPin, OUTPUT);
51
    pinMode(ledPin , OUTPUT);
52
53
    hitungParameterRegresi (regresi_a, regresi_b);
54
55
  }
56
57
  // the loop function runs over and over again forever
58
  void loop() {
      // Clears the trigPin
       digital Write (trigPin, LOW);
61
      delayMicroseconds (2);
62
63
       // Sets the trigPin on HIGH state for 10 micro seconds
       digital Write (trigPin, HIGH);
      delayMicroseconds (10);
66
       digital Write (trigPin, LOW);
67
68
       // Reads the echoPin, returns the sound wave travel time in
       microseconds
      duration = pulseIn (echoPin, HIGH);
70
      // Calculating the distance
      distance = duration * 0.034 / 2;
74
      // distance = 4:
75
76
      //duration = 27;
       float perkiraanWaktu = 1.0 * regresi_b * distance + regresi_a
78
79
      isiJson("distance", distance);
81
      isiJson ("duration", perkiraan Waktu);
82
83
      // Serial . println (jsonString);
       antares.isi(jsonString);
85
       antares.isi(jsonString);
       antares.send(projectName, deviceName);
87
88
      safetyDistance = distance;
90
       if (safetyDistance <= 5) {
91
         jarak = "Ketinggian Air Saat Ini Adalah : ";
92
        jarak += int(distance);
         jarak += "*meter n";
94
         jarak += " *Air Sudah Mencapai Ketinggian maksimal Awas
95
       Banjir!! \n";
         jarak += " Perkiraan waktu";
96
        jarak += String(perkiraanWaktu);
```

```
jarak += " menit ";
98
         if (kirimTBot) {
           myBot.sendMessage(id, jarak, "");
            kirimTBot = false;
101
102
103
         digitalWrite(buzzerPin, HIGH);
         digitalWrite(ledPin, HIGH);
         Serial.println("Banjir");
106
       }
107
108
       else {
         digitalWrite (buzzerPin, LOW);
109
         digitalWrite(ledPin, LOW);
         kirimTBot = true;
         Serial.println("Aman");
113
114
116
   void isiJson (String key, int value) {
118
     DynamicJsonBuffer jsonBuffer;
     JsonObject& object = jsonBuffer.parseObject(jsonString);
120
     object[key] = value;
     String newInsert;
     object.printTo(newInsert);
     jsonString = newInsert;
124
125
126
   void hitungParameterRegresi(float &a, float &b)
128
     float X = 0;
129
     float Y = 0;
130
     float X2 = 0;
     float Y2 = 0;
132
     float XY = 0;
134
135
     int jumlahData = sizeof(data) / sizeof(Data);
136
     for (int i = 1; i < jumlahData; i++)
138
       float x = abs(data[i].distance - data[i - 1].distance) / data
139
       [i]. selisihWaktu;
       float y = data[i].selisihWaktu;
140
141
       X +=
142
       Y +=
              y;
       X2 += (x * x);
144
       Y2 += (y * y);
145
       XY += x * y;
146
147
148
     a = ((Y * X2) - (X * XY)) / ((jumlahData * X2) - (X * X));
149
     b = ((jumlahData * XY) - (X * Y)) / ((jumlahData * X2) - (X * X)
150
       ));
151
```

```
Serial . print ("X=");
152
     Serial. println(X);
153
     Serial.print("Y=");
154
     Serial.println(Y);
155
     Serial. print ("X2=");
156
     Serial.println(X2);
157
     Serial . print ("Y2=");
158
     Serial.println(Y2);
159
     Serial.print("XY=");
160
     Serial. println(XY);
161
     Serial.print("a=");
     Serial.println(a);
163
     Serial.print("b=");
164
     Serial.println(b);
165
166
168
```

Langkah berikutnya mengupload program tersbut. Apabila tidak error akan megirimkan sebuah notifikasi peringatan banjir. notifikasi tersebut akan dikirimkan apabila sensor ultrasonik membaca jarak ketinggian air kurang dari 5 meter.



Gambar 3.81 Tampilan Notifikasi DI Telegram

Gambar di atas merupakan isi notifikasi dari bot telegram. Pada notifikasi tersebut berisi tentang ketinggian air dan prediksi berapa lama waktu yang dibutuhkan air untuk mencapai volume tertentu.

PEMBUATAN APLIKASI MONITORING AIR

Android adalah sistem operasi yang dirancang oleh Google dengan basis kernel Linux untuk mendukung kinerja perangkat elektronik layar sentuh, seperti tablet atau smartphone. Jadi, android digunakan dengan sentuhan, gesekan ataupun ketukan pada layar gadget anda. [9]

Android salah satu sistem operasi atau *operating system* berbasis *mobile* yang sangat banyak di gunakan sekarang ini. Utamanya pada telepon pintar (*smartphone*) ataupun tablet.[9]

Android bersifat *open source* atau bebas digunakan, dimodifikasi, diperbaiki dan didistribusikan oleh para pembuat ataupun pengembang perangkat lunak. Dengan sifat *open source* perusahaan teknologi bebas menggunakan OS ini diperangkatnya tanpa lisensi (gratis).

Android Studio adalah Lingkungan Pengembangan Terpadu (*Integrated Development Environment*/IDE) resmi untuk pengembangan aplikasi Android, yang didasarkan pada IntelliJ IDEA. Selain sebagai editor kode dan fitur developer IntelliJ yang andal, Android Studio menawarkan banyak fitur yang meningkatkan produktivitas Anda dalam membuat aplikasi Android, seperti: [10]

1. Sistem build berbasis Gradle yang fleksibel

- 2. Emulator yang cepat dan kaya fitur
- 3. Lingkungan terpadu tempat mengembangkan aplikasi untuk semua perangkat Android.
- 4. Terapkan Perubahan untuk melakukan push pada perubahan kode dan resource ke aplikasi yang sedang berjalan tanpa memulai ulang aplikasi.
- 5. Template kode dan integrasi GitHub untuk membuat fitur aplikasi umum dan mengimpor kode sampel.
- 6. Framework dan fitur pengujian yang lengkap.
- 7. Fitur lint untuk merekam performa, kegunaan, kompatibilitas versi, dan masalah lainnya.
- 8. Dukungan C++ dan NDK.
- 9. Dukungan bawaan untuk Google Cloud Platform, yang memudahkan integrasi Google Cloud Messaging dan App Engine.

Pada Android studio ini memiliki kekurangan dan kelebihan serta struktur project yang telah dibahas atau dijelaskan secara detail pada buku versi 1. Pada pembuatan aplikasi monitoring ini menggunakan bahasa Java .

Java adalah sebuah bahasa pemrograman yang sangat populer yang dikembangkan oleh Sun Microsystems (saat ini dimiliki oleh Oracle). Dikembangkan lama setelah C dan C++, Java menggabungkan banyak fitur-fitur canggih dari bahasa-bahasa canggih tersebut, sambil mengatasi beberapa kelemahan mereka. Walaupun demikian, tingkat kecanggihan bahasa pemrograman bergantung pada *library* mereka. *Library* ini ada untuk membantu para developer untuk membuat aplikasi.[11]

Android sangat bergantung kepada sifat-sifat dasar dari Java tersebut. Android SDK mengandung banyak *library* Java standard (*library* struktur data, *library* matematika, *library* grafik, *library networking* dan apapun yang dapat Anda inginkan) dan juga *library special* Android yang dapat membantu mengembangkan aplikasi Android.

4.1 Pembuatan Aplikasi Monitoring PKA

Untuk membuat sebuah aplikasi monitoring ini menggunakan sebuah software yaitu android studio dengan menggunakan bahasa java, yang telah dibahas pada bab sebelumnya dan buku versi 1.

4.1.1 Pembuatan Splash Screen

Splash screen merupakan sebuah tampilan awal program yang muncul sementara sebelum masuk ke menu utama. Berikut langkah-langkahnya:

1. Buka aplikasi android studio kemudian pilih start a new android studio project



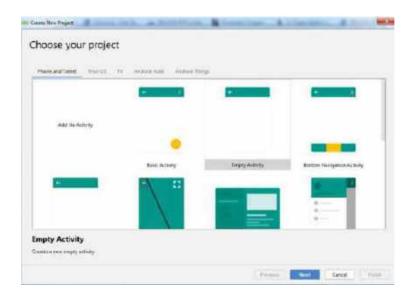
Gambar 4.1 Tampilan Awal Android Studio

2. Pilih *Activity*

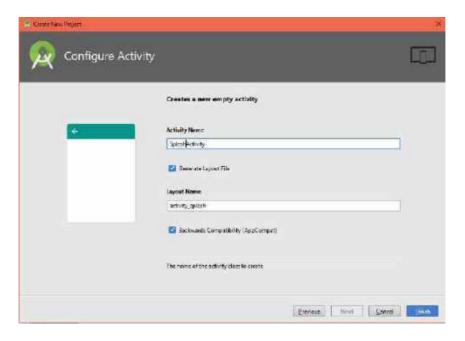
Activity (aktivitas) merupakan sebuah komponen aplikasi yang menyediakan layar yang digunakan pengguna untuk berinteraksi guna melakukan sesuatu, misalnya memilih nomor ponsel, mengambil foto, mengirim email, atau menampilkan peta.

Tiap *Activity* (aktivitas) diberi sebuah jendela untuk menggambar antarmuka penggunanya. Jendela ini biasanya mengisi layar.

Pilih Empty Activity kemudian klik Next.



Gambar 4.2 Tampilan Memilih Activity

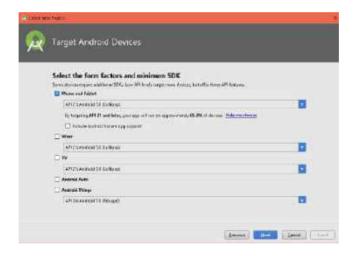


Gambar 4.3 Membuat Activity splash Activity

- 3. Konfigurasi *Project* Setelah memilih *activity*, langkah selanjutnya yaitu kita harus memberikan konfigurasi pada *project* android yang kita buat. Adapun yang harus di konfigurasi yaitu :
 - (a) Name: isikan nama project
 - (b) Package name: isikan nama perusahaan atau pembuat aplikasi.
 - (c) Save location: tentukan file project aplikasi android ini akan disimpan pada direktori apa.
 - (d) Language: pilih bahasa pemrograman yang akan digunakan, Java atau Kotlin. Pada tutorial ini kita menggunakan Java.
 - (e) Minimum API level: tentukan API berapa yang akan digunakan. Pada bulan April 2017 google merilis daftar versi OS yang banyak digunakan tahun 2017 dan hasilnya Lollipop (OS versi 5.0 dengan SDK 21/22) ada di urutan nomor satu. Jadi untuk minimum SDKnya kita pilih API 15 yaitu untuk versi OS Ice Cream Sandwich ke atas. Tetapi dapat disesuaikan sesuai kebutuhan.

7200	
Application wave	
Nestrong	
Company day with	
Mitarian and Applicate	
Project Rocker	
Characters Charactery Andread Scalad Ayusto Mantarings	J.
Parkage range	
nikannyhumminettiny enifethy	1,100
□ bickele C++ aggort	
mikati Keda support	

Gambar 4.4 Konfigurasi Project



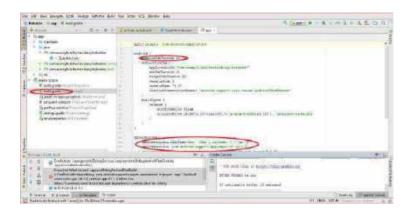
Gambar 4.5 Konfigurasi Project

4. Proses Gradle



Gambar 4.6 Proses Gradle

Jika pertama kali tampilan di android studio error maka yang bermasalah yaitu mengenai gradle, untuk mengatasinya harus menyesuaikan versinya. Dengan cara mengubah versi gradlenya sesuai dengan apa yang diminta oleh android studio. Untuk menggantinya masuk ke GradleScript-build gradle dan ubah versinya dibagian yang telah dilingkari seperti gambar berikut ini:

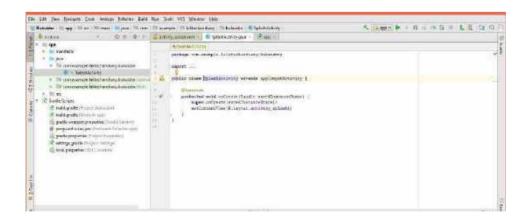


Gambar 4.7 Build Gradle

Tunggu sampai proses gradle selesai. Biasanya memakan waktu beberapa menit. Lama atau sebentarnya tergantung spesifikasi ram dan prosesor yang digunakan pada komputer atau laptop.

5. Workspace Android Studio

Jika proses Gradle telah selesai, maka akan terbuka workspace project Android Studio seperti gambar dibawah ini (pertama kali terbuka adalah tab SplashActivity.java. Disinilah nantinya kita memberikan kode-kode java agar aplikasi bisa melakukan sebuah perintah).

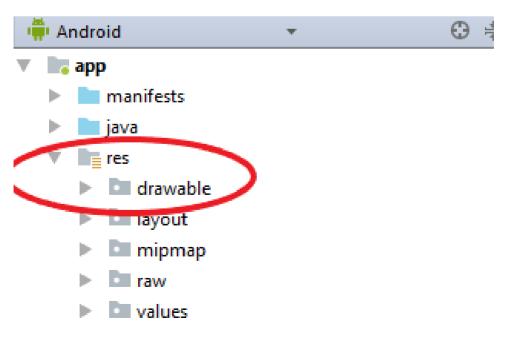


Gambar 4.8 Workspace

Masukan kode berikut ini untuk membuat splash screen di splashActivity.java

```
package com. example. lalitachandiany. monitoring;
  import android.content.Intent;
  import android.support.v7.app.AppCompatActivity;
  import android.os.Bundle;
  public class SplashActivity extends AppCompatActivity {
       @Override
9
      protected void onCreate(Bundle savedInstanceState) {
10
           super.onCreate(savedInstanceState);
           setContentView (R. layout. activity_splash);
13
           Thread thread = new Thread(){
14
               public void run () {
15
                   try {
                        sleep (4000);
                   } catch (InterruptedException e) {
18
                        e.printStackTrace();
19
                   }finally {
                        startActivity (new Intent (SplashActivity.this,
        HomeActivity.class));
                        finish();
23
               }
24
           };
25
           thread.start();
26
      }
27
28
```

Kemudian untuk tampilan Splashscreen harus memasukan gambar yang akan digunakan kedaam folder drawable.



Gambar 4.9 Drawable

Kemudian untuk membuat tampilanya diprogram dalam activityplash.xml dengan kode berikut ini :

```
<?xml version="1.0" encoding="utf-8"?>
 <android.widget.LinearLayout xmlns:android="http://schemas.</pre>
      android.com/apk/res/android"
      xmlns:app="http://schemas.android.com/apk/res-auto"
      xmlns:tools="http://schemas.android.com/tools"
4
      android: layout_width="match_parent"
5
      android: layout_height="match_parent"
6
      android: orientation = "vertical"
      android: background="@drawable/spalsh"
8
      tools: context="com. example. lalitachandiany. monitoring.
0
      Splash Activity">
12 </android.widget.LinearLayout>
```

4.1.2 Pembuatan Halaman Menu

Menu merupakan sususan perintah dari perangkat lunak (program) yang jika dieksekusi akan menjalankan suatu perintah tertentu dari aplikasi. Menu digunakan sebagai alternatif dari antarmuka baris perintah. [12]

Halaman menu aplikasi monitoring PKA terdapat slider dan 4 menu utama yaitu *about*, ketinggian air, keterangan air dan *don't panic*. Untuk membuatnya sebagai berikut :

1. Membuat Slider

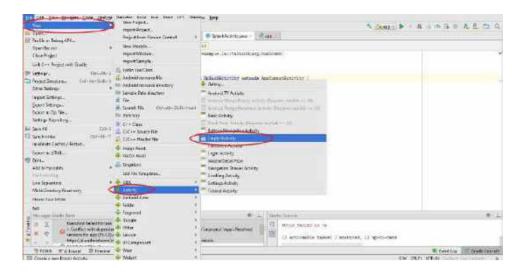
Slider menampilkan kumpulan gambar yang berjalan jika mengklik suatu button atau secara otomatis menerapkan transisi. Dengan kata lain, slider adalah slideshow gambar. Untuk membuat slider pertama yang harus dilakukan yaitu memasukan gambar apa saja yang akan ditampilkan dengan cara memasukanya kedalam folder drawable. [12]

Apabila memasukan gambar kedalam gradle maka import library untuk slider dengan cara memasukan perintahnya di gradle.



Gambar 4.10 Library Slider

Setelah memasukan atau *import library* maka harus membuat sebuah *activity* baru untuk home. *Activity home* ini digunakan untuk 4 menu dan slider yang telah dijelaskan sebelumnya untuk membuat *activity* klik file-new-activity-empaty activity.



Gambar 4.11 Activity Baru

Setelah membuat *activity* baru maka masukan kode berikut ini kedalam file home.java

```
public class HomeActivity extends AppCompatActivity {
      @Override
3
      protected void onCreate(Bundle savedInstanceState) {
4
           super.onCreate(savedInstanceState);
           setContentView (R. layout . activity_home);
           //inisialisasi
           SliderLayout sliderLayout = (SliderLayout) findViewById(R
       .id.slider);
           // Load image dari URL
9
           // Load Image Dari res/drawable
10
          HashMap<String, Integer > file_maps = new HashMap<String,
       Integer >();
           file_maps.put("Sudahkah Anda Siap", R. drawable.siap);
           file_maps.put("Bersihkan Sampah", R. drawable.sungai);
           file_maps.put("Banjir", R. drawable.banjirempat);
14
           file_maps.put("Tampilan Prototipe", R. drawable.full);
           for(String name : file_maps.keySet()){
16
               TextSliderView textSliderView = new TextSliderView (
       this);
               // initialize a SliderLayout
18
               textSliderView
19
                       . description (name)
                       .image(file_maps.get(name))
                        . setScaleType (BaseSliderView . ScaleType . Fit );
               //add your extra information
23
               textSliderView.bundle(new Bundle());
               textSliderView.getBundle()
25
                        .putString("extra", name);
26
               sliderLayout.addSlider(textSliderView);
           }
28
           sliderLayout.setPresetTransformer(SliderLayout.
       Transformer. Accordion);
```

```
sliderLayout.setPresetIndicator(SliderLayout.
PresetIndicators.Center_Bottom);
sliderLayout.setCustomAnimation(new DescriptionAnimation
());
sliderLayout.setDuration(4000);
```

Kemudian untuk tampilanya masukan kode berikut ini kedalam file homeactivity.xml.

```
<?xml version="1.0" encoding="utf-8"?>
 < Relative Layout
      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_width="match_parent"
      android: layout_height="match_parent"
      android:id="@+id/activity_main"
      tools: context="com. example. lalitachandiany. sliderdua.
      MainActivity">
10
          <com. daimajia. slider. library. SliderLayout
               android: id="@+id/slider"
               android: layout_width="match_parent"
               android: layout_height="200dp" />
  </RelativeLayout>
```

Maka hasil nya seperti berikut ini:



Gambar 4.12 Tampilan Slider

2. Membuat Menu Di Home

Aplikasi monitoring PKA terdapat 4 menu yaitu menu *about*, ketinggian air, keterangan air dan *don't panic*. Untuk membuat menu ini kita tidak perlu membuat *activity* baru karena menu ini akan ditampilkan secara bersamaan dengan slider, maka memprogramnya di *activity home*.

Pertama yang dilakukan yaitu mengimport *library cardview* karena untuk membuat menu ini dibutuhkan sebuah *library cardview*. Caranya sama dengan langkah sebelumnya dan masukan perintah seperti dibawah ini :

```
dependencies {
   implementation fileTree(dir: 'libs', include: ['*.jar'])
   implementation 'com android support:appcompat-v7:26.1.0'
   implementation 'com.android.support:cardview-v7:26.1.0'
   implementation 'com.aquaroup pisassospicassos2.3.2'
```

Gambar 4.13 Library Cardview

Jika telah memasukan *library cardview* maka masukan gambar yang akan dijadikan sebagai icon menu ke drawable dengan cara yang sama seperti sebelumnya. Setelah itu masukan kode berikut kedalam file Home.java.

```
/inisialisasi
      ImageView awas = findViewById(R.id.awas);
          ImageView air = findViewById(R.id.air);
          ImageView banjir = findViewById(R.id.banjir);
          ImageView panic = findViewById(R.id.panic);
6
          awas.setOnClickListener(new View.OnClickListener() {
               @Override
               public void onClick(View view) {
                   Intent awas = new Intent (HomeActivity.this,
      Monitoring Activity . class);
                   startActivity (awas);
              }
          });
14
           banjir.setOnClickListener(new View.OnClickListener() {
               @Override
               public void onClick(View view) {
                   Intent banjir = new Intent (HomeActivity.this,
18
      AboutActivity.class);
                   startActivity (banjir);
          });
           air.setOnClickListener(new View.OnClickListener() {
               @Override
24
               public void onClick(View view) {
25
                   Intent air = new Intent (HomeActivity.this,
26
      AirActivity.class);
                   startActivity(air);
27
28
```

Jika digabungkan dengan *code* slider sebelumnya maka seperti berikut:

```
package com. example. lalitachandiany. monitoring;
3 import android.content.Intent;
4 import android.support.v7.app.AppCompatActivity;
5 import android.os.Bundle;
6 import android.view.View;
  import android.widget.ImageView;
  import com. daimajia. slider. library. Animations.
      Description Animation;
  import com. daimajia. slider. library. SliderLayout;
  import com. daimajia. slider. library. SliderTypes. BaseSliderView;
  import com. daimajia. slider. library. SliderTypes. TextSliderView;
  import java.util.HashMap;
15
  public class HomeActivity extends AppCompatActivity {
16
      @Override
18
      protected void onCreate(Bundle savedInstanceState) {
           super.onCreate(savedInstanceState);
20
           setContentView (R. layout . activity_home);
           //inisialisasi
           SliderLayout sliderLayout = (SliderLayout) findViewById(R
      .id.slider);
           // Load image dari URL
24
           // Load Image Dari res/drawable
25
          HashMap<String, Integer> file_maps = new HashMap<String,
      Integer >();
           file_maps.put("Sudahkah Anda Siap", R. drawable.siap);
           file_maps.put("Bersihkan Sampah", R. drawable.sungai);
28
           file_maps.put("Banjir", R. drawable.banjirempat);
           file_maps.put("Tampilan Prototipe", R. drawable.full);
           for (String name : file_maps.keySet()){
               TextSliderView textSliderView = new TextSliderView (
      this);
               // initialize a SliderLayout
               textSliderView
34
                        . description (name)
36
                       .image(file_maps.get(name))
                        . setScaleType (BaseSliderView . ScaleType . Fit );
               //add your extra information
38
```

```
textSliderView.bundle(new Bundle());
30
               textSliderView.getBundle()
40
                        .putString("extra",name);
               sliderLayout.addSlider(textSliderView);
42
           }
           sliderLayout.setPresetTransformer(SliderLayout.
       Transformer. Accordion);
           sliderLayout.setPresetIndicator(SliderLayout.
       PresetIndicators. Center_Bottom);
           sliderLayout.setCustomAnimation(new DescriptionAnimation
46
       ());
           sliderLayout.setDuration(4000);
47
48
49
  //inisialisasi
50
      ImageView awas = findViewById(R.id.awas);
           ImageView air = findViewById(R.id.air);
52
           ImageView banjir = findViewById(R.id.banjir);
           ImageView panic = findViewById(R.id.panic);
           awas.setOnClickListener(new View.OnClickListener() {
               @Override
               public void onClick(View view) {
58
                    Intent awas = new Intent (HomeActivity.this,
       Monitoring Activity . class);
                    startActivity (awas);
60
               }
61
           });
62
           banjir.setOnClickListener(new View.OnClickListener() {
               @Override
               public void onClick(View view) {
66
                    Intent banjir = new Intent (HomeActivity.this,
67
       AboutActivity.class);
                    startActivity (banjir);
               }
           });
70
           air.setOnClickListener(new View.OnClickListener() {
               @Override
74
               public void onClick(View view) {
                    Intent air = new Intent (HomeActivity.this,
       AirActivity.class);
                    startActivity(air);
76
               }
           });
78
           panic.setOnClickListener(new View.OnClickListener() {
80
               @Override
81
               public void onClick(View view) {
82
                    Intent panic = new Intent (HomeActivity.this,
83
       Panic Activity . class);
                    startActivity (panic);
84
               }
85
           });
86
```

```
88 }
```

Setelah itu untuk tampilanya tambahkan *code* berikut ini kedalam file activity-home.xml.

```
<LinearLayout
           android: layout_width="match_parent"
           android: layout_height="wrap_content"
           android: clipToPadding="false"
           android: gravity="center"
           android: orientation="horizontal">
          <android.support.v7.widget.CardView
               android: id="@+id/bankcardId"
               android: layout_width="140dp"
               android: layout_height="160dp"
               android: layout_margin="10dp"
               android: clickable="true"
               android: foreground="?android: attr/
      selectableItemBackground">
               <LinearLayout
16
                   android: layout_width="match_parent"
                   android: layout_height="match_parent"
                   android: background="@color/colorPrimaryDark"
                   android: gravity="center"
20
                   android: orientation = "vertical">
                   <ImageView
                        android: id="@+id/banjir"
                        android: layout_width="80dp"
25
                        android: layout_height="80dp"
26
                        android: padding="10dp"
                        android: src="@drawable/banjirr"/>
28
                   <TextView
30
                        android: layout_width="wrap_content"
31
                        android: layout_height="wrap_content"
                        android: layout_marginTop="10dp"
                        android: background="@android:color/
34
      holo_blue_bright"
                        android: text="About"
                        android: textColor="? attr/colorAccent"
                        android: textStyle="bold" />
38
                   <View
                        android: layout_width="match_parent"
                        android: layout_height="1dp"
                        android: layout_margin="10dp"
42
                        android: background="@color/lightgray"/>
43
                   <TextView
                        android: layout_width="wrap_content"
                        android: layout_height="wrap_content"
                        android: gravity="center"
                        android: padding="5dp"
                        android: text="Apa itu Banjir?"
50
```

```
android: textColor="@android: color/darker_gray
                         android: textStyle="bold"/>
52
                </LinearLayout>
54
           </android.support.v7.widget.CardView>
           <android . support . v7 . widget . CardView
                android: layout_width="140dp"
58
                android: layout_height="160dp"
                android: layout_margin="10dp"
                android: clickable="true"
                android: foreground="?android: attr/
62
       selectableItemBackground">
63
               <LinearLayout
                    android: layout_width="match_parent"
                    android: layout_height="match_parent"
                    android: background="@color/colorPrimaryDark"
67
                    android: gravity="center"
                    android: orientation = "vertical">
70
                    <ImageView
                         android: id="@+id/awas"
                         android: layout_width="90dp"
                         android: layout_height="80dp"
74
                         android: padding="10dp"
75
                         android: src="@drawable/awas" />
76
                    <TextView
78
                         android: layout_width="wrap_content"
79
                         android: layout_height="wrap_content"
80
                         android: layout_marginTop="10dp"
81
                         android: background="@android:color/
       holo_blue_bright"
                         android: text="Ketinggian AIR"
83
                         android: textColor="@color/colorAccent"
84
                         android: textStyle="bold" />
                    <View
87
                         android: layout_width="match_parent"
88
                         android: layout_height="1dp"
                         android: layout_margin="10dp"
                         android: background="@color/lightgray" />
0.1
92
                    <TextView
93
                         android: layout_width="wrap_content"
                         android: layout_height="wrap_content"
05
                         android: gravity="center"
96
                         android: padding="5dp"
97
                         android: text="Monitoring"
                         android: textColor="@android: color/darker_gray
                         android: textStyle="bold"/>
100
101
                </LinearLayout>
```

```
</android . support . v7 . widget . CardView>
103
       </LinearLayout>
106
       <LinearLayout
107
            android: layout_width="match_parent"
108
            android: layout_height="wrap_content"
            android: clipToPadding="false"
            android: gravity="center"
            android: orientation="horizontal">
112
           <android . support . v7 . widget . CardView
114
                android: layout_width="140dp"
                android: layout_height="160dp"
116
                android: layout_margin="10dp"
                android: clickable="true"
                android: foreground="?android: attr/
110
       selectableItemBackground">
                <LinearLayout
                     android: layout_width="match_parent"
                     android: layout_height="match_parent"
                     android: background="@color/colorPrimaryDark"
124
                     android: gravity="center"
                    android: orientation = "vertical">
                    <ImageView
128
                         android: id="@+id/air"
129
                         android: layout_width="80dp"
                         android: layout_height="80dp"
                         android: padding="10dp"
                         android: src="@drawable/air"/>
133
134
                    <TextView
                         android: layout_width="wrap_content"
136
                         android: layout_height="wrap_content"
                         android: layout_marginTop="10dp"
138
139
                         android: background="@android:color/
       holo_blue_bright"
                         android: text="Keterangan Air"
140
141
                         android: textColor="@color/colorAccent"
                         android: textStyle="bold" />
                    <View
1/4/4
                         android: layout_width="match_parent"
                         android: layout_height="1dp"
                         android: layout_margin="10dp"
                         android: background="@color/lightgray" />
148
149
                    <TextView
                         android: layout_width="wrap_content"
                         android: layout_height="wrap_content"
                         android: gravity="center"
                         android: padding="5dp"
154
                         android: text="Kamu Harus Tau"
155
```

```
android: textColor="@android: color/darker_gray
156
                         android: textStyle="bold"/>
158
                </LinearLayout>
159
            </android.support.v7.widget.CardView>
160
            <android . support . v7 . widget . CardView
                android: layout_width="140dp"
163
                android: layout_height="160dp"
164
                android: layout_margin="10dp"
                android: clickable="true"
166
                android: foreground="? android: attr/
167
        selectableItemBackground">
168
                <LinearLayout
                     android: layout_width="match_parent"
170
                     android: layout_height="match_parent"
                     android: background="@color/colorPrimaryDark"
                     android: gravity="center"
                     android: orientation = "vertical">
174
                    <ImageView
176
                         android: id="@+id/panic"
                         android: layout_width="64dp"
178
                         android: layout_height="64dp"
179
                         android: padding="10dp"
180
                         android: src="@drawable/panic"/>
181
                    <TextView
183
                         android: layout_width="wrap_content"
184
                         android: layout_height="wrap_content"
185
                         android: layout_marginTop="10dp"
186
                         android: background="@android:color/
        holo_blue_bright'
                         android: text="Don't Panic"
188
                         android: textColor="@color/colorAccent"
189
                         android: textStyle="bold" />
191
                    <View
192
193
                         android: layout_width="match_parent"
                         android: layout_height="1dp"
                         android: layout_margin="10dp"
195
                         android: background="@color/lightgray" />
106
197
                    <TextView
198
                         android: layout_width="wrap_content"
                         android: layout_height="wrap_content"
200
                         android: gravity="center"
201
                         android: padding="5dp"
202
                         android: text="Wajib Baca!!"
203
                         android: textColor="@android: color/darker_gray
204
                         android: textStyle="bold"/>
205
206
                </LinearLayout>
207
```

```
</android.support.v7.widget.CardView>
208
       </LinearLayout>
       <LinearLayout
            android: layout_width="match_parent"
213
            android: layout_height="wrap_content"
214
            android: clipToPadding="false"
215
            android: gravity="center"
216
            android: orientation="horizontal">
217
219
       </LinearLayout>
220
222 </LinearLayout>
```

Jika digabungkan dengan *code* slider sebelumnya maka seperti berikut:

```
<?xml version="1.0" encoding="utf-8"?>
  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/</pre>
       android"
      xmlns:app="http://schemas.android.com/apk/res-auto"
      xmlns:tools="http://schemas.android.com/tools"
      android: layout_width="match_parent"
      android: layout_height="match_parent"
      android: background="@color/backgroundcolor"
      android: gravity="center"
      android: orientation="vertical"
      android: padding="10dp">
      <com. daimajia. slider. library. SliderLayout
           android: id="@+id/slider"
13
           android: layout_width="match_parent"
           android:layout_height="150dp" />
15
16
      <LinearLayout
           android: layout_width="match_parent"
18
           android: layout_height="wrap_content"
           android: clipToPadding="false"
20
           android: gravity="center"
           android: orientation="horizontal">
23
          <android . support . v7 . widget . CardView
               android: id="@+id/bankcardId"
               android: layout_width="140dp"
26
               android: layout_height="160dp"
               android: layout_margin="10dp"
               android: clickable="true"
               android: foreground="? android: attr/
30
       selectableItemBackground">
               <LinearLayout
                   android: layout_width="match_parent"
33
                    android: layout_height="match_parent"
34
                    android: background="@color/colorPrimaryDark"
35
                    android: gravity="center"
                    android: orientation="vertical">
```

```
38
                   <ImageView
                        android: id="@+id/banjir"
                        android: layout_width="80dp"
41
                        android: layout_height="80dp"
42
                        android: padding="10dp"
                        android: src="@drawable/banjirr"/>
                   <TextView
46
                        android: layout_width="wrap_content"
47
                        android: layout_height="wrap_content"
                        android: layout_marginTop="10dp"
49
                        android: background="@android:color/
50
       holo_blue_bright"
                        android: text="About"
51
                        android: textColor="? attr/colorAccent"
                        android: textStyle="bold" />
54
                   <View
                        android: layout_width="match_parent"
                        android: layout_height="1dp"
                        android: layout_margin="10dp"
58
                        android: background="@color/lightgray" />
60
                   <TextView
                        android: layout_width="wrap_content"
                        android: layout_height="wrap_content"
63
                        android: gravity="center"
64
                        android: padding="5dp"
                        android: text="Apa itu Banjir?"
                        android: textColor="@android: color/darker_gray
67
       ,,
                        android: textStyle="bold"/>
68
               </LinearLayout>
70
           </android.support.v7.widget.CardView>
           <android . support . v7 . widget . CardView
               android: layout_width="140dp"
               android: layout_height="160dp"
76
               android: layout_margin="10dp"
               android: clickable="true"
               android: foreground="?android: attr/
       selectableItemBackground">
               <LinearLayout
80
                    android: layout_width="match_parent"
                    android: layout_height="match_parent"
82
                    android: background="@color/colorPrimaryDark"
83
                    android: gravity="center"
84
                    android: orientation="vertical">
                   <ImageView
87
                        android:id="@+id/awas"
88
                        android: layout_width="90dp"
89
                        android: layout_height="80dp"
```

```
android: padding="10dp"
91
                         android: src="@drawable/awas" />
                    <TextView
0.4
                         android: layout_width="wrap_content"
05
                         android: layout_height="wrap_content"
                         android: layout_marginTop="10dp"
97
                         android: background="@android:color/
       holo_blue_bright"
                         android: text="Ketinggian AIR"
                         android: textColor="@color/colorAccent"
                         android: textStyle="bold" />
101
102
                    <View
                         android: layout_width="match_parent"
104
                         android: layout_height="1dp"
                         android: layout_margin="10dp"
106
                         android: background="@color/lightgray" />
108
                    <TextView
                         android: layout_width="wrap_content"
                         android: layout_height="wrap_content"
                         android: gravity="center"
                         android: padding="5dp"
113
                         android: text="Monitoring"
114
                         android: textColor="@android: color/darker_gray
                         android: textStyle="bold"/>
116
                </LinearLayout>
118
            </android.support.v7.widget.CardView>
119
120
       </LinearLayout>
       <LinearLayout
123
            android: layout_width="match_parent"
124
            android: layout_height="wrap_content"
125
            android: clipToPadding="false"
126
            android: gravity="center"
            android: orientation = "horizontal">
128
           <android . support . v7 . widget . CardView
                android: layout_width="140dp"
                android: layout_height="160dp"
                android: layout_margin="10dp"
                android: clickable="true"
134
                android: foreground="? android: attr/
       selectableItemBackground">
136
                <LinearLayout
                     android: layout_width="match_parent"
                    android: layout_height="match_parent"
139
                     android: background="@color/colorPrimaryDark"
140
                     android: gravity="center"
141
                    android: orientation = "vertical">
142
```

```
<ImageView
144
                         android: id="@+id/air"
145
                         android: layout_width="80dp"
146
                         android: layout_height="80dp"
147
                         android: padding="10dp"
148
                         android: src="@drawable/air"/>
149
                    <TextView
                         android: layout_width="wrap_content"
                         android: layout_height="wrap_content"
                         android: layout_marginTop="10dp"
                         android: background="@android:color/
155
       holo_blue_bright"
                         android: text="Keterangan Air"
156
                         android: textColor="@color/colorAccent"
                         android: textStyle="bold" />
                    <View
160
                         android: layout_width="match_parent"
161
                         android: layout_height="1dp"
                         android: layout_margin="10dp"
                         android: background="@color/lightgray" />
164
165
                    <TextView
166
                         android: layout_width="wrap_content"
                         android: layout_height="wrap_content"
                         android: gravity="center"
169
                         android: padding="5dp"
170
                         android: text="Kamu Harus Tau"
                         android: textColor="@android: color/darker_gray
                         android: textStyle="bold"/>
173
174
                </LinearLayout>
            </android . support . v7 . widget . CardView>
176
           <android . support . v7 . widget . CardView
178
179
                android: layout_width="140dp"
                android: layout_height="160dp"
180
                android: layout_margin="10dp"
181
                android: clickable="true"
182
                android: foreground="?android: attr/
       selectableItemBackground">
19/
                <LinearLayout
185
                     android: layout_width="match_parent"
186
                     android: layout_height="match_parent"
                     android: background="@color/colorPrimaryDark"
188
                     android: gravity="center"
189
                     android: orientation = "vertical">
190
                    <ImageView
                         android: id="@+id/panic"
193
                         android: layout_width="64dp"
194
                         android: layout_height="64dp"
195
                         android: padding="10dp"
```

```
android: src="@drawable/panic"/>
197
                    <TextView
199
                         android: layout_width="wrap_content"
200
                         android: layout_height="wrap_content"
201
                         android: layout_marginTop="10dp"
202
                         android: background="@android:color/
       holo_blue_bright"
                         android: text="Don't Panic"
204
                         android: textColor="@color/colorAccent"
205
                         android: textStyle="bold" />
207
                    <View
208
                         android: layout_width="match_parent"
209
                         android: layout_height="1dp"
210
                         android: layout_margin="10dp"
                         android: background="@color/lightgray" />
                    <TextView
214
                         android: layout_width="wrap_content"
                         android: layout_height="wrap_content"
                         android: gravity="center"
                         android: padding="5dp"
218
                         android: text="Wajib Baca!!"
219
                         android: textColor="@android: color/darker_gray
                         android: textStyle="bold"/>
                </LinearLayout>
            </android.support.v7.widget.CardView>
224
225
       </LinearLayout>
226
       <LinearLayout
            android: layout_width="match_parent"
            android: layout_height="wrap_content"
230
            android: clipToPadding="false"
            android: gravity="center"
            android: orientation="horizontal">
234
235
       </LinearLayout>
  </LinearLayout>
```

4.1.3 Pembuatan Menu About

Menu About merupakan salah satu menu di aplikasi monitoring PKA. Menu ini berisi tentang sebab akibat terjadinya banjir. Dimana informasi tersebut disajikan melalui video dan tulisan.

Pertama yaitu harus membuat sebuah *activity* baru yang diberi nama *about*. Kemudian masukan video yang akan digunakan kedalam folder raw. Kemudian masukan *code* berikut ini kedalam file about.java

```
package com. example. lalitachandiany. monitoring;
1
  import android.content.Intent;
4 import android.net.Uri;
5 import android.provider.MediaStore;
6 import android.support.v7.app.AppCompatActivity;
  import android.os.Bundle;
  import android.view.View;
  import android.widget.Button;
  import android.widget.MediaController;
  import android.widget.VideoView;
  public class AboutActivity extends AppCompatActivity {
      private Button btnback;
15
      @Override
      protected void onCreate(Bundle savedInstanceState) {
          super.onCreate(savedInstanceState);
18
          setContentView (R. layout . activity_about);
          VideoView videoView = findViewById(R.id.video_view);
          Button btnback = (Button) findViewById(R.id.button1);
          String videoPath = "android.resource://" + getPackageName() +
24
      "/" + R.raw.about;
          Uri uri = Uri.parse(videoPath);
25
          videoView.setVideoURI(uri):
26
          MediaController mediaController = new MediaController(this);
          videoView.setMediaController(mediaController);
29
          mediaController.setAnchorView(videoView);
30
          btnback.setOnClickListener(new View.OnClickListener() {
               public void onClick(View v) {
                   // TODO Auto-generated method stub
34
                   startActivity (new Intent (AboutActivity.this,
      HomeActivity.class));
                   finish();
37
          });
38
      }
39
40
```

Untuk tampilanya masukan *code* berikut ini kedalam file activityabout.xml

```
<?xml version="1.0" encoding="utf-8"?>
 <LinearLayout
      xmlns: android="http://schemas.android.com/apk/res/android"
      xmlns:app="http://schemas.android.com/apk/res-auto"
4
      xmlns:tools="http://schemas.android.com/tools"
      android: layout_width="match_parent"
6
      android: layout_height="match_parent"
7
      tools: context="com. example. lalitachandiany. monitoring.
8
      AboutActivity">
      <LinearLayout
10
          android: layout_width="match_parent"
```

```
android: layout_height="match_parent"
           android: background="@drawable/bcempat"
           android: gravity="center"
14
           android: orientation = "vertical">
16
          < Video View
               android: id="@+id/video_view"
18
               android: layout_width="550dp"
19
               android: layout_height="250dp" />
20
21
          <TextView
               android: layout_width="wrap_content"
24
               android: layout_height="wrap_content"
               android: layout_marginTop="10dp"
26
               android: text="Sekilas Tentang banjir"
               android: textStyle="bold" />
28
20
          <View
30
               android: layout_width="match_parent"
               android: layout_height="1dp"
               android: layout_margin="10dp"
               android: background="@color/lightgray" />
34
35
          <TextView
               android: layout_width="wrap_content"
               android: layout_height="130dp"
38
               android: gravity="fill_vertical"
39
               android: padding="5dp"
               android:text="Banjir merupakan salah satu bencana yang
41
       sering terjadi di Indonesia khususnya didaerah kurang serapan air
       . Selain itu Banjir adalah kondisi dimana suatu wilayah terendam
       air karena luapan air yang berlebihan. Untuk lebih jelas Silahkan
      Tonton Video"
               android: textAppearance="@style/TextAppearance.AppCompat.
42
      Body2"
               android: textColor="@android: color/black"
43
               android: textSize="14sp"
               android: textStyle="bold" />
46
          <Button
47
               android:id="@+id/button1"
               android: layout_width="134dp"
               android: layout_height="42dp"
               android: background="@drawable/bcempat"
               android: text="Kembali"
52
               android: textColor="@color/colorAccent" />
54
55
      </LinearLayout>
56
57
58
  </LinearLayout>
```

4.1.4 Pembuatan Menu Ketinggian Air

Menu ketinggian air merupakan sebuah menu untuk menampilkan informasi mengenai ketinggian air yang dibaca oleh sensor jarak secara realtime.

Langkah pertama kita harus mendownload library antares dengan cara berikut :

- 1. Login ke antares kemudian pilih *documentation* lalu pilih android dan *go to tutorial*
- 2. Kemudian download library dengan cara memilih download.



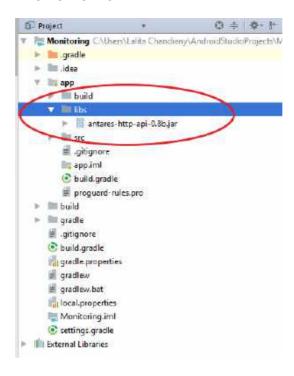
Gambar 4.14 Library Antares

3. Setelah itu masukan *library* yang telah di*download* ke folder libs yang ada di android studio.Klik pada bagian android kemudian pilih libs.



Gambar 4.15 Cara Masuk ke Folder Libs

4. Setelah itu *copy-Paste library* jar yang telah didownload ke app -libs. Maka jika sudah *copy - paste* tampilanya seperti berikut.



Gambar 4.16 Tampilan Folder Libs

 Agar bisa mengeksekusi API Antares, dibutuhkan akses Internet dan agar bisa mengakses Internet di Android, dengan tambahkan baris berikut di Android-Manifest.xml.

```
| <!--- IMPORTANT!!!!!!! --->
| 2 <!--- Tambahkan line berikut di AndroidManifest.xml Anda --->
| 3 <!--- Setelah manifest --->
| 4 < uses - permission android: name="android.permission.internet" />
| 5 <!--- Sebelum application --->
```

 Kemudian buat activity baru dengan nama monitoring. Activity ini bertujuan untuk menu ketinggian air. Lalu masukan code berikut kedalam file monitoring.java

```
package com.example.lalitachandiany.monitoring;

import android.content.Intent;
import android.os.Handler;
import android.os.Message;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

import org.json.JSONException;
```

```
import org. json. JSONObject;
16
  import id.co.telkom.iot.AntaresHTTPAPI;
  import id.co.telkom.iot.AntaresResponse;
20
  public class Monitoring Activity extends AppCompatActivity
      implements AntaresHTTPAPI. OnResponseListener {
    // Deklarasi button , text view dll atau deklarasi variabel
22
      private Button btnRefresh;
23
      private Button btnback;
24
      private TextView txtData, txtData2;
26
      private String TAG = "ANTARES-API";
      private AntaresHTTPAPI antaresAPIHTTP;
28
      private String dataDevice;
20
      private ProgressBar progressBar;
      private TextView persentase;
      private int Value;
      private int Hasil;
36
      @Override
38
      protected void on Create (Bundle saved Instance State) {
          super.onCreate(savedInstanceState);
40
          setContentView (R. layout. activity_monitoring);
41
     inisialisasi butron progresbar text view dll
42
          progressBar = findViewById(R.id.progress);
          persentase = findViewById(R.id.persentase);
44
          progressBar.setProgress(0); //Set Progress Dimulai Dari O
45
          // — Inisialisasi UI yang digunakan di aplikasi — //
          btnRefresh = (Button) findViewById(R.id.btnRefresh);
          btnback = (Button) findViewById(R.id.back);
          txtData = (TextView) findViewById(R.id.txtData);
50
          txtData2 = (TextView) findViewById(R.id.txtDatadua);
          // - Inisialisasi API Antares - //
          //antaresAPIHTTP = AntaresHTTPAPI.getInstance();
54
55
          antaresAPIHTTP = new AntaresHTTPAPI();
          antaresAPIHTTP.addListener(this);
          btnRefresh.setOnClickListener(new View.OnClickListener()
      {
               @Override
58
               public void onClick(View v) {
                   antaresAPIHTTP.getLatestDataofDevice("
      d5e3724d6720d3e2:164864b3c1aef838", "nodemcu-banjir","
      ketinggia-air");
61
          });
64
          Button btnback = (Button) findViewById(R.id.back);
66
```

```
// setonclik merupakan method untuk memberikan perintah apa yg
       hrus di lakukan ketika klik buton
           btnback.setOnClickListener(new View.OnClickListener() {
                public void onClick(View v) {
69
                    // TODO Auto-generated method stub
70
                    startActivity (new Intent (Monitoring Activity. this,
        HomeActivity.class));
                    finish();
               }
           });
74
       }
76
78
       @Override
79
       public void onResponse(AntaresResponse antaresResponse) {
           // — Cetak hasil yang didapat dari ANTARES ke System
81
       Log —— //
82
           Log.d(TAG, Integer.toString(antaresResponse.getRequestCode
83
       ()));
           if (antares Response . getRequestCode () == 0) {
84
                try {
85
                    JSONObject body = new JSONObject(antaresResponse.
86
       getBody());
                    dataDevice = body.getJSONObject("m2m:cin").
87
       getString("con");
                    runOnUiThread(new Runnable() {
88
                        @ Override
                        public void run() {
90
                             //txtData.setText(dataDevice);
91
                             String s = dataDevice;
92
                             String parts[] = s.split(":");
93
95
                             txtData.setText(parts[1].substring(0,1));
96
                             Hasil = Integer.parseInt(parts[1].
97
       substring (0,1);
                             persentase.setText(String.valueOf(Hasil)
98
       + "m");
                             txtData2.setText(parts[2].substring(0,3))
QQ
                             final Handler handler = new Handler(){
100
                                 @Override
101
                                 public void handleMessage (Message msg
       ) {
                                     super.handleMessage(msg);
103
                                     // Menampung semua data yang
104
       ingin diproses oleh thread
                                     persentase.setText(String.valueOf
105
       (Value)+ "m");
106
107
                                      if (Value == Hasil && Value != 0){
108
```

```
Toast.makeText(
109
       getApplicationContext(), "Progress Completed", Toast.
       LENGTH_SHORT).show();
                                       Value++:
                                  }
113
                              };
114
116
                              Thread thread = new Thread(new Runnable()
        {
                                  @Override
118
                                  public void run() {
                                      try {
120
                                           for (int w=0; w \le Hasil; w++)
                                                progressBar.setProgress(w
       ); // Memasukan Value pada ProgressBar
                                                // Mengirim pesan dari
       handler, untuk diproses didalam thread
                                                handler.sendMessage(
124
       handler.obtainMessage());
                                                Thread.sleep(100); //
125
       Waktu Pending 100ms/0.1 detik
                                       }catch(InterruptedException ex){
                                           ex.printStackTrace();
128
                                       }
129
                              });
                              thread.start();
134
136
                         }
                     });
                    Log.d(TAG, dataDevice);
138
139
                } catch (JSONException e) {
140
                     e.printStackTrace();
141
142
           }
```

7. Kemudian untuk tampilan nya masukan *code* berikut kedalam file activitymonitoring.xml

```
tools: context="com. example. lalitachandiany. monitoring.
      Monitoring Activity">
      <TextView
10
           android: id="@+id/txtData"
           android: layout_width="wrap_content"
           android: layout_height="wrap_content"
           android: layout_above="@+id/persentase"
           android:layout_marginBottom="16dp"
15
           android: layout_toStartOf="@+id/back"
16
           android: background="@android: color/holo_blue_bright"
           android: text="Distance"
18
           android: textSize="24sp" />
19
20
      <TextView
           android: id="@+id/txtDatadua"
           android: layout_width="wrap_content"
23
           android: layout_height="wrap_content"
24
           android: layout_alignBaseline="@+id/txtData"
           android: layout_alignBottom="@+id/txtData"
           android: layout_alignEnd="@+id/progress"
           android: background="@android: color/holo_blue_bright"
28
           android: text="Duration"
           android: textSize="24sp" />
34
      <Button
           android: id="@+id/btnRefresh"
           android: layout_width="wrap_content"
           android: layout_height="wrap_content"
           android: layout_alignParentBottom="true"
           android: layout_centerHorizontal="true"
           android: layout_marginBottom="81dp"
41
           android: backgroundTint="@color/colorPrimaryDark"
           android: text="Refresh" />
      <TextView
           android: id="@+id/textView"
47
           android: layout_width="wrap_content"
           android: layout_height="wrap_content"
           android: layout_alignParentTop="true"
           android: layout_centerHorizontal="true"
           android: layout_marginTop="34dp"
           android: shadowColor="@color/cardview_dark_background"
52
           android: text="Ketinggian AIR Saat Ini:"
           android: textColor="@android: color/holo_red_dark"
54
           android: textSize="30sp" />
55
      <TextView
           android: id="@+id/textView2"
           android: layout_width="wrap_content"
           android: layout_height="wrap_content"
60
           android: layout_alignParentStart="true"
           android: layout_below = "@+id / textView"
62
```

```
android: layout_marginStart="11dp"
           android: layout_marginTop="41dp"
           android: text="Distance"
           android: textColor="@android: color/background_dark"
66
           android: textSize="24sp" />
67
       <TextView
           android: id="@+id/textView3"
           android: layout_width="wrap_content"
           android: layout_height="wrap_content"
72
           android: layout_alignBaseline="@+id/textView2"
           android: layout_alignBottom="@+id/textView2"
74
           android: layout_alignStart="@+id/txtDatadua"
75
           android: text="Duration"
76
           android: textColor="@android: color/background_dark"
           android: textSize="24sp" />
       <Button
80
           android: id="@+id/back"
81
           android: layout_width="134dp"
           android: layout_height="40dp"
           android: layout_alignParentBottom="true"
84
           android: layout_centerHorizontal="true"
85
           android: layout_marginBottom="15dp"
86
           android: background="@drawable/bcempat"
           android: text="Kembali" />
89
       < Progress Bar
90
           android: id="@+id/progress"
           style="?android: attr/progressBarStyleHorizontal"
92
           android: layout_width="300dp"
           android: layout_height="wrap_content"
           android: layout_below = "@+id / persentase"
           android: layout_centerHorizontal="true"
           android: layout_gravity="center_vertical"
97
           android: background="@android:color/holo_blue_bright"
98
           android: max="100"
           android: progress="0" />
       <TextView
103
           android: id="@+id/persentase"
           android: layout_width="50dp"
           android: layout_height="wrap_content"
           android: layout_centerHorizontal="true"
           android: layout_centerVertical="true"
           android: background="@android: color/holo_blue_bright"
108
           android: gravity="center"
           android: text="0 m"
           android: textColor="@color/colorAccent"
           android: textSize="15sp"
           android: textStyle="bold" />
114
  </RelativeLayout>
```

4.1.5 Pembuatan Menu Keterangan Air

Menu keterangan air adalah sebuah menu yang menampilkan sebuah halaman yang berisi tentang keterangan level-level ketinggian air.

Untuk membuat menu keterangan air ini hal yang pertama kali harus dibuat yaitu membuat sebuah *activity* baru dengan nama ketinggian air. Kemudian masukan *code* berikut kedalam file ketinggianair.java

```
package com. example. lalitachandiany. monitoring;
  import android.content.Intent;
  import android.support.v7.app.AppCompatActivity;
  import android.os.Bundle;
  import android.view.View;
  import android.widget.Button;
  public class AirActivity extends AppCompatActivity {
9
      private Button btnback;
10
      @Override
      protected void onCreate(Bundle savedInstanceState) {
          super.onCreate(savedInstanceState);
          setContentView (R. layout . activity_air);
14
           Button btnback = (Button) findViewById(R.id.button1);
          btnback.setOnClickListener(new View.OnClickListener() {
               public void onClick(View v) {
19
                   // TODO Auto-generated method stub
20
                   startActivity (new Intent (AirActivity.this,
      HomeActivity.class));
                   finish();
22
               }
          });
24
      }
25
```

Kemudian untuk tampilan menu ketinggian airnya masukan perintah berikut di file activityketinggianair.xml

```
<?xml version="1.0" encoding="utf-8"?>
 <LinearLayout
      xmlns: android="http://schemas.android.com/apk/res/android"
      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"
      tools:context="com.example.lalitachandiany.monitoring.AirActivity
8
      <LinearLayout
          android: layout_width="match_parent"
          android: layout_height="match_parent"
          android: background="@drawable/bcempat"
          android: gravity="center"
14
          android: orientation = "vertical">
16
```

```
<ImageView
               android: id="@+id/about"
18
               android: layout_width="263dp"
19
               android: layout_height="150dp"
20
               android: padding="10dp"
               android: src="@drawable/ombak" />
          <TextView
24
               android: layout_width="wrap_content"
               android: layout_height="wrap_content"
26
               android: layout_marginTop="10dp"
27
               android: text="Ketinggian Air"
28
               android: textStyle="bold" />
29
30
          <View
               android: layout_width="match_parent"
               android: layout_height="1dp"
               android: layout_margin="10dp"
34
               android: background="@color/lightgray" />
          <TextView
               android: layout_width="278dp"
38
               android: layout_height="162dp"
30
               android: gravity="center"
40
               android: padding="5dp"
               android:text="Jika sensor membaca jarak ketinggian air
42
      20-10m maka aman dari bahaya banjir.
               Apabila sensor membaca ketinggian air kurang dari 5m dan
43
       adanya suara serta lampu led menyala artinya AWAS BANJIR !!"
               android: textAppearance="@style/TextAppearance.AppCompat.
44
      Body2" />
45
46
          <Button
               android:id="@+id/button1"
               android: layout_width="136dp"
               android: layout_height="52dp"
49
               android: background="@drawable/bcempat"
50
               android: text="Kembali" />
51
53
      </LinearLayout>
  </LinearLayout>
```

4.1.6 Pembuatan Menu Don't Panic

Pada menu *don't ponic* akan menampilkan sebuah video dan text yang berisi tentang langkah-langkah apa saja yang harus dilakukan ketika datangnya banjir. Isi pada menu ini bertujuan agar *user* tidak panik dan tau apa yang harus dilakukan ketika banjir datang.

Untuk membuatnya sama seperti langkah sebelumnya yaitu membuat activity baru dengan nama panic, kemudian masukan code ke file panic.java sebagai berikut.

package com.example.lalitachandiany.monitoring;

```
import android.content.Intent;
4 import android.net.Uri;
5 import android.support.v7.app.AppCompatActivity;
6 import android.os.Bundle;
7 import android.view.View;
8 import android.widget.Button;
  import android.widget.MediaController;
  import android.widget.VideoView;
  public class Panic Activity extends App Compat Activity {
      private Button btnback;
13
14
      @Override
      protected void onCreate(Bundle savedInstanceState) {
16
          super.onCreate(savedInstanceState);
          setContentView (R. layout . activity_panic);
10
          VideoView videoView = findViewById(R.id.video_view);
           Button btnback = (Button) findViewById(R.id.button1);
           String videoPath = "android.resource://" + getPackageName() +
      "/" + R.raw.panic;
          Uri uri = Uri.parse(videoPath);
24
          videoView.setVideoURI(uri);
25
           MediaController mediaController = new MediaController(this);
          videoView.setMediaController(mediaController);
28
           mediaController.setAnchorView(videoView);
30
          btnback.setOnClickListener(new View.OnClickListener() {
               public void onClick(View v) {
                   // TODO Auto-generated method stub
                   startActivity (new Intent (Panic Activity . this,
35
      HomeActivity.class));
36
                   finish();
               }
          });
38
30
      }
40
```

Kemudian untuk tampilan menu *don't pannic* masukan perintah berikut di file panic.xml

```
android: layout_height="match_parent"
           android: background="@drawable/bcempat"
           android: gravity="center"
14
           android: orientation = "vertical">
16
          < Video View
               android: id="@+id/video_view"
18
               android: layout_width="550dp"
               android: layout_height="250dp" />
20
21
          <TextView
22
               android: layout_width="wrap_content"
23
               android: layout_height="wrap_content"
24
               android:layout_marginTop="10dp"
               android: text="Ikuti Petunjuk"
26
               android: textStyle="bold" />
27
28
          <View
               android: layout_width="match_parent"
30
               android: layout_height="1dp"
               android: layout_margin="10dp"
               android: background="@color/lightgray" />
34
          <TextView
35
               android: layout_width="284dp"
               android: layout_height="103dp"
               android: gravity="fill_vertical"
38
               android: padding="5dp"
39
               android:text="Pada saat banjir masyarakat tidak harus
       panik hal pertama yang harus di lakukan :
   1. Selmatkan barang-barang berharga.
41
   2. Pergi ketempat yang lebih tinggi.
42
   3. Selalu pantau tentang informasi Banjir."
43
               android: textAppearance="@style/TextAppearance.AppCompat.
      Body2" />
45
          <Button
46
               android:id="@+id/button1"
               android: layout_width="134dp"
               android: layout_height="40dp"
49
               android: background="@drawable/bcempat"
50
               android: text="Kembali" />
54
55
      </LinearLayout>
  </LinearLayout>
```

4.2 Tampilan Aplikasi Monitoring

Tampilan dari aplikasi monitoring yang sebelumnya telah dibuat, jika mengikuti semua langkah-langkahnya tanpa adanya step yang terlewatkan maka akan menjadi seperti berikut:



Gambar 4.17 Tampilan Aplikasi DI Emulator



Gambar 4.18 Splash Screen



Gambar 4.19 Tampilan Menu Utama



Gambar 4.20 Tampilan Menu About



Gambar 4.21 Tampilan Ketinggian Air



Gambar 4.22 Tampilan Menu Keterangan Air



Gambar 4.23 Tampilan Menu Don't Panic



Gambar 4.24 Tampilan Notifikasi Telegram

PEMBUATAN AUGMENTED REALITY PROTOTIPE PKA

Pada bab ini membahas mengenai pembuatan augmented reality. Pembahasan mengenai augmented reality telah dijelaskan pada bab sebelumnya. Untuk membuat augmented reality hal pertama yang harus dilakukan yaitu menDownload *software* Unity dengan menggunakan Unity Hub.

Unity Hub adalah aplikasi yang digunakan oleh Unity untuk mengatur segala kebutuhan dasar seperti:

- 1. Manajemen akun Unity.
- 2. Manajemen Lisensi.
- 3. Membuat Project baru.
- 4. Menginstall dan mengatur versi Unity yang digunakan pada Project.
- 5. Menjalankan Unity dengan berbagai versi.
- 6. Mengatur target build dari project tanpa harus membuka Unity Editor.
- 7. Menjalankan dua Unity Editor sekaligus tanpa harus bentrok.

5.1 Install Unity 3D

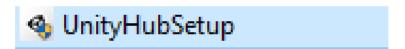
Langkah pertama yang harus dilakukan yaitu mendownload *software* unity. Untuk menDownload software unity lebih baik menggunakan unity hub. Berikut langkah - langkahnya:

1. Kunjungi https://unity3d.com/get-unity/download kemudian klik Unity Hub.



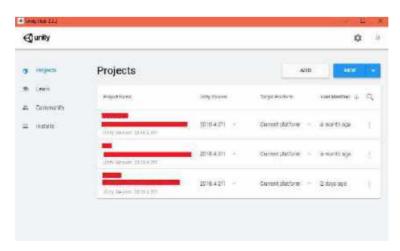
Gambar 5.1 Tampilan Halaman Download Unity Hub

2. Jika sudah terDownload *install* file unityhub.exe seperti pada umumnya menginstall.



Gambar 5.2 Unity Hub. Exe

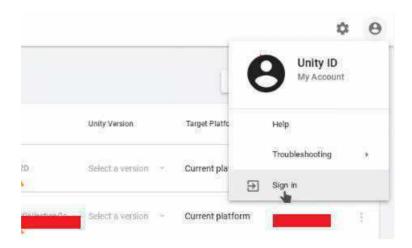
3. Jika sudah terinstall sebelumnya tampilan awal ketika membuka unity hub,menampilkan daftar project yang sudah dibuat sebelumnya seperti gambar dibawah ini.



Gambar 5.3 Tampilan Awal Unity Hub

Tetapi jika pertama kali menginstallnya maka tidak akan ada daftar project.

4. Masuk ke akun unity dengan cara Klik ikon profile pada pojok kanan atas, lalu pilih *Sign In*.



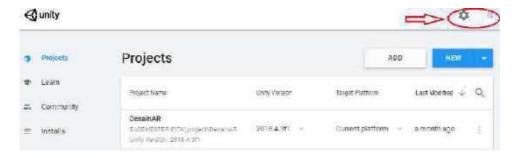
Gambar 5.4 *Sign In* Unity Hub

5. Maka akan muncul jendela *sign in*. Kemudian masukan username dan password dari akun Unity yang sudah dibuat atau bisa login dengan Facebook dan Google.



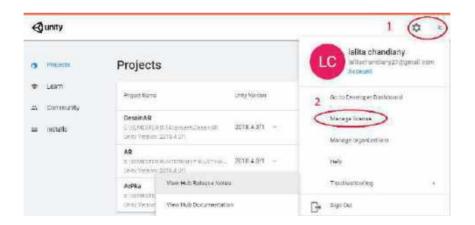
Gambar 5.5 Sign In Unity Hub

Jika sudah berhasil login, maka pada pojok kanan atas akan ada nama dari akun yang telah kita buat. Seperti berikut.



Gambar 5.6 Tampilan Sesudah Login Unity Hub

6. Sebelum menginstal Unity, harus mengatur lisensinya terlebih dahulu. Tapi jika telah melakukannya, hanya perlu mengupdatenya saja. klik menu profile kemudian pilih Manage License.



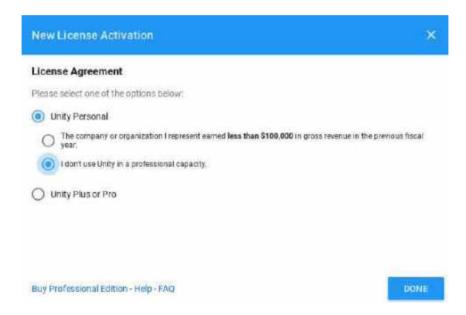
Gambar 5.7 Manage License

7. Setelah itu, akan muncul jendela untuk mengatur lisensi. Kemudian klik *Active new license* jika belum mempunyai license dan klik *Check for updates* jika sudah mempunyai license.



Gambar 5.8 Tampilan Manage License

8. Klik Activate New License untuk membuat lisensi baru.



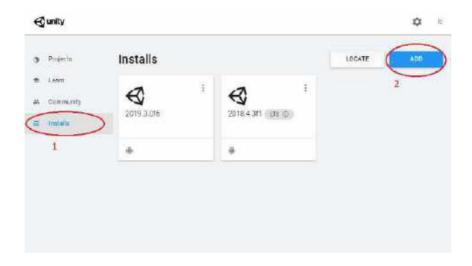
Gambar 5.9 Tampilan License Activation

Keterangan:

- (a) Unity Personal: jika menggunakan unity secara pribadi dan untuk belajar.
- (b) Unity Plus or Pro: jika akan membuat game untuk di publish dengan serius atau perusahaan.

Karena akan menggunakan Unity untuk belajar saja, jadi pilihlah lisensi Unity Personal: I dont use Unity in a professional capacity. Tetapi jika akan menggunakannya untuk membuat game dan menghasilkan uang dapat diubah jenis lisensinya.

9. Setelah itu *install* Unity dengan cara masuk ke menu Install dan pilih ADD untuk menginstal Unity.



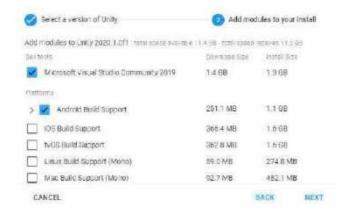
Gambar 5.10 Tampilan Menu Install

10. Maka akan muncul jendela untuk memilih versi Unity yang akan diinstal. Pilih salah satu yang akan diinstal, kemudian klik NEXT.



Gambar 5.11 Tampilan Menu Install

11. Kemudian, akan diminta untuk menentukan modul apa saja yang akan diinstal. Pilih Android Build Support, Windows Build Support (Mono) dan Vuforia dan Documentation jika ingin menambah modul silahkan centang atau pilih sesuai kebutuhan.



Gambar 5.12 Tampilan Memilih Modul

12. Setelah itu klik Done. Maka Unity Hub akan melakukan download.

5.2 Membuat Augmented Reality PKA

Setelah menginstall unity dengan mengikuti langkah-langkahnya, selanjutnya membuat sebuah augmented reality PKA. Untuk membuatnya terlebih dahulu harus membuat sebuah database marker dengan menggunakan vuforia developer.

5.2.1 Membuat Database Marker

Database marker ini berfungsi sebagai penyimpanan marker. Marker tersebut dapat berupa gambar barcode, kartun, dll. Untuk membuat database marker ini menggunakan vuforia developer. Berikut langkah - langkahnya:

- 1. Kunjungi https://developer.vuforia.com/.
- 2. Kemudian klik login jika sudah mempunyai akun dan klik registrasi jika belum memiliki akun.



Gambar 5.13 Tampilan vuforia

jika belum mempunyai akun klik registrasi kemudian masukan nama , negara, dll yang terdapat di dalam kolom. Kemudian

ens Nerre * Alica	Chendleny
positive?	En 1974 Indonésia
rui Midmit*	USATURNA *
ecword?	Cardini Kasawoni,**
✓ 1m not a robot	

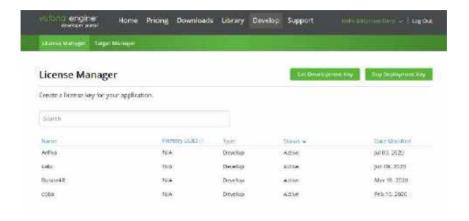
Gambar 5.14 Tampilan Registrasi Vuforia

Jika sudah mempunyai akun klik login kemudian masukan email dan password yang sudah terdaftar.

vulcrio engine	Home Priding Downloads Library Develop Suppor	t togái (legisle)
	Login Login with your Vuloria developer account to (seemload software get bornse keys and participate in the Vuloria community	
	Types Com Test Reported Come proper Year Property Come property Year	
	F12gt1 parameter	

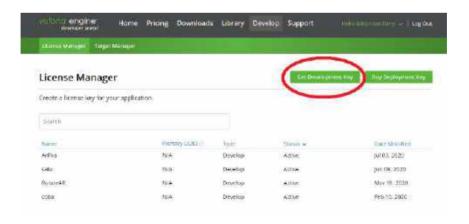
Gambar 5.15 Tampilan Login Vuforia

3. Setelah login akan menampilkan halaman awal yang dimana apabila telah membuat sebuah database sebelumnya, akan ditampilkan database yang telah dibuat tersebut.



Gambar 5.16 Tampilan Halaman Awal Vuforia

4. Selanjutnya klik Get Development Key untuk mendapatkan license key.



Gambar 5.17 Development Key Vuforia

5. Jika telah diklik maka akan muncul sebuah halaman yang harus mengisikan nama license yang akan dibuat kemudian ceklis bagian paling bawah dan klik *confirm*.

Valoria engine	Home Pricing	Downloads	Library	Develop	Support	Hulla fellomandum — Loe DA
Don't Manager Torge	f Manageri					
Sect to Laure Meneger						
Add a free Dev	elopment Lice	nse Key				
erika kod san manganik satar License Key Dewloo						
Frice, No Charge Reco Usago: 1,000 per mor Cloud Targets: 1,000 VisMark Templates: 1 Action VisMarke: 100						
a by chadling this act, i at the certis and condition	continuesige that this isce and the violence surveyop	nse way is subject or agreement.	10			
Contes						

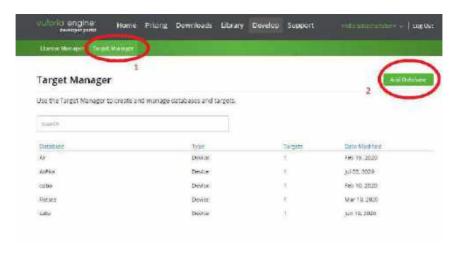
Gambar 5.18 Development Key Vuforia

Apabila sudah membuat license key maka akan adanya sebuah license key yang nantinya akan digunakan pada saat pembuatan Augmented reality.

vulctic engine		Pricing	Downloads	Library	Develop	Support	
Litema Manager	farget Manager						
Lichardiniger + Affici							
ArPka	Sideht Loanne Cop						
Literse Sky	1446						
Please copy the Been	se key below into y	our app					
AS-TOTTO://FAARTR cgATe.ug.TTGgRoTevit 02Gp-bdev/eFREDeaa 02w.0100HBoFCRbook	EEE LYSIMOTTOAKTYATAI PLOKUIDOKKKUODABET	estatines	achtarite Tolary daelettelledig	Challe to The	CHECKE STERM	(Jentings) grains	
Plan Type: Develop							
Status: Active							
Crested Jul 10, 2020	12-56						
Chester In my total							

Gambar 5.19 License Key Vuforia

6. Untuk membuat database markernya yaitu dengan cara klik "**Target Manager**" kemudian klik "**Add Database**".



Gambar 5.20 Database Vuforia

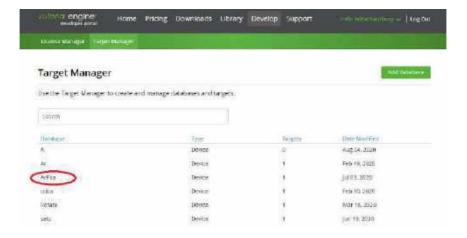
7. Selanjutnya akan muncul jendela create database. Dimana harus megisikan nama database dengan memilih *tipe* "**Device**". Nama dari database dapat disesuaikan dengan kebutuhan.

Database Name * ArPka		
Туре:		
Device		
O Cloud		
O VuMark		

Gambar 5.21 Create Database Vuforia

Jika sudah mengisi nama database kemudian klik "**create**" sehingga database langsung terbuat.

8. Apabila database sudah terbuat maka akan langsung muncul pada halaman awal target manager. Kemudian klik database yang telah dibuat.



Gambar 5.22 Database Vuforia

9. Setelah diklik database yang telah dibuat kemudian tambahkan marker atau target dengan cara klik "**target**"



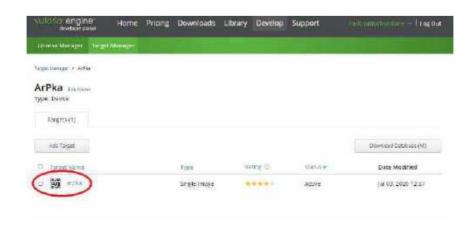
Gambar 5.23 Menambahkan Target Vuforia

10. Selanjutnya akan muncul sebuah jendela add target . Dimana kita harus memasukan sebuah gambar yang akan dijadikan sebagai marker Augmented Reality yang akan dibuat dengan tipe *single image*, memasukan width dan Name. Jika sudah klik "**Add**"

Add Target			
Type:			
		00	
Single Image	Cabold	Cylinder	30 Object
File:			
w.i/e			Growse
log or png (max file	lmb;		
Width:			
50			
same scale as your a unit scale. The sarge	augmented virtual o	ints. The size of the ta orgent, Yufforla uses in culated when you uplo	eters as the default
Name			
Name: aroka			
arpka		han a sarget is dietecte	d in your application

Gambar 5.24 Add Target Vuforia

Apabila sudah ditambahkan targetnya maka akan secara otomatis pada databasenya tertambahkan untuk melihat marker atau target yang telah ditambahkan klik nama targetnya dan langsung muncul marker atau target yang telah ditambahkan sebelumnya.



Gambar 5.25 Marker atau Target Vuforia

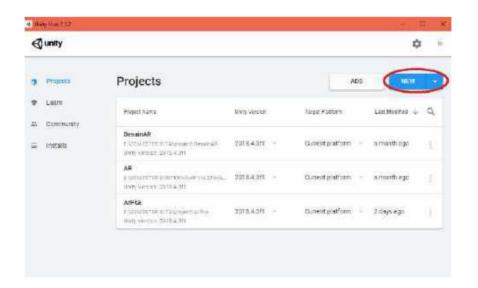


Gambar 5.26 Marker atau Target Yang Ditambahkan

5.2.2 Membuat Augmented Reality PKA Pada Unity

Tahapan selanjutnya setelah membuat database marker yaitu membuat sebuah augmented reality PKA. Membuat augmented reality ini menggunakan sebuah *software* unity. Pembahasan mengenai unity sudah dibahas pada bab sebelumnya. Berikut langkah - langkah membuat augmented reality PKA:

1. Pertama buka unity hub dan klik "New"



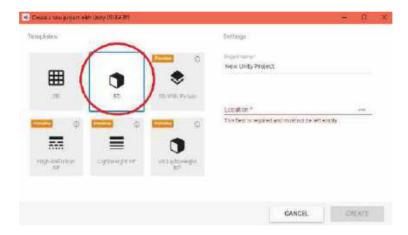
Gambar 5.27 Tampilan Unity Hub

Jika sudah mengklik new maka akan muncul jendela create new project.



Gambar 5.28 Tampilan Create New Project

2. Pilih templates 3D.



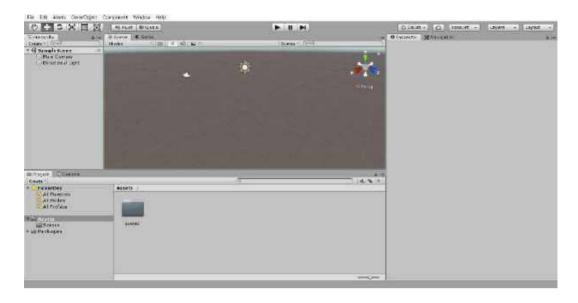
Gambar 5.29 Templates 3D

3. Masukan Project name dan location. Pada hal ini dapat disesuaikan sesuai kebutuhan.



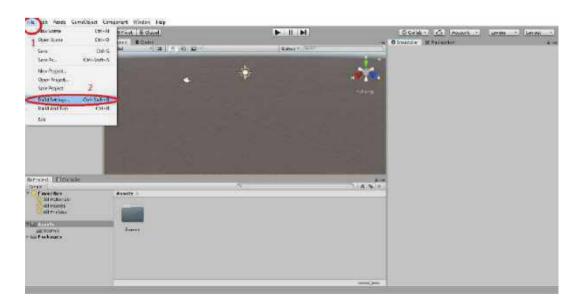
Gambar 5.30 Project Name dan Location

Setelah memasukan keduanya lalu klik "Create" kemudian akan muncul tampilan awal unity seperti berikut.



Gambar 5.31 Project Name dan Location

4. Apabila telah muncul tampilan awal unity selanjutnya mengatur XR setting untuk mengaktifkan (ON) vuforia Augmented Realitynya. Dengan cara klik file kemudian Build Settings.



Gambar 5.32 Mengatur XR Setting

5. Maka akan tampil jendela Build Setting lalu klik "Player Settings.



Gambar 5.33 Mengatur XR Setting

Maka akan muncul Inspector Player Setting.



Gambar 5.34 Inspector Player Setting

6. Setelah muncul inspector ceklis pada bagian Vuforia Augmented reality. Hal ini bertujuan agar pada saat proses *run* tidak ada error dan agar support vuforia.



Gambar 5.35 Inspector Player Setting

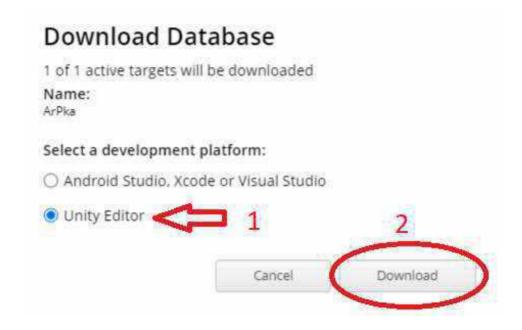
Tahapan selanjutnya yaitu mengimport package database marker, dengan cara menDownload terlebih dahulu database marker yang sudah dibuat sebelumnya.

7. Login vuforia developer kemudian klik target manager dan pilih database yang sebelumnya dibuat. Setelah muncul halaman database klik "Download database (all)".



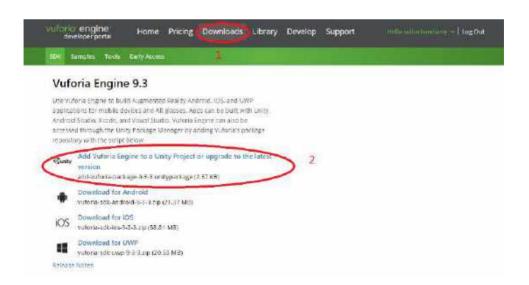
Gambar 5.36 Download Database Marker

8. Maka akan muncul sebuah tampilan Download database, pilih tipe "Unity Editor kemudian klik Download.



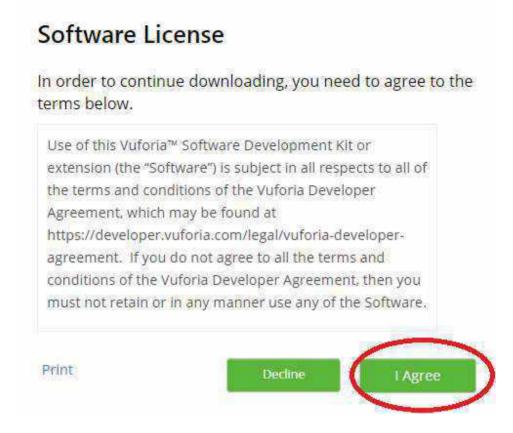
Gambar 5.37 Download Database Marker

9. Setelah menDownload database selanjutnya menDownload SDK Vuforia Engine dengan cara masuk pada menu Download kemudian pilih "Add Vuforia Engine to a Unity Project or upgrade to the latest version."



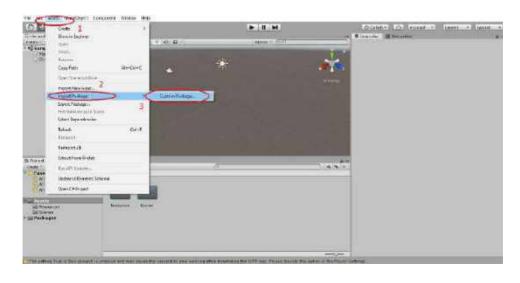
Gambar 5.38 Download SDK Vuforia Engine

10. Jika muncul jendela Software License klik "*Agree*" sehingga SDknya otomatis terDownload.



Gambar 5.39 Download SDK Vuforia Engine

11. Apabila sudah menDownload database dan SDK, selanjutnya mengimport keduanya kepada unity. Pertama import SDKnya terlebih dahulu dengan cara klik "asset - Import Package - Custome Package".



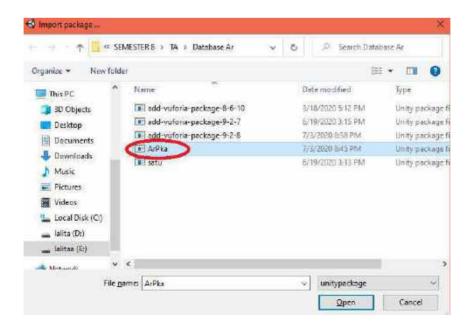
Gambar 5.40 Import Package

12. Kemudian pilih file SDK vuforia yang telah diDownload.



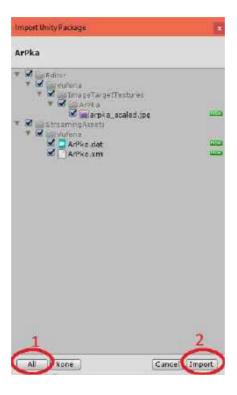
Gambar 5.41 Import Package SDK Vuforia

13. Selanjutnya import database marker yang telah diDownload dengan cara yang sama .



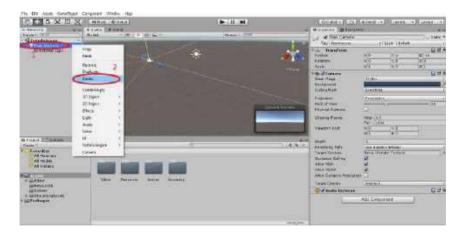
Gambar 5.42 Import Package Database

14. Lalu klik "all" kemudian klik "import"



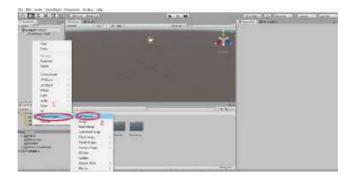
Gambar 5.43 Import Package Database

15. Jika semuanya di import, selanjutnya yaitu menghapus main camera dengan cara klik kanan pada "main camera" lalu klik "delete



Gambar 5.44 Delete Main Camera

16. Main camera telah dihapus selanjutnya menambahkan AR camera yaitu klik kanan pada "scene kemudian klik "Vuforia Engine" lalu klik "AR Camera".



Gambar 5.45 Menambahkan AR Camera

Jika muncul jendela Vuforia engine Asset klik "import".



Gambar 5.46 Import AR Camera

17. Selanjutnya menambahkan Image target dengan cara klik kanan pada "scene kemudian klik "Vuforia Engine" lalu klik "Image".



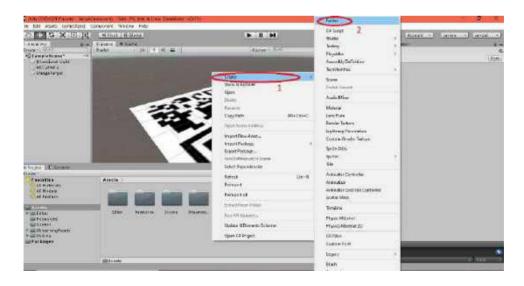
Gambar 5.47 Menambahkan Image Target

Apabila image target sudah ditambahkan dan database marker telah diimport maka akan muncul sebuah image target dengan gambar marker yang sesuai pada database yang sebelumnya telah dibuat.



Gambar 5.48 Image Target Marker

18. Tambahkan sebuah 3D objek yang akan dimunculkan ketika *scan* marker. Cara menambahkannya yaitu buat terlebih dahulu folder dengan cara klik kanan pada bagian "asset kemudian klik "Create" lalu pilih "Folder" beri nama disesuaikan kebutuhan.



Gambar 5.49 Membuat Folder

Apabila telah membuat folder maka secara otomatis akan ada sebuah folder baru dalam asset.



Gambar 5.50 Folder Yang Sudah Dibuat

19. Kemudian drag 3D objek yang telah dibuatkan sebelumnya ke dalam folder yang telah dibuat.



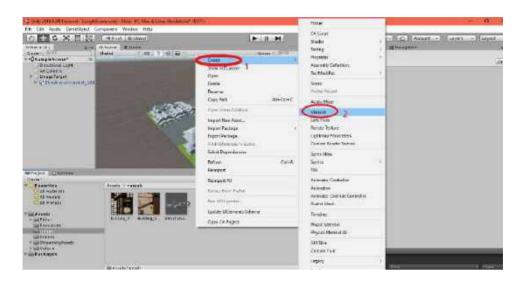
Gambar 5.51 Menambahkan 3D Objek ke dalam Folder

20. Jika sudah ditambahkan ke dalam folder kemudian drag 3D objek tersebut ke dalam Image Target.



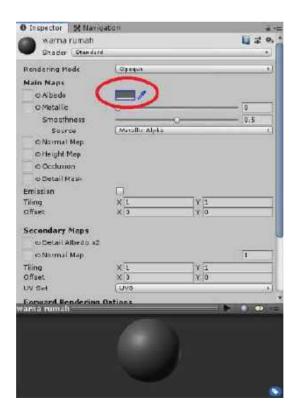
Gambar 5.52 Menambahkan 3D Objek ke dalam Image Target

21. Selanjutnya setelah menambahkan 3D objek rumah ke dalam image target langkah berikutnya memberi warna pada rumah dengan cara membuat material pada folder rumah. Klik kanan kemudian klik "Create" lalu "Material".



Gambar 5.53 Menambahkan Material

22. Pilih warna yang diinginkan dengan cara klik paint tool.



Gambar 5.54 Menambahkan Warna Material

23. Pilih warna yang diinginkan dan disesuaikan dengan kebutuhan.



Gambar 5.55 Menambahkan Warna Material

24. Drag material pada 3D objek rumah yang ingin diberi warna.



Gambar 5.56 Menambahkan Warna Material Pada 3D Objek

25. Atur ukuran 3D objek rumah dengan cara klik "3D objek" lalu klik "scale tool"



Gambar 5.57 Mengatur Ukuran 3D Objek

26. Selanjutnya atur posisi 3D objek rumah hingga posisinya berada diatas image target dengan cara klik "3D objek" kemudian klik "Move tool"



Gambar 5.58 Mengatur Posisi 3D objek

Atur posisi 3D objek rumah hingga berada diatas image target seperti pada gambar berikut :



Gambar 5.59 Posisi 3D Objek Rumah

27. Tambahkan sungai dengan cara menambahkan 3D objek cube dengan cara klik kanan pada nama "3D objek" kemudian klik "3D object" lalu pilih "cube".



Gambar 5.60 Menambahkan 3D Objek Cube

28. Atur posisi dan ukuran 3D objek cube seperti persegi panjang agar mirip seperti sungai.



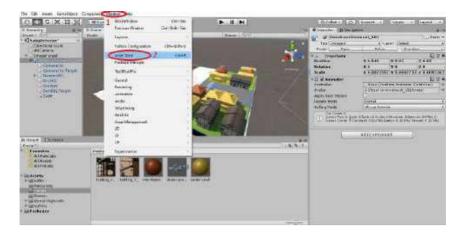
Gambar 5.61 Mengatur Ukuran dan Posisi 3D Cube

Atur posisi dan ukuran cube hingga seperti pada gambar berikut :



Gambar 5.62 Posisi dan Ukuran 3D Cube

29. Agar terlihat seperti sungai sungguhan tambahkan texture dengan cara klik "windows" lalu klik "Asset Store



Gambar 5.63 Asset Store

30. Atur ukuran tampilan asset store dengan cara klik kanan halaman asset store kemudian klik **"Maximize".**



Gambar 5.64 Maximize Asset Store

31. Cari texture air dengan cara ketikan "Stylize Water Texture" pada kolom search lalu pilih texture yang gratis untuk di Download.



Gambar 5.65 Mendownload Texture

32. Jika sudah pernah mendownloadnya klik import dan jika belum klik Download kemudian import.



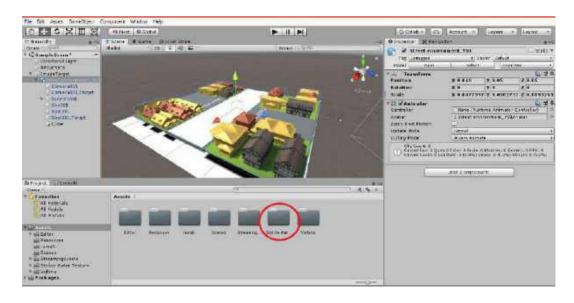
Gambar 5.66 Download Texture

33. Klik "All" lalu klik "Import"



Gambar 5.67 Import Texture

34. Secara otomatis maka akan langsung tertambahkan folder texture pada asset lalu klik folder tersebut.



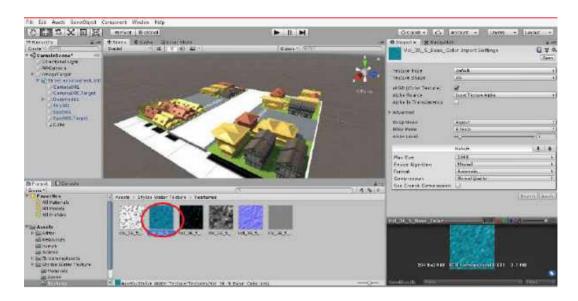
Gambar 5.68 Folder Texture

35. Kemudian klik Folder "Texture"



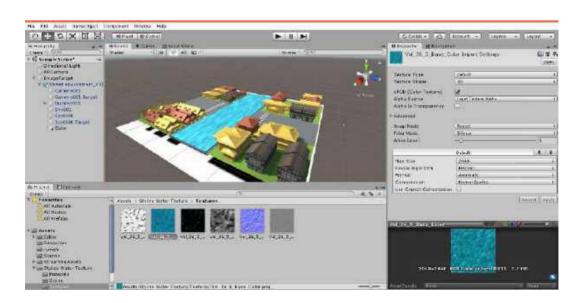
Gambar 5.69 Folder Texture

36. Lalu drag texture air kepada 3D object cube.



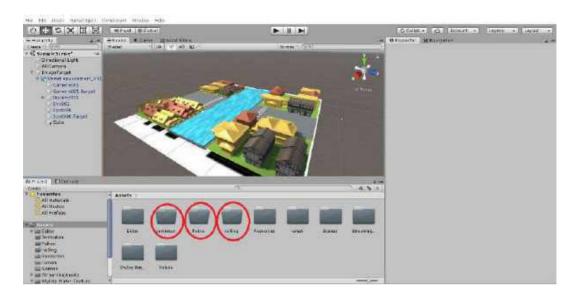
Gambar 5.70 Memberi Warna 3D Cube

Drag texture air hingga seperti gambar berikut ini:



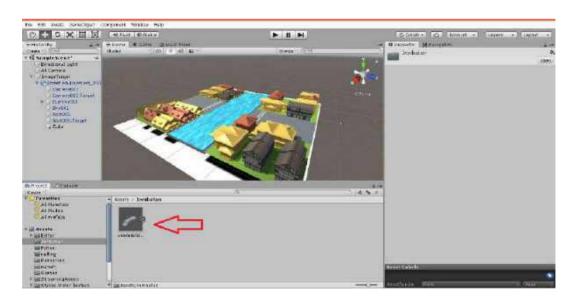
Gambar 5.71 3D Object Cube Diberi Texture

37. Buat 3 folder baru dengan beri nama Jembatan, Pohon dan Railing dengan cara yang telah dijelaskan sebelumnya.



Gambar 5.72 Membuat Folder Pohon, Jembatan dan Railing

38. Tambahkan 3D object jembatan ke dalam folder jembatan.



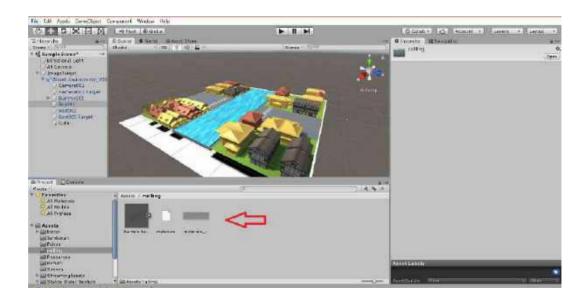
Gambar 5.73 3D Object Jembatan

39. Tambahkan 3D objek pohon pada folder pohon.



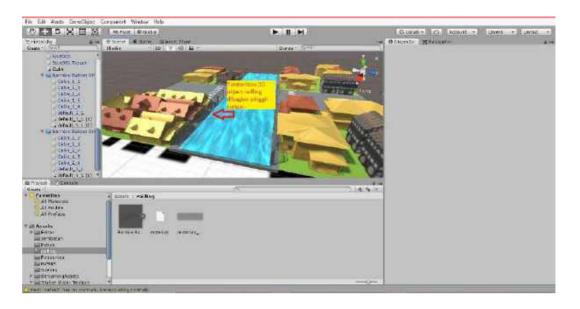
Gambar 5.74 3D Object Pohon

40. Tambahkan 3D objek Railing pada folder Railing



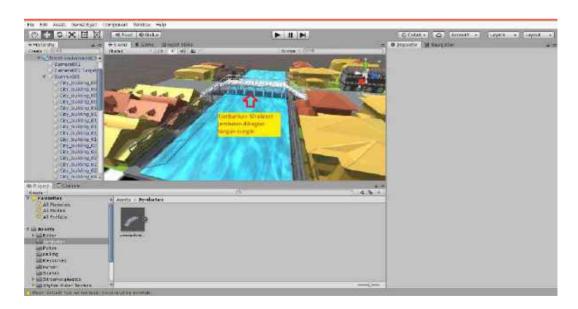
Gambar 5.75 3D Object Railing

41. Drag 3D object Railing pada 3D objek rumah kemudian atur posisi dan ukuranya. Posisikan Railing di bagian pinggir sungai



Gambar 5.76 Menambahkan 3D Object Railing

42. Drag 3D object Jembatan kepada 3D object rumah kemudian atur posisi dan ukurannya. Kemudian posisikan jembatan di bagian tengah sungai.



Gambar 5.77 Menambahkan 3D Object Jembatan

43. Drag 3D object pohon kepada 3D object rumah, lalu atur posisi dan ukuranya. Kemudian sesuaikan letaknya.



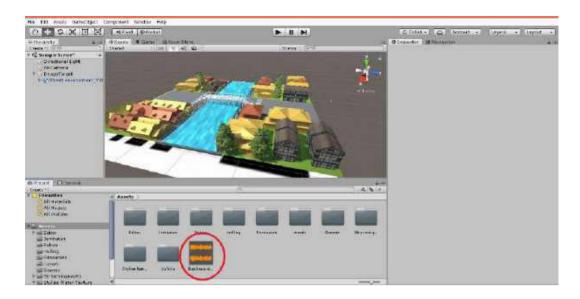
Gambar 5.78 Menambahkan 3D Object Pohon

44. Selanjutnya akan menambahkan sebuah audio untuk penjelasanya dengan cara klik kanan pada "scene" kemudian klik "audio" lalu pilih "Audio Source".



Gambar 5.79 Menambahkan Audio Source

45. Kemudian Drag audio yang akan digunakan ke dalam unity.



Gambar 5.80 Menambahkan Audio

46. Selanjutnya klik "image target" lalu klik kanan pada bagian "Default Trackable Event Handler (Script)" lalu pilih "Edit Script"



Gambar 5.81 Edit Script

47. kemudian tambahkan code berikut:

```
2 Copyright (c) 2017 PTC Inc. All Rights Reserved.
4 Copyright (c) 2010-2014 Qualcomm Connected Experiences, Inc.
5 All Rights Reserved.
6 Confidential and Proprietary - Protected under copyright and
      other laws.
  ______
9 using UnityEngine;
10 using Vuforia;
12 /// <summary>
 /// A custom handler that implements the ITrackableEventHandler
      interface.
14 ///
15 /// Changes made to this file could be overwritten when upgrading
       the Vuforia version.
  /// When implementing custom event handler behavior, consider
      inheriting from this class instead.
17 /// </summary>
  public class DefaultTrackableEventHandler: MonoBehaviour,
      ITrackableEventHandler
19
      public AudioSource aSource:
      public AudioClip aClip;
      #region PROTECTED_MEMBER_VARIABLES
      protected TrackableBehaviour mTrackableBehaviour;
      protected TrackableBehaviour. Status m_PreviousStatus;
25
      protected TrackableBehaviour. Status m_NewStatus;
26
27
28
      #endregion // PROTECTED_MEMBER_VARIABLES
30
      #region UNITY_MONOBEHAVIOUR_METHODS
31
      protected virtual void Start()
         mTrackableBehaviour = GetComponent<TrackableBehaviour >();
35
         if (mTrackableBehaviour)
36
             mTrackableBehaviour. RegisterTrackableEventHandler(
      this):
      }
39
      protected virtual void OnDestroy()
40
         if (mTrackableBehaviour)
42
             mTrackableBehaviour. UnregisterTrackableEventHandler(
43
      this);
      }
45
      #endregion // UNITY_MONOBEHAVIOUR_METHODS
```

```
#region PUBLIC_METHODS
49
      /// <summary>
50
               Implementation of the ITrackableEventHandler function
51
       called when the
       111
               tracking state changes.
52
      /// </summary>
53
      public void OnTrackableStateChanged(
54
           TrackableBehaviour. Status previous Status,
55
           TrackableBehaviour. Status newStatus)
      {
57
           m_PreviousStatus = previousStatus;
58
           m_NewStatus = newStatus:
59
60
           if (newStatus == TrackableBehaviour. Status.DETECTED ||
               newStatus == TrackableBehaviour.Status.TRACKED |
               newStatus == TrackableBehaviour. Status.
      EXTENDED_TRACKED)
           {
               Debug.Log("Trackable" + mTrackableBehaviour.
       TrackableName + " found");
               OnTrackingFound();
66
               if (mTrackableBehaviour.TrackableName == "arpka")
67
                   aSource. PlayOneShot(aClip);
               }
70
           }
           else if (previous Status == Trackable Behaviour, Status.
      TRACKED &&
                    newStatus == TrackableBehaviour.Status.NO_POSE)
           {
74
               Debug.Log("Trackable" + mTrackableBehaviour.
75
       TrackableName + " lost");
               OnTrackingLost();
76
                if (mTrackableBehaviour.TrackableName == "arpka")
78
                   aSource. Stop();
               }
80
           }
81
           else
82
           {
               // For combo of previous Status = UNKNOWN + new Status =
      UNKNOWN | NOT_FOUND
               // Vuforia is starting, but tracking has not been
85
       lost or found yet
               // Call OnTrackingLost() to hide the augmentations
               OnTrackingLost();
87
           }
88
      }
89
      #endregion // PUBLIC_METHODS
92
      #region PROTECTED_METHODS
93
94
      protected virtual void OnTrackingFound()
```

```
96
           var rendererComponents = GetComponentsInChildren<Renderer
       >(true);
           var colliderComponents = GetComponentsInChildren<Collider
98
           var canvasComponents = GetComponentsInChildren < Canvas > (
99
       true);
100
           // Enable rendering:
101
           foreach (var component in rendererComponents)
102
                component.enabled = true;
104
           // Enable colliders:
105
           foreach (var component in colliderComponents)
106
                component.enabled = true;
107
           // Enable canvas':
109
           foreach (var component in canvasComponents)
                component.enabled = true;
114
       protected virtual void OnTrackingLost()
116
           var rendererComponents = GetComponentsInChildren<Renderer
       >(true);
           var colliderComponents = GetComponentsInChildren<Collider
118
       >(true):
           var canvasComponents = GetComponentsInChildren < Canvas > (
       true);
120
           // Disable rendering:
           foreach (var component in rendererComponents)
                component.enabled = false;
124
           // Disable colliders:
           foreach (var component in colliderComponents)
126
                component.enabled = false;
128
           // Disable canvas':
129
130
           foreach (var component in canvas Components)
                component.enabled = false;
       }
134
       #endregion // PROTECTED_METHODS
135
```

Apabila sudah ditambahkan code tersebut maka akan secara otomatis adanya "A source" dan "A clip" pada "Default Trackable Event Handler (Script)"

▶ Materials Light Probes LO# 4 Bland Probes Reflection Probes None (Transform) a Anchor Override Cast Shadows On Receive Shadows 1 Motion Vectors Per Object Motion Lightmap Static To an able generative of hybridge for this Mack Banders, please analis the 'Lightmep fitatic' property. Dynamic Occluded □ ♥ 4. Image Target Mesh 11054 (Mesh Filter) Mesh ImageTargetMesh11054 Default Trackable Event Handler (Script) 国学品 Some. None (Audio Source) A Source 0 None (Audie Clip) A Clin a arpkaMaterial
Shoder Unlit/Texture U 7 0. Add Component

- 48. Drag audio sorce kepada "A Source dan drag audio yang akan digunakan kepada "A clip".
- 49. Untuk membuat sebuah 3D object seperti berputar maka buat script rotate. Buat Script C dengan cara klik kanan pada asset lalu klik "Create" kemudian pilih "C".

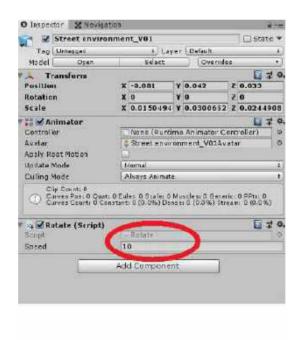


Gambar 5.82 Membuat Script C

50. Kemudian tambahkan kode berikut:

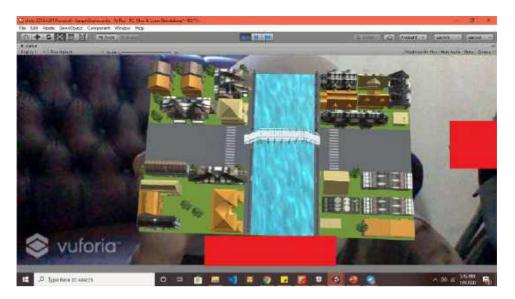
```
using System. Collections;
2 using System. Collections. Generic;
  using UnityEngine;
  public class Rotate: MonoBehaviour
6
      public int speed;
      // Start is called before the first frame update
      void Start()
0
      {
10
      }
      // Update is called once per frame
14
      void Update()
15
           transform. Rotate (new Vector3 (0, speed * Time. deltaTime, 0))
17
      }
18
19
```

51. Drag script rotate kepada 3D object rumah lalu sesuaikan speednya.



Gambar 5.83 Menambahkan Script dan Mengatur Speed Rotate

Apabila dirun maka akan muncul sebuah augmented reality yang telah dibuat seperti berikut :



Gambar 5.84 Augmented Reality PKA

Gambar diatas merupakan hasil akhir dari augmented reality. Apabila di run maka akan muncul sebuah 3D objek yang dapat berputar dengan disertai audio penjelasan prototipe PKA.

DAFTAR PUSTAKA

- 1. A. Juansyah, "Pembangunan aplikasi child tracker berbasis assisted-global positioning system (a-gps) dengan platform android," *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, vol. 1, no. 1, pp. 1–8, 2015.
- 2. A. Sumarudin, M. Yani, W. P. Putra, F. Amri, and P. Paskal, "Sistem pemantauan dan peringatan dini potensi banjir sungai cimanuk berbasis internet of things (iot)," in *Prosiding Industrial Research Workshop and National Seminar*, vol. 8, 2017, pp. 639–646.
- 3. T. L. Moe and P. Pathranarakul, "An integrated approach to natural disaster management," *Disaster Prevention and Management: An International Journal*, 2006.
- 4. K. K. Patel, S. M. Patel *et al.*, "Internet of things-iot: definition, characteristics, architecture, enabling technologies, application & future challenges," *International journal of engineering science and computing*, vol. 6, no. 5, 2016.
- 5. D. M. Nursyswanti and D. W. Astuti, "Perancangan aplikasi mobile remote control berbasis android pada robot lego mindstorms nxt 2.0," *Jurnal Teknologi Elektro*, vol. 6, no. 3.
- A. A. Soebroto, I. Cholissodin, R. C. Wihandika, M. T. Frestantiya, and Z. El Arief, "Prediksi tinggi muka air (tma) untuk deteksi dini bencana banjir menggunakan svrtviwpso," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 2, pp. 79–86, 2015.
- 7. W. Indianto, A. H. Kridalaksana *et al.*, "Perancangan sistem prototipe pendeteksi banjir peringatan dini menggunakan arduino dan php," 2018.

- 8. A. Akhiruddin, "Rancang bangun alat pendeteksi ketinggian air sungai sebagai peringatan dini banjir berbasis arduino nano," *JET (Journal of Electrical Technology)*, vol. 3, no. 3, pp. 174–179, 2018.
- 9. W. M. Shah, F. Arif, A. Shahrin, and A. Hassan, "The implementation of an iot-based flood alert system," *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS*, vol. 9, no. 11, pp. 620–623, 2018.
- 10. N. Ahmad, "Iot based flood alert management system," 2019.
- 11. D. Satria, S. Yana, R. Munadi, and S. Syahreza, "Prototype of google maps-based flood monitoring system using arduino and gsm module," *Int. Res. J. Eng. Technol*, vol. 4, no. 10, pp. 1044–1047, 2017.
- 12. R. Sulistyowati, H. A. Sujono, and A. K. Musthofa, "Sistem pendeteksi banjir berbasis sensor ultrasonik dan mikrokontroler dengan media komunikasi sms gateway," *Skripsi. Fakultas Teknologi Industri Institut Teknologi Adhi Tama Surabaya*, 2015.

TUTORIAL PEMBUATANPROTOTIPE PREDIKSI KETINGGIAN AIR (PKA) DAN AUGMENTED REALITY BERBASIS IOT VERSI 2

Buku ini membahas tentang bagaimana cara membuat sebuah prototipe prediksi ketinggian air (PKA) untuk mendeteksi banjir berbasis IoT versi 2 dengan mengimplementasikan metode regresi linear sederhana. Prototipe ini dibuat sebagai peringatan dini banjir , peringatan tersebut berupa sebuah notifikasi melalui telegram dan indikator led dan buzzer menyala. Dalam notifikasi tersebut terdapat jarak ketinggian air dan prediksi berapa lama waktu yang dibutuhkan air sampai pada perumahan (miniature).

Kemudian menjelaskan tentang bagaimana cara membuat sebuah augmented reality untuk ilustrasi prototipe pendeteksi banjir. Pada augmented reality ini dapat mengetahui penjelasan mengenai prototipe. Untuk memonitoring ketinggian air yaitu melalui aplikasi berbasis android. Pada buku ini juga membahas lengkap dari software maupun hardware yang dibutuhkan, prosses pembuatannya hingga contoh pemograman, sehingga mempermudah pembaca untuk membuat prototipe pendeteksi banjir tersebut.

Kreatif Industri Nusantara

Jl. Ligar Nyawang No. 2 Bandung 40191

Tel. 022 2045-8529

Email: awangga@kreatif.co.id

