



**TUGAS AKHIR - IS 184853**

**RANCANG BANGUN APLIKASI ANTRIAN PINTAR  
(STUDI KASUS : SERVICE DESK DPTSI)**

***DESIGN AND DEVELOPMENT OF SMART QUEUE  
MANAGEMENT SYSTEM (STUDY CASE : SERVICE  
DESK DPTSI)***

Arief Putranto

NRP 0521 13 40000 124

Dosen Pembimbing

Dr. Eng. Febriliyan Samopa, S.Kom, M.Kom.

DEPARTEMEN SISTEM INFORMASI

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya 2020

**FINAL PROJECT - IS 184853**

**DESIGN AND DEVELOPMENT OF SMART QUEUE  
MANAGEMENT SYSTEM (STUDY CASE : SERVICE  
DESK DPTSI)**

**Arief Putranto**

**NRP 0521 13 40000 124**

**Academic Promotor**

**Dr. Eng. Febriliyan Samopa, S.Kom, M.Kom.**

**INFORMATION SYSTEM DEPARTMENT**

**Faculty of Intelligent Electrical and Informatics Technology**

**Institute Technology Sepuluh Nopember**

**Surabaya 2020**

**FINAL PROJECT - IS 184853**

**RANCANG BANGUN APLIKASI ANTRIAN PINTAR  
(STUDI KASUS : SERVICE DESK DPTSI)**

**Arief Putranto**

**NRP 0521 13 40000 124**

**Dosen Pembimbing**

**Dr. Eng. Febriliyan Samopa, S.Kom, M.Kom.**

**DEPARTEMEN SISTEM INFORMASI**

**Fakultas Teknologi Elektro dan Informatika Cerdas**

**Institut Teknologi Sepuluh Nopember**

**Surabaya 2020**

**LEMBAR PENGESAHAN****RANCANG BANGUN APLIKASI ANTRIAN PINTAR  
(STUDI KASUS : SERVICE DESK DPTSI)****TUGAS AKHIR**

Disusun Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer (S.Kom)  
pada  
Departemen Sistem Informasi

Fakultas Teknologi Elektro dan Informatika Cerdas (ELECTICS)  
Institut Teknologi Sepuluh Nopember

Oleh

**Arief Putranto**

**05211340000124**

Surabaya, 14 Agustus 2020

**Kepala Departemen Sistem Informasi**

**Dr. Muljahidin, ST., MT.**  
**NIP. 197010102003121001**



**LEMBAR PERSETUJUAN**  
**RANCANG BANGUN APLIKASI ANTRIAN PINTAR**  
**(STUDI KASUS : SERVICE DESK DPTS)**

**TUGAS AKHIR**

Disusun Untuk Memenuhi Salah Satu Syarat Memperoleh Gelar  
Sarjana Komputer

Pada

Departemen Sistem Informasi

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Oleh :

**ARIEF PUTRANTO**

**NRP. 0521 13 40000 124**

Disetujui Tim Penguji : Tanggal Ujian : 17 Juli 2020

Periode Wisuda: September 2020

**Dr. Eng. Febriliyan Samopa, S.Kom, M.Kom. (Pembimbing I)**

**Bekti Cahyo Hidayanto, S.Si, M.Kom.**

**(Penguji I)**

**Dr. Bambang Setiawan, S.Kom., M.T.**

**(Penguji II)**

## **RANCANG BANGUN APLIKASI ANTRIAN PINTAR (STUDI KASUS : SERVICE DESK DPTSI)**

**Nama Mahasiswa** : Arief Putranto  
**NRP** : 0521 13 40000 124  
**Departemen** : Sistem Informasi  
**Pembimbing** : Dr. Eng. Febriliyan Samopa,  
S.Kom, M.Kom.

### **ABSTRAK**

*Antrian merupakan hal yang banyak kita jumpai dimana saja, contohnya saja ketika kita berbelanja di toko ataupun dimana saja kita harus menunggu orang-orang sebelum kita untuk dilayani terlebih dahulu sebelum kita mendapat pelayanan dari toko tersebut. Lama waktu kita menunggu tersebut bisa sangat bervariasi tergantung pada panjangnya antrian sebelum kita dan juga barang-barang yang dibeli oleh orang lain yang sudah antri terlebih dahulu sebelum kita. Terkadang kita harus menunggu sangat lama agar kita bisa membayar barang yang akan kita beli.*

*Selain berdasarkan waktu tunggu yang bisa memakan waktu sangat lama, pada kasus lain seperti pada mengantri di bank yang sudah menggunakan sistem antrian muncul masalah lain lagi seperti nomor yang terlewati. Pada kasus ini terkadang sebagai pengguna jasa bank, kita datang ke bank dan mengambil nomor antrian terlebih dahulu. Namun ketika nomor yang kita ambil terlihat masih jauh terkadang kita akan melakukan aktivitas lain seperti keluar untuk mencari makan, ke toilet, dan sebagainya. Namun setelah kita kembali dari aktivitas lain tadi ternyata nomor kita sudah terlewati sehingga kita harus mengambil nomor antrian baru dan menunggu lagi.*

*Untuk mengantisipasi hal tersebut maka penelitian ini dilakukan dan diharapkan nanti kedepannya penelitian ini dapat dimanfaatkan dan diterapkan oleh berbagai pihak pemilik jasa agar pelanggannya tidak mengalami hal-hal yang telah dipaparkan sebelumnya. Didalam mengerjakan penelitian ini, peneliti akan menggunakan database MongoDB sebagai wadah untuk menyimpan nomor antrian dan data yang dibutuhkan, serta*

*menggunakan NodeJs dalam mengelola dan memproses data tersebut. Untuk memudahkan pengguna dalam pengambilan nomor antrian peneliti akan menyediakan 2 pilihan yaitu melalui aplikasi pada smartphone dan pengambilan nomor melalui halaman website.*

*Hasil dari penelitian ini adalah sebuah sistem antrian pintar dengan fitur multi tenant, multi layanan, multi loket dan pembaruan nomor antrian secara otomatis. Sehingga memungkinkan sistem ini dapat dipakai pada berbagai jenis industri.*

**Kata kunci : NodeJs, MongoDB, Aplikasi Antrian, Antrian Cerdas, Antrian Multi Tenant, Antrian Multi Layanan, Antrian Multi Loket, Pembaruan Nomor Antrian Otomatis**

**DESIGN AND DEVELOPMENT OF SMART QUEUE  
MANAGEMENT SYSTEM  
(STUDY CASE : SERVICE DESK DPTSI)**

**Student Name** : Arief Putranto  
**NRP** : 0521 13 40000 124  
**Departement** : Sistem Informasi  
**Supervisor.** : Dr. Eng. Febriliyan Samopa,  
S.Kom, M.Kom.

**ABSTRACT**

*Queue is the most common thing we see in almost anywhere, like when we go to shopping on the store or market or wherever we need to be patiently wait another people before us to be served before we are server from that market or store. The time we used to wait for us to be served is vary depend on how long the queue before us and how many product they bought. Sometimes a lot of time is used only for buy drink or food or thing we buy.*

*Besides of how much time we spend to queue, sometimes we got another issue like our number is missed when we queue on bank or hospital that already used queue management. On this case sometimes as customer we came to bank to get the queue number first. Then as we see that there will be a lot of number of queue before us, we decide to do another thing like buy some food, going to rest room, and do another thing we need to do. When we got back from that activity we missed our number, so we need to get new number and wait again like before.*

*In order to handle this kind of event, this research is being done and hopefully in the near future this research can be used and implemented at a lot of place and industry. While doing this research the researcher will use MongoDB as it's database, this was used as place or container to save queue number and whatever data it's needed. Not only MongoDB the researcher also using NodeJs for manage and processing the data from MongoDB. To improve user experience for the user, researcher will provide with 2 option at developing which is developing application on smartphone and website.*



*Result of this research is a smart queue management system with multi tenant / service provider, multi services, multi counter and automatic renewal of queue number to prevent the queue number being missed. Also with that feature, this system is possible to be used at any kind of industry.*

**Keyword : NodeJs, MongoDB, Queue Management, Smart Queue Management, Multi Service Provider, Multi Services, Multi Counter, Autorenewal Queue Number**

## SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertandatangan di bawah ini:

Nama : Arief Putranto  
NRP : 05211340000124  
Tempat/Tanggal lahir : Kediri / 9 Juli 1995  
Fakultas/Departemen : Fakultas Teknologi Elektro dan  
Informatika Cerdas / Departemen  
Sistem Informasi  
Nomor Telp/Hp/email : 085735865273

Dengan ini menyatakan dengan sesungguhnya bahwa penelitian/makalah/tugas akhir saya yang berjudul

**Rancang Bangun Aplikasi Antrian Pintar (Studi Kasus : Service Desk DPTSI)**

**Bebas Dari Plagiarisme Dan Bukan Hasil Karya Orang Lain.**

Apabila dikemudian hari ditemukan seluruh atau sebagian penelitian/makalah/tugas akhir tersebut terdapat indikasi plagiarisme, maka saya bersedia menerima sanksi sesuai peraturan dan ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat dengan sesungguhnya dan untuk dipergunakan sebagaimana mestinya.

Surabaya, 10 Agustus 2020



Arief Putranto  
NRP 05211340000124

## KATA PENGANTAR

*Alhamdulillahirabbil'alamin*, puji syukur atas rahmat Allah subhanahu wa ta'ala karena atas rahmat dan anugerah-Nya penulis dapat menyelesaikan tugas akhir dengan judul “RANCANG BANGUN APLIKASI ANTRIAN PINTAR (STUDI KASUS : SERVICE DESK DPTSI)” sebagai salah satu syarat kelulusan Program Sarjana Departemen Sistem Informasi, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember Surabaya. Tentunya dalam pengerjaan tugas akhir ini penulis telah mendapatkan dukungan dan bantuan dari banyak pihak. Oleh karena itu, penulis hendak menyampaikan ucapan terimakasih kepada :

1. Allah SWT yang telah memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir tepat waktu.
2. Ayah dan Ibu penulis, yang terus memotivasi dan mengingatkan untuk memprioritaskan penyelesaian tugas akhir di atas pekerjaan yang lainnya.
3. Bapak Febriliyan Samopa, selaku Dosen Pembimbing penulis, yang dengan sabar sudah membimbing dan memotivasi penulis untuk menyelesaikan tugas akhir ini.
4. Sahabat-sahabat seperantuan penulis : Adlie Yudha Tinumbang, Farin Reggie Hananta, Hendra Ody Permana, Egidia Rizka Marzam Zunaya yang selalu memberikan support kepada penulis.
5. Asisten laboratorium Infrastrukture dan Keamanan Sistem Informasi yang membantu penulis dalam hal administrasi penyelesaian tugas akhir.
6. Para Dosen Departemen Sistem Informasi.
7. Pihak-pihak lain yang telah mendukung dan membantu dalam kelancaran penyelesaian tugas akhir.

Penulis menyadari bahwa Tugas Akhir ini tidak akan pernah sempurna dan pasti memiliki kekurangan baik dalam penulisan maupun hasil akhir. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun sebagai bahan perbaikan untuk penelitian kedepannya. Terakhir, semoga Tugas Akhir ini dapat memberikan manfaat bagi para pembaca.

Surabaya, 10 Agustus 2020

Penulis

## DAFTAR ISI

<b>KATA PENGANTAR</b> .....	<b>vii</b>
<b>DAFTAR ISI</b> .....	<b>ix</b>
<b>DAFTAR GAMBAR</b> .....	<b>xi</b>
<b>DAFTAR TABEL</b> .....	<b>xiv</b>
<b>1. BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1 Latar belakang masalah .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah.....	3
1.4 Tujuan.....	3
1.5 Manfaat .....	3
1.6 Relevansi.....	4
<b>2. BAB II TINJAUAN PUSTAKA</b> .....	<b>5</b>
2.1 Penelitian sebelumnya .....	5
2.2 Landasan Teori .....	7
2.2.1. NodeJs.....	8
2.2.2. MongoDB.....	8
2.2.3. React Native .....	8
2.2.4. Websocket .....	9
<b>3. BAB III METODOLOGI</b> .....	<b>11</b>
3.1 Diagram Metodologi.....	11
3.1.1. Studi Literatur.....	12
3.1.2. Analisa dan Design .....	12
3.1.3. Perancangan Sistem .....	12
3.1.4. Validasi Sistem .....	12
3.1.5. Implementasi Sistem .....	13
3.1.6. Testing Sistem .....	13
3.1.7. Deploy.....	13
3.1.8. Pembuatan Dokumen .....	13
3.2 Jadwal Kegiatan .....	15
<b>4. BAB IV PERANCANGAN</b> .....	<b>17</b>
4.1. Tahapan Analisis .....	17
4.1.1. Menggali kebutuhan.....	17
4.1.2. Pembuatan daftar kebutuhan fungsional.....	19
4.1.3. Membuat daftar pengguna sistem.....	21

4.1.4.	Pengelompokan kebutuhan berdasarkan user ....	22
4.2.	Proses Bisnis Tiap User .....	23
4.2.1.	Guest Mobile .....	23
4.2.2.	User Mobile.....	25
4.2.3.	User Website .....	26
4.2.4.	Admin Website.....	29
4.2.5.	Administrator Website.....	42
4.3.	Use Case Diagram.....	51
4.3.1.	Design fitur mobile .....	52
4.3.2.	Design fitur web .....	53
4.4.	Tahap Desain .....	55
4.4.1.	Desain Sequence Diagram.....	56
4.4.2.	Desain Class Diagram .....	59
<b>5.</b>	<b>BAB V IMPLEMENTASI.....</b>	<b>61</b>
5.1.	Pembuatan desain antarmuka pengguna .....	61
5.2.	Pembuatan database .....	75
5.3.	Pembuatan Script atau Coding.....	75
<b>6.</b>	<b>BAB VI HASIL DAN PEMBAHASAN.....</b>	<b>81</b>
6.1.	Verifikasi sistem.....	81
6.2.	Pengujian pada hasil pembuatan sistem antrian pintar	81
<b>7.</b>	<b>BAB VII KESIMPULAN DAN SARAN .....</b>	<b>107</b>
7.1.	Kesimpulan .....	107
7.2.	Saran.....	107
<b>8.</b>	<b>Daftar Pustaka.....</b>	<b>108</b>
<b>9.</b>	<b>BIODATA PENULIS.....</b>	<b>109</b>
<b>LAMPIRAN A .....</b>		<b>110</b>
A.1.	Sequence Diagram Sistem Antrian Pintar.....	110

## DAFTAR GAMBAR

Gambar 3.1 Design Flowchart .....	11
Gambar 4.1 Flow guest mobile .....	24
Gambar 4.2 Flow user mobile.....	25
Gambar 4.3 Flow user website mendapatkan nomor pemanggilan selanjutnya.....	26
Gambar 4.4 Flow user website pemanggilan nomor antrian.....	27
Gambar 4.5 Flow user website lihat loket dan user yang dilayani	28
Gambar 4.6 Flow user website mendapatkan nomor antrian baru	29
Gambar 4.7 Flow admin website tambah user .....	30
Gambar 4.8 Flow admin website ubah user.....	31
Gambar 4.9 Flow admin website pencarian user.....	32
Gambar 4.10 Flow admin website hapus user.....	33
Gambar 4.11 Flow admin website penambahan loket .....	34
Gambar 4.12 Flow admin website ubah loket.....	35
Gambar 4.13 Flow admin website pencarian loket .....	36
Gambar 4.14 Flow admin website hapus loket .....	37
Gambar 4.15 Flow admin website assign user ke loket.....	38
Gambar 4.16 Flow admin website penambahan layanan.....	39
Gambar 4.17 Flow admin website ubah layanan.....	40
Gambar 4.18 Flow admin website pencarian layanan .....	41
Gambar 4.19 Flow admin website hapus layanan .....	42
Gambar 4.20 Flow administrator website tambah user.....	43
Gambar 4.21 Flow administrator website ubah user .....	44
Gambar 4.22 Flow administrator website pencarian user.....	45
Gambar 4.23 Flow administrator website hapus user.....	46
Gambar 4.24 Flow administrator website assign user ke penyedia layanan .....	47
Gambar 4.25 Flow administrator website tambah penyedia layanan .....	48
Gambar 4.26 Flow administrator website ubah penyedia layanan .....	49

Gambar 4.27 Flow administrator website pencarian penyedia layanan .....	50
Gambar 4.28 Flow administrator website hapus penyedia layanan .....	51
Gambar 4.29 Desain use case aktor guest mobile .....	52
Gambar 4.30 Desain use case aktor user mobile .....	53
Gambar 4.31 Design use case <i>user</i> website .....	54
Gambar 4.32 Design use case <i>admin</i> website .....	55
Gambar 4.33 Design use case administrator website .....	55
Gambar 4.34 Sequence diagram fitur admin kelola user .....	57
Gambar 4.35 Sequence diagram fitur admin kelola <i>services</i> .....	58
Gambar 4.36 Class Diagram .....	59
Gambar 5.1 Halaman pilih penyedia layanan mobile .....	62
Gambar 5.2 Halaman pilih layanan .....	63
Gambar 5.3 Halaman pilih metode pengambilan nomor .....	64
Gambar 5.4 Halaman pengambilan nomor berdasarkan waktu .....	65
Gambar 5.5 Halaman pengambilan nomor antrian .....	66
Gambar 5.6 Halaman login .....	67
Gambar 5.7 Halaman pilih loket .....	68
Gambar 5.8 Halaman pemanggilan dan melanjutkan nomor antrian .....	69
Gambar 5.9 Halaman daftar pemanggilan nomor antrian .....	70
Gambar 5.10 Halaman pemanggilan dan melanjutkan nomor antrian .....	71
Gambar 5.11 Halaman pengambilan nomor antrian .....	71
Gambar 5.12 Halaman kelola loket .....	72
Gambar 5.13 Halaman kelola layanan .....	73
Gambar 5.14 Halaman kelola user .....	73
Gambar 5.15 Halaman kelola penyedia layanan .....	74
Gambar 5.16 Halaman kelola user .....	75
Gambar 5.17 Desain database .....	77
Gambar 5.18 Contoh code controller .....	78
Gambar 5.19 Contoh code model .....	78
Gambar 5.20 Contoh code view .....	79



Gambar A.1 Sequence diagram <i>user mobile</i> .....	110
Gambar A.2 Sequence <i>guest mobile</i> .....	111
Gambar A.3 Sequence diagram <i>user</i> website pemanggilan nomor antrian.....	112
Gambar A.4 Sequence diagram user website melanjutkan nomor antrian.....	113
Gambar A.5 Sequence diagram user website ambil nomor antrian .....	114
Gambar A.6 Sequence diagram user website daftar pemanggilan nomor antrian.....	115
Gambar A.7 Sequence diagram <i>admin</i> website kelola <i>user</i> .....	116
Gambar A.8 Sequence diagram <i>admin</i> website kelola <i>services</i> .	117
Gambar A.9 Sequence diagram <i>admin</i> website kelola loket.....	118
Gambar A.10 Sequence diagram <i>administrator</i> website kelola <i>user</i> .....	119
Gambar A.11 Sequence diagram <i>administrator</i> website kelola <i>service provider</i> .....	120

## DAFTAR TABEL

Table 2.1 Penelitian Terkait-1 .....	5
Table 2.2 Penelitian Terkait-2 .....	6
Table 2.3 Penelitian Terkait-3 .....	6
Table 2.4 Penelitian Terkait-4 .....	6
Table 2.5 Penelitian Terkait-5 .....	7
Table 3.1 Jadwal Kegiatan.....	15
Table 4.1 Tabel perbandingan fitur aplikasi antrian.....	18
Table 4.2 Tabel kebutuhan fungsional .....	19
Table 4.3 Daftar nama tabel sistem antrian pintar.....	59
Table 6.1 Test case aktor guest mobile.....	81
Table 6.2 Test case aktor user mobile .....	84
Table 6.3 Test case aktor user website .....	85
Table 6.4 Test case aktor admin website fitur kelola loket.....	87
Table 6.5 Test case aktor admin website fitur kelola user .....	91
Table 6.6 Test case admin website fitur kelola layanan / <i>services</i> .....	94
Table 6.7 Test case aktor administrator website fitur user.....	97
Table 6.8 Test case aktor administrator website fitur penyedia layanan .....	101

# **BAB I**

## **PENDAHULUAN**

Pada bab I yaitu pendahuluan, penulis akan menjabarkan mengenai tugas akhir yang meliputi latar belakang masalah, perumusan masalah, batasan tugas akhir, tujuan tugas akhir dan relevansi atau manfaat tugas akhir. Dengan pendahuluan yang terdapat pada Bab I, diharapkan sebuah gambaran terhadap permasalahan yang akan diselesaikan pada tugas akhir ini dapat dipahami.

### **1.1 Latar belakang masalah**

Antrian merupakan hal yang sering kita jumpai di dalam keseharian kita, seperti ketika kita ingin membeli kebutuhan pokok seperti beras kita harus mengantri agar kita dilayani oleh penjual atau pemilik toko sebelum akhirnya kita dapat membawa pulang kebutuhan pokok tersebut.

Lama waktu yang dibutuhkan oleh seseorang untuk menunggu agar dia dapat dilayani oleh penjual jasa itu sangat bervariasi dan tergantung pada beberapa hal seperti banyaknya antrian yang ada sebelum kita serta banyaknya kebutuhan yang dibeli oleh orang yang antri sebelum kita. Terkadang kita tidak perlu menunggu lama agar kita dapat dilayani namun terkadang kita harus menunggu cukup lama agar kita dapat dilayani.

Smartphone atau dapat kita sebut sebuah telepon genggam yang banyak digunakan oleh masyarakat modern seperti sekarang ini. Hampir semua kalangan masyarakat memiliki telepon genggam seperti ini dikarenakan kemudahan dan fitur yang ditawarkan oleh smartphone ini sangatlah banyak, selain itu karena memiliki fitur yang sangat banyak, smartphone juga memiliki tingkat fleksibilitas dan portabilitas yang sangat baik sehingga dapat dengan mudah untuk dibawa kemanapun kita berada. Oleh karena itu penggunaan dari smartphone ini dapat dimanfaatkan salah satunya seperti aplikasi pengambilan nomor antrian secara online.

Salah satu contoh penggunaan smartphone untuk menangani antrian sekarang ini dapat kita temui di beberapa rumah sakit yang ada di Indonesia. Pada beberapa rumah sakit yang telah menggunakan aplikasi antrian online kita tidak harus datang ketempat tersebut untuk mengantri dan kita hanya perlu mendaftarkan diri kita melalui aplikasi smartphone atau web untuk mendapatkan nomor antrian. Manfaat yang didapatkan dari penggunaan aplikasi antrian ini yaitu kita bisa mengambil nomor antrian dimanapun kita berada namun kelemahannya ada kemungkinan nomor antrian kita dapat terlewati.

Tidak hanya pada rumah sakit, kita dapat melihat penggunaan aplikasi antrian online ini pada kantor-kantor bank. Berbeda dengan rumah sakit, aplikasi antrian yang ada di bank masih mengharuskan kita untuk datang ke bank tersebut untuk mengambil nomor antrian, namun setelah kita mendapatkan nomor antrian kita dapat duduk dan melihat nomor antrian terakhir pada sebuah layar yang ada. Selanjutnya kita hanya perlu menunggu giliran kita untuk dipanggil. Manfaat yang didapatkan dari penggunaan aplikasi antrian ini yaitu kemungkinan nomor antrian kita dapat terlewati sangatlah rendah namun kelemahannya bisa jadi kita harus menunggu dalam waktu yang cukup lama.

Penulis menggunakan Service Desk pada DPTSI ITS sebagai validasi dan implementasi aplikasi antrian dikarenakan pada Service Desk DPTSI ITS masih belum memiliki aplikasi antrian serta diharapkan kedepannya aplikasi antrian ini dapat digunakan pada tempat lain yang membutuhkan.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang telah dijelaskan diatas, maka rumusan masalah dari pembuatan tugas akhir ini adalah sebagai berikut:

1. Bagaimana pengguna bisa mengetahui kondisi antrian saat ini?

2. Bagaimana cara untuk mendapatkan nomor antrian pada waktu tertentu?
3. Bagaimana cara untuk mendapatkan nomor antrian yang sesuai dengan kedatangan kita?
4. Bagaimana cara agar nomor antrian yang kita miliki tidak terlewati?
5. Bagaimana agar aplikasi ini dapat digunakan diberbagai tempat?
6. Bagaimana agar aplikasi ini bisa memiliki banyak layanan?
7. Bagaimana agar aplikasi ini bisa memiliki banyak loket?

### **1.3 Batasan Masalah**

Mengacu kepada rumusan masalah diatas, batasan permasalahan dari tugas akhir ini antara lain :

1. Aplikasi hanya dikembangkan pada Android versi 6.0 (Marshmallow).
2. Aplikasi hanya dikembangkan dengan menggunakan bahasa pemrograman Node versi 14.4.0.
3. Menggunakan database MongoDB.

### **1.4 Tujuan**

Tujuan dari tugas akhir ini adalah untuk membuat sebuah aplikasi antrian yang mampu memberikan informasi terbaru mengenai antrian secara real time serta mencegah terjadinya terlewatnya nomor antrian ketika sedang dalam perjalanan menuju lokasi.

### **1.5 Manfaat**

Manfaat yang akan diperoleh dengan tugas akhir ini antara lain:

1. Pengguna dapat mengetahui informasi terbaru mengenai antrian yang ingin dituju.
2. Pengguna dapat melakukan aktivitas lain dari pada menunggu.

3. Pengguna tidak perlu tergesa-gesa karena takut nomor antriannya terlewati.
4. Pengguna tidak harus menunggu ditempat untuk mengetahui informasi tentang antriannya.
5. Pengguna jasa dapat memanfaatkan antrian ini tidak hanya pada satu tempat saja, namun dapat digunakan oleh cabang-cabang dari pengguna jasa tersebut.

## **1.6 Relevansi**

Relevansi tugas akhir ini terhadap laboratorium Infrastruktur dan Keamanan Teknologi Informasi adalah berkaitan dengan mata kuliah pilihan untuk laboratorium tersebut seperti Pemrograman Perangkat Bergerak serta mata kuliah yang diajarkan pada departemen Sistem Informasi pada Institut Teknologi Sepuluh Nopember (ITS) seperti Pemrograman Berbasis Web, Manajemen dan Administrasi Basis Data, dan Konstruksi dan Pengujian Perangkat Lunak.

## **BAB II**

### **TINJAUAN PUSTAKA**

Pada bab ini, berisikan tinjauan pustaka yang akan digunakan dalam penelitian tugas akhir ini, yang mencakup penelitian-penelitian sebelumnya, dasar teori dan metode yang digunakan selama pengerjaan

#### **2.1 Penelitian sebelumnya**

Penelitian sebelumnya merupakan referensi yang menjadi acuan dalam pengerjaan tugas akhir oleh penulis. Referensi-referensi yang digunakan sebagai acuan berisi teori maupun topik yang hampir serupa dengan penelitian ini, diantaranya :

Table 2.1 Penelitian Terkait-1

Judul	Communicating and displaying real-time data with websocket [1].
Nama, Tahun	V. Pimentel and B. G. Nickerson, 2012.
Gambaran Umum Penelitian	Pada penelitian ini, penulis memberikan gambaran mengenai berbagai jenis komunikasi antara client dan server, mulai dari menggunakan HTTP Polling, HTTP Long Polling, dan WebSocket [1].
Keterkaitan penelitian	Memberikan pengetahuan mengenai apa itu websocket dan apa kegunaan dari websocket.

Table 2.2 Penelitian Terkait-2

Judul	Web-based control application using WebSocket [2].
Nama, Tahun	Furukawa Y, 2011.
Gambaran umum penelitian	Pada penelitian ini, penulis memberikan gambaran mengenai definisi dari websocket dan bagaimana penggunaan websocket didalam suatu aplikasi [2]
Keterkaitan penelitian	Memberikan contoh mengenai penerapan websocket dengan menggunakan javascript

Table 2.3 Penelitian Terkait-3

Judul	Implementation of Location based Services in Android using GPS and Web Services [3].
Nama, Tahun	Singhal Manav, Shukla Anupam, 2012.
Gambaran umum penelitian	Pada penelitian ini, penulis memberikan penjelasan mengenai definisi dari Global Positioning System (GPS) dan proses pengambilan data lokasi tersebut pada android [3].
Keterkaitan penelitian	Memberikan pengetahuan mengenai bagaimana cara untuk mendapatkan posisi GPS pada aplikasi android.

Table 2.4 Penelitian Terkait-4

Judul	Implementation of haversine formula for counting event visitor in the radius based on Android application [4].
-------	--



Nama, Tahun	Cecep Nurul Anam, Manaf Khaerul, Atmadja Aldy Rialdy, Aurum Digital Khrisna, 2016.
Gambaran umum penelitian	Pada penelitian ini, penulis memberikan penjelasan perhitungan jarak pada dua titik berdasarkan latitude dan longitude dengan metode haversine [4].
Keterkaitan penelitian	Memberikan pengetahuan mengenai bagaimana penerapan haversine formula untuk menghitung jarak lokasi sekarang dengan tujuan.

Table 2.5 Penelitian Terkait-5

Judul	High-quality text-to-speech synthesis : an overview [5].
Nama, Tahun	Cecep Nurul Anam, Manaf Khaerul, Atmadja Aldy Rialdy, Aurum Digital Khrisna, 2016.
Gambaran umum penelitian	Pada penelitian ini, penulis memberikan penjelasan mengenai Text-To-Speech (TTS) synthesizer, dimana ini digunakan untuk memproses suatu kata atau kalimat agar menjadi suara [5].
Keterkaitan penelitian	Memberikan pengetahuan mengenai apa itu TTS kepada penulis.

## 2.2 Landasaan Teori

Dasar teori merupakan kumpulan teori yang digunakan dalam pengerjaan Tugas Akhir pada penelitian ini. Teori yang digunakan sebagai acuan dalam pengerjaan Tugas Akhir tidak berupa pendapat yang dikeluarkan secara pribadi tetapi harus mengacu kepada penelitian yang telah dilakukan sebelumnya maupun berdasarkan buku yang

telah diterbitkan. Dasar teori yang digunakan oleh penulis untuk penelitian ini adalah sebagai berikut :

### **2.2.1. NodeJs**

NodeJs merupakan salah satu framework dari Javascript yang cukup terkenal didalam dunia pengembangan suatu sistem. NodeJs sendiri merupakan framework yang memiliki performa yang sangat baik serta dapat menjalankan beberapa aplikasinya secara bersamaan (concurrent program) [6]. Perbedaan dengan aplikasi concurrent lainnya adalah NodeJs ini tidak bergantung pada pendekatan multi-threading pada umumnya, namun menggunakan pendekatan asynchronous input / output dengan berbasis pada even-driven programming model [6].

### **2.2.2. MongoDB**

MongoDb ini sendiri merupakan salah satu jenis database yang menggunakan No-SQL Database. Berbeda dengan SQL Database yang menggunakan table sebagai pengolahannya. MongoDb ini menggunakan konsep Key dan Value sehingga dalam segi performa bisa lebih cepat dibandingkan SQL Database.

### **2.2.3. React Native**

React Native merupakan salah satu jenis aplikasi yang digunakan untuk membuat aplikasi yang dapat digunakan oleh berbagai jenis sistem operasi dengan satu jenis / code dasar yang sama [7]. Itu artinya hanya ketika kita ingin membuat aplikasi berbasis android lalu kita ingin membuat

versi iosnya, kita tidak perlu menuliskan ulang kode atau menerjemahkan kode dari aplikasi yang sudah kita buat agar dapat berjalan pada ios.

#### **2.2.4. WebSocket**

Sebelum kita membahas websocket, sebelumnya mari kita bahas mengenai HTTP Polling terlebih dahulu. HTTP Polling merupakan suatu rangkaian dari request dan response message. Suatu client mengirimkan request kepada server dan ketika server menerima request, server merespon dengan mengirim pesan baru. jika ternyata terdapat pesan maka server akan mengirimkan pesan tersebut, namun jika tidak ada pesan baru kepada client maka server akan mengirimkan pesan kosong kepada client [1].

Dengan menggunakan HTTP Polling yang berkelanjutan, suatu aplikasi harus mengulangi HTTP headers dalam setiap request dari client dan setiap response dari server. tergantung pada aplikasinya, ini dapat menyebabkan komunikasi yang overhead / berlebihan. WebSocket protocol menyediakan Full-duplex, komunikasi 2 arah yang beroperasi pada satu socket yang berada di web dan dapat membantu membuat aplikasi web yang real time [1].

WebSocket protokol memiliki 2 bagian yaitu handshake dan data transfer. Handshake sendiri terdiri dari pesan yang dikirimkan oleh client dan response yang dikirimkan oleh server [1].

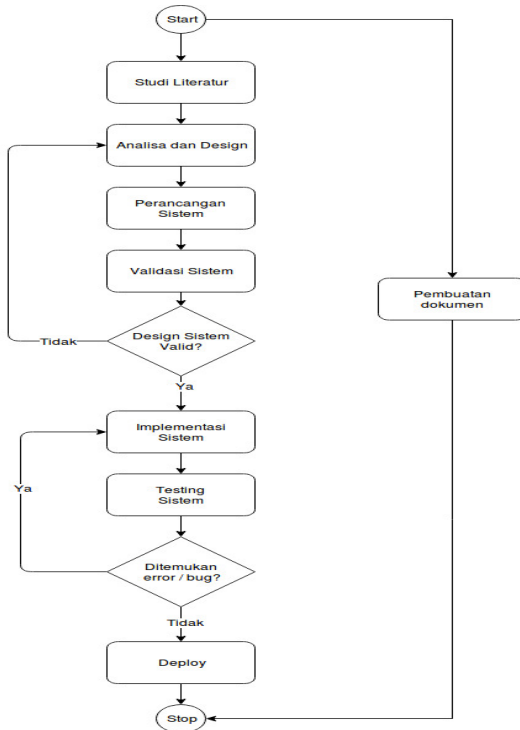
(halaman ini sengaja dikosongkan)

## BAB III METODOLOGI

Pada bab ketiga ini, akan dijabarkan mengenai langkah-langkah dalam pengerjaan Tugas Akhir.

### 3.1 Diagram Metodologi

Berikut ini adalah gambar diagram metodologi yang digunakan.



Gambar 3.1 Design Flowchart

### **3.1.1. Studi Literatur**

Pada tahap ini dilakukan pengumpulan literatur yang mendukung dalam menyelesaikan tugas akhir ini. Literatur yang tersaji merupakan penjelasan konsep – konsep atau penelitian sebelumnya yang pernah dilakukan dan didokumentasikan dalam buku, jurnal, maupun situs web. Keluaran proses ini adalah pemahaman mengenai konsep dan knowledge gap pada penelitian sebelumnya.

### **3.1.2. Analisa dan Design**

Pada tahapan ini dilakukan analisa kebutuhan apa saja yang diperlukan oleh aplikasi seperti fitur apa saja yang harus dimiliki oleh aplikasi dan informasi apa saja yang perlu ditampilkan oleh aplikasi serta bagaimana informasi tersebut diletakkan dan ditampilkan dalam bentuk situs web dan aplikasi android.

### **3.1.3. Perancangan Sistem**

Pada tahapan ini akan dilakukan perancangan sistem sehingga dapat diperoleh desain menyeluruh dari aplikasi yang akan dibuat secara rinci.

### **3.1.4. Validasi Sistem**

Pada tahapan ini akan dilakukan validasi dari perencanaan aplikasi yang telah dibuat sebelumnya dengan cara:

1. Memastikan apakah desain aplikasi sudah mencakup semua kebutuhan yang telah didapatkan dari tahapan analisa dan desain?

2. Memastikan apakah desain aplikasi benar-benar dapat diwujudkan dengan infrastruktur yang sekarang ini atau tidak?
3. Memastikan apakah semua fitur yang akan dibuat benar-benar dibutuhkan dalam aplikasi?

### **3.1.5. Implementasi Sistem**

Pada tahapan ini akan dilakukan proses pembuatan aplikasi sesuai dengan desain yang telah dibuat dari fase perencanaan. Keluaran dari tahap ini berupa aplikasi yang telah siap untuk dilakukan uji coba / testing.

### **3.1.6. Testing Sistem**

Pada tahapan ini akan dilakukan pengujian terhadap aplikasi yang telah dibuat, pengujian yang akan dilakukan dibagi menjadi 2 jenis yaitu pengujian secara fungsional atau dilihat dari fungsi tiap fitur apakah sudah berjalan sebagai mana mestinya, dan pengujian secara non fungsional seperti dari segi kecepatan aplikasi, performa aplikasi, dan aplikasi tersebut mampu diakses oleh berapa banyak pengguna.

### **3.1.7. Deploy**

Pada fase ini akan dilakukan proses yang mana apabila tidak ditemukan error/bug dari hasil fase sebelumnya (testing sistem) maka aplikasi yang dibuat sudah siap digunakan dan diterapkan pada bagian service desk di DPTSI ITS.

### **3.1.8. Pembuatan Dokumen**

Dokumen dibuat sesuai format yang disediakan dari Departemen Sistem Informasi serta

dilakukan dari tahap studi literatur hingga tahap implementasi sistem. Pada tahap awal, dokumen dibuat dengan mengatur format dan konten yang harus ada di dalam penyusunan dokumen tugas akhir. Di dalam laporan tersebut mencakup:

**A. Bab I Pendahuluan**

Dalam bab ini dijelaskan mengenai latar belakang, rumusan dan batasan masalah, tujuan dan manfaat pengerjaan tugas akhir ini.

**B. Bab II Tinjauan Pustaka dan Dasar Teori**

Dijelaskan mengenai penelitian-penelitian serupa yang telah dilakukan baik serta teori maupun praktik yang menunjang permasalahan yang dibahas pada tugas akhir ini.

**C. Bab III Metodologi**

Dalam bab ini dijelaskan mengenai tahapan – tahapan apa saja yang harus dilakukan dalam pengerjaan tugas akhir

**D. Bab IV Perancangan**

Pada bab ini akan dijelaskan bagaimana rancangan dari penelitian tugas akhir yang meliputi penggalian kebutuhan dan desain sistem.

**E. Bab V Implementasi**

Pada bagian ini menjelaskan tentang proses implementasi dan uji coba / testing yang akan dilakukan pada sistem agar didapatkan sistem yang dapat mencakup semua kebutuhan dalam tugas akhir ini.

**F. Bab VI Kesimpulan dan Saran**

Berisi tentang kesimpulan dan saran yang ditujukan untuk kelengkapan penyempurnaan tugas akhir ini.



### 3.2 Jadwal Kegiatan

Dalam proses pengerjaan Tugas Akhir ini terdapat jadwal pengerjaan. Berikut ini merupakan tabel rincinan jadwal kegiatan yang berlangsung selama melakukan penelitian.

Table 3.1 Jadwal Kegiatan

Kegiatan	Maret		April				Mei				Juni	
	3	4	1	2	3	4	1	2	3	4	1	2
3.1 Studi Literatur												
3.2 Analisa dan Desain												
3.3 Perancangan Sistem												
3.4 Implementasi Sistem												
3.5 Testing Sistem												
3.6 Deploy												
3.7 Pembuatan Laporan Tugas Akhir												

(halaman ini sengaja dikosongkan)

## **BAB IV**

### **PERANCANGAN**

Pada bab ini akan mengenai perancangan sistem sesuai dengan yang telah dijelaskan pada bab sebelumnya dengan urutan studi literatur terlebih dahulu, setelah studi literatur selesai dilakukan kemudian akan dilanjutkan dengan tahapan analisis dan design dimana pada tahapan ini akan ditemukan rincian kebutuhan yang harus dipenuhi dalam bentuk use case diagram, sequence diagram serta class diagram. Selanjutnya akan dilanjutkan pada tahapan perancangan sistem dimana pada tahapan ini akan dihasilkan gambaran-gambaran tampilan sistem serta fitur-fitur yang akan dimiliki oleh sistem. Selanjutnya akan masuk pada tahapan implementasi sistem dimana pada tahapan ini akan dilakukan proses coding dengan menggunakan NodeJs serta beracuan pada luaran dari tahap perancangan sistem. Selanjutnya akan diteruskan pada tahapan uji coba / testing pada sistem yang telah dibuat apakah telah sesuai dengan rancangan dan fitur dapat dijalankan sebagaimana mestinya.

Berdasarkan uraian tersebut maka pada bab perancangan ini hanya akan membahas bagian-bagian utama dari sistem dan untuk penjelasan yang lebih rinci dapat dilihat pada lampiran.

#### **4.1. Tahapan Analisis**

Setelah peneliti selesai mencari dan mempelajari literatur yang terkait dengan penelitian maka pada Tahapan Analisis ini peneliti akan menggali kebutuhan yang diperlukan untuk pembuatan sistem antrian pintar. Berikut ini tahapan yang dilakukan oleh penulis :

##### **4.1.1. Menggali kebutuhan**

Pada tahapan ini penulis melakukan penggalan lebih dalam mengenai kebutuhan apa yang

sebenarnya dibutuhkan dalam pembuatan sistem antrian pintar. Dalam pengerjaan tahapan ini penulis melakukan perbandingan terhadap sistem antrian yang sudah ada. Berikut ini adalah hasil yang didapatkan oleh penulis.

Table 4.1 Tabel perbandingan fitur aplikasi antrian

Fitur	Qiwii (qiwii.id)	Aplikasi Antrian Online (developeraplikasi)	Portal RSUA (Rumah Sakit Universitas Airlangga)	Ehealth (Surabaya.go.id)
Multi tenant	Ya	Tidak	Tidak	Ya
Multi layanan	Ya	Ya	Ya	Ya
Multi loket	Ya	Ya	Ya	Ya
Memilih waktu kedatangan	Ya	Tidak	Tidak	Ya
Melakukan pengambilan nomor antrian berdasarkan waktu yang dipilih	Ya	Tidak	Tidak	Ya
Pengambilan nomor antrian dimana saja	Ya	Ya	Ya	Ya
Pembatalan antrian	Ya	Ya	Tidak	Ya
Melanjutkan antrian yang sudah dibuat	Ya	Ya	Tidak	Ya
Menampilkan informasi	Ya	Ya	Tidak	Tidak

Fitur	Qiwii (qiwii.id)	Aplikasi Antrian Online (developeraplikasi)	Portal RSUA (Rumah Sakit Universitas Airlangga)	Ehealth (Surabaya.go.id)
kondisi antrian secara real time				
Otomatis memperbarui nomor antrian apabila sudah melewati	Tidak	Tidak	Tidak	Tidak
Support aplikasi mobile	Ya	Tidak	Ya	Ya
Pemanggilan nomor antrian	Ya	Ya	Ya	Ya

#### 4.1.2. Pembuatan daftar kebutuhan fungsional

Berdasarkan hasil perbandingan fitur-fitur yang ada maka dapat ditentukan daftar kebutuhan dari sistem yang ingin dibuat seperti pada Table 4.2. Untuk kelebihan dari aplikasi penelitian ini maka ada penambahan fitur seperti otomatis memperbarui nomor antrian ketika sudah melewati.

Table 4.2 Tabel kebutuhan fungsional

Kode	Kebutuhan	Deskripsi
KF-01	Multi tenant	Sistem dapat memiliki beberapa tenant / penyedia layanan seperti Bank BCA, Bank Danamon, dan Bank Mandiri
KF-02	Multi layanan	Sistem dapat memiliki lebih dari satu jenis layanan seperti Customer Service dan Teller

Kode	Kebutuhan	Deskripsi
KF-03	Multi loket	Sistem dapat memiliki beberapa loket.
KF-04	Memilih waktu kedatangan	Sistem menyediakan pilihan kepada user untuk menentukan waktu kedatangannya.
KF-05	Melakukan pengambilan nomor antrian berdasarkan waktu yang dipilih	Sistem dapat mengambil nomor antrian yang sesuai dengan waktu yang dipilih oleh pengguna.
KF-06	Pengambilan nomor antrian dimana saja	Sistem dapat mengambil nomor antrian dimanapun pengguna berada.
KF-07	Pembatalan antrian	Sistem dapat membatalkan nomor antrian
KF-08	Melanjutkan antrian yang sudah dibuat	Sistem dapat melanjutkan kembali nomor antrian yang sudah dibuat oleh pengguna
KF-09	Menampilkan informasi kondisi antrian secara real time	Sistem dapat memberikan informasi kondisi terkini dari antrian yang sedang berjalan secara <i>real time</i>
KF-10	Otomatis memperbarui nomor antrian apabila sudah terlewati	Sistem dapat mengambil nomor antrian baru secara otomatis ketika nomor tersebut sudah terlewati
KF-11	Support aplikasi mobile	Sistem tidak hanya memiliki aplikasi dalam bentuk website namun juga termasuk mobile / phone
KF-12	Pemanggilan nomor antrian	Sistem dapat melakukan pemanggilan nomor antrian
KF-13	Mengelola user yang ada	Sistem dapat mengelola user yang ada seperti halnya menambahkan user baru, mengubah user, serta menghapus user.

#### **4.1.3. Membuat daftar pengguna sistem**

Setelah melakukan analisa terhadap hasil perbandingan fitur dan telah didapatkan daftar kebutuhan fungsional yang harus dipenuhi oleh sistem. Tahapan selanjutnya yaitu mengetahui siapa saja pengguna dari sistem ini serta apa peranan dari masing-masing pengguna tersebut agar setiap pengguna mengetahui apa saja hal yang dapat mereka lakukan didalam sistem dan hal yang tidak dapat mereka lakukan didalam sistem. Pengguna dari sistem ini dikategorikan menjadi 5 seperti berikut ini :

##### **A. Guest Mobile**

Guest Mobile ini merupakan pengguna aplikasi yang mengakses sistem dari aplikasi smartphone yang akan dibuat. Fitur yang diberikan kepada pengguna ini yaitu pengambilan nomor antrian, melihat kondisi antrian saat ini, melanjutkan antrian yang sudah dibuat, dan membatalkan antrian.

##### **B. User Mobile**

User Mobile merupakan user yang telah ditunjuk oleh admin web untuk melakukan proses pemanggilan nomor antrian dan memanggil nomor antrian berikutnya.

##### **C. User Web**

User web merupakan user yang telah ditunjuk oleh admin web untuk melakukan proses pemanggilan nomor antrian, mengambil nomor antrian baru (untuk user yang datang tanpa menggunakan aplikasi smartphone), serta dapat menampilkan halaman antrian antrian yang sedang dipanggil.

D. Admin Web

Admin web merupakan user yang dapat mengelola sistem seperti membuat user web, menunjuk user web pada loket tertentu, mengelola loket, serta mengelola layanan apa saja yang dimiliki penyedia layanan tersebut.

E. Administrator / root

Administrator merupakan user yang dapat membuat admin web, mengelola penyedia layanan, serta dapat menunjuk admin web untuk penyedia layanan tertentu.

**4.1.4. Pengelompokan kebutuhan berdasarkan user**

Setelah kita mendapatkan daftar kebutuhan fungsional dan daftar jenis pengguna dari aplikasi ini. Selanjutnya akan kita kelompokkan kebutuhan fungsional berdasarkan user untuk mencari fitur apa yang kita butuhkan.

Users	Kebutuhan Fungsional
Guest Mobile	<ul style="list-style-type: none"><li>- KF-04 - Memilih waktu kedatangan</li><li>- KF-05 - Melakukan pengambilan nomor antrian berdasarkan waktu yang dipilih</li><li>- KF-06 - Pengambilan nomor antrian dimana saja</li><li>- KF-07 - Pembatalan antrian</li><li>- KF-08 - Melanjutkan antrian yang sudah dibuat</li><li>- KF-09 - Menampilkan informasi kondisi antrian secara real time</li><li>- KF-10 - Otomatis memperbarui nomor antrian apabila sudah terlewati</li><li>- KF-11 - Support aplikasi mobile</li></ul>
User Mobile	<ul style="list-style-type: none"><li>- KF-12 - Pemanggilan nomor antrian</li></ul>



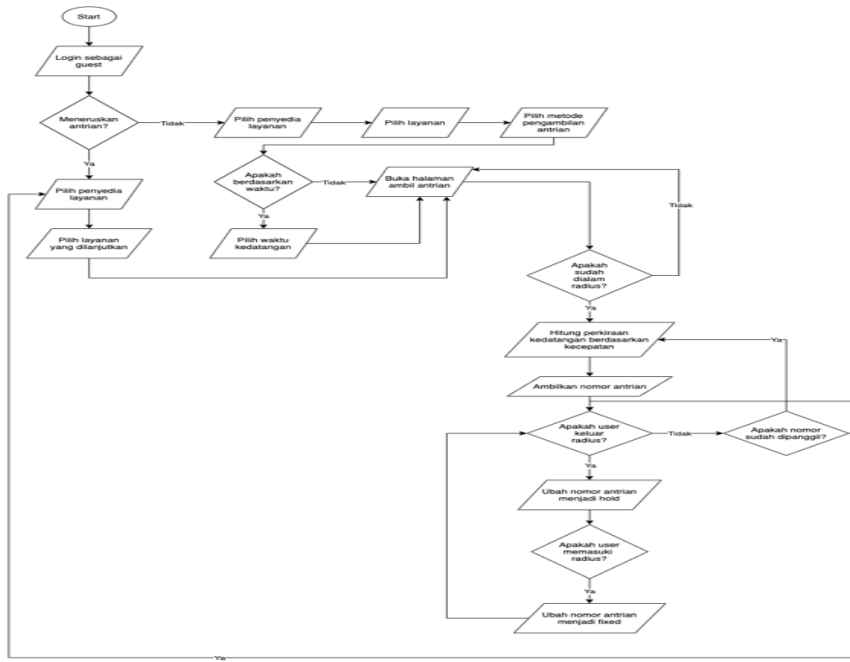
Users	Kebutuhan Fungsional
User Website	<ul style="list-style-type: none"> <li>- KF-09 - Menampilkan informasi kondisi antrian secara real time</li> <li>- KF-12 - Pemanggilan nomor antrian</li> </ul>
Admin Website	<ul style="list-style-type: none"> <li>- KF-02 - Multi layanan</li> <li>- KF-03 - Multi loket</li> <li>- KF-13 - Mengelola user yang ada</li> </ul>
Administrator Website	<ul style="list-style-type: none"> <li>- KF-01 - Multi tenant</li> <li>- KF-13 - Mengelola user yang ada</li> </ul>

## 4.2. Proses Bisnis Tiap User

Setelah tahap analisis selesai, kita sudah mendapatkan daftar kebutuhan fungsional dan daftar jenis pengguna dari sistem yang akan dibuat. Selanjutnya pada tahapan ini akan dijelaskan bagaimana proses bisnis dari setiap jenis pengguna dan cara mereka berinteraksi dengan sistem.

### 4.2.1. *Guest Mobile*

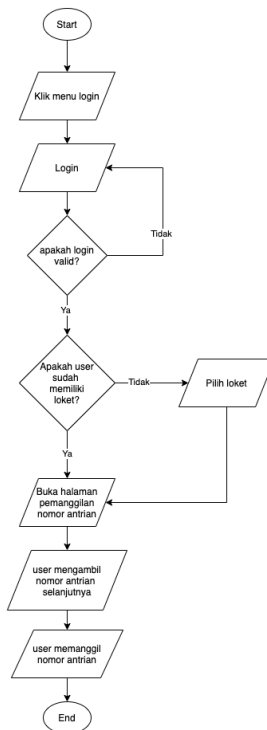
*Guest mobile* ini merupakan pengguna aplikasi mobile dimana untuk mengambil nomor antrian mereka dapat langsung menggunakan aplikasi tanpa harus mendaftar terlebih dahulu, selanjutnya mereka dapat memilih layanan yang akan dituju serta metode pengambilan nomor antrian (melalui waktu kedatangan atau posisi pengguna). Pada Gambar 4.1 dijelaskan mengenai flow dari *guest mobile* dalam berinteraksi dengan sistem.



Gambar 4.1 Flow guest mobile

#### 4.2.2. *User Mobile*

User mobile ini merupakan pengguna aplikasi mobile dimana untuk menggunakan fiturnya diharuskan untuk melakukan login terlebih dahulu. Pada Gambar 4.2 dijelaskan mengenai flow dari user manual dalam berinteraksi dengan sistem.



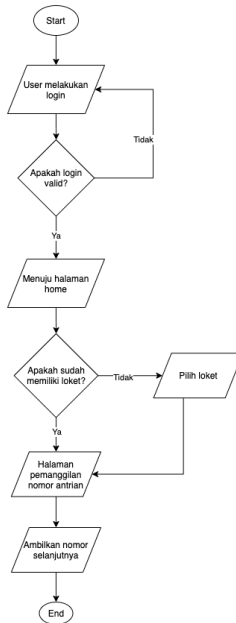
Gambar 4.2 Flow user mobile

### 4.2.3. User Website

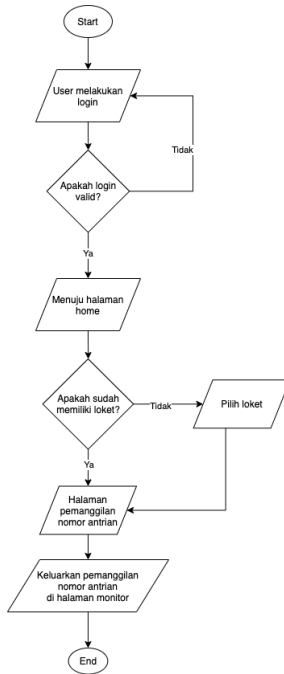
User website disini merupakan pengguna aplikasi website dimana untuk mengaksesnya pengguna harus login terlebih dahulu. Fitur yang dimiliki oleh user ini antara lain :

#### a. Mendapatkan nomor antrian selanjutnya / memanggil nomor antrian

Pada Gambar 4.3 dijelaskan bagaimana langkah untuk mendapatkan pemanggilan langkah selanjutnya. Sedangkan pada Gambar 4.4 dijelaskan bagaimana langkah untuk melakukan pemanggilan nomor antrian.



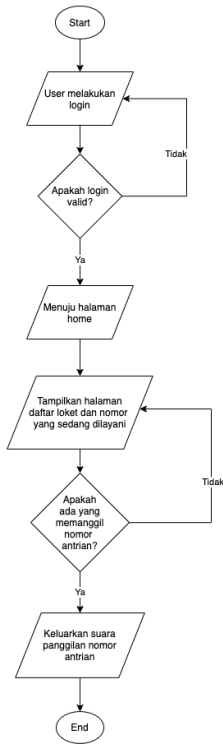
Gambar 4.3 Flow user website mendapatkan nomor pemanggilan selanjutnya



Gambar 4.4 Flow user website pemanggilan nomor antrian

**b. Melihat daftar loket dan user yang dilayani**

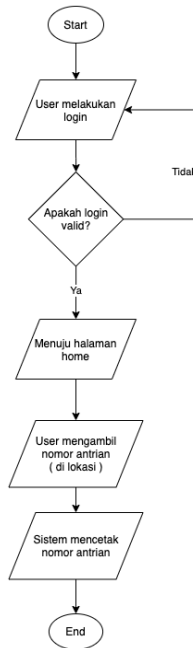
Pada Gambar 4.5 dijelaskan bagaimana proses untuk melihat daftar loket dan user yang sedang dilayani. Dan apabila terdapat pemanggilan nomor antrian maka pada halaman ini akan mengeluarkan suara.



Gambar 4.5 Flow user website lihat loket dan user yang dilayani

**c. Mengambil nomor antrian baru**

Pada Gambar 4.6 dijelaskan bagaimana flow dari user dalam mengambil nomor antrian baru.



Gambar 4.6 Flow user website mendapatkan nomor antrian baru

#### 4.2.4. Admin Website

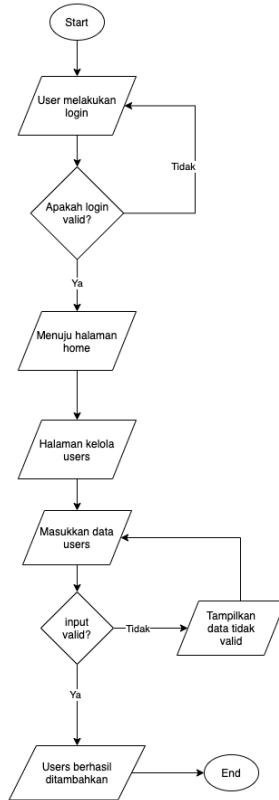
Admin website disini merupakan pihak yang mengelola antrian. Untuk mengaksesnya user harus melakukan login terlebih dahulu dan memiliki role admin. Fitur yang dimiliki antara lain sebagai berikut ini :

##### 1. Kelola User

Kelola user merupakan module yang digunakan untuk mengatur user. Berikut ini proses bisnis dari kelola user:

##### a. Tambah User

Proses penambahan user baru dapat dilihat pada Gambar 4.7.

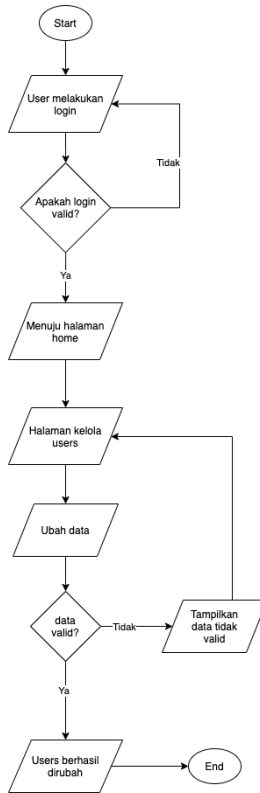


Gambar 4.7 Flow admin website tambah user

### b. Ubah User

Proses perubahan data user dapat dilihat pada Gambar 4.8.

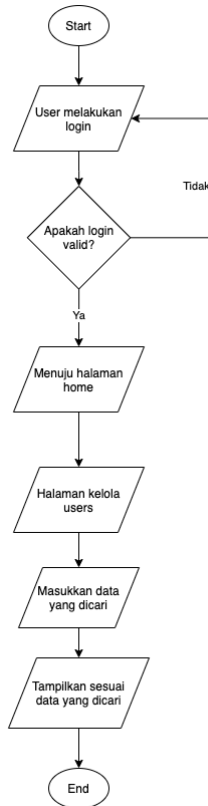




Gambar 4.8 Flow admin website ubah user

**c. Cari User**

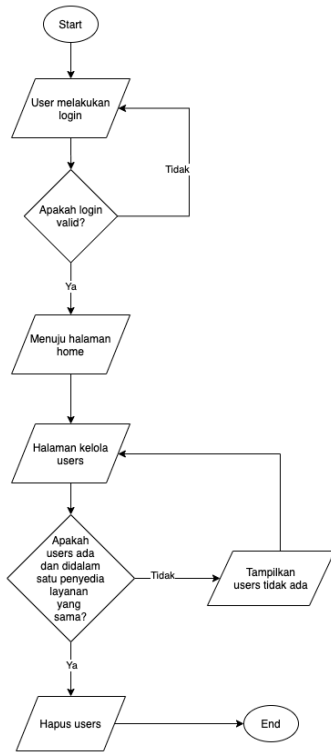
Proses pencarian user dapat dilihat pada Gambar 4.8.



Gambar 4.9 Flow admin website pencarian user

**d. Hapus User**

Proses hapus user dapat dilihat pada Gambar 4.10.



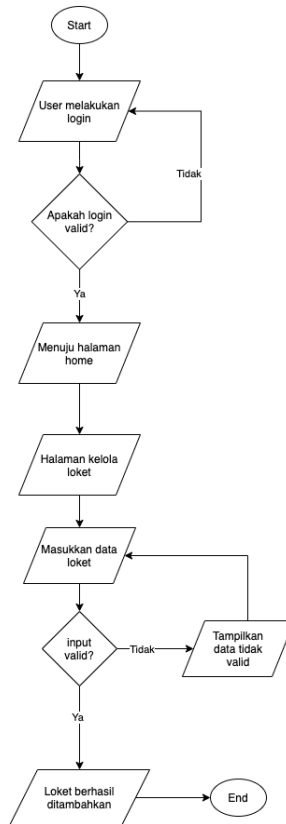
Gambar 4.10 Flow admin website hapus user

## 2. Kelola Loket

Kelola loket merupakan module yang digunakan untuk mengatur loket. Berikut ini proses bisnis dari kelola loket :

### a. Tambah Loket

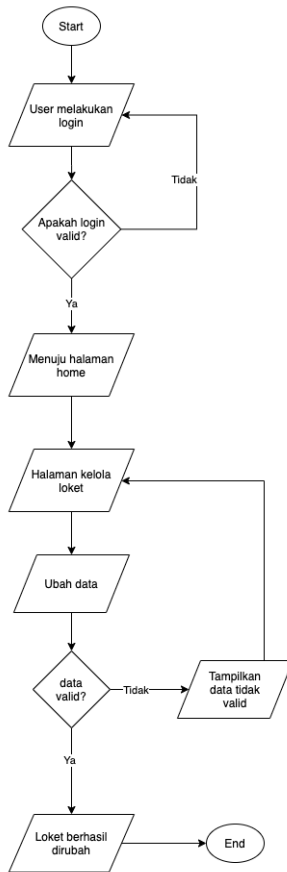
Untuk flow penambahan loket baru dapat dilihat pada Gambar 4.11.



Gambar 4.11 Flow admin website penambahan loket

**b. Ubah Loket**

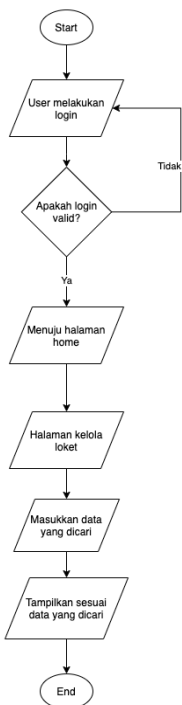
Untuk flow ubah loket dapat dilihat pada Gambar 4.12.



Gambar 4.12 Flow admin website ubah loket

**c. Cari Loket**

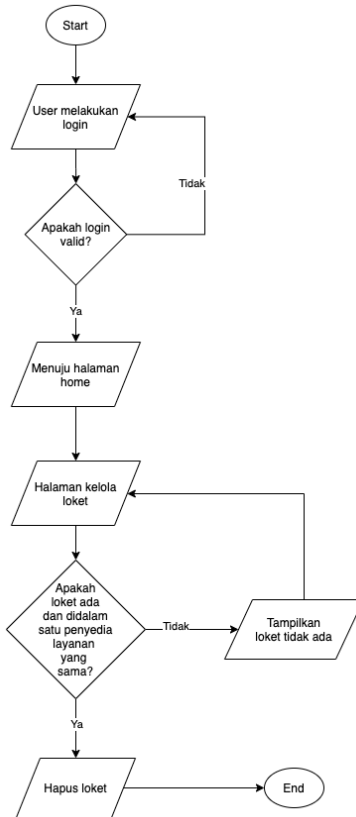
Untuk flow pencarian loket dapat dilihat pada Gambar 4.13.



Gambar 4.13 Flow admin website pencarian loket

**d. Hapus Loket**

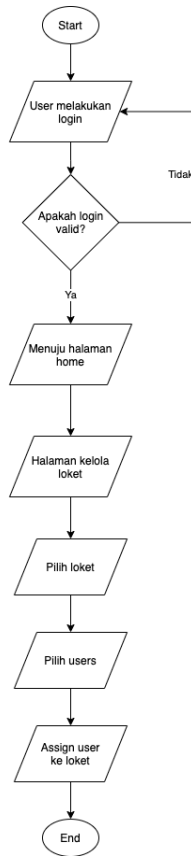
Untuk flow hapus loket dapat dilihat pada Gambar 4.14.



Gambar 4.14 Flow admin website hapus loket

**e. Assign User ke Locket**

Untuk melakukan *assign* user kedalam loket dapat dilihat pada Gambar 4.15.



Gambar 4.15 Flow admin website assign user ke loket

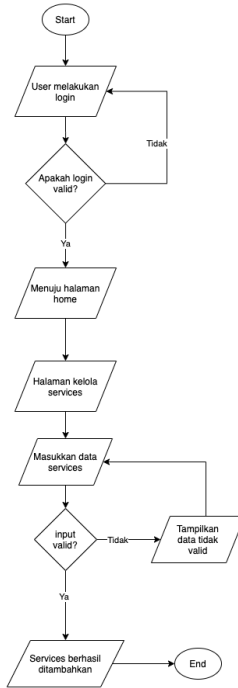
### 3. Kelola *Services* / Layanan

Kelola *services* atau layanan merupakan module yang digunakan untuk mengatur layanan. Berikut ini proses bisnis dari kelola layanan :

#### a. Tambah Layanan



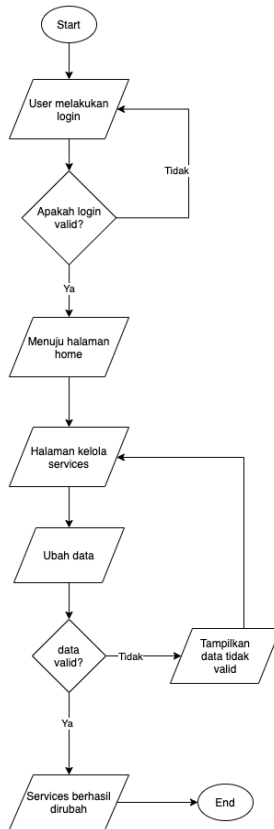
Untuk menambahkan layanan baru dapat dilihat pada Gambar 4.16.



Gambar 4.16 Flow admin website penambahan layanan

### b. Ubah Layanan

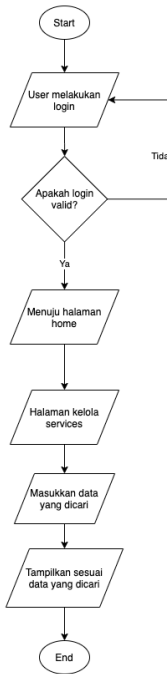
Untuk mengubah layanan dapat dilihat pada Gambar 4.17.



Gambar 4.17 Flow admin website ubah layanan

### c. Cari Layanan

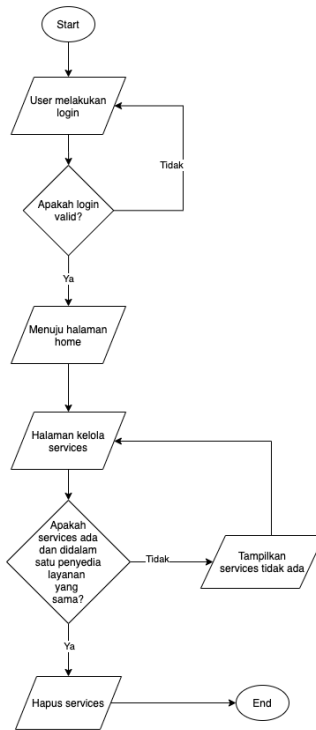
Untuk melakukan pencarian layanan dapat dilihat pada Gambar 4.18.



Gambar 4.18 Flow admin website pencarian layanan

#### d. Hapus Layanan

Untuk menghapus layanan yang sudah ada dapat dilihat pada Gambar 4.19.



Gambar 4.19 Flow admin website hapus layanan

#### 4.2.5. Administrator Website

Administrator website merupakan pihak yang mengelola user, mengatur akses user dan mengelola penyedia layanan / tenant. Berikut ini proses bisnis dari administrator website :

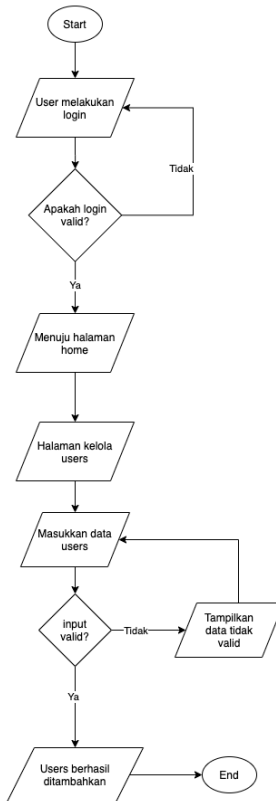
##### 1. Kelola User

Kelola user merupakan module yang digunakan untuk melakukan crud terhadap

user. Berikut ini proses bisnis dari kelola user :

**a. Tambah User**

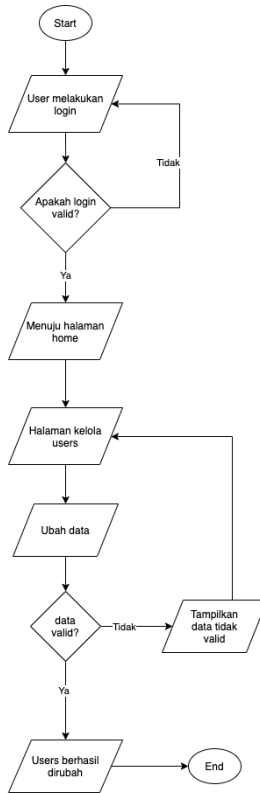
Proses penambahan user baru dapat dilihat pada Gambar 4.20.



Gambar 4.20 Flow administrator website tambah user

**b. Ubah User**

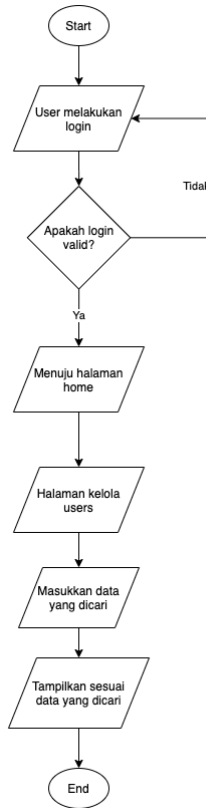
Proses perubahan data user dapat dilihat pada Gambar 4.21.



Gambar 4.21 Flow administrator website ubah user

**c. Cari User**

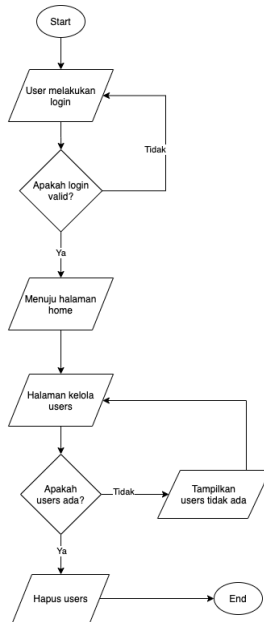
Proses pencarian user dapat dilihat pada Gambar 4.22.



Gambar 4.22 Flow administrator website pencarian user

**d. Hapus User**

Proses hapus user dapat dilihat pada Gambar 4.23.

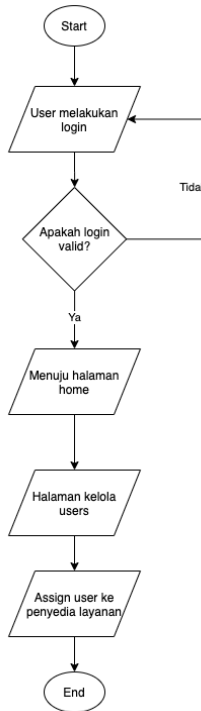


Gambar 4.23 Flow administrator website hapus user

**e. Assign User ke Penyedia Layanan**

Untuk melakukan assign user kepada penyedia layanan yang sudah ada dapat dilihat pada Gambar 4.24.





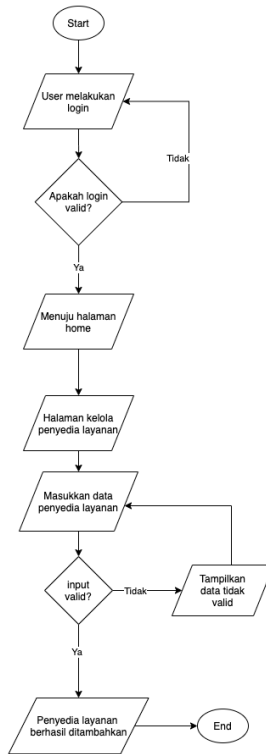
Gambar 4.24 Flow administrator website assign user ke penyedia layanan

## 2. Kelola Penyedia Layanan

Kelola penyedia layanan merupakan module yang digunakan untuk mengatur tenant atau penyedia layanan. Berikut ini merupakan proses bisnis dari kelola penyedia layanan:

### a. Tambah Penyedia Layanan

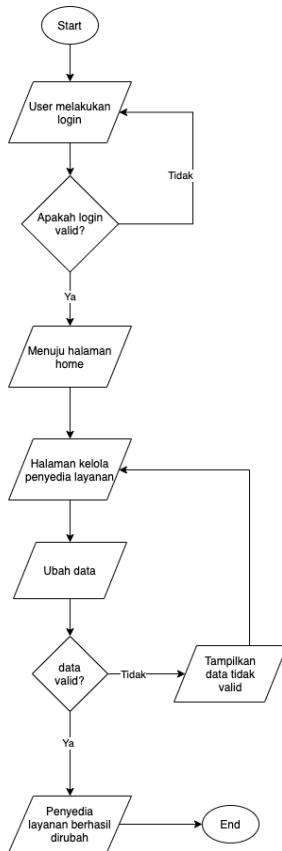
Untuk menambahkan penyedia layanan baru dapat dilihat pada Gambar 4.25.



Gambar 4.25 Flow administrator website tambah penyedia layanan

### b. Ubah Penyedia Layanan

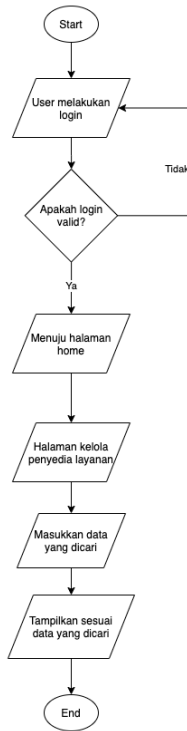
Untuk mengubah penyedia layanan yang sudah ada dapat dilihat pada Gambar 4.26.



Gambar 4.26 Flow administrator website ubah penyedia layanan

### c. Cari Penyedia Layanan

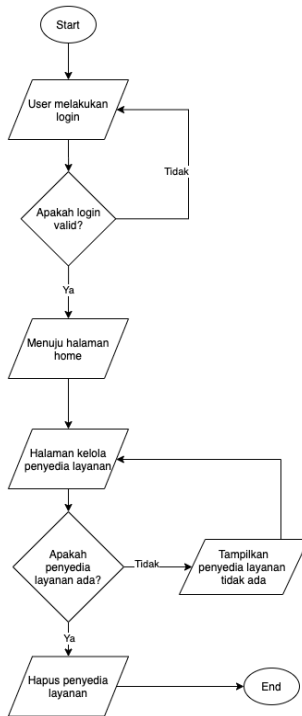
Untuk melakukan pencarian terhadap penyedia layanan yang sudah ada dapat dilihat pada Gambar 4.27.



Gambar 4.27 Flow administrator website pencarian penyedia layanan

#### d. Hapus Penyedia Layanan

Untuk menghapus penyedia layanan yang ada dapat dilihat pada Gambar 4.28.



Gambar 4.28 Flow administrator website hapus penyedia layanan

### 4.3. Use Case Diagram

Use case diagram atau diagram use case merupakan pemodelan untuk menggambarkan kelakuan (behavior) sistem yang akan dibuat. Dalam use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Dengan pengertian yang cepat, diagram use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan

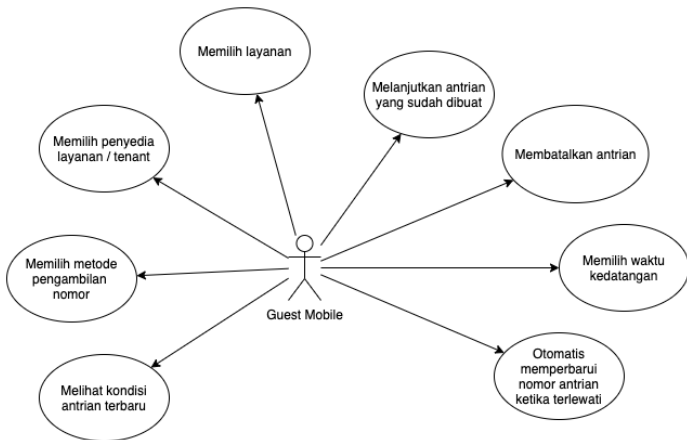
fungsi-fungsi tersebut. Berikut uraian singkat dan use case diagram dari fitur mobile dan website.

#### 4.3.1. Design fitur mobile

Fitur mobile merupakan aplikasi yang disediakan didalam bentuk aplikasi mobile atau aplikasi smartphone. Aplikasi ini memiliki 2 jenis pengguna yaitu *guest* dan *user*. Berikut ini use case diagram untuk kedua jenis pengguna tersebut :

##### a. *Guest*

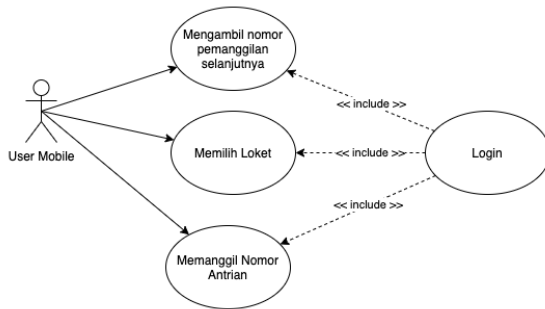
*Guest* merupakan jenis pengguna yang dapat melakukan aktivitas seperti pengambilan nomor antrian, pembatalan antrian dan melakukan pembaruan nomor secara otomatis jika terlewati seperti terlihat pada Gambar 4.29.



Gambar 4.29 Desain use case aktor guest mobile

**b. User**

User merupakan jenis pengguna yang berperan seperti loket sehingga user ini dapat melakukan aktivitas seperti pemanggilan nomor antrian dan memanggil nomor antrian selanjutnya seperti yang terlihat pada Gambar 4.30.



Gambar 4.30 Desain use case aktor user mobile

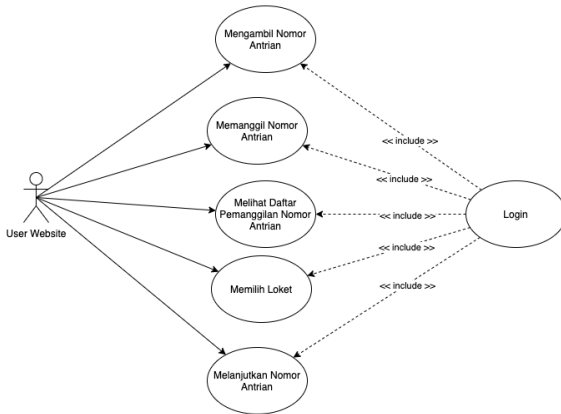
**4.3.2. Design fitur web**

Fitur web atau website merupakan aplikasi berbasis website sehingga dapat diakses melalui browser. Aplikasi ini memiliki 3 jenis pengguna yaitu *user*, *admin* dan *administrator*. Berikut ini use case diagram untuk ketiga jenis pengguna tersebut :

**a. User**

User merupakan jenis pengguna yang berperan seperti loket namun pada fitur website ini user juga dapat berperan dalam pengambilan nomor antrian serta bisa juga berfungsi sebagai monitor status loket serta nomor mana yang sedang dilayani oleh loket

tersebut seperti yang terlihat pada Gambar 4.31.

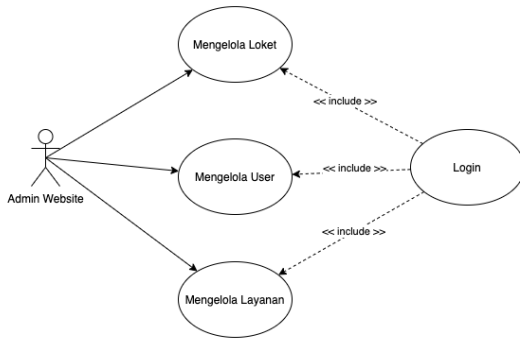


Gambar 4.31 Design use case *user website*

**b. Admin**

*Admin* merupakan jenis pengguna yang dapat melakukan aktivitas untuk mengelola user, layanan, dan loket seperti yang terlihat pada Gambar 4.32.

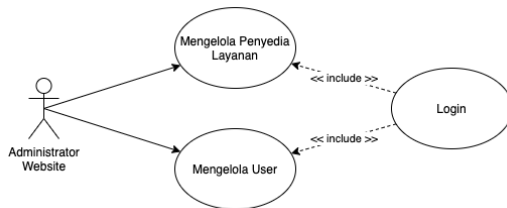




Gambar 4.32 Design use case *admin website*

**c. Administrator**

Administrator merupakan jenis pengguna yang dapat melakukan aktivitas untuk mengelola user termasuk menunjuk atau *assign* user ke suatu penyedia layanan atau *tenant* dan mengelola penyedia layanan layanan atau *tenant* seperti yang terlihat pada Gambar 4.33.



Gambar 4.33 Design use case administrator website

**4.4. Tahap Desain**

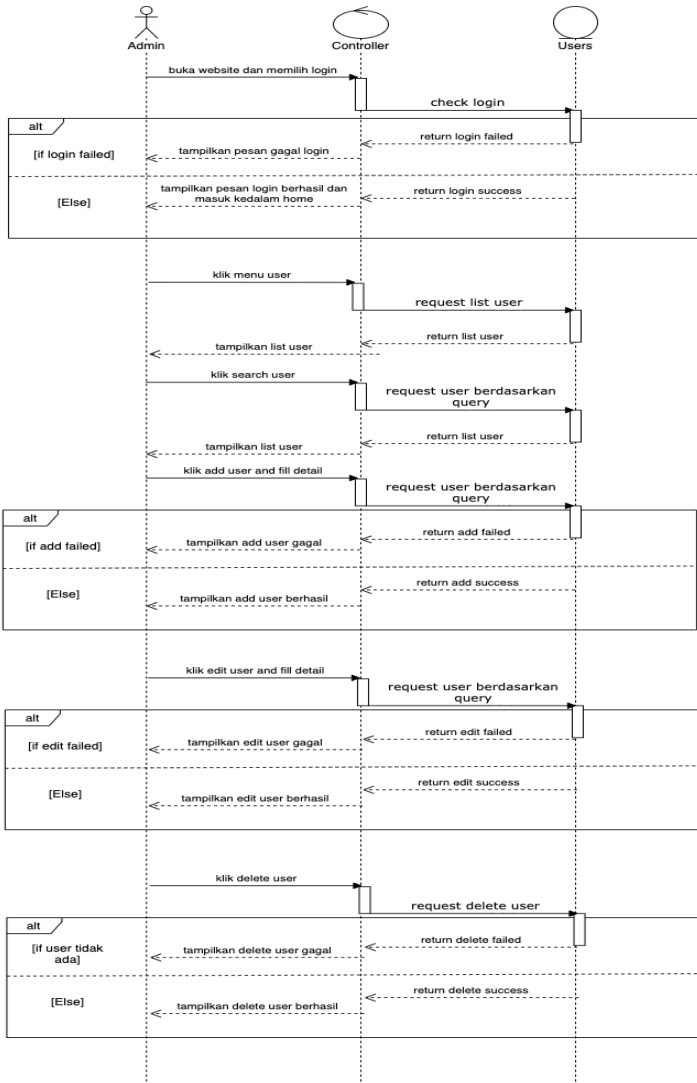
Tahap desain merupakan tahapan yang dilakukan setelah tahap analisis. Tahap desain digunakan untuk mengubah

kebutuhan-kebutuhan yang sudah dijabarkan pada tahap analisis menjadi representasi ke dalam bentuk “blueprint” sistem yakni sequence diagram, class diagram dan desain antarmuka sebelum tahap pembuatan kode atau script dimulai. Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya.

#### **4.4.1. Desain Sequence Diagram**

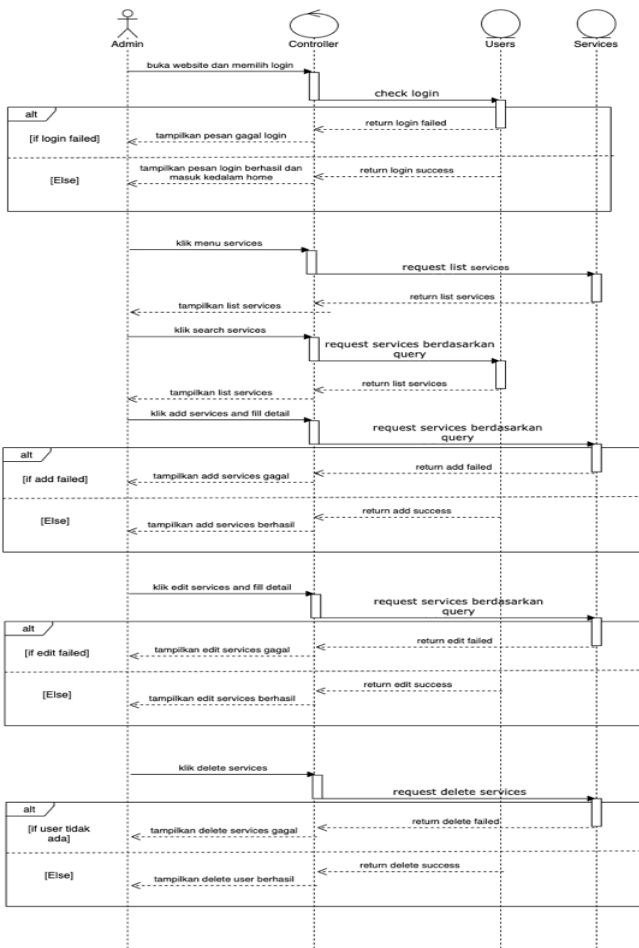
Desain sequence diagram dibuat berdasarkan dari desain use case dari masing-masing use case yang ada pada aplikasi. Berikut ini adalah beberapa contoh sequence diagram yang ada pada sistem antrian pintar untuk lebih lengkapnya ada pada Lampiran A.

- a. Sequence diagram admin kelola *user*



Gambar 4.34 Sequence diagram fitur admin kelola user

b. Sequence diagram admin kelola *service*



Gambar 4.35 Sequence diagram fitur admin kelola *services*

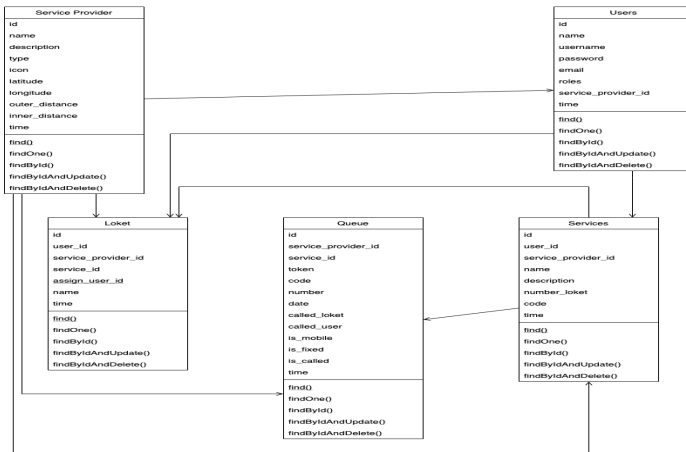
#### 4.4.2. Desain Class Diagram

Class diagram digunakan untuk menampilkan kelas - kelas di dalam sistem dan hubungan relasi yang memberikan gambaran sistem secara statis. Daftar tabel yang akan dibuat untuk class diagram dapat dilihat pada Tabel 4.4.

Table 4.3 Daftar nama tabel sistem antrian pintar

Nomor	Nama tabel
1	Service Provider
2	Users
3	Services
4	Loket
5	Queue

Sedangkan untuk bentuk relasi dari dari class diagram dapat dilihat pada Gambar 4.36.



Gambar 4.36 Class Diagram

(halaman ini sengaja dikosongkan)

## **BAB V**

### **IMPLEMENTASI**

Pada bab ini menjelaskan hasil dari proses penentuan studi kasus dan perancangan perangkat penggalian data yang didapatkan melalui wawancara, observasi, dan analisis dokumen.

Pada bab ini akan menjelaskan mengenai uraian proses coding serta proses pengujian beserta hasilnya.

#### **5.1. Pembuatan desain antarmuka pengguna**

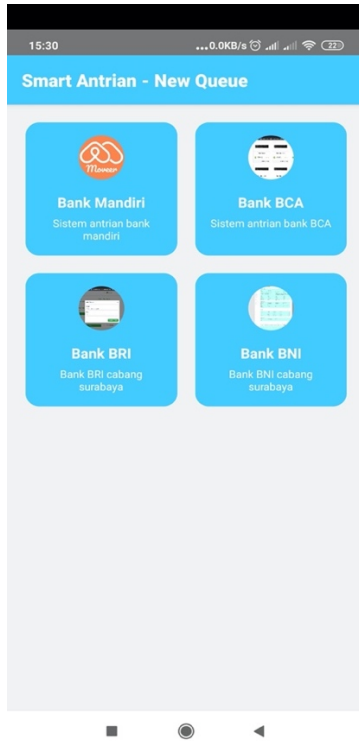
Desain antarmuka atau biasa kita sebut dengan desain GUI(Graphical User Interface). Tujuan dari adanya desain antarmuka ini adalah untuk membuat interaksi antara pengguna dengan sistem menjadi lebih mudah dipahami serta tidak membuat pengguna kebingungan ketika menggunakan sistem. Berikut ini desain antarmuka sistem yang dibangun:

##### **1. Fitur *Guest Mobile***

Fitur *guest mobile* merupakan fitur yang dapat diakses oleh setiap pengguna aplikasi melalui smartphone mereka tanpa harus melakukan login terlebih dahulu.

##### **a. Halaman Pilih Penyedia Layanan**

Halaman pilih penyedia layanan ini adalah halaman dimana user dapat memilih layanan apa yang akan dia gunakan lalu dengan menekan layanan tersebut maka user dapat lanjut ke halaman selanjutnya yaitu halaman pilih layanan. Untuk lebih jelasnya dapat dilihat pada Gambar 5.1, disana terdapat beberapa contoh penyedia layanan yang ada.

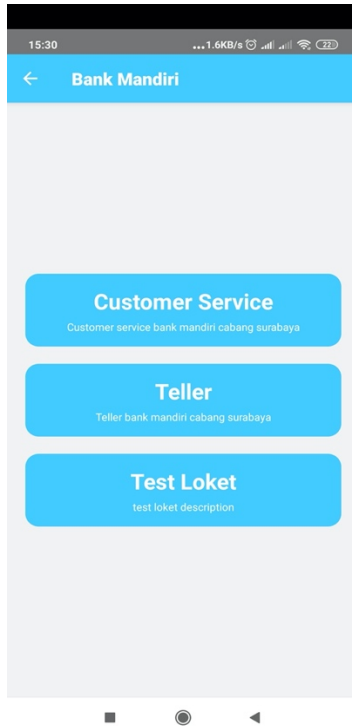


Gambar 5.1 Halaman pilih penyedia layanan mobile

**b. Halaman Pilih Layanan**

Halaman pilih layanan ini merupakan halaman yang digunakan oleh guest mobile untuk memilih layanan apa yang akan digunakan. Seperti pada Gambar 5.2 terlihat beberapa contoh layanan yang ada pada contoh bank mandiri.



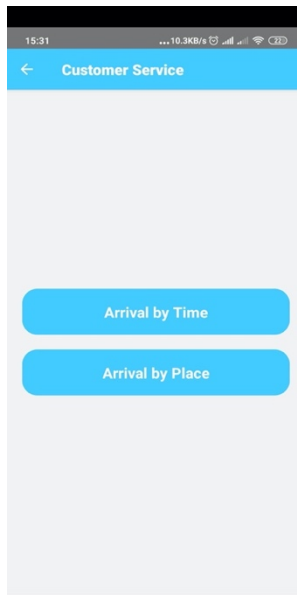


Gambar 5.2 Halaman pilih layanan

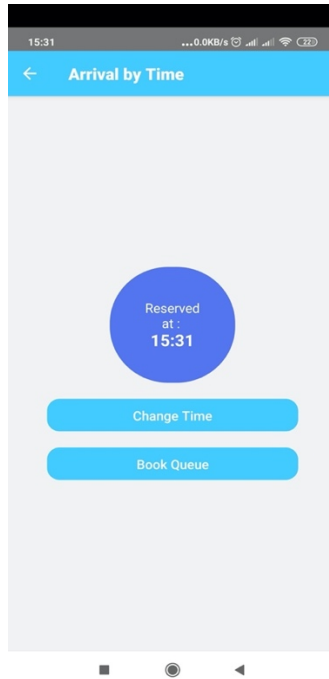
### c. **Halaman Pilih Metode Pengambilan Nomor**

Halaman pilih metode pengambilan nomor ini digunakan untuk memilih metode apa yang akan digunakan dalam pengambilan nomor, dalam aplikasi ini ada 2 metode yang dapat digunakan yaitu berdasarkan lokasi dan berdasarkan waktu. Untuk metode yang berdasarkan pada lokasi maka aplikasi akan melakukan pengecekan terhadap posisi user sekarang dalam pengambilan nomor antrian.

Untuk metode yang berdasarkan waktu maka user akan diambilkan nomor dengan syarat user sudah berada didekat lokasi dan sudah mendekati waktu yang ditentukan oleh user. Untuk lebih jelasnya dapat dilihat pada Gambar 5.3 untuk halaman pemilihan metode tersebut dan Gambar 5.4 untuk pengambilan nomor berdasarkan waktu.



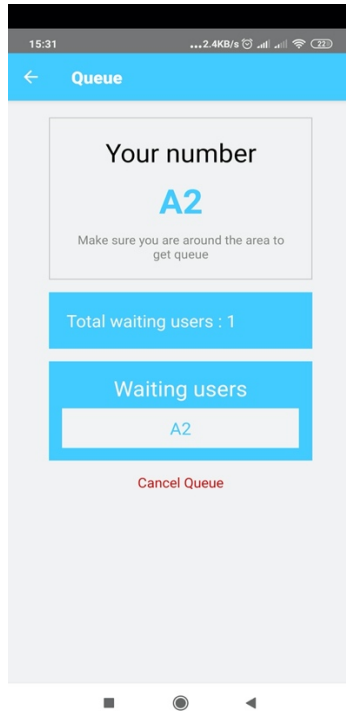
Gambar 5.3 Halaman pilih metode pengambilan nomor



Gambar 5.4 Halaman pengambilan nomor berdasarkan waktu

**d. Halaman Pengambilan Nomor Antrian**

Halaman pengambilan nomor antrian ini digunakan untuk mengambil nomor antrian berdasarkan metode pengambilan nomor antrian yang sudah dipilih sebelumnya. Untuk lebih jelasnya dapat dilihat pada Gambar 5.5 dimana pengguna dapat melihat nomor antriannya dan berapa banyak user yang sedang menunggu serta pada urutan berapa dia berada.



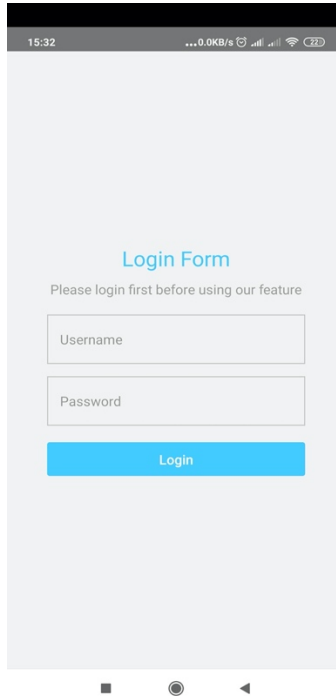
Gambar 5.5 Halaman pengambilan nomor antrian

## 2. **Fitur *User Mobile***

Fitur *user mobile* ini adalah fitur-fitur yang dapat digunakan oleh user yang telah ditentukan oleh admin untuk mengelola loket, seperti pemanggilan nomor antrian, melanjutkan nomor antrian ke nomor selanjutnya, mengambil nomor antrian, dan menampilkan daftar loket beserta nomor pengguna yang sedang dilayani. Fitur ini terbatas hanya pada fitur yang ada pada aplikasi mobile.

### a. **Halaman Login**

Halaman login ini digunakan agar tidak semua orang dapat mengakses fitur – fitur yang dimiliki oleh *user*. Untuk detail halaman login ini seperti apa dapat dilihat pada Gambar 5.6.

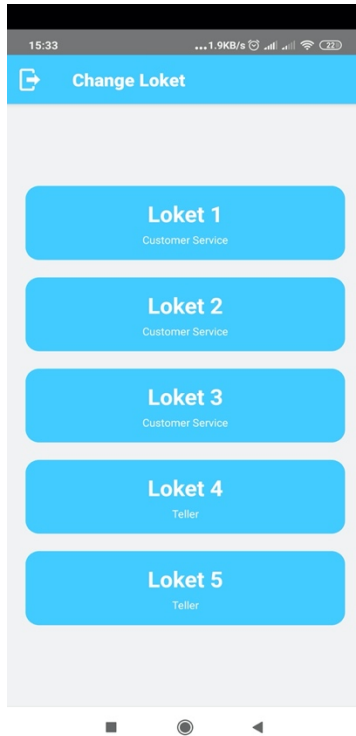


The image shows a mobile application login screen. At the top, there is a status bar with the time 15:32, network speed 0.0KB/s, and icons for cellular, Wi-Fi, and battery. The main content area has a light gray background. The title 'Login Form' is centered in blue. Below it, a message reads 'Please login first before using our feature'. There are two white input fields with gray borders: the first is labeled 'Username' and the second is labeled 'Password'. Below these fields is a prominent blue button with the text 'Login' in white. At the bottom of the screen, the standard Android navigation bar is visible with back, home, and recent apps icons.

Gambar 5.6 Halaman login

**b. Halaman Pilih Loker**

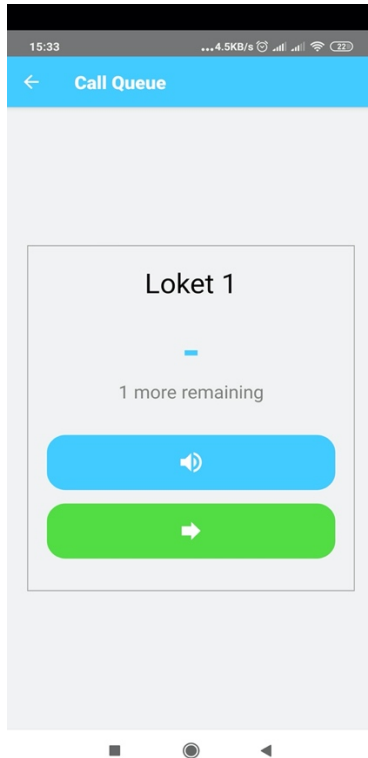
Halaman pilih loket ini digunakan ketika *user* ingin berpindah dari loket satu ke loket yang lain. Untuk lebih jelasnya halaman tersebut seperti apa, dapat dilihat pada Gambar 5.7.



Gambar 5.7 Halaman pilih loket

- c. **Halaman Pemanggilan Nomor Antrian**  
Halaman pemanggilan nomor antrian ini digunakan untuk memanggil nomor antrian sehingga pengguna tau bahwa sekarang adalah nomor antriannya. Bentuk halaman tersebut dapat dilihat pada Gambar 5.8.
- d. **Halaman Melanjutkan Antrian**  
Halaman melanjutkan antrian ini digunakan untuk mendapatkan nomor antrian setelah pengguna selesai dilayani. Tampilan yang digunakan pada halaman ini menjadi satu

dengan halaman pemanggilan nomor antrian yaitu pada Gambar 5.8.



Gambar 5.8 Halaman pemanggilan dan melanjutkan nomor antrian

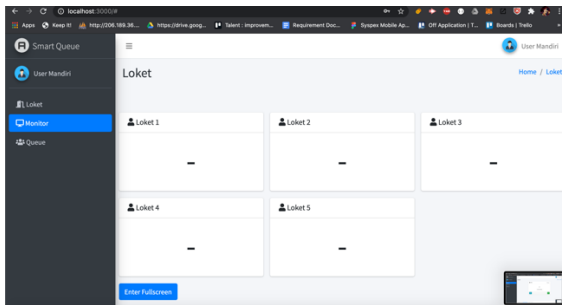
### 3. Fitur User Web

Fitur *user web* ini adalah fitur-fitur yang dapat digunakan oleh user yang telah ditentukan oleh admin untuk mengelola loket, seperti pemanggilan nomor antrian, melanjutkan nomor antrian ke nomor selanjutnya, mengambil nomor antrian, dan menampilkan daftar loket beserta nomor pengguna yang sedang dilayani. Fitur ini

terbatas hanya pada fitur yang ada pada aplikasi website.

**a. Halaman Daftar Pemanggilan Nomor Antrian**

Halaman daftar pemanggilan nomor antrian ini digunakan untuk memberitahukan kepada user yang berada pada sekitar lokasi bahwa nomor terakhir atau yang sedang dilayani saat ini adalah user dengan nomor antrian sekian. Untuk tampilan halaman ini dapat dilihat pada Gambar 5.9.



Gambar 5.9 Halaman daftar pemanggilan nomor antrian

**b. Halaman Pemanggilan Nomor Antrian**

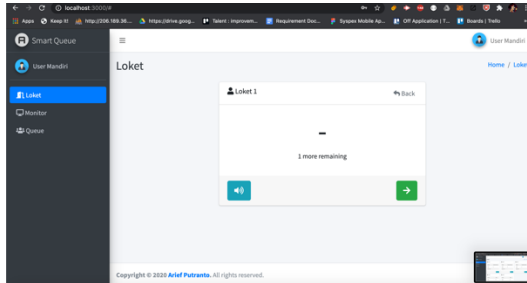
Halaman pemanggilan nomor antrian ini merupakan halaman yang digunakan *user* untuk memanggil nomor antrian. Untuk tampilan halaman ini dapat dilihat pada Gambar 5.10.

**c. Halaman Melanjutkan Antrian**

Halaman melanjutkan ini digunakan oleh *user* untuk mengambil nomor antrian selanjutnya. Untuk tampilan halaman ini



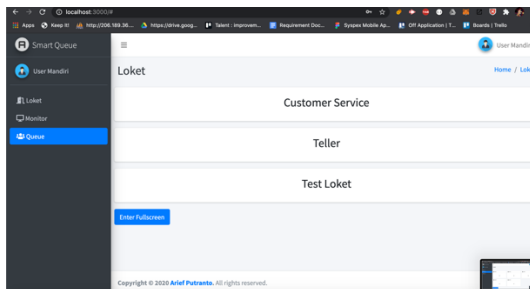
sama dengan halaman pemanggilan nomor antrian yaitu Gambar 5.10.



Gambar 5.10 Halaman pemanggilan dan melanjutkan nomor antrian

#### d. Halaman Pengambilan Nomor Antrian

Halaman pengambilan nomor antrian ini digunakan oleh pengguna aplikasi yang tidak menggunakan aplikasi mobile dan lebih memilih untuk datang dan antri di tempat. Tampilan halaman tersebut dapat dilihat pada Gambar 5.11.



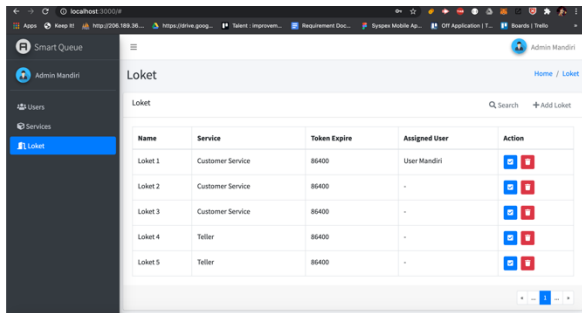
Gambar 5.11 Halaman pengambilan nomor antrian

### 4. Fitur Admin Web

Fitur admin web merupakan fitur – fitur yang hanya dapat digunakan oleh admin seperti mengelola loket, layanan dan user.

### a. Halaman Locket

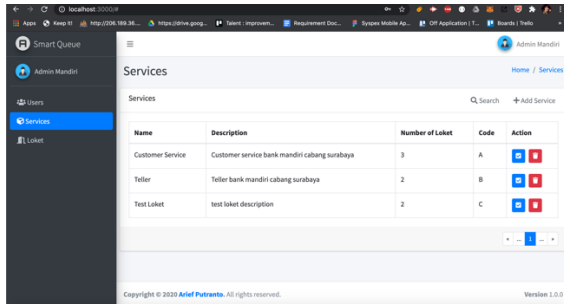
Halaman locket digunakan untuk mengelola locket yang dimiliki oleh penyedia layanan tersebut. Untuk lebih jelasnya dapat melihat tersebut. Untuk lebih jelasnya dapat melihat Gambar 5.12. Didalam halaman tersebut terdapat beberapa fitur lain seperti *add* locket untuk menambah locket, *search* untuk mencari locket, tombol *edit* untuk merubah locket serta tombol delete untuk menghapus locket.



Gambar 5.12 Halaman kelola locket

### b. Halaman Layanan

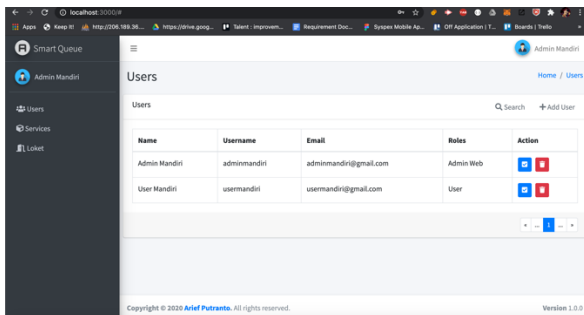
Halaman layanan digunakan untuk mengelola layanan yang dimiliki oleh penyedia layanan. Untuk lebih jelasnya dapat dilihat pada Gambar 5.13. Didalam halaman tersebut terdapat beberapa fitur tambahan seperti *add service* untuk menambah layanan baru, fitur *search* untuk mencari layanan, fitur ubah untuk mengubah layanan dan fitur hapus untuk menghapus layanan.



Gambar 5.13 Halaman kelola layanan

### c. Halaman User

Halaman *user* digunakan untuk mengelola user yang ada pada penyedia layanan tersebut. Untuk lebih jelasnya dapat melihat Gambar 5.14. Didalam halaman tersebut terdapat beberapa fitur tambahan seperti *search* untuk mencari user, *add user* untuk menambah user, *ubah user* untuk mengubah *user* dan *hapus user* untuk menghapus *user*.



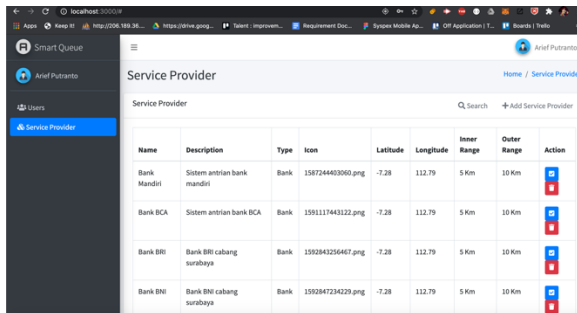
Gambar 5.14 Halaman kelola user









## 5. Fitur Administrator Web

Fitur administrator ini adalah fitur - fitur yang hanya dapat digunakan oleh administrator sebagai pengelola keseluruhan sistem.

#### a. Halaman Penyedia Layanan

Halaman penyedia layanan ini digunakan untuk mengelola daftar penyedia layanan. Untuk tampilan dari halaman ini dapat dilihat pada Gambar 5.15. Didalam halaman ini terdapat beberapa fitur tambahan seperti *search* untuk mencari penyedia layanan, *add service provider* untuk menambahkan penyedia layanan yang baru, *edit service provider* untuk mengubah data penyedia layanan, dan *hapus service provider* untuk menghapus penyedia layanan.



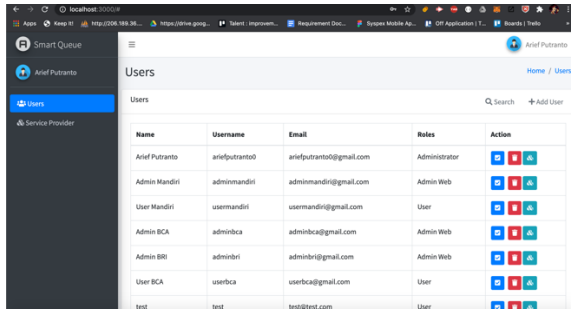
Name	Description	Type	Icon	Latitude	Longitude	Inner Range	Outer Range	Action
Bank Mandiri	Sistem antrian bank mandiri	Bank	1587244403060.png	-7.28	112.79	5 Km	10 Km	 
Bank BCA	Sistem antrian bank BCA	Bank	1591117443122.png	-7.28	112.79	5 Km	10 Km	 
Bank BRI	Bank BRI cabang surabaya	Bank	1592843256467.png	-7.28	112.79	5 Km	10 Km	 
Bank BNI	Bank BNI cabang surabaya	Bank	1592847234229.png	-7.28	112.79	5 Km	10 Km	 

Gambar 5.15 Halaman kelola penyedia layanan

#### b. Halaman User

Halaman *user* digunakan untuk mengelola keseluruhan *user* yang ada pada sistem. Untuk lebih jelasnya dapat dilihat pada Gambar 5.16. Didalam halaman tersebut terdapat beberapa fitur tambahan seperti *add user* untuk menambahkan *user*, *search* untuk mencari *user*, *edit user* untuk mengubah

*user*, hapus *user* untuk menghapus *user*, dan *assign user* untuk memasukkan *user* kedalam penyedia layanan.



Gambar 5.16 Halaman kelola user

## 5.2. Pembuatan database

Database dibuat berdasarkan desain model data fisik. Desain data model fisik yang sudah dibuat akan langsung degenerate kedalam bentuk database spesifik. Pada pembuatan sistem antrian pintar ini, peneliti menggunakan database yang dibuat dalam bentuk database MongoDB. Bentuk schema dari database yang digunakan pada penelitian ini dapat dilihat pada Gambar 5.17.

## 5.3. Pembuatan Script atau Coding

Proses coding dilakukan setelah database dan user interface selesai dibuat. Bahasa pemrograman yang digunakan adalah Javascript dengan menggunakan framework NodeJs. Pembuatan source code atau script ini dilakukan dengan konsep Model, View, dan Controller (MVC). Berikut ini merupakan contoh script dari model, view dan controller pada halaman admin fitur kelola penyedia layanan.

**a. Source code controller**

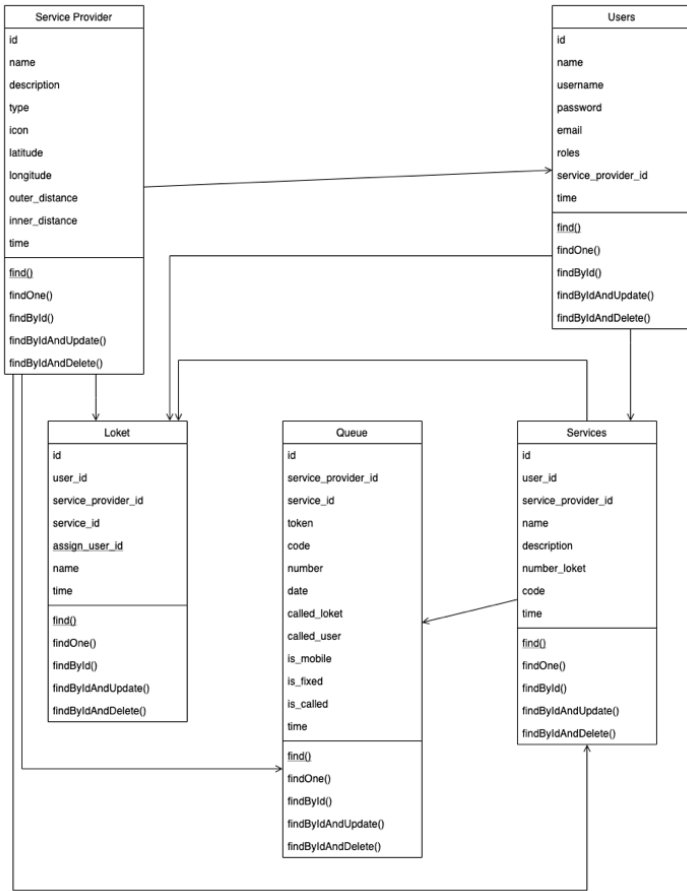
Source code controller function `listServiceProvider` pada Gambar 5.18 merupakan penghubung antara model dengan view dimana pada fungsi ini controller mengambil data dari model `ServiceProvider` lalu menampilkan data tersebut pada view `ServiceProvider.vue`.

**b. Source code model**

Source code model seperti pada Gambar 5.19 digunakan untuk membuat schema pada `mongodb`, sehingga memungkinkan kita untuk mengambil data penyedia layanan dari `mongodb` dengan schema tersebut.

**c. Source code view**

Source code view seperti pada Gambar 5.20 digunakan untuk menampilkan daftar penyedia layanan yang ada. View juga berfungsi sebagai tempat dimana data tersebut dapat ditampilkan dalam bentuk yang menarik seperti `map` dan `chart`.



Gambar 5.17 Desain database

```

class ServiceProviderController {
  async listServiceProvider (req, reply) {
    var page = req.query && req.query.page ? req.query.page : 1
    var perPage = req.query && req.query.perPage ? req.query.perPage : 10
    var name = req.query && req.query.name ? req.query.name : null
    var description = req.query && req.query.description ? req.query.description : null
    var type = req.query && req.query.type ? req.query.type : null

    const options = {
      page: page,
      limit: perPage
    }

    var condition = {}
    if (name !== null) {
      condition.name = { $regex: name, $options: "i" }
    }

    if (description !== null) {
      condition.description = { $regex: '.*' + description + '.*' }
    }

    if (type !== null) {
      condition.type = { $regex: type, $options: "i" }
    }

    try {
      const serviceProvider = await ServiceProvider.paginate(condition, options)
      reply.send({'statusCode': 200, 'message': '', 'data': serviceProvider})
    } catch (e) {
      reply.send({'statusCode': 500, 'message': e.message, 'data': {}})
    }
  }
}

```

Gambar 5.18 Contoh code controller

```

const mongoose = require('mongoose')
const mongoosePaginate = require('mongoose-paginate-v2')

const schema = new mongoose.Schema({
  name: String,
  description: String,
  type: String,
  icon: String,
  latitude: Number,
  longitude: Number,
  outer_distance: Number,
  inner_distance: Number,
  time: {type: Date, default: Date.now}
})

schema.plugin(mongoosePaginate)

module.exports = mongoose.model('ServiceProvider', schema)

```

Gambar 5.19 Contoh code model



```

QueueC Find Results x ServiceProviderController.js x ServiceProvider.vue x ServiceProvider.js x user.js x Qu
<template>
  <!-- Content Header (Page header) -->
  <router-link class="text-decoration: none" :to="{ url: '/', name: 'Home' }, { url: '/service-provider', name: 'Service Provider' }"/>
  </router-link>
  </content-header>
  <!-- Main content -->
  <section class="content">
    <div class="container">
      <div class="row">
        <div class="col-lg-12">
          <div class="card">
            <div class="card-header">
              <div class="card-title">Service Provider</div>
              <div class="card-tools">
                <div class="nav nav-pills ml-auto">
                  <div class="nav-item">
                    <div class="nav-link" href="#" data-toggle="modal" data-target="#search-modal"><i class="fa fa-search"></i> Search</div>
                  </div>
                  <div class="nav-item">
                    <div class="nav-link" href="#" data-toggle="modal" data-target="#add-modal" @click="setIsCreate(true)"><i class="fa fa-plus"></i> Add Service Provider</div>
                  </div>
                </div>
              </div>
            </div>
            <div class="card-body">
              <table class="table table-striped table-bordered">
                <thead>
                  <tr>
                    <th>Name</th>
                    <th>Description</th>
                    <th>Type</th>
                    <th>Icon</th>
                    <th>Latitude</th>
                    <th>Longitude</th>
                    <th>Inner Range</th>
                    <th>Outer Range</th>
                    <th>Action</th>
                  </tr>
                </thead>
                <tbody>
                  <tr>
                    <td>{{ service.name }}</td>
                    <td>{{ service.description }}</td>
                    <td>{{ service.type }}</td>
                    <td>{{ service.icon }}</td>
                    <td>{{ Math.round(service.latitude * 100) / 100 }}</td>
                    <td>{{ Math.round(service.longitude * 100) / 100 }}</td>
                    <td>{{ service.inner_distance }} Km</td>
                    <td>{{ service.outer_distance }} Km</td>
                    <td>
                      <button class="btn btn-sm btn-primary" title="Update Service Provider" data-toggle="modal" data-target="#add-modal" @click="modalUpdateServiceProvider(service)" style="margin-right: 5px;"><i class="fa fa-check-square"></i> Update</td>
                    </td>
                  </tr>
                </tbody>
              </table>
            </div>
          </div>
        </div>
      </div>
    </div>
  </section>

```

Gambar 5.20 Contoh code view

(halaman ini sengaja dikosongkan)

## **BAB VI**

### **HASIL DAN PEMBAHASAN**

Pada bab akan dijelaskan hasil dan pembahasan dari pembuatan sistem antrian pintar.

#### **6.1. Verifikasi sistem**

Verifikasi ini tidak dapat dilakukan karena aplikasi ini dibuat dengan tujuan untuk umum dan tidak spesifik pada jenis perusahaan tertentu. Oleh karena itu verifikasi disini hanya dilakukan sebatas aplikasi ini mampu untuk menjawab kebutuhan fungsional maupun non fungsional dan memastikan aplikasi ini berjalan serta tercapainya tujuan dari dibuatnya penelitian ini.

#### **6.2. Pengujian pada hasil pembuatan sistem antrian pintar**

Pengujian dilakukan setelah pengkodean/implementasi selesai. Pengujian dilakukan berdasarkan kebutuhan fungsional dan kebutuhan non-fungsional. Pengujian kebutuhan fungsional dilakukan dengan menggunakan test use case. Pengujian dengan menggunakan metode ini dilakukan dengan cara menguji skenario utama dan skenario alternative pada masing-masing use case. Berikut ini merupakan hasil dari pengujian untuk masing-masing Use Case Test.

Table 6.1 Test case aktor guest mobile

Kebutuhan Fungsional	Step / langkah	Hasil yang diharapkan	Hasil
Memilih penyedia layanan	- User membuka aplikasi.	Muncul daftar penyedia layanan	Berhasil dan sesuai harapan

Memilih layanan	<ul style="list-style-type: none"> <li>- User membuka aplikasi.</li> <li>- User menekan satu dari penyedia layanan</li> </ul>	Muncul daftar layanan sesuai penyedia layanan yang dipilih	Berhasil dan sesuai harapan
Memilih metode pengambilan nomor antrian	<ul style="list-style-type: none"> <li>- User membuka aplikasi.</li> <li>- User menekan satu dari penyedia layanan</li> <li>- User menekan salah satu dari layanan</li> </ul>	Muncul pilihan pengambilan berdasarkan waktu atau tempat	Berhasil dan sesuai harapan
Memilih waktu kedatangan	<ul style="list-style-type: none"> <li>- User membuka aplikasi.</li> <li>- User menekan satu dari penyedia layanan</li> <li>- User menekan salah satu dari layanan</li> <li>- User memilih datang berdasarkan waktu</li> </ul>	Muncul halaman pemilihan waktu kedatangan user	Berhasil dan sesuai harapan
Melanjutkan antrian yang telah dibuat	<ul style="list-style-type: none"> <li>- User membuka aplikasi.</li> <li>- User menggeser layar dari ujung kiri ke kanan</li> <li>- User menekan “Resume Queue”</li> </ul>	Muncul daftar antrian yang sudah dibuat user	Berhasil dan sesuai harapan
Membatalkan antrian	<ul style="list-style-type: none"> <li>- User membuka aplikasi.</li> <li>- User menggeser layar dari ujung kiri ke kanan</li> </ul>	Antrian yang dipilih user dihapus dari daftar melanjutkan antrian	Berhasil dan sesuai harapan

	<ul style="list-style-type: none"> <li>- User menekan “Resume Queue”</li> <li>- User menekan salah satu penyedia antrian</li> <li>- User memilih layanan antrian yang dilanjutkan</li> <li>- User menekan “Cancel Queue”</li> </ul>		
Melihat kondisi antrian terbaru	<ul style="list-style-type: none"> <li>- User membuka aplikasi.</li> <li>- User menggeser layar dari ujung kiri ke kanan</li> <li>- User menekan “Resume Queue”</li> <li>- User menekan salah satu penyedia antrian</li> <li>- User memilih layanan antrian yang dilanjutkan</li> </ul>	Muncul informasi mengenai jumlah user yang menunggu diantrian yang sama	Berhasil dan sesuai harapan
Otomatis memperbarui nomor antrian ketika terlewati	<ul style="list-style-type: none"> <li>- User membuka aplikasi.</li> <li>- User menggeser layar dari ujung kiri ke kanan</li> <li>- User menekan “Resume Queue”</li> <li>- User menekan salah satu penyedia antrian</li> <li>- User memilih layanan antrian yang dilanjutkan</li> </ul>	User mendapatkan nomor antrian baru	Berhasil dan sesuai harapan

	- Membiarkan nomor antrian terpanggil		
--	---------------------------------------	--	--

Table 6.2 Test case aktor user mobile

Kebutuhan Fungsional	Step / langkah	Hasil yang diharapkan	Hasil
Login	<ul style="list-style-type: none"> <li>- User membuka aplikasi.</li> <li>- User menggeser layar dari ujung kiri ke kanan</li> <li>- User menekan "Login"</li> <li>- User mengisi username</li> <li>- User mengisi password</li> <li>- User menekan "Login"</li> </ul>	Muncul halaman pemilihan loket yang masih kosong	Berhasil dan sesuai harapan
Login dengan data salah	<ul style="list-style-type: none"> <li>- User membuka aplikasi.</li> <li>- User menggeser layar dari ujung kiri ke kanan</li> <li>- User menekan "Login"</li> <li>- User mengisi username</li> <li>- User mengisi password</li> <li>- User menekan "Login"</li> </ul>	Muncul peringatan user atau password salah	Berhasil dan sesuai harapan

Memilih loket	<ul style="list-style-type: none"> <li>- User membuka aplikasi</li> <li>- User melakukan login</li> </ul>	Muncul pilihan loket yang tersedia berdasarkan user	Berhasil dan sesuai harapan
Mengambil nomor pemanggilan selanjutnya	<ul style="list-style-type: none"> <li>- User membuka aplikasi</li> <li>- User melakukan login</li> <li>- User menekan salah satu loket</li> <li>- User menekan icon panah ke kanan</li> </ul>	Nomor antrian dibawah nama loket berubah	Berhasil dan sesuai harapan
Memanggil nomor antrian	<ul style="list-style-type: none"> <li>- User membuka aplikasi</li> <li>- User melakukan login</li> <li>- User menekan salah satu loket</li> <li>- User menekan icon speaker</li> </ul>	Nomor antrian dipanggil dari halaman pemanggilan antrian pada fitur website	Berhasil dan sesuai harapan

Table 6.3 Test case aktor user website

Kebutuhan Fungsional	Step / langkah	Hasil yang diharapkan	Hasil
Login	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User mengisi username</li> <li>- User mengisi password</li> <li>- User menekan "Login"</li> </ul>	Muncul halaman home	Berhasil dan sesuai harapan

Login dengan data salah	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User mengisi username</li> <li>- User mengisi password</li> <li>- User menekan "Login"</li> </ul>	Muncul peringatan user atau password salah	Berhasil dan sesuai harapan
Memilih loket yang kosong	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan loket</li> </ul>	Muncul pilihan loket yang tersedia berdasarkan user	Berhasil dan sesuai harapan
Sudah memiliki loket	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan loket</li> </ul>	Muncul halaman pemanggilan loket	Berhasil dan sesuai harapan
Mengambil nomor pemanggilan selanjutnya	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan salah satu loket</li> <li>- User menekan icon panah ke kanan</li> </ul>	Nomor antrian dibawah nama loket berubah	Berhasil dan sesuai harapan
Memanggil nomor antrian	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan salah satu loket</li> </ul>	Nomor antrian dipanggil dari halaman pemanggilan antrian pada fitur website	Berhasil dan sesuai harapan



	- User menekan icon speaker		
Mengambil nomor antrian	- User membuka website - User melakukan login - User menekan "Queue" - User menekan salah satu layanan	Muncul halaman cetak dengan nomor antrian dan antrian	Berhasil dan sesuai harapan
Melihat daftar pemanggilan nomor antrian	- User membuka website - User melakukan login - User menekan "Monitor"	Muncul halaman daftar loket beserta nomor yang sekarang sedang dilayani	Berhasil dan sesuai harapan

Table 6.4 Test case aktor admin website fitur kelola loket

Kebutuhan Fungsional	Step / langkah	Hasil yang diharapkan	Hasil
Login	- User membuka website - User mengisi username - User mengisi password - User menekan "Login"	Muncul halaman home	Berhasil dan sesuai harapan
Login dengan data salah	- User membuka website - User mengisi username - User mengisi password	Muncul peringatan user atau password salah	Berhasil dan sesuai harapan

	- User menekan “Login”		
Menambah loket	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan loket</li> <li>- User menekan “Add Locket”</li> <li>- User mengisi nama</li> <li>- User memilih “Service”</li> <li>- User mengisi token expired</li> <li>- User memilih “assigned user”</li> <li>- User menekan “+ add loket”</li> </ul>	Loket baru bertambah pada halaman kelola loket	Berhasil dan sesuai harapan
Menambah loket di layanan yang penuh	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan loket</li> <li>- User menekan “Add Locket”</li> <li>- User mengisi nama</li> <li>- User mengisi service</li> <li>- User mengisi token expired</li> <li>- User menekan “+ add loket”</li> </ul>	Muncul warning jumlah loket sudah maksimal	Berhasil dan sesuai harapan

Mencari loket	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan loket</li> <li>- User menekan "Search"</li> <li>- User mengisi "name"</li> <li>- User memilih "services"</li> <li>- User memilih "assigned user"</li> <li>- User menekan "Search"</li> </ul>	Tampil daftar loket sesuai dengan data yang dicari	Berhasil dan sesuai harapan
Mengubah loket	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan loket</li> <li>- User menekan icon centang</li> <li>- User mengisi nama</li> <li>- User memilih "Service"</li> <li>- User mengisi token expired</li> <li>- User memilih "assigned user"</li> <li>- User menekan "+ Update Loket"</li> </ul>	Data loket berubah sesuai dengan yang dirubah	Berhasil dan sesuai harapan

<p>Mengubah loket ke layanan yang penuh</p>	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan loket</li> <li>- User menekan icon centang</li> <li>- User mengisi nama</li> <li>- User memilih "Service"</li> <li>- User mengisi token expired</li> <li>- User memilih "assigned user"</li> <li>- User menekan "+ Update Loket"</li> </ul>	<p>Muncul peringatan bahwa jumlah loket sudah maksimal</p>	<p>Berhasil dan sesuai harapan</p>
<p>Hapus loket</p>	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan loket</li> <li>- User menekan icon tempat sampah</li> <li>- Muncul popup konfirmasi hapus</li> <li>- User menekan "Ok"</li> </ul>	<p>Loket terhapus</p>	<p>Berhasil dan sesuai harapan</p>

Table 6.5 Test case aktor admin website fitur kelola user

Kebutuhan Fungsional	Step / langkah	Hasil yang diharapkan	Hasil
Login	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User mengisi username</li> <li>- User mengisi password</li> <li>- User menekan “Login”</li> </ul>	Muncul halaman home	Berhasil dan sesuai harapan
Login dengan data salah	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User mengisi username</li> <li>- User mengisi password</li> <li>- User menekan “Login”</li> </ul>	Muncul peringatan user atau password salah	Berhasil dan sesuai harapan
Menambah user	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Users”</li> <li>- User menekan “Add User”</li> <li>- User mengisi username</li> <li>- User mengisi name</li> <li>- User mengisi email</li> <li>- User mengisi password</li> </ul>	User baru bertambah pada halaman kelola users	Berhasil dan sesuai harapan

	- User menekan “+ add user”		
Menambah user dengan data yang sudah ada	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Users”</li> <li>- User menekan “Add User”</li> <li>- User mengisi username</li> <li>- User mengisi name</li> <li>- User mengisi email</li> <li>- User mengisi password</li> <li>- User menekan “+ add user”</li> </ul>	Muncul warning user sudah ada	Berhasil dan sesuai harapan
Mencari user	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Users”</li> <li>- User menekan “Search”</li> <li>- User mengisi username</li> <li>- User mengisi name</li> <li>- User mengisi email</li> <li>- User menekan “Search”</li> </ul>	Tampil daftar user sesuai dengan data yang dicari	Berhasil dan sesuai harapan

Mengubah user	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Users”</li> <li>- User menekan icon centang</li> <li>- User mengisi username</li> <li>- User mengisi name</li> <li>- User mengisi email</li> <li>- User mengisi password</li> <li>- User menekan “+ Update User”</li> </ul>	Data user berubah sesuai dengan yang dirubah	Berhasil dan sesuai harapan
Mengubah user dengan data yang sudah ada	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Users”</li> <li>- User menekan icon centang</li> <li>- User mengisi username</li> <li>- User mengisi name</li> <li>- User mengisi email</li> <li>- User mengisi password</li> <li>- User menekan “+ Update User”</li> </ul>	Muncul peringatan bahwa user sudah ada	Berhasil dan sesuai harapan

Hapus user	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Users”</li> <li>- User menekan icon tempat sampah</li> <li>- Muncul popup konfirmasi hapus</li> <li>- User menekan “Ok”</li> </ul>	User terhapus	Berhasil dan sesuai harapan
------------	---	---------------	-----------------------------

Table 6.6 Test case admin website fitur kelola layanan / *services*

Kebutuhan Fungsional	Step / langkah	Hasil yang diharapkan	Hasil
Login	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User mengisi username</li> <li>- User mengisi password</li> <li>- User menekan “Login”</li> </ul>	Muncul halaman home	Berhasil dan sesuai harapan
Login dengan data salah	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User mengisi username</li> <li>- User mengisi password</li> <li>- User menekan “Login”</li> </ul>	Muncul peringatan user atau password salah	Berhasil dan sesuai harapan



Menambah layanan	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Services”</li> <li>- User menekan “Add Service”</li> <li>- User mengisi name</li> <li>- User mengisi description</li> <li>- User mengisi number of loket</li> <li>- User mengisi code</li> <li>- User menekan “+ Add Service”</li> </ul>	Layanan baru bertambah pada halaman kelola layanan / service	Berhasil dan sesuai harapan
Menambah layanan dengan data yang sudah ada	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Services”</li> <li>- User menekan “Add Service”</li> <li>- User mengisi name</li> <li>- User mengisi description</li> <li>- User mengisi number of loket</li> <li>- User mengisi code</li> <li>- User menekan “+ Add Service”</li> </ul>	Muncul warning layanan sudah ada	Berhasil dan sesuai harapan

Mencari layanan	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Services”</li> <li>- User menekan “Search”</li> <li>- User mengisi name</li> <li>- User menekan “Search”</li> </ul>	Tampil daftar layanan sesuai dengan data yang dicari	Berhasil dan sesuai harapan
Mengubah layanan	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Services”</li> <li>- User menekan icon centang</li> <li>- User mengisi name</li> <li>- User mengisi description</li> <li>- User mengisi number of loket</li> <li>- User mengisi code</li> <li>- User menekan “+ Update Service”</li> </ul>	Data layanan berubah sesuai dengan yang dirubah	Berhasil dan sesuai harapan
Mengubah layanan dengan data yang sudah ada	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Services”</li> </ul>	Muncul peringatan bahwa layanan sudah ada	Berhasil dan sesuai harapan

	<ul style="list-style-type: none"> <li>- User menekan icon centang</li> <li>- User mengisi name</li> <li>- User mengisi description</li> <li>- User mengisi number of loket</li> <li>- User mengisi code</li> <li>- User menekan “+ Update Service”</li> </ul>		
Hapus layanan	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Services”</li> <li>- User menekan icon tempat sampah</li> <li>- Muncul popup konfirmasi hapus</li> <li>- User menekan “Ok”</li> </ul>	Layanan terhapus	Berhasil dan sesuai harapan

Table 6.7 Test case aktor administrator website fitur user

Kebutuhan Fungsional	Step / langkah	Hasil yang diharapkan	Hasil
Login	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User mengisi username</li> <li>- User mengisi password</li> </ul>	Muncul halaman home	Berhasil dan sesuai harapan

	- User menekan “Login”		
Login dengan data salah	- User membuka website - User mengisi username - User mengisi password - User menekan “Login”	Muncul peringatan user atau password salah	Berhasil dan sesuai harapan
Menambah user	- User membuka website - User melakukan login - User menekan “Users” - User menekan “Add User” - User mengisi username - User mengisi name - User mengisi email - User mengisi role - User mengisi password - User menekan “+ add user”	User baru bertambah pada halaman kelola users	Berhasil dan sesuai harapan
Menambah user dengan data yang sudah ada	- User membuka website - User melakukan login - User menekan “Users”	Muncul warning user sudah ada	Berhasil dan sesuai harapan

	<ul style="list-style-type: none"> <li>- User menekan “Add User”</li> <li>- User mengisi username</li> <li>- User mengisi name</li> <li>- User mengisi email</li> <li>- User mengisi role</li> <li>- User mengisi password</li> <li>- User menekan “+ add user”</li> </ul>		
Mencari user	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Users”</li> <li>- User menekan “Search”</li> <li>- User mengisi username</li> <li>- User mengisi name</li> <li>- User mengisi email</li> <li>- User mengisi role</li> <li>- User menekan “Search”</li> </ul>	Tampil daftar user sesuai dengan data yang dicari	Berhasil dan sesuai harapan
Mengubah user	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> </ul>	Data user berubah sesuai dengan yang dirubah	Berhasil dan sesuai harapan

	<ul style="list-style-type: none"> <li>- User menekan “Users”</li> <li>- User menekan icon centang</li> <li>- User mengisi username</li> <li>- User mengisi name</li> <li>- User mengisi email</li> <li>- User mengisi role</li> <li>- User mengisi password</li> <li>- User menekan “+ Update User”</li> </ul>		
Mengubah user dengan data yang sudah ada	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Users”</li> <li>- User menekan icon centang</li> <li>- User mengisi username</li> <li>- User mengisi name</li> <li>- User mengisi email</li> <li>- User mengisi role</li> <li>- User mengisi password</li> <li>- User menekan “+ Update User”</li> </ul>	Muncul peringatan bahwa user sudah ada	Berhasil dan sesuai harapan

Hapus user	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Users”</li> <li>- User menekan icon tempat sampah</li> <li>- Muncul popup konfirmasi hapus</li> <li>- User menekan “Ok”</li> </ul>	User terhapus	Berhasil dan sesuai harapan
Assign user ke penyedia layanan	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Users”</li> <li>- User menekan icon kubus</li> <li>- User memilih “Service Provider”</li> <li>- User menekan “Update”</li> </ul>	Muncul notifikasi user berhasil di assign ke penyedia layanan	Berhasil dan sesuai harapan

Table 6.8 Test case aktor administrator website fitur penyedia layanan

Kebutuhan Fungsional	Step / langkah	Hasil yang diharapkan	Hasil
Login	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User mengisi username</li> </ul>	Muncul halaman home	Berhasil dan sesuai harapan

	<ul style="list-style-type: none"> <li>- User mengisi password</li> <li>- User menekan “Login”</li> </ul>		
Login dengan data salah	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User mengisi username</li> <li>- User mengisi password</li> <li>- User menekan “Login”</li> </ul>	Muncul peringatan user atau password salah	Berhasil dan sesuai harapan
Menambah penyedia layanan	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Service Provider”</li> <li>- User menekan “Add Service Provider”</li> <li>- User mengisi name</li> <li>- User mengisi description</li> <li>- User mengisi type</li> <li>- User memilih icon</li> <li>- User mengisi inner distance</li> <li>- User mengisi outer distance</li> <li>- User memilih lokasi pada map</li> </ul>	Penyedia layanan baru bertambah pada halaman kelola penyedia layanan	Berhasil dan sesuai harapan



	- User menekan “+ Add Service Provider”		
Menambah penyedia layanan dengan data yang sudah ada	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Service Provider”</li> <li>- User menekan “Add Service Provider”</li> <li>- User mengisi name</li> <li>- User mengisi description</li> <li>- User mengisi type</li> <li>- User memilih icon</li> <li>- User mengisi inner distance</li> <li>- User mengisi outer distance</li> <li>- User memilih lokasi pada map</li> <li>- User menekan “+ Add Service Provider”</li> </ul>	Muncul warning penyedia layanan sudah ada	Berhasil dan sesuai harapan
Mencari penyedia layanan	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Services Provider”</li> </ul>	Tampil daftar penyedia layanan sesuai dengan data yang dicari	Berhasil dan sesuai harapan

	<ul style="list-style-type: none"> <li>- User menekan “Search”</li> <li>- User mengisi name</li> <li>- User mengisi description</li> <li>- User mengisi type</li> <li>- User menekan “Search”</li> </ul>		
Mengubah penyedia layanan	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Services Provider”</li> <li>- User menekan icon centang</li> <li>- User mengisi name</li> <li>- User mengisi description</li> <li>- User mengisi type</li> <li>- User memilih icon</li> <li>- User mengisi inner distance</li> <li>- User mengisi outer distance</li> <li>- User memilih lokasi pada map</li> <li>- User menekan “+ Update Service Provider”</li> </ul>	Data penyedia layanan berubah sesuai dengan yang dirubah	Berhasil dan sesuai harapan
Mengubah penyedia layanan	<ul style="list-style-type: none"> <li>- User membuka website</li> </ul>	Muncul peringatan bahwa	Berhasil dan sesuai harapan

dengan data yang sudah ada	<ul style="list-style-type: none"> <li>- User melakukan login</li> <li>- User menekan “Services Provider”</li> <li>- User menekan icon centang</li> <li>- User mengisi name</li> <li>- User mengisi description</li> <li>- User mengisi type</li> <li>- User memilih icon</li> <li>- User mengisi inner distance</li> <li>- User mengisi outer distance</li> <li>- User memilih lokasi pada map</li> <li>- User menekan “+ Update Service Provider”</li> </ul>	penyedia layanan sudah ada	
Hapus penyedia layanan	<ul style="list-style-type: none"> <li>- User membuka website</li> <li>- User melakukan login</li> <li>- User menekan “Services Provider”</li> <li>- User menekan icon tempat sampah</li> <li>- Muncul popup konfirmasi hapus</li> <li>- User menekan “Ok”</li> </ul>	Penyedia layanan terhapus	Berhasil dan sesuai harapan

(halaman ini sengaja dikosongkan)

## **BAB VII**

### **KESIMPULAN DAN SARAN**

Bab ini membahas mengenai kesimpulan yang diperoleh dari semua proses yang telah dilakukan dan menguraikan beberapa saran terhadap pengembangan selanjutnya.

#### **7.1. Kesimpulan**

Berdasarkan dengan pengerjaan tugas akhir dengan judul "Rancang Bangun Aplikasi Antrian Pintar (Studi Kasus: Service Desk DPTSI)" yang telah dilakukan dapat disimpulkan beberapa hal sebagai berikut:

- Sistem antrian pintar dengan fitur multi tenant, multi layanan, multi loket dan pembaruan nomor antrian secara otomatis berhasil dibuat dengan baik dan telah memenuhi semua aspek yang terdapat pada fase pengujian sehingga alat siap untuk digunakan.
- Sistem antrian pintar dapat membantu pihak-pihak yang menggunakan jasa antrian sehingga mereka bisa memanfaatkan waktu mereka agar lebih bermanfaat.

#### **7.2. Saran**

Berdasarkan kesimpulan dari tugas akhir ini, maka penulis merekomendasikan beberapa saran yang diberikan untuk pengembangan kedepan sebagai berikut :

- Perlu ditambahkan fitur untuk menampilkan daftar riwayat antrian atau halaman report agar penyedia jasa dapat memantau antrian yang mereka miliki setiap saat
- Perlu penambahan fitur seperti dashboard agar memudahkan admin dalam memantau website.

## Daftar Pustaka

- [1] V. Pimentel and B. G. Nickerson, "Communicating and displaying real-time data with websocket," *IEEE Internet Comput.*, vol. 16, no. 4, pp. 45–53, 2012.
- [2] Y. Furukawa, "Web-based control application using WebSocket," *ICALEPCS2011*, pp. 673–675, 2011.
- [3] M. Singhal and A. Shukla, "Implementation of location based services in android using GPS and web services," *Int. J. Comput. Sci. Issues IJCSI*, vol. 9, no. 1, p. 237, 2012.
- [4] C. N. Alam, K. Manaf, A. R. Atmadja, and D. K. Aurum, "Implementation of haversine formula for counting event visitor in the radius based on Android application," in *2016 4th International Conference on Cyber and IT Service Management*, 2016, pp. 1–6.
- [5] T. Dutoit, "High-quality text-to-speech synthesis: An overview," *J. Electr. Electron. Eng. Aust.*, vol. 17, pp. 25–36, 1997.
- [6] S. Tilkov and S. Vinoski, "Node. js: Using JavaScript to build high-performance network programs," *IEEE Internet Comput.*, vol. 14, no. 6, pp. 80–83, 2010.
- [7] R. Native, "React Native," *Última Revisión*, 2018.
- [8] A. Sheth and C. Ramakrishnan, "Semantic (Web) Technology In Action: Ontology Driven Information Systems for Search, Integration and Analysis," *IEEE Data Eng. Bull. Spec. Issue Mak. Semantic Web Real*, vol. 26, no. 4, pp. 40–8, 2003.
- [9] K. Sugiyama, K. Hatano, and M. Yoshikawa, "Adaptive Web search based on user profile constructed without any effort from users," *Thirteen. Int. World Wide Web Conf. Proc. WWW2004*, pp. 675–684, 2004, doi: 10.1145/988672.988764.

## BIODATA PENULIS

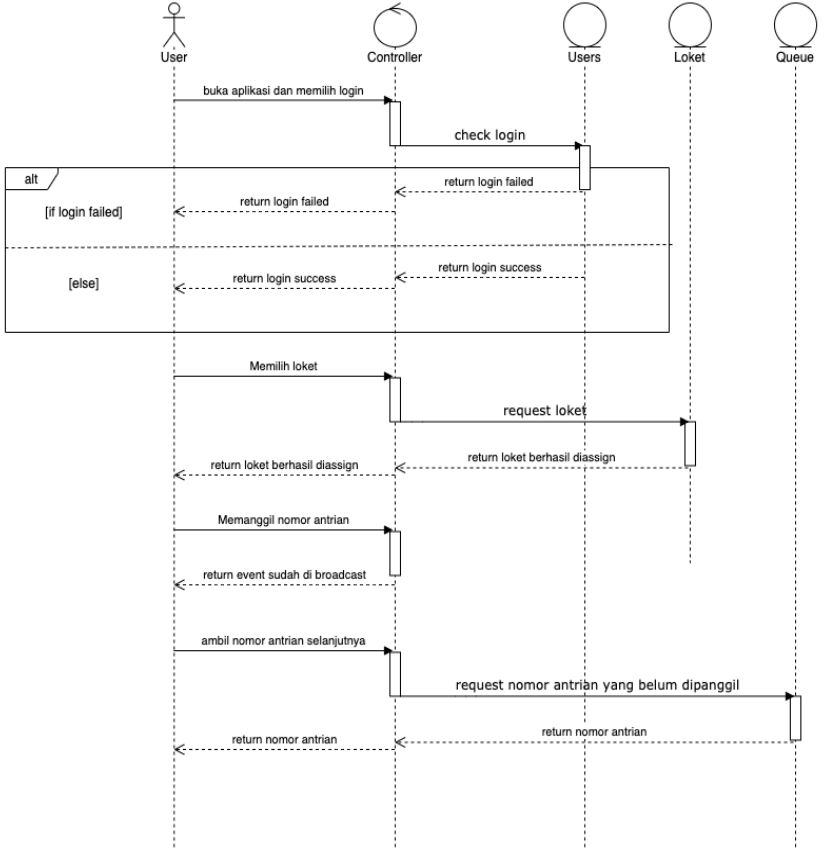


Penulis bernama lengkap Arief Putranto, dilahirkan di kota Kediri, 9 Juli 1995, merupakan anak ketiga dari tiga bersaudara. Penulis telah menempuh pendidikan formal di SDN Sukorame 2 Kediri, SMPN 1 Kediri, dan SMAN 1 Kediri. Penulis meneruskan pendidikan tinggi negeri di Departemen Sistem Informasi Institut Teknologi Sepuluh Nopember, Surabaya dan terdaftar dengan NRP 05211340000124. Pengalaman selama menjadi mahasiswa di ITS adalah sebagai salah satu anggota syiar pada organisasi KISI. Penulis juga pernah melaksanakan kerja praktik selama satu bulan setengah pada tahun 2016 di PT Bea Cukai Tanjung Perak Surabaya, Surabaya.

Penulis mengambil bidang minat Infrastruktur dan Keamanan Teknologi Informasi (IKTI) pada penyelesaian Penelitian Tugas Akhir dengan topik Rancang Bangun Aplikasi Antrian Pintar (Studi Kasus: Service Desk DPTSI). Untuk menghubungi penulis, dapat melalui e-mail: [ariefputranto0@gmail.com](mailto:ariefputranto0@gmail.com).

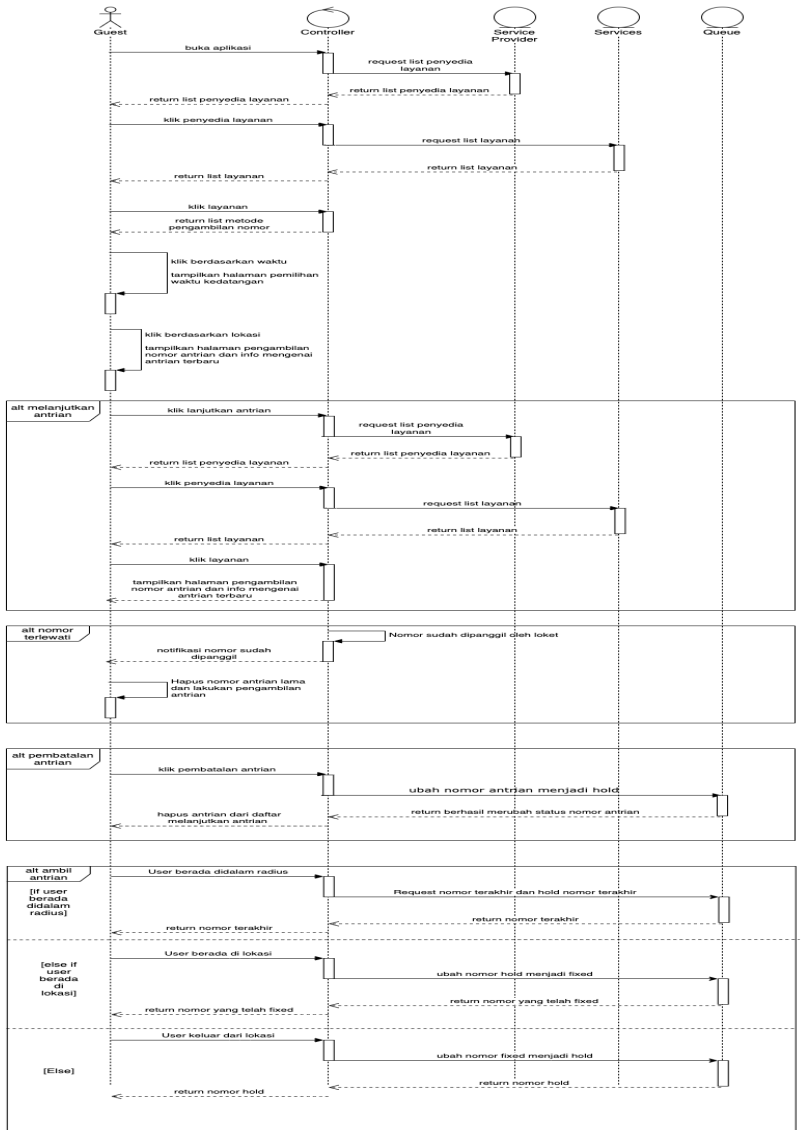
# LAMPIRAN A

## A.1. Sequence Diagram Sistem Antrian Pintar

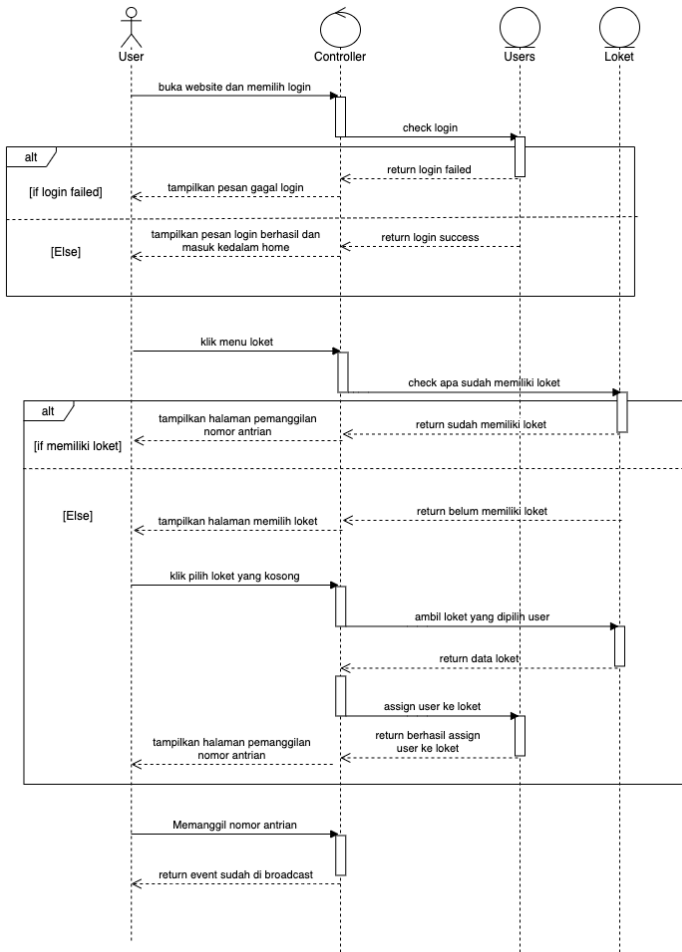


Gambar 0.1 Sequence diagram *user mobile*

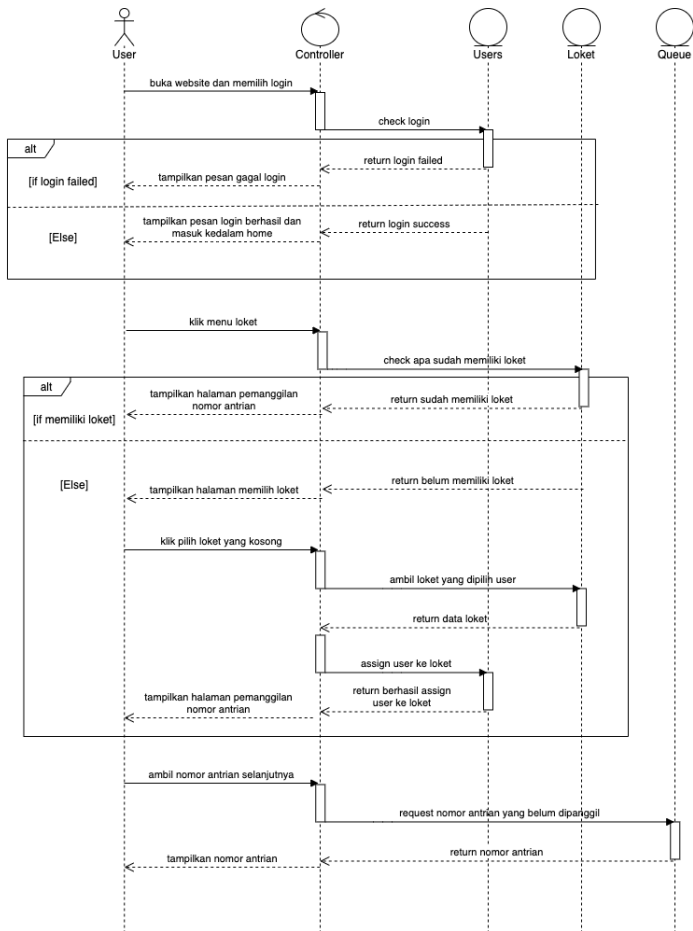




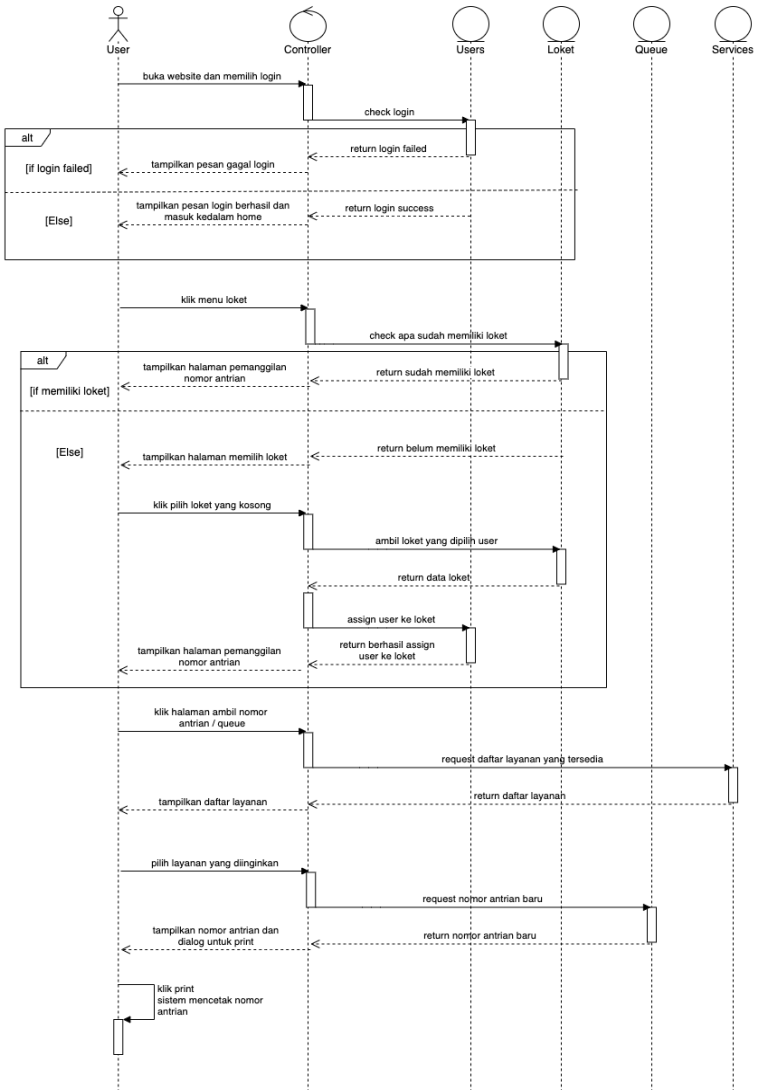
Gambar 0.2 Sequence *guest mobile*



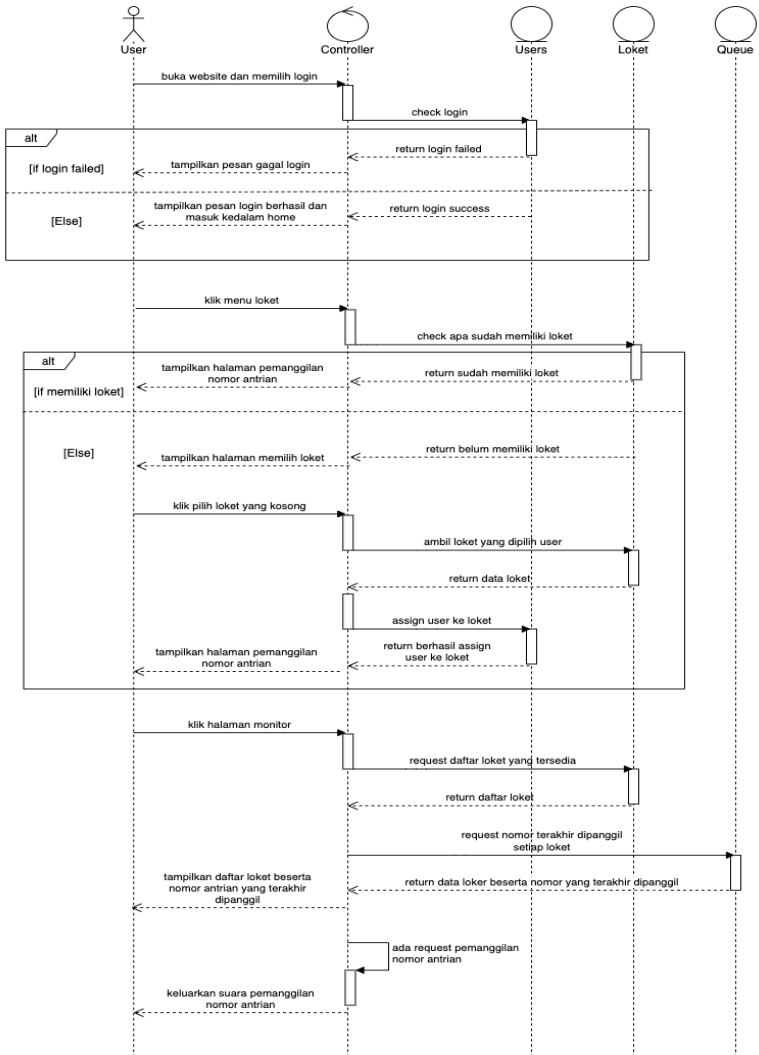
Gambar 0.3 Sequence diagram *user* website pemanggilan nomor antrian



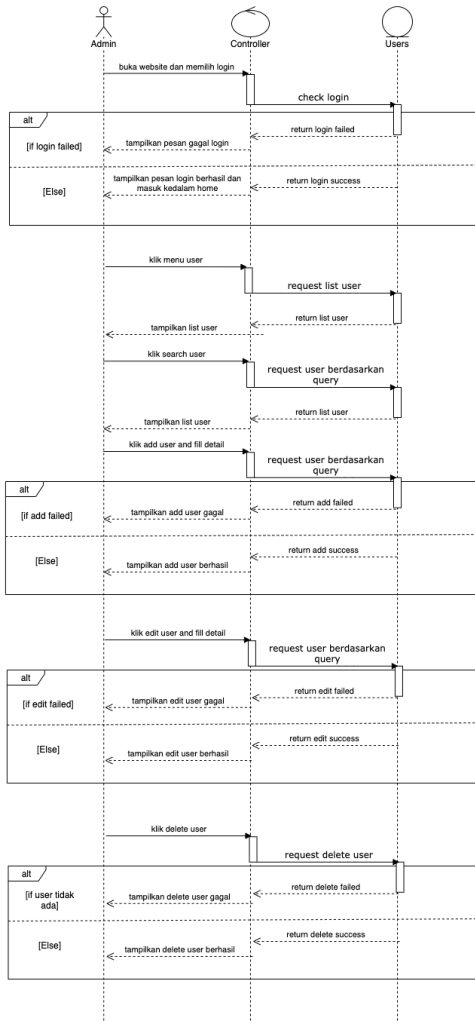
Gambar 0.4 Sequence diagram user website melanjutkan nomor antrian



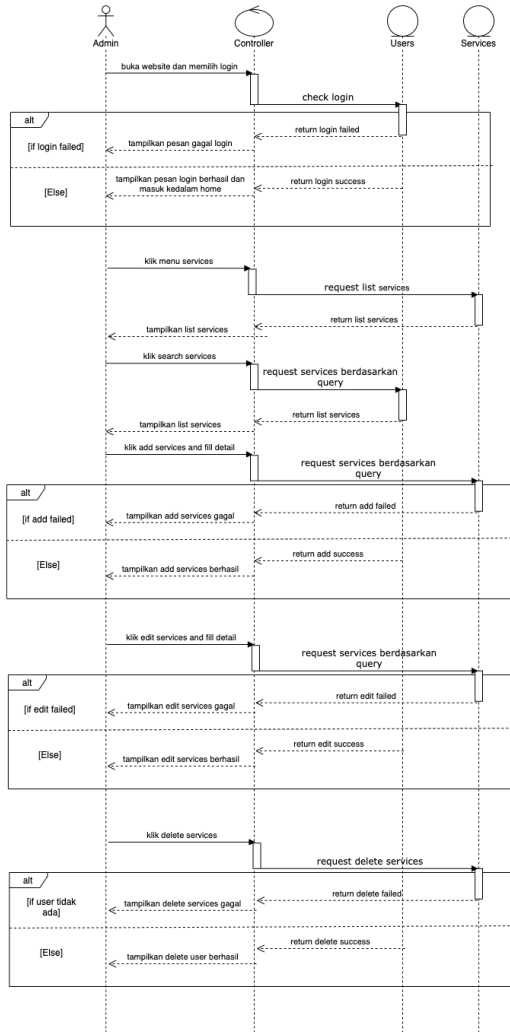
Gambar 0.5 Sequence diagram user website ambil nomor antrian



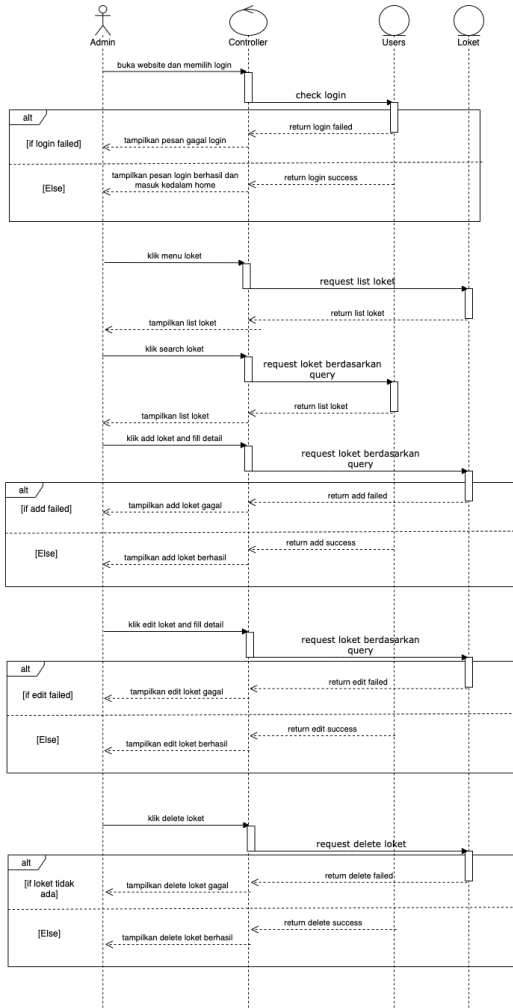
Gambar 0.6 Sequence diagram user website daftar pemanggilan nomor antrian



Gambar 0.7 Sequence diagram *admin* website kelola *user*

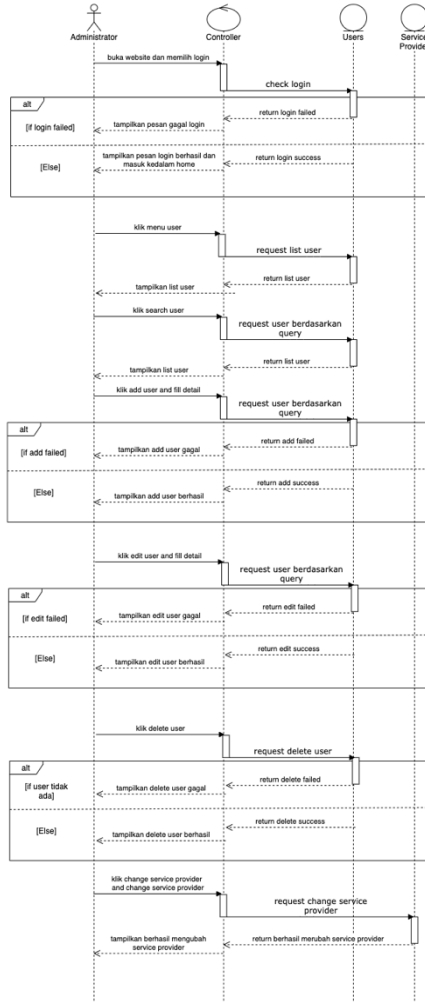


Gambar 0.8 Sequence diagram *admin* website kelola *services*

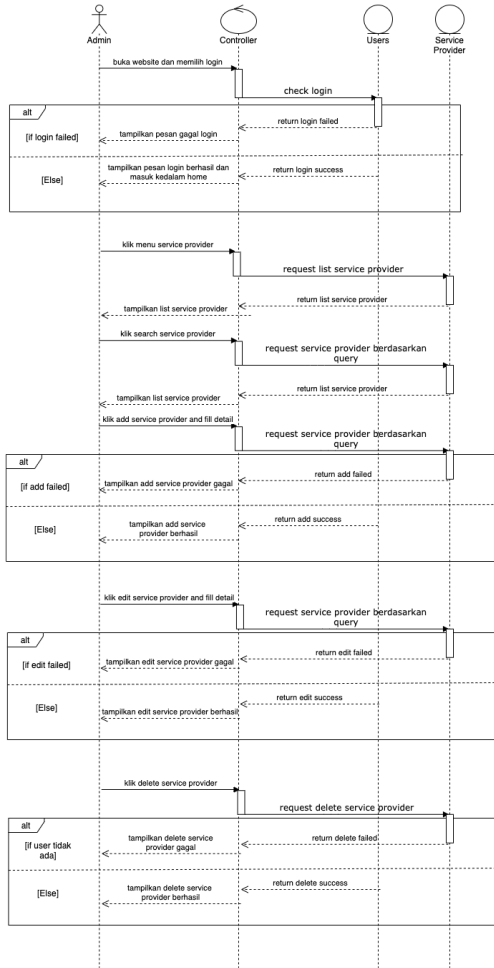


Gambar 0.9 Sequence diagram *admin* website kelola loket





Gambar 0.10 Sequence diagram *administrator* website kelola *user*



Gambar 0.11 Sequence diagram *administrator* website kelola *service provider*