

**IMPLEMENTASI VOICE RECOGNITION  
MENGUNAKAN METODE HIDDEN MARKOV  
MODELS PADA PEMBANGUNAN SISTEM  
RESEPSIONIS BERBASIS MOBILE**

**Tugas Akhir**

**Diajukan untuk Memenuhi Salah Satu Persyaratan Mencapai Derajat  
Sarjana Informatika**



Dibuat Oleh:

**I GUSTI MADE SUMARIANA**

**16 07 09030**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ATMA JAYA YOGYAKARTA  
2020**

## HALAMAN PENGESAHAN

Tugas Akhir Berjudul

IMPLEMENTASI VOICE RECOGNITION MENGGUNAKAN METODE HIDDEN MARKOV MODELS  
PADA PEMBANGUNAN SISTEM RESEPSIONIS BERBASIS MOBILE

yang disusun oleh

I GUSTI MADE SUMARIANA

160709030

dinyatakan telah memenuhi syarat pada tanggal 28 Juli 2020

		Keterangan
Dosen Pembimbing 1	: Yulius Harjoseputro, ST., MT.	Telah menyetujui
Dosen Pembimbing 2	: B. Yudi Dwiandiyanta, ST., MT.	Telah menyetujui
Tim Penguji		
Penguji 1	: Yulius Harjoseputro, ST., MT.	Telah menyetujui
Penguji 2	: Dr. Pranowo, S.T., M.T.	Telah menyetujui
Penguji 3	: Prof. Ir. Suyoto, MSc., PhD	Telah menyetujui

Yogyakarta, 28 Juli 2020

Universitas Atma Jaya Yogyakarta

Fakultas Teknologi Industri

Dekan

ttd

Dr. A. Teguh Siswantoro, M.Sc



# **PERNYATAAN ORISINALITAS & PUBLIKASI ILMIAH**

Saya yang bertanda tangan di bawah ini:

Nama Lengkap : I Gusti Made Sumariana  
NPM : 160709030  
Program Studi : Informatika  
Fakultas : Teknologi Industri  
Judul Penelitian : Implementasi Voice Recognition Menggunakan  
Metode Hidden Markov Models Pada Pembangunan Sistem Resepsionis Berbasis  
Mobile

Menyatakan dengan ini:

1. Tugas Akhir ini adalah benar tidak merupakan salinan sebagian atau keseluruhan dari karya penelitian lain.
2. Memberikan kepada Universitas Atma Jaya Yogyakarta atas penelitian ini, berupa Hak untuk menyimpan, mengelola, mendistribusikan, dan menampilkan hasil penelitian selama tetap mencantumkan nama penulis.
3. Bersedia menanggung secara pribadi segala bentuk tuntutan hukum atas pelanggaran Hak Cipta dalam pembuatan Tugas Akhir ini.

Demikianlah pernyataan ini dibuat dan dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 22 Juli 2020

Yang menyatakan,

I Gusti Made Sumariana  
160709030

# **PERNYATAAN PERSETUJUAN DARI INSTANSI ASAL PENELITIAN**

Saya yang bertanda tangan di bawah ini:

Nama Lengkap Pembimbing : I Nyoman Darma Kotama

Jabatan : Manager

Departemen : Mobile Apps

Menyatakan dengan ini:

Nama Lengkap : I Gusti Made Sumariana

NPM : 16 070 9030

Program Studi : Informatika

Fakultas : Teknologi Industri

Judul Penelitian : Implementasi Voice Recognition Menggunakan Metode Hidden Markov Models Pada Pembangunan Sistem Resepsionis Berbasis Mobile

1. Penelitian telah selesai dilaksanakan pada perusahaan.
2. Perusahaan telah melakukan sidang internal berupa kelayakan penelitian ini dan akan mencantumkan lembar penilaian secara tertutup kepada pihak universitas sebagai bagian dari nilai akhir mahasiswa.
3. Memberikan kepada Instansi Penelitian dan Universitas Atma Jaya Yogyakarta atas penelitian ini, berupa hak untuk menyimpan, mengelola, mendistribusikan, dan menampilkan hasil penelitian selama tetap mencantumkan nama penulis.

Demikianlah pernyataan ini dibuat dan dapat dipergunakan sebagaimana mestinya.

Denpasar, 22 Juli 2020

Yang menyatakan,

I Nyoman Darma Kotama  
Mobile Apps Manager

## **HALAMAN PERSEMBAHAN**

**Semua akan indah pada waktu-Nya**



## KATA PENGANTAR

Puji dan syukur penulis haturkan kepada Tuhan Yang Maha Esa karena berkat rahmat dan karunia-Nya penulis dapat menyelesaikan pembuatan tugas akhir “Rancang Bangun Aplikasi Pariwisata Berbasis Lokasi” ini dengan baik.

Penulisan tugas akhir ini bertujuan untuk memenuhi salah satu syarat untuk mencapai derajat sarjana Informatika dari Program Studi Informatika, Fakultas Teknologi Industri di Universitas Atma Jaya Yogyakarta.

Penulis menyadari bahwa dalam pembuatan tugas akhir ini penulis telah mendapatkan bantuan, bimbingan, dan dorongan dari banyak pihak. Untuk itu, pada kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Ida Sang Hyang Widhi Wasa yang selalu membimbing dalam iman-Nya, memberikan berkat-Nya, dan menyertai penulis selalu.
2. Bapak Dr. A. Teguh Siswantoro, selaku Dekan Fakultas Teknologi Industri, Universitas Atma Jaya Yogyakarta.
3. Bapak Yulius Harjoseputro, S.T., M.T., selaku dosen pembimbing I yang telah membimbing dan memberikan masukan serta motivasi kepada penulis untuk menyelesaikan tugas akhir ini.
4. Bapak B. Yudi Dwiandiyanta, S.T., M.T., selaku dosen pembimbing II yang telah membimbing dan memberikan masukan serta motivasi kepada penulis untuk menyelesaikan tugas akhir ini.

Demikian laporan tugas akhir ini dibuat, dan penulis mengucapkan terima kasih kepada semua pihak. Semoga laporan ini dapat bermanfaat bagi pembaca.

Yogyakarta, 22 Juli 2020

I Gusti Made Sumariana

160709030

# DAFTAR ISI

IMPLEMENTASI VOICE RECOGNITION MENGGUNAKAN METODE HIDDEN MARKOV MODELS PADA PEMBANGUNAN SISTEM RESEPSIONIS BERBASIS MOBILE .....	i
PERNYATAAN ORISINALITAS & PUBLIKASI ILMIAH.....	iii
PERNYATAAN PERSETUJUAN DARI INSTANSI ASAL PENELITIAN .....	iv
HALAMAN PERSEMBAHAN .....	v
KATA PENGANTAR .....	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR .....	ix
DAFTAR TABEL.....	xi
INTISARI.....	xii
BAB I. PENDAHULUAN .....	1
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	3
1.3. Batasan Masalah.....	3
1.4. Tujuan Penelitian .....	4
1.5. Metode Penelitian.....	4
1.6. Sistematika Penulisan .....	5
BAB II. TINJAUAN PUSTAKA.....	7
BAB III. LANDASAN TEORI.....	11
3.1. Resepsionis / <i>Front office</i> .....	11
3.2. Android Studio .....	11
3.3. Kotlin .....	11
3.4. Voice Recognition.....	12
3.5. Hidden Markov Models .....	13
3.6. Firebase .....	15
3.7. Firebase Realtime Database .....	16

3.8.	Firebase Cloud Messaging .....	16
3.9.	MVP (Model View Presenter) .....	17
<b>BAB IV. ANALISIS DAN PERANCANGAN SISTEM .....</b>		<b>18</b>
4.1.	Analisis Sistem.....	18
4.2.	Lingkup Masalah.....	18
4.3.	Perspektif Produk .....	19
4.4.	Fungsi Produk .....	20
4.4.1	Use Case Diagram.....	20
4.4.2	Tabel Use Case Diagram.....	21
4.4.3	Physical Data Model (PDM).....	53
4.5.	Kebutuhan Antarmuka .....	53
4.5.1.	Antarmuka Pengguna .....	53
4.5.2.	Antarmuka Perangkat Keras .....	54
4.5.3.	Antarmuka Perangkat Lunak .....	54
4.6.	Perancangan .....	54
4.6.1.	Perancangan Arsitektur .....	54
4.6.2.	Perancangan Antarmuka .....	57
<b>BAB V. IMPLEMENTASI DAN PENGUJIAN SISTEM .....</b>		<b>77</b>
5.1.	Implementasi Sistem Implementasi Antarmuka .....	77
5.2.	Pengujian Fungsionalitas Perangkat Lunak .....	94
5.3.	Hasil Pengujian Terhadap Pengguna .....	124
<b>BAB VI. PENUTUP .....</b>		<b>126</b>
6.1.	Kesimpulan .....	126
6.2.	Saran.....	126
<b>DAFTAR PUSTAKA .....</b>		<b>127</b>



## DAFTAR GAMBAR

Gambar 3. 1 Langkah Proses <i>Voice Recognition</i> .....	14
Gambar 3. 2 Hidden Markov Models dengan 2 keadaan dan 2 simbol output yaitu A dan B .....	14
Gambar 4. 1 Diagram Use Case Sistem Virtual Receptionist.....	20
Gambar 4. 2 Physical Data Model Sistem Virtual Receptionist. ....	53
Gambar 4. 3 Overview Sistem Virtual Receptionist.....	55
Gambar 4. 4 Arsitektur Perangkat Lunak .....	56
Gambar 4. 5 Deployment Diagram .....	57
Gambar 4. 6 Halaman Utama Aplikasi Receptionist .....	58
Gambar 4. 7 Dialog Form Input Nama Aplikasi Receptionist.....	59
Gambar 4. 8 Alert Dialog Voice Assistant .....	59
Gambar 4. 9 Halaman Guest Pada Aplikasi Receptionist.....	60
Gambar 4. 10 Halaman Appointment Aplikasi Receptionist.....	61
Gambar 4. 11 Halaman Website, Mobile Apps & Marketing Aplikasi Receptionist .....	62
Gambar 4. 12 Halaman Programming School .....	63
Gambar 4. 13 Alert Dialog Loading Aplikasi Receptionist.....	64
Gambar 4. 14 Alert Dialog Success Aplikasi Receptionist.....	65
Gambar 4. 15 Alert Dialog Success-Redirected Aplikasi Receptionist.....	66
Gambar 4. 16 Alert Dialog All Busy Aplikasi Receptionist.....	67
Gambar 4. 17 Alert Dialog Error Aplikasi Receptionist.....	68
Gambar 4. 18 Halaman Login Aplikasi Receptionist-Compact.....	69
Gambar 4. 19 Halaman Dashboard Aplikasi Receptionist-Compact.....	70
Gambar 4. 20 Bottom Sheet Dialog Detail Profile User Aplikasi Receptionist-Compact .....	71
Gambar 4. 21 Bottom Sheet Dialog Update Profile.....	72
Gambar 4. 22 Halaman Pegawai Aplikasi Receptionist-Compact .....	73
Gambar 4. 23 Bottom Sheet Register Employee .....	74
Gambar 4. 24 Alert Dialog Panggilan Notifikasi Masuk.....	75
Gambar 4. 25 Halaman Riwayat tamu Aplikasi Receptionist-Compact.....	76

Gambar 5. 1 Bottom Sheet Register Pegawai .....	77
Gambar 5. 2 Fungsi handleWriteMember.....	78
Gambar 5. 3 Halaman Dashboard Aplikasi Receptionist-compact.....	78
Gambar 5. 4 Fungsi updateStatus Pada DetailProfilePresenter .....	79
Gambar 5. 5 Halaman Website, Mobile Apps & Marketing.....	79
Gambar 5. 6 Fungsi fetchMemberList pada MemberPresenter .....	80
Gambar 5. 7 Dialog Loading Panggilan Notifikasi.....	80
Gambar 5. 8 Fungsi startTargeting .....	81
Gambar 5. 9 Fungsi listenToTargetResponse .....	81
Gambar 5. 10 Fungsi listenAcceptTarget dan setOnWaiting Pada NotifyPresenter .....	82
Gambar 5. 11 fungsi redirectTarget pada NotifyPresenter .....	83
Gambar 5. 12 Incoming Push Notification .....	84
Gambar 5. 13 Alert Dialog Success .....	84
Gambar 5. 14 Alert Dialog Success Redirected.....	85
Gambar 5. 15 Alert Dialog All Busy .....	85
Gambar 5. 16 Voice Assistant Dialog.....	86
Gambar 5. 17 Inisialisasi Instance dari kelas TextToSpeech.....	87
Gambar 5. 18 Sistem Menginisialisasi Instance dari kelas SpeechRecognizer .....	87
Gambar 5. 19 Tombol mikropon mengtrigger fungsi startListening .....	88
Gambar 5. 20 SpeechRecognition Callback.....	89
Gambar 5. 21 Fungsi Abstrak onResult .....	90
Gambar 5. 22 Ilustrasi Forward pass recursion.....	91
Gambar 5. 23 Ilustrasi forward algorithm yang memperlihatkan nilai dari $\alpha_j(t)$ pada setiap cell.....	92
Gambar 5. 24 Forward Algorithm yang dilakukan oleh object SpeechRecognizer	92
Gambar 5. 25 Word Error Rate Formula .....	93

## DAFTAR TABEL

Tabel 2. 1 Perbandingan Penelitian.....	9
Tabel 5. 1 Tingkat Akurasi <i>Voice Recognition</i> .....	93
Tabel 5. 2 Pengujian Fungsionalitas Perangkat Lunak .....	95
Tabel 5. 3 Hasil Penilaian Kuisisioner.....	125



# INTISARI

## Implementasi Voice Recognition Menggunakan Metode Hidden Markov Models Pada Pembangunan Sistem Resepsionis Berbasis Mobile

I Gusti Made Sumaraina

16 070 9030

PT. Timedoor Indonesia adalah salah satu perusahaan IT terbesar di Denpasar, Bali. Namun sampai saat ini belum memiliki pegawai resepsionis. Selama ini PT. Timedoor Indonesia menerima tamu dengan cara konvensional, yaitu tamu yang datang mengetuk pintu ruang kerja pegawai ketika ingin bertemu dengan salah seorang pegawai. Cara konvensional seperti ini akan menjadi masalah ketika tamu yang datang merupakan calon konsumen baru atau kurir paket yang tidak tahu apa-apa tentang perusahaan, tamu seperti ini biasanya akan menunggu di lobby sambil menunggu orang yang lewat untuk ditanyai.

Oleh karena itu dibutuhkan Sistem *Virtual Receptionist* yang dapat menggantikan posisi seorang pegawai resepsionis secara garis besar yaitu menjadi perantara antara tamu dan pegawai perusahaan dengan mengimplementasi fitur *Voice Recognition* menggunakan metode *hidden markov model (HMM)*. Sistem ini akan berbasis *mobile* dengan platform *Android* menggunakan bahasa pemrograman *Kotlin* dengan design pattern *Model View Presenter (MVP)* dan *Firebase* sebagai *Backend-service*.

Hasil dari penelitian ini adalah sebuah Sistem Virtual Receptionist. Dengan adanya penerapan sistem virtual receptionist ini, perusahaan dapat memfasilitasi tamu yang datang untuk bertemu dengan pegawai secara efektif dan efisien. Sistem akan menjadi jembatan antara tamu dan pegawai yang ingin bertemu untuk suatu kepentingan. Selain itu sistem ini juga dilengkapi dengan fitur *voice assistant* yang memiliki tingkat akurasi sebesar 86.5868% guna memberikan pengalaman lebih saat menggunakan Sistem *Virtual Receptionist*.

**Kata Kunci:** Resepsionis, *Virtual Receptionist*, *Voice Recognition*, *Hidden Markov Models*, *Android*, *Firebase*

Dosen Pembimbing I : Yulius Harjoseputro, S.T., M.T.

Dosen Pembimbing II : B. Yudi Dwiandiyanta, S.T., M.T.

Jadwal Sidang Tugas Akhir : 28 Juli 2020

# BAB I. PENDAHULUAN

## 1.1. Latar Belakang

Dalam sebuah perusahaan sering terdengar istilah *front office*. *Front office* adalah departemen yang letaknya paling depan dalam sebuah perusahaan [1]. *Front office* juga memiliki bagian salah satunya yaitu resepsionis. Resepsionis merupakan bagian dari department *Front office* pada sebuah perusahaan yang bertugas untuk menerima dan menyambut tamu. Karena Resepsionis adalah orang pertama yang bertemu dengan tamu maka, seorang resepsionis harus mengetahui informasi-informasi dasar mengenai perusahaan baik itu layanan atau jasa yang ditawarkan [2].

Dalam sebuah organisasi/ perusahaan keberadaan resepsionis sangatlah penting karena, seorang resepsionis adalah orang pertama yang bertemu langsung dengan tamu, hal ini harus memberikan kesan pertama yang baik agar tamu mau menggunakan jasa yang ditawarkan oleh sebuah perusahaan, maka dari itu seorang resepsionis harus memiliki kemampuan berkomunikasi yang baik [3].

Dengan kemajuan teknologi digital saat ini menyebabkan penggunaan perangkat mobile meningkat sangat pesat dari kalangan muda sampai dewasa saat ini sudah hampir semua menggunakan perangkat mobile untuk menunjang kepentingan sehari-hari. Dengan tingginya permintaan konsumen terhadap perangkat mobile membuat perusahaan teknologi bersaing ketat untuk menarik perhatian konsumen, hal ini ditunjukkan dengan munculnya banyak perangkat mobile dengan berbagai macam brand dan berbagai macam sistem operasi. Di dunia terdapat 2 sistem operasi mobile yang mendominasi pasar teknologi mobile yaitu Android dan IOS. Android sendiri merupakan sistem operasi mobile yang dikembangkan oleh Google Inc sedangkan IOS merupakan sistem operasi mobile yang dikembangkan oleh Apple Inc. Menurut StatCounter pada periode september 2019 memperlihatkan bahwa pangsa pasar perangkat mobile dengan sistem operasi Android mengalahkan sistem operasi IOS dengan perbandingan yang cukup tinggi yaitu 76.24% untuk Android dan 22.48% untuk IOS [4]. Persentase ini menunjukkan

bahwa pengguna perangkat mobile dengan sistem operasi Android lebih banyak jika dibandingkan dengan perangkat mobile dengan sistem operasi IOS.

Perkembangan teknologi digital juga meningkatkan kualitas bisnis misalnya dulu mencatat data keuangan dalam bentuk berkas dokumen fisik, sekarang sudah bisa disimpan kedalam media digital bahkan bisa di simpan di cloud storage (media penyimpanan online). Hal tersebut tentu menghemat biaya, dan juga meminimalisir terjadinya kehilangan atau kerusakan data. Dikarenakan perkembangan teknologi yang begitu pesat, sedikit demi sedikit pekerjaan manusia bisa digantikan oleh sebuah sistem. Salah satu contohnya yaitu *Virtual Receptionist* yaitu resepsionis berbasis virtual/sistem, dimana pekerjaan seorang resepsionis dapat digantikan oleh sebuah Sistem. Hal ini tentu sangat membantu beberapa perusahaan yang belum/ tidak memiliki pegawai resepsionis, karena dengan adanya sistem *Virtual Receptionist* dinilai dapat memberikan layanan yang sama memuaskannya dengan pegawai resepsionis [5].

Selain mengetahui seluk-beluk perusahaan, seorang Resepsionis juga harus bisa berkomunikasi yang baik pada tamu, sehingga penyampaian informasi dan layanan dapat diberikan secara maksimal. Maka dari itu seorang Resepsionis diwajibkan memiliki kemampuan komunikasi yang baik guna memberikan impresi pertama yang baik terhadap tamu. Untuk memberikan pengalaman yang sama memuaskannya dengan pegawai Resepsionis, Virtual Resepsionis akan lebih baik jika menerapkan teknologi *speech/Voice Recognition*.

*Speech* atau *Voice Recognition* dikenal juga dengan sebutan Automatic Speech Recognition (ASR), Dimana ini merupakan sebuah proses yang dilakukan oleh sistem untuk mengkonversi sinyal ucapan menjadi serangkaian kata-kata begitupun sebaliknya menggunakan algoritma khusus yang diimplementasikan pada sebuah program komputer. *Voice Recognition* sangat berpotensi menjadi sebuah metode interaksi yang menjanjikan antara manusia dan komputer [6].

Salah satu metode yang dapat digunakan untuk *speech recognition* adalah *Hidden Markov Models*. *Hidden Markov Models* merupakan model statistik dimana suatu sistem yang dimodelkan diasumsikan sebagai markov proses dengan kondisi yang tidak terobservasi Suatu HMM dapat dianggap sebagai jaringan Bayesian

dinamis yang sederhana atau ‘simplest dynamic bayesian network’ [7]. *Hidden Markov Models* adalah metode atau pendekatan terbaik untuk *Voice Recognition* karena sangat fleksibel [8].

Berdasarkan latar belakang tersebut, maka penulis ingin melakukan Implementasi *Voice Recognition* Menggunakan Metode *Hidden Markov Models* Pada Pembangunan Sistem Resepsionis Berbasis Mobile yang terdiri dari dua aplikasi yaitu Receptionist dan Receptionist-compact.

## 1.2. Rumusan Masalah

Berdasarkan latar belakang yang sudah dipaparkan sebelumnya, maka dapat ditarik rumusan masalah yaitu.

1. Bagaimana mengimplementasi *Voice Recognition* pada pembangunan sistem resepsionis menggunakan metode *Hidden Markov Models* (HMM) berbasis mobile yang terdiri dari dua aplikasi yaitu Receptionist dan Receptionist-compact.
2. Mengukur tingkat akurasi *Voice Recognition* yang diimplementasi.

## 1.3. Batasan Masalah

Batasan-batasan yang harus diperhatikan pada penelitian ini yaitu:

1. Aplikasi dapat dijalankan dengan baik pada perangkat mobile dimana aplikasi Receptionist akan digunakan oleh tamu dan dijalankan disebuah Perangkat Tablet Android. Sedangkan aplikasi Receptionist-compact akan digunakan oleh karyawan internal perusahaan.
2. IDE yang digunakan untuk pembangunan aplikasi ini adalah Android Studio dengan bahasa pemrograman Kotlin.
3. Aplikasi dibangun menggunakan design pattern MVP (Model View Presenter) dan menggunakan Firebase sebagai Api services.
4. Aplikasi dapat memberikan tutorial guide pada tamu tentang penggunaan aplikasi Receptionist dengan mengimplementasi fungsi *Voice Recognition*.
5. Implementasi *Voice Recognition* akan menggunakan metode *Hidden Markov Models*.

#### 1.4. Tujuan Penelitian

Tujuan dalam penelitian ini yaitu.

1. Mengimplementasi *Voice Recognition* pada Sistem Receptionist menggunakan metode *Hidden Markov Models* (HMM) berbasis mobile yang terdiri dari dua aplikasi yaitu Receptionist dan Receptionist-compact.
2. Mengukur tingkat akurasi pengenalan dari *Voice Recognition* yang diimplementasi.

#### 1.5. Metode Penelitian

Dalam Penelitian ini dilakukan berbagai langkah metodologi antara lain sebagai berikut:

1. Studi Pustaka

Metode ini dilakukan dengan cara mempelajari literatur dari jurnal, skripsi, beserta website yang berkaitan untuk pengumpulan data, dan buku-referensi sebagai bahan dasar untuk memecahkan masalah.

2. Wawancara

Metode wawancara dilakukan untuk mengetahui spesifikasi atau kebutuhan dari perangkat lunak yang akan dibangun. Subjek dari metode wawancara ini adalah manager dari Team Mobile PT. Timedoor Indonesia. Dengan menggunakan metode wawancara, penulis akan semakin mudah mengetahui perangkat lunak seperti apa yang akan dibangun.

3. Analisis

Analisis kebutuhan perangkat lunak dilakukan setelah mengumpulkan keseluruhan kebutuhan berdasarkan hasil wawancara. Analisis kebutuhan perangkat lunak berupa fungsi produk, kebutuhan antarmuka. Selain itu, tahap ini juga membuat diagram use case dan mockup berupa gambaran tampilan sistem yang dibuat sesuai dengan alur pembuatan Sistem *Virtual Receptionist*.

4. Perancangan Sistem

Perancangan perangkat lunak yaitu berupa perancangan arsitektur, perancangan antarmuka. Perancangan dilakukan berdasarkan hasil dari



analisis yang dilakukan sebelumnya.

5. Pengkodean Sistem

Metode ini dilakukan dengan cara mengimplementasikan hasil rancangan kedalam sebuah program. Hasil dari metode ini berupa program.

6. Pengujian Sistem

Tahap pengujian dilakukan ketika aplikasi sudah selesai dibangun dan dilakukan pengujian fungsi–fungsi yang ada pada program. Tahap pengujian dilakukan untuk menemukan bug atau error dan pengecekan kelengkapan kebutuhan sistem oleh pihak PT Timedoor Indonesia.

### 1.6. Sistematika Penulisan

Dalam Penelitian ini terdapat sistematika penulisan yaitu sebagai berikut:

#### **PENDAHULUAN**

Bab ini akan membahas mengenai latar belakang, rumusan masalah, dan batasan masalah yang berkaitan dengan penelitian dan metode yang akan diterapkan dalam penilitan.

#### **TINJAUAN PUSTAKA**

Bab ini berisi penjelasan dan penjabaran beberapa studi yang telah berhasil dilakukan sebelum pembangunan aplikasi mobile untuk sistem resepsionis. Penulis telah melakukan pengamatan tentang pengertian resepsionis dari beberapa sumber, selain itu penulis juga telah melakukan pengamatan terhadap beberapa sistem resepsionis/front-office yang memiliki kesamaan dengan aplikasi yang akan dibangun oleh penulis.

#### **LANDASAN TEORI**

Bab ini akan memuat penjelasan dan penjabaran dasar teori yang akan digunakan untuk analisis dan perancangan perangkat lunak dalam pembangunan aplikasi mobile untuk sistem resepsionis.

#### **ANALISIS DAN PERANCANGAN SISTEM**

Bab ini akan membahas mengenai kebutuhan sistem yang akan dibuat.

#### **IMPLEMENTASI DAN PENGUJIAN SISTEM**

Bab ini akan membahas mengenai pengimplementasian dan pengujian sistem

yang akan dibuat.

## **PENUTUP**

Bab ini berisi kesimpulan dan saran dari sistem yang telah dibuat oleh penulis.



## BAB II. TINJAUAN PUSTAKA

Resepsionis merupakan bagian dari department *Front office* pada sebuah perusahaan yang bertugas untuk menerima dan menyambut tamu. Karena Resepsionis adalah orang pertama yang bertemu dengan tamu maka, seorang resepsionis harus mengetahui informasi-informasi dasar mengenai perusahaan baik itu layanan atau jasa yang ditawarkan [2].

Besarnya biaya tambahan yang harus dikeluarkan perusahaan jika harus memperkerjakan seorang resepsionis, membuat beberapa perusahaan lebih memilih menggunakan *Virtual Receptionist*. *Virtual Receptionist* atau resepsionis digital merupakan sebuah sistem yang bertugas untuk menggantikan posisi seorang resepsionis menjadi sebuah sistem yang dapat diandalkan. Sebuah *Virtual Receptionist* ditampilkan pada sebuah layar monitor yang memungkinkan tamu untuk memilih layanan yang dibutuhkan olehnya [5]. Guna memberikan pengalaman yang lebih nyata pada pengguna, sebuah sistem virtual resepsionis akan lebih optimal jika mengimplementasikan fungsi *Voice Recognition*.

*Voice Recognition* atau *Speech Recognition* merupakan sebuah fitur yang sering ditemui di perangkat komputer yang dapat mengubah signal analog berupa suara menjadi kode digital yang dapat digunakan untuk memberi perintah pada sistem agar melakukan sesuatu yang diinginkan [9].

Sistem *Virtual Receptionist* bukanlah sepenuhnya suatu hal yang baru. Adapun beberapa sistem resepsionis atau *front office* yang telah berhasil dibuat oleh penulis lain, yaitu sebagai berikut.

Dalam rancang sistem informasi *front office* hotel minang permai pacitan, membahas tentang pembangunan sebuah aplikasi desktop untuk membantu *front office*/resepsionis dari Hotel Minang Permai Pacitan dalam mengelola data dibagian *front office*. Sistem ini dibangun dengan tools *Microsoft Visual Basic 6.0* dan menggunakan bahasa pemrograman *Visual Basic 6.0*. Sistem ini dirancang untuk bisa mengelola data konsumen lebih mudah dan cepat, dan dapat memberikan informasi yang akurat dan cepat kepada konsumen sehingga kinerja resepsionis menjadi lebih efektif dan efisien. Target dari sistem ini adalah resepsionis/*front*

*office* dari Hotel Minang Permai Pacitan. Keunggulan dari sistem ini yaitu menikmati produktifitas dan efektifitas dari resepsionis karena mempermudah pengelolaan data konsumen. Kekurangan sistem ini adalah tidak fleksibel karena harus dijalankan disebuah komputer desktop [10].

Dalam Sistem Informasi Reservasi Kamar Hotel Pada Hotel Posters MICE, membahas tentang sebuah aplikasi desktop untuk mengelola data pemesanan dari pelanggan. Sistem ini dibangun menggunakan tools *Visual Basic NET 2010* dan bahasa pemrograman *Visual Basic* dan menggunakan metode *waterfall*. Sistem ini bertujuan untuk mempercepat proses pendaftaran saat dibagain resepsionis yang awalnya masih dilakukan secara manual. Penerapan aplikasi ini dapat mempermudah kinerja pegawai dalam mengelola data pelanggan serta informasi yang berguna untuk menunjang bisnis hotel. Target dari sistem ini adalah resepsionis dari Hotel Poster MICE. Keunggulan sistem ini yaitu dengan adanya sistem ini pengelolaan data di bagian *Front office* tidak perlu lagi secara manual, hal ini sangat meningkatkan kinerja pegawai dengan signifikan. Kekurangan dari sistem ini yaitu menggunakan metode *waterfall*, dimana spesifikasi dari pembangunan sistem ini harus diketahui dari awal, jika ada perubahan ditengah-tengah akan sulit untuk di modifikasi [11].

Dalam Sistem Layanan Hotel Dan Otomatisasi Sistem Reservasi Hotel Dengan Teknologi Radio Frequency Identification, sistem ini membahas tentang automisasi sistem reservasi pada sebuah hotel. Sistem ini dibangun menggunakan berbagai macam modul seperti *microcontroller*, *RFID reader*, modul pengontrol daya listrik, dan lain sebagainya. Pembangunan sistem ini diharapkan dapat menggantikan posisi seorang resepsionis dalam melayani tamu yang ingin menginap di hotel. Kelebihan sistem ini yaitu data yang diberikan sangat akurat dan dapat diandalkan, otomatisasi ini juga mempermudah pihak menejemen hotel dalam mengelola bisnis hotelnya. Kekurangan sistem ini yaitu biaya yang dikeluarkan untuk membangun sistem ini tergolong mahal karena harus membeli alat / tools khusus [12].

Berdasarkan penelitian yang telah dilakukan sebelumnya, maka penulis membangun aplikasi mobile untuk sistem resepsionis yang terdiri dari 2(dua)

aplikasi yaitu Receptionist dan Receptionist-compact. Aplikasi Receptionist akan berjalan pada Tablet dengan Sistem Operasi Android, sedangkan untuk Receptionist-compact akan dijalankan pada sebuah smartphone dengan sistem operasi android.

Tabel 2.1 menunjukkan perbandingan aplikasi yang akan dibangun oleh penulis dengan aplikasi/sistem yang telah berhasil dikembangkan oleh penulis lain.

Tabel 2. 1 Perbandingan Penelitian

<b>Sumber</b>	L. Markito and L. Yulianto [10]	Ari Prabowo dan Endang Retnoningsih [11]	Vania Angelina dan Hadian Satria Utama [12]	Sumariana (2019)*
<b>Judul</b>	Rancang Bangun Sistem Informasi <i>Front office</i> Hotel Minang Permai Pacitan	Sistem Informasi Reservasi Kamar Hotel Pada Hotel Posters MICE	Sistem Layanan Hotel Dan Otomatisasi Sistem Reservasi Hotel Dengan Teknologi Radio Frequency Identification	Implementasi <i>Voice Recognition</i> Menggunakan Metode <i>Hidden Markov Models</i> Pada Pembangunan Sistem Resepsionis Berbasis Mobile.
<b>Tools</b>	Microsoft Visual Basic 6.0	Visual Basic NET 2010	HyperTerminal	Android Studio
<b>Platform</b>	Desktop	Desktop	Desktop	Android
<b>Bahasa Pemrogramman</b>	Visual Basic 6.0	Visual Basic 6.0	-	Kotlin

<b>Sasaran Pengguna</b>	Resepsionis dari Hotel Minang Permai Pacitan	Resepsionis dari Hotel Poster MICE	Pengunjung Hotel	Untuk Client/tamu dan pegawai internal dari PT. Timedoor Indonesia
-------------------------	--	------------------------------------	------------------	--

\*) Penelitian yang dilakukan.



## **BAB III. LANDASAN TEORI**

Adapun beberapa penjelasan dan penjabaran dasar teori yang berkaitan dengan sistem yang akan dibuat dalam penelitian ini.

### **3.1. Resepsionis / *Front office***

Resepsionis merupakan bagian dari department *Front office* pada sebuah perusahaan yang bertugas untuk menerima dan menyambut tamu, selain itu resepsionis juga berfungsi sebagai penyedia layanan administrasi guna menunjang kepuasan tamu. Resepsionis harus mengetahui informasi-informasi dasar mengenai perusahaan baik itu layanan atau jasa yang ditawarkan [2].

Maka secara garis besar resepsionis adalah orang pertama yang berhadapan langsung dengan tamu, dan resepsionis harus dapat memberikan informasi yang sesuai diinginkan oleh tamu [1].

### **3.2. Android Studio**

Android Studio adalah Integrated Development Environment (IDE) khusus yang digunakan untuk membangun aplikasi mobile berbasis android dan Android Studio merupakan IDE resmi yang dibuat oleh Google inc berdasarkan IntelliJ IDEA untuk memfasilitasi developer dalam pengembangan aplikasi android [13]. Android Studio sangatlah fleksibel, yakni bisa diinstall di segala platform seperti Mac, Windows, Linux, bahkan di sistem operasi yang tergolong sederhana seperti Chrome OS. Hal ini tentu menjadi kelebihan Android Studio jika dibandingkan dengan XCode yang merupakan IDE khusus untuk membangun aplikasi berbasis IOS, hal ini dikarenakan XCode hanya bisa digunakan di environment MacOS beda halnya dengan Android Studio yang bisa digunakan di segala platform [14]. Dalam proses pembangunan aplikasi android, Android Studio mendukung berbagai bahasa pemrograman seperti Kotlin, Java, dan C++ [15].

### **3.3. Kotlin**

Kotlin adalah salah satu bahasa pemrograman statically - typed programming language yang mendukung pemrograman berorientasi objek dan

pemrograman fungsional. Kotlin memiliki syntax dan konsep yang sama dengan bahasa lain termasuk C#, Java, dan Scala. Kotlin dikembangkan oleh Kotlin Foundation, sebuah group yang dibentuk oleh JetBrains dan Google. Google secara resmi mengumumkan bahwa pembangunan aplikasi android mendukung bahasa pemrograman kotlin, maka dari itu dokumentasi untuk pembuatan aplikasi android akan tersedia untuk bahasa pemrograman kotlin. Android API tertentu seperti Android KTX dibuat spesifik untuk kotlin, namun ditulis menggunakan bahasa java, maka dari itu dapat digunakan untuk java maupun kotlin. Interoperabilitas kotlin dengan java merupakan sebuah kesinambungan ini berarti developer dapat memanggil kode dari java ke kotlin begitu juga sebaliknya [16].

### **3.4. Voice Recognition**

*Voice Recognition* merupakan sebuah perangkat lunak atau perangkat keras dari komputer yang memiliki kemampuan untuk mengekstrak atau meng-decode suara manusia. *Voice Recognition* biasanya berguna untuk mengoperasikan sebuah perangkat, baik itu melakukan suatu tindakan atau menulis pada software pengolah kata tanpa harus menggunakan keyboard, mouse ataupun menekan tombol apapun [17].

Dalam Sistem *Voice Recognition* sinyal suara atau soundwave akan di transformasi menjadi sekumpulan vektor yang tersusun menggunakan metode speech processing khusus. Dimana metode tersebut akan mengubah deretan vektor tersebut kedalam bentuk *phoneme*. Phoneme adalah sebuah unit sound yang membedakan antara satu kata dengan kata lainnya, sebagai contoh kata “thumb” dan kata “dumb” adalah dua kata yang berbeda yang bisa dibedakan dengan cara mensubstitusi satu phoneme yaitu “th” menjadi “d”, maka phoneme sangat mempengaruhi arti dari sebuah kata karena setiap orang memiliki cara pengucapan yang berbeda dan aksen yang berbeda [6].

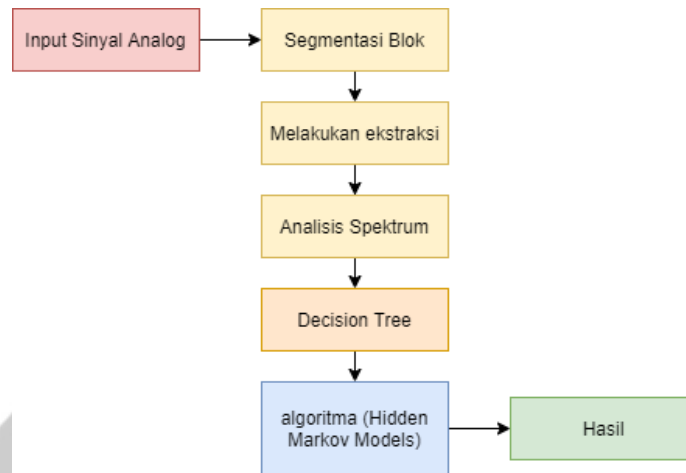


Proses dari Sistem *Voice Recognition* dibedakan menjadi beberapa langkah yaitu sebagai berikut [6].

1. **Step 1** yaitu sinyal analog akan disegmentasi menjadi blok yang diberi spasi guna mendapatkan karakteristik dari sinyal, seperti total energi, *Zero Crossing Strength* diberbagai rentang frekuensi dan lain sebagainya. Dengan karakteristik yang didapat, deretan vektor akan digabungkan pada setiap blok dengan *phoneme* untuk menghasilkan sebuah *string phoneme*.
2. **Step 2** pada langkah ini analisis spektrum diterapkan pada setiap blok dengan menggunakan teknik pengkodean prediktif linier, Fast Fourier Transform (FFT) dan Bank of Frequency Filter.
3. **Step 3** pada langkah ini proses pengambilan keputusan dijalankan disetiap blok. Setiap Phoneme memiliki perbedaan yang akan mempersempit hasil dari pengenalan.
4. **Step 4** langkah ini akan digunakan untuk meningkatkan kinerja dari proses pengambilan keputusan untuk menghasilkan akurasi yang tertinggi dengan menggunakan algoritma khusus. Untuk setiap kosakata, sebuah algoritma akan dibangun dan kemudian *string phoneme* akan dibandingkan satu sama lain. Salah satu metode untuk membandingkan phoneme ini adalah *Hidden Markov Models*.

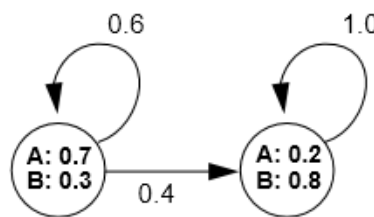
### 3.5. Hidden Markov Models

Dalam proses *Voice Recognition* terdapat 4 langkah penerjemahan dari sinyal analog ke sinyal digital. Ketika proses penerjemahan masuk ke tahap 4 metode khusus akan digunakan untuk meningkatkan hasil output dari penerjemahan sehingga mendapatkan hasil dengan akurasi tertinggi. Salah satu metode algoritma yang bisa digunakan untuk membandingkan serangkaian *phoneme* adalah *Hidden Markov Models*.



Gambar 3. 1 Langkah Proses *Voice Recognition*

Seperti yang terlihat pada gambar 3.1 dimana metode *Hidden Markov Models* digunakan untuk menyempurnakan pengambilan keputusan pada langkah ketiga proses *Voice Recognition*. *Hidden Markov Models* adalah sebuah model statistika probabilitas berupa rantai keadaan yang dihubungkan dengan transisi. Setiap keadaan dapat terlihat, dimana satu satunya parameter yang perlu diperhatikan adalah kemungkinan dari transisi antar kondisi [18].



Gambar 3. 2 *Hidden Markov Models* dengan 2 keadaan dan 2 simbol output yaitu A dan B [8]

Proses dimulai saat keadaan awal (*initial state*), di setiap langkah waktu diskrit, transisi dipindahkan ke keadaan baru dan kemudian satu simbol output di generate pada keadaan itu. Hasil pemilihan transisi dan output adalah acak dan ditentukan oleh distribusi probabilitas. *Hidden Markov Models* juga bisa disebut *black box method* atau metode kotak hitam karena urutan dari simbol output yang

dihasilkan dapat diamati, namun urutan keadaan yang dikunjungi oleh transisi disembunyikan. Maka dari itu Disebut *Hidden Markov Models* [8].

Dalam Sistem *Voice Recognition*, *Hidden Markov Models* suatu keadaan akan di interpretasikan sebagai sebuah acoustic models, yang mengindikasi suara apa yang sepertinya terdengar saat segmentasi *Voice Recognition*, dimana *Hidden Markov Models* akan melakukan pengecekan probabilitas untuk menempatkan suatu *phoneme* dan diikuti oleh *phoneme* berikutnya lalu mnenentukan apakah frase tersebut benar atau memungkinkan [8].

### **3.6.Firebase**

Firestore adalah sebuah platform pengembangan aplikasi mobile dan website yang menyediakan berbagai macam servis yang memudahkan developer dalam membangun aplikasinya sehingga menciptakan produk yang berkualitas. Pada tahun 2011, sebelum firestore diberi nama firestore, dulu bernama Envolv. Dimana Envolv memberikan layanan kepada developer sebuah API yang memungkinkan integrasi fungsi online chat atau live chat pada sebuah website. Namun para developer saat itu menggunakan Envolv untuk mengirimkan data aplikasi yang lebih besar dari chat message. Hal ini menyebabkan founder dari envolve memisahkan layanan chat system dengan arsitektur real time. April 2012, Firestore terbentuk atas 2 perusahaan terpisah yang menyediakan layanan Backend-as-a-Service dengan fungsionalitas real-time [19].

Firestore merupakan Backend-as-a-Service, dengan firestore developer tidak perlu melakukan server-side programming, dan programmer dapat dengan mudah mengintegrasikan aplikasi mobile/website hanya dengan Firestore SDK. Setiap aplikasi mobile/website pasti memiliki backend services yang di design untuk mendukung front-end (User Interface). Dikarenakan tidak memerlukan pemrograman server-side, Backend-as-a-Service adalah sebuah metode yang dapat digunakan untuk menghubungkan aplikasi mobile/website ke Cloud-based-Service dengan mudah [20].

### **3.7. Firebase Realtime Database**

Google Firebase juga menyediakan 2(dua) layanan database, salah satunya Firebase Realtime Database. Realtime database adalah sebuah sistem database yang memiliki "real-time constraint" atau keterikatan terhadap komputasi dengan waktu yang bersamaan. Ini berarti database harus dapat mengontrol suatu situasi untuk menerima data, memprosesnya, lalu mengembalikan data secara cepat dan mempengaruhi situasi saat itu juga. Realtime database berbeda dengan database tradisional karena memiliki keterikatan waktu yang sangat singkat yaitu microsecond ataupun nanosecond, hal ini tentu sangat penting untuk melakukan komputasi data yang cepat dan untuk mengurangi kesalahan dalam transaksi data. Firebase Realtime Database menggunakan konsep Realtime Database maka dari itu Firebase Realtime Database memiliki keunggulan jika dibandingkan dengan database tradisional [21].

Berbeda dengan database pada umumnya yang menggunakan SQL, Firebase Realtime Database menggunakan Cloud-hosted NoSQL database. Data pada Firebase Realtime Database disimpan dalam sebuah JSON tree dengan nodes yang mempermudah developer untuk menyimpan data dan mensinkronisasinya dengan semua user diwaktu yang bersamaan [20].

### **3.8. Firebase Cloud Messaging**

Firebase Cloud Messaging merupakan layanan messaging gratis dan cross-platform yang memungkinkan developer untuk mengirim push notification ke audiens tanpa harus khawatir dengan server-side programming. Selain itu developer juga dapat menggunakan Firebase Cloud Messaging untuk notifikasi upstream, dimana Firebase Cloud Messaging menerima pesan dari aplikasi client, atau menerima pesan untuk aplikasi ketika ada data baru yang siap untuk diunduh. Hal ini untuk memastikan komunikasi antara aplikasi server dan client hanya terjadi jika diperlukan.

Firebase Cloud Messaging mendukung tipe pesan yaitu notification messages dan data messages. Notification message adalah ketika aplikasi client berperilaku sesuai dengan keadaan apakah aplikasi berjalan di background atau tidak ketika

menerima push notifikasi dari Firebase Cloud Messaging. Jika aplikasi berjalan di background maka firebase SDK akan secara otomatis memproses pesan dan menampilkannya sebagai notifikasi di panel notifikasi device. Untuk Data Message, developer bisa mengirim custom data untuk aplikasi client [22].

### **3.9. MVP (Model View Presenter)**

Dalam membangun sebuah aplikasi atau website terdapat banyak design pattern mulai dari MVC (Model View Controller), MVVM(Model View ViewModel), dan MVP(Model View Presenter). Namun pada pembangunan aplikasi mobile untuk sistem resepsionis ini akan dibangun menggunakan design pattern MVP. Design pattern MVP(Model View Presenter) memiliki prinsip memisahkan logic code dengan ui code dimana Presenter sebagai perantara antara Model dan View, maka dari itu presenter ini hanya merupakan sebuah contract yang mengikat View dan Model sehingga Model dan View hanya berkomunikasi satu sama lain melalui presenter [22].

Dikarenakan oleh terpisahnya antara logic code dan ui code membuat struktur aplikasi menjadi lebih rapi. Bentuk dari struktur file menjadi modular, sehingga ketika terjadinya perbaikan atau penambahan fitur lebih mudah dilakukan [23].

## BAB VI. PENUTUP

### 6.1. Kesimpulan

Berdasarkan hasil dari perancangan, pembahasan, dan pengujian Sistem *Virtual Receptionist* maka dapat disimpulkan beberapa hal antara lain:

1. Penulis telah berhasil mengimplementasi *Voice Recognition* menggunakan metode *Hidden Markov Models* pada pembangunan Sistem *Virtual Receptionist* dan berhasil membantu perusahaan PT. Timedoor Indonesia dalam menggantikan posisi seorang pegawai Receptionist untuk menjadi jembatan antara tamu yang datang dengan pegawai yang ingin ditemui.
2. Implementasi *Voice Recognition* menggunakan metode *Hidden Markov Models* menghasilkan akurasi dengan skor WER yaitu 13.4132% atau dengan akurasi sebesar 86.5856%.

### 6.2.Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran agar sistem ini dapat lebih baik, diantaranya Integrasi Artificial Intelligence pada fitur *Voice Recognition* dimana saat ini kemampuan voice assistant masih terbatas karena tidak bisa meningkatkan kemampuan dengan sendirinya.

## DAFTAR PUSTAKA

- [1] D. R. Maiziva, “PERANAN RECEPTIONIST PADA *FRONT OFFICE* DEPARTEMENT HOTEL GRAND ZURI PEKANBARU,” vol. 4, no. 2, pp. 1–14, 2017.
- [2] F. Saputro, E. Devi, and H. Putri, “Aktivitas Receptionist Dalam Menangani Tamu Personal Di AMEERA BOUTIQUE HOTEL YOGYAKARTA,” *Khasanah Ilmu*, vol. V, no. 2, pp. 13–24, 2014.
- [3] S. Deering, “Are Receptionists the Most Important Person in the Office?,” *theundercoverrecruiter.com*, 2015. [Online]. Available: <https://theundercoverrecruiter.com/are-receptionists-the-most-important-person-in-the-office/>. [Accessed: 20-Jan-2020].
- [4] “Mobile Operating System Market Share Worldwide,” 2019. [Online]. Available: <https://gs.statcounter.com/os-market-share/mobile/worldwide>. [Accessed: 08-Oct-2019].
- [5] F. Bazzano and F. Lamberti, “Human-robot interfaces for interactive receptionist systems and wayfinding applications,” *Robotics*, vol. 7, no. 3, 2018.
- [6] S. Naziya S. and R. R. Deshmukh, “Speech Recognition System – A Review,” *IOSR J. Comput. Eng.*, vol. 18, no. 04, pp. 01–09, 2016.
- [7] Q. Nada, C. Ridhuandi, P. Santoso, and D. Apriyanto, “Speech Recognition dengan Hidden Markov Model untuk Pengenalan dan Pelafalan Huruf Hijaiyah,” *J. Al-AZHAR Indones. SERI SAINS DAN Teknol.*, vol. 5, no. 1, p. 19, 2019.
- [8] J. Tebelskis, “Speech Recognition Using Neural Networks,” vol. CMU-CS-95-, p. 173, 1995.
- [9] A. W. Dani, A. Adriansyah, and D. Hermawan, “Perancangan Aplikasi Voice Command Recognition Berbasis Android Dan Arduino Uno,” *J. Teknol. Elektro*, vol. 7, no. 1, pp. 11–19, 2016.
- [10] L. Markito and L. Yulianto, “Rancang Bangun Sistem Informasi *Front office*

- Hotel Minang Permai Pacitan,” vol. 5700, no. November, pp. 1–6, 2013.
- [11] A. Prabowo *et al.*, “Sistem Informasi Reservasi Kamar Hotel Pada Hotel Posters MICE,” vol. 1, no. 2, pp. 147–160, 2017.
- [12] V. Angelina, “Sistem Layanan Hotel Dan Otomatisasi Sistem Reservasi Hotel Dengan Teknologi Radio Frequency Identification,” vol. 16, no. 2, pp. 149–164, 2014.
- [13] “Meet Android Studio : Android Developers,” 2019. [Online]. Available: <https://developer.android.com/studio/intro>. [Accessed: 09-Oct-2019].
- [14] Google, “Download Android Studio and SDK Tools,” 2019. [Online]. Available: <https://developer.android.com/studio#downloads>. [Accessed: 09-Oct-2019].
- [15] A. Sinicki, “I want to develop Android Apps — What languages should I learn?,” 2019. [Online]. Available: <https://www.androidauthority.com/develop-android-apps-languages-learn-391008/>. [Accessed: 09-Oct-2019].
- [16] A. Developers, “Kotlin overview,” *developer.android.com*, 2019. [Online]. Available: <https://developer.android.com/kotlin/overview>. [Accessed: 19-Jan-2020].
- [17] Computerhope, “Voice recognition,” *computerhope.com*, 2019. [Online]. Available: <https://www.computerhope.com/jargon/v/voicreco.htm>. [Accessed: 21-Jan-2020].
- [18] M. E. B. Prasetyo, “Teori Dasar Hidden Markov Model,” *Makal. Probab. Stat. STEI*, pp. 1–5, 2011.
- [19] GeekyAnts, “Introduction to Firebase,” *Hackernoon.com*, 2017. [Online]. Available: <https://hackernoon.com/introduction-to-firebase-218a23186cd7>. [Accessed: 19-Jan-2020].
- [20] S. Chaudhary, “What Is Firebase? Firebase Benefits and Usage,” *bytegenius.in*, 2020. [Online]. Available: <https://www.bytegenius.in/blog/what-is-firebase-its-benefits-and-usage/>. [Accessed: 19-Jan-2020].
- [21] B. Nicolas and B. Lucie, “Real-Time database : Firebase INFO-H-415 :



Advanced database,” pp. 1–23, 2017.

- [22] J. Thornsby, “Sending Android push notifications with Firebase Cloud Messaging,” *androidauthority.com*, 2018. [Online]. Available: <https://www.androidauthority.com/android-push-notifications-with-firebase-cloud-messaging-925075/>. [Accessed: 19-Jan-2020].
- [23] J. Kim, “Getting Started with MVP (Model View Presenter) on Android,” *raywenderlich.com*, 2018. [Online]. Available: <https://www.raywenderlich.com/7026-getting-started-with-mvp-model-view-presenter-on-android#toc-anchor-004>. [Accessed: 19-Jan-2020].

