

**RANCANG BANGUN SISTEM INFORMASI GEOGRAFIS
UNTUK PEMETAAN LOKASI TEMPAT IBADAH DI KOTA
MATARAM MENGGUNAKAN METODE EXTREME
PROGRAMMING**

Tugas Akhir
Untuk memenuhi sebagian persyaratan
mencapai derajat Sarjana S-1 Program Studi Teknik Informatika



Oleh:

RICKY SATRIA

F1D 016 077

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MATARAM
2021**

TUGAS AKHIR

RANCANG BANGUN SISTEM INFORMASI GEOGRAFIS UNTUK PEMETAAN LOKASI TEMPAT IBADAH DI KOTA MATARAM MENGGUNAKAN METODE EXTREME PROGRAMMING

Oleh :

RICKY SATRIA

F1D016077

Telah diperiksa oleh Tim Pembimbing:

1. Pembimbing Utama



Tanggal:
29/01/2021

Royana Afwani, ST., MT.
NIP. 198507072014042001

2. Pembimbing Pendamping



Tanggal:
29/01/2021

Ir. Sri Endang Anjarwani, M.Kom
NIP. 196604032006042001

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Teknik
Universitas Mataram



Prof. Dr. Eng. I Gede Pasek Suta Wijaya, ST., MT.
NIP. 197311302000031001

TUGAS AKHIR

RANCANG BANGUN SISTEM INFORMASI GEOGRAFIS UNTUK PEMETAAN LOKASI TEMPAT IBADAH DI KOTA MATARAM MENGGUNAKAN METODE EXTREME PROGRAMMING

Oleh :

RICKY SATRIA

F1D016077

Telah diujikan di depan penguji

Pada tanggal 27 Januari 2021

Dan dinyatakan telah memenuhi syarat mencapai derajat Sarjana S-1
Program Studi Teknik Informatika

Susunan Tim Penguji :

1. Penguji 1



Tanggal: 28/01/2021

Nadiyasari Agitha, S.Kom.,M.MT

NIP. 198608132018032001

2. Penguji 2



Tanggal: 28/01/2021

Moh. Ali Albar, ST., M.Eng.

NIP. 198311252015041002

3. Penguji 3



Tanggal: 29/01/2021

Ari Hernawan, S.Kom., M.Sc.

NIP. 199001142019031018

Mengetahui,
Dekan Fakultas Teknik
Universitas Mataram

Akmaluddin, ST., M.Sc Eng., Ph.D.

NIP. 196812311994121001

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan bahwa tugas akhir ini dengan judul “Rancang Bangun Sistem Informasi Geografis Untuk Pemetaaan Lokasi Tempat Ibadah di Kota Mataram menggunakan Metode *Extreme Programming*” sepenuhnya adalah karya sendiri. Tidak ada bagian di dalamnya yang merupakan plagiat dari karya orang lain dan saya tidak melakukan penjiplakan atau pengutipan dengan cara yang tidak sesuai dengan etika keilmuan yang berlaku. Atas pernyataan ini, saya siap menanggung resiko/sanksi yang dijatuuhkan kepada saya apabila kemudian ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya ini, atau ada klaim dari pihak lain terhadap keaslian karya saya ini.

Mataram, 28 Januari 2021

Yang membuat pernyataan,

RICKY SATRIA

PRAKATA

Assalamu'alaikum Warahmatullahi Wabarakatuh

Segala puji bagi Allah SWT yang telah memberikan rahmat dan karunianya kepada penulis, sehingga penulis dapat menyelesaikan skripsi dengan judul “Rancang Bangun Sistem Informasi Geografis Untuk Pemetaaan Lokasi Tempat Ibadah di Kota Mataram mengguanakan Metode *Extreme Programming*” tepat pada waktunya sebagai syarat untuk menyelesaikan Program Studi (S1) pada Program Sarjana Fakultas Teknik Jurusan Informatika Universitas Mataram. Shalawat dan salam senantiasa tercurahkan kepada Rasulullah SAW yang mengantarkan manusia dari zaman kegelapan ke zaman yang terang benderang.

Penulis tentu menyadari Tugas Akhir ini masih jauh dari kata sempurna dan masih terdapat kesalahan dan kekurangan di dalamnya dikarenakan terbatasnya pengalaman dan pengetahuan yang dimiliki penulis. Untuk itu, penulis mengharapkan segala bentuk kritik serta saran dari pembaca untuk Tugas Akhir ini, agar Tugas Akhir ini dapat menjadi lebih baik lagi.

Demikian yang bisa disampaikan, Terimakasih atas semua pihak yang telah membantu dalam penyelesaian Tugas Akhir ini baik moril maupun materil. Semoga Tugas Akhir ini dapat bermanfaat bagi para pembaca.

Wassalamual'aikum Warahmatullahi Wabaraktu

Mataram, 28 Januari 2021

UCAPAN TERIMA KASIH

Penulis menyadari bahwa selesainya Tugas Akhir ini tentunya bukan hanya dari usaha penulis saja. Tugas Akhir ini bisa selesai tepat waktu tentunya berkat dukungan dari semua pihak yang terlibat juga. Oleh karena itu, pada kesempatan ini penulis menghaturkan terima kasih yang sebesar-besarnya kepada:

1. Kedua Orang Tua dan Kakak-Kakak yang selalu memberikan dukungan kepada penulis dalam segala lini kehidupan selama perkuliahan.
2. Ibu Royana Afwani, ST., MT. selaku dosen pembimbing utama yang telah memberikan bimbingan dan arahan kepada penulis selama penyusunan Tugas Akhir sehingga dapat selesai dengan baik.
3. Ibu Ir. Sri Endang Anjarwani, M.Kom. selaku dosen pembimbing pendamping yang telah memberikan bimbingan dan arahan kepada penulis selama penyusunan Tugas Akhir sehingga dapat selesai dengan baik.
4. Pihak Kementerian Agama Provinsi NTB yang membantu penggerjaan Tugas Akhir ini.
5. Teman-teman Informatika Angkatan 2016 selaku teman diskusi dan teman pada masa perkuliahan.
6. Gilang Ramdhani Putra, Reza Rismawandi yang membantu dan menemani penggerjaan Tugas Akhir ini.
7. Semua pihak yang tidak dapat saya sebutkan namanya satu persatu yang telah memberikan dukungan baik moril maupun materil sehingga penulis dapat menyelesaikan Tugas Akhir dengan baik.

Semoga Allah SWT selalu memberikan rahmat dan hidayah-Nya dan memberikan imbalan yang setimpal atas bantuan yang diberikan kepada penulis

ABSTRAK

Rumah ibadah merupakan suatu tempat bertemunya para umat beragama untuk beribadah menurut ajaran agama atau kepercayaan mereka masing-masing. Mengetahui informasi lokasi tempat ibadah sangatlah penting untuk memenuhi kewajiban peribadatan. Di Kota Mataram, pencarian tempat ibadah terbilang masih cukup sulit karena tidak mengetahui posisi/jalan di sekitar mereka berada. Dari sisi kemenag juga masih belum memiliki system pengelolaan data tempat ibadah, saat ini pengelolaan data tempat ibadah masih dilakukan menggunakan Ms.Excel yang membuat pendataan menjadi tidak efektif dan membutuhkan waktu yang lama dalam melakukan pencarian terhadap data-data yang ada, media informasi dan pemetaan tempat ibadah di Kota Mataram juga belum ada. Pada penelitian Tugas Akhir ini, dirancang dan dibangun sebuah Sistem Informasi Geografis Pemetaan Lokasi Tempat Ibadah di Kota Mataram guna untuk memberikan informasi dan tata letak mengenai tempat ibadah yang ada di Kota Mataram karena saat dilakukan pencarian pencarian tempat ibadah berarti letak geografnnya yang lebih utama dibutuhkan. Dengan adanya sistem yang telah dibuat ini dapat memberikan manfaat bagi masyarakat umum dalam mengetahui informasi dan lokasi tempat ibadah yang ada di Kota Mataram, terbukti dengan adanya hasil kuisioner yang diberikan dapat dilihat pada Tabel 4.25 menunjukkan bahwa 40,33% sangat setuju, 47,85% setuju, dan 11,83% cukup setuju. Dan dari sisi kemenag juga dapat mempermudah dalam melakukan pendataan dari system yang membuat pengelolaan menjadi lebih efektif dan efisien, dapat dibuktikan melalui hasil kuesioner pada Tabel 4.24 menunjukkan bahwa 66,67% sangat setuju, 25% setuju, dan 8,33% adalah Cukup terkait dengan kemudahan dalam mengakses aplikasi, tampilan mudah di fahami, aplikasi berjalan dengan baik, membantu pihak kemenag dalam mengelola data tempat ibadah yang ada. Dalam penelitian Tugas Akhir ini digunakan metode *Extreme Programming* dalam proses pengembangan sistemnya, karena dirasa cocok sebab salah satu karakteristik XP itu sendiri melibatkan *client* selama proses pembangunan sistem sedangkan dalam kasus penelitian Tugas Akhir ini pihak Kemenag NTB sebagai *client* yang terlibat langsung dalam proses pembangunan sistem sehingga meminimalisir terjadinya kesalahan pada sistem dan sesuai dengan kebutuhan *client*, dapat dibuktikan pada hasil pengujian *black box* yang telah dilakukan pada Tabel 4.23 menunjukkan bahwa seluruh fungsi bekerja dengan baik pada system dan diterima sehingga dapat dikatakan system yang dibuat telah berjalan sesuai dengan kebutuhan *client*.

Kata Kunci : SIG, Kota Mataram, tempat ibadah, *Extreme Programming*

DAFTAR ISI

HALAMAN PENGESAHAN	ii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR	iv
PRAKATA	v
UCAPAN TERIMA KASIH	vi
ABSTRAK	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xii
DAFTAR KODE SUMBER	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah.....	4
1.4 Tujuan	4
1.5 Manfaat	5
2.2 Sistematika Penulisan	5
BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI	6
2.1 Tinjauan Pustaka.....	6
2.2 Dasar Teori.....	8
2.2.1 Google Maps API	8
2.2.2 GPS	9
2.2.3 Website	9
2.2.4 Sistem Informasi.....	9
2.2.5 Sistem Informasi Geografis.....	10
2.2.6 Framework <i>CodeIgniter</i>	10
2.2.7 PHP	11
2.2.8 XAMPP.....	11
2.2.9 Apache	11
2.2.10 Web server.....	11
2.2.11 PHPMyAdmin	11
2.2.12 MySQL	12
2.2.13 Visual Studio Code	12
2.2.14 Black Box Testing	12
2.2.15 Bootstrap	12

2.2.16	<i>Postman</i>	13
2.2.17	<i>Android</i>	13
2.2.18	Android Studio	13
2.2.19	<i>Java Development Kit (JDK)</i>	14
2.2.20	Metode pengembangan sistem <i>Extreme Programming</i>	14
2.2.21	<i>Personal Extreme Programming</i>	16
	BAB III METODE PENELITIAN	23
3.1	Alat dan Bahan	23
3.1.1	Alat	23
3.1.2	Bahan.....	23
3.2	Diagram alir penelitian	24
3.2.1	Studi Literatur.....	24
3.2.2	<i>Planning</i>	24
3.2.3	<i>Design</i>	32
3.2.4	<i>Coding</i>	60
3.2.5	<i>Testing</i>	63
3.2.6	<i>Release</i>	67
	BAB IV HASIL DAN PEMBAHASAN	68
4.1	Implementasi metode <i>Extreme Programming</i>	68
4.1.1	<i>Planning</i>	68
4.1.2	<i>Design</i>	72
4.1.3	<i>Coding</i>	73
4.1.4	<i>Testing</i>	107
4.2	Hasil evaluasi metode <i>extreme programming</i>	113
	BAB V PENUTUP	118
5.1	Kesimpulan.....	118
5.2	Saran	119
	DAFTAR PUSTAKA	120
	LAMPIRAN	124

DAFTAR GAMBAR

Gambar 2.1 Fishbone Diagram.....	8
Gambar 2.2 Siklus Extreme Programming.....	14
Gambar 3.1 Diagram Alir Penelitian.....	24
Gambar 3.2 Usecase Diagram	33
Gambar 3.3 Class Diagram Pada Web	37
Gambar 3.4 Class Diagram Pada Android.....	38
Gambar 3.5 Activity Diagram Proses Login Admin	39
Gambar 3.6 Activity Diagram Proses Menambah Data Tempat Ibadah	40
Gambar 3.7 Activity Diagram Proses Mengubah Data Tempat Ibadah	41
Gambar 3.8 Activity Diagram Proses Menghapus Data Tempat Ibadah.....	42
Gambar 3.9 Activity Diagram Proses Melihat Daftar Tempat Ibadah.....	42
Gambar 3.10 Activity Diagram Proses Mencari Lokasi Tempat Ibadah.....	43
Gambar 3.11 Activity Diagram Proses Melihat Jarak Lokasi Tempat Ibadah	44
Gambar 3.12 Sequence Diagram Proses Login Admin.....	44
Gambar 3. 13 Sequence Diagram Proses Menambah Data Tempat Ibadah	45
Gambar 3.14 Sequence Diagram Proses Mengubah Data Tempat Ibadah	45
Gambar 3. 15 Sequence Diagram Proses Menghapus Data Tempat Ibadah.....	46
Gambar 3.16 Sequence Diagram Proses Melihat Lokasi Tempat Ibadah	46
Gambar 3. 17 Sequence Diagram Proses Melihat Jarak/Rute Lokasi Tempat Ibadah ...	47
Gambar 3. 18 Sequence Diagram Proses Mencari Lokasi Tempat Ibadah.....	47
Gambar 3. 19 Entity Relationship Diagram Sistem	48
Gambar 3.20 Rancangan Tampilan Halaman Utama Pada Web	51
Gambar 3.21 Rancangan Tampilan Halaman Login	51
Gambar 3.22 Rancangan Tampilan Halaman Beranda Admin.....	52
Gambar 3.23 Rancangan Tampilan Halaman Kelola Data Tempat Ibadah.....	52
Gambar 3.24 Rancangan Tampilan Halaman Tambah Data Tempat Ibadah	53
Gambar 3.25 Rancangan Tampilan Halaman Edit Data Tempat Ibadah.....	53
Gambar 3.26 Rancangan Tampilan Hapus Data Tempat Ibadah	54
Gambar 3.27 Rancangan Tampilan Profil Admin.....	54
Gambar 3.28 Rancangan Tampilan Halaman Kelola Data Admin	55
Gambar 3.29 Rancangan Tampilan Halaman Tambah Data Admin	55
Gambar 3.30 Rancangan Tampilan Halaman Edit Data Admin	56
Gambar 3.31 Rancangan Tampilan Hapus Data Admin	56
Gambar 3.32 Rancangan Tampilan Halaman Peta Tempat Ibadah.....	57
Gambar 3.33 Rancangan Tampilan List Tempat Ibadah Android	57
Gambar 3.34 Rancangan Tampilan Informasi Detail Tempat Ibadah Android	58
Gambar 3.35 Rancangan Tampilan About Us Android	58
Gambar 3.36 Rancangan Tampilan Halaman Peta Tempat Ibadah Pada Basis Web	59
Gambar 3.37 Rancangan Tampilan Import Data Tempat Ibadah Pada Basis Web.....	59
Gambar 3.38 Arsitektur Sistem	60
Gambar 3.39 Alur Kerja TDD	61
Gambar 4.1 Implementasi Database	74
Gambar 4.2 Implementasi Tabel Tb_admin	74
Gambar 4.3 Implementasi Tabel Tb_bidang	75
Gambar 4.4 Implementasi Tabel Tb_jenis	75
Gambar 4.5 Implementasi Tabel Tb_ti.....	75

Gambar 4.6 Implementasi Tabel Tb_kecamatan	76
Gambar 4.7 Implementasi Tabel Tb_kabupaten	76
Gambar 4.8 Implementasi Class Controller.....	78
Gambar 4.9 Implementasi Class Model	78
Gambar 4.10 Implementasi Class View	79
Gambar 4.11 Implementasi Class Activity.....	79
Gambar 4.12 Adapter	81
Gambar 4.13 Implementasi Class Model	81
Gambar 4.14 Implementasi Class Service	81
Gambar 4.15 Layout (XML).....	82
Gambar 4.16 Tampilan Halaman Login.....	83
Gambar 4.17 Tampilan Logout.....	83
Gambar 4.18 Tampilan Halaman Profile Admin	85
Gambar 4.19 Tampilan Halaman Daftar Admin.....	86
Gambar 4.20 Tampilan Detail Data Admin.....	87
Gambar 4.21 Tampilan Tambah Data Admin.....	88
Gambar 4.22 Tampilan Edit Data Admin.....	89
Gambar 4.23 Tampilan Hapus Data Admin	90
Gambar 4. 24 Tampilan Halaman Tambah Data Tempat Ibadah	91
Gambar 4.25 Tampilan Import Data Tempat Ibadah	92
Gambar 4.26 Tampilan Import Data Tempat Ibadah	93
Gambar 4.27 Tampilan Export Data Tempat Ibadah	94
Gambar 4.28 Tampilan Daftar Tempat Ibadah Basis Web Sisi Admin	95
Gambar 4.29 Tampilan Daftar Tempat Ibadah Basis Web Sisi Masyarakat	95
Gambar 4.30 Tampilan Detail Data Tempat Ibadah Basis Web	96
Gambar 4.31 Tampilan Daftar Tempat Ibadah Basis Android	96
Gambar 4.32 Tampilan Detail Tempat Ibadah Basis Android	97
Gambar 4.33 Tampilan Halaman Beranda Pada Sisi Admin.....	98
Gambar 4.34 Tampilan Halaman Edit Data Tempat Ibadah.....	99
Gambar 4.35 Tampilan Hapus Data Tempat Ibadah.....	100
Gambar 4.36 Tampilan Peta Lokasi Tempat Ibadah Basis Android	101
Gambar 4.37 Tampilan Peta Rute Meunuju Lokasi Tempat Ibadah Basis Android	103
Gambar 4.38 Tampilan Halaman Peta Lokasi Tempat Ibadah Basis Web	104
Gambar 4.39 Tampilan Halaman Peta Lokasi Tempat Ibadah Basis Web	105
Gambar 4.40 Grafik Perentase Rata-Rata Responden Pengguna Aplikasi SITABAH	113

DAFTAR TABEL

Tabel 2.1 Twelve Core Practices Compared	16
Tabel 3.1 User Story Cards.....	27
Tabel 3.2 Pembagian Iterasi Story	32
Tabel 3.3 Penjelasan CRC-Card	34
Tabel 3.4 Tabel Admin.....	49
Tabel 3.5 Tabel Bidang	49
Tabel 3. 6 Tabel Tempat Ibadah	49
Tabel 3.7 Tabel Kecamatan	50
Tabel 3.8 Tabel Jenis Tempat Ibadah.....	50
Tabel 3.9 Tabel Kabupaten.....	50
Tabel 3. 10 Rencana Pengujian Pada Login	64
Tabel 3. 11 Rencana Pengujian Tambah Data Admin	65
Tabel 3. 12 Skor Penilaian.....	66
Tabel 4.1 Penambahan User Stories.....	69
Tabel 4.2 Hasil Perencanaan Iterasi	71
Tabel 4.3 Hasil Design CRC-Card Dan Spike Solution Prototype	73
Tabel 4.4 Pengujian Login System	84
Tabel 4.5 Pengujian Logout System	85
Tabel 4.6 Pengujian Halaman Profile Admin	86
Tabel 4.7 Pengujian Menampilkan Data Admin.....	87
Tabel 4.8 Pengujian Tambah Data Admin	88
Tabel 4.9 Pengujian Mengubah Data Admin.....	89
Tabel 4.10 Pengujian Mengahpus Data Admin	91
Tabel 4.11 Pengujian Tambah Data Tempat Ibadah	92
Tabel 4. 12 Pengujian Import Data Tempat Ibadah	93
Tabel 4.13 Pengujian Unduh Template Format Excel Import Data.....	94
Tabel 4.14 Pengujian Export Data Excel Tempat Ibadah.....	94
Tabel 4.15 Pengujian Menampilkan Data Tempat Ibadah	97
Tabel 4.16 Pengujian Pada Halaman Beranda Sisi Admin.....	99
Tabel 4.17 Pengujian Ubah Data Tempat Ibadah	100
Tabel 4. 18 Pengujian Mengahpus Data Tempat Ibadah.....	101
Tabel 4. 19 Pengujian Peta Lokasi Tempat Ibadah Basis Android	102
Tabel 4.20 Pengujian Peta Rute Meunuju Lokasi Tempat Ibadah Basis Android	103
Tabel 4.21 Pengujian Peta Lokasi Tempat Ibadah Basis Web	104
Tabel 4.22 Pengujian Peta Rute Menuju Lokasi Tempat Ibadah Basis Web	105
Tabel 4.23 Hasil Pengujian Black Box	108
Tabel 4.24 Hasil Pengujian Kuisioner Dengan Pegawai KEMENAG NTB	111
Tabel 4.25 Hasil Pengujian Kuisioner Dengan Masyarakat	112
Tabel 4.26 Hasil Analisa Nilai Metode Extreme Programming	113
Tabel 4.27 rincian waktu penggerjaan	115

DAFTAR KODE SUMBER

Kode Sumber 3.1 Contoh Source Code Point 1.....	62
Kode Sumber 3. 2 Yang Ada Di File Helper	63
Kode Sumber 3. 3 Kode Yang Dieksekusi	63
Kode Sumber 4.1 Implementasi Fungsi Get Pada Controller	77
Kode Sumber 4.2 Implementasi Fungsi Get Pada Model	77
Kode Sumber 4.3 Sebelum Di Refactor	106
Kode Sumber 4.4 Setelah Di Refactor	107

BAB I

PENDAHULUAN

1.1 Latar Belakang

Agama merupakan suatu sistem yang mampu mengatur tata keimanan dan kepercayaan serta ibadah pada Tuhan Yang Maha Kuasa disertai dengan tata kaidah yang berkaitan langsung dengan ciri pergaulan manusia dengan manusia lainnya ataupun manusia dengan lingkungan sekitarnya[1]. Tempat ibadah merupakan suatu tempat bertemunya para umat beragama untuk beribadah menurut ajaran agama atau kepercayaan mereka masing-masing. Tempat peribadatan setiap harinya atau waktu tertentu selalu banyak dikunjungi oleh umatnya. Kunjungan ke tempat ibadah tersebut akan meningkat ketika ada kegiatan khusus atau perayaan hari besar keagamaan sehingga banyak interaksi terjadi[2]. Sarana peribadatan seperti rumah ibadah merupakan bangunan yang memiliki ciri – ciri tertentu yang khusus dipergunakan untuk beribadat bagi para pemeluk agama masing – masing[3].

Berdasarkan kuesioner yang telah dilakukan terhadap 61 responden, 55,7% data responden menyatakan bahwa fungsi rumah ibadah yaitu digunakan bagi seseorang yang hendak ingin meningkatkan hubungan spiritual dengan Tuhan-Nya dan 34,4% data responden menyatakan rumah ibadah digunakan bagi orang muslim yakni untuk mendapatkan lebih banyak pahala daripada sholat dirumah. 62,3% data responden menyatakan cukup sulit dalam melakukan pencarian tempat ibadah di Kota Mataram, yang disebabkan karena 68,8% data responden menunjukkan bahwa mereka tidak mengetahui posisi/jalan di sekitar mereka berada. Adapun berbagai macam solusi yang dilakukan masyarakat saat ini dalam mengatasi masalah tersebut yaitu 34,4% data responden memilih bertanya kepada penduduk sekitar, 32.8% memilih untuk menelusuri jalanan dan 29.5% memilih untuk menggunakan *google maps*. 75,4% data responden berpendapat bahwa mengetahui informasi lokasi tempat ibadah sangatlah penting untuk memenuhi kewajiban peribadatan. Untuk mengetahui penting atau tidaknya dibangun Sistem Informasi Geografis untuk Pemetaan Lokasi Tempat ibadah di Kota Mataram, didapatkan hasil data responden 54.1% berpendapat sangat penting 44.3%

berpendapat penting, dan 1.6% berpendapat kurang penting. Dilihat dari hasil kuesioner tersebut dapat dikatakan bahwa masyarakat setuju untuk dibangunnya sistem informasi pemetaan lokasi tempat ibadah di Kota Mataram tersebut. Dan 75,4% data responden memilih untuk dibuatkan sistem berbasis *android*, karena lebih efektif dan efisien dalam penggunaannya.

Berdasarkan hasil wawancara yang telah dilakukan terhadap salah satu *staff* sub. bagian Inmas Kementerian Agama provinsi NTB yaitu bapak H. Toni, menyebutkan bahwa dalam pengelolaan data informasi mengenai tempat ibadah yang ada saat ini masih menggunakan cara konvensional yaitu menggunakan Ms. Excel, sehingga pendataan sering kali tidak efektif serta membutuhkan waktu yang cukup lama ketika melakukan pencarian terhadap data-data yang ada. Berdasarkan observasi yang penulis lakukan juga tidak adanya media informasi dan pemetaan tempat ibadah di Kota Mataram, sistem informasi geografis untuk lokasi tempat ibadah di Kota Mataram juga masih belum ada.

Sistem Informasi Geografis (SIG) merupakan suatu sistem informasi yang dirancang untuk bekerja dengan data yang bereferensi spasial atau berkordinat geografis. SIG berarti sifat dari informasinya yaitu mengenai obyek-obyek atau hal-hal yang ada atau terjadi atau diperkirakan terjadi di muka bumi, tepatnya disuatu lokasi entah itu wilayah yang luas atau kecil kecil, bisa rumah, kampung, desa, kota, hutan, sawah, negara, bahkan dunia, tergantung dari maksud. Geografis atau informasi geografis bisa juga ditandai dengan data-data seperti koordinat [4]. Dalam penelitian Tugas Akhir ini, SIG sangat cocok untuk mengatasi masalah yang telah dipaparkan sebelumnya, guna untuk memberikan informasi dan tata letak mengenai tempat ibadah yang ada di Kota Mataram karena saat dilakukan pencarian tempat ibadah berarti letak geografisnya yang lebih utama dibutuhkan.

Menurut salah satu *staff* sub. bagian Inmas Kementerian Agama provinsi NTB yaitu Hj. Diah Purnawati, sistem yang sudah ada di kantor Kementerian Agama provinsi NTB saat ini yaitu sistem informasi berbasis *website* yang dapat diakses melalui link berikut: <https://ntb.kemenag.go.id>. Berdasarkan hal tersebut sistem informasi yang akan penulis buat dapat menjadi salah satu fitur atau menu dalam sistem informasi yang sudah ada, dimana guna dari system yang akan dibuat

yaitu untuk memberikan informasi kepada masyarakat umum dan dapat dikelola oleh pihak Kemenag prov. NTB mengenai data tempat ibadah yang ada.

Dalam penelitian Tugas Akhir ini digunakan metode *Extreme Programming* atau XP. Pada XP terdapat model Personal *Extreme Programming* atau PXP yang proses pengembangannya dapat dilakukan oleh *programmer* tunggal, dimana dalam kasus penelitian Tugas Akhir ini dikerjakan oleh penulis sendiri. Dalam XP menekankan konsep lebih berinteraksi dengan *client* dalam proses pengembangan sistem dan berfokus pada sistem yang dikembangkan, sehingga meminimalisir terjadinya kesalahan pada sistem dan sesuai dengan kebutuhan *client*, dalam kasus penelitian Tugas Akhir ini pihak Kementerian Agama prov. NTB sebagai *client*.

Berdasarkan hal tersebut, maka penulis akan merancang dan membangun suatu sistem informasi geografis yang berjudul “Rancang Bangun Sistem Informasi Geografis Untuk Pemetaan Lokasi Tempat Ibadah Di Kota Mataram Menggunakan Metode *Extreme Programming*”. Diharapkan setelah terciptanya sistem ini nantinya dapat membantu pihak Kemenag prov. NTB dalam mengelola data informasi tempat ibadah yang ada serta membantu masyarakat umum dalam mengetahui informasi dan lokasi mengenai tempat ibadah yang ada di Kota Mataram dengan mudah.

1.2 Rumusan Masalah

Berdasarkan uraian latar belakang yang ada maka dirumuskan masalah dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana melakukan rancang bangun sistem informasi berbasis website dan android yang dapat menampilkan infomasi dan lokasi tempat ibadah di Kota Mataram dalam bentuk peta digital yang dapat diakses secara online dan memberikan rute/jalur perjalanan dari posisi user berada menuju tempat ibadah yang diinginkan?
2. Bagaimana melakukan rancang bangun sistem informasi berbasis web yang dapat digunakan sebagai media informasi dan mempermudah Kemenag prov. NTB dalam melakukan pengelolaan terhadap data tempat ibadah yang ada?
3. Bagaimana mengimplementasikan metode Extreme Programming dalam merancang dan membangun “Sistem Informasi Geografis Untuk Pemetaan Lokasi Tempat Ibadah Di Kota Mataram”?

1.3 Batasan Masalah

Penelitian ini memiliki batasan-batasan masalah untuk memberikan lingkup penelitian agar lebih terfokus ketika penggerjaan. Adapun batasan masalah yang diberikan adalah sebagai berikut:

1. Sistem informasi tempat ibadah ini menyediakan informasi berbasis web dan android yang hanya dapat diakses secara online.
2. Pada sistem informasi berbasis web digunakan oleh masyarakat umum untuk mengetahui informasi mengenai tempat ibadah dan digunakan oleh pihak kemenag prov. NTB untuk mengelola data tempat ibadah yang sudah ada.
3. Pada sistem informasi berbasis android digunakan oleh masyarakat umum untuk mengetahui informasi dan lokasi mengenai tempat ibadah dalam bentuk peta digital dengan rute/jalur dari posisi user berada menuju ke tempat peribadatan yang diinginkan.
4. Sistem informasi ini diperuntukan hanya lokasi tempat ibadah yang ada di Kota Mataram.
5. Pembuatan peta tempat ibadah ini menggunakan peta dari *Google maps*.
6. Sistem informasi ini hanya menampilkan lokasi tempat ibadah, tidak dengan yang lain.
7. Pengguna tidak dapat menambahkan lokasi pada sistem.

1.4 Tujuan

Adapun tujuan dari penulisan tugas akhir ini adalah sebagai berikut:

1. Membantu masyarakat dalam memberikan informasi mengenai tempat ibadah di Kota Mataram dan memberikan jalur perjalanan menuju tempat ibadah yang dibutuhkan.
2. Membantu mempermudah Kementerian Agama Provinsi NTB dalam melakukan pengelolaan terhadap data tempat ibadah yang ada.
3. Mengimplementasikan metode *Extreme Programming* dalam proses perancangan dan pembangunan “Sistem Informasi Geografis Untuk Pemetaan Lokasi Tempat Ibadah Di Kota Mataram”

1.5 Manfaat

Adapun manfaat dari penulisan tugas akhir ini adalah sebagai berikut:

- 1 Memudahkan Kemenag Prov. NTB dalam mengelola data tempat ibadah yang ada.
- 2 Memudahkan masyarakat dalam mencari tau informasi dan lokasi mengenai tempat ibadah serta memudahkan dalam menggapai tempat ibadah yang ada di Kota Mataram.

2.2 Sistematika Penulisan

Sistematika penulisan tugas akhir adalah sebagai berikut:

- 1 Bab 1 Pendahuluan
Bab ini mengemukakan latar belakang, rumusan masalah, batasan masalah, tujuan, serta sistematika penulisan yang digunakan untuk menyusun laporan tugas akhir.
- 2 Bab 2 Tinjauan Pustaka dan Dasar Teori
Bab ini berisi dasar teori yang digunakan dalam melakukan analisis, perencangan dan implementasi tugas akhir yang dilakukan pada bab-bab selanjutnya.
- 3 Bab 3 Metodologi Perancangan
Bab ini berisi tentang pelaksanaan, alat, bahan, jalannya perencanaan, dan hasil yang diharapkan.
- 4 Bab 4 Hasil dan Pembahasan
Bab ini memuat tentang analisa dan pembahasan hasil penelitian yang telah dilakukan.
- 5 Bab 5 Penutup
Bab ini berisi kesimpulan dan saran-saran berdasarkan hasil pembahasan yang diproleh.

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Pembuatan sistem informasi mengenai tempat ibadah telah banyak dilakukan oleh peneliti terdahulu. Adapun sistem informasi yang telah dibuat oleh peneliti terdahulu diantaranya adalah sebagai berikut:

Penelitian yang berjudul “Implementasi *Geographic Information System (GIS) Penentuan Tampat Ibadah Masjid Di Kecamatan Kempas Kabupaten Indragiri Hilir Provinsi Riau*” yang bertujuan untuk dapat membantu masyarakat terutama yang berasal dari luar kota Kecamatan Kempas maupun dari daerah-daerah lain untuk menemukan suatu tempat ibadah di daerah Kecamatan Kempas. Aplikasi WebGis ini dibuat dengan menggunakan GIS, *Dreamweaver CS 5*, *MySql*, dan *Web Browser* sebagai media untuk menyajikan informasinya. Dimana informasi yang disajikan hanya informasi mengenai lokasi atau letak tempat ibadah masjid saja yang ada di Kecamatan Kempas. Untuk metode pengembangan sistem yang digunakan yaitu SDLC (*System Development Life Cycle*). SDLC merupakan metodologi klasik yang digunakan oleh analis sistem dan programmer dalam mengembangkan, memilihara, dan menggunakan sistem informasi [8].

Pada penelitian yang berjudul “Sistem Informasi Geografis Persebaran Tempat Ibadah Dan Kapasitasnya Berdasarkan Jumlah Penduduk Di Kota Banjarmasin” merupakan suatu pembangunan sistem informasi geografis tempat ibadah berbasis web dengan menggunakan data-data yang didapatkan dari Kementerian Agama Kota Banjarmasin, serta peta administrasi Kota Banjarmasin. Aplikasi WebGis ini dibangun menggunakan Quantum GIS dan Google Maps. Dengan adanya aplikasi ini akan dapat dijadikan pertimbangan untuk pengambilan keputusan dalam menentukan layak atau tidak untuk mendirikan tempat ibadah baru di Kota Banjarmasin [9].

Dalam penelitian yang berjudul “Sistem Informasi Masjid Di Surakarta” dibangun untuk menampilkan informasi mengenai masjid yang ada di Surakarta. Dimana tujuannya tidak hanya untuk memberikan kemudahan kepada masyarakat untuk mencari informasi seputar masjid di Surakarta, tetapi juga digunakan pihak IT di Kementerian Agama Kantor Kota Surakarta untuk memantau segala sesuatu

yang ada di dalam *website* serta dapat menyebarluaskan informasi menjadi lebih efektif dan efisien. Aplikasi berbasis *website* ini dibangun menggunakan bahasa pemrograman PHP dan MySQL untuk penyimpanan *database*-nya, serta untuk *user interface*-nya dibangun menggunakan bahasa pemrograman HTML dan CSS. *Website* pada penelitian ini dilengkapi fitur *searching*, *maps*, dan *newsfeed* [10].

Pada penelitian yang berjudul “Membangun Aplikasi WebGis Untuk Tempat Ibadah Di Provinsi Lampung” yang membahas tentang pembuatan aplikasi *WebGis* yang dapat digunakan untuk mencari tempat ibadah yang ada di Lampung secara online. Aplikasi pada penelitian ini dibangun menggunakan bahasa pemrograman HTML dan PHP dengan *database* MySql. Untuk metode pengembangan sistem yang digunakan yaitu *Extreme Programming*. *Extreme Programming* adalah sebuah pendekatan atau model pengembangan perangkat lunak yang mencoba menyederhanakan berbagai tahapan dalam proses pengembangan tersebut sehingga menjadi lebih adaptif dan fleksibel [5].

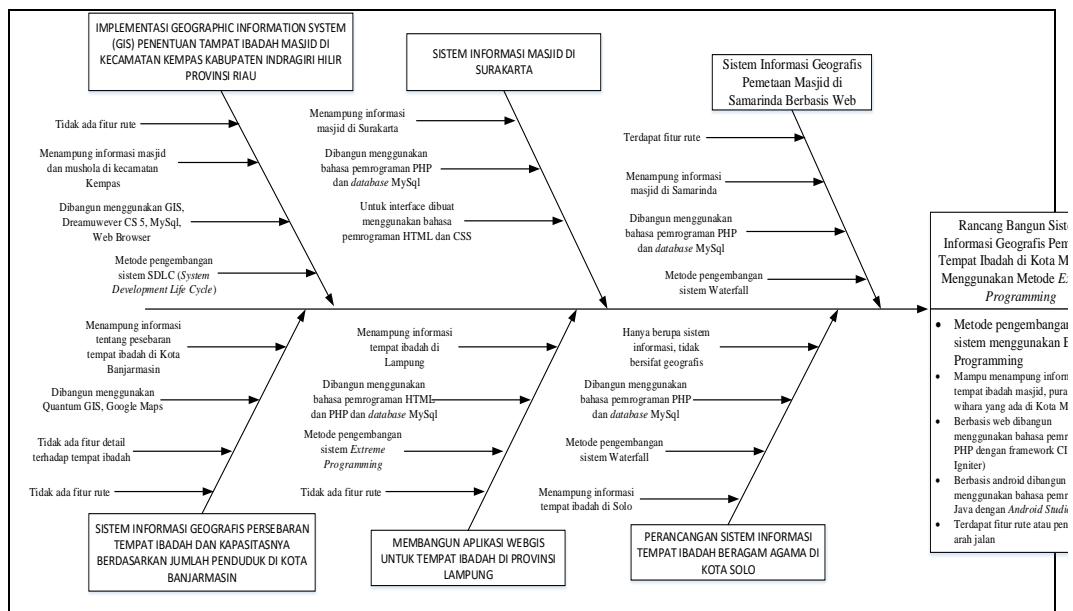
Dalam penelitian yang berjudul “Perancangan Sistem Informasi Tempat Ibadah Beragam Agama Di Kota Solo” yang membahas tentang pembuatan sistem informasi berbasis *website* yang berguna untuk menampilkan dan memuat informasi tempat ibadah beragam agama di Kota Solo, yaitu: Masjid, Gereja, Pura, Vihara, dan Kelenteng. Sistem informasi berbasis website ini Dibangun menggunakan bahasa pemrograman PHP dan *database* MySql. Untuk metode pengembangan sistem yang digunakan yaitu metode *waterfall* [5].

Penelitian yang berjudul “Sistem Informasi Geografis Pemetaan Masjid Di Samarinda Berbasis Web” merupakan suatu pembangunan sistem informasi geografis pemetaan masjid berbasis web yang bertujuan untuk memberikan informasi kepada masyarakat yang ada di Samarinda tentang tempat ibadah masjid beserta kegiatan yang ada didalamnya dalam visualisasi peta menggunakan Google Maps API dan terdapat fitur rute yang menggambarkan rute arah jalan dari posisi *user* berada menuju lokasi masjid yang diinginkan. Aplikasi pada penelitian ini dibuat menggunakan bahasa pemrograman PHP dan untuk *database*-nya menggunakan MySQL. Untuk metode pengembangan sistem yang digunakan yaitu *waterfall*. *Waterfall* memiliki beberapa tahapan dalam pengembangan sistem yang

dilakukan, yakni: tahap perencanaan, tahap analisis, tahap perancangan, tahap penerapan, tahap pengujian dan tahap pemeliharaan[11].

Berdasarkan tinjauan pustaka yang telah dipaparkan, penulis akan merancang dan membangun Sistem Informasi Geografis Untuk Pemetaan Lokasi Tempat Ibadah di Kota Mataram menggunakan metode *Extreme Programming*. Studi kasus ini diambil karena belum ada penelitian tentang SIG pemetaan lokasi tempat ibadah di Kota Mataram sehingga penulis mengusulkan tugas akhir dengan judul tersebut. Tujuan dari penelitian ini yaitu untuk mempermudah masyarakat umum dalam mencari mencari tau informasi dan lokasi tempat ibadah yang diinginkan serta memudahkan pihak Kemenag prov. NTB dalam mengelola data tempat ibadah yang ada.

Pada Gambar 2.1 berikut merupakan *fishbone* diagram dari tinjauan 6 pustaka dan usulan sistem yang akan dikembangkan:



Gambar 2.1 *Fishbone* Diagram

2.2 Dasar Teori

Berikut merupakan teori-teori dasar atau umum yang digunakan dalam penelitian tugas akhir ini.

2.2.1 Google Maps API

Svennerberg telah mencatat bahwa, Google Maps API adalah API yang paling popular di internet. Hasil pencatatan pada bulan Mei tahun 2010 menyatakan bahwa 43% mashup (aplikasi dan situs web yang menggabungkan dua atau lebih

sumber data) menggunakan Google Maps API. Beberapa tujuannya dari penggunaan Google aMaps API adalah melihat lokasi, serta mencari alamat dan lain sebagainya [12].

Google maps API adalah fungsi-fungsi pemrograman yang disediakan oleh Google maps agar Google maps bisa di integrasikan kedalam Web atau aplikasi yang sedang buat. Contoh sederhanya misalkan penulis ingin membuat Sistem informasi Geografis kampus di jogja, dengan memanfaatkan Google Maps API penulis bisa membuat GIS tanpa perlu memikirkan Peta Jogja, penulis tinggal pake Google maps dan memanggil fungsi fungsi yang dibutuhkan seperti menampilkan peta, menempatkan *marker* dan lain lain sabagainya [13].

2.2.2 GPS

Global Positioning System (GPS) berfungsi untuk mengetahui letak tempat yang akan dituju dan mengetahui dimana pengguna berada, memberikan informasi yang tepat dan akurat mengenai posisi, kecepatan, arah, dan waktu dengan bantuan sinyal satelit [12].

2.2.3 Website

Website merupakan sebuah kumpulan halaman-halaman *web* beserta *file-file* pendukungnya, seperti *file* gambar, *video*, dan *file* digital lainnya yang disimpan pada sebuah *web server* yang umumnya dapat diakses melalui internet. Atau dengan kata lain, *website* adalah sekumpulan *folder* dan *file* yang mengandung banyak perintah dan fungsi fungsi tertentu, seperti fungsi tampilan, fungsi menangani penyimpanan data, dsb[14].

2.2.4 Sistem Informasi

Sistem merupakan kumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan dalam usaha mencapai suatu tujuan. Informasi adalah hasil pemrosesan data yang diperoleh dari setiap elemen sistem tersebut menjadi bentuk yang mudah dipahami dan merupakan pengetahuan yang relevan dan dibutuhkan oleh orang untuk menambah pemahamannya terhadap fakta-fakta yang ada. Sistem Informasi juga dapat didefinisikan sebagai perangkat elemen yang bekerja mengumpulkan, memproses, menyimpan, dan menyebarkan informasi untuk mendukung pengambilan keputusan, koordinasi, pengawasan, analisis, dan visualisasi dalam organisasi [15].

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan [16].

2.2.5 Sistem Informasi Geografis

Sistem Informasi Geografis merupakan suatu sistem informasi yang dirancang untuk bekerja dengan data yang berasal dari spasial atau berkoordinat geografis atau dengan kata lain SIG adalah suatu sistem basis data dengan kemampuan khusus untuk menangani data yang berasal dari keruangan (spasial) bersamaan dengan seperangkat operasi kerja.

Sistem informasi geografis memiliki tiga unsur yakni sebagai berikut.

- 1 Sistem, dapat diartikan sebagai berbagai hal yang saling berkaitan atau saling mempengaruhi dalam mengerjakan proses untuk satu tujuan dalam SIG.
- 2 Informasi, sesuai dengan karakter SIG, informasi disini tentu saja adalah informasi tentang bumi (geografis) dengan apa yang ada di bumi. SIG merupakan media untuk menggambarkan apa yang ada di bumi dengan segala yang ada sesuai dengan tempat atau lokasi dia berada. Informasi inilah yang menjadi obyek kerja SIG.
- 3 Geografis, geografis dalam SIG berarti sifat dari informasinya yaitu mengenai obyek-obyek atau hal-hal yang ada atau terjadi atau diperkirakan terjadi di muka bumi, tepatnya disuatu lokasi entah itu wilayah yang luas atau kecil kecil, bisa rumah, kampung, desa, kota, hutan, sawah, negara, bahkan dunia, tergantung dari maksud. Geografis atau informasi geografis bisa juga ditandai dengan data-data seperti koordinat [4].

2.2.6 Framework CodeIgniter

Codeigniter adalah sebuah *framework* PHP yang dibuat berdasarkan *design pattern model view controller* atau biasa disingkat MVC. *Design Pattern* adalah kumpulan penjelasan mengenai metode-metode bagaimana cara menyelesaikan suatu masalah yang umum ditemui dalam proses perancangan perangkat lunak (*Software Design*) [14].

2.2.7 PHP

PHP Merupakan singkatan *recursive* dari PHP : *Hypertext Preprocessor*. Pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1994 [17]. PHP merupakan bahasa pemograman untuk membuat web. PHP dapat digunakan untuk membuat halaman *web* yang dinamis. PHP dapat digunakan pada sistem operasi Windows, Mac OS, Linux, dan sistem operasi yang lainnya.

2.2.8 XAMPP

XAMPP adalah perangkat yang menggabungkan tiga aplikasi kedalam satu paket yaitu Apache, MySQL,dan PhpMyAdmin, Dengan Xampp pekerjaan menjadi sangat dimudahkan karena dapat menginstalasi dan mengkonfigurasi ketiga aplikasi tersebut dengan sekaligus dan otomatis. Xampp telah mengalami perkembangan dari waktu ke waktu.versi yang terbaru adalah revisi dari yang terdahulu, sehingga lebih baik dan lebih lengkap. Aplikasi utama dalam paket Xampp yakni terdiri atas *web server* Apache, MySQL, PHP, dan PhpMyAdmin.

2.2.9 Apache

Apache adalah sebuah *web server open source*, jadi semua orang dapat menggunakannya secara gratis, bahkan anda bisa mengedit kode programnya. fungsi utama dari *apache* yakni menghasilkan halaman *web* yang benar sesuai dengan yang dibuat oleh seorang *web programmer*, dengan menggunakan kode PHP [18].

2.2.10 Web server

Web Server merupakan sebuah perangkat lunak dalam *server* yang berfungsi menerima permintaan (*request*) berupa halaman web melalui HTTP atau HTTPS dari *klien* yang dikenal dengan *browser web* dan mengirimkan kembali (*response*) hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen HTML [17].

2.2.11 PHPMyAdmin

PHPMyAdmin merupakan *front-end* MySQL berbasis *web*. PHPMyAdmin dibuat dengan menggunakan PHP. Saat ini, PHPMyAdmin banyak digunakan dalam hampir semua penyedia hosting yang ada di internet. PHPMyAdmin mendukung berbagai fitur administrasi MySQL termasuk manipulasi database, tabel, index dan juga dapat mengeksport data ke dalam berbagai format data [17].

2.2.12 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang *multithread, multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia[17]. MySQL adalah sistem manajemen *database* yang sering digunakan bersama PHP. PHP juga mendukung pada Microsoft Access, Database Oracle, dBase, dan sistem manajemen database lainnya. SQL (*Structured Query Language*) adalah bahasa terstruktur yang digunakan secara khusus untuk mengolah database, dan MySQL merupakan sebuah sistem manajemen *database* [18].

2.2.13 Visual Studio Code

Visual Studio Code menyediakan pengembang dengan pilihan baru alat pengembang yang menggabungkan kesederhanaan dan pengalaman ramping dari *code editor* yang terbaik dari apa yang pengembang butuhkan untuk siklus kode-builddebug inti mereka. *Visual Studio Code* adalah editor kode pertama, dan alat pengembangan lintas *platform* pertama - mendukung OS X, Linux, dan Windows [19].

2.2.14 Black Box Testing

Black Box adalah teknik pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, penguji dapat mendefinisikan kumpulan kondisi masukan dan melakukan pengetesan pada spesifikasi fungsional. Tujuan *Black Box Testing* untuk menunjukkan fungsi perangkat lunak tentang cara beroperasinya, apakah pemasukan data keluaran telah berjalan sebagaimana yang telah diharapkan dan apakah informasi yang disimpan serta eksternal selalu dijaga kemutakhirannya. *Black Box Testing* bukanlah solusi alternatif dari *White Box Testing* tapi lebih sebagai pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White Box Testing*. *Black Box Testing* mengidentifikasi jenis kesalahan dalam beberapa kategori antara lain fungsi yang tidak benar atau tidak ada, kesalahan antarmuka (*interface errors*), kesalahan pada struktur data dan akses basis data, kesalahan performansi (*performance errors*) dan kesalahan inisialisasi maupun terminasi[20].

2.2.15 Bootstrap

Bootstrap merupakan *framework* HTML, CSS, *Javascript* populer untuk membangun situs web yang *responsive*. RWD (*Responsive Web Design*) adalah

desain situs yang otomatis akan menyesuaikan diri agar tampil baik di semua perangkat dari ponsel sampai dekstop. Bootstrap terdiri dari satu set *file* css, *javascript*, dan *jquery*[21].

2.2.16 Postman

Postman adalah satu-satunya *platform* pengembangan API yang lengkap. Alat bawaan *Postman* mendukung setiap tahap siklus hidup API. Anda dapat merancang dan men-*debug*, secara otomatis menguji, mendokumentasikan, memantau, dan menerbitkan API.

Postman tersedia untuk MacOS, Windows, dan Linux. *Platform* pengembangan API lengkap *Postman* memungkinkan kita untuk membuat permintaan, membangun dan mengelola koleksi, dan mengatur siklus hidup API yang kita miliki[22].

2.2.17 Android

Android adalah sebuah sistem operasi pada *handphone* yang bersifat terbuka dan berbasis pada sistem operasi Linux. *Android* bisa digunakan oleh setiap orang yang ingin menggunakannya pada perangkat mereka. *Android* menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang akan digunakan untuk bermacam peranti bergerak. Awalnya, Google Inc. membeli *Android* Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan *Android*, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, TMobile, dan Nvidia[23].

2.2.18 Android Studio

Android studio adalah IDE (Integrated Development Environment) resmi untuk pengembangan aplikasi *Android* dan bersifat open source atau gratis. Peluncuran *Android Studio* ini diumumkan oleh Google pada 16 mei 2013 pada event Google I/O Conference untuk tahun 2013. Sejak saat itu, *Android Studio* mengantikan *Eclipse* sebagai IDE resmi untuk mengembangkan aplikasi *Android*.

Android studio sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan *Eclipse* disertai dengan ADT plugin (Android Development Tools). *Android studio* memiliki fitur : a. Projek berbasis pada Gradle Build b. Refactory dan pembenahan bug yang cepat c. Tools baru yang bernama “Lint” dikalim dapat

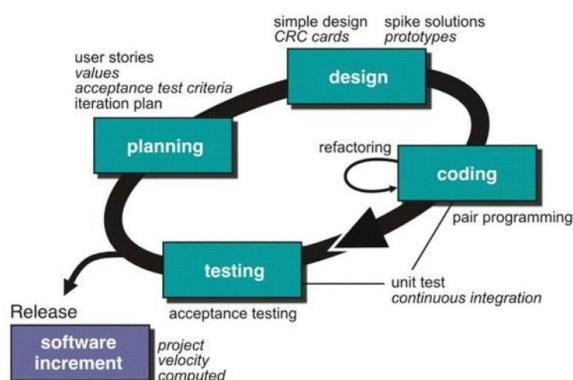
memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat. d. Mendukung Proguard And App-signing untuk keamanan. e. Memiliki GUI aplikasi android lebih mudah f. Didukung oleh Google Cloud Platfrom untuk setiap aplikasi yang dikembangkan [24].

2.2.19 Java Development Kit (JDK)

Java Development Kit (JDK) adalah sekumpulan perangkat lunak yang dapat kamu gunakan untuk mengembangkan perangkat lunak yang berbasis Java, sedangkan JRE adalah sebuah implementasi dari Java Virtual Machine yang benarbenar digunakan untuk menjalankan program java. Biasanya, setiap JDK berisi satu atau lebih JRE dan berbagai alat pengembangan lain seperti sumber compiler java, bundling, debuggers, development libraries dan lain sebagainya[24].

2.2.20 Metode pengembangan sistem *Extreme Programming*

Extreme Programming (XP) merupakan sebuah proses rekayasa perangkat lunak yang cenderung menggunakan pendekatan berorientasi objek dan sasaran dari metode ini adalah tim yang dibentuk dalam skala kecil sampai medium serta metode ini juga sesuai jika tim dihadapkan dengan *requirement* yang tidak jelas maupun terjadi perubahan-perubahan requirement yang sangat cepat [6]. XP ditujukan untuk tim kecil dan menengah, ukuran tim dibatasi antara tiga dan maksimal dua puluh anggota proyek, dan *pair programming* (dua *programmer* yang melakukan pengkodean di satu computer) merupakan karakteristik dari XP. Dibandingkan dengan *scrum* yang beroperasi dalam tim yang terdiri dari lima hingga sembilan anggota [25]. Pada Gambar 2.2 berikut merupakan kerangka kerja metode Extreme Programming (XP) in meliputi: *Planning*, *Design*, *Coding* dan *Testing* :



Gambar 2.2 Siklus *Extreme Programming*

Siklus *Extreme Programming* membagi sebuah proyek menjadi beberapa fase yaitu:

- 1 *Planning* : Pembuatan perencanaan untuk mendapatkan gambaran fitur dan fungsi dari perangkat lunak yang akan dibangun. Aktivitas *planning* dimulai dengan membuat kumpulan gambaran atau cerita (*user stories*) yang telah diberikan oleh *client* kemudian dari hasil *user stories* tersebut maka akan dilakukan perencanaan kebutuhan-kebutuhan mana yang akan dikerjakan proses ini dinamakan *iteration planning*.
- 2 *Design* : Metode ini menekankan desain aplikasi yang sederhana (*simple design*). Pada fase ini digunakan *Class Responsibilities Collaboration Card* (CRC-Card) untuk merancang sebuah sistem dan untuk mengetahui interaksi atau hubungan antar objek yang ada pada sistem. CRC-Card merupakan sebuah kartu indeks yang terbagi menjadi tiga bagian yaitu nama kelas, *responsibility*, dan *collaborator* yang digunakan untuk mendeklarasikan objek-objek yang ada pada kelas dan kelas yang saling berhubungan. Digunakannya CRC-Card pada metode XP adalah karena mengacu pada nilai kesederhanaan yang ada pada metode XP itu sendiri.
- 3 *Coding* : XP menerapkan konsep *Pair Programming* dimana proses *coding* dikembangkan oleh dua orang *programmer*, namun XP juga dapat dilakukan oleh *programmer* tunggal yang disebut sebagai model Personal *Extreme Programmer* (PXP). Model PXP ini memiliki prinsip, nilai, dan langkah kerja yang sama dengan XP[7]. Dalam membangun sebuah aplikasi pada metode XP ini terlebih dahulu dilakukan pengujian terhadap kode program yang akan diimplementasikan. Pengujian kode program pada fase ini menggunakan teknik *Test Driven Development* (TDD) dimana melakukan pengembangan perangkat lunak berdasarkan hasil pengujian yang telah dilakukan, hal ini bermaksud untuk dapat mengurangi waktu pengembangan dan dapat minimalisir adanya *bug* atau *error* pada aplikasi[26].
- 4 *Testing* : Dalam XP juga terdapat *acceptance test* atau biasa disebut *customer test*. Pengujian ini dilakukan oleh *customer* yang berfokus kepada fitur dan fungsi dari sistem secara keseluruhan. *Acceptance test* ini berasal dari *user stories* yang telah diimplementasikan [27].

- 5 Setelah semua pengujian selesai dilakukan, apabila terdapat *error* atau kesalahan pada aplikasi maka akan segera dilakukan perbaikan pada struktur kode atau dilakukan *refactoring* pada *error* tersebut. Dan apabila terdapat penambahan permintaan kebutuhan (*software increment*) dari pengguna maka akan dicatat dan masuk ke dalam *iteration planning*. Jika tidak ada *error* ataupun permintaan kebutuhan baru oleh pengguna maka aplikasi akan diluncurkan atau *release*[26].

2.2.21 Personal Extreme Programming

Personal Extreme Programming atau PXP merupakan metode pengembangan perangkat lunak yang termasuk kedalam *agile software development*. PXP merupakan metode pengembangan dari *Extreme Programming* yang disesuaikan agar dapat dikerjakan oleh pengembang tunggal[28]. Pada Tabel 2.1 menunjukkan pemetaan antara XP dan PXP dalam hal dua belas praktik inti[29].

Tabel 2.1 *Twelve Core Practices Compared*

No	Core Practices	Extreme Programming (XP)	Personal Extreme Programming (PXP)
1	<i>The Planning Game</i>	Kegiatan utama dalam <i>planning game</i> adalah menulis, estimasi, prioritas atau <i>user stories</i> yang dilakukan pengembang <i>client</i> . <i>Client</i> dan pengembangan berkumpul untuk memutuskan fitur apa dari sistem yang diperlukan yang akan menjadi nilai maksimum bagi bisnis.	Jika pengembang bekerja untuk dirinya sendiri atau cukup tahu untuk menggantikan pelanggan, maka seseorang hanya perlu berganti peran (bermain peran) selama perencanaan. Merupakan praktik yang baik untuk menulis <i>user stories</i> , memperkirakan, memprioritaskan, dan melacak saat mengembangkan sendiri. Menulis dan menggunakan

			cerita adalah cara yang baik untuk memecah proyek besar menjadi bagian-bagian kecil yang lebih mudah ditangani dan dikelola.
2	<i>Small Releases</i>	Sistem sederhana yang berisi serangkaian fitur yang berguna dimasukkan ke dalam produksi lebih awal dan sering diperbarui dalam siklus yang singkat.	Rilis awal dan sering kali dapat diterapkan dengan mudah dengan pemrogram tunggal seperti dengan tim pengembang.
3	<i>Metaphor</i>	Setiap proyek memiliki "nama sistem" dan deskripsi yang membantu memandu proses pengembangan dan komunikasi antara semua pihak.	Tim satu orang dapat menggunakan dan menyempurnakan metafora apa pun yang terbukti paling baik. Selalu pilih metafora yang membantu Anda memahami bagian-bagian sistem yang Anda bicarakan
4	<i>Simple Design</i>	Desain paling sederhana selalu digunakan untuk membangun aplikasi selama memenuhi persyaratan bisnis saat ini. Jangan khawatir tentang persyaratan masa depan karena persyaratan berubah seiring waktu.	Sebagai pengembang tunggal, mudah untuk mempertahankan desain yang sederhana. Solusinya adalah menggunakan desain tingkat tinggi sebagai tujuan untuk dikerjakan, tetapi mengembangkan desain alternatif yang lebih sederhana dan lebih baik

			tetapi tetap memberikan semua fungsionalitas.
5	<i>Testing</i>	<i>Test-driven development</i> adalah salah satu kekuatan utama XP. Perangkat lunak yang dikembangkan dengan XP divalidasi setiap saat. Sebelum fitur baru ditambahkan, tes ditulis untuk memverifikasi perangkat lunak. Perangkat lunak tersebut kemudian dikembangkan untuk lulus tes ini.	Saat menulis modul baru, programmer harus menulis antarmuka terlebih dahulu, kemudian unit test, dan baru kemudian melanjutkan untuk mengimplementasikan modul. Dalam satu orang, setelan tes tim dapat dengan mudah ditulis dan digunakan. Pengembang harus menguji dan mengkompilasi kodanya sebelum memasukkannya ke dalam baseline refactor. Sampai saat itu, kode tetap berada di baseline pengembangan
6	<i>Refactoring</i>	Merupaan teknik untuk meningkatkan desain basis kode yang sudah ada. Esensinya adalah menerapkan serangkaian transformasi kecil yang melestarikan perilaku yang meningkatkan struktur kode. Melakukannya dalam langkah-langkah kecil mengurangi risiko terjadinya kesalahan	Sebagai seorang pemrogram tunggal, seseorang dapat dan harus mempraktikkan pemfaktoran ulang ke hulu kecuali dalam situasi di mana Anda tidak memiliki izin untuk mengubah kode yang tidak Anda miliki. Kode yang difaktorkan ulang harus diintegrasikan

			dengan baseline produksi dan diuji.
7	<i>Pair Programming</i>	Pemrogram yang menggunakan XP dipasangkan dan menulis semua kode produksi menggunakan satu mesin per pasang. Ini membantu kode untuk terus ditinjau saat sedang ditulis. Pemrograman berpasangan telah terbukti menghasilkan kode berkualitas tinggi dengan sedikit atau tanpa penurunan produktivitas.	Saat bekerja sebagai programmer mandiri, manfaat dari pemrograman berpasangan akan hilang. Anda bisa meminta rekan kerja untuk sering memberi Anda pengingat. Panduan informal dan uji filosofi pertama dapat diterapkan.
8	<i>Collective Code Ownership</i>	Semua kode adalah milik setiap anggota tim, tidak ada satu pun anggota tim yang memiliki sepotong kode dan siapa pun dapat membuat perubahan pada basis kode kapan saja. Ini mendorong semua orang untuk menyumbangkan ide-ide baru ke semua segmen proyek.	Jika Anda adalah satu-satunya pengembang dalam proyek ini maka tidak ada masalah karena Anda memiliki semua kode sumber. Namun, Anda tidak akan mendapatkan keuntungan dari masukan pemrogram lain. Pengembang dapat mempertahankan versi kode dalam repositori dan teknik kontrol versi dapat diterapkan.
9	<i>Continuous Integration</i>	Tujuan dari integrasi berkelanjutan adalah untuk mencegah atau mengurangi	Jika pengembang bekerja sendiri dengan tidak ada orang lain yang membuat

		<p>penyebaran kode dari basis kode utama; semakin sering kode diintegrasikan ke dalam basis kode utama, semakin kecil kemungkinan terjadinya pengalihan. Sistem perangkat lunak dibangun dan diintegrasikan beberapa kali sehari; setidaknya semua perubahan diintegrasikan ke dalam basis kode utama setidaknya sekali sehari. Setiap build diuji menggunakan kasus uji yang disiapkan.</p>	<p>perubahan pada basis kode utama, maka tidak ada masalah karena basis kode berfungsi sebagai catatan linier dari pekerjaan seseorang. Integrasi tidak akan menyebabkan konflik apa pun dan bebas masalah tetapi masih cukup mudah untuk menyimpang dari basis kode utama semakin lama Anda bekerja tanpa mengintegrasikan. Oleh karena itu, integrasi berkelanjutan tetap diperlukan. Cara terbaik untuk mengembangkannya adalah dengan mengerjakan tugas, mengintegrasikan, dan kemudian pindah ke tugas berikutnya. Dengan cara ini divergensi dijaga seminimal mungkin</p>
10	<i>40-Hour Week</i>	<p>Programmer dalam proyek XP biasanya mematuhi 40 jam kerja seminggu untuk menjaga produktivitas dan menghindari kelelahan</p>	<p>Semua yang diperlukan untuk melakukan ini adalah mematuhi 40 jam seminggu dan berhenti bekerja pada hari yang Anda butuhkan. Artinya berhenti bekerja saat tidak produktif lagi, stres atau</p>

			lelah untuk mengurangi rasa lelah dan membuat Anda dalam kondisi prima.
11	<i>On-site Customer</i>	Satu atau lebih pelanggan yang akan menggunakan sistem yang dibangun dialokasikan ke tim pengembangan. Pelanggan membantu memandu pengembangan dan diberdayakan untuk memprioritaskan dan menyatakan persyaratan atau menjawab pertanyaan apa pun yang mungkin dimiliki pengembang. Hal ini memastikan bahwa ada komunikasi yang efektif dengan pelanggan dan akibatnya akan lebih sedikit dokumentasi yang diperlukan	Jika Anda adalah pelanggan Anda sendiri (setidaknya pada awalnya) maka bergumam pada diri sendiri adalah hal yang wajar. Jika Anda memiliki pelanggan lain maka komunikasi melalui email atau telepon mungkin akan meredakan masalah ini selama pelanggan terbuka untuk komunikasi.
12	<i>Coding Standards</i>	Setiap orang menggunakan standar pengkodean yang sama yang memudahkan untuk bekerja berpasangan dan berbagi kepemilikan semua kode. Seseorang seharusnya tidak dapat mengetahui siapa yang mengerjakan kode apa dalam proyek XP.	Sebagai tim satu orang, cara Anda memilih untuk membuat kode adalah standar pengkodean Anda. Tetapi beberapa standar pengkodean harus diikuti untuk menjaga konsistensi selama pengembangan

BAB III

METODE PENELITIAN

3.1 Alat dan Bahan

Dalam penelitian Tugas Akhir ini memerlukan beberapa alat yang terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*) serta bahan yang berupa data-data dan informasi yang diperlukan.

3.1.1 Alat

Adapun alat-alat yang digunakan dalam penelitian Tugas Akhir ini dapat dirincikan sebagai berikut.

1. Laptop ACER Aspire 4739 dengan *processor* Core i3 2.4GbHz *hardisk* 500 Gb menggunakan OS (*operating system*) Windows 10 Pro.
2. *CodeIgniter*, merupakan *framework* bahasa pemrograman PHP yang digunakan untuk membangun sistem berbasis *web*.
3. XAMPP, sebagai penyedia *localhost web server*.
4. MySQL, sebagai *server database*.
5. Visual Studio Code, sebagai *text editor* pada saat melakukan proses *coding*.
6. Microsoft Word 2016, digunakan untuk menyusun laporan.
7. Microsoft Visio 2016, digunakan untuk membuat rancangan diagram.
8. Mendeley, digunakan untuk membuat sitasi dan daftar pustaka.
9. Postman, digunakan sebagai *tester* terhadap penyedia layanan API.
10. Android Studio, digunakan untuk membangun sistem berbasis *android*.

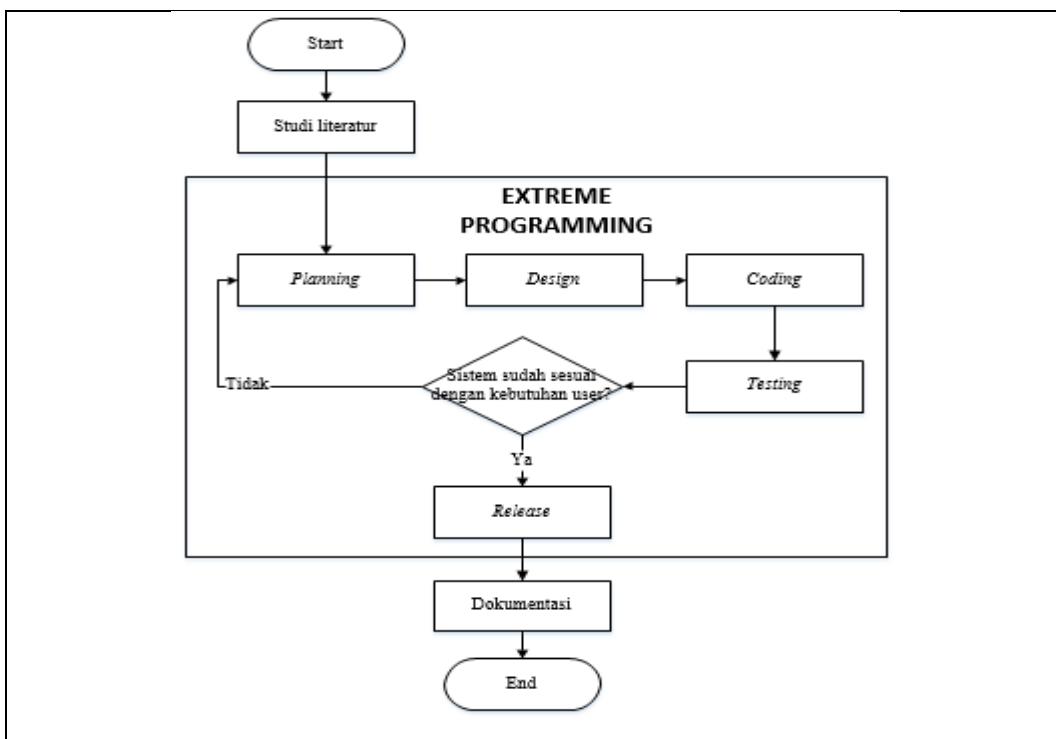
3.1.2 Bahan

Adapun sumber data sebagai bahan yang digunakan dalam penelitian Tugas Akhir ini adalah sebagai berikut.

1. Data setiap tempat ibadah yang ada di Kota Mataram beserta deskripsinya.yang didapat dari kemenag NTB.
2. Data *latitude* dan *longitude* tempat ibadah di Kota Mataram.

3.2 Diagram alir penelitian

Pada penelitian Tugas Akhir ini menggunakan metodologi pengembangan sistem *Extreme Programming* yang terdiri dari 4 tahapan yaitu : *planning*, *design*, *coding*, dan *testing*. Tahap-tahap pengembangan sistem tersebut diilustrasikan pada diagram alir penelitian pada Gambar 3.1 berikut :



Gambar 3.1 Diagram Alir Penelitian

3.2.1 Studi Literatur

Tahap pertama dalam penelitian Tugas Akhir ini adalah studi literatur. Studi literatur dilakukan dengan mengumpulkan data dan memahami teori-teori dari berbagai buku, skripsi, jurnal dan penelitian-penelitian sebelumnya yang berkaitan dengan pembuatan dan perancangan aplikasi pemetaan tempat ibadah pada penelitian Tugas Akhir ini.

3.2.2 *Planning*

Pada tahap ini merupakan analisa terhadap kebutuhan dalam pengembangan sistem yang akan dibuat berupa data-data informasi meliputi lokasi tempat ibadah, deskripsi tempat ibadah, garis bujur dan garis lintang tempat ibadah, hingga foto sesungguhnya dari tempat ibadah yang ada di Kota Mataram dan perencanaan untuk mendapatkan gambaran fitur dan fungsi dari perangkat lunak yang akan dibangun.

Sistem ini ditujukan untuk masyarakat umum yang ingin mendatangi tempat ibadah yang diinginkan dan pihak Kemenag prov. NTB sebagai admin yang mengelolanya. Sistem pada aplikasi ini dapat memberikan informasi terkait dengan destinasi tempat ibadah seperti lokasi pada peta, rute dari lokasi pengguna ke tempat ibadah. Pada penelitian Tugas Akhir ini, penulis menggunakan beberapa metode dalam pengumpulan kebutuhan yaitu melalui kuesioner, wawancara, mengambil titik koordinat dari *google maps* dan observasi secara langsung.

1. Kuesioner

Pada penelitian Tugas Akhir ini, kuesioner ditujukan kepada masyarakat umum guna untuk mendapatkan respon dari masyarakat terkait dengan penting atau tidak dibangunnya sistem pada penelitian Tugas Akhir ini. Kuesioner ini dilakukan dengan cara memberikan pertanyaan-pertanyaan terkait dengan apa kegunaan tempat ibadah, seberapa pentingnya tempat ibadah untuk memenuhi kewajiban peribadatan, kesulitan apa saja yang dialami masyarakat dalam mencari informasi dan lokasi mengenai tempat ibadah, solusi apa saja yang telah dilakukan masyarakat untuk mengatasi masalah tersebut dan penting atau tidakkah masyarakat jika dibuatnya sistem informasi geografis, serta fitur yang diharapkan oleh *user* terhadap sistem yang akan dibuat.

Berdasarkan hasil-hasil kuesioner terhadap 61 responden yang telah dipaparkan sebelumnya, dapat disimpulkan bahwa masyarakat setuju untuk dibangunnya sistem pemetaan lokasi tempat ibadah ini, serta fitur yang diharapkan oleh *user* pada sistem yang akan dibuat.

2. Wawancara

Pada wawancara yang dilakukan di kantor Kementerian Agama Prov. NTB dengan Bapak H. Toni selaku *staff* sub. bagian Inmas Kementerian Agama Prov. NTB, dijelaskan tentang proses penyimpanan data tempat ibadah yang ada saat ini masih dilakukan secara konvensional yaitu menggunakan Ms. Excel. Dengan cara seperti itu untuk melakukan pengelolaan ataupun pencarian terhadap data tempat ibadah yang ada menjadikan proses nya tidak efektif dan efisien. Penulis menawarkan untuk melakukan penelitian pengembangan sebuah sistem informasi yang dapat membantu pihak

kemenag dalam mengelola data tempat ibadah yang ada, tidak hanya itu penulis juga menawarkan untuk dibuatkan sistem pemetaan lokasi tempat ibadah yang ada di Kota Mataram yang dapat dijangkau oleh masyarakat umum dalam memenuhi kewajiban peribadatan. Adapun hasil lain dari wawancara yang telah dilakukan yaitu sebagai berikut :

- a. Terdapat 5 jenis tempat ibadah yang ada di Kota Mataram, yaitu masjid, pura, gereja, vihara, dan krenteng.
 - b. Mendapatkan data informasi mengenai tempat ibadah yang ada di Kota Mataram seperti nama tempat ibadah, alamat, kecamatan, tahun berdiri dll.
 - c. Pihak kemenag hanya mengelola data mengenai masjid, pura, gereja, vihara, dan krenteng, tidak dengan musholla.
 - d. Mendapatkan *user stories* untuk kebutuhan sistem yang diinginkan oleh pengguna seperti pada Tabel 3.1.
3. Mengambil data dari *Google maps*

Pengambilan data dari *google maps* pada penelitian Tugas Akhir ini tidak hanya bertujuan untuk mendapatkan banyaknya tempat ibadah yang ada di Kota Mataram, tetapi juga bertujuan untuk mendapatkan data *latitude* dan *longitude* atau titik koordinat terhadap lokasi masing-masing tempat ibadah. Adapun guna dari pada titik koordinat ini yakni untuk pembuatan *marker* pada peta sistem yang akan dibuat.

4. Observasi Langsung

Karena tidak semua tempat ibadah yang ada di Kota Mataram terdaftar dalam *google maps*, sehingga perlu dilakukan observasi langsung untuk mendapat lebih banyak data mengenai adanya tempat ibadah di Kota Mataram. Observasi dilakukan dengan cara menelusuri atau mencari langsung keberadaan tempat ibadah yang ada di Kota Mataram. Hasil yang didapat dari observasi yang dilakukan yakni terdapat beberapa tempat ibadah salah satu contohnya GKII Kalvari Ampenan yang masih berada dalam wilayah Kota Mataram dan terdapat dalam data kemenag, namun tidak terdaftar dalam *google maps*. Oleh karena itu guna dari pada observasi langsung yang penulis lakukan adalah untuk mendaftarkan tempat ibadah yang tidak terdaftar pada

google maps ke dalam peta sistem yang akan dibuat, sehingga nantinya dapat menjadi acuan sebagai sistem informasi rujukan untuk mengetahui lokasi tempat ibadah yang valid posisinya.

3.2.2.1 User stories

Pada tahap ini pengguna menentukan requirement-nya dan ditulis dalam bentuk *user story*. *User story card* yang baik menerapkan model INVEST yaitu *Independent*, *Negotiable*, *Valuable*, *Estimable*, dan *Small*. *Independent* berarti tidak adanya ketergantungan antar *user story*, *Negotiable* berarti seluruh detil diperoleh melalui kolaborasi, *Valuable* berarti memiliki nilai bagi pengguna, *Estimable* berarti dapat diperkirakan baik dalam segi waktu dan biaya, *Testable* berarti dapat ditentukan *acceptance criteria*-nya. Sebuah *user story* dapat dikatakan “selesai” jika *acceptance criteria user story* tersebut telah dipenuhi[30].

Pembahasan penelitian Tugas Akhir ini dimulai dengan merangkum *user story* dalam bentuk tabel (Tabel 3.1). Terdapat 4 poin dasar *user story* yaitu, nama *user story*, deskripsi tujuan *user story* berupa paragraf pendek, estimasi waktu berapa lama *user story* akan diimplementasikan, dan tingkat kepentingan *user story* (*seperti must have, should have*)[31]. *User story* pada Tabel 3.1 dikelompokkan berdasarkan pengguna aplikasi baik secara umum maupun lebih detail berdasarkan kebutuhan. Pengguna aplikasi adalah masyarakat umum yang tidak perlu login untuk dapat mengakses sistem dan pihak Kemenag prov. NTB admin yang harus login untuk dapat mengakses dan mengelola data pada sistem. Berikut merupakan rangkuman dari *user story* pada sistem yang akan dibuat dapat dilihat pada Tabel 3.1.

Tabel 3.1 *User Story Cards*

No	Kode US	Judul	Deskripsi	Acceptance Criteria	Estimasi (hari)
1	US-01	Login	Terdapat 2 jenis sub. Bagian yaitu Bimas Islam yang berperan	Terdapat tampilan <i>login</i> dimana super admin dan admin diharuskan memasukkan <i>username</i> dan	4

			sebagai admin dan Inmas yang berperan sebagai super admin yang dapat masuk ke dalam sistem. Dan admin dapat <i>logout</i> dari system	<i>password</i> yang telah terdaftar dalam sistem untuk dapat mengakses sistem. Sedangkan masyarakat tidak perlu login untuk dapat mengakses sistem.	
2	US-02	Melihat data admin	Sebagai Super Admin, saya dapat melihat daftar admin berupa <i>list</i> admin serta deskripsinya.	Setelah super admin melakukan login, terdapat tampilan daftar admin pada menu navbar dan terdapat <i>list</i> admin beserta deskripsinya.	2
3	US-03	Menambah data admin	Sebagai Super Admin, saya dapat menambah data admin baru beserta deskripsinya.	Setelah super admin melakukan login, pada bagian menu navbar daftar admin, super admin dapat menambah data admin baru beserta deskripsinya dengan cara mengisi <i>form</i> untuk tambah data admin baru.	2

4	US-04	Mengubah data admin	Sebagai Super Admin, saya dapat mengubah data admin.	Pada bagian <i>list</i> admin, super admin dapat mengubah data admin yang dinginkan dengan memasukkan data admin yang baru yang terdapat dalam <i>form</i> ubah data admin.	2
5	US-05	Menghapus data admin	Sebagai Super Admin, saya dapat menghapus data admin.	Pada bagian <i>list</i> admin, super admin dapat menghapus data admin yang dinginkan dengan memilih data admin yang akan dihapus dan kemudian menekan tombol hapus data admin.	2
6	US-06	Melihat daftar tempat ibadah	Sebagai Super Admin, Admin, dan masyarakat saya dapat melihat daftar tempat ibadah meliputi nama tempat ibadah dan deskripsinya	Setelah super admin dan admin melakukan login, terdapat tampilan daftar tempat ibadah pada menu navbar dan terdapat <i>list</i> tempat ibadah serta deskripsinya, dan masyarakat langsung dapat melihat <i>list</i> tempat ibadah tanpa perlu login.	6

7	US-07	Menambah data tempat ibadah	Sebagai Super Admin, dan Admin, saya dapat menambah data tempat ibadah baru beserta deskripsinya.	Pada bagian <i>list</i> tempat ibadah, super admin dan admin dapat mengubah data tempat ibadah yang dinginkan dengan memasukkan data tempat ibadah yang baru yang terdapat dalam <i>form</i> ubah data tempat ibadah.	2
8	US-08	Mengubah data tempat ibadah	Sebagai Super Admin, dan Admin, saya dapat mengubah data tempat ibadah.	Setelah super admin dan admin melakukan login, pada bagian menu navbar daftar tempat ibadah, super admin dan admin dapat menambah data tempat ibadah baru beserta deskripsinya dengan cara mengisi <i>form</i> untuk tambah data tempat ibadah baru.	2
9	US-09	Menghapus data tempat ibadah	Sebagai Super Admin, dan Admin, saya dapat menghapus data tempat ibadah.	Setelah super admin dan admin melakukan login, pada bagian menu navbar daftar tempat ibadah, super admin dan admin dapat menghapus data tempat ibadah dengan	2

				cara menekan tombol untuk hapus data tempat ibadah.	
10	US-10	Melihat Peta Lokasi tempat ibadah	Masyarakat sebagai pengguna, saya dapat mengakses lokasi terkini	Terdapat tampilan peta dan tombol untuk menentukan lokasi terkini pengguna	3
11	US-11	Melihat Rute Lokasi Tempat Ibadah	Masyarakat sebagai pengguna, saya dapat mengakses rute / jalur menuju tempat ibadah	Terdapat tampilan peta dan pilihan untuk menuju lokasi tujuan serta terdapat tombol untuk dapat mengetahui rute dari lokasi pengguna menuju lokasi tempat ibadah.	3
Total story points					30

Pada penelitian Tugas Akhir ini, direncanakan implementasi program dapat terselesaikan dalam waktu 30 hari aktif kerja, hari aktif kerja disini mencakup hari *weekday*. *Velocity* merupakan waktu yang diperlukan untuk menyelesaikan tiap iterasi. nilai *velocity* bisa ditentukan dengan cara memperkirakan nilai *velocity* itu sendiri. Nilai *velocity* ditentukan sendiri oleh pihak pengembang[32]. Nilai *velocity* yang digunakan pada penelitian Tugas Akhir ini adalah 10.

3.2.2.2 Iteration plan

Release plan merupakan proses menentukan *user stories* mana yang akan diimplementasikan pada tiap iterasi. *Release plan* meliputi perencanaan *estimasi story* dan pembagian iterasi untuk setiap *story*. Dalam mencari jumlah iterasi digunakan dengan membagi *total story* dengan *velocity*[31]. Penelitian Tugas Akhir ini memiliki *story points* yang berjumlah 30 dan nilai *velocity* adalah 10, maka dapat

diketahui jumlah iterasi adalah $30/10 = 3$ iterasi. Tiap iterasi pada pembuatan sistem pada penelitian Tugas Akhir ini akan diselesaikan dalam kurun waktu 10 hari. Pembagian iterasi dan alokasi *story* dapat dilihat pada Tabel 3.2.

Tabel 3.2 Pembagian Iterasi Story

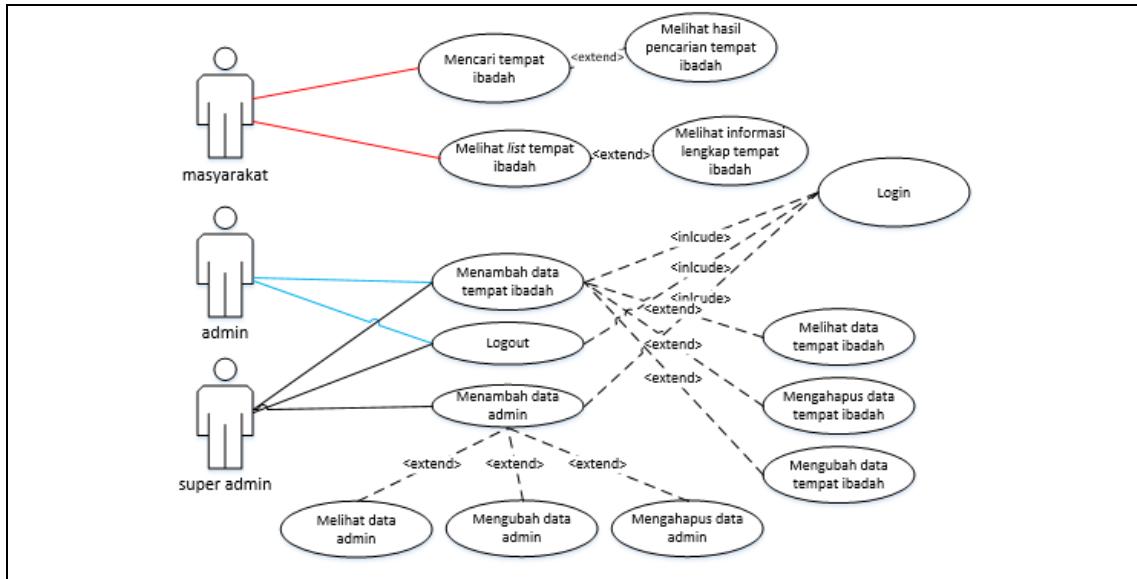
No	Kode US	Deskripsi	Estimasi (hari)
<i>Iteration – 1</i>			
1	US-01	Login ke dalam system dan mengakhiri sistem	4
2	US-06	Melihat daftar dan detail data tempat ibadah	6
<i>Velocity</i>			10
<i>Iteration – 2</i>			
3	US-02	Melihat daftar dan detail data admin	2
4	US-03	Menambah data admin	2
5	US-04	Mengubah data admin	2
6	US-05	Menghapus data admin	2
7	US-07	Menambah data tempat ibadah	2
<i>Velocity</i>			10
<i>Iteration – 3</i>			
8	US-08	Mengubah data tempat ibadah	2
9	US-09	Menghapus data tempat ibadah	2
10	US-10	Melihat peta lokasi tempat ibadah	3
11	US-11	Melihat rute lokasi tempat ibadah	3
<i>Velocity</i>			10

3.2.3 Design

Desain sistem merupakan tahapan perancangan atau penggambaran tentang bagaimana sistem yang dikembangkan akan bekerja. Pada penelitian Tugas Akhir ini, sistem dirancang dengan konsep berorientasi objek menggunakan UML dan *Class Responsibilities Collaboration Card* (CRC-Card) untuk mengetahui interaksi atau hubungan antar objek yang ada pada sistem.

3.2.3.1 Usecase Diagram

Berikut merupakan *usecase diagram* dari Sistem Informasi Geografis Pemetaan Tempat Ibadah di Kota Mataram dapat dilihat pada Gambar 3.2:



Gambar 3.2 Usecase Diagram

Gambar 3.2 merupakan *usecase* dengan aktor super admin, admin dan masyarakat umum. Admin merupakan pihak kemenag yg berperan sebagai admin yang dapat melakukan aksi pada sistem, terlebih dahulu admin harus melakukan *login* menggunakan akun yang sudah diberikan oleh super admin. Berikut yaitu aksi-aksi yang dapat dilakukan oleh admin:

1. Melakukan *login*.
2. Melihat daftar tempat ibadah.
3. Mencari tempat ibadah.
4. Melihat informasi hasil pencarian.
5. Melihat informasi lengkap tempat ibadah.
6. Mengelola data tempat ibadah.
7. Mengakhiri sistem.

Kemudian masyarakat merupakan pengguna yang dapat mengakses sistem ini. berikut merupakan aksi-aksi yang dapat dilakukan oleh masyarakat:

1. Melihat daftar tempat ibadah
2. Mencari tempat ibadah.
3. Melihat informasi hasil pencarian
4. Melihat informasi lengkap tempat ibadah

Kemudian pihak kemenag juga merupakan Super admin yang dapat mengakses sistem ini, dimana aksi yang dilakukan sama halnya dengan admin dan *user*, namun super admin dapat mengelola data admin.

3.2.3.2 Class Responsibilities Collaboration Card (CRC-Card)

CRC-Card didapat dari penjabaran hubungan *responsibility* antar *class* di sistem sesuai dengan yang telah digambarkan dalam *usecase* diagram. CRC-Card merupakan sebuah kartu indeks yang terbagi menjadi tiga bagian yaitu nama kelas, *responsibility*, dan *collaborator* yang digunakan untuk mendeklarasikan objek-objek yang ada pada kelas dan kelas yang saling berhubungan. Digunakannya CRC-Card pada metode XP adalah karena mengacu pada nilai kesederhanaan yang ada pada metode XP itu sendiri[26]. *Spike solution prototype* merupakan skema desain *prototype* berdasarkan CRC Cards dari *user stories*. Penjelasan CRC-Card pada penelitian Tugas Akhir ini dapat dilihat pada Tabel 3.3 :

Tabel 3.3 Penjelasan CRC-Card

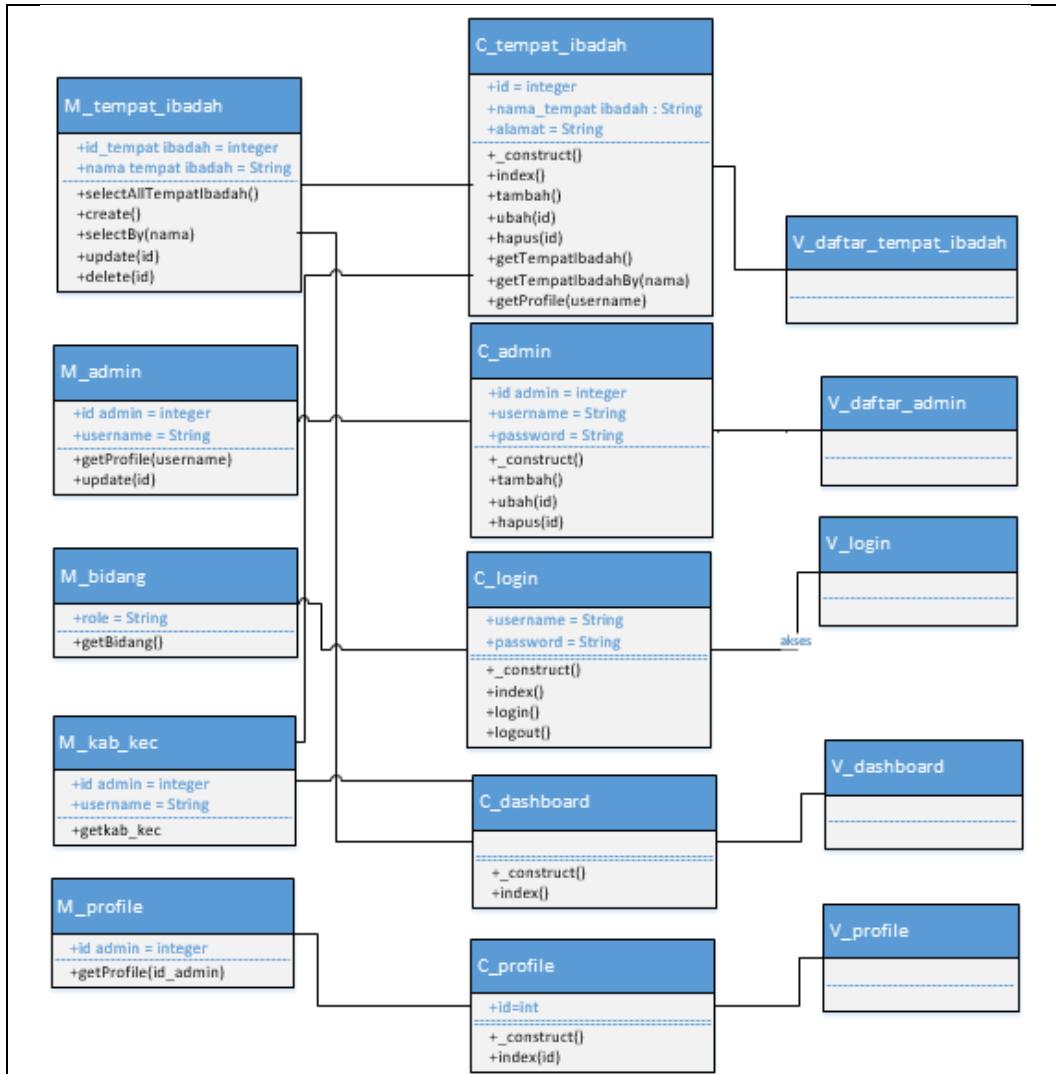
No	Nama	Class	Responsibilities	Collaborator
1.	Tempat ibadah basis web	C Tempat Ibadah	Mengatur jalannya data tempat ibadah yang akan dikelola maupun yang akan ditampilkan pada sistem.	M tempat ibadah, V daftar tempat ibadah
		M tempat ibadah	Digunakan untuk mengambil dan mengelola data tempat ibadah seperti menambah, mengubah serta menghapus data tempat ibadah dalam <i>database</i> .	C tempat ibadah
		V daftar tempat ibadah	Menampilkan data tempat ibadah yang tersusun dalam bentuk tabel.	C tempat ibadah, M Tempat ibadah
		M kab kec	Untuk mengambil data kabupaten dan kecamatan	C tempat ibadha
2.	Admin	C admin	Mengatur jalannya data admin yang akan dikelola	M admin, V daftar admin

			maupun yang akan ditampilkan pada sistem.	
		M admin	Digunakan untuk mengambil dan mengelola data admin seperti menambah, mengubah serta menghapus data admin dalam <i>database</i> .	C admin
		V daftar admin	Menampilkan data admin yang tersusun dalam bentuk tabel	C admin, M admin
3.	<i>Login</i>	C login	Mengatur jalannya validasi pengguna yang akan mengakses system dan untuk mengakhiri system (<i>logout</i>)	M admin, V login
		M admin	Digunakan untuk validasi kesesuaian <i>username</i> dan <i>password</i> pengguna yang ada pada <i>database</i> .	C login,
		V login	Menampilkan <i>form</i> untuk login	C login
4.	<i>Dashboard</i>	C dashboard	Untuk mengatur jalannya data yang akan ditampilkan pada halaman <i>dashboard</i>	M tempat ibadah, V dashboard
		V dashboard	Menampilkan daftar tempat ibadah dan peta lokasi serta rute tempat ibadah.	C dashboard
		M tempat ibadah	Digunakan untuk mengambil data tempat dalam <i>database</i> .	C dashboard

		M kab kec	Untuk mengambil data kabupaten dan kecamatan	C tempat ibadha
5.	<i>Profile</i>	C profile	Untuk menampilkan dan mengatur jalannya data <i>profile</i> admin yang telah login	M admin, V profile
		V profile	Untuk menampilkan data admin yang telah login	C profile
		M profile	Untuk mengambil data admin yang telah <i>login</i>	C profile
6	Tempat ibadah basis android	C Tempat Ibadah	Berguna sebagai interface yang ada pada API agar dapat terhubung dengan <i>database</i>	M tempat ibadah
		M tempat ibadah	Digunakan untuk mengambil data tempat ibadah dari <i>database</i>	C tempat ibadah
		List Fragment	Menampilkan daftar tempat ibadah dalam bentuk <i>listview</i>	XML, adapter, C tempat ibadah
		Map Fragment	Menampilkan peta lokasi tempat ibadah dalam	XML, adapter, C tempat ibadah

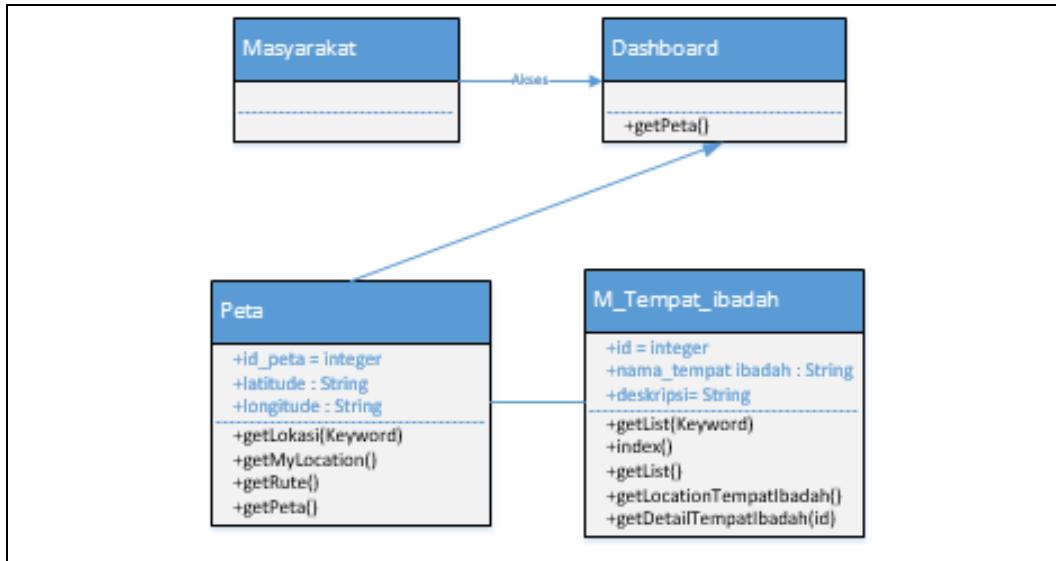
3.2.3.3 Class diagram

Berikut merupakan *class diagram* dari Sistem Informasi Geografis Pemetaan Tempat Ibadah di Kota Mataram dapat dilihat pada Gambar 3.3:



Gambar 3.3 Class Diagram Pada Web

Pada Gambar 3.3 merupakan *class diagram* pada sistem informasi pemetaan tempat ibadah pada web. Sistem ini menggunakan metode analisa dan perancangan berorientasi objek dan menggunakan pola arsitektur *Model View Controller* (MVC) di setiap modul, sehingga dilakukan perancangan kelas – kelas yang ada pada sistem yang terdiri dari *view*, *controller* dan *model*. Pada *Controller* terdapat kelas “C_tempat_ibadah”, “C_admin”, dan “C_login”. Kemudian pada *Model* terdapat beberapa kelas yaitu “M_tempat_ibadah”, “M_admin”, dan “M_kab_kec”. Sedangkan *View* ini bertugas untuk menerima dan menampilkan data kepada *user*.



Gambar 3.4 Class Diagram Pada *Android*

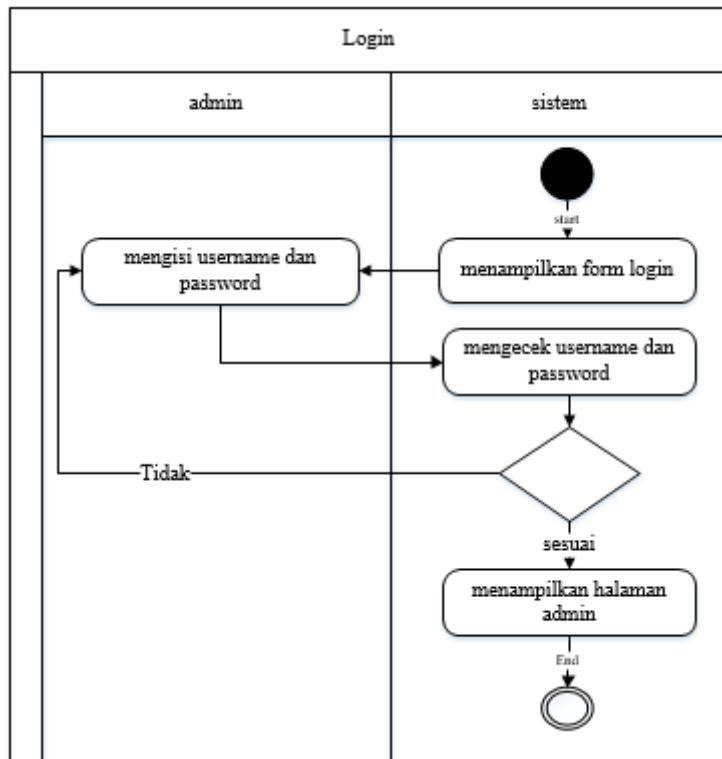
Pada Gambar 3.4 merupakan class diagram pada aplikasi *android* dimana masyarakat langsung dapat mengakses halaman *dashboard* yang memiliki *class* tempat ibadah untuk melihat daftar tempat ibadah yang dimana tempat ibadah memiliki peta / koordinatnya masing-masing.

3.2.3.4 Activity Diagram

Berikut merupakan *usecase diagram* dari Sistem Informasi Geografis Pemetaan Tempat Ibadah di Kota Mataram:

1. Proses *login* admin

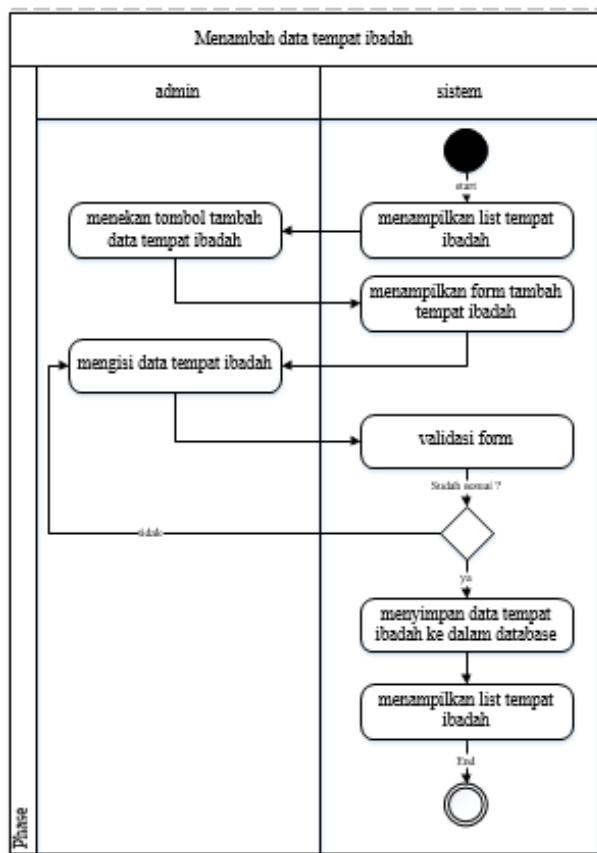
Pada sistem pemetaan tempat ibadah ini, admin dapat melakukan aktivitas *login* dengan menggunakan *username* dan *password* yang sudah diberikan. Terlebih dahulu admin mengisi form *login* yang ditampilkan sistem dengan *username* dan *password* yang sudah diberikan tersebut. Kemudian sistem melakukan validasi apakah *username* dan *password* sesuai dengan yang terdapat dalam *database*. Jika sesuai maka admin dapat mengakses halaman admin, namun jika tidak sesuai maka admin harus mengisi form *login* dengan *username* dan *password* yang sesuai. *Activity diagram* proses *login* admin dapat dilihat pada Gambar 3.5 sebagai berikut:



Gambar 3.5 Activity Diagram Proses *Login* Admin

2. Proses menambah data tempat ibadah

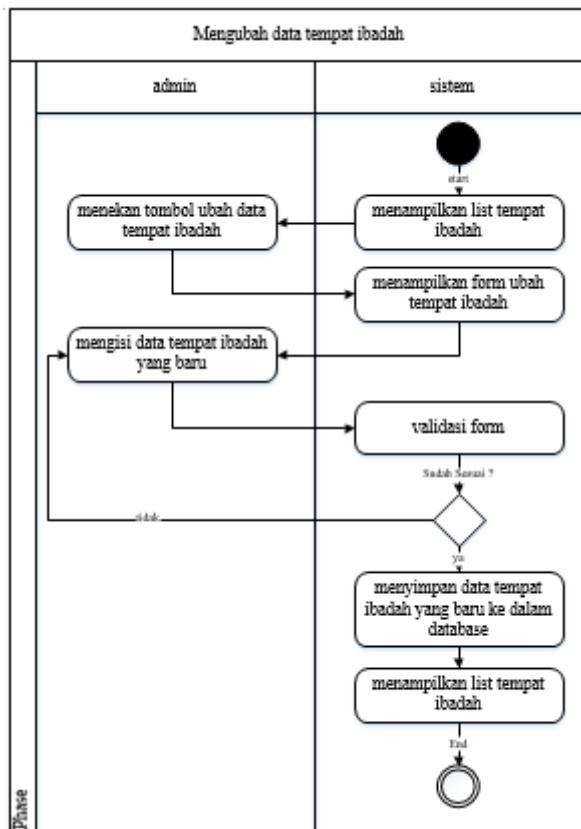
Pada sistem pemetaan tempat ibadah ini, admin dapat melakukan aktivitas menambah data tempat ibadah. Dimana terlebih dahulu sistem akan menampilkan *list* data tempat ibadah, kemudian admin dapat menambah data dengan mengisi *form* tambah tempat ibadah yang ditampilkan sistem. Sistem melakukan validasi apakah data yang diisi sudah sesuai, jika tidak admin akan diarahkan kembali mengisi form, jika sudah sesuai maka data tempat ibadah tersebut akan masuk ke dalam *list* tempat ibadah dan tersimpan dalam *database*. *Activity diagram* proses menambah data tempat ibadah dapat dilihat pada Gambar 3.6 sebagai berikut:



Gambar 3.6 Activity Diagram Proses Menambah Data Tempat Ibadah

3. Proses mengubah data tempat ibadah

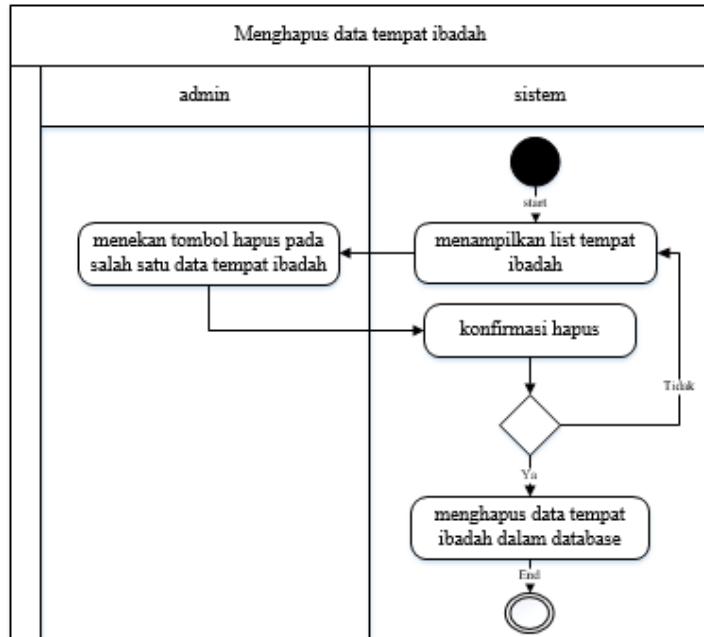
Pada sistem pemetaan tempat ibadah ini, admin dapat melakukan aktivitas mengubah data tempat ibadah. Terlebih dahulu sistem akan menampilkan *list* tempat ibadah, kemudian admin dapat memilih tempat ibadah mana yang datanya ingin diubah, setelah itu sistem akan menampilkan *form* ubah data tempat ibadah. Dalam *form* tersebut admin akan mengisi data tempat ibadah yang baru, setelah dilakukan validasi kesesuaian data maka data tempat ibadah yang baru tersebut dapat disimpan dalam *database*. *Activity* diagram proses mengubah data tempat ibadah dapat dilihat pada Gambar 3.7 sebagai berikut:



Gambar 3.7 Activity Diagram Proses Mengubah Data Tempat Ibadah

4. Proses menghapus data tempat ibadah

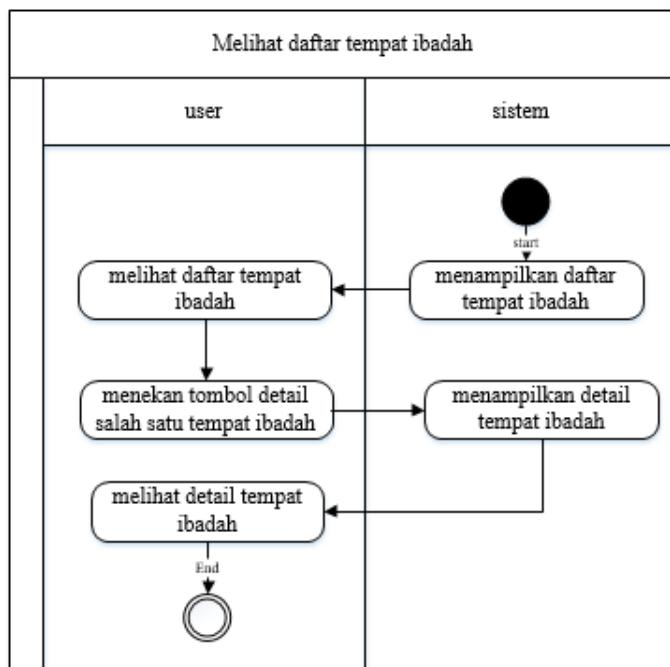
Pada sistem pemetaan tempat ibadah ini, admin dapat melakukan aktivitas menghapus data tempat ibadah. Terlebih dahulu sistem akan menampilkan list tempat ibadah, kemudian admin dapat memilih tempat ibadah mana yang datanya ingin dihapus, setelah itu sistem akan menghapus data tempat ibadah tersebut dari dalam *database*. Activity diagram proses menghapus data tempat ibadah dapat dilihat pada Gambar 3.8 sebagai berikut:



Gambar 3.8 Activity Diagram Proses Menghapus Data Tempat Ibadah

5. Proses melihat daftar tempat ibadah

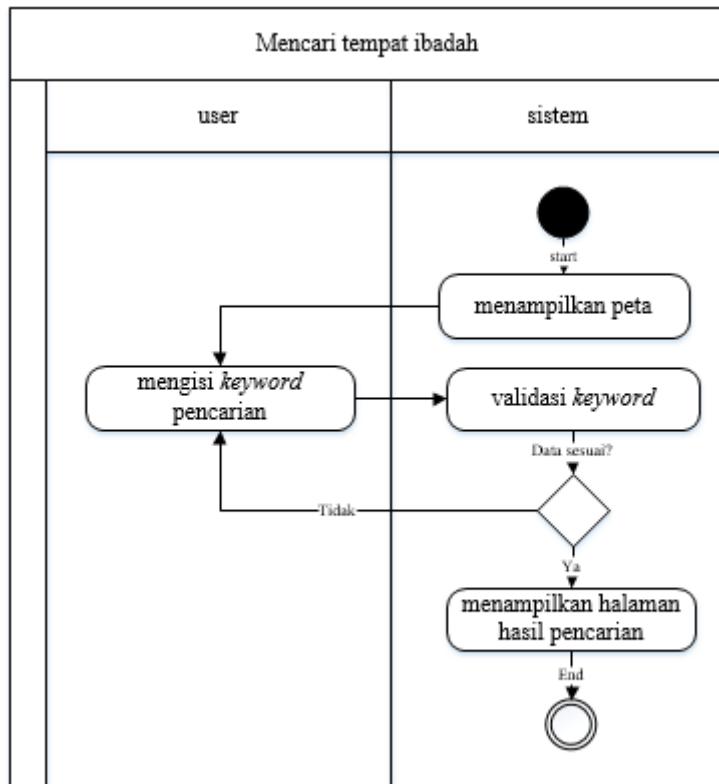
Pada sistem pemetaan tempat ibadah ini, *user* dan admin dapat melakukan aktivitas melihat tempat ibadah. Terlebih dahulu sistem akan menampilkan daftar tempat ibadah, kemudian pengguna dapat melihat daftar tempat ibadah serta detail dari tempat ibadah yang ditampilkan oleh sistem. *Activity* diagram proses melihat daftar tempat ibadah dapat dilihat pada Gambar 3.9 sebagai berikut:



Gambar 3.9 Activity Diagram Proses Melihat Daftar Tempat Ibadah

6. Proses mencari lokasi tempat ibadah

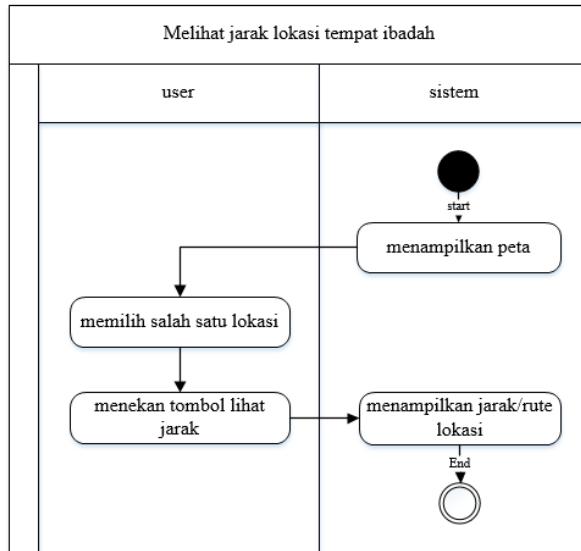
Pada sistem pemetaan tempat ibadah ini, *user* dapat melakukan aktivitas mencari lokasi tempat ibadah. Terlebih dahulu sistem akan menampilkan peta, kemudian *user* dapat memasukkan *keyword* yang ditampilkan sistem, kemudian sistem melakukan validasi apakah *keyword* sesuai dengan yang terdapat dalam *database*. Jika sesuai maka sistem akan menampilkan hasil pencarian, namun jika tidak sesuai maka user harus mengisi *keyword* yang sesuai. *Activity diagram* proses mencari lokasi tempat ibadah dapat dilihat pada Gambar 3.10 sebagai berikut:



Gambar 3.10 Activity Diagram Proses Mencari Lokasi Tempat Ibadah

7. Proses melihat jarak / rute lokasi tempat ibadah

Pada sistem pemetaan tempat ibadah ini, *user* dan admin dapat melakukan aktivitas melihat jarak atau rute lokasi tempat ibadah. Terlebih dahulu sistem akan menampilkan peta, kemudian pengguna dapat memilih salah satu lokasi tempat ibadah yang ditampilkan sistem, kemudian sistem akan menampilkan jarak atau rute menuju lokasi tempat ibadah. *Activity diagram* proses melihat jarak/rute lokasi tempat ibadah dapat dilihat pada Gambar 3.11 sebagai berikut:

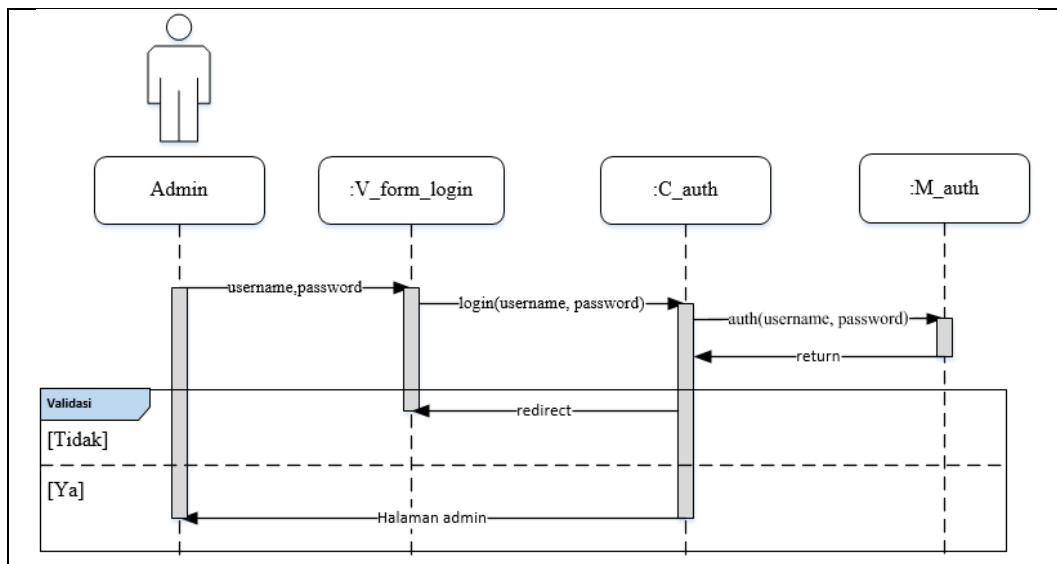


Gambar 3.11 Activity Diagram Proses Melihat Jarak Lokasi Tempat Ibadah

3.2.3.5 Sequence Diagram

Berikut merupakan *sequence diagram* dari Sistem Informasi Geografis Pemetaan Tempat Ibadah di Kota Mataram:

1. Proses *login admin*

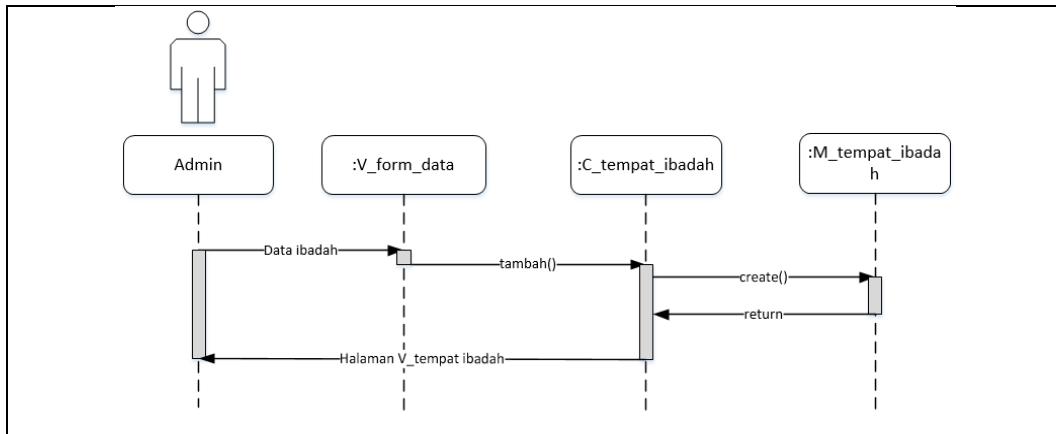


Gambar 3.12 Sequence Diagram Proses Login Admin

Pada Gambar 3.12 merupakan *sequence diagram* proses *login admin*, proses *login* dimulai ketika *admin* memasukkan *username* dan *password* halaman “*V_form_login*”. Kemudian hasil *input form* tersebut akan dikirim dengan mengakses fungsi “*login()*” pada *class* “*C_auth*” yang ada di *Controller* untuk diperiksa dan dilakukan validasi apakah *username* dan *password* yang dimasukkan sesuai dengan yang ada pada *database* atau tidak dengan menjalankan fungsi

“auth()” yang ada di *Model*. Jika ada dan sesuai dengan data yang ada pada *database*, maka akan diarahkan ke *View* atau halaman “V_halaman_admin”, jika tidak sesuai maka akan diarahkan ke *View* atau halaman “V_form_login”.

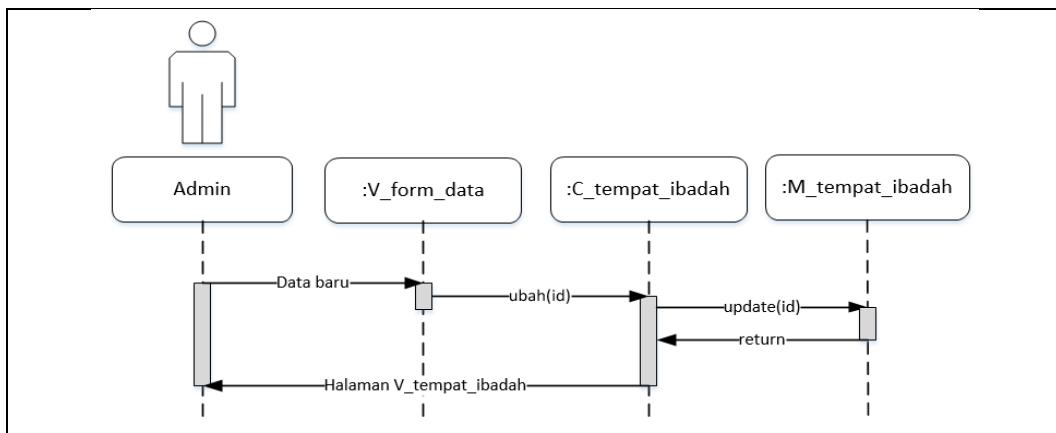
2. Proses menambah data tempat ibadah



Gambar 3.13 Sequence Diagram Proses Menambah Data Tempat Ibadah

Pada Gambar 3.13 merupakan *sequence diagram* proses menambah data tempat ibadah, proses tambah data tempat ibadah dimulai ketika *admin* memasukkan data tempat ibadah pada *View* atau halaman “V_form_data”. Kemudian hasil *input form* tersebut akan dikirim dengan mengakses fungsi “tambah()” pada *class* “C_tempat_ibadah” yang ada di *Controller*, kemudian dari *class* “C_tempat_ibadah” menjalankan fungsi “create()” yang ada di *class* “M_tempat_ibadah” pada *Model* yang berfungsi untuk menyimpan data tempat ibadah ke dalam *database*. Setelah data tersimpan admin akan langsung diarahkan ke halaman “V_tempat_ibadah”.

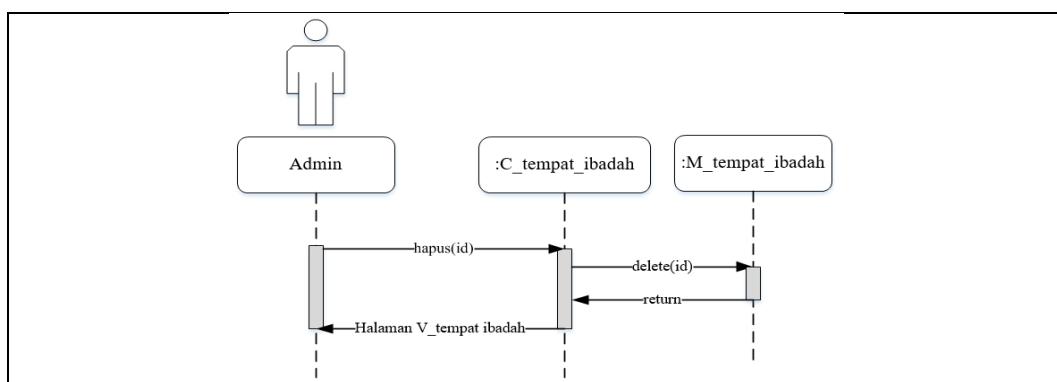
3. Proses mengubah data tempat ibadah



Gambar 3.14 Sequence Diagram Proses Mengubah Data Tempat Ibadah

Pada Gambar 3.14 merupakan *sequence diagram* proses mengubah data tempat ibadah, proses mengubah data tempat ibadah dimulai ketika *admin* memasukkan data tempat ibadah yang baru pada *View* atau halaman “*V_form_data*”. Kemudian hasil *input form* tersebut akan dikirim dengan mengakses fungsi “*ubah()*” pada *class* “*C_tempat_ibadah*” yang ada di *Controller*, kemudian dari *class* “*C_tempat_ibadah*” menjalankan fungsi “*update()*” yang ada di *class* “*M_tempat_ibadah*” pada *Model* yang berfungsi untuk menyimpan data tempat ibadah yang baru ke dalam *database* dan setelah tersimpan admin akan diarahkan ke halaman “*V_tempat_ibadah*”.

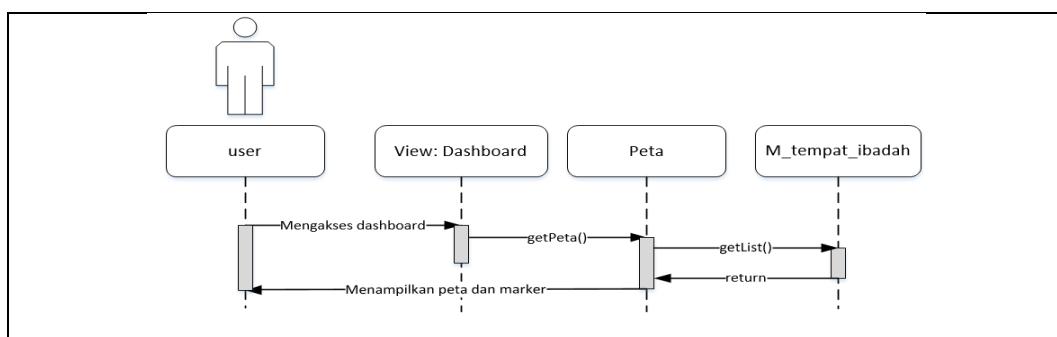
4. Proses menghapus data tempat ibadah



Gambar 3.15 Sequence Diagram Proses Menghapus Data Tempat Ibadah

Pada Gambar 3.15 merupakan *sequence diagram* proses menghapus data tempat ibadah, proses menghapus data tempat ibadah dimulai ketika *admin* mengakses fungsi “*hapus()*” pada *class* “*C_tempat_ibadah*” yang ada di *Controller*, kemudian data akan terhapus dari dalam *database* melalui fungsi “*deleted()*” pada *class* “*M_tempat_ibadah*” yang ada di *Model*. Setelah data terhapus, admin akan diarahkan ke halaman “*V_tempat_ibadah*”.

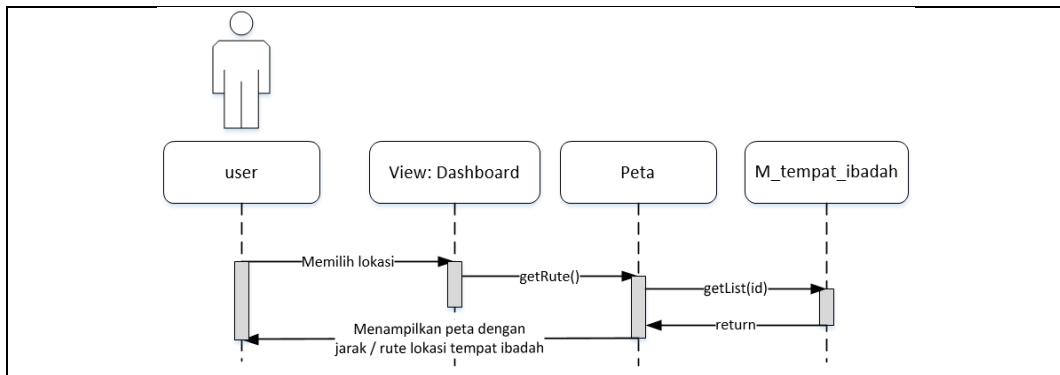
5. Proses melihat lokasi tempat ibadah



Gambar 3.16 Sequence Diagram Proses Melihat Lokasi Tempat Ibadah

Pada Gambar 3.16 merupakan *sequence diagram* proses melihat lokasi tempat ibadah, proses melihat lokasi tempat ibadah dimulai ketika *user* pertama kali masuk ke aplikasi setelah menekan tombol “map” dan memanggil fungsi “getPeta()” yang ada di *class Peta* dan “getList()” yang ada di *class M_tempat_ibadah* untuk mengambil data detail tempat ibadah beserta *longitude* dan *latitude*, kemudian sistem akan menampilkan peta digital beserta dengan marker atau lokasi tempat ibadah yang ada di Kota Mataram.

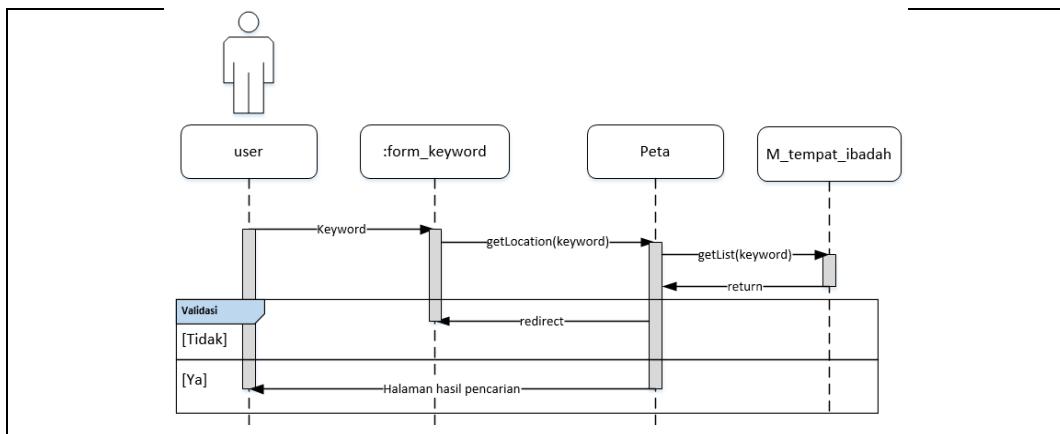
6. Proses melihat jarak/rute lokasi tempat ibadah



Gambar 3. 17 Sequence Diagram Proses Melihat Jarak/Rute Lokasi Tempat Ibadah

Pada Gambar 3.17 merupakan *sequence diagram* proses melihat jarak/rute lokasi tempat ibadah, proses melihat jarak lokasi tempat ibadah dimulai ketika *user* masuk ke aplikasi setelah menekan tombol “map” dan memilih salah satu lokasi tempat ibadah yang ada, sistem akan memanggil fungsi “getRute()” yang ada di *class Peta* dan “getList(id)” yang ada di *class M_tempat_ibadah* yang berfungsi untuk mengambil *longitude* dan *latitude* dan sistem akan menampilkan rute menuju lokasi tempat ibadah yang dipilih.

7. Proses mencari lokasi tempat ibadah

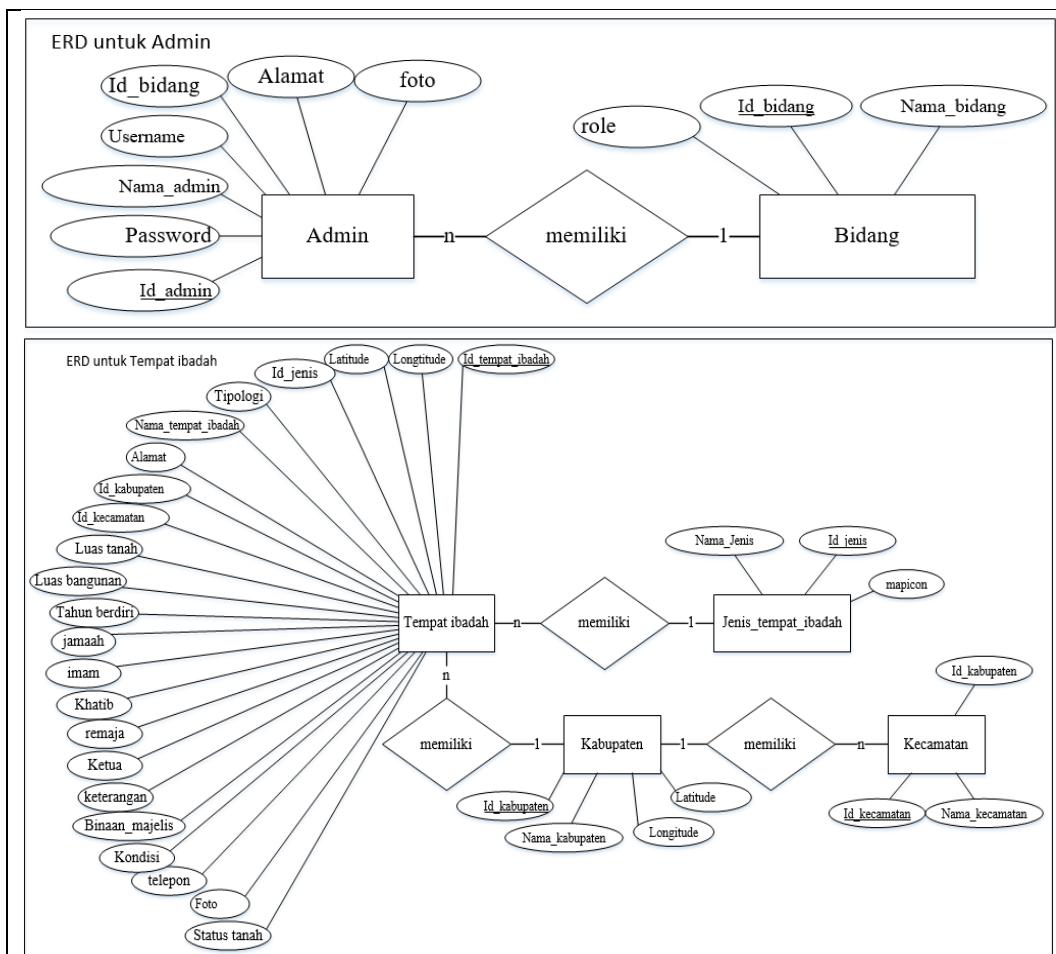


Gambar 3. 18 Sequence Diagram Proses Mencari Lokasi Tempat Ibadah

Pada Gambar 3.18 merupakan *sequence diagram* proses mencari lokasi tempat ibadah, proses mencari lokasi tempat ibadah dimulai ketika *user* memasukkan *keyword* dan menekan tombol “search”, kemudian sistem akan memanggil fungsi “getLocataion(keyword)” yang ada di *class* Peta dan “getList(keyword)” yang ada di *class* M_tempat_ibadah. Jika terdapat kesesuaian *keyword* yang dimasukkan dengan yang ada di *database* maka sistem akan menampilkan hasil pencarian, jika tidak *user* diarahkan untuk memasukkan *keyword* yang sesuai.

3.2.3.6 Entity Relationship Diagram

Berikut merupakan *Entity Relationship Diagram* (ERD) dari Sistem Informasi Geografis Pemetaan Tempat Ibadah di Kota Mataram dapat dilihat pada Gambar 3.19:



Gambar 3. 19 Entity Relationship Diagram Sistem

Berdasarkan ERD pada Gambar 3.19 maka didapatkan tabel-tabel *database* sebagai berikut:

Tabel 3.4 Tabel Admin

No.	Nama Field	Type	Keterangan
1.	<u>Id_admin</u>	Int(5)	<i>Primary Key</i>
2.	<i>Username</i>	Varchar(10)	
3.	<i>Password</i>	Varchar(10)	
4.	Nama_admin	Varchar(30)	
5	Alamat	Varchar(30)	
6.	Id_bidang	int(5)	<i>Foreign key</i>
7	Foto	Varchar(30)	

Tabel 3.5 Tabel Bidang

No.	Nama Field	Type	Keterangan
1.	<u>Id_bidang</u>	Int(5)	<i>Primary Key</i>
2.	Nama_bidang	Varchar(10)	
3.	Role	Int(3)	

Tabel 3. 6 Tabel Tempat Ibadah

No.	Nama Field	Type	Keterangan
1.	<u>Id_tempat_ibadah</u>	Int(5)	<i>Primary Key</i>
2.	Id_Jenis	Varchar(5)	<i>Foreign Key</i>
3.	Nama_tempat_ibadah	Varchar(30)	
4.	Alamat_tempat_ibadah	Varchar(100)	
5.	Id_Kecamatan	Varchar(5)	<i>Foreign Key</i>
6.	Foto_tempat_ibadah	Varchar(100)	
7.	<i>Latitude</i>	Varchar(25)	
8.	<i>Longitude</i>	Varchar(25)	
9.	Telepon	Varchar(15)	
10.	Id_kabupaten	Int(3)	<i>Foreign Key</i>
11	Luas tanah	Varchar(15)	
12	Status tanah	Varchar(15)	
13	Luas bangunan	Varchar(15)	

14	Tahun berdiri	Varchar(15)	
15	jamaah	Varchar(15)	
16	Imam	Varchar(15)	
17	Khatib	Varchar(15)	
18	Remaja	Varchar(15)	
19	Ketua	Varchar(15)	
20	Keterangan	Varchar(15)	
21	Binaan majelis	Varchar(15)	
22	Kondisi	Varchar(15)	

Tabel 3.7 Tabel Kecamatan

No.	Nama Field	Type	Keterangan
1.	<u>Id_kecamatan</u>	int(5)	<i>Primary Key</i>
2.	Nama_kecamatan	Varchar(10)	
3.	Id_kebupaten	int(5)	<i>Foreign Key</i>

Tabel 3.8 Tabel Jenis Tempat Ibadah

No.	Nama Field	Type	Keterangan
1.	<u>Id_jenis</u>	int(5)	<i>Primary Key</i>
2.	Nama_jenis	Varchar(15)	
3.	Mapicon	Varchar(25)	

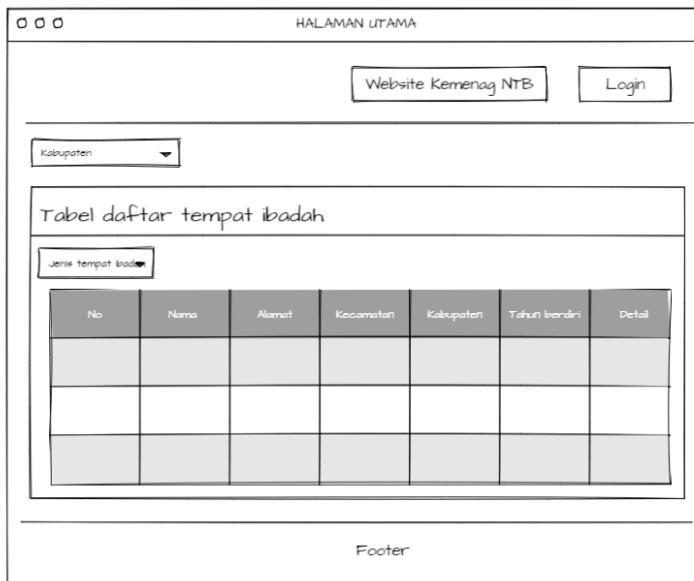
Tabel 3.9 Tabel Kabupaten

No.	Nama Field	Type	Keterangan
1.	<u>Id_kabupaten</u>	int(5)	<i>Primary Key</i>
2.	Nama_kabupaten	Varchar(10)	
3.	Latitude	Varchar(25)	
4.	Longitude	Varchar(25)	

3.2.3.7 Desain *interface*

Setelah melakukan analisis terhadap kebutuhan dan perancangan sistem, tahapan selanjutnya dari penelitian Tugas Akhir ini yaitu merancangan *interface* dari sistem yang akan dibuat. Berikut ini adalah rancangan *interface* dari sistem yang akan dibuat:

1. Halaman utama sistem pada web



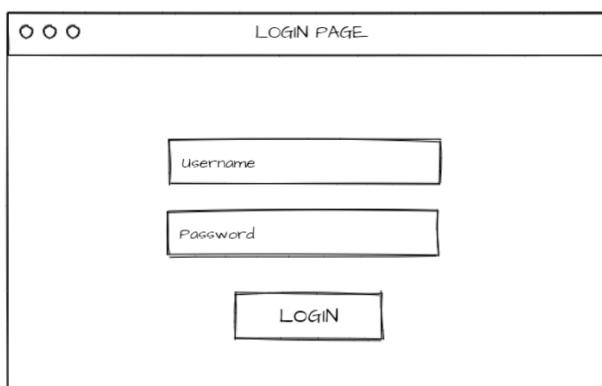
Rancangan tampilan halaman utama pada web yang terdiri dari:

- Header: HALAMAN UTAMA, Website Kemenag NTB, Login.
- Form dropdown: Kabupaten.
- Tabel daftar tempat ibadah:
 - Kolom: No, Nama, Alamat, Kecamatan, Kabupaten, Tahun berdiri, Detail.
 - Baris: 5 baris kosong.
- Footer.

Gambar 3.20 Rancangan Tampilan Halaman Utama Pada Web

Pada Gambar 3.20 merupakan rancangan halaman utama dalam web. Pada halaman basis web ini pengguna langsung dapat melihat daftar dan detail tempat ibadah yang tersusun dalam bentuk tabel. Dan terdapat menu login untuk admin login dan menu untuk melihat *website* kemenag NTB.

2. Halaman *login* admin



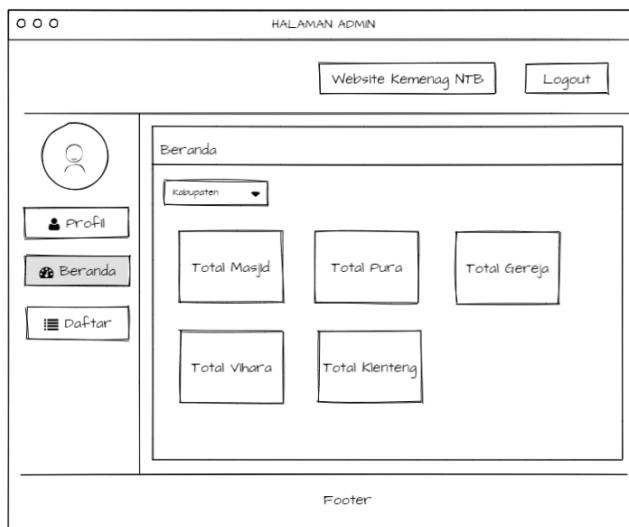
Rancangan tampilan halaman *login* admin yang terdiri dari:

- Header: LOGIN PAGE.
- Form input: Username, Password.
- Form button: LOGIN.

Gambar 3.21 Rancangan Tampilan Halaman *Login*

Pada Gambar 3.21 merupakan rancangan untuk halaman *login* yang akan ditampilkan pada sistem basis web. Pada halaman ini, jika admin ingin mengakses sistem maka harus mengisi *username* dan *password* yang sesuai dengan yang terdapat dalam *database*.

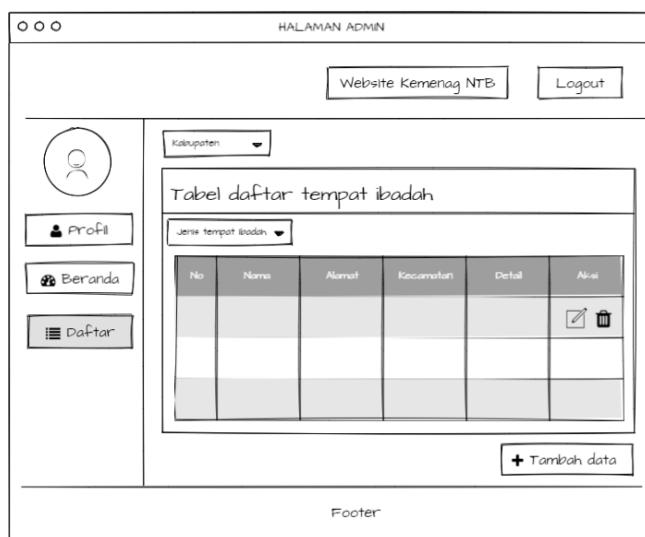
3. Halaman beranda admin



Gambar 3.22 Rancangan Tampilan Halaman Beranda Admin

Pada Gambar 3.22 merupakan rancangan untuk halaman beranda yang akan ditampilkan pada sistem basis web pada saat super admin dan admin berhasil melakukan *login*. Pada halaman ini admin dapat melihat jumlah tempat ibadah yang ada berdasarkan kabupaten.

4. Halaman kelola data tempat ibadah



Gambar 3.23 Rancangan Tampilan Halaman Kelola Data Tempat Ibadah

Pada Gambar 3.23 merupakan rancangan untuk halaman kelola data tempat ibadah yang akan ditampilkan pada sistem basis web. Pada halaman ini, super admin dan admin dapat melihat data tempat ibadah yang tersusun dalam bentuk tabel, serta dapat mengelola data tersebut seperti menambahkan data baru, mengubah data, dan menghapus data tempat ibadah yang ada, termasuk mengatur titik koordinat / *longitude* dan *latitude* lokasi tempat ibadah yang ada.

5. Halaman tambah data tempat ibadah

Rancangan tampilan halaman tambah data tempat ibadah. Tampilan ini terdiri dari dua bagian utama: sidebar kiri dan konten utama.

- Bagian Kanan (Konten Utama):** Terdapat judul "HALAMAN ADMIN" di bagian atas. Di bawahnya, terdapat tombol "Website Kemenag NTB" dan "Logout".
Selanjutnya, terdapat form input dengan judul "Form input data tempat ibadah". Di bagian bawah form tersebut terdapat tombol "+ Tambah data".
- Bagian Kiri (Sidebar):** Terdapat ikon profil berbentuk lingkaran dengan simbol manusia di dalamnya.
 - Tombol "Profil" (ikon pengguna).
 - Tombol "Beranda" (ikon rumah).
 - Tombol "Daftar" (ikon daftar).
- Footer:** Terdapat tulisan "Footer" di bagian bawah.

Gambar 3.24 Rancangan Tampilan Halaman Tambah Data Tempat Ibadah

Pada Gambar 3.24 merupakan rancangan untuk halaman tambah data tempat ibadah yang akan ditampilkan pada sistem basis web. Pada halaman ini, super admin dan admin dapat menambahkan data tempat ibadah ke dalam system.

6. Halaman edit data tempat ibadah

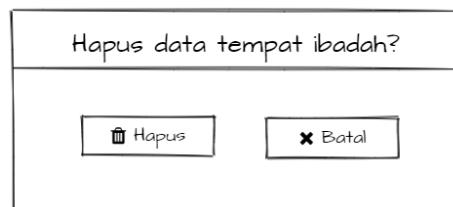
Rancangan tampilan halaman edit data tempat ibadah. Tampilan ini terdiri dari dua bagian utama: sidebar kiri dan konten utama.

- Bagian Kanan (Konten Utama):** Terdapat judul "HALAMAN ADMIN" di bagian atas. Di bawahnya, terdapat tombol "Website Kemenag NTB" dan "Logout".
Selanjutnya, terdapat form edit dengan judul "Form edit data tempat ibadah". Di bagian bawah form tersebut terdapat tombol "Edit data".
- Bagian Kiri (Sidebar):** Terdapat ikon profil berbentuk lingkaran dengan simbol manusia di dalamnya.
 - Tombol "Profil" (ikon pengguna).
 - Tombol "Beranda" (ikon rumah).
 - Tombol "Daftar" (ikon daftar).
- Footer:** Terdapat tulisan "Footer" di bagian bawah.

Gambar 3.25 Rancangan Tampilan Halaman Edit Data Tempat Ibadah

Pada Gambar 3.25 merupakan rancangan untuk halaman edit data tempat ibadah yang akan ditampilkan pada sistem basis web. Pada halaman ini, super admin dan admin dapat mengubah data tempat ibadah yang ada di dalam sistem.

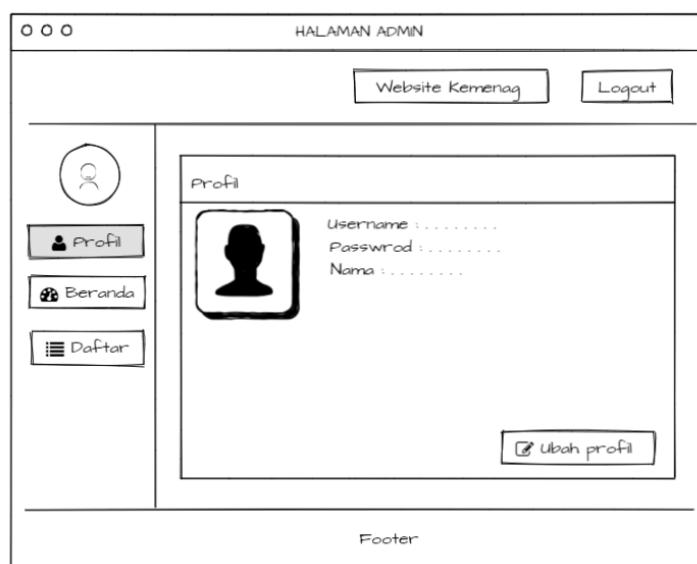
7. Tampilan hapus data tempat ibadah



Gambar 3.26 Rancangan Tampilan Hapus Data Tempat Ibadah

Pada Gambar 3.26 merupakan rancangan untuk tampilan hapus data tempat ibadah yang akan ditampilkan pada sistem basis web. Pada halaman ini, super admin dan admin dapat menghapus data tempat ibadah yang ada di dalam sistem.

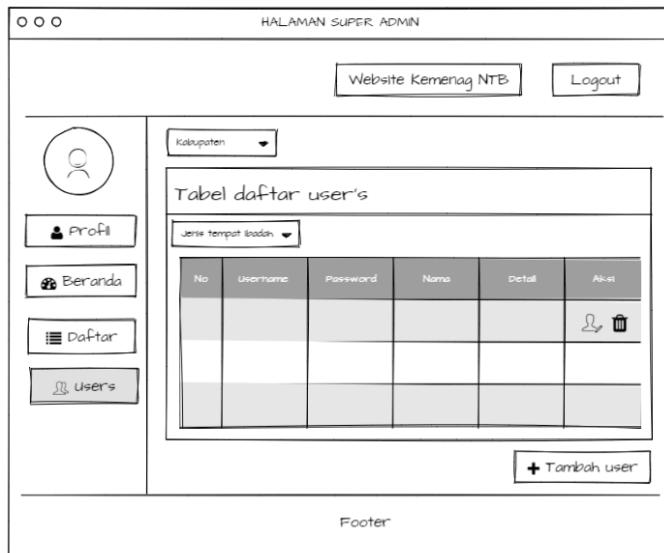
8. Halaman profil admin



Gambar 3.27 Rancangan Tampilan Profil Admin

Pada Gambar 3.27 merupakan rancangan untuk halaman profil admin yang akan ditampilkan pada sistem basis web. Pada halaman ini admin dapat melihat serta dapat mengubah data profil admin itu sendiri.

9. Halaman kelola data admin

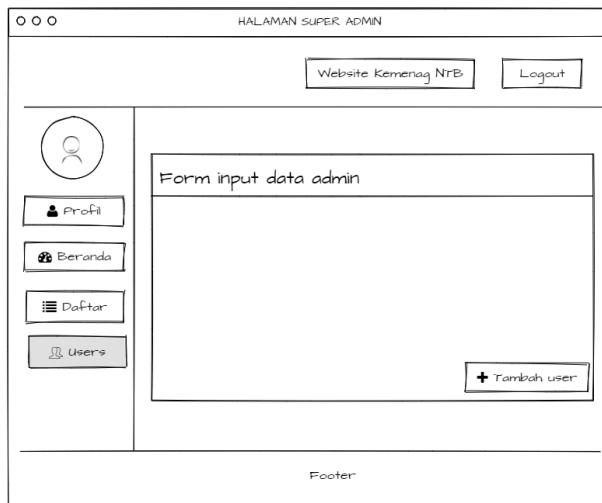


Rancangan tampilan halaman kelola data admin. Halaman ini dibuka dengan judul "HALAMAN SUPER ADMIN". Di bagian atas, terdapat tombol "Website Kemenag NRB" dan "Logout". Di sebelah kiri, terdapat sidebar dengan ikon profil dan menu: "Profil", "Beranda", "Daftar", dan "Users". Di bagian tengah, terdapat form "Tabel daftar user's" yang menunjukkan tabel dengan kolom: No, Username, Password, Nama, Detail, dan Aksi. Di bawah tabel, terdapat tombol "+ Tambah user". Di bagian bawah, terdapat tulisan "Footer".

Gambar 3.28 Rancangan Tampilan Halaman Kelola Data Admin

Pada Gambar 3.28 merupakan rancangan untuk halaman kelola data admin yang akan ditampilkan pada sistem basis web. Pada halaman ini, super admin dapat melihat data admin yang tersusun dalam bentuk tabel, serta dapat mengelola data admin tersebut seperti menambahkan data baru, mengubah data, dan menghapus data admin yang ada.

10. Halaman tambah data admin

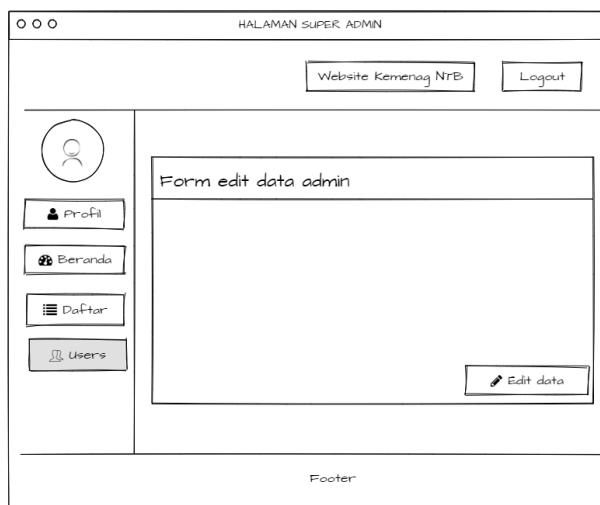


Rancangan tampilan halaman tambah data admin. Halaman ini dibuka dengan judul "HALAMAN SUPER ADMIN". Di bagian atas, terdapat tombol "Website Kemenag NRB" dan "Logout". Di sebelah kiri, terdapat sidebar dengan ikon profil dan menu: "Profil", "Beranda", "Daftar", dan "Users". Di bagian tengah, terdapat form "Form input data admin" yang menunjukkan area kosong untuk input data. Di bawah form, terdapat tombol "+ Tambah user". Di bagian bawah, terdapat tulisan "Footer".

Gambar 3.29 Rancangan Tampilan Halaman Tambah Data Admin

Pada Gambar 3.29 merupakan rancangan untuk halaman tambah data admin yang akan ditampilkan pada sistem basis web. Pada halaman ini, hanya super admin yang dapat menambahkan data admin ke dalam system.

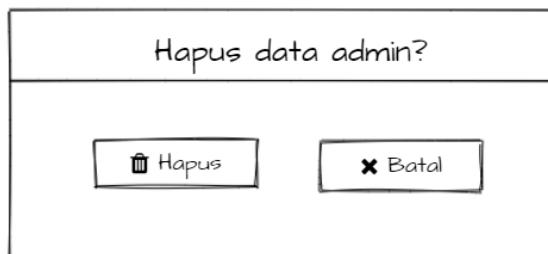
11. Halaman edit data admin



Gambar 3.30 Rancangan Tampilan Halaman Edit Data Admin

Pada Gambar 3.30 merupakan rancangan untuk halaman edit data admin yang akan ditampilkan pada sistem basis web. Pada halaman ini, super admin dapat mengubah data admin ke dalam system.

12. Tampilan hapus data admin



Gambar 3.31 Rancangan Tampilan Hapus Data Admin

Pada Gambar 3.31 merupakan rancangan untuk hapus data admin yang akan ditampilkan pada sistem basis web. Pada halaman ini, super admin dapat menghapus data admin yang ada di dalam system.

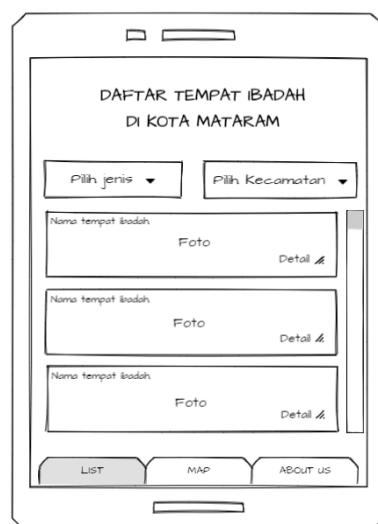
13. Rancangan *map android*



Gambar 3.32 Rancangan Tampilan Halaman Peta Tempat Ibadah

Pada Gambar 3.32 merupakan rancangan untuk mencari lokasi tempat ibadah yang akan ditampilkan pada sistem informasi geografis pemetaan tempat ibadah. Pada halaman ini, *user* dapat melihat daftar lokasi-lokasi tempat ibadah yang divisualisasikan ke dalam bentuk peta kota Mataram. Pada halaman ini, *user* dapat mencari lokasi tempat ibadah berdasarkan jenis tempat ibadah yang diinginkan serta dapat melihat jarak dan gambar rute atau penunjuk arah jalan dari posisi *user* menuju lokasi tempat ibadah yang diinginkan.

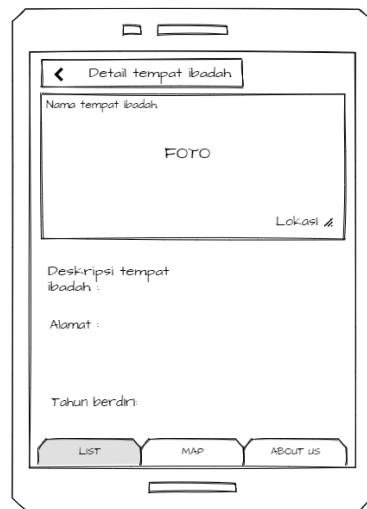
14. Rancangan *list tempat ibadah android*



Gambar 3.33 Rancangan Tampilan *List Tempat Ibadah Android*

Pada Gambar 3.33 merupakan rancangan untuk menampilkan daftar tempat ibadah yang akan ditampilkan pada sistem informasi geografis pemetaan tempat ibadah. Pada halaman ini, *user* dapat melihat daftar-daftar tempat ibadah yang ada di Kota Mataram berdasarkan filter jenis tempat ibadah diinginkan.

15. Rancangan informasi detail tempat ibadah *android*



Gambar 3.34 Rancangan Tampilan Informasi Detail Tempat Ibadah *Android*

Pada Gambar 3.34 merupakan rancangan untuk menampilkan informasi detail tempat ibadah yang akan ditampilkan pada sistem informasi geografis pemetaan tempat ibadah. Pada halaman ini, *user* dapat melihat informasi detail dari tempat ibadah seperti nama, foto, alamat, luas tanah dsb.

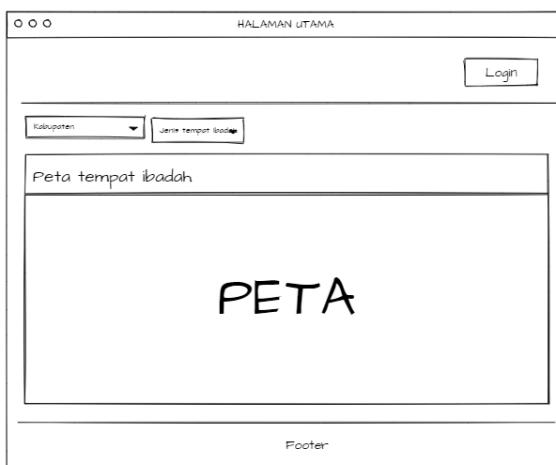
16. Rancangan *about us*



Gambar 3.35 Rancangan Tampilan About Us *Android*

Pada Gambar 3.35 merupakan rancangan untuk menampilkan informasi *about us* yang akan ditampilkan pada sistem informasi geografis pemetaan tempat ibadah. Pada halaman ini, *user* dapat melihat informasi terkait deskripsi dari aplikasi dan dapat menghubungi email jika terdapat pertanyaan atau masukkan terkait dengan sistem, sehingga admin dan *user* dapat saling berinteraksi.

17. Rancangan peta basis web



Gambar 3.36 Rancangan Tampilan Halaman Peta Tempat Ibadah Pada Basis Web

Pada Gambar 3.36 merupakan rancangan untuk mencari lokasi tempat ibadah yang akan ditampilkan pada sistem informasi geografis pemetaan tempat ibadah basis web. Pada halaman ini, *user* dapat melihat daftar lokasi-lokasi tempat ibadah yang divisualisasikan ke dalam bentuk peta kota Mataram. Pada halaman ini, *user* dapat mencari lokasi tempat ibadah berdasarkan jenis tempat ibadah dan kabupaten yang diinginkan serta dapat melihat jarak dan gambar rute atau penunjuk arah jalan dari posisi *user* menuju lokasi tempat ibadah yang diinginkan.

18. Rancangan import data tempat ibadah

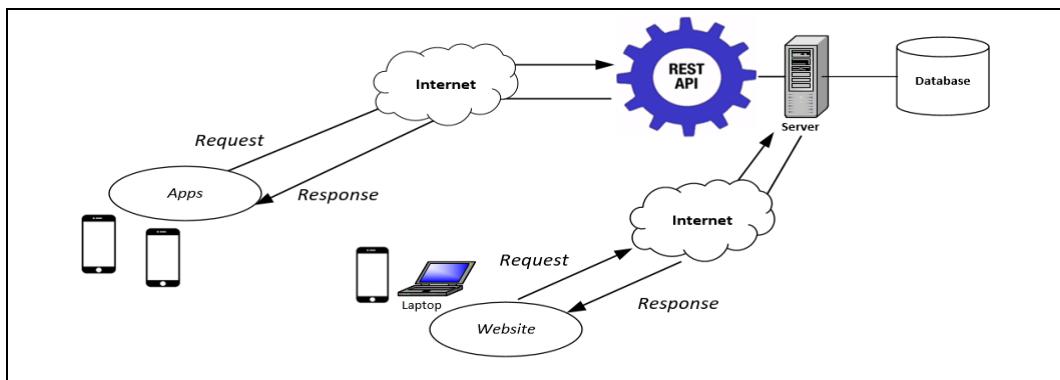
A wireframe diagram of a form titled "Import data tempat ibadah". It has two dropdown menus: "pilih Kabupaten" and "pilih file excel". At the bottom are two buttons: "+ import" and "x Batal".

Gambar 3.37 Rancangan Tampilan Import Data Tempat Ibadah Pada Basis Web

Pada Gambar 3.37 merupakan rancangan untuk *import* data tempat ibadah yang akan ditampilkan pada system basis web. Pada halaman ini, admin dapat memilih kabupaten dan memasukkan file berupa excel yang untuk dapat mengimportnya data ke dalam *database*.

3.2.3.8 Desain arsitektur sistem

Berikut merupakan desain arsitektur sistem dari Sistem Informasi Geografis Pemetaan Tempat Ibadah di Kota Mataram dapat dilihat pada Gambar 3.38:



Gambar 3.38 Arsitektur Sistem

Pada Gambar 3.38 merupakan desain arsitektur sistem yang akan dikembangkan. Pada basis *android* dapat diakses melalui *multi-device*, dimana *device* yang mengakses sistem mengirimkan *request* atau permintaan melalui jaringan internet dan langsung terhubung ke server kemudian akan mengakses database dan mengembalikan *response* berupa html.

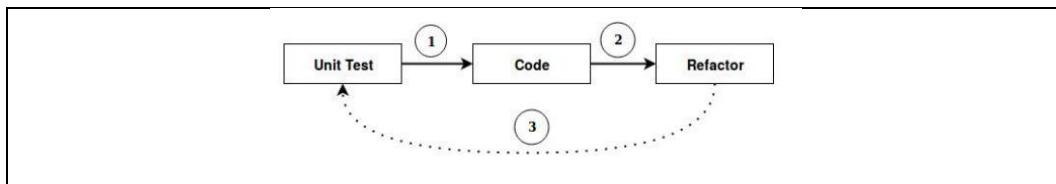
Pada basis *application* hanya dapat diakses melalui *android*, dimana mengirimkan *request* melalui jaringan internet terhadap layanan API yang tersedia, setelah terhubung maka server akan mengakses database dan mengembalikan *response*.

3.2.4 Coding

Setelah mengetahui gambaran fungsi-fungsi kebutuhan dari sistem yang akan dibuat, tahapan selanjutnya adalah proses implementasi rancangan sistem yang sudah dilakukan ke dalam bentuk bahasa pemrograman (*coding*). Proses pengkodingan pada pembuatan sistem penelitian Tugas Akhir ini dilakukan oleh penulis sendiri karena mengacu pada karakteristik dari Personal Extreme Programming (PXP) yang dimana proses pengkodingan hanya dilakukan oleh programmer tunggal[7]. Pembuatan untuk basis *web* pada sistem penelitian Tugas

Akhir ini dibangun dengan bahasa pemrograman PHP menggunakan *framework* *Code Igniter* sebagai *backend* dan *template bootstrap* untuk *frontend*. Sedangkan untuk basis *android* dibangun dengan bahasa pemrograman Kotlin. Adapun teknologi tambahan yang digunakan yaitu *google maps API* digunakan sebagai pembuatan peta, MySQL digunakan untuk manajemen *database*, dan Postman digunakan sebagai *tester* terhadap penyedia layanan API.

Pada fase *coding* dalam XP terdapat pengujian kode program menggunakan teknik *Test Driven Development* (TDD) dimana melakukan pengembangan perangkat lunak berdasarkan hasil pengujian yang telah dilakukan, hal ini bermaksud untuk dapat mengurangi waktu pengembangan dan dapat minimalisir adanya *bug* atau *error* pada aplikasi[26]. TTD memiliki alur kerja seperti Gambar 3.39 sebagai berikut:



Gambar 3.39 Alur Kerja TDD

Pengembang menuliskan *unit test* terlebih dahulu, sesuai spesifikasi dan bisnis proses dari sistem yang diinginkan oleh pengguna (*user stories*).

Setelah membuat *unit test*, pengembang menuliskan kode program untuk menyelesaikan *unit test* sampai *pass* tidak ada *error*.

Pengembang melakukan *refactoring* kode program yaitu mengubah struktur program agar lebih mudah dipahami dan dimodifikasi, tanpa harus mengubah dari *behaviour* program[33].

Berikut merupakan contoh dari penggunaan TDD:

1. Menentukan *unit test*

Misal terdapat sebuah contoh kasus seperti login sistem tidak dapat diakses melalui *link*. Dari contoh kasus tersebut kita mendapatkan spesifikasi gambaran tentang nama *function* atau *method*-nya, fungsinya, inputnya dan output yang akan dibuat.

Nama *function* : auth()

Fungsi : validasi untuk dapat mengakses sistem

Input : *username* dan *password*

Output : Hasil proses

Didapatkan *unit test* seperti inputnya *username* : macan dan *password* : macan hasilnya berhasil login, selain itu termasuk mengakses login sistem melalui *link* maka akan menolak akses.

2. Mengimplementasikan *unit test / coding*

Implementasi *code* atau biasa disebut melakukan pengkodingan terdahap *unit test*. Misal berikut implementasi *code* dari *unit test* pada *point 1*

```
1. <?php
2. defined('BASEPATH') OR exit('No direct script access
   allowed');
3. class C_login extends CI_Controller {
4.     public function __construct(){
5.         parent::__construct();
6.     }
7.     public function auth(){
8.     {
9.         $ci = get_instance();
10.
11.        $v_username = $ci->session->userdata('username');
12.        if (!$v_username) {
13.            redirect('c_login');
14.        } else {
15.            $role_bidang = $ci->session-
>userdata('role');
16.            $menu = $ci->uri->segment(1);
17.            if ($role_bidang == 2 && ($menu ==
'c_admin' || $menu == 'c_bidang')) {
18.                redirect('c_login/blocked');
19.            }
20.        }
21.    }
22. }
```

Kode Sumber 3.1 Contoh *Source Code Point 1*

Pada Kode Sumber 3.1 merupakan *source code* contoh dari penerapan pada *point 1* yang diasumsikan tidak ada *error*. Jika di tes dengan mengakses login sistem melalui *link*, maka sistem akan menolak akses.

Pada tahap ini diasumsikan bahwa *code* yang dibuat telah benar dan tidak ada *error / bug*, serta telah memenuhi prasyarat dan siap dilakukan *refactoring*. Sebab jika terdapat *error/bug* pada code tersebut, maka tidak dapat dilakukan *refactoring*, karena *refactoring* hanya dapat dilakukan jika tidak ada *error* pada *code* [33].

3. *Refactoring*

Pada Kode Sumber 3.2 dan 3.3 merupakan contoh bentuk *refactoring* dari pada *source code* 3.1

```

1. <?php
2. function auth() {
3.     $ci = get_instance();
4.     $v_username = $ci->session->userdata('username');
5.     if (!$v_username) {
6.         redirect('c_login');
7.     } else {
8.         $role_bidang = $ci->session->userdata('role');
9.         $menu = $ci->uri->segment(1);
10.        if ($role_bidang == 2 && ($menu ==
11.            'c_admin' || $menu == 'c_bidang')) {
12.                redirect('c_login/blocked');
13.            }
14.    }
15. }

```

Kode Sumber 3. 2 Yang Ada Di File Helper

```

13. <?php
14. defined('BASEPATH') OR exit('No direct script access
   allowed');
15. class C_tempat_ibadah extends CI_Controller {
16.     public function __construct() {
17.         parent::__construct();
18.         auth();
19.     }
20. }

```

Kode Sumber 3. 3 Kode Yang Dieksekusi

Diasumsikan bahwa hasil *refactoring* berjalan dengan benar dan hasilnya outputnya sesuai dengan yang diinginkan yaitu login sistem melalui *link* akan ditolak atau pengguna harus memasukkan *username* dan *password* yang sesuai. Sehingga jika terdapat modul baru, tidak perlu menuliskan ulang *code*, cukup memanggil nama fungsinya saja. *Refactoring* hanya bertujuan untuk merubah struktur *code* supaya lebih singkat dan menghemat memori serta mudah dipahami dan tidak menambahkan fungsionalitas atau tetap sesuai dengan spesifikasi dan tidak merusak fungsi program sebelumnya.

3.2.5 Testing

Pada tahap ini dilakukan pengujian (*testing*) untuk mengetahui sistem telah berjalan sesuai dengan rancangan atau tidak. Pengujian merupakan kegiatan dimana suatu sistem atau komponen dieksekusi dibawah kondisi tertentu, hasilnya diamati atau dicatat untuk kemudian dievaluasi berdasarkan aspek sistem atau komponen[34]. Dalam tahap *testing* pada XP digunakan *acceptance testing* yang berguna untuk mengevaluasi kesesuaian sistem dengan kebutuhan bisnis dan menilai apakah itu dapat diterima. Dalam *acceptance testing* terdapat beberapa pengujian diantaranya yaitu *Alpha* dan *Beta testing*[35]. Pengujian *alpha* merupakan pengujian yang bertujuan untuk memastikan bahwa aplikasi yang diuji

dapat berjalan dengan lancar tanpa gangguan *error* atau *bug*. Pengujian *Beta* merupakan pengujian yang bersifat langsung di lingkungan yang sebenarnya dengan penyebaran kuesioner yang akan dihitung untuk dapat diambil kesimpulan terhadap penilaian aplikasi yang dibangun.

Penekanan pada tahap ini adalah tentang pengujian, pengujian yang dilakukan menggunakan *alpha testing* dengan metode *black box* dan *beta testing* dengan penyebaran kuesioner, dimana pengguna akan menilai sejauh mana aplikasi berjalan sesuai dengan tujuan[34]. Digunakan pengujian *alpha* dan *beta* karena dirasa cocok atau sesuai dengan tujuan untuk mengetahui sistem yang dikembangkan telah sesuai atau tidak dengan kebutuhan *client*.

Pada Gambar 3.21 merupakan rancangan halaman login sistem dimana terdapat beberapa rencana pengujian. Pada rencana pengujian *username* dan *password*, data akan valid jika *username* atau *password* diisi dengan benar dan tidak boleh kosong. Sebaliknya, data tidak akan valid jika diisi dengan tidak benar atau kosong. Berikut merupakan rencana pengujian pada login:

Tabel 3. 10 Rencana Pengujian Pada *Login*

Id	Skenario pengujian	Test case	Hasil yang diharapkan
L01	<i>Username</i> dan <i>password</i> tidak diisi kemudian klik tombol login	Username : (kosong) Password: (kosong)	Sistem akan menolak akses dan menampilkan “Form <i>username</i> dan <i>password</i> harus diisi!”.
L02	Mengetik <i>username</i> dan <i>password</i> tidak diisi atau kosong kemudian klik tombol login	Username : (macan) Password: (kosong)	Sistem akan menolak akses dan menampilkan “Form <i>password</i> harus diisi!”.
L03	<i>Username</i> tidak diisi dan <i>password</i> diisi kemudian klik tombol login	Username : (kosong) Password: (123)	Sistem akan menolak akses dan menampilkan “Form <i>username</i> harus diisi!”.
L04	<i>Username</i> dan <i>password</i> diisi tidak benar kemudian klik tombol login	Username : (salah) Password:	Sistem akan menolak akses dan menampilkan

		(salah)	“Username dan password salah!”.
L05	<i>Username</i> dan <i>password</i> diisi dengan benar kemudian klik tombol login	Username : (benar) Password: (benar)	Sistem menerima akses login dan kemudian langsung menampilkan halaman admin

Pada Gambar 3.28 merupakan rancangan halaman kelola data admin sistem dimana terdapat beberapa rencana pengujian menambah data admin. Pada rencana pengujian *username* atau *password*, data akan valid jika *username* atau *password* diisi atau tidak boleh kosong. Sebaliknya, data tidak valid jika data kosong. Pada rencangan pengujian menambah *username* baru juga data akan berhasil ditambahkan jika *username* belum terdaftar dalam sistem, jika sudah terdaftar tidak berhasil ditambahkan. Rencana pengujian ini juga berlaku terhadap mengubah data admin. Berikut merupakan rencana pengujian pada penambahan data admin:

Tabel 3. 11 Rencana Pengujian Tambah Data Admin

Id	Skenario pengujian	<i>Test case</i>	Hasil yang diharapkan
T01	<i>Username</i> dan <i>password</i> tidak diisi kemudian klik tombol tambah	Username: (kosong) Password: (kosong)	Sistem gagal menambahkan data admin dan menampilkan “Form <i>username</i> dan <i>password</i> harus diisi!”.
T02	Mengisi <i>username</i> dan <i>password</i> tidak diisi atau kosong kemudian klik tombol tambah	Username: (macan) Password: (kosong)	Sistem gagal menambahkan data admin dan menampilkan “Form <i>password</i> harus diisi!”.
T03	<i>Username</i> tidak diisi dan <i>password</i> diisi kemudian klik tombol tambah	Username: (kosong) Password: (123)	Sistem gagal menambahkan data admin dan menampilkan “Form <i>username</i> harus diisi!”.
T04	<i>Username</i> yang diisi sudah terdaftar dalam <i>database</i>	Username: (terisi = database)	Sistem gagal menambahkan data admin dan

			menampilkan “ <i>Username</i> telah terdaftar!”.
T05	<i>Username</i> dan <i>password</i> diisi dengan benar kemudian klik tombol tambah, dan <i>username</i> belum terdaftar dalam <i>database</i>	Username: (terisi) Password: (terisi)	Sistem berhasil menambahkan data admin dan kemudian langsung menampilkan daftar admin yang ada.

Digunakan 33 responden dalam melakukan kuesioner pada penelitian Tugas Akhir ini, sebagaimana dikemukakan oleh Cohen, semakin besar *sample* dari besarnya populasi yang ada adalah semakin baik, akan tetapi ada jumlah batas minimal yang harus diambil oleh peneliti yaitu sebanyak 30 *sample* [36]. Penulis membagi responden menjadi masyarakat 31 orang dan 2 orang pegawai Kemenag bidang Inmas dan Bimas Islam yang berperan sebagai admin sistem yang akan menggunakannya. Tujuan dari kuesioner ini adalah menilai apakah aplikasi berjalan sesuai dengan tujuan.

Berikut merupakan tabel skor penilaian dapat dilihat pada Tabel 3.12 berikut:

Tabel 3. 12 Skor Penilaian

Tingkat kepuasan	Nilai
Sangat setuju	5
Setuju	4
Cukup	3
Tidak setuju	2
Sangat tidak setuju	1

Untuk melakukan perhitungan kuesioner digunakan rumus:

$$\text{Persentase} = J / N *100 \%$$

Keterangan :

J : Total nilai jawaban yang diberikan pengguna pada setiap nomor.

N : Jumlah pengguna

Selanjutnya dilakukan proses perhitungan persentase rata-rata dari setiap poin jawaban, perhitungan yang digunakan adalah menggunakan rumus sebagai berikut:

$$\text{Rata-rata} = (\text{Persentase ke-1} + \dots + \text{Persentase ke-6}) / 6$$

3.2.6 *Release*

Langkah terakhir yang dilakukan setelah dilakukannya pengujian sistem yaitu penerapan sistem (*release*). Sistem dapat dikatakan *release* jika sistem sudah berjalan atau sesuai dengan kebutuhan pengguna berdasarkan dari penjabaran *user stories* yang didapat pada Tabel 3.1. Namun, aplikasi tetap harus ada dalam pendampingan jika suatu saat terdapat *bug* atau pembaharuan terhadap sistem.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi metode *Extreme Programming*

Dalam proses pengembangan sistem informasi geografis untuk pemetaan lokasi tempat ibadah ini digunakan metode *extreme programming* seperti yang telah dipaparkan sebelumnya pada diagram alir penelitian pada Gambar 3.1, guna untuk mengetahui pengembangan sistem mencapai hasil yang diinginkan oleh *client* atau tidak. Pembahasan yang akan dijelaskan pada sub bab ini sama halnya dengan siklus metode *extreme programming* yaitu meliputi *planning*, *design*, *coding* dan *testing*.

4.1.1 *Planning*

Pada tahap *planning* ini sebelumnya pengembang telah melakukan diskusi dengan *client* untuk mendapatkan *user stories* yang dimana pada kasus penelitian tugas akhir ini pihak kemenag NTB merupakan *client*. Tujuan dari *user stories* ini yaitu untuk mendapatkan gambaran tentang bagaimana system yang dibuat berjalan. Didapatkan total 11 *user stories* dari *client* yang telah dipaparkan sebelumnya dapat dilihat pada Tabel 3.1.

Pada tahap ini juga pengembang dan *client* telah melakukan diskusi terhadap penentuan iterasi yang terdiri dari *user stories* yang akan dikerjakan terlebih dahulu. Dimana guna penentuan iterasi ini yaitu untuk mengetahui prioritas *user stories* dan estimasi waktu penggerjaan. Didapatkan 3 iterasi dan tiap iterasi dikerjakan selama 10 hari, sehingga secara keseluruhan rencana system akan dikerjakan selama 30 hari. Rincian penentuan iterasi tersebut dapat dilihat pada Tabel 3.2.

Selama proses iterasi berlangsung terdapat beberapa permintaan penambahan user stories dari *client* seperti yang dirincikan pada Tabel 4.1 berikut.:

Tabel 4.1 Penambahan *User Stories*

No	Kode US	Judul	Deskripsi	Acceptance Criteria	Estimasi (hari)
1	US-12	Profile	Sebagai admin, dapat melihat informasi dan mengubah data diri	Terdapat halaman profile admin yang telah login ke dalam system serta dapat mengubah profile tersebut.	4
2	US-13	Beranda	Sebagai admin, dapat melihat halaman beranda dengan total tempat ibadah	Terdapat halaman beranda untuk admin yang dapat melihat total dari masing – masing jenis tempat ibadah	2
3	US-14	Import data excel tempat ibadah	Sebagai admin, dapat menambahkan data tempat ibadah dengan cara memasukkan file excel yang berisi data tempat ibadah, agar mempermudah penginputan data dalam jumlah yang banyak	Terdapat form untuk menginputkan data dalam bentuk file excel, dan pemilihan kabupaten.	5

4	US-15	Unduh template format excel import data	Sebagai admin, dapat mendownload format excel untuk import data tempat ibadah, sehingga dapat menyesuaikan kolom – kolom masukan data yang diinginkan sistem	Terdapat tombol untuk mengunduh format data tempat ibadah dalam bentuk file excel	2
5	US-16	Export data excel tempat ibadah	Sebagai admin, dapat melakukan export excel, sehingga dapat melakukan pembacaan data tempat ibadah yang ada dalam bentuk <i>hardcopy</i>	Terdapat tombol untuk mengexport data tempat ibadah dalam bentuk file excel.	3
6	US-17	Melihat peta lokasi tempat ibadah pada basis web	Sebagai masyarakat, dapat melihat peta lokasi tempat ibadah pada basis web, namun terdapat filter kabupaten yang ada di provinsi NTB. Sehingga masyarakat juga mengetahui peta lokasi yang ada di	Terdapat peta lokasi tempat ibadah pada basis web, yang tidak hanya dapat menampilkan di kabupaten Kota Mataram, melainkan di kabupaten yang ada pada provinsi NTB juga.	8

			kabupaten selain di NTB		
7	US-18	Melihat rute lokasi tempat ibadah pada basis web	Sebagai masyarakat, dalam melihat rute lokasi menuju tempat ibadah pada basis web	Terdapat tombol rute menuju lokasi tempat ibadah	2

Penambahan *user stories* pada Tabel 4.1 tersebut menyebabkan pengembang bersama dengan *client* melakukan perencanaan ulang terhadap penyesuaian pengerjaan tiap iterasi. Hasil perencanaan iterasi dapat dilihat pada Tabel 4.2 sebagai berikut.

Tabel 4.2 Hasil Perencanaan Iterasi

No	Kode US	Deskripsi	Estimasi (hari)
<i>Iteration – 1</i>			
1	US-01	Login ke dalam system dan mengakhiri sistem	3
2	US-12	Profile admin <i>login</i>	4
3	US-02	Melihat data admin	3
<i>Velocity</i>			10
<i>Iteration – 2</i>			
4	US-03	Menambah data admin	2
5	US-04	Mengubah data admin	2
6	US-05	Menghapus data admin	2
7	US-07	Menambah data tempat ibadah	4
<i>Velocity</i>			10
<i>Iteration – 3</i>			

8	US-14	Import data excel tempat ibadah	5
9	US-15	Unduh template format excel import data	2
10	US-16	Export data excel tempat ibadah	3
<i>Velocity</i>			10
<i>Iteration – 4</i>			
11	US-06	Melihat daftar tempat ibadah	6
12	US-13	Beranda halaman admin	2
13	US-08	Mengubah data tempat ibadah	2
<i>Velocity</i>			10
<i>Iteration – 5</i>			
14	US-09	Menghapus data tempat ibadah	2
15	US-10	Melihat peta lokasi tempat ibadah basis <i>android</i>	7
16	US-11	Melihat rute lokasi tempat ibadah android	1
<i>Velocity</i>			10
<i>Iteration – 6</i>			
17	US-17	Melihat peta lokasi tempat ibadah pada basis web	8
18	US-18	Melihat rute menuju lokasi tempat ibadah pada basis web	2
<i>Velocity</i>			10

Terdapat 3 iterasi tambahan dan dimana tiap iterasi dikerjakan selama 10 hari, sehingga rencana waktu penggeraan system bertambah menjadi 60 hari.

4.1.2 *Design*

Setelah dilakukannya perencanaan terhadap *user stories* yang ada, selanjutnya pada tahap ini akan dibahas mengenai hasil *design* atau perancangan terdahap *user stories* yang didapat. Rancangan yang dibuat pengembangan merupakan CRC *card* seperti yang telah dijabarkan pada Tabel 3.3 dan *Spike Prototype Solution* merupakan skema desain *prototype* berdasarkan CRC *card*. Berikut merupakan hasil pembuatan *design* dapat dilihat pada Tabel 4.3

Tabel 4.3 Hasil Design CRC-Card Dan Spike Solution Prototype

<i>Iteration</i>	Kode US	Nama CRC Card	<i>Spike Solution Prototype</i> Ditujukan gambar
1	US-01	Login	Gambar 3.21
	US-12	Profile	Gambar 3.27
	US-02	Admin	Gambar 3.28
2	US-03	Admin	Gambar 3.29
	US-04	Admin	Gambar 3.30
	US-05	Admin	Gambar 3.31
	US-07	Tempat ibadah basis web	Gambar 3.24
3	US-14	Tempat ibadah basis web	Gambar 3.37
	US-15	Tempat ibadah basis web	-
	US-16	Tempat ibadah basis web	-
4	US-06	Dashboard, tempat ibadah basis web dan android	Gambar 3.20, 3.23, dan 3.33
	US-13	Tempat ibadah basis web	Gambar 3.22
	US-08	Tempat ibadah basis web	Gambar 3.25
5	US-09	Tempat ibadah basis web	Gambar 3.26
	US-10	Tempat ibadah basis android	Gambar 3.32
	US-11	Tempat ibadah basis android	Gambar 3.32
6	US-17	Dashboard	Gambar 3.36
	US-18	Dashboard	Gambar 3.36

4.1.3 Coding

Pada sub-bab ini, akan dibahas mengenai hasil implementasi (*coding*) SIG Pemetaan Lokasi Tempat Ibadah. Implementasi yang dilakukan telah dibuat sesuai dengan perancangan yang dijabarkan pada Tabel 4.3. Selain itu, pada bab ini juga akan dibahas mengenai hasil pengujian dari sistem yang telah dibangun berdasarkan dengan *user stories* yang didapat. Pembahasan yang akan dijelaskan

pada sub bab ini meliputi implementasi *database*, implementasi API, implementasi *class*, implementasi *interface*, dan *refactoring*.

4.1.3.1 Implementasi *database*

Berikut merupakan implementasi *database* dari Sistem Informasi Geografis Untuk Pemetaan Lokasi Tempat Ibadah dapat dilihat pada Gambar 4.1

Table	Action
tb_admin	Browse Structure Search Insert Empty Drop
tb_bidang	Browse Structure Search Insert Empty Drop
tb_jenis	Browse Structure Search Insert Empty Drop
tb_kabupaten	Browse Structure Search Insert Empty Drop
tb_kecamatan	Browse Structure Search Insert Empty Drop
tb_ti	Browse Structure Search Insert Empty Drop
6 tables	Sum

Gambar 4.1 Implementasi *Database*

Gambar 4.1 merupakan implementasi *database* yang bernama “db_sig_rest”, terdapat 6(enam) tabel terdiri tb_admin, tb_bidang, tb_jenis, tb_kabupaten, tb_kecamatan, dan tb_ti. Pada *database* digunakan sebagai media menyimpan data pada sistem yang dibuat Berikut merupakan struktur dari masing masing tabel:

1. Tabel tb_admin

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	admin_id	int(10)			No	None		AUTO_INCREMENT
2	admin_username	varchar(255)	latin1_swedish_ci		Yes	NULL		
3	admin_password	varchar(255)	latin1_swedish_ci		Yes	NULL		
4	admin_nama	varchar(255)	latin1_swedish_ci		Yes	NULL		
5	admin_alamat	varchar(255)	latin1_swedish_ci		Yes	NULL		
6	admin_bidang	varchar(255)	latin1_swedish_ci		Yes	NULL		
7	admin_foto	varchar(255)	latin1_swedish_ci		Yes	NULL		

Gambar 4.2 Implementasi Tabel Tb_admin

Gambar 4.2 merupakan implementasi tabel tb_admin. Terdapat field admin_id, admin_username, admin_password, admin_nama, admin_alamat, admin_bidang, dan admin_foto. Tabel ini digunakan untuk validasi admin login, serta menampilkan dan mengelola data admin. Tabel ini berelasi dengan tabel tb_bidang.

2. Tabel tb_bidang

#	Name	Type	Collation	Attributes	Null	Default	Comments
1	bidang_id	int(3)			No	None	
2	bidang_nama	varchar(20)	latin1_swedish_ci		Yes	NULL	
3	bidang_role	int(3)			Yes	NULL	

Gambar 4.3 Implementasi Tabel Tb_bidang

Gambar 4.3 merupakan implementasi tabel tb_bidang. Terdapat field bidang_id, bidang_nama, dan bidang_role. Tabel ini digunakan untuk *role* bidang admin.

3. Tabel tb_jenis

#	Name	Type	Collation	Attributes	Null	Default
1	jenis_id	int(10)		No	None	
2	jenis_nama	varchar(20)	latin1_swedish_ci	Yes	NULL	
3	mapicon	varchar(255)	latin1_swedish_ci	Yes	NULL	

Gambar 4.4 Implementasi Tabel Tb_jenis

Gambar 4.4 merupakan implementasi tabel tb_jenis. Terdapat field jenis_id, jenis_nama, dan mapicon. Tabel ini digunakan sebagai kategori jenis tempat ibadah. Tabel ini berelasi dengan tabel tb_ti.

4. Tabel tb_ti

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ti_id	int(15)		No	None		AUTO_INCREMENT	
2	ti_jenis	varchar(255)	latin1_swedish_ci	Yes	NULL			
3	ti_tipologi	varchar(255)	latin1_swedish_ci	Yes	NULL			
4	ti_nama	varchar(255)	latin1_swedish_ci	Yes	NULL			
5	ti_alamat	varchar(255)	latin1_swedish_ci	Yes	NULL			
6	ti_kabupaten	varchar(255)	latin1_swedish_ci	Yes	NULL			
7	ti_kecamatan	varchar(255)	latin1_swedish_ci	Yes	NULL			
8	ti_luas_tanah	varchar(255)	latin1_swedish_ci	Yes	NULL			
9	ti_status_tanah	varchar(255)	latin1_swedish_ci	Yes	NULL			
10	ti_luas_bangunan	varchar(255)	latin1_swedish_ci	Yes	NULL			
11	ti_tahun_berdiri	varchar(255)	latin1_swedish_ci	Yes	NULL			
12	ti_jamaah	varchar(255)	latin1_swedish_ci	Yes	NULL			
13	ti_imam	int(10)		Yes	NULL			
14	ti_khatib	int(10)		Yes	NULL			
15	ti_remaja	int(10)		Yes	NULL			
16	ti_ketua	varchar(255)	latin1_swedish_ci	Yes	NULL			
17	ti_keterangan	varchar(100)	latin1_swedish_ci	Yes	NULL			
18	ti_binaan_majelis	varchar(255)	latin1_swedish_ci	Yes	NULL			
19	ti_kondisi	varchar(255)	latin1_swedish_ci	Yes	NULL			
20	ti_telepon	varchar(255)	latin1_swedish_ci	Yes	NULL			
21	latitude	varchar(255)	latin1_swedish_ci	Yes	NULL			
22	longitude	varchar(255)	latin1_swedish_ci	Yes	NULL			
23	ti_foto	varchar(255)	latin1_swedish_ci	Yes	NULL			

Gambar 4.5 Implementasi Tabel Tb_ti

Gambar 4.5 merupakan implementasi tabel tb_ti. Terdapat field ti_id, ti_jenis, ti_tipologi, ti_nama, ti_alamat dan lainnya dapat dilihat pada Gambar 4.5. tabel ini berguna untuk menyimpan data tempat ibadah yang ada. Tabel ini berelasi dengan tabel tb_jenis, tb_kabupaten, dan tb_kecamatan.

5. Tabel tb_kecamatan

#	Name	Type	Collation	Attributes	Null	Default
1	kec_id	int(3)			No	None
2	kec_nama	varchar(20)	latin1_swedish_ci		No	None
3	id_kab	int(3)			Yes	NULL

Gambar 4.6 Implementasi Tabel Tb_kecamatan

Gambar 4.6 merupakan implementasi tabel tb_kecamatan. Terdapat field kec_id, kec_nama, dan id_kab. Tabel ini digunakan untuk menyimpan data kecamatan yang ada pada sistem. Tabel ini berelasi dengan tb_kabupaten.

6. Tabel tb_kabupaten

#	Name	Type	Collation	Attributes	Null	Default
1	kab_id	int(3)			No	None
2	kab_nama	varchar(20)	latin1_swedish_ci		Yes	NULL
3	kab_latitude	varchar(255)	latin1_swedish_ci		Yes	NULL
4	kab_longitude	varchar(255)	latin1_swedish_ci		Yes	NULL

Gambar 4.7 Implementasi Tabel Tb_kabupaten

Gambar 4.7 merupakan implementasi tabel tb_kabupaten. Terdapat field kab_id, kab_nama, kab_latitude dan kab_longitude. Tabel ini digunakan untuk penyimpanan data kabupaten ada pada sistem. Tabel ini berelasi dengan tabel tb_kecamatan.

4.1.3.2 Implementasi API

Berikut merupakan implementasi API digunakan *controller* dan *model* dalam pembuatannya. API ini bertujuan agar sistem pada basis *android* dapat terintegrasi atau terhubung dengan *database* yang dibuat.

Berikut merupakan fungsi get pada *class controller* yang digunakan untuk *interface* agar terhubung dengan *class model* APInya.

```
1. public function index_get()
2. {
3.     $v_id_kab = $this->get('kab_id');
4.     $v_id_jenis = $this->get('jenis_id');
5.
6.     if (($v_id_kab != null) && ($v_id_jenis != null)) {
7.         $v_tempat_ibadah = $this->M_tempat_ibadah-
>selectByKabJenis($v_id_kab,$v_id_jenis);
8.     }elseif (($v_id_kab != null) && ($v_id_jenis === null))
9.     {
10.        $v_tempat_ibadah = $this->M_tempat_ibadah-
>selectByKabupaten($v_id_kab);
11.    }elseif (($v_id_kab === null) && ($v_id_jenis != n
ull)) {
```

```

11.           $v_tempat_ibadah = $this->M_tempat_ibadah-
>selectByJenis($v_id_jenis);
12.       }else{
13.           $v_tempat_ibadah = $this->M_tempat_ibadah-
>selectAlltempatIbadah();
14.       }
15.       if ($v_tempat_ibadah) {
16.           $this->response([
17.               'status' => true,
18.               'data' => $v_tempat_ibadah
19.           ], REST_Controller::HTTP_OK);
20.       }else {
21.           $this->response([
22.               'status' => false,
23.               'respone' => 'id is not found'
24.           ], REST_Controller::HTTP_NOT_FOUND);
25.       }
26.   }

```

Kode Sumber 4.1 Implementasi Fungsi *Get* Pada *Controller*

Berikut merupakan fungsi yang terdapat pada *class model* yang digunakan untuk mengambil data dalam *database*.

```

1. public function selectByJenis($jenis){
2.     $sql='SELECT * FROM tb_ti LEFT JOIN tb_kabupaten ON tb_
ti.ti_kabupaten = tb_kabupaten.kab_id LEFT JOIN tb_kecamatan
ON tb_kecamatan.id_kab = tb_kabupaten.kab_id AND tb_kecamatan.
kec_id = tb_ti.ti_kecamatan JOIN tb_jenis ON tb_ti.ti_jenis =
tb_jenis.jenis_id WHERE ti_jenis =?';
3.     return $this->db->query($sql,$jenis)->result_array();
4. }
5. public function selectAlltempatIbadah(){
6.     $sql='SELECT * FROM tb_ti LEFT JOIN tb_kabupaten ON tb_
ti.ti_kabupaten = tb_kabupaten.kab_id LEFT JOIN tb_kecamatan
ON tb_kecamatan.id_kab = tb_kabupaten.kab_id AND tb_kecamatan.
kec_id = tb_ti.ti_kecamatan JOIN tb_jenis ON tb_ti.ti_jenis =
tb_jenis.jenis_id';
7.     $query=$this->db->query($sql)->result_array();
8. }

```

Kode Sumber 4.2 Implementasi Fungsi *Get* Pada *Model*

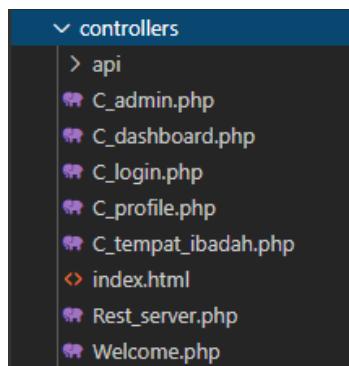
4.1.3.3 Implementasi *class*

Berdasarkan perancangan *class diagram* yang telah dirancang pada bab 3, implementasi *class* yang dilakukan dirasa sudah sesuai dengan perancangan tersebut. *Class - class* ini merupakan *class* yang berisi bahasa pemrograman PHP yang dibuat untuk mengimplementasikan sistem basis *web* sesuai dengan perancangan yang dilakukan berupa *sourcecode (coding)* dan bahasa pemrograman Kotlin untuk mengimplementasikan system basis *android*. Berikut merupakan implementasi *class* yang dilakukan dalam pengembangan Sistem Informasi Geografis Untuk Pemetaan Lokasi Tempat Ibadah.

4.1.3.3.1 Basis web

Berikut merupakan rincian *class – class* yang digunakan dalam membuat system basis *web* pada “Sistem Informasi Geografis untuk Pemetaan Lokasi Tempat Ibadah di Kota Mataram menggunakan Metode *Extreme Programming*”.

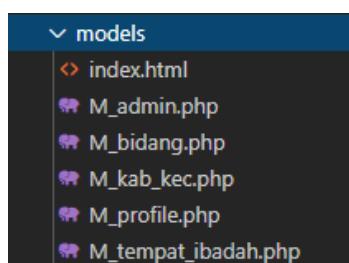
1. Class controller



Gambar 4.8 Implementasi *Class Controller*

Gambar 4.8 merupakan implementasi *class* turunan dari *class controller* yang terdiri dari C_admin, C_dashboard, C_login, C_profile C_tempat_ibadah, dan Rest_server. *Class – class* ini merupakan *class* yang menjadi pengatur atau penghubung pertukaran data antara *class model* dan *view*.

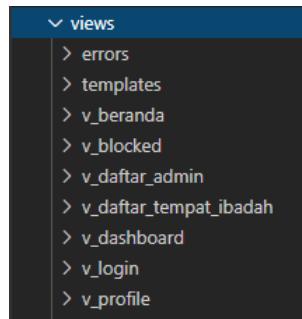
2. Class model



Gambar 4.9 Implementasi *Class Model*

Gambar 4.9 merupakan implementasi *class* turunan dari *class model* yang terdiri dari M_admin, M_bidang, M_kab_kec, M_profile, dan M_tempat_ibadah. Pada *class* ini nantinya berguna untuk mengatur atau memanipulasi data dari *database* sesuai dengan intruksi dari *controller*.

3. Class view



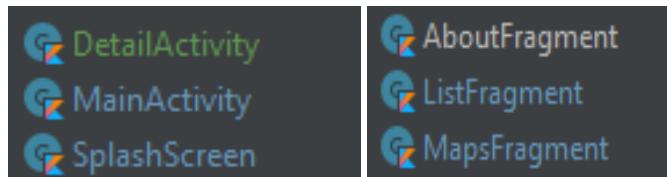
Gambar 4.10 Implementasi Class View

Gambar 4.10 merupakan implementasi *class view* yang terdiri dari V_beranda, V_daftar_admin, V_daftar_tempat_ibadah, V_dashboard, V_login, dan V_profile. Dimana *class* ini nantinya bertugas untuk menyajikan informasi.

4.1.3.3.2 Basis android

Berikut merupakan *class – class* yang digunakan dalam membuat system basis *android* pada “Sistem Informasi Geografis untuk Pemetaan Lokasi Tempat Ibadah di Kota Mataram menggunakan Metode *Extreme Programming*”.

1. Class activity



Gambar 4.11 Implementasi Class Activity

Gambar 4.11 merupakan implementasi *Package Activity* yang terdiri dari DetailActivity, MainActivity, SplashScreen, AboutFragment, ListFragment, dan MapFragment. *Package Activity* ini merupakan sebuah kelas yang nantinya terhubung ke Layout (XML) atau tampilan dari aplikasi. *Activity* ini nantinya tempat dilakukan koding untuk membuat fungsi-fungsi yang akan ditampilkan pada sebuah halaman.

a. DetailActivity

Merupakan kelas yang akan menampilkan informasi detail mengenai data tempat ibadah yang terdapat pada menu *list* maupun pada menu peta. Informasi yang ditampilkan pada kelas ini yaitu nama, jenis tempat ibadah, alamat, kabupaten, kecamatan dsb sesuai dengan data yang terdapat pada *database*. Kelas ini terhubung dengan layout XML *activity_detail*.

b. **MainActivity**

Kelas MainActivity merupakan kelas utama pada aplikasi ini, kegunaan dari MainActivity ini sebagai *handler* atau penghubung antara ListFragment, MapsFragment, dan AboutFragment yang sudah dibuat. Pada Kelas ini juga digunakan untuk membuat menu navigasi aplikasi seperti list, map, dan about us. Kelas ini juga terhubung dengan layout xml activity_main.xml.

c. **SplashScreen**

Merupakan kelas yang pertama kali ditampilkan kepada pengguna aplikasi berupa tampilan *loading* atau logo dari aplikasi. Kelas ini memiliki tampilan *delay* selama 3 detik yang kemudian diarahkan ke kelas MapsFragment. Kelas ini terhubung ke activity_splash_screen.xml

d. **AboutFragment**

Merupakan kelas yang berguna untuk menampilkan deskripsi singkat dari aplikasi dan terdapat informasi kontak Kemenag NTB. Kelas ini terhubung dengan layout XML fragment_about.xml.

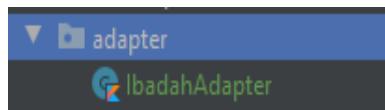
e. **ListFragment**

Merupakan kelas yang berguna untuk menampilkan data – data tempat ibadah *list view*. Di kelas ini terdapat beberapa fungsi seperti menampilkan menu *dropdown* filter jenis tempat ibadah, fungsi pencarian dan menampilkan detail data tempat ibadah. Kelas ini terhubung dengan layout xml fragment_list.xml.

f. **MapsFragment**

Merupakan kelas yang berguna untuk menampilkan daftar tempat ibadah dalam bentuk *icon* yang tersaji pada peta aplikasi. Kelas ini juga memiliki fungsi pencarian yang berguna untuk mencari lokasi tempat ibadah yang ada pada aplikasi, fungsi filter yang berguna untuk menampilkan lokasi tempat ibadah berdasarkan jenis tempat ibadah, dan menampilkan informasi detail berdasarkan tempat ibadah yang dipilih. Kelas ini juga terhubung pada google maps, sehingga ketika pengguna melakukan klik pada salah satu daftar tempat ibadah, maka akan menampilkan tombol untuk melihat rute dan diarahkan langsung ke google maps. Kelas ini terhubung dengan layout XML fragment_maps.xml.

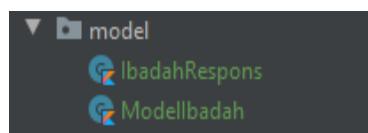
2. Adapter (Kotlin)



Gambar 4.12 Adapter

Berbeda dengan *package Activity*, package adapter pada Gambar 4.12 ini merupakan kelas yang menghubungkan antara data yang ada pada *database* ke dalam sebuah *listview* pada aplikasi.

3. Model (Kotlin)



Gambar 4.13 Implementasi *Class Model*

Gambar 4.13 merupakan implementasi *Package Model* yang terdiri dari IbadahRespons dan Modellibadah. Package model ini merupakan pengimplementasian pada sebuah tabel di *database*. Dimana dapat dilihat nama setiap kelas yang dibuat merupakan nama yang ada pada Tabel di *database*.

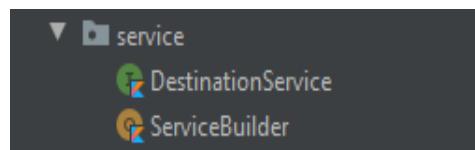
a. IbadahRespons

IbadahRespons merupakan sebuah data *class* yang terhubung dengan kelas Modellibadah, dimana kelas ini berguna untuk memberikan informasi terkait dengan status dan data yang digunakan pada API Modellibadah, jika statusnya true maka akan menampilkan data dalam bentuk *array*, namun jika false akan menampilkan pesan error.

b. Modellibadah

Kelas Modellibadah merupakan sebuah data *class* yang berguna untuk mengimplementasikan *field* yang ada pada Tabel tb_ti pada database seperti ti_id, ti_nama, ti_alamat, dsb. Kelas ini nantinya akan banyak digunakan.

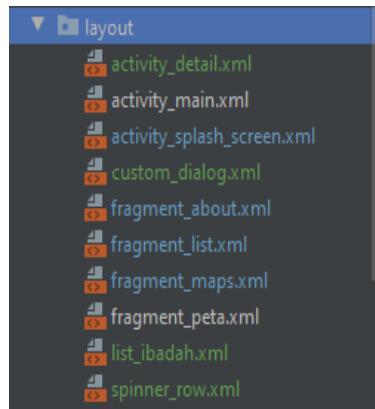
4. Service (Kotlin)



Gambar 4.14 Implementasi *Class Service*

Gambar 4.14 merupakan implementasi *Package Service* yang terdiri dari *DestinationService* dan *ServiceBuilder*. *Package Service* merupakan cara yang digunakan untuk terhubung dengan API server sistem. Dimana *ServiceBuilder* ini merupakan kelas untuk menghubungkan aplikasi dengan *Base url* API sistem, dan kelas ini juga terhubung dengan kelas *interface* *DestinationService*. Sedangkan kelas *DestinationService* digunakan untuk mengimplementasi *endpoint* dari *base url* yang telah dibuat pada API.

5. Layout (XML)



Gambar 4.15 Layout (XML)

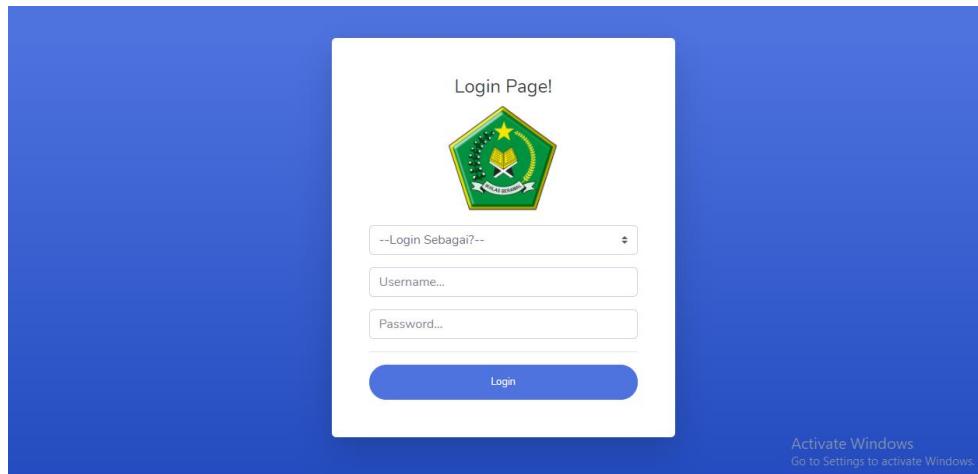
Gambar 4.15 merupakan implementasi Layout yang berguna untuk tempat membuat sebuah tampilan yang nantinya akan dilihat oleh pengguna.

4.1.3.4 Implementasi *interface*

Berikut merupakan implementasi *interface* sistem yang telah dibuat beserta dengan hasil pengujian yang telah uji oleh *client* berdasarkan perencanaan dari *user stories* yang dijabarkan.

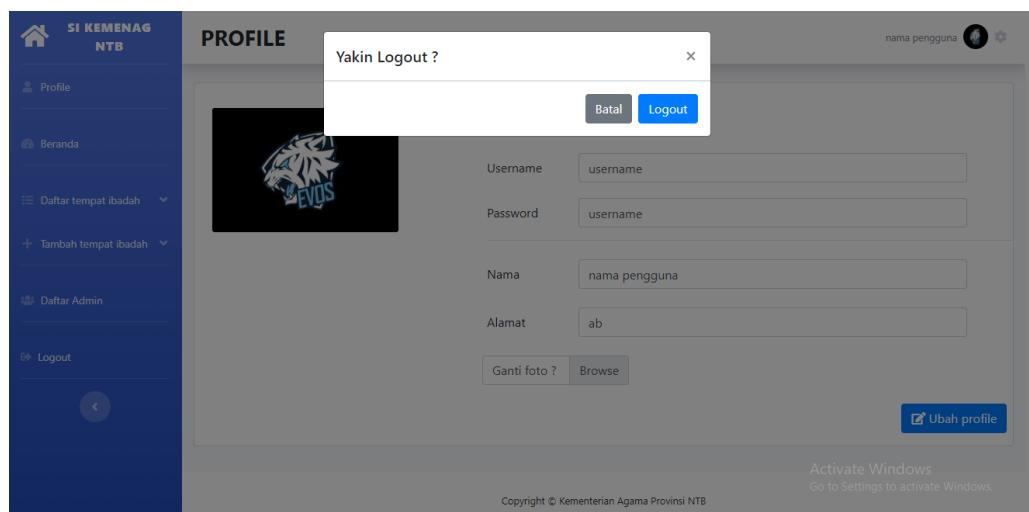
1. US-01

Berikut merupakan tampilan sistem dari kasus kode *user stories* US-01 yang telah dibuat dapat dilihat pada Gambar 4.16 berikut:



Gambar 4.16 Tampilan Halaman *Login*

Pada Gambar 4.16 merupakan implementasi halaman *login* pada basis *web* yang akan digunakan admin untuk dapat login ke dalam sistem. Pada halaman ini admin diminta untuk memilih sub bag, dan memasukkan *username* dan *password* yang telah terdaftar dalam sistem.



Gambar 4.17 Tampilan *Logout*

Pada Gambar 4.17 merupakan implementasi tampilan *logout* yang digunakan admin untuk mengakhiri sistem.

Pengujian sistem pada kasus kode *user stories* US-01 ini dilakukan untuk menguji apakah fungsi *login* dan *logout* sistem sudah berjalan dengan baik dan benar. Pengujian tersebut dijelaskan pada Tabel 4.4 dan tabel 4.5.

Tabel 4.4 Pengujian *Login* System

Skenario pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
Bidang, <i>Username</i> , dan <i>password</i> tidak diisi kemudian klik tombol login	Sistem akan menolak akses dan akan dirujuk kembali ke halaman login	Sesuai	Diterima
Bidang tidak dipilih, Mengetik <i>username</i> dan <i>password</i> tidak diisi atau kosong kemudian klik tombol login	Sistem akan menolak akses dan akan dirujuk kembali ke halaman login	Sesuai	Diterima
Bidang dipilih, <i>Username</i> tidak diisi dan <i>password</i> diisi kemudian klik tombol login	Sistem akan menolak akses dan akan dirujuk kembali ke halaman login	Sesuai	Diterima
Bidang, <i>Username</i> dan <i>password</i> diisi tidak benar kemudian klik tombol login	Sistem akan menolak akses dan akan dirujuk kembali ke halaman login	Sesuai	Diterima
Bidang, <i>Username</i> dan <i>password</i> diisi dengan benar kemudian klik tombol login	Sistem menerima akses <i>login</i> dan kemudian langsung menampilkan halaman admin sesuai <i>role</i> bidang admin	Sesuai	Diterima

Sistem hanya akan menerima akses jika bidang, *username* dan *password* yang diisi sesuai dengan yang ada di *database*, selain itu sistem akan menolak akses.

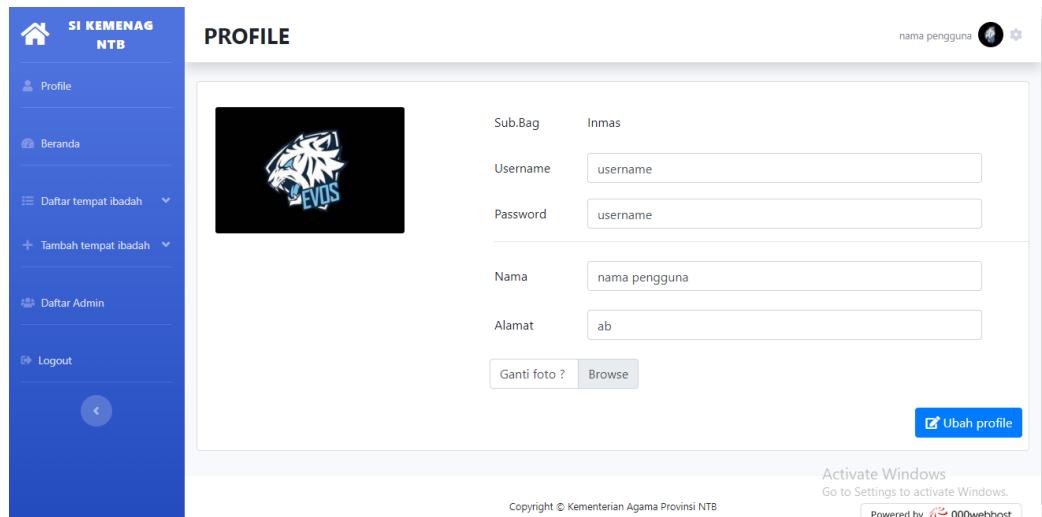
Tabel 4.5 Pengujian *Logout System*

Skenario pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
Menekan tombol <i>batal</i>	Proses <i>logout</i> tidak akan terjadi dan dirujuk kembali ke halaman admin	Sesuai	Diterima
Menekan tombol <i>logout</i>	Proses <i>logout</i> terjadi dan diarahkan langsung ke halaman <i>login</i>	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.4 dan table 4.5, dapat dikatakan system berhasil dan sesuai keinginan *client*.

2. US-12

Berikut merupakan tampilan halaman *profile* admin sistem dari kasus kode *user stories* US-12 yang telah dibuat dapat dilihat pada Gambar 4.18 berikut:



Gambar 4.18 Tampilan Halaman *Profile* Admin

Pada Gambar 4.18 merupakan implementasi tampilan halaman *profile* admin pada basis *web*. Pada halaman ini admin dapat melihat *profile* dan dapat mengubah data pribadi admin.

Pengujian sistem pada kasus kode *user stories* US-12 ini dilakukan untuk menguji apakah fungsi melihat dan mengubah *profile* admin pada sistem sudah berjalan dengan baik dan benar. Pengujian tersebut dijelaskan pada Tabel 4.6

Tabel 4.6 Pengujian Halaman Profile Admin

Skenario pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
Menekan tombol halaman profile admin	Data informasi profile admin berhasil ditampilkan	Sesuai	Diterima
Mengisi dan mengubah data profile baru kemudian menekan tombol edit profile	Data informasi profile yang baru berhasil diubah	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.6 dapat dikatakan system berhasil dan sesuai kebutuhan *client*.

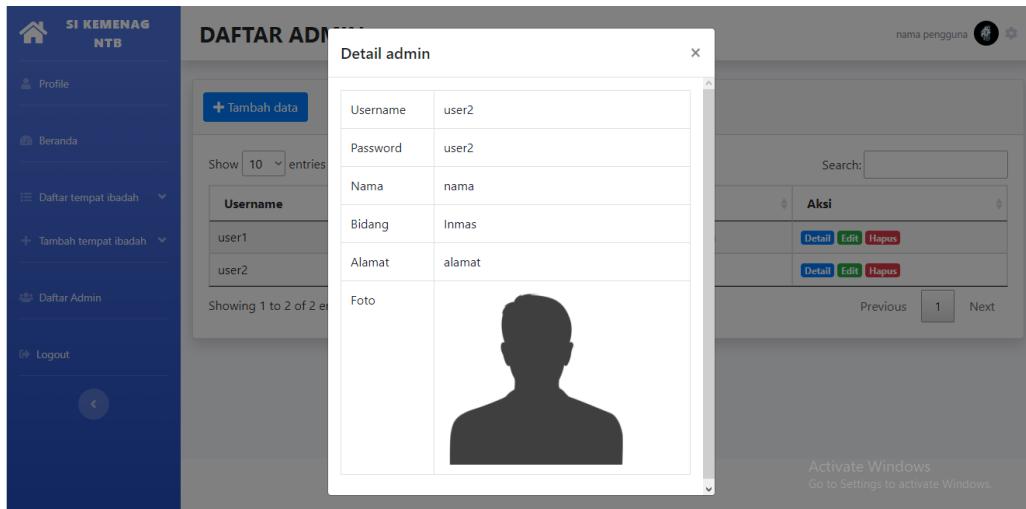
3. US-02

Berikut merupakan tampilan sistem dari kasus kode *user stories* US-02 yang telah dibuat dapat dilihat pada Gambar 4.19 dan 4.20 berikut:

Username	Password	Nama	Bidang	Aksi
user1	user1	nama	Bimas islam	[Detail] [Edit] [Hapus]
user2	user2	nama	Inmas	[Detail] [Edit] [Hapus]

Gambar 4.19 Tampilan Halaman Daftar Admin

Pada Gambar 4.19 merupakan implementasi tampilan halaman daftar admin pada basis *web*. Pada halaman ini super admin dapat melihat daftar admin yang tersusun dalam bentuk tabel dan dapat melihat informasi detail data admin.



Gambar 4.20 Tampilan Detail Data Admin

Pada Gambar 4.20 merupakan implementasi tampilan detail data admin pada basis web yang dapat dilihat oleh super admin.

Dilakukan pengujian untuk mengetahui apakah fungsi untuk menampilkan daftar dan detail data admin sudah berjalan dengan baik dan benar. Pengujian fungsi tersebut dapat dilihat pada Tabel 4.7

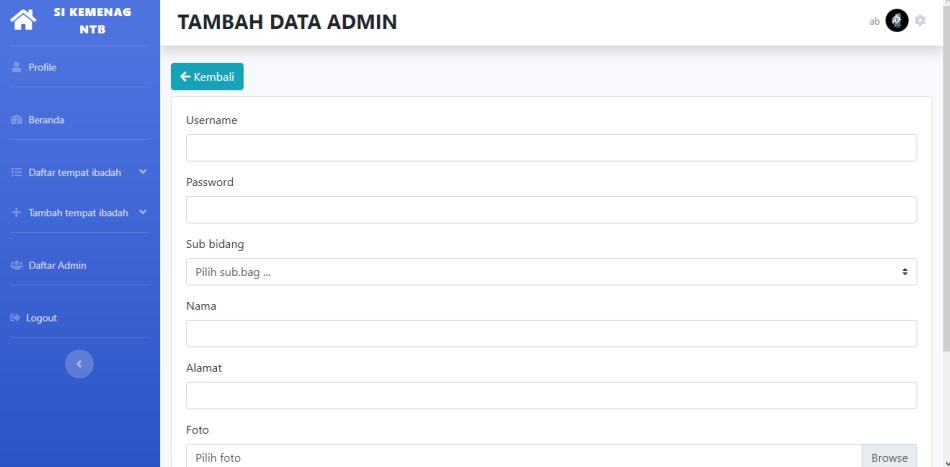
Tabel 4.7 Pengujian Menampilkan Data Admin

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol halaman daftar admin	Daftar data admin berhasil ditampilkan	Sesuai	Diterima
Menekan tombol detail data admin yang dipilih	Detail data admin yang dipilih berhasil ditampilkan	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.7 dapat dikatakan system berhasil dan sesuai kebutuhan *client*.

4. US-03

Berikut merupakan tampilan halaman tambah data admin dari kasus kode *user stories* US-03 yang telah dibuat dapat dilihat pada Gambar 4.21 berikut:



Gambar 4.21 Tampilan Tambah Data Admin

Pada Gambar 4.21 merupakan implementasi tampilan halaman tambah data admin pada basis web. Pada halaman ini super admin dapat menambahkan data admin seperti username, password, nama dsb.

Pengujian fungsi pada kasus kode *user stories* US-03 ini dilakukan untuk menguji apakah fungsi untuk menambahkan data admin sudah berjalan dengan baik dan benar. Pengujian fungsi tersebut dapat dilihat pada Tabel 4.8

Tabel 4.8 Pengujian Tambah Data Admin

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Sub bidang tidak dipilih, serta <i>Username</i> dan <i>password</i> tidak diisi	Data tidak berhasil ditambah	Sesuai	Diterima
Sub bidang dipilih, <i>Username</i> dan <i>password</i> diisi. Namun <i>username</i> telah terdaftar dalam sistem dan <i>password</i> kurang 3 kalimat	Data tidak berhasil ditambah	Sesuai	Diterima
Sub bidang dipilih, serta <i>password</i> diisi dengan benar dan	Sistem akan melakukan proses	Sesuai	Diterima

<i>username</i> belum tedaftar dalam <i>database</i>	dan data berhasil ditambahkan		
---	----------------------------------	--	--

Sistem hanya akan berhasil menambahkan data jika sub bidang terpilih, dan *username* yang diinputkan belum terdaftar dalam *database* serta *password* yang diinputkan lebih dari 3 kata, selain itu sistem akan menolak akses untuk menambahkan data admin. Sehingga berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.8 dapat dikatakan system berhasil dan sesuai kebutuhan *client*.

5. US-04

Berikut merupakan tampilan halaman edit data admin dari kasus kode *user stories* US-04 yang telah dibuat dapat dilihat pada Gambar 4.22 berikut:

Gambar 4.22 Tampilan Edit Data Admin

Pada Gambar 4.22 merupakan implementasi tampilan halaman edit data admin pada basis web. Pada halaman ini super admin dapat mengubah data admin seperti mengubah *username*, *password*, *nama* dsb.

Pengujian fungsi pada kasus kode *user stories* US-04 ini dilakukan untuk menguji apakah fungsi untuk mengubah data admin sudah berjalan dengan baik dan benar. Pengujian fungsi tersebut dapat dilihat pada Tabel 4.9

Tabel 4.9 Pengujian Mengubah Data Admin

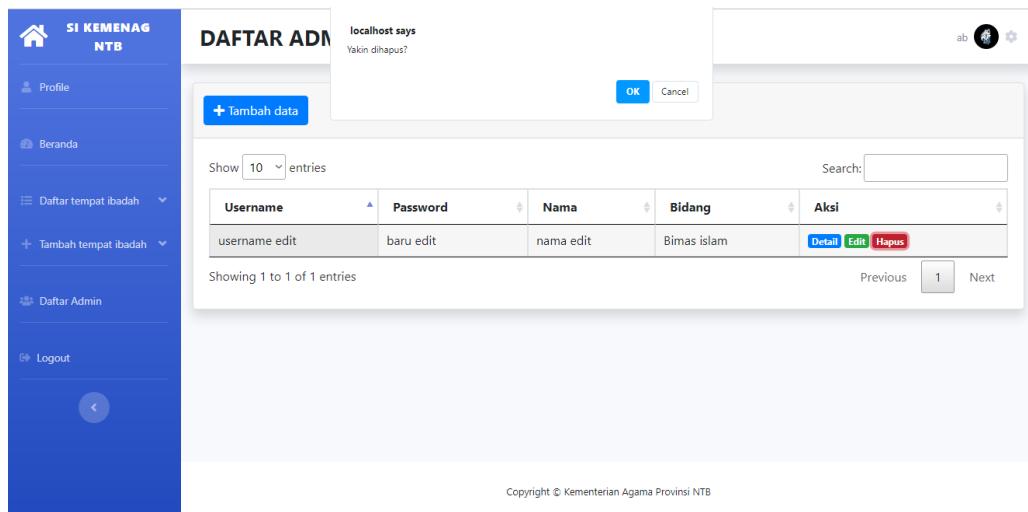
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Sub bidang tidak dipilih, serta <i>Username</i> atau <i>password</i> tidak diisi	Data tidak berhasil diubah	Sesuai	Diterima

Sub bidang dipilih, <i>Username</i> dan <i>password</i> diisi. Namun <i>username</i> telah terdaftar dalam sistem dan <i>password</i> kurang 3 kalimat	Data tidak berhasil diubah	Sesuai	Diterima
Sub bidang dipilih, serta <i>password</i> diisi dengan benar dan <i>username</i> belum tedafta r dalam <i>database</i>	Sistem akan melakukan proses dan data berhasil diubah	Sesuai	Diterima

Sistem hanya akan berhasil mengubah data admin jika sub bidang terpilih, dan *username* yang diinputkan belum terdaftar dalam *database* serta *password* yang diinputkan lebih dari 3 kata, selain itu sistem akan menolak akses untuk mengubah data admin. Sehingga berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.9 dapat dikatakan system berhasil dan sesuai kebutuhan *client*.

6. US-05

Berikut merupakan tampilan hapus data admin dari kasus kode *user stories* US-05 yang telah dibuat dapat dilihat pada Gambar 4.23 berikut:



Gambar 4.23 Tampilan Hapus Data Admin

Pada Gambar 4.23 merupakan implementasi tampilan hapus data admin pada basis *web*. Pada halaman ini super admin dapat menghapus data admin.

Pengujian fungsi pada kasus kode *user stories* US-05 ini dilakukan untuk menguji apakah fungsi untuk menghapus data admin sudah berjalan dengan baik dan benar. Pengujian fungsi tersebut dapat dilihat pada Tabel 4.10

Tabel 4.10 Pengujian Mengahpus Data Admin

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol “Cancel”	Data tidak terhapus	Sesuai	Diterima
Menekan tombol “Ok”	Data terhapus dalam sistem	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.10 dapat dikatakan system berhasil dan sesuai kebutuhan *client*.

7. US-07

Berikut merupakan tampilan halaman tambah data tempat ibadah dari kasus kode *user stories* US-07 yang telah dibuat dapat dilihat pada Gambar 4.23 berikut:

Gambar 4. 24 Tampilan Halaman Tambah Data Tempat Ibadah

Gambar 4.24 merupakan implementasi halaman tambah data tempat ibadah yang akan ditampilkan dalam sistem basis *web*. Pada halaman ini admin dapat melakukan penambahan data tempat ibadah ke dalam sistem.

Berikut merupakan tabel pengujian system terhadap kasus kode *user stories* US-07 dapat dilihat pada table 4.11.

Tabel 4.11 Pengujian Tambah Data Tempat Ibadah

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Nama atau alamat tempat ibadah tidak diisi, serta kabupaten atau kecamatan tidak dipilih	Sistem akan melakukan proses dan data tidak berhasil ditambahkan ke dalam <i>database</i>	Sesuai	Diterima
Nama dan alamat tempat ibadah diisi, serta kabupaten dan kecamatan dipilih	Sistem akan melakukan proses dan data berhasil ditambahkan ke dalam <i>database</i>	Sesuai	Diterima

Sistem hanya akan berhasil menambahkan data jika *form* nama dan alamat tempat ibadah terisi serta kabupaten dan kecamatan terpilih, selain itu sistem akan menolak akses untuk menambahkan data tempat ibadah. Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.11 dapat dikatakan system yang dibuat berhasil dan sesuai dengan kebutuhan *client*.

8. US-14

Berikut merupakan tampilan halaman tambah data tempat ibadah dari kasus kode *user stories* US-14 yang telah dibuat dapat dilihat pada Gambar 4.25 berikut:

Tipologi	Nama	Alamat	Kabupaten	Kecamatan	Aksi
Masjid Jami	Masjid Al-Falah	Gatep Permai Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Al-Furqon	Taman Sari Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Khairul Huda	Irigasi Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Baitul Salam	Taman Gajah Mada Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Baitul Hikmah	Taman Gajah Mada Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Hayatul Abyan	Pejeruk Abyan Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Nurul Ikhlas	Pejeruk Bangket Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Raya	Hubbul Wathan Islamic Centre	Jln. Udayana No. 1 Kota Mataram	Kota Mataram	Selaparang	[Detail] [Edit] [Hapus]

Gambar 4.25 Tampilan Import Data Tempat Ibadah

Gambar 4.25 merupakan implementasi tampilan halaman *import* data pada basis *web*. Terdapat tombol *import* data yang digunakan oleh admin untuk meng-

import data tempat ibadah dalam bentuk *file excel*. Berikut merupakan tabel pengujian system dapat dilihat pada table 4.12.

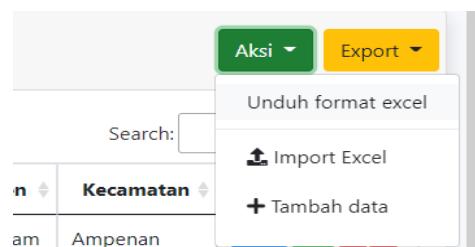
Tabel 4. 12 Pengujian Import Data Tempat Ibadah

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol “ <i>import data</i> ”	Menampilkan <i>form</i> untuk memasukkan file <i>excel</i> data tempat ibadah	Sesuai	Diterima
Menekan tombol “ <i>pilih file excel</i> ”	Menampilkan jendela baru untuk memasukkan file <i>excel</i>	Sesuai	Diterima
Menekan tombol “ <i>import</i> ”	Memproses dan memasukkan data ke dalam <i>database</i>	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.12 dapat dikatakan system yang dibuat berhasil dan sesuai dengan kebutuhan *client*.

9. US-15

Berikut merupakan tampilan untuk mengunduh format *excel* import data tempat ibadah pada sistem dari kasus kode *user stories* US-15 yang telah dibuat dapat dilihat pada Gambar 4.26 berikut:



Gambar 4.26 Tampilan *Import Data Tempat Ibadah*

Gambar 4.26 merupakan implementasi tampilan tombol untuk mengunduh format *excel* data tempat ibadah pada basis web. Tombol tersebut dapat digunakan oleh admin untuk mengunduh format *excel* dari data tempat ibadah yang akan di *import*. Berikut merupakan tabel pengujian system dapat dilihat pada table 4.13.

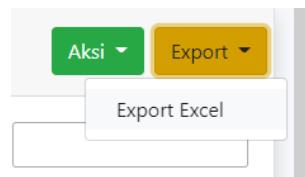
Tabel 4.13 Pengujian Unduh Template Format *Excel Import* Data

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol “ <i>unduh format excel</i> ”	Format excel data tempat ibadah terunduh	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.13 dapat dikatakan system yang dibuat berhasil dan sesuai dengan kebutuhan *client*.

10. US-16

Berikut merupakan tampilan tombol untuk *export* data tempat ibadah pada sistem dari kasus kode *user stories* US-16 yang telah dibuat dapat dilihat pada Gambar 4.27 berikut:



Gambar 4.27 Tampilan *Export* Data Tempat Ibadah

Gambar 4.27 merupakan implementasi tampilan tombol pada basis *web* untuk *export* data tempat ibadah dalam bentuk *file excel*. Berikut merupakan tabel pengujian system dapat dilihat pada table 4.14.

Tabel 4.14 Pengujian Export Data *Excel* Tempat Ibadah

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol “ <i>export</i> ”	Data tempat ibadah terunduh dalam bentuk excel	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.14 dapat dikatakan system yang dibuat berhasil dan sesuai dengan kebutuhan *client*.

11. US-06

Berikut merupakan tampilan halaman daftar tempat ibadah yang ada pada basis *web* dan *android* dari kasus kode *user stories* US-06 yang telah dibuat dapat dilihat pada Gambar 4.28, 4.29, 4.30, 4.31 dan 4.32 berikut:

Tipologi	Nama	Alamat	Kabupaten	Kecamatan	Aksi
Masjid Jami	Masjid Baiturrahman	Kapitan Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Al-Falah	Gatep Permai Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Al-Furqon	Taman Sari Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Khairul Huda	Irigasi Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Baitul Salam	Taman Gajah Mada Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Baitul Hikmah	Taman Gajah Mada Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Hayatul Abyan	Pejeruk Abyan Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Jami	Masjid Nurul Ikhlas	Pejeruk Bangket Ampenan	Kota Mataram	Ampenan	[Detail] [Edit] [Hapus]
Masjid Raya	Hubbul Wathan Islamic Centre	Jln. Udayana No. 1 Kota Mataram	Kota Mataram	Selaparang	[Detail] [Edit] [Hapus]

Gambar 4.28 Tampilan Daftar Tempat Ibadah Basis Web Sisi Admin

Pada Gambar 4.28 merupakan implementasi tampilan halaman daftar tempat ibadah pada basis *web* dari sisi admin. Pada halaman ini admin dapat melihat daftar tempat ibadah yang tersusun dalam bentuk tabel.

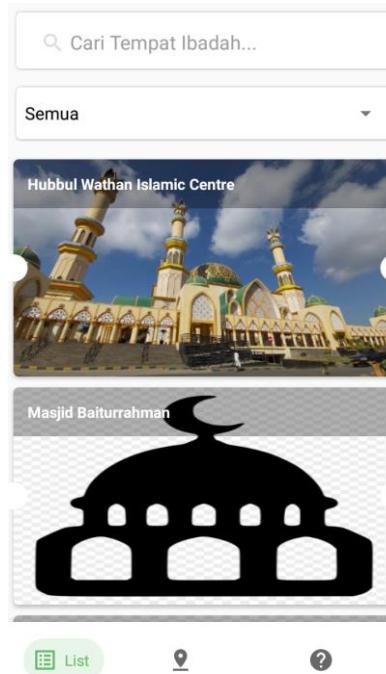
Jenis	Nama	Alamat	Kabupaten	Kecamatan	Aksi
Gereja	GBI Glow Mataram	Jl. Sriwijaya No. 83 Mataram	Kota Mataram	Mataram	[Detail]
Gereja	GBI Rock Ministry Mataram	Jl. IGK Jelantik Gosa 23 Gebang Mataram-NTB	Kota Mataram	Mataram	[Detail]
Gereja	GKII Bethel Cakranegara	Jl. Tumpang Sari 43 Cakranegara	Kota Mataram	Cakranegara	[Detail]
Gereja	GKII Kalvari Ampenan	Jl. Panjitalar No. 211 Ampenan	Kota Mataram	Ampenan	[Detail]
Gereja	GKII Rhema Cakranegara	Jl. Tumpang Sari No. 43a Cakranegara	Kota Mataram	Cakranegara	[Detail]
Gereja	GKPB 'Masa Depan Cerah' Mataram	Jl. Brawijaya No. 111 Sweta	Kota Mataram	Mataram	[Detail]
Gereja	GKT Gloria Cakranegara	Jl. Subak II No. 15 Cakranegara	Kota Mataram	Cakranegara	[Detail]
Gereja	GKT Jemaat Ampenan	JL Majapahit No.47	Kota Mataram	Ampenan	[Detail]

Gambar 4.29 Tampilan Daftar Tempat Ibadah Basis Web Sisi Masyarakat

Pada Gambar 4.29 merupakan implementasi tampilan halaman daftar tempat ibadah pada basis *web* dari sisi masyarakat. Pada halaman ini masyarakat umum dapat melihat daftar tempat ibadah yang tersusun dalam bentuk tabel.

Gambar 4.30 Tampilan Detail Data Tempat Ibadah Basis Web

Pada Gambar 4.30 merupakan implementasi tampilan detail informasi tempat ibadah pada basis *web*. Yang dapat dilihat oleh admin maupun masyarakat umum.



Gambar 4.31 Tampilan Daftar Tempat Ibadah Basis *Android*

Pada Gambar 4.31 merupakan implementasi tampilan halaman daftar tempat ibadah pada basis *android*. Pada halaman ini masyarakat umum dapat melihat daftar tempat ibadah berupa *list view*. Pada halaman ini masyarakat umum

dapat melihat mencari tempat ibadah dan dapat menampilkan daftar tempat ibadah berdasarkan filter jenis tempat ibadah yang diinginkan.



Hubbul Wathan Islamic Centre

• Masjid Raya

Informasi

Alamat	Jln. Udayana No. 1 Kota Mataram
Kabupaten	Kota Mataram
Kecamatan	Selaparang
Luas tanah	10.000 m2
Status tanah	Wakaf
Luas bangunan	1.500 m2
Tahun berdiri	2014
Jamaah	150 - 200
Imam	4

Gambar 4.32 Tampilan Detail Tempat Ibadah Basis Android

Pada Gambar 4.32 merupakan implementasi tampilan halaman detail informasi tempat ibadah pada basis *android*. Pada halaman ini masyarakat umum dapat melihat detail informasi tempat ibadah seperti nama, gambar alamat dsb.

Dilakukan pengujian sistem untuk mengetahui apakah fungsi untuk menampilkan daftar dan detail data tempat ibadah yang ada pada basis *web* dan *android* sudah berjalan dengan baik dan benar. Pengujian fungsi tersebut dapat dilihat pada Tabel 4.15

Tabel 4.15 Pengujian Menampilkan Data Tempat Ibadah

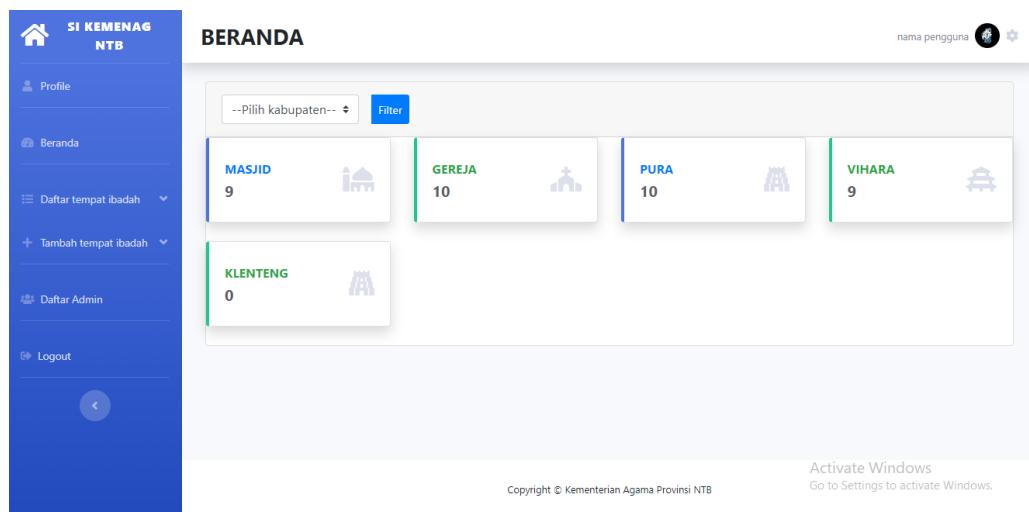
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol halaman data tempat ibadah (<i>web</i> atau <i>android</i>)	Data tempat ibadah berhasil ditampilkan	Sesuai	Diterima
Menekan tombol detail data tempat ibadah yang dipilih (<i>web</i> atau <i>android</i>)	Detail data tempat ibadah yang dipilih berhasil ditampilkan	Sesuai	Diterima

Memasukkan <i>keyword</i> pada fitur pencarian terhadap data tempat ibadah (<i>web</i> atau <i>android</i>)	Menampilkan hasil <i>keyword</i> pencarian	Sesuai	Diterima
Menggunakan filter kabupaten dan jenis tempat ibadah(<i>web</i>)	Menampilkan hasil tempat ibadah berdasarkan filter yang diinginkan	Sesuai	Diterima
Menggunakan filter jenis tempat ibadah (<i>android</i>)	Menampilkan hasil tempat ibadah berdasarkan filter yang diinginkan	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.15 dapat dikatakan system berhasil dan sesuai kebutuhan *client*.

12. US-13

Berikut merupakan tampilan halaman beranda sisi admin pada basis *web* dari kasus kode *user stories* US-13 yang telah dibuat dapat dilihat pada Gambar 4.33 berikut:



Gambar 4.33 Tampilan Halaman Beranda Pada Sisi Admin

Pada Gambar 4.33 merupakan implementasi halaman beranda pada basis *web*. Pada halaman ini admin dapat memlihat total masing – masing jenis tempat ibadah berdasarkan kabupaten yang ada di provinsi NTB.

Dilakukan pengujian sistem untuk mengetahui apakah fungsi untuk menampilkan daftar dan detail data tempat ibadah sudah berjalan dengan baik dan benar. Pengujian fungsi tersebut dapat dilihat pada Tabel 4.16

Tabel 4.16 Pengujian Pada Halaman Beranda Sisi Admin

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol halaman beranda	Total masing – masing jenis tempat ibadah berhasil ditampilkan	Sesuai	Diterima
Menggunakan filter kabupaten dan jenis tempat ibadah	Menampilkan hasil total masing – masing tempat ibadah berdasarkan filter yang diinginkan	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.16 dapat dikatakan system berhasil dan berjalan sesuai kebutuhan *client*.

13. US-08

Berikut merupakan tampilan halaman untuk mengubah data tempat ibadah pada sisi admin dari kasus kode *user stories* US-13 yang telah dibuat dapat dilihat pada Gambar 4.34 berikut:

Gambar 4.34 Tampilan Halaman Edit Data Tempat Ibadah

Gambar 4.34 merupakan implementasi halaman edit data tempat ibadah yang akan ditampilkan dalam sistem basis *web*. Pada halaman ini admin dapat mengubah informasi tempat ibadah seperti mengubah nama, alamat dll. Berikut

merupakan table pengujian system terhadap kasus kode *user stories* US-08 dapat dilihat pada table 4.17.

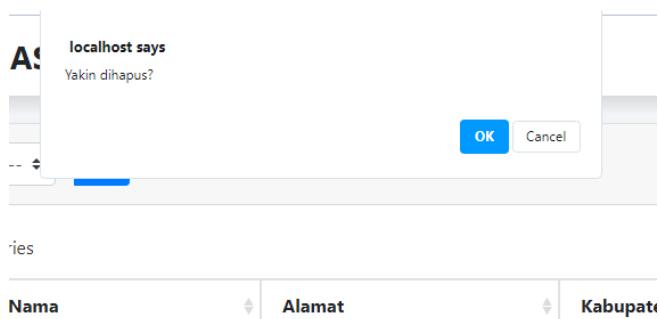
Tabel 4.17 Pengujian Ubah Data Tempat Ibadah

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Nama atau alamat tidak diisi, serta kabupaten atau kecamatan tidak dipilih	Sistem akan melakukan proses dan data tidak berhasil ditambahkan	Sesuai	Diterima
Nama dan alamat diisi, serta kabupaten dan kecamatan dipilih	Sistem akan melakukan proses dan data berhasil ditambahkan	Sesuai	Diterima

Sama seperti halnya pengujian menambahkan data tempat ibadah, sistem hanya akan berhasil mengubah data jika *form* nama dan alamat tempat ibadah terisi serta kabupaten dan kecamatan terpilih, selain itu sistem akan menolak akses untuk mengubah data tempat ibadah. Berdasarkan hasil pengujian dan kesimpulan yang didapat pada table 4.17, dapat dikatakan bahwa sistem yang dibuat benar dan sesuai dengan kebutuhan *client*.

14. US-09

Berikut merupakan tampilan untuk menghapus data tempat ibadah pada sisi admin dari kasus kode *user stories* US-09 yang telah dibuat dapat dilihat pada Gambar 4.35 berikut:



Gambar 4.35 Tampilan Hapus Data Tempat Ibadah

Pada Gambar 4.35 merupakan implementasi tampilan untuk admin menghapus data tempat ibadah dari sistem pada basis *web*. Pengujian fungsi pada kasus kode *user stories* US-09 ini dilakukan untuk menguji apakah fungsi untuk

menghapus data tempat ibadah sudah berjalan dengan baik dan benar. Pengujian fungsi tersebut dapat dilihat pada Tabel 4.18

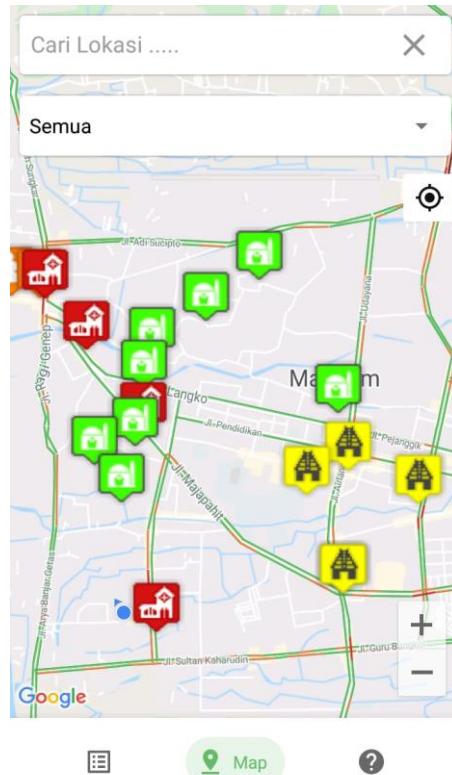
Tabel 4. 18 Pengujian Mengahpus Data Tempat Ibadah

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol “Cancel”	Data tidak terhapus	Sesuai	Diterima
Menekan tombol “Ok”	Data terhapus dalam sistem	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.18 dapat dikatakan system berhasil dan sesuai keinginan *client*.

15. US-10

Berikut merupakan implementasi tampilan dari sisi pengguna yang dapat melihat peta lokasi tempat ibadah pada basis *android* dari kasus kode *user stories* US-10 yang telah dibuat dapat dilihat pada Gambar 4.36 berikut:



Gambar 4.36 Tampilan Peta Lokasi Tempat Ibadah Basis *Android*

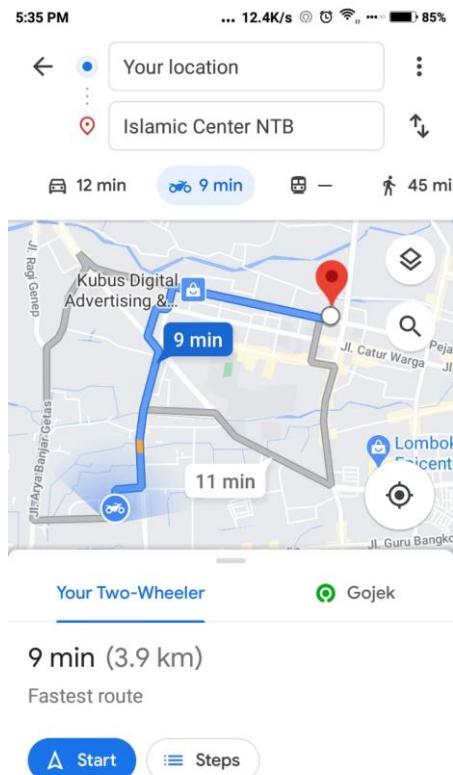
Gambar 4.36 merupakan implementasi halaman peta lokasi tempat ibadah yang akan ditampilkan dalam sistem basis *android*. Pada halaman ini masyarakat umum dapat melihat daftar tempat ibadah dalam bentuk peta digital. Pada halaman ini juga masyarakat dapat melakukan berbagai hal seperti mencari lokasi tempat ibadah, menampilkan lokasi tempat ibadah berdasarkan filter jenis tempat ibadah, melihat lokasi terkini pengguna, dapat melihat detail informasi tempat ibadah yang dipilih, dan dapat melihat rute lokasi tempat ibadah yang dipilih. Berikut merupakan table pengujian system terhadap kasus kode *user stories* US-10 dapat dilihat pada table 4.19.

Tabel 4. 19 Pengujian Peta Lokasi Tempat Ibadah Basis *Android*

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol halaman peta lokasi tempat ibadah	Peta lokasi tempat ibadah berhasil ditampilkan	Sesuai	Diterima
Memasukkan nama tempat ibadah atau <i>keyword</i> ke dalam fitur pencarian	Menampilkan hasil pencarian tempat ibadah berdasarkan <i>keyword</i> yang dimasukkan	Sesuai	Diterima
Memilih jenis tempat ibadah yang diinginkan berdasarkan filter jenis tempat ibadah	Peta lokasi tempat ibadah berhasil ditampilkan berdasarkan jenis tempat ibadah yang dipilih	Sesuai	Diterima
Menekan tombol lokasi terkini pengguna	Lokasi terkini pengguna berhasil ditampilkan	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.19 dapat dikatakan system berhasil dan sesuai kebutuhan *client*.

Berikut merupakan implementasi tampilan untuk dapat melihat peta rute menuju lokasi pada basis *android* dari kasus kode *user stories* US-11 yang telah dibuat dapat dilihat pada Gambar 4.37 berikut:



Gambar 4.37 Tampilan Peta Rute Meunuju Lokasi Tempat Ibadah Basis *Android*

Gambar 4.37 merupakan implementasi halaman peta lokasi tempat ibadah yang akan ditampilkan dalam sistem basis *android*. Berikut merupakan table pengujian system terhadap kasus kode *user stories* US-11 dapat dilihat pada table 4.20.

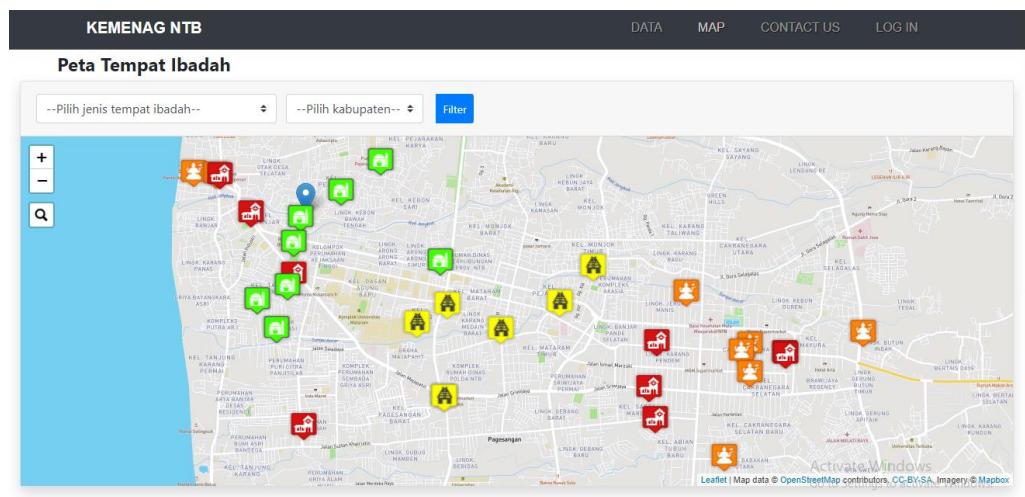
Tabel 4.20 Pengujian Peta Rute Meunuju Lokasi Tempat Ibadah Basis *Android*

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol rute menuju lokasi tempat ibadah	Rute menuju lokasi tempat ibadah berhasil ditampilkan	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.20 dapat dikatakan system berhasil dan sesuai kebutuhan *client*.

17. US-17

Berikut merupakan implementasi tampilan halaman peta lokasi tempat ibadah pada basis *web* dari kasus kode *user stories* US-17 yang telah dibuat dapat dilihat pada Gambar 4.38 berikut:



Gambar 4.38 Tampilan Halaman Peta Lokasi Tempat Ibadah Basis Web

Gambar 4.38 merupakan implementasi halaman peta lokasi tempat ibadah basis *web*. Pada halaman basis *web* ini masyarakat umum dapat melihat lokasi, serta mencari tempat ibadah berdasarkan jenis tempat ibadah dan kabupaten yang diinginkan yang tersaji dalam basis *web*. Berikut merupakan table pengujian system terhadap kasus kode *user stories* US-17 dapat dilihat pada table 4.21.

Tabel 4.21 Pengujian Peta Lokasi Tempat Ibadah Basis Web

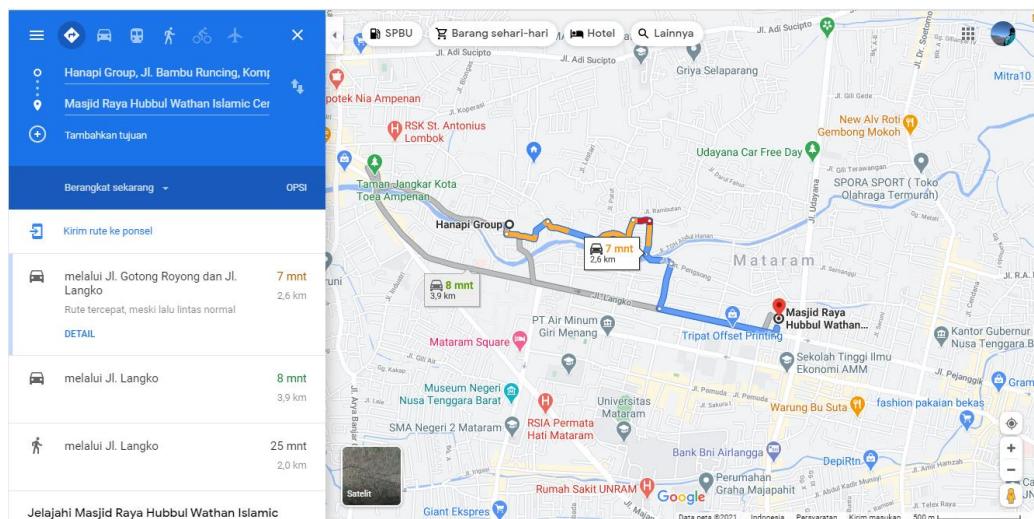
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol halaman lokasi tempat ibadah	Menampilkan data lokasi tempat ibadah dalam bentuk map	Sesuai	Diterima
Menekan salah satu lokasi tempat ibadah	Menampilkan deskripsi singkat dan terdapat tombol detail dan rute tempat ibadah	Sesuai	Diterima
Menekan tombol detail tempat ibadah	Data detail tempat ibadah berhasil ditampilkan	Sesuai	Diterima

Memilih kabupaten dan jenis tempat ibadah dan kemudian menekan tombol filter	Data tempat ibadah yang ditampilkan berdasarkan filter kabupaten dan jenis tempat ibadah	Sesuai	Diterima
--	--	--------	----------

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.21 dapat dikatakan system yang dibuat berhasil dan berjalan sesuai keinginan *client*.

18. US-18

Berikut merupakan implementasi tampilan halaman peta rute menuju lokasi tempat ibadah pada basis *web* dari kasus kode *user stories* US-18 yang telah dibuat dapat dilihat pada Gambar 4.39 berikut:



Gambar 4.39 Tampilan Halaman Peta Lokasi Tempat Ibadah Basis Web

Gambar 4.39 merupakan implementasi halaman peta rute menuju lokasi tempat ibadah basis *web*. Berikut merupakan table pengujian system terhadap kasus kode *user stories* US-18 dapat dilihat pada table 4.22.

Tabel 4.22 Pengujian Peta Rute Menuju Lokasi Tempat Ibadah Basis Web

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol rute menuju lokasi tempat ibadah	Rute menuju lokasi tempat ibadah berhasil ditampilkan	Sesuai	Diterima

Berdasarkan hasil pengujian dan kesimpulan dari pengujian yang telah dilakukan pada Tabel 4.22 dapat dikatakan system yang dibuat berhasil dan berjalan sesuai kebutuhan *client*.

4.1.3.5 Refactoring

Refactoring merupakan proses yang dilakukan untuk mengubah struktur atau menyerdahakan *code* tanpa harus mengubah fungsi fitur pada sistem yang telah dibuat. *Refactoring* biasa dilakukan ketika terdapat duplikasi baris *code*. Pada sistem yang telah dibuat, proses *refactoring* dilakukan pada Sumber Kode 4.3 berikut.

```
1. <?php
2. defined('BASEPATH') OR exit('No direct script access allowed')
;
3. class C_profile extends CI_Controller {
4.     public function __construct(){
5.         parent::__construct();
6.     }
7.     public function index (){
8.         if (!$this->session->userdata('id_username')) {
9.             redirect('c_login');
10.        }else {
11.            $role_bidang = $this->session-
>userdata('role');
12.            $menu = $this->uri->segment(1);
13.            if ($role_bidang == 2 && $menu == 'c_admin') {
14.                redirect('c_login/blocked');
15.            }
16.        }
17.    }
18. }
```

Kode Sumber 4.3 Sebelum Di *Refactor*

Baris *code* 8 sampai 17 pada Kode Sumber 4.3 merupakan baris *code* yang berguna pada sistem untuk menolak akses pengguna yang hendak ingin mengakses atau masuk ke halaman admin tanpa memasukkan *username* dan *password* yang ada pada halaman *login* admin. Dimana baris *code* tersebut digunakan di setiap *class controller* yang berhubungan dengan halaman admin seperti *class C_profile*, *C_beranda*, *C_tempat_ibadah*, dan *C_admin*. Untuk menghindari adanya duplikat *code* yang digunakan, maka pengembang melakukan *refactor* terhadap baris *code* 8 sampai 17 yang dimana baris *code* tersebut dibuatkan sebuah fungsi yang berguna sebagaimana mestinya yaitu untuk menolak akses pengguna yang hendak ingin mengakses atau masuk ke halaman admin tanpa memasukkan *username* dan

password yang ada pada halaman *login* admin. Dimana hasil *refactor* tersebut dapat dilihat pada Kode Sumber 4.4 berikut.

```
1. <?php
2. defined('BASEPATH') OR exit('No direct script access allowed')
;
3. class C_profile extends CI_Controller {
4.     public function __construct(){
5.         parent::__construct();
6.         cek_login();
7.     }
}
```

Kode Sumber 4.4 Setelah Di *Refactor*

Kode Sumber 4.4 merupakan hasil *refactoring* yang telah dilakukan dimana pada baris *code* 6 merupakan sebuah fungsi yang berguna sebagaimana telah dipaparkan sebelumnya. Dibandingkan dengan Kode Sumber 4.3 sebelum di *refactor*, Kode Sumber 4.4 setelah di *refactor* jauh lebih singkat dan efisien dalam penggunaannya, sebab nantinya hanya fungsi inilah yang akan dipanggil pada setiap kelas *controller* yang berhubungan dengan halaman admin agar pengguna yang hendak ingin mengakses halaman admin tanpa melakukan *login* akan ditolak oleh sistem.

4.1.4 Testing

Walaupun pengujian system telah dilakukan pengujian sebelumnya, namun system harus tetap diuji lagi bermaksud untuk memastikan agar system yang dibuat tidak terjadi *error* atau *bug* pada system dan sesuai dengan keinginan *client*. Pengujian pada tahap ini juga dilakukan tidak hanya untuk mengetahui adanya *error* atau tidak namun juga untuk mengetahui kualitas sistem dari sisi pengguna.

4.1.4.1 Hasil pengujian *black box*

Metode pengujian *black box* merupakan metode pengujian yang menguji fungsi - fungsi di dalam sistem untuk menentukan apakah fungsi-fungsi tersebut sudah berjalan sesuai harapan atau tidak. Pada pengujian ini, fungsi – fungsi pada fitur yang ada pada sistem diuji secara langsung oleh pihak kemenag NTB pada bidang bimas islam yaitu kepada ibu Haiya baidariyyah dan bidang inmas yaitu kepada Drs.H. Saipun Nasri, M.Pd.

Berikut merupakan hasil pengujian sistem yang telah dibuat dapat dilihat pada Tabel 4.23.

Tabel 4.23 Hasil Pengujian *Black Box*

No	Fitur	Ditunjukan tabel	Hasil uji
1.	Login ke dalam system	Tabel 4.4	Diterima
2.	Mengakhiri system (<i>logout</i>)	Tabel 4.5	Diterima
3.	Melihat dan mengubah profile admin <i>login</i>	Tabel 4.6	Diterima
4.	Melihat daftar data admin	Tabel 4.7	Diterima
5.	Menambah data admin	Tabel 4.8	Diterima
6.	Mengubah data admin	Tabel 4.9	Diterima
7.	Menghapus data admin	Tabel 4.10	Diterima
8.	Menambah data tempat ibadah	Tabel 4.11	Diterima
9.	Mengimport data <i>excel</i> tempat ibadah	Tabel 4.12	Diterima
10.	mengunduh template format <i>excel</i> import data	Tabel 4.13	Diterima
11.	menexport data <i>excel</i> tempat ibadah	Tabel 4.14	Diterima
12.	Melihat daftar tempat ibadah	Tabel 4.15	Diterima
13.	Melihat Beranda halaman sisi admin	Tabel 4.16	Diterima
14.	Mengubah data tempat ibadah	Tabel 4.17	Diterima
15.	Menghapus data tempat ibadah	Tabel 4.18	Diterima
16.	Melihat peta lokasi tempat ibadah basis <i>android</i>	Tabel 4.19	Diterima
17.	Melihat rute lokasi tempat ibadah basis <i>android</i>	Tabel 4.20	Diterima
18.	Melihat peta lokasi tempat ibadah basis <i>web</i>	Tabel 4.21	Diterima
19.	Melihat rute lokasi tempat ibadah basis <i>web</i>	Tabel 4.22	Diterima

Berdasarkan table hasil pengujian *black box* yang telah dipaparkan pada Tabel 4.23 dapat dikatakan bahwa sistem yang dibuat telah berhasil berjalan dengan baik dan benar sesuai perancangan serta telah sesuai dengan kebutuhan *client*

4.1.4.2 Hasil pengujian *Mean Opinion Score (MOS)*

Pengujian ini bertujuan untuk mengetahui bagaimana kualitas sistem dilihat dari sisi pengguna. Pada pengujian ini menggunakan kuesioner dengan mencari responden untuk mencoba menjalankan sistem, lalu memberikan pernyataan berupa kuesioner. Penyebaran kuesioner dilakukan menggunakan *google form*. Sebelum diberikannya pertanyaan, hal yang pertama dilakukan yaitu memberikan penjelasan tentang kegunaan dari sistem yang telah dibuat serta alur kerja dari sistem yang dibuat.

Pengujian ini dilakukan oleh 33 responden yang terdiri dari 2 pegawai kemenag NTB sebagai admin, dan sisanya dipilih secara acak yang terdiri dari 31 orang sebagai pengguna aplikasi. Hasil dari jawaban responden nantinya akan dihitung dan ditarik kesimpulan mengenai hasil pengujian sistem. Berikut daftar pertanyaan yang diajukan kepada responden:

1. Admin
 - a. Apakah tampilan aplikasi mudah dipahami?
 - b. Apakah aplikasi mudah digunakan?
 - c. Apakah aplikasi ini dapat membantu pihak kemenag NTB dalam mengelola data tempat ibadah yang ada?
 - d. Apakah aplikasi sudah sesuai dengan kebutuhan pengguna?
 - e. Apakah aplikasi berjalan dengan baik?
 - f. Apakah fitur yang disediakan dapat mempermudah pengaksesan aplikasi?
2. Masyarakat
 - a. Apakah tampilan aplikasi mudah dipahami?
 - b. Apakah aplikasi mudah digunakan?
 - c. Apakah aplikasi ini dapat membantu masyarakat dalam mencari informasi dan lokasi mengenai tempat ibadah yang diinginkan?
 - d. Apakah aplikasi sudah sesuai dengan kebutuhan pengguna?
 - e. Apakah aplikasi berjalan dengan baik?
 - f. Apakah fitur yang disediakan dapat mempermudah pengaksesan aplikasi?

Responden diminta untuk menjawab kuesioner tersebut dengan pilihan jawaban sebagai berikut:

1 = Tidak Setuju

2 = Kurang Setuju

3 = Cukup

4 = Setuju

5 = Sangat Setuju

Berdasarkan jawaban tersebut, dilakukan perhitungan persentase dari setiap jawaban yang diberikan oleh pengguna di setiap pertanyaannya. Rumus untuk menghitung persentase jawaban adalah sebagai berikut:

Persentase = $J / N * 100\%$

J : Total nilai jawaban yang diberikan pengguna pada setiap nomor.

N : Jumlah pengguna.

Selanjutnya dilakukan proses perhitungan persentase rata-rata dari setiap poin jawaban, perhitungan yang digunakan adalah menggunakan rumus sebagai berikut:

$$\text{Rata-rata} = (\text{Persentase ke-1} + \dots + \text{Persentase ke-6}) / 6$$

Persentase jawaban responden terhadap kuesioner yaitu, tidak setuju, kurang setuju, cukup, setuju dan sangat setuju dapat dihitung secara keseluruhan dan didefinisikan sebagai nilai rata-rata. Dengan menggunakan rumus perhitungan diatas, perhitungan untuk hasil jawaban pengguna pada setiap pertanyaan pada kuesioner adalah sebagai berikut:

1. Admin

Pada Tabel 4.24 merupakan hasil pengujian kuisioner yang telah dilakukan terhadap 2 pegawai kemenag NTB yaitu pada bidang bimas islam yaitu kepada ibu Haiya baidariyyah dan bidang inmas yaitu kepada Drs.H. Saipun Nasri, M.Pd. Hal yang pertama dilakukan yaitu menjelaskan fitur-fitur apa saja yang terdapat dalam sistem yang telah dibuat. Kemudian pegawai kemenag NTB mencoba menggunakan sistem dan mengisi kuisioner yang ada. Sehingga didapatkan hasil bahwa rata-rata yang menjawab sangat setuju adalah 66,67%, setuju 25%, Cukup 8,33%, Kurang setuju 0% dan Tidak Setuju adalah 0%

Keterangan :

Pertanyaan 1 = Apakah tampilan sistem informasi mudah dipahami?

Pertanyaan 2 = Apakah sistem mudah digunakan?

Pertanyaan 3 = Apakah aplikasi ini dapat membantu pihak kemenag NTB dalam mengelola data tempat ibadah yang ada di aplikasi?

Pertanyaan 4 = Apakah aplikasi sudah sesuai dengan kebutuhan anda?

Pertanyaan 5 = Apakah aplikasi berjalan dengan baik?

Pertanyaan 6 = Apakah fitur yang disediakan dapat mempermudah pengaksesan aplikasi?

SS = Sangat setuju, S = Setuju, C = Cukup, KS = Kurang Setuju, Ts = Tidak Setuju

Tabel 4.24 Hasil Pengujian Kuisisioner Dengan Pegawai KEMENAG NTB

Pertanyaan	SS		S		C		KS		TS	
	total	%	total	%	total	%	total	%	total	%
1	1	50	1	50	0	0	0	0	0	0
2	1	50	1	50	0	0	0	0	0	0
3	2	100	0	0	0	0	0	0	0	0
4	2	100	0	0	0	0	0	0	0	0
5	1	50	1	50	0	0	0	0	0	0
6	1	50	0	0	1	50	0	0	0	0
Rata-rata	66,67%		25%		8,33%		0%		0%	

2. Masyarakat

Pada Tabel 4.25 merupakan hasil pengujian kuisioner yang telah dilakukan terhadap 31 orang. Hal yang pertama dilakukan yaitu menjelaskan fitur-fitur apa saja yang terdapat dalam sistem yang telah dibuat. Kemudian masyarakat diminta mencoba menggunakan sistem dan mengisi kuesioner yang ada. Sehingga didapatkan hasil bahwa rata-rata yang menjawab sangat setuju adalah 40,33%, setuju 47,83%, Cukup 11,83%, Kurang setuju 0% dan Tidak Setuju adalah 0%

Keterangan :

Pertanyaan 1 = Apakah tampilan sistem informasi mudah dipahami?

Pertanyaan 2 = Apakah sistem mudah digunakan?

Pertanyaan 3 = Apakah aplikasi ini dapat membantu masyarakat mengetahui informasi mengenai tempat ibadah yang dengan mudah?

Pertanyaan 4 = Apakah aplikasi sudah sesuai dengan kebutuhan masyarakat?

Pertanyaan 5 = Apakah aplikasi berjalan dengan baik?

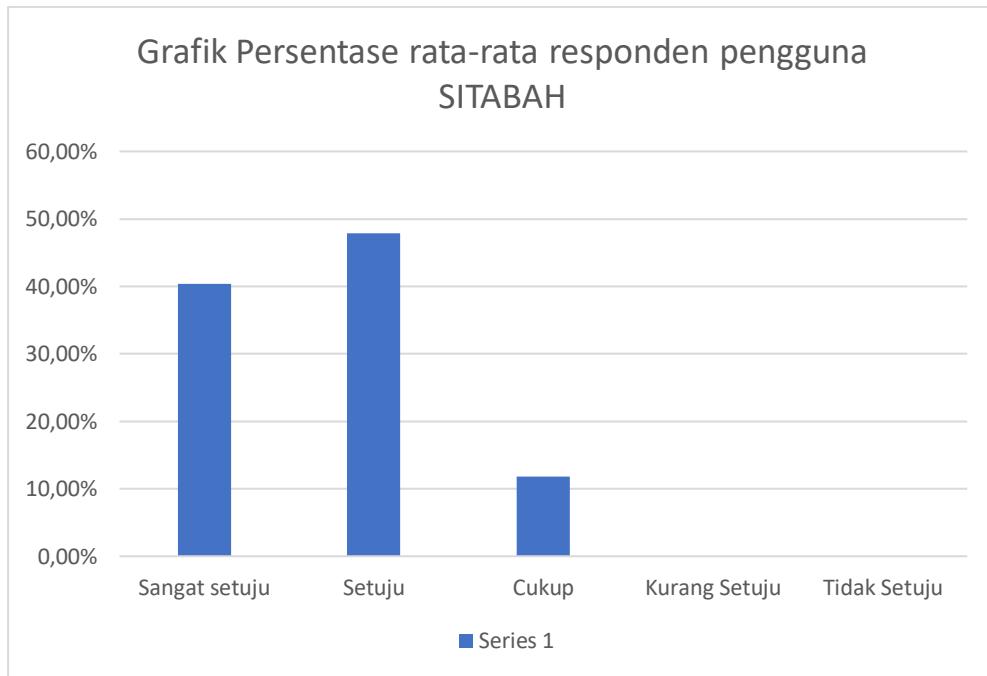
Pertanyaan 6 = Apakah fitur yang disediakan dapat mempermudah pengaksesan aplikasi?

SS = Sangat setuju, S = Setuju, C = Cukup, KS = Kurang Setuju, Ts = Tidak Setuju

Tabel 4.25 Hasil Pengujian Kuisioner Dengan Masyarakat

Pertanyaan	SS		S		C		KS		TS	
	total	%	total	%	total	%	total	%	total	%
1	11	35,5	16	51,6	4	12,9	0	0	0	0
2	12	38,7	18	58,1	1	3,2	0	0	0	0
3	15	48,4	14	45,2	2	6,5	0	0	0	0
4	14	45,2	13	41,9	4	12,9	0	0	0	0
5	9	29	12	38,7	10	32,3	0	0	0	0
6	14	45,2	16	51,6	1	3,2	0	0	0	0
Rata-rata	40,33%		47,85%		11,83%		0%		0%	

Dapat dilihat pada Gambar 4.40 yang merupakan grafik persentase rata-rata responden yang telah menggunakan Aplikasi SITABAH, dapat disimpulkan bahwa 40,33% sangat setuju, 47,85% setuju, dan 11,83% cukup setuju terkait dengan kemudahan dalam mengakses aplikasi, tampilan mudah di fahami, aplikasi berjalan dengan baik, membantu pengguna dalam menemukan informasi mengenai tempat ibadah dengan mudah.



Gambar 4.40 Grafik Perentase Rata-Rata Responden Pengguna Aplikasi SITABAH

4.2 Hasil evaluasi metode *extreme programming*

Proses pengembangan sistem selama 6 iterasi yang telah dilakukan pengembang berupa pengerjaan *user stories* yang melebihi estimasi pengerjaan yang direncanakan. Pada proses pengembangan sistem dilakukan, terdapat *user stories* yang pengerjaannya melebihi waktu estimasi yang ditentukan yang disebabkan karena *device* yang digunakan mengalami masalah serta kurangnya pengalaman pengembang dalam mengerjakan modul sehingga membuat pengembang tidak dapat mengestimasikan waktu pengerjaan secara tepat.

Adanya penambahan *user stories* oleh *client* juga menyebabkan terjadinya penambahan 3 iterasi seperti yang telah dipaparkan pada Tabel 4.2. Hal itu tentunya menyebabkan perencanaan keseluruhan waktu pengerjaan sistem dari bertambah 30 hari, dari estimasi awal 30 hari menjadi 60 hari.

Berikut merupakan analisa nilai terhadap metode *extreme programming* yang telah diimplementasikan selama proses pengembangan sistem berlangsung dapat dilihat pada Tabel 4.26.

Tabel 4.26 Hasil Analisa Nilai Metode *Extreme Programming*

Variabel	Analisa
<i>Communication</i>	Komunikasi dengan klien memudahkan pengembang untuk mengetahui sistem yang diinginkan klien (<i>user stories</i>).

	Komunikasi terjadi setiap kali sebuah <i>user stories</i> atau fitur selesai dikerjakan. Dimana yang menjadi penghambat pada kegiatan komunikasi ini yaitu menyesuaikan waktu pertemuan dengan klien.
<i>Couregae</i>	Keberanian diterapkan pada saat adanya permintaan penambahan fitur oleh klien. Hambatan yang terjadi ketika penambahan fitur yaitu pengembang harus menerima setiap perubahan, sehingga ketika terdapat fitur yang diminta dirasa cukup sulit untuk dikerjakan, maka akan mengakibatkan waktu penggerjaan yang dibutuhkan akan menjadi lebih lama.
<i>Simplicity</i>	Pase ini merupakan kesederhanaan yang diterapkan pada saat proses <i>design</i> . Bagi pengembang dilakukan perancangan yang sederhana bermaksud untuk mengurangi waktu proses penggerjaan. Dan kesederhaan juga diterapkan pada saat <i>coding</i> , agar baris <i>code</i> yang dibuat mudah dipahami.
<i>Feedback</i>	Timbal balik termasuk dalam komunikasi, karena pada <i>feedback</i> dari klien inilah pengembang mengetahui ada atau tidaknya masukan yang akan diberikan terkait dengan system diinginkan.

Berikut merupakan rincian waktu pelaksanaan yang dilakukan dapat dilihat pada Tabel 4.27.

Tabel 4.27 rincian waktu penggerjaan

Dilihat pada Tabel 4.27 bahwa terdapat beberapa user story yang setelah selesai dikerjakan dapat langsung diterima oleh client dan ada juga yang tidak. seperti yang terlihat pada Tabel 4.27 terdapat beberapa user story yang memiliki lebih dari satu cycle yang artinya user story tersebut masih perlu ditambahkan fitur fungsinya.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian dan pembangunan sistem yang telah dilakukan, dapat ditarik beberapa kesimpulan diantaranya:

1. Sistem yang dibuat dapat menyampaikan informasi dan lokasi mengenai data tempat ibadah di Kota Mataram berbasis *android* dan data tempat ibadah se-NTB berbasis *web* yang dapat memberikan jalur perjalanan menuju tempat ibadah yang diinginkan dan dapat mengetahui persebaran lokasi tempat ibadah yang ada di provinsi NTB.
2. Dengan adanya sistem yang telah dibuat ini dapat memberikan manfaat bagi masyarakat umum dalam mengetahui informasi dan lokasi tempat ibadah yang ada di Kota Mataram dengan adanya fitur pendukung seperti fitur pencarian, fitur filter berdasarkan kategori jenis tempat ibadah dan filter kabupaten, serta fitur rute menuju lokasi tempat ibadah yang diinginkan. Dapat dibuktikan dengan hasil kuisioner yang diberikan pada Tabel 4.25 yang menunjukkan bahwa 40,33% sangat setuju, 47,85% setuju, dan 11,83% cukup setuju.
3. Dapat membantu mempermudah Kementerian Agama Provinsi NTB dalam melakukan pengelolaan dan pencarian terhadap data tempat ibadah yang ada. Terbukti dari hasil kuisioner yang telah dilakukan oleh 2 pegawai pihak kemenag NTB pada Tabel 4.24 menunjukkan bahwa 66,67% sangat setuju, 25% setuju, dan 8,33% adalah Cukup terkait dengan kemudahan dalam mengakses aplikasi, tampilan mudah di fahami, aplikasi berjalan dengan baik, membantu pihak kemenag dalam mengelola data tempat ibadah yang ada.
4. Metode *Extreme Programming* cocok digunakan pada pembuatan aplikasi tempat ibadah ini, karena salah satu karakteristik XP itu sendiri melibatkan *client* selama proses pembangunan sistem sedangkan dalam kasus penelitian Tugas Akhir ini pihak Kemenag NTB sebagai *client* yang terlibat langsung dalam proses perancangan dan pembangunan sistem sehingga berdasarkan hasil pengujian yang telah dilakukan, dapat dikatakan sistem yang dibuat berjalan dengan benar dan telah sesuai dengan kebutuhan *client*. Terbukti dari dapat dilihat pada hasil pengujian *black box* pada Tabel 4.23 menunjukkan bahwa seluruh fungsi bekerja dengan baik pada system dan diterima sehingga system yang dibuat telah berjalan sesuai dengan fungsi yang dirancang.

5.2 Saran

Dikarenakan masih banyak kekurangan dari penulis dalam mengembangkan sistem ini, terdapat beberapa saran untuk dapat membuat sistem ini menjadi lebih baik pada waktu yang akan datang, diantaranya:

1. *Device* yang digunakan dalam membuat sistem harus diperhatikan terlebih dahulu, karena dalam XP jika terjadi kerusakan *device* pada saat proses pembuatan sistem berlangsung, maka akan mengganggu estimasi penggerjaan system menjadi lebih lama dari yang telah direncanakan.
2. *Skill programming* pengembang untuk lebih ditingkatkan lagi, agar pada saat proses pengembangan sistem berlangsung menjadikan waktu estimasi penggerjaan yang direncanakan lebih akurat.
3. Sebaiknya aplikasi pada basis *android* tidak hanya diterapkan di Kota Mataram saja, melainkan dapat diterapkan di provinsi atau di kota-kota lainnya.
4. Sebaiknya sistem yang dibuat disertai dengan fitur dua Bahasa, yaitu Bahasa Inggris dan Bahasa Indonesia untuk membantu pengguna sistem yang tidak berasal dari Indonesia.
5. Pada menu map basis *web* atau *android*, sebaiknya ditambahkan *polygon* atau *polyline* pada setiap kabupaten atau kecamatan menjadikan sistem informasi yang lebih efektif dalam pembacaan wilayahnya.
6. Desain *interface* lebih dioptimalkan lagi terutama pada gradasi warna aplikasi *mobile*-nya.
7. Pada sistem ditampilkan panduan penggunaan aplikasinya agar mempermudah penggunaan sistem bagi orang awam.

DAFTAR PUSTAKA

- [1] *Kamus Besar Bahasa Indonesia*. Jakarta: PUSAT BAHASA DEPARTEMEN PENDIDIKAN NASIONAL, 2008.
- [2] Komite Teknis Perumusan Standar Pelayanan Masyarakat pada Kementerian Lingkungan Hidup dan Kehutanan, “Standar Pelayanan Masyarakat pada Fasilitas Publik Standar.” 2016.
- [3] M. Agama and D. Negeri, “Pedoman Pelaksanaan Tugas Kepala Daerah/Wakil Kepala Daerah Dalam Pemeliharaan Kerukunan Umat Beragama, Pemberdayaan Forum Kerukunan Umat Beragama, Dan Pendirian Rumah Ibadat,” 2006.
- [4] G. S. Perdana, “Sistem informasi geografis tempat olahraga di provinsi daerah istimewa yogyakarta berbasis web,” 2017.
- [5] Sukatmi and F. A. Ani, “Membangun Aplikasi Webgis Untuk Tempat Ibadah Di Provinsi Lampung,” *Cendikia*, vol. XVI, pp. 115–119, 2018.
- [6] A. Supriyatna, “Metode Extreme Programming Pada Pembangunan Web Aplikasi Seleksi Peserta Pelatihan Kerja,” *J. Tek. Inform.*, vol. 11, no. 1, pp. 1–18, 2018, doi: 10.15408/jti.v11i1.6628.
- [7] R. Pamungkas, *Implementasi Model Personal Extreme Programming (Pxp) Dalam Pengembangan Sistem Informasi Geografis Pemetaan Flora Dan Fauna Di Taman Nasional Meru Betiri*. 2018.
- [8] D. Y. Prasetyo and Apriyanto, “Implementasi Geographic Information System (GIS) Penentuan Tampat Ibadah Masjid Di Kecamatan Kempas Kabupaten Indragiri Hilir Provinsi Riau,” *SISTEMASI*, vol. 8, pp. 10–18, 2019.
- [9] N. Alamsyah and Wagino, “Sistem Informasi Geografis Persebaran Tempat Ibadah Dan Kapasitasnya Berdasarkan Jumlah Penduduk Di Kota Banjarmasin,” *Technologia*, vol. 9, no. 2, pp. 133–138, 2018.
- [10] C. PARAMITA, “Sistem informasi masjid di surakarta,” 2016.
- [11] S. Maharani, D. Apriani, A. H. Kridalaksana, P. Studi, I. Komputer, and U. Mulawarman, “Sistem informasi geografis pemetaan masjid di samarinda berbasis web,” vol. 11, pp. 9–20, 2017.

- [12] W. N. NUGROHO, “Aplikasi Pencarian Masjid Terdekat Di Kota Bandar Lampung Berbasis Mobile Menggunakan Algoritma Dijkstra,” 2017.
- [13] R. S. HAMSYAH, “Rancang Bangun Aplikasi Go-Ban Untuk Mencari Dan Memanggil Teknisi Tambal Ban Menggunakan Google Maps Api,” 2018.
- [14] Suhartini, M. Sadali, and Y. K. Putra, “Sistem Informasi Berbasis Web Sma Al-Mukhtariyah Mamben Lauk Berbasis Php Dan Mysql Dengan Framework Codeigniter,” *infotek*, vol. 3, no. 1, pp. 79–84, 2020.
- [15] M. Arifin and R. H. H. Hs, “Perancangan Sistem Informasi Pusat Karir Sebagai Upaya Meningkatkan Relevansi Antara Lulusan Dengan Dunia Kerja Menggunakan UML,” *IC-Tech*, vol. XII, no. 2, pp. 42–49, 2017.
- [16] T. Sutabri, *KONSEP SISTEM INFORMASI*. Yogyakarta, 2012.
- [17] A. Solichin, “Pemrograman Web dengan PHP dan MySQL,” pp. 1–122.
- [18] B. Utama, “Aplikasi Peminjaman Dan Pengembalian Barang Menggunakan Rfid Dan Barcode Scanner Berbasis Web,” 2019.
- [19] T. Kahlert and K. Giza, “Visual Studio Code - Code Editing. Redefined,” *Mikrosoft*, vol. 1, no. March, pp. 1–26, 2016.
- [20] A. Krismadi, A. F. Lestari, A. Pitriyah, I. W. P. A. Mardangga, M. Astuti, and A. Saifudin, “Pengujian Black Box berbasis Equivalence Partitions pada Aplikasi Seleksi Promosi Kenaikan Jabatan,” *J. Teknol. Sist. Inf. dan Apl.*, vol. 2, no. 4, p. 155, 2019, doi: 10.32493/jtsi.v2i4.3771.
- [21] W. Widjantoro, “Buku Panduan Bootstrap,” 2015.
- [22] M. Lenz, “Postman,” *Python Contin. Integr. Deliv.*, pp. 1–12, 2019, doi: 10.1007/978-1-4842-4281-0_1.
- [23] M. Ichwan and F. Hakiky, “Pengukuran Kinerja Goodreads Application Programming Interface (API) Pada Aplikasi Mobile Android (Studi Kasus Untuk Pencarian Data Buku),” vol. 2, no. 2, pp. 13–21, 2011.
- [24] J. Andi, ‘‘Pembangunan Aplikasi Child Tracker Berbasis Assisted – Global Positioning System (A-GPS) Dengan Platform Android,’’ *J. Ilm. Komput. dan Inform.*, vol. 1, no. 1, pp. 1–8, 2015.

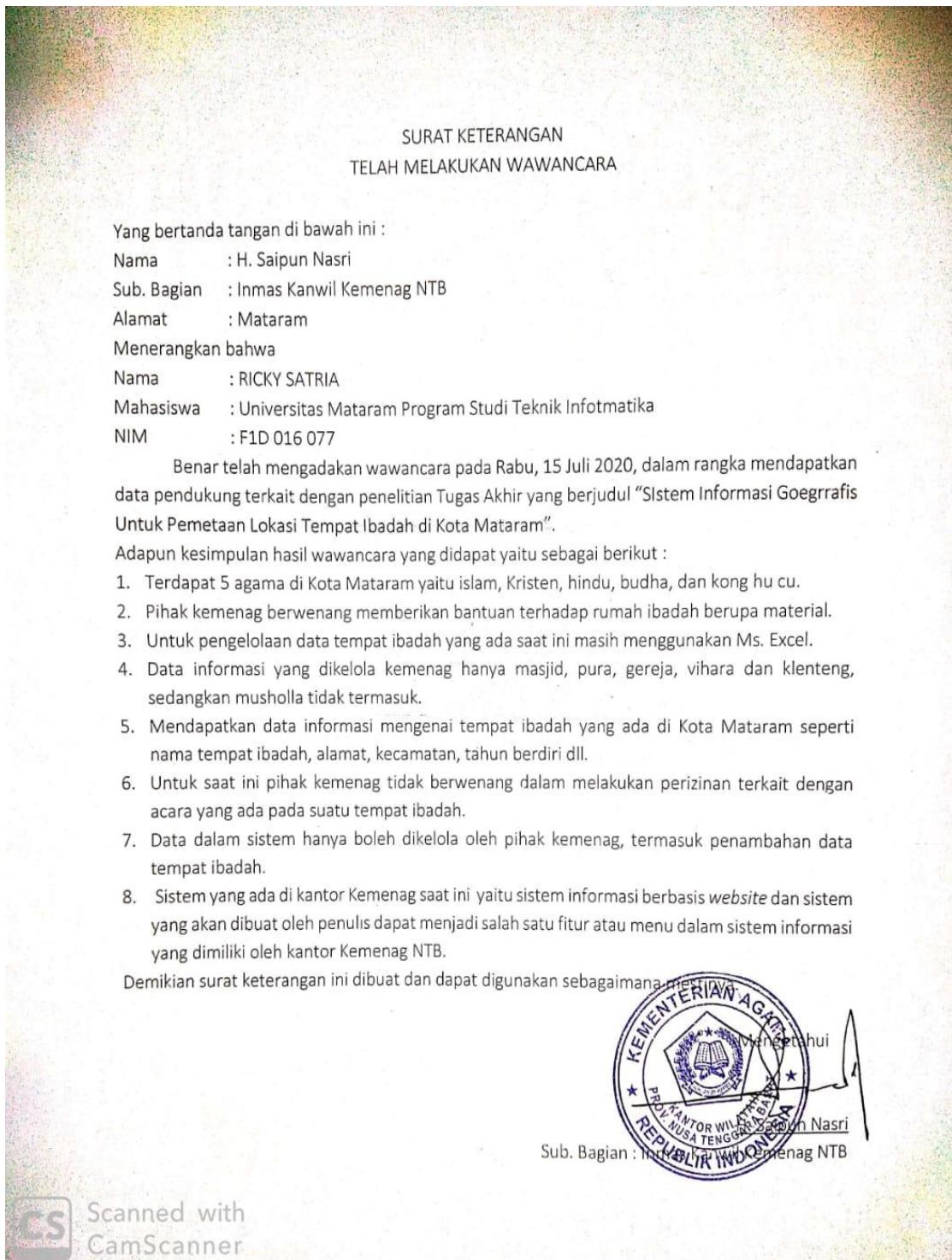
- [25] P. Abrahamson, Outi Salo, Jussi Ronkainen, and Juhani Warsta, “Agile software development methods: Review and analysis,” *VTT Publ.*, p. 112, 2002.
- [26] A. Prasetya, A. H. Brata, and M. T. Ananta, “Pengembangan Aplikasi Pemesanan Lapangan Futsal Di Kota Malang Berbasis Android Menggunakan Metode Pengembangan Extreme Programming (Studi Kasus Champion Tidar, Zona SM Futsal, dan Viva Futsal),” *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 2, no. 12, pp. 7293–7301, 2018.
- [27] A. Lisna, “Extreme Programming,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2016, doi: 10.1017/CBO9781107415324.004.
- [28] M. Ulfî, G. I. Marthasari, and N. Ilyas, “Implementasi Metode Personal Extreme Programming dalam Pengembangan Sistem Manajemen Transaksi Perusahaan (Studi Kasus: CV. Todjoe Sinar Group),” vol. 2, no. 3, pp. 261–268, 2020.
- [29] R. Agarwa and D. Umphress, “Extreme programming for a single person team,” *Proc. 46th Annu. Southeast Reg. Conf. XX, ACM-SE 46*, no. August, pp. 82–87, 2008, doi: 10.1145/1593105.1593127.
- [30] R. A. Azdy and A. Rini, “Penerapan Extreme Programming Dalam Membangun Aplikasi Pengaduan Layanan Pelanggan (Palapa) Pada Perguruan Tinggi,” vol. 5, no. 2, pp. 197–206, 2018, doi: 10.25126/jtiik.201852658.
- [31] M. A. F. Z and A. Wibowo, “Aplikasi gameplay edukasi pencegahan obesitas dengan menggunakan algoritma astar dan greedy pada pencarian jalur makanan,” vol. 10, no. 2016.
- [32] A. Mu, “BAB III, metodologi penelitian,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2019, doi: 10.1017/CBO9781107415324.004.
- [33] A. N. A. Thohari and A. E. Amalia, “Implementasi Test Driven Development Dalam Pengembangan Aplikasi Berbasis Web,” *SITECH J. Sist. Inf. dan Teknol.*, vol. 1, no. 1, pp. 1–10, 2018, doi: 10.24176/sitech.v1i1.2255.
- [34] S. Masripah and L. Ramayanti, “Penerapan Pengujian Alpha Dan Beta Pada Aplikasi Penerimaan Siswa Baru,” *J. Swabumi*, vol. 8, no. 1, pp. 100–105, 2020.
- [35] A. B. Mutiara, U. Gunadarma, A. Muslim, U. Gunadarma, T. Oswari, and U.

Gunadarma, “Testing Implementasi Website Rekam Medis Elektronik Opeltnasys Dengan Metode Acceptance Testing,” no. February 2016, 2014, doi: 10.13140/RG.2.1.1159.5286.

- [36] R. A. Lestari, “Pengaruh Kepemimpinan Partisipatif D an Komitmen Organisasi Terhadap Efektifitas Implementasi Rencana Stratejik Pada Madrasah Aliyah D i Kabupaten Sukabumi,” pp. 1–25, 2014.
- [37] A. Suandi, F. N. Khasanah, and E. Retnoningsih, “Pengujian Sistem Informasi E-commerce Usaha Gudang Cokelat Menggunakan Uji Alpha dan Beta,” *Inf. Syst. Educ. Prof.*, vol. 2, no. 1, pp. 61–70, 2017.

LAMPIRAN

Surat keterangan telah melakukan wawancara :



Scanned with
CamScanner

Keterangan telah melakukan kuesioner terhadap pihak kemenag NTB

KUESIONER HASIL PENGUJIAN
SISTEM INFORMASI GEOGRAFIS PEMETAAN LOKASI TEMPAT IBADAH DI KOTA MATARAM

Nama : Drs. H. Saipun Nasri, M.Pd

Jabatan: Pranata Humas

Tabel kuesioner

No	Pertanyaan	SS	S	C	KS	TS
1	Apakah tampilan sistem informasi mudah dipahami?	✓				
2	Apakah sistem mudah digunakan?	✓				
3	Apakah aplikasi ini dapat membantu pihak kemenag NTB dalam mengelola data tempat ibadah yang ada di aplikasi?	✓				
4	Apakah aplikasi sudah sesuai dengan kebutuhan anda?	✓				
5	Apakah aplikasi berjalan dengan baik?	✓				
6	Apakah fitur yang disediakan dapat mempermudah pengaksesan aplikasi?			✓		

Keterangan :

SS : Sangat Setuju

S : Setuju

C : Cukup

KS : Kurang Setuju

TS : Tidak Setuju



KUESIONER HASIL PENGUJIAN
SISTEM INFORMASI GEOGRAFIS PEMETAAN LOKASI TEMPAT IBADAH DI KOTA MATARAM

Nama : Haya Baydariyyah
Jabatan: Pejabat Bidang Iman Islam

Tabel kuesioner

No	Pertanyaan	SS	S	C	KS	TS
1	Apakah tampilan sistem informasi mudah dipahami?	✓				
2	Apakah sistem mudah digunakan?	✓				
3	Apakah aplikasi ini dapat membantu pihak kemenag NTB dalam mengelola data tempat ibadah yang ada di aplikasi?		✓			
4	Apakah aplikasi sudah sesuai dengan kebutuhan anda?		✓			
5	Apakah aplikasi berjalan dengan baik?	✓				
6	Apakah fitur yang disediakan dapat mempermudah pengaksesan aplikasi?	✓				

Keterangan :

SS : Sangat Setuju

S : Setuju

C : Cukup

KS : Kurang Setuju

TS : Tidak Setuju

Mataram, 25 Januari 2021

