

**PERANCANGAN SISTEM INFORMASI PELAYANAN TAMU
PADA DEPARTMENT CONCIERGE HOTEL CENTURY
PARK BERBASIS ANDROID**

SKRIPSI

PROGRAM STUDI SISTEM INFORMASI



OLEH :

NAMA : LUKMAN AMIM

NIM : 011605503125040

**FAKULTAS TEKNIK
UNIVERSITAS SATYA NEGARA INDONESIA
JAKARTA
2020**

**DESIGN OF INFORMATION SYSTEM
GUEST SERVICE IN THE DEPARTMENT CONCIERGE
ON CENTURY PARK HOTEL BASED ANDROID**

ESSAY

STUDY PROGRAM INFORMATION SYSTEM



NAME : LUKMAN AMIM

ID : 011605503125040

**FACULTY OF ENGINEERING
UNIVERSITY OF SATYA NEGARA INDONESIA
JAKARTA
2020**

**PERANCANGAN SISTEM INFORMASI PELAYANAN TAMU
PADA DEPARTMENT CONCIERGE HOTEL CENTURY
PARK BERBASIS ANDROID**

SKRIPSI

Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar

SARJANA KOMPUTER

Program Studi Sistem Informasi



OLEH :

NAMA : LUKMAN AMIM

NIM : 011605503125040

**FAKULTAS TEKNIK
UNIVERSITAS SATYA NEGARA INDONESIA
JAKARTA
2020**

SURAT PERNYATAAN KARYA SENDIRI

Nama : Lukman Amim

Nim : 011605503125040

Program Studi : Sistem Informasi

Menyatakan bahwa Skripsi/ Tugas Akhir ini adalah murni hasil karya sendiri dan seluruh isi Skripsi/ Tugas Akhir menjadi tanggung jawab saya sendiri. Apabila saya mengutip dari karya orang lain maka saya mencantumkan sumbernya sesuai dengan ketentuan yang berlaku. Saya bersedia dikenakan sanksi pembatalan Skripsi/ Tugas Akhir ini apabila terbukti melakukan tindakan plagiat (penjiplakan).

Demikian pernyataan ini saya buat dengan sebenarnya.

Jakarta, 17 Februari 2020
Lukman Amim
011605503125040
DJP

LEMBAR PENGESAHAN SKRIPSI

Nama : Lukman Amim
NIM : 011605503125040
Jurusan : Sistem Informasi
Judul Skripsi : "Perancangan Sistem Informasi Pelayanan Tamu Pada
Department Concierge Hotel Century Park Berbasis
Android"

Tanggal Ujian : 17 Februari 2020

Jakarta, 17 Februari 2020



Dosen Pembimbing I

Dosen Pembimbing II

(Kiki Kusumawati, ST., MMSI.)

(Priongo Hendradi, S.Kom., MMSI.)

Dekan

Ketua Program Studi



(Ir. Nuchayati, M.Si.)



(Agung Priambodo, S.Kom., M.Kom.)

LEMBAR PENGESAHAN PENGUJI

PERANCANGAN SISTEM INFORMASI PELAYANAN TAMU PADA
DEPARTMENT CONCIERGE HOTEL CENTURY PARK BERBASIS
ANDROID



Anggota Penguji I

(Istiqomah Sumadikarta, ST., M.Kom.)

Anggota Penguji II

(Riama Sibarani, S.Si., MMSI.)

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT atas berkat dan hikmat yang diberikan, penulis dapat menyusun skripsi dengan judul *“Perancangan Sistem Informasi Pelayanan Tamu pada Department Concierge di Hotel Century Park Berbasis Android”*, sebagai salah satu syarat yang harus dipenuhi dalam memperoleh gelar sarjana komputer.

Penulis sangat menyadari bahwa tanpa bantuan dari banyak pihak, mungkin penulis akan mendapatkan banyak kesulitan dan hambatan dalam menyusun tugas ini.

Penulis menyadari bahwa tugas kuliah ini masih jauh dari sempurna baik bentuk maupun isinya. Penulis sangat mengharapkan saran dan kritik guna penyempurnaannya.

Pada kesempatan ini, tidak lupa penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan bimbingan, pengarahan, nasihat dan pemikiran dalam menyelesaikan skripsi ini, terutama kepada :

1. Kepada Allah SWT yang telah memberi saya nikmat kesehatan dan kekuatan jasmani maupun rohani selama saya menyelesaikan skripsi ini.
2. Ibu Ir. Nurhayati, M.Si selaku Dekan Fakultas Teknik Universitas Satya Negara Indonesia.
3. Ibu Kiki Kusumawati, ST.,MMSI., selaku dosen pembimbing I yang memberikan masukan dalam penyusunan skripsi ini.

4. Bpk Prionggo Hendradi, S.Kom.,MMSI., selaku dosen pembimbing II yang memberikan masukan, serta inputan selama bimbingan dalam penulisan skripsi.
5. Bpk Agung Priambodo S.Kom., M.Kom., selaku Ketua Program Studi Sistem Informasi.
6. Karyawan Hotel Century Park khususnya pada Department Front Office
7. Kepada Kedua Orang tua kami Bpk Nasorin dan Ibu Suriyah tercinta yang selalu mendoakan dan memberi dukungan untuk kelancaran penulisan ini
8. Istri tersayang yang selalu mendoakan dan memberikan motivasi serta dorongan dalam menyelesaikan skripsi ini.
9. Teman-teman di Program studi Sistem Informasi yang telah banyak membantu dan saling memberi semangat.

Akhir kata penulis saya berharap semoga skripsi ini dapat bermanfaat bagi para pembaca dan pihak yang berkepentingan.

Jakarta, 17 Februari 2020

Penulis

ABSTRAK

Kurangnya tingkat disiplin *staff* bellboy dalam menjalankan prosedur kerja dan *standart operation procedure* (SOP) yang masih manual mengakibatkan belum adanya pengarsipan data barang titipan maupun data *customer* yang terkomputerisasi di department concierge Hotel Century Park, serta belum adanya sistem informasi pelayanan tamu yang berfungsi memudahkan *staff* concierge dalam menjalankan tugasnya. Tujuan dari penelitian ini adalah merancang sistem informasi pelayanan tamu pada department concierge. Dengan melakukan wawancara langsung kepada Chief concierge Hotel Century Park untuk mendapatkan data yang dibutuhkan, diharapkan dari aplikasi yang dibuat dapat membantu *team* concierge dalam melayani *customer* dan mempermudah mencari data barang titipan maupun *customer* ketika diperlukan.

Kata kunci : Pelayanan Tamu, Concierge, Sistem Pelayanan Bellboy, Android.

ABSTRACT

The lack of bellboy staff discipline in carrying out work procedures and standard operation procedures (SOP) that are still manual results in the absence of archiving of safekeeping data and computerized customer data at Century Park Hotel concierge departments, as well as the absence of a guest service information system that functions to facilitate concierge staff in carrying out their duties. The purpose of this study is to design a guest service information system at the concierge department. By conducting direct interviews with the Chief Concierge of Century Park Hotel to get the required data, it is expected that the application created can help the concierge team in serving customers and making it easier to find the item of safekeeping and customer data when needed.

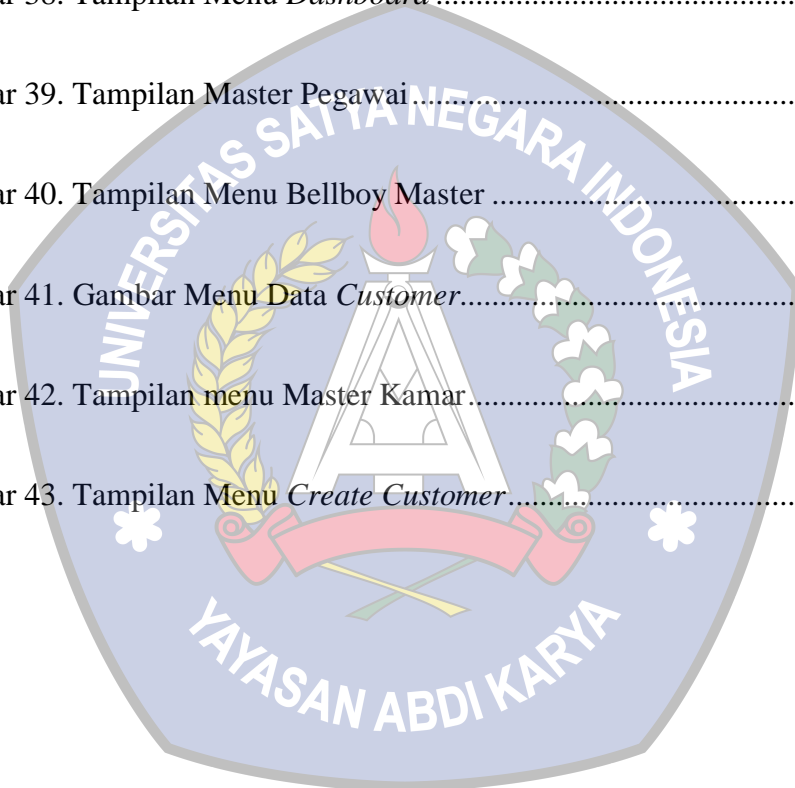
Keywords : Guest Services, Concierge, Bellboy Service System, Android.

DAFTAR GAMBAR

Gambar 1. File Proyek di tampilan Android	16
Gambar 2. File Proyek dalam Tampilan Problem.....	17
Gambar 3. Jendela Utama Android Studio	17
Gambar 4. Logo Hotel Century Park	25
Gambar 5. Struktur Organisasi Department FO.....	28
Gambar 6. Kerangka Berpikir.....	34
Gambar 7. <i>Flow Map Check in</i>	35
Gambar 8. <i>Flow Map Check out</i>	37
Gambar 9. <i>Flow Map Usulan Check in</i>	39
Gambar 10. <i>Flow Map Usulan Check out</i>	41
Gambar 11. <i>Use Case Diagram</i>	43
Gambar 12. <i>Activity Diagram Register Pegawai</i>	44
Gambar 13. <i>Activity Diagram Log in</i>	45
Gambar 14. <i>Activity Diagram Bellcaptain Input</i>	46
Gambar 15. <i>Activity Diagram Bellboy Kelola Barang</i>	47
Gambar 16. <i>Activity Diagram Customer Input Komentar</i>	48

Gambar 17. <i>Sequence Diagram Register</i>	49
Gambar 18. <i>Sequence Diagram Log in</i>	50
Gambar 19. <i>Sequence Diagram List Barang</i>	51
Gambar 20. <i>Sequence Diagram Input Komentar</i>	52
Gambar 21. <i>Rancangan Register</i>	58
Gambar 22. <i>Rancangan Log in</i>	59
Gambar 23. <i>Rancangan List Input Barang</i>	60
Gambar 24. <i>Rancangan List Bellboy</i>	61
Gambar 25. <i>Rancangan List Check out</i>	62
Gambar 26. <i>Rancangan List Inbox</i>	63
Gambar 27. <i>Rancangan View Detail Barang</i>	64
Gambar 28. <i>Rancangan Halaman Komentar</i>	65
Gambar 29. <i>Rancangan Pilihan Layanan</i>	66
Gambar 30. <i>Rancangan edit detail barang</i>	69
Gambar 31. <i>Rancangan halaman Log in Web</i>	70
Gambar 32. <i>Rancangan halaman Dashboard</i>	71
Gambar 33. <i>Rancangan halaman Master Pegawai</i>	72

Gambar 34. Rancangan tampilan halaman Master Kamar.....	72
Gambar 35. Rancangan Halaman Bellboy Master	73
Gambar 36. Rancangan Transaksi Bellboy	73
Gambar 37. Tampilan Menu Log in.....	74
Gambar 38. Tampilan Menu <i>Dashboard</i>	75
Gambar 39. Tampilan Master Pegawai.....	75
Gambar 40. Tampilan Menu Bellboy Master	76
Gambar 41. Gambar Menu Data <i>Customer</i>	76
Gambar 42. Tampilan menu Master Kamar.....	77
Gambar 43. Tampilan Menu <i>Create Customer</i>	78

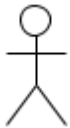


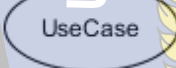


DAFTAR TABEL

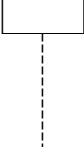
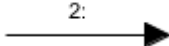
Tabel 1. Jadwal Penelitian (<i>Gantt Chart</i>)	26
Tabel 2. Analisa Kebutuhan Sistem	35
Tabel 3. Desain Tabel Master Kamar	56
Tabel 4. Desain Tabel Master Pegawai	56
Tabel 5. Desain Tabel Master Jabatan	57
Tabel 6. Desain Tabel Customer	57
Tabel 7. Desain Tabel Master Barang	58
Tabel 8. Desain Tabel Transaksi Bellboy	58
Tabel 9. Desain Tabel Transaksi Tamu	59
Tabel 10. <i>Software Testing</i>	83

DAFTAR SIMBOL

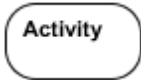



Simbol *Use Case Diagram*

Simbol <i>Use Case Diagram</i>			
No.	Gambar	Nama	Keterangan
1.		<i>Actor</i>	Peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i>
3.		<i>Generalization</i>	Hubungan dimana objek anak berbagi perilaku dan dari objek induk
7.		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8.		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur


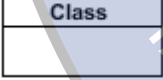
Simbol *Sequence Diagram*

Simbol <i>Sequence Diagram</i>			
No.	Gambar	Nama	Keterangan
1.		<i>Lifeline</i>	Peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i>
2.		<i>Message</i>	Hubungan dimana perubahan yang terjadi pada suatu element mandiri

Simbol *Activity Diagram*

Simbol <i>Activity Diagram</i>			
No.	Gambar	Nama	Keterangan
1.		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2.		<i>Decision</i>	Digunakan untuk menggambarkan suatu keputusan yang harus diambil.
3.		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4.		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan

Simbol *Class Diagram*

Simbol <i>Class Diagram</i>			
No.	Gambar	Nama	Keterangan
1.		<i>Generalization</i>	Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk.
3.		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama

DAFTAR ISI

LEMBAR COVER

LEMBAR JUDUL i

LEMBAR PENGESAHAN SKRIPSI..... ii

LEMBAR PENGESAHAN PENGUJI iii

DAFTAR GAMBAR iv

DAFTAR TABEL..... v

DAFTAR SIMBOL..... vi

BAB I PENDAHULUAN 1

A. Latar Belakang 1

B. Rumusan Masalah 2

C. Ruang Lingkup 3

D. Tujuan dan Manfaat 3

1. Tujuan 3

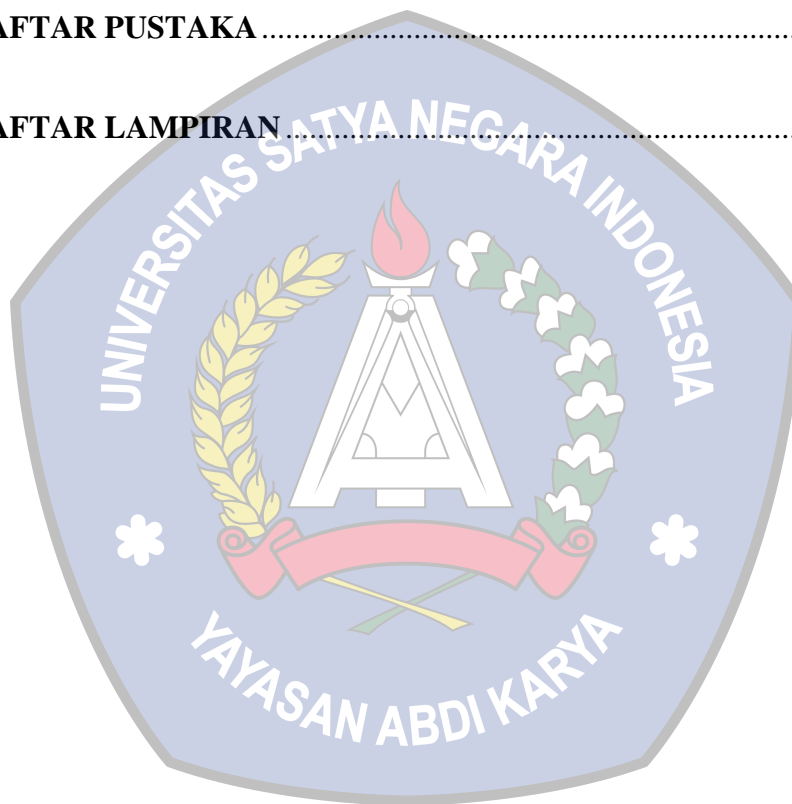
2. Manfaat 4

E. Sistematika Penulisan 5

BAB II LANDASAN TEORI	6
A. Tinjauan Pustaka	6
B. Pengertian Concierge	7
C. Pengertian Hotel	7
D. Pengertian Pelayanan	8
E. Definisi Sistem Informasi.....	8
F. Pengertian Analisis Sistem.....	11
G. Pengertian Perancangan Sistem.....	12
H. Pengertian Aplikasi	12
I. Pengertian Android.....	12
J. Pengertian Java.....	13
K. Pengertian Android SDK.....	14
L. Android Studio	14
M. UML (<i>Unified Modeling Language</i>).....	19
N. Pengujian Perangkat Lunak / <i>Software Testing</i>	20
O. Pengujian <i>White Box</i>	21
P. Pengujian <i>Black Box</i>	22
Q. <i>JavaScript Object Notation</i> (JSON)	23
R. Pengertian Basis Data.....	24
S. <i>System Development Life Cycle</i> (SDLC).....	24
BAB III ANALISIS HASIL DAN PEMBAHASAN	26
A. Waktu dan Tempat Penelitian	26
B. Logo Perusahaan	27

C. Sejarah Singkat Hotel Century Park.....	27
D. Struktur Organisasi.....	30
E. Metode Pengumpulan Data	33
F. Metodologi Pengembangan Sistem	34
G. Analisis Kebutuhan Sistem	34
H. Kerangka Berpikir	35
I. Analisa Sistem Berjalan	36
J. Analisa Sistem Usulan.....	41
K. Perancang Sistem.....	45
L. <i>Use Case Diagram</i>	45
M. <i>Activity Diagram</i>	46
N. <i>Sequence Diagram</i>	50
O. <i>Class Diagram</i>	54
P. Spesifikasi Basis Data	55
Q. Perancangan Antar Muka	60
1. Rancangan pada Aplikasi Android.....	60
2. Rancangan pada Aplikasi Website.....	70
BAB IV IMPLEMENTASI DAN EVALUASI	74
A. Hasil dan Pembahasan	74
1. Tampil Menu <i>Login</i>	74
2. Tampilan Halaman <i>Dashboard</i>	75
3. Tampilan master Pegawai	75
4. Tampilan Menu Bellboy Master	76

5. Tampilan Menu <i>Input Customer</i>	77
6. Tampilan Menu Master Kamar	77
7. Tampilan Menu <i>Create Customer</i>	78
B. <i>Testing Software</i>	83
BAB V KESIMPULAN DAN SARAN	86
DAFTAR PUSTAKA	87
DAFTAR LAMPIRAN	89



BAB I

PENDAHULUAN

A. Latar Belakang

Hotel Century Park adalah salah satu hotel berbintang 4 di Jakarta yang mempunyai 476 kamar, dimana lokasi hotel yang sangat strategis di area bisnis Jakarta. Persaingan di bidang jasa pelayanan hotel sangat tinggi yaitu di tandai dengan banyaknya hotel-hotel baru di sekitar Jakarta.

Pelayanan (*service*) pada dunia perhotelan merupakan kunci untuk meningkatkan tingkat hunian di sebuah hotel. Tingkat hunian hotel yang tinggi yaitu rata-rata 70% - 90%. Pelayanan kepada pelanggan dituntut untuk lebih efektif dan efisien. *Department Concierge* merupakan salah satu bagian dari *Front Office*. Kepuasan seorang pelanggan merupakan timbal balik dari kualitas pelayanan yang prima. Ada kelemahan ketika banyaknya permintaan tamu, yang terjadi di lapangan seorang bellboy tidak menjalankan prosedur yang ada yaitu salah satunya tidak merekap barang di *errand card*. Sehingga tidak ada catatan barang yang dibawanya, yang seharusnya seorang bellboy harus merekap berdasarkan jumlah dan jenis barang di *errand card*. Di dalam kertas *errand card* terdapat no.kamar, *no payroll bellboy*, jenis kegiatan (*check-in/check-out*) jenis barang dan tanggal. Ketika bellboy selesai mengerjakan tugas, selanjutnya harus direkap kedalam buku laporan (*control sheet*).

Semua laporan masih ditulis secara manual sehingga belum ada pengarsipan data yang baik dan aman. Seperti laporan pada *control sheet* masih berupa lembaran-lembaran kertas yang tidak di file dalam *folder* dokumen melainkan hanya digulung, kemudian dalam jangka waktu 2 sampai 3 bulan ketika laci sudah penuh maka *control sheet* tersebut dibuang dan tidak ada pengarsipan. Sehingga kesulitan mencari data barang maupun data tamu ketika diperlukan.

Dari uraian di atas penulis berinisiatif membuat rancangan sistem informasi pelayanan tamu berbasis android. Dimana aplikasi ini nantinya dapat digunakan oleh *staff concierge* maupun *customer*. Untuk meningkatkan daya saing dan nilai jual Hotel Century Park dengan hotel lain disekitar Jakarta maupun diluar Jakarta. Keamanan data barang dan data customer lebih terjamin karena sudah terintegrasi dengan sistem database, dan pengarsipan menjadi lebih baik. Secara tidak langsung dapat meningkatkan kualitas pelayanan kepada tamu yang menginap. Sehingga di harapkan dapat meningkatkan *revenue* hotel.

B. Rumusan Masalah

Berdasarkan latarbelakang diatas, maka didapatkan rumusan masalah sebagai berikut:

1. Bagaimana merancang sistem informasi pelayanan tamu pada *Department Concierge* di Hotel Century Park berbasis android?
2. Bagaimana mengimplementasikan pemanfaatan aplikasi pelayanan tamu pada *Department Concierge* di Hotel Century Park?

C. Ruang Lingkup

Agar tidak keluar dari materi pembahasan maka akan diberikan beberapa batasan masalah, sebagai berikut:

1. Perancangan sistem informasi ini hanya berfokus pada *Department Concierge* di Hotel Century Park
2. Mengimplementasikan sistem informasi pelayanan tamu berbasis android pada *Department Concierge* di Hotel Century Park.

D. Tujuan dan Manfaat

1. Tujuan

Tujuan dari penelitian ini, antara lain:

1. Untuk merancang dan membuat sistem informasi pelayanan tamu pada *Department Concierge* di Hotel Century Park berbasis android
2. Menghasilkan sistem informasi pelayanan tamu berbasis android pada *Department Concierge* di Hotel Century Park berbasis android
3. Mengetahui *feedback* dari *Customer* melalui aplikasi berbasis android.

2. Manfaat

Sedangkan manfaat yang diperoleh dari penelitian ini, antara lain:

a) Bagi pembaca:

1. Sebagai pedoman untuk peneliti lain yang sejenis dengan sistem pelayanan tamu pada *Department Concierge* di Hotel Century Park Berbasis android
2. Memberikan edukasi kepada pembaca tentang Sistem informasi pelayanan tamu pada *Department Concierge* di Hotel Century Park Berbasis android.

b) Bagi Perusahaan:

1. Mendapatkan solusi pengarsipan data barang dan *customer* pada *Department Concierge* dengan rancangan sistem informasi pelayanan tamu.
2. Meningkatkan kualitas pelayanan kepada tamu-tamu yang menginap di Hotel Century Park sehingga secara tidak langsung dapat meningkatkan pendapatan (*revenue*) perusahaan.

c) Bagi Peneliti

1. Dapat mengimplementasikan ilmu analisa dan perancangan sistem informasi yang didapat dari perkuliahan.
2. Dapat memperdalam ilmu peneliti, khususnya dalam merancang sebuah sistem informasi.
3. Untuk mengetahui pengarsipan data barang dan *customer* terhadap solusi sistem pelayanan pada *Department Concierge*.

d) Bagi *User*

1. Dengan adanya aplikasi ini bellboy dapat lebih mudah dalam melihat dan merekap hasil kerjanya.
2. Dengan adanya aplikasi ini seorang bellcaptain dapat lebih mudah dalam mencari data barang dan *customer* jika diperlukan.
3. Bellcaptain dapat lebih mudah dalam mendelegasikan tugas kepada bellboy.
4. Mempermudah *customer* dalam mendapatkan layanan bellboy.
5. Dapat mempermudah *customer* dalam mendapatkan informasi tentang barang bawaannya.

E. Sistematika Penulisan

Untuk memberikan gambaran secara keseluruhan dari penulisan terdiri dari lima bab dengan rincian sebagai berikut :

BAB I : PENDAHULUAN

Pada bab ini dibahas tentang latar belakang, rumusan masalah, batasan masalah, tujuan dan manfaat, serta sistematika penulisan.

BAB II : LANDASAN TEORI

Pada bab ini menjelaskan tentang metode pengumpulan data, teori-teori dan konsep-konsep yang relevan dengan masalah yang diteliti.

BAB III : ANALISIS DAN HASIL PEMBAHASAN

Bab ini menguraikan secara rinci analisis kebutuhan sistem, kerangka berfikir, metodologi penelitian yang digunakan dalam analisis sistem dan perancangan sistem.

BAB IV : IMPLEMENTASI DAN EVALUASI

Bab ini berisi implementasi dan evaluasi terhadap manajemen proyek yang telah diterapkan dalam proses pembangunan perangkat lunak, serta pengujian sistem.

BAB V : KESIMPULAN DAN SARAN

Pada bab ini dibahas tentang kesimpulan dari pembahasan secara menyeluruh serta saran-saran untuk pengembangan lebih lanjut.



BAB II

LANDASAN TEORI

A. Tinjauan Pustaka

Dalam hal ini penulis menggunakan referensi jurnal skripsi tentang perancangan sistem pelayanan tamu pada perhotelan untuk menjadi kajian teori yang ada, berikut adalah jurnal sebagai referensi yang digunakan oleh penulis :

1. Rancang bangun sistem pelayanan tamu pada hotel Rensa Jakarta oleh Nuzuliarini Nuris dan Eka Rini Yulia, Mahasiswa Sistem informasi STMIK Nusa Mandiri Jakarta. Dalam penelitian ini bertujuan untuk menghasilkan sebuah sistem informasi pelayanan kamar yang lebih cepat, tepat guna, efektif dan efisien pada Hotel Rensa. Hasil dari penelitian ini adalah mempermudah karyawan Hotel Rensa dalam proses pelayanan, membantu dalam proses penginputan data, pencarian data, dan laporan data transaksi, dapat meminimalisasi adanya kesalahan dan mengoptimalkan keamanan data.
2. Perancangan Sistem Informasi Pelayanan Hotel Menggunakan Metode *Waterfall* oleh Tuti Haryanti, Mahasiswa Akademi manajemen informatika AMIK BSI Karawang. Teknologi komputer digunakan dalam bentuk aplikasi pengolahan data komputer untuk pelayanan hotel. Dengan memanfaatkan teknologi komputer maka akan mempermudah suatu

pekerjaan tertentu seperti halnya pengolahan data lebih cepat, keputusan yang akan diambil lebih cepat, menghemat tempat atau ruang penyimpanan data dan lain sebagainya. Sistem administrasi yang digunakan oleh Hotel Nusantara dalam menangani penyewaan kamar dapat dilakukan dengan cara komputerisasi.

B. Pengertian Concierge

Mengacu pada buku yang berjudul “*Front Office Praktis*” oleh Prasetyo Aji prakoso(2017:105) Istilah concierge berasal dari bahasa perancis, yang memiliki arti “*keeper of the keys*”. *Keeper* disini maksudnya adalah penjaga atau pemelihara, dimana sebagai penjaga concierge menjadi poin utama kontak dengan tamu hotel yang mencari informasi dan bantuan selama mereka tinggal. Concierge adalah petugas hotel yang bertanggung jawab atas penanganan dan penyimpanan barang-barang tamu.

C. Pengertian Hotel

Istilah hotel di kenal di Amerika pada tahun 1750 dan berkembang pesat pada tahun 1820. Hotel sebagaimana jenis akomodasi lain berasal dari kata “INN” yang dapat di artikan sebagai usaha menyewakan sebagian dari rumahnya kepada orang lain yang memerlukan kamar untuk menginap. Prakoso (2017:1)

D. Pengertian Pelayanan

Pada dasarnya pelayanan merupakan suatu bentuk penilaian konsumen terhadap tingkat pelayanan yang di terima (*perceived service*) dengan tingkat pelayanan yang diharapkan (*expected service*). Bagi usaha perhotelan kuncinya adalah menyesuaikan atau melebihi harapan mutu jasa yang di inginkan konsumen. (Prakoso, 2017:12)

E. Definisi Sistem Informasi

Mengacu pada penulis Hamim Tohari (2014:2) dalam bukunya yang berjudul "Astah Analisis Serta Perancangan Sistem Informasi Melalui Pendekatan UML". Sistem adalah sekumpulan objek-objek yang saling berelasi dan berinteraksi, serta hubungan antar objek bisa dilihat sebagai satu kesatuan yang dirancang untuk mencapai satu tujuan yang telah ditetapkan.

Ada beberapa elemen yang membentuk sebuah sistem, yaitu :

1. Tujuan

Setiap sistem memiliki tujuan (*Goal*). Tujuan inilah yang menjadi pemotivasi yang mengarahkan sistem. Tanpa tujuan, sistem menjadi tak terarah dan tak terkendali.

2. Masukan

Masukan (*input*) sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan yang diproses.

3. Proses

Proses merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran yang berguna dan lebih bernilai.

4. Keluaran

Keluaran (*output*) merupakan hasil dari pemrosesan. Pada sistem informasi, keluaran bisa berupa suatu informasi, saran, cetakan laporan, dan sebagainya.

5. Batas

Yang disebut batas (*boundary*) sistem adalah pemisah antara sistem dan daerah di luar sistem (lingkungan). Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem.

6. Mekanisme pengendalian dan umpan balik

Mekanisme pengendalian (*control mechanism*) diwujudkan dengan menggunakan umpan balik (*feedback*), yang mencuplik keluaran. Umpan balik ini digunakan untuk mengendalikan baik masukan maupun proses. Tujuannya adalah untuk mengatur agar sistem berjalan sesuai dengan tujuan.

7. Lingkungan

Lingkungan adalah segala sesuatu yang berada di luar sistem. Lingkungan bisa berpengaruh terhadap operasi sistem dalam arti bisa merugikan atau menguntungkan sistem itu sendiri. Lingkungan yang merugikan tentu saja harus ditahan dan dikendalikan supaya tidak mengganggu kelangsungan operasi sistem, sedangkan yang menguntungkan tetap harus terus dijaga, karena akan memacu terhadap kelangsungan hidup sistem.

Menurut Hamim Tohari (2014:7), informasi adalah data yang telah diproses sedemikian rupa sehingga memiliki arti yang lebih bermanfaat bagi pengguna.

Informasi merupakan aset penting bagi suatu institusi atau organisasi. Oleh karena itu, informasi harus berkualitas, dijaga dan dipelihara dengan baik.

Sedangkan kualitas dari sebuah informasi tergantung dari tiga hal :

a. Akurat

Akurat berarti informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan. Informasi harus akurat karena sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan yang dapat merubah atau merusak informasi tersebut.

b. Tepat pada waktunya

Tepat waktu berarti informasi yang disampaikan ke penerima tidak terlambat, karena informasi adalah landasan untuk mengambil suatu keputusan.

c. Relevan

Relevan berarti informasi tersebut memiliki manfaat untuk pemakainya.

Relevansi informasi untuk tiap-tiap orang dengan yang lainnya berbeda.

Berdasarkan buku karya Hamim Tohari (2014:9). Sistem informasi adalah kumpulan dari sub-sub sistem baik fisik maupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berguna.

F. Pengertian Analisis Sistem

Menurut Hamim Tohari (2014:16), analisis sistem adalah tahapan merencanakan dan menerapkan rancangan sistemnya sesuai dengan kebutuhan user.

Terdapat empat langkah dasar yang harus dilakukan oleh analis sistem pada saat akan melakukan pembanguna dan pengembangan sistem, langkah tersebut antara lain :

1. *Identify*

Merupakan langkah pertama dalam tahap analisis sistem, dengan cara mengidentifikasi terlebih dahulu sebuah masalah, sebelum dilakukan pemecahkan permasalahannya.

2. *Understand*

Langkah ini dapat dilakukan dengan mempelajari secara terinci bagaimana sistem yang ada beroperasi.

3. *Analyze*

Langkah ini dapat dilakukan berdasarkan data yang diperoleh dari hasil penelitian yang telah dilakukan.

4. *Report*

Setelah membuat laporan hasil analisis selanjutnya diserahkan ke panitia pengarah (*steering committee*) yang nantinya akan diteruskan ke manajemen.

G. Pengertian Perancangan Sistem

Menurut Hamim Tohari (2014:16), perancangan tahap lanjutan setelah analisis berorientasi objek dimana tujuan sistem diorganisasikan kedalam sub-sistem berdasar struktur analisis dan arsitektur yang dibutuhkan.

H. Pengertian Aplikasi

Menurut Asropudin (2013:6), aplikasi adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Ms.Word*, *Ms.Excel*.

I. Pengertian Android

Android adalah merupakan sebuah sistem operasi seluler dan komputer tablet dan layar sentuh (*touchscreen*) yang berbasis linux. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang akan digunakan untuk bermacam peranti bergerak. Awalnya, *Google Inc.* membeli *Android Inc.* Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk *Google*, *HTC*, *Intel*, *Motorola*, *Qualcomm*, *T-Mobile*, dan *Nvidia*. Pada saat perilisan perdana Android, 5 November 2007 Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, *Google* merilis kode-kode Android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler. (Yudhanto.et al:2018).

J. Pengertian Java

Menurut Budi Raharjo , Imam Heryanto, Arif haryono (2010) *java* adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di *Sun Microsystems* saat ini merupakan bagian dari *Oracle* dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal. Aplikasi-aplikasi berbasis *Java* umumnya dikompilasi ke dalam p-code (*bytecode*) dan dapat dijalankan pada berbagai Mesin *Virtual Java* (JVM). *Java* merupakan bahasa pemrograman yang bersifat umum/non-spesifik (*general purpose*), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi *Java* mampu berjalan di beberapa platform sistem operasi yang berbeda, *Java* dikenal pula dengan slogannya, "Tulis sekali, jalankan di mana pun". Saat ini *Java* merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis *web*.

K. Pengertian Android SDK

Menurut Nazruddin Safaat H (2011), android SDK adalah *tools API* (*Application Programming Interface*) yang diperlukan untuk mulai

mengembangkan aplikasi pada *platform android* menggunakan bahasa pemrograman. Untuk memperkaya aplikasi dengan fungsionalitas canggih, iklan, mendorong pemberitahuan dan banyak lagi, sebagian besar pengembang aplikasi menerapkan perangkat pengembangan perangkat lunak tertentu.

L. Android Studio

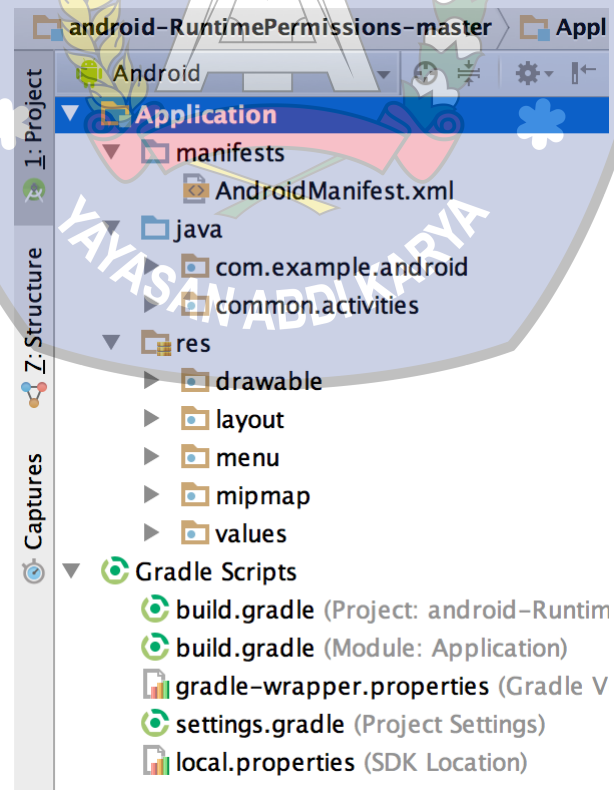
Android studio merupakan lingkungan pengembangan terpadu - *integrated development environment (IDE)* untuk pengembangan aplikasi android, berdasarkan [IntelliJ IDEA](#). Selain merupakan editor kode *IntelliJ* dan alat pengembang yang berdaya guna, Android Studio menawarkan fitur lebih banyak untuk meningkatkan produktivitas pada saat membuat aplikasi Android, misalnya:

- a. Sistem versi berbasis *gradle yang fleksibel*
- b. *Emulator* yang cepat dan kaya fitur
- c. Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android
- d. *Instant Run* untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru
- e. Template kode dan *integrasi GitHub* untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh
- f. Alat pengujian dan kerangka kerja yang ekstensif

- g. Alat *Lint* untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah-masalah lain.

Secara *default*, Android Studio akan menampilkan file proyek dalam tampilan proyek Android, tampilan disusun berdasarkan modul untuk memberikan akses cepat ke *file* sumber utama proyek. Semua *file* versi terlihat di bagian atas di bawah *gradle scripts* dan masing-masing modul aplikasi berisi folder berikut:

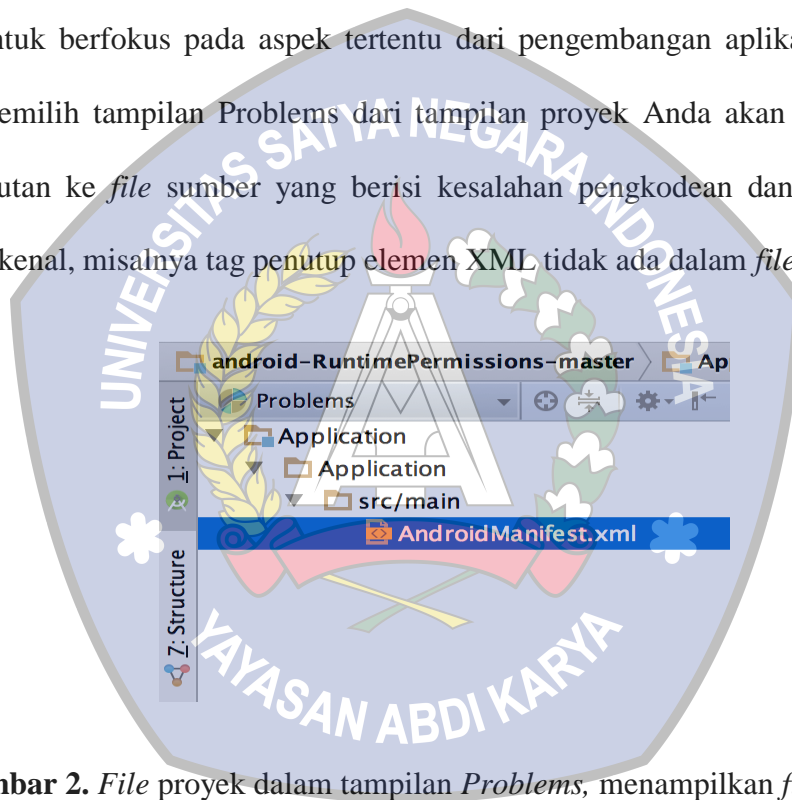
- a. **manifests**: berisi file android *manifest.xml*.
- b. **java**: berisi file kode sumber *java*, termasuk kode pengujian *JUnit*.
- c. **res**: Berisi semua sumber daya bukan kode, seperti tata letak XML, string UI, dan gambar *bitmap*.



Gambar 1. *File proyek di tampilan Android.*

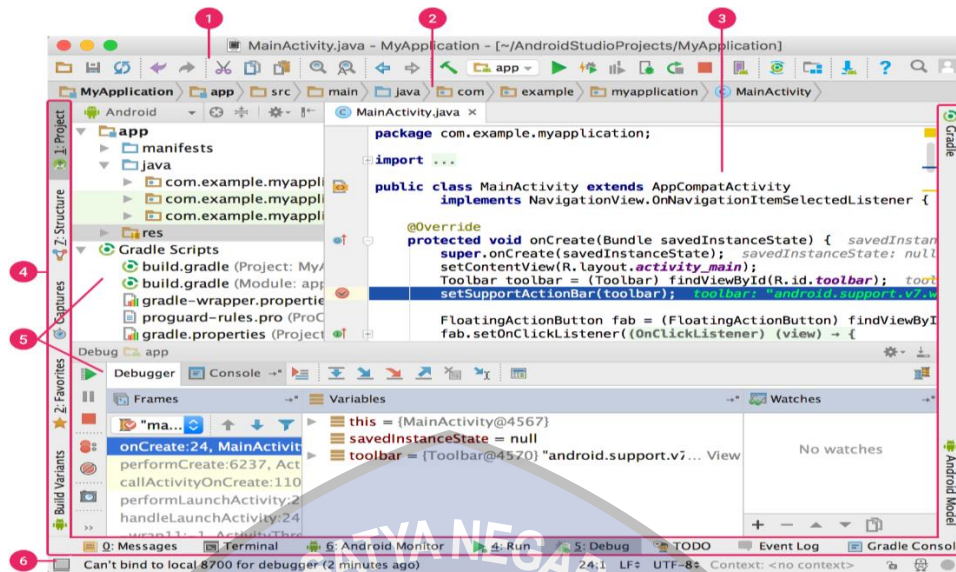
Sumber : <https://developer.android.com>

Struktur proyek Android pada *disk* berbeda dari representasi rata ini. Untuk melihat struktur *file* sebenarnya dari proyek ini, pilih *Project* dari menu tarik turun *Project* atau juga bisa menyesuaikan tampilan *file* proyek untuk berfokus pada aspek tertentu dari pengembangan aplikasi. Misalnya, memilih tampilan *Problems* dari tampilan proyek Anda akan menampilkan tautan ke *file* sumber yang berisi kesalahan pengkodean dan *sintaks* yang dikenal, misalnya tag penutup elemen XML tidak ada dalam *file* tata letak.



Gambar 2. *File proyek dalam tampilan Problems, menampilkan file tata letak yang bermasalah.*

Sumber : <https://developer.android.com>



Gambar 3. *Jendela utama Android Studio.* Sumber :

<https://developer.android.com>

Bagian dari jendela utama Android Studio, antara lain :

1. Bilah alat digunakan untuk melakukan berbagai jenis tindakan, termasuk menjalankan aplikasi dan meluncurkan alat android.
2. Bilah navigasi digunakan sebagai navigasi di antara proyek dan membuka file untuk diedit. Bilah ini memberikan tampilan struktur yang terlihat lebih ringkas dalam jendela project.
3. Jendela *editor* adalah tempat untuk membuat dan memodifikasi kode. Bergantung pada jenis *file* saat ini, editor dapat berubah. Misalnya, ketika melihat *file* tata letak, *editor* menampilkan layout *editor*.

4. Bilah jendela alat muncul di luar jendela IDE dan berisi tombol yang berfungsi untuk meluaskan atau mengecilkan jendela alat individual.
5. Jendela alat berfungsi sebagai pengelolaan proyek, penelusuran, kontrol versi, dan banyak lagi.
6. Bilah status menampilkan status proyek dan IDE, serta setiap peringatan atau pesan.

Jendela utama juga bisa dirubah sesuai keinginan untuk memberi kesan ruang layar yang lebih luas dengan menyembunyikan atau memindahkan bilah alat dan jendela alat. Selain itu bisa juga dengan menggunakan pintasan *keyboard* untuk mengakses sebagian besar fitur IDE.

M. UML (*Unified Modeling Language*)

Menurut Adi Nugroho (2014:22), "UML (*Unified Modeling Language*) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma (berorientasi objek)." Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.. UML terdiri dari himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (*OOP*) serta aplikasinya.

Ada 4 (empat) macam diagram dalam *Unified Modeling Language* (UML), yaitu :

1. *use case Diagram*, *Use case* bekerja dengan cara mendeskripsikan tipikal interaksi antara pengguna sebuah sistem (*aktor*) dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai.
2. *Sequence Diagram*, Diagram ini memperlihatkan interaksi yang menekankan pada pengiriman pesan (*message*) dalam suatu waktu tertentu.
3. *State Chart Diagram*, Diagram ini memperlihatkan *state-state* pada sistem, memuat *state*, *transisi*, *event*, dan aktifitas. Diagram ini terutama penting untuk memperlihatkan sifat dinamis dari antarmuka, kelas, kolaborasi dan terutama penting pada pemodelan sistem-sistem yang reaktif.
4. *Activity Diagram*, Diagram ini memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi dalam suatu sistem dan memberi tekanan pada aliran kendali antar objek.

N. Pengujian perangkat lunak/*Software testing*

Pengujian perangkat lunak adalah proses menganalisis perilaku program dengan tujuan untuk menemukan apakah memenuhi persyaratan yang ditentukan atau tidak. Dengan kata lain pengujian perangkat lunak adalah mengelola sistem dalam urutan aliran untuk mengidentifikasi segala

cacat, kesalahan, bug, dan hilang dalam persyaratan aktual. Proses ini merupakan komponen penting dari jaminan kualitas perangkat lunak, dan banyak organisasi perangkat lunak dengan sumber daya 40% fokus pada pengujian, kemudian pengujian bisa sangat mahal dalam kasus seperti perangkat lunak yang kritis, contohnya kontrol penerbangan. (Wicaksono, 2017: 353).

O. Pengujian White Box

Pengujian *White Box* sangat efektif untuk mendeteksi dan menyelesaikan masalah, karena bug sering ditemukan sebelum mereka menyebabkan masalah. Pengujian *White Box* atau biasa disebut juga *Glass Testing*, *Open Box Testing*, *White Box analysis*, *Clear Box testing* atau *Clear box Analysis*, adalah strategi untuk *debugging* perangkat lunak di mana tester memiliki pengetahuan yang sangat baik tentang bagaimana komponen program berinteraksi. Metode ini dapat digunakan untuk aplikasi layanan *Web*, dan jarang praktis untuk *debugging* di sistem dan jaringan besar. Pengujian *White Box* dianggap sebagai kode. (Wicaksono, 2017: 353).

1. Keuntungan dari Pengujian White Box

1. Baris kode tambahan dapat dihilangkan yang dapat menyebabkan cacat tersembunyi.

2. Pengetahuan penguji tentang kode, cakupan maksimum diperoleh selama penulisan skenario pengujian.
3. Membantu dalam mengoptimalkan kode.

2. Kerugian Pengujian *White Box*

1. Penguji yang terampil diperlukan untuk melakukan pengujian *White Box* yang meningkatkan biaya.
2. Terkadang tidak mungkin untuk menemukan kesalahan tersembunyi yang dapat menimbulkan masalah karena banyak jalur yang tidak diuji.
3. Sulit untuk mempertahankan pengujian kotak putih karena penggunaan alat khusus seperti penganalisa kode dan alat *debugging* diperlukan.

P. Pengujian *Black Box*

Pengujian *Black box* adalah pengujian perangkat lunak berdasarkan persyaratan output. Dalam pengujian ini tidak perlu ada pengetahuan tentang struktur internal atau pengkodean dalam program. Penguji tidak menyadari Arsitektur sistem dan tidak memiliki akses ke kode sumber. Selama pengujian kotak hitam, tester akan berinteraksi dengan antarmuka pengguna sistem dengan memberikan input dan memeriksa output tanpa mengetahui bagaimana dan di mana input tersebut dikerjakan. Kotak hitam adalah algoritma yang tidak memberikan penjelasan tentang cara kerjanya. Dalam pembuatan film, *Black box* adalah perangkat atau

Hardware khusus yang khusus digunakan untuk fungsi tertentu.
(Wicaksono, 2017: 355.)

1. Keuntungan dari pengujian *Black box*

1. Tidak ada pengetahuan tentang implementasi, bahasa pemrograman atau sistem operasi.
2. Efisien untuk segmen kode besar.
3. Secara jelas memisahkan perspektif pengguna dari perspektif pengembang melalui peran yang jelas terlihat.

2. Kerugian Pengujian *Black box*.

1. Kasus uji sulit untuk dirancang.
2. Cakupan terbatas selama hanya sejumlah skenario uji yang dipilih yang benar benar dilakukan.
3. Pengujian tidak efisien, karena tester hanya memiliki pengetahuan yang terbatas tentang suatu aplikasi.
4. Cakupan Buta, tester tidak dapat fokus pada segmen kode tertentu atau area rawan kesalahan.

Q. *JavaScript Object Notation (JSON)*

JavaScript *Object Notation* (JSON). JSON merupakan suatu format pertukaran data yang ringan, yang mudah di baca dan ditulis, baik bagi manusia maupun mesin. (Hansun.S,et al:2018).

R. *Pengertian Basis Data (Database)*

Menurut Kustiyaningsih (2011:146), *database* adalah struktur penyimpanan data. Untuk menambah, mengakses dan memproses data yang disimpan dalam sebuah *database* komputer, diperlukan sistem manajemen database seperti *MYSQL Server*”.

S. *System Development Live Cycle (SDLC)*

SDLC adalah tahapan yang umum dilakukan dalam pembuatan atau perbaikan/penggantian sebuah sistem dan mengacu juga pada metodologi yang digunakan untuk mengembangkan sistem informasi.(Suprpto, 2018:6).

a. *Waterfall* merupakan model SDLC yang tertua dan paling populer. Pada SDLC model *waterfall*, terdapat beberapa urutan aktivitas sebagai berikut:

1. *Project Planning/ feasibility study*: menentukan tujuan atau target utama proyek.
2. *System Analysis/requirement definition* :Mendefinisikan tujuan proyek menjadi fungsi-fungsi dan operasi yang lebih terperinci, yang harus

disediakan oleh sistem informasi, termasuk menganalisis kebutuhan informasi *end-user*.

3. *System design* : Menggambarkan fitur dan operasi yang diharapkan dapat disediakan oleh sistem informasi secara lebih terinci, termasuk rancangan tampilan antarmuka, aturan bisnis, diagram proses dan dokumentasi lainnya.
4. Implementasi : Mewujudkan rancangan sistem dalam bentuk kode program/aplikasi.
5. Integrasi dan *testing* : Menggabungkan semua elemen dalam satu lingkungan uji yang terpadu, untuk memeriksa *error*, *bugs* dan *interoperability*
6. *Acceptance*, instalasi dan *deployment* : tahap terakhir pada suatu siklus awal pengembangan *software*, yaitu *software* dipasang dan digunakan langsung pada operasional bisnis yang sebenarnya.



BAB III

ANALISIS HASIL DAN PEMBAHASAN

A. Waktu dan Tempat Penelitian

Waktu penelitian ini di mulai dari bulan September 2019, dan penelitian ini di laksanakan di Hotel Century Park, Jl Pintu satu senayan, Jakarta 10270.

Berikut adalah tabel *Gantt Chart* untuk penjadwalannya:

Tabel 1. Jadwal Penelitian

Kegiatan	September2019				Oktober 2019				Novembr2019				Desember2019				Januari 2020			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Pengumpulan data		✓	✓	✓	✓															
Analisis Kebutuhan					✓	✓	✓	✓												
Desain Sistem								✓	✓	✓	✓	✓								
Perancangan Sistem											✓	✓	✓	✓	✓	✓				
Testing sistem																	✓	✓	✓	
Implementasi sistem																				✓

Pada *Gantt chart* diatas di ambil berdasarkan kegiatan riset yang telah di lakukan pada bulan September 2019, yaitu observasi. Kemudian dilakukan analisa kebutuhan hingga implementasi di bulan Januari 2019.

B. Logo Perusahaan



Gambar 4. Logo Hotel Century Park

Sumber : <http://indonesiadentalexpo.com>.

C. Sejarah Singkat Hotel Century Park

Hotel Century Park adalah hotel berbintang 4 yang merupakan suatu hotel yang tadinya di sediakan khusus untuk atlet. Karena Hotel Century Park mempunyai hubungan dengan koni. Oleh karena itu fasilitas kamar yg di sediakan untuk atlet terdiri 3 lantai saja dan selebihnya di sediakan atau di sewakan untuk umum. Masa pembangunan hotel ini adalah selama 2 thn dan memulai pengoperasiannya pada bulan desember 1991, dan akan mengembangkan sistem pelayanan dengan standar hotel internasional.

Didirikan pada tahun 1989 oleh presiden dan CEO yaitu Brian Deason dengan sasaran mengoperasikan hotel dan *resort*, utamanya di belahan asia pasifik dari pusat bisnisnya di Hongkong. Perusahaan Hotel Century Park ini bergerak di bidang perhotelan sesuai dengan namanya. Setelah saya analisa Hotel Century Park ini mempunyai tipe Organisasi lini dan fungsional di karenakan Hotel Century Park merupakan organisasi yang besar, mempunyai karyawan yang besar pula, sangat memaksimalkan spesialisasi karyawannya dan setiap departemen mempunyai kepala departemen yang bertanggung jawab atas departemennya. Lalu struktur organisasinya jika menurut cara pembuatannya merupakan gabungan antara horizontal dan vertikal, karena ada atasan dan yang setingkat. Dan jika menurut isi di dalamnya yaitu skema organisasi struktur karena yang dibuat hanya unit-unitnya saja bukan nama atasan.

Seiring pergantian manajemen baru yaitu salah satunya pada tahun 2010 mengalami perubahan nama menjadi Hotel Century Park dengan menghilangkan nama atletnya karena dengan salah satu alasan merubah mindset masyarakat umum bahwasanya Hotel Century Park bukanlah hotel khusus untuk para atlet melainkan tersedia juga untuk tamu-tamu pada umumnya. Hotel Century Park adalah hotel bintang 4 dengan 17 lantai dan memiliki 476 kamar. Lokasi hotel yang sangat strategis yaitu di jantung kota dan pusat area bisnis kemudian di kelilingi pusat perbelanjaan yaitu diantaranya ada Fx Mall, Plaza Senayan, Ratu Plaza dan Senayan City. Hotel Century Park merupakan salah satu hotel bisnis di Jakarta.

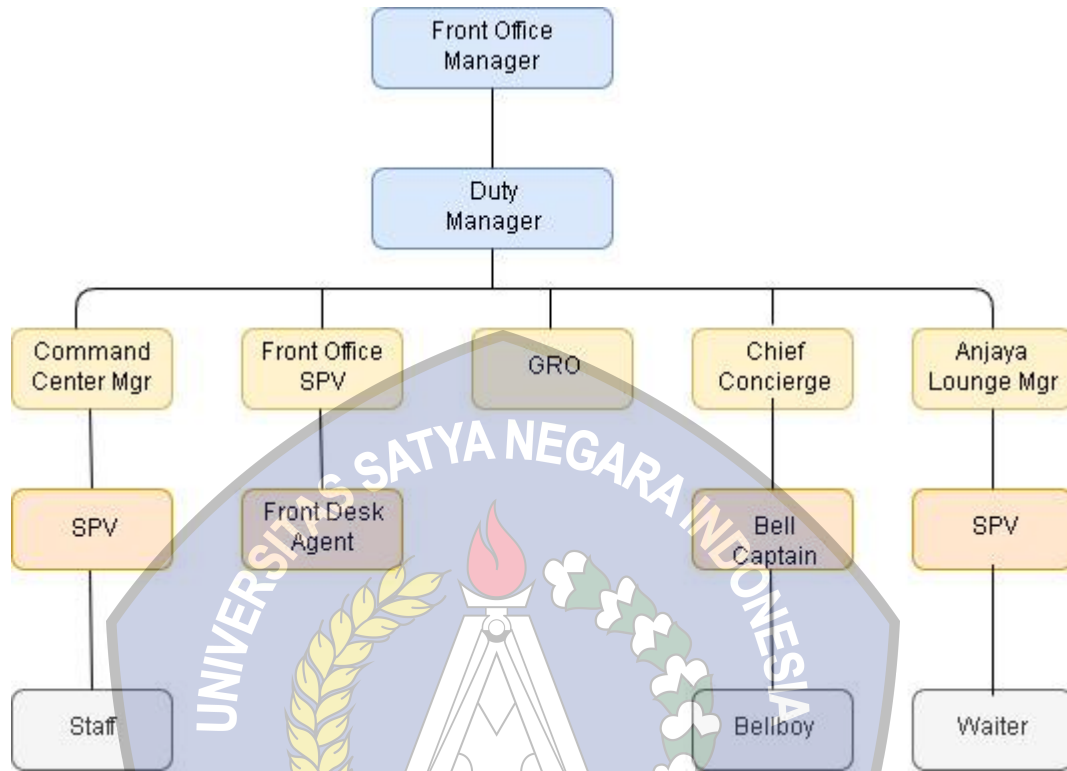
Visi merupakan sesuatu yang di inginkan oleh perusahaan yaitu Hotel Century Park yang ingin di capai dimasa mendatang. Visi Hotel Century Park adalah :

1. Menjadi pilihan utama hotel bisnis berbintang empat yang bertaraf internasional.
2. Menjadi yang terbaik dengan cara bekerja lebih efektif, aktif dan bersemangat setiap harinya adalah janji kita.

Misi adalah upaya yang di lakukan perusahaan untuk mencapai visi dan tujaun jangka panjang. Misi Hotel Century Park adalah :

1. Melanjutkan renovasi hotel dan fasilitas lainnya, untuk dapat bersaing denagan hotel sekelasnya dan dapat memenuhi kebutuhan serta kepuasan pelanggan.
2. Memberikan layanan “*Value for Money*” dan mempererat tali silaturahmi dengan pelanggan agar tetap setia dan menjadikan hotel kita sebagai rumah kedua.

D. Struktur Organisasi



Gambar 5. Struktur organisasi *department Front Office*

Pada gambar 5 menjelaskan struktur organisasi pada *department front office*, dimulai dari jabatan teratas yaitu manager front office sampai bagian staff. Adapun penjelasan mengenai tugas dan tanggung jawab ada pada bagian berikutnya.

Adapun tugas dan tanggung jawab dari masing-masing jabatan pada gambar antara lain:

1. Front Office Manager

Menjalankan perintah yang disampaikan oleh General Manager, bertanggung jawab atas *department front office*, memberi motivasi dan arahan kepada para chief department yang ada dibagian *front office*.

2. Duty Manager

Bertanggung jawab atas semua department head dibawah MOD (*manager of duty*), menjalankan perintah yang disampaikan oleh Front Office Manager. Menggantikan tugas FOM ketika sewaktu-waktu berhalangan. *Handling complaint* ketika ada tamu yang complain.

3. Chief Concierge

Bertanggung jawab atas department concierge dari operasional setiap harinya. Mengatur semua staff di concierge dan memberi arahan kepada bellcaptain dan team. Membuat *schedule training* setiap bulannya untuk para bawahannya.

4. Supervisor (all department)

Mengatur kerja para bawahannya (*staff*), Membuat *Job Descriptions* untuk Staf Bawahanya, Bertanggung jawab atas hasil kerja Staf, Memberi motivasi kerja kepada Staf Bawahanya, Membuat Jadwal Kegiatan Kerja untuk karyawan, Memberikan Briefing bersama Staf.

5. Tugas *Receptionist*

Reception adalah salah satu seksi dari *front office* (FO) yang mempunyai tanggung jawab terhadap penanganan tamu *check-in*, stay sampai *check-out*. Seorang reception adalah orang yang penting bagi tamu dan hotel. Bagi tamu petugas reception adalah orang yang paling sering di hubungi selama mereka menginap. Bagi hotel reception adalah orang yang menciptakan dan mempertahankan citra yang baik dan membuat tamu ingin datang kembali ke hotel.

6. Tugas telephone operator (*callcenter*)

Callcenter adalah salah satu seksi yang ada pada front office hotel yang mana mempunyai tugas dan tanggung jawab menangani/ menjawab sambungan telephone masuk (*incoming call*) atau keluar (*outgoing call*). sekarang bahwa sedang trend dimana *rule* dari *callcenter* hotel juga menjalankan tugas reservasi, *upselling*, dan sekaligus pemandu berbagai informasi tertentu yang di arahkan untuk terjadinya transaksi. Jadi tidak pasif menerima dan menyambungkan saja.

7. Tugas Bellboy

Bellboy adalah petugas pelayanan barang di *front office* yang mengurus barang bawaan tamu serta tugas lainnya pada saat tamu melakukan *check-in*, *check-out*, maupun *change room*.

8. Tugas Bellcaptain

Bellcaptain yaitu atasan dari bellboy diman tugasnya adalah mengatur jalannya operasional dari team bellboy diantaranya mendelegasikan tugas untuk mengantar barang bawaan tamu atau mengambilkan barang tamu dari kamar.

9. Tugas GRO

Guest relation officer adalah salah satu jabatan di *front office* yang fungsinya sebagai perantara atau jembatan antara tamu terutama tamu VIP dengan pihak hotel. Memberikan informasi tentang fasilitas hotel dan penanganan keluhan.

10. Tugas Concierge

Concierge adalah nama salah satu department bagian dari *Front Office*. Pada concierge ada beberapa bagian yaitu *Chief Concierge*, *Bellcaptain*, *Bellboy*, *Doorman* dan *Doorgirl*. Tugas dari concierge adalah menerima titipan barang maupun dokumen dari tamu maupun untuk tamu yang menginap, tugas lainnya adalah tempat mendapatkan informasi tentang kebutuhan tamu yang menginap, contohnya *ticketing*, *travelling*, reservasi golf maupun *restaurant*.

E. Metode pengumpulan data

Dalam pelaksanaan penelitian ini diperlukan data dan penelitian yang nantinya digunakan sebagai rujukan untuk merancang sebuah Sistem Informasi Pelayanan Tamu.

Adapun metode penelitian yang digunakan dalam penelitian ini :

1. Observasi, yaitu dengan meninjau dan menganalisa bagaimana berjalannya sistem berjalan pada *department concierge*.
2. Wawancara, Wawancara atau interview adalah proses memperoleh keterangan untuk tujuan penelitian dengan cara tanya jawab, sambil bertatap muka antara penanya atau pewawancara dengan penjawab atau responden. Metode wawancara ini dipilih karena efektif serta dapat mengetahui data informasi yang dibutuhkan secara langsung. Yaitu dengan mewawancarai Bapak Kundarto seorang *chief concierge* dan bapak Dahyar Fauzi *front office manager* salah satu *Department Head* terkait untuk mendapatkan informasi dan data.
3. *Study Literatur*, Melalui metode studi ini dapat memperoleh data atau informasi dengan cara mengumpulkan, mempelajari dan membaca berbagai referensi baik itu dari buku-buku, jurnal, makalah, internet dan berbagai sumber lainnya yang menunjang dalam penulisan ini.

F. Metodologi Pengembangan Sistem

Untuk metode pengembangan sistem sendiri penulis memilih metode *waterfall* Karena menurut peneliti, metode ini sangat relevan untuk perancangan dan cocok untuk sistem yang berbasis sekuensial, dimana setiap tahapan memiliki pengaruh yang penting, mulai dari analisa kebutuhan, desain sistem, perancangan program, pengujian program, hingga penerapan program.

G. Analisis Kebutuhan Sistem

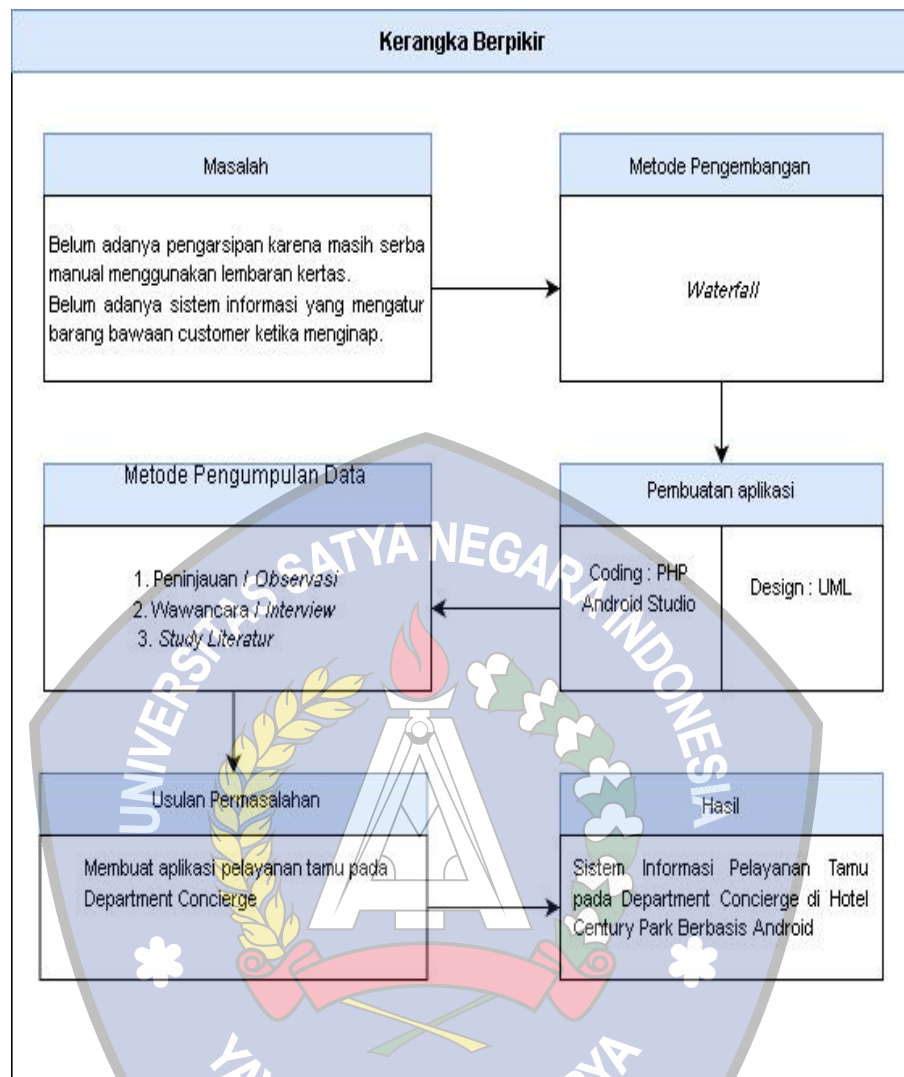
Adapun kebutuhan sistem untuk membangun aplikasi sistem pelayanan tamu pada department concierge dapat dilihat pada tabel dibawah ini:

Tabel 2. Analisa kebutuhan sistem

<i>Software</i>	<i>Hardware</i>
Sistem Operasi Windows 10	Laptop Acer Aspire V 13 Touch
Xampp(Php 7.1,Mysql)	Ram 4GB, <i>Processor</i> Intel Core i5-6200u 2,3 GHz
<i>Browser</i> (Mozilla)	V3-372T-53H4
Drawer IO.com	

H. Kerangka Berpikir

Pada perancangan Sistem Informasi Pelayanan Tamu pada *Department Concierge* di Hotel Century Park Berbasis Android dibutuhkan beberapa tahapan yang harus dilalui untuk dapat menghasilkan sistem informasi yang berguna bagi *Department Concierge* sendiri maupun Hotel Century Park dan *Management*.



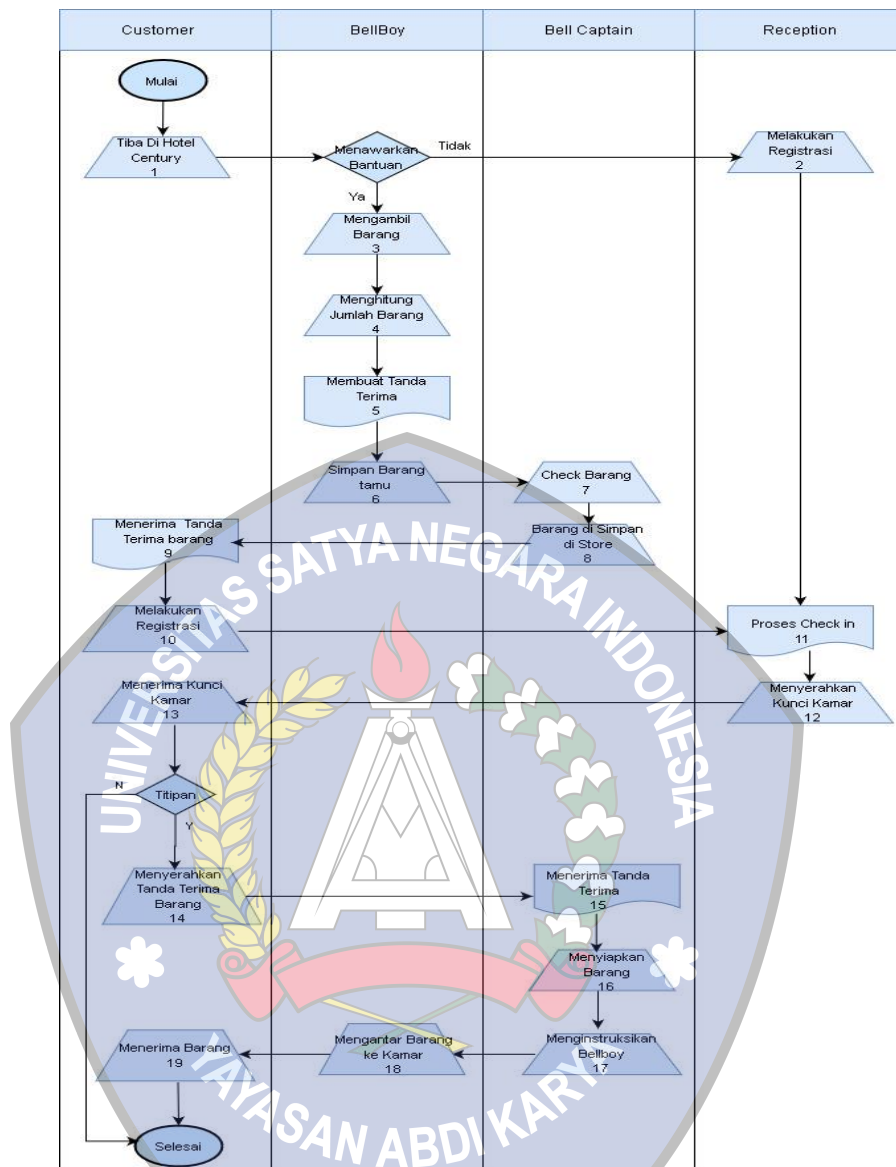
Gambar 6. Kerangka Berpikir

I. Analisa Sistem Berjalan

1. *Flow Map* Proses *Check-in*

Dibuat oleh peneliti untuk mengetahui alur proses sistem berjalan maupun usulan dari peneliti pada *department Concierge*.

Adapun flowmap berjalan pada *department Concierge* adalah sebagai berikut :



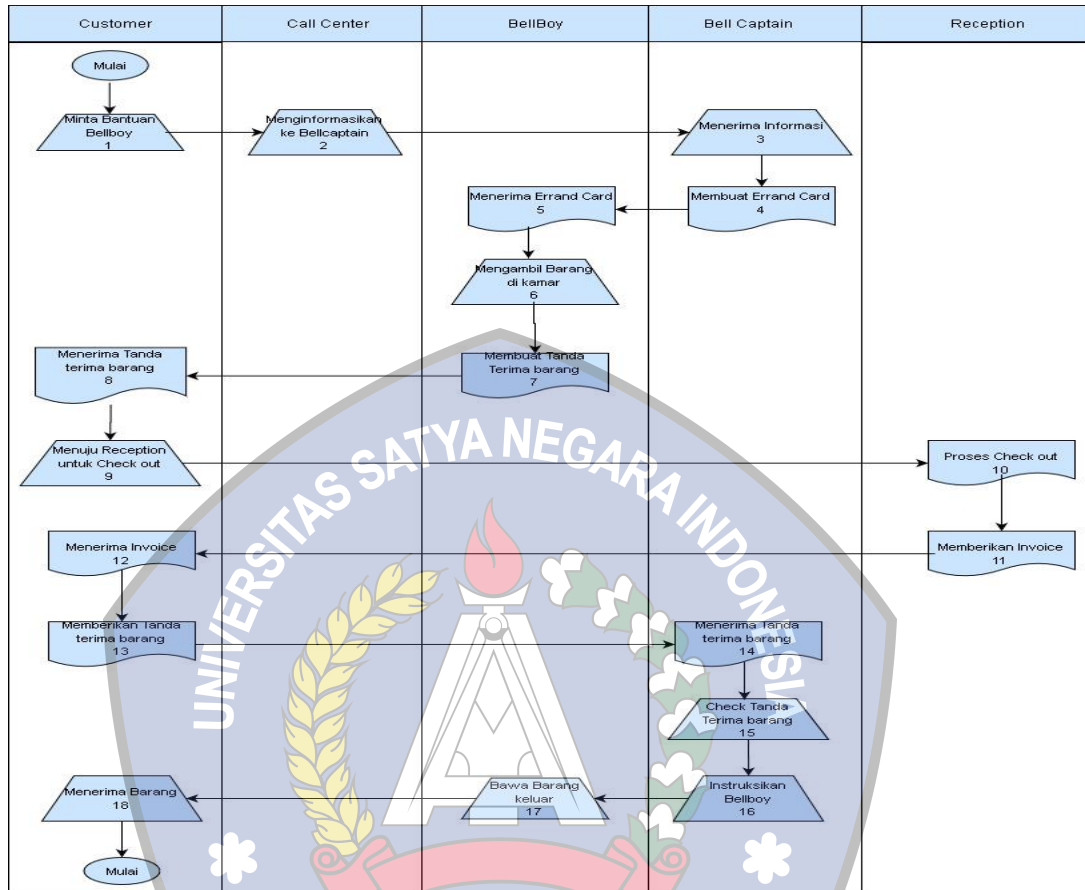
Gambar 7. Flow map Check-in

adapun flowmap berjalan berikut adalah deskripsinya :

1. Dimulai ketika customer tiba di Hotel Century Park
2. Customer langsung melakukan registrasi ke front office
3. Bellboy membantu mengambil barang dari kendaraan
4. Bellboy kemudian menghitung jumlah barang customer.

5. Bellboy membuat tanda terima dan memberikan tanda terima barang kepada customer, dimana ada dua bagian yang sama nomer tag nya, yang satu ikat di barang dan bagian lain untuk *customer*
6. Bellboy membawa barang ke store concierge untuk disimpan di *store*
7. Bellcaptain *check* barang yang diterima dari bellboy
8. Setelah selesai dicheck, didata kemudian disimpan bellcaptain membuatkan *errand card*
9. *Customer* menerima tanda terima barang
10. *Customer* melakukan registrasi ke reception
11. Reception melakukan proses *check in*
12. Reception memberikan kunci kamar setelah selesai registrasi
13. Tamu menerima kunci kamar
14. Setelah mendapatkan kunci kamar customer datang ke concierge dengan membawa tanda terima untuk pengambilan barangnya.
15. Bellcaptain menerima tanda terima barang dari *customer*.
16. Bellcaptain mempersiapkan barang yang akan di ambil oleh *customer*
17. Bellcaptain menginstruksikan kepada bellboy untuk mengantar barang ke kamar.
18. Bellboy mengantar barang ke kamar *customer*
19. *Customer* menerima barang dari bellboy.

2. Flow Map Check-out



Gambar 8. Flow Map Check-out

Adapun *Flow Map* berjalan berikut adalah deskripsinya :

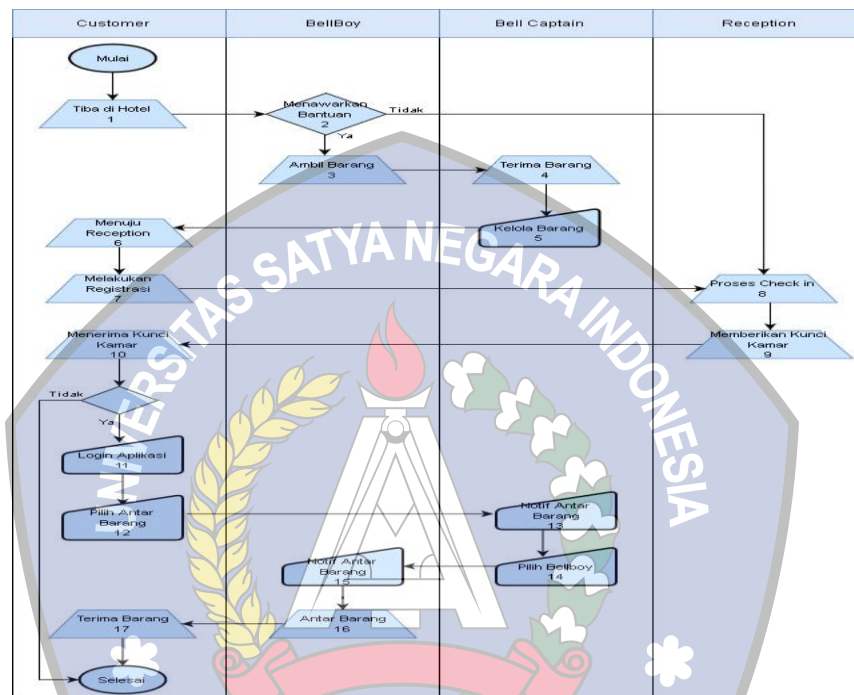
1. Customer telpon ke call center untuk minta bantuan untuk membawa barang
2. *Command center* menginformasikan ke Bellcaptain melalui telpon
3. Bellcaptain menyiapkan errand card untuk bellboy
4. Bellcaptain menginstruksikan bellboy untuk mengambil barang ke kamar Customer

5. Bellboy menerima errand card dari bellcaptain
6. Setelah mendapat errand card bellboy menuju kamar customer untuk mengambil barang.
7. Bellboy membuat tanda terima barang kemudian diberikan kepada customer
8. Customer menerima tanda terima barang dari bellboy
9. Customer menuju reception untuk melakukan check out
10. Reception melakukan proses check out
11. Setelah selesai proses checkout kemudian memberikan invoice ke customer.
12. Customer menerima invoice dari reception
13. Customer menuju concierge dengan menyerahkan tanda terima barangnya.
14. Bellcaptain menerima tanda terima barang dari customer.
15. Bellcaptain check nomer tanda terima dengan nomer yang ada di barang
16. Menginstruksikan kepada bellboy untuk membawakan barang customer
17. Bellboy membawa barang tamu keluar lobby
18. Customer menerima barang dari bellboy.

J. Analisa Sistem Usulan

1. Flow Map Usulan Check in

Adapun flow map check in usulan adalah:



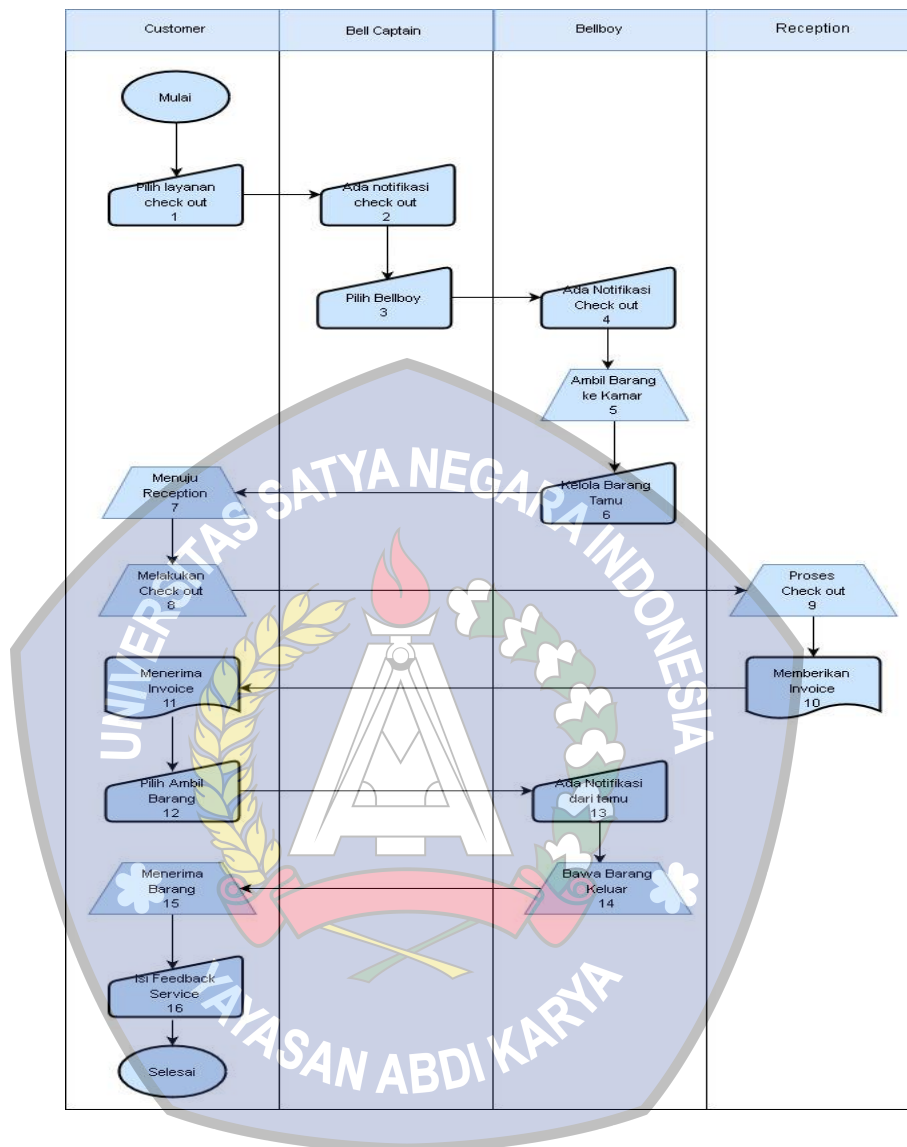
Gambar 9. Flow Map usulan Check in

Adapun deskripsi dari gambar usulan di atas adalah:

1. Dimulai dari *customer* tiba di hotel century park
2. Bellboy menawarkan bantuan ke *customer*, jika mau dibantu maka barang akan dibawa oleh bellboy dan jika tidak mau dibantu maka *customer* akan langsung menuju *reception*.
3. Bellboy membantu *customer* membawakan barangnya.
4. Bellcaptain menerima barang dari bellboy untuk disimpan di *store*

5. Bellcaptain input jumlah dan jenis barang ke aplikasi dan ambil gambar barang tersebut.
6. *Customer* menuju *reception* untuk melakukan registrasi *check in*
7. *Customer* melakukan registrasi *check in*
8. *Reception* melakukan registrasi *input data customer* pada *dashboard* aplikasi pelayanan concierge
9. *Reception* memberikan kunci kamar kepada *customer*
10. *Customer* menerima kunci kamar dari *reception*
11. *Customer log in* aplikasi android pelayanan concierge
12. Pilih layanan antar barang pada aplikasi android
13. Belcaptain mendapatkan notifikasi *inbox* dari *customer*
14. Belcaptain pilih bellboy dengan menggunakan aplikasi untuk mengerjakan tugas antar barang.
15. Bellboy menerima notifikasi *inbox* dari bellcaptain
16. Bellboy mengantar barang ke kamar *customer*.
17. *Customer* menerima barang dari bellboy.

2. Flow Map usulan *Check out*



Gambar 10. Flow Map usulan Check out

Deskripsi dari flow map usulan di atas adalah:

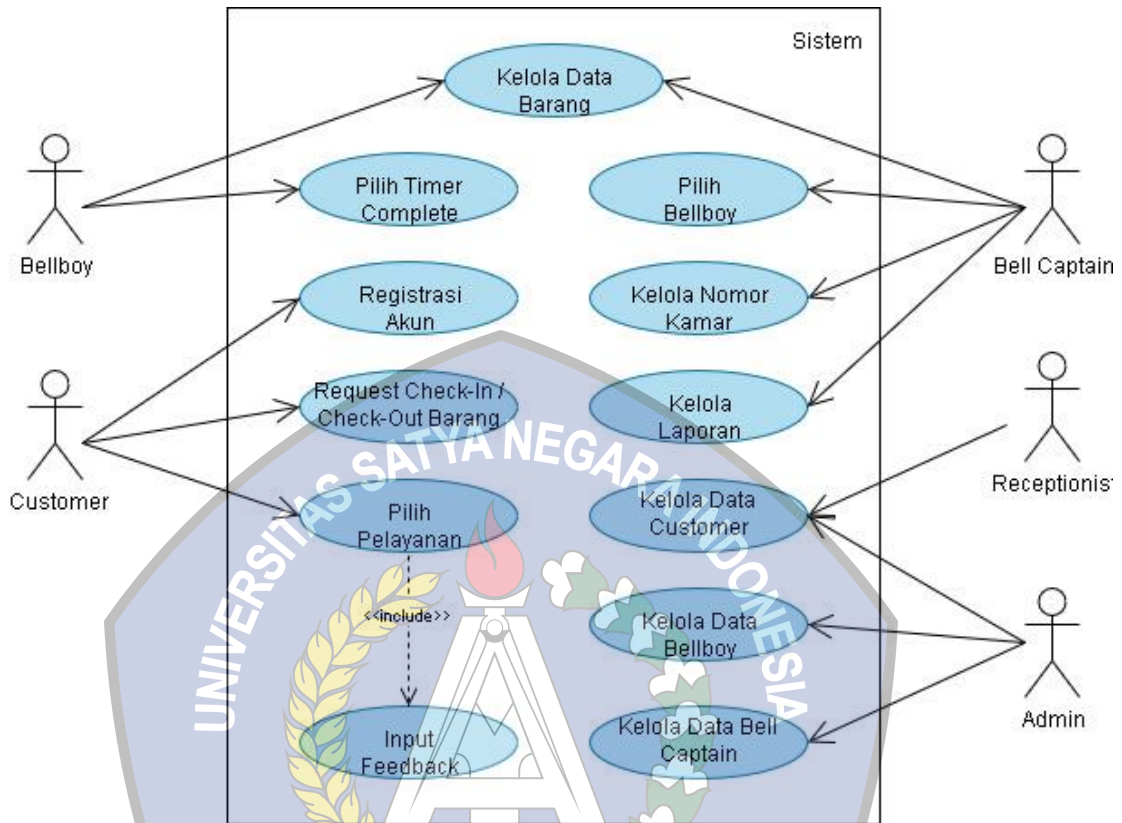
1. Dimulai dari *customer* memilih layanan *check out* dari aplikasi
2. Bellcaptain menerima notifikasi *inbox* pada aplikasinya dari *customer*
3. Bellcaptain memilih bellboy dari aplikasi
4. Bellboy menerima notifikasi *inbox check out* dari bellcaptain

5. Bellboy menuju kamar untuk mengambil barang *customer*
6. Bellboy kelola (tambah, kurangi) barang tamu jika ada perubahan pada *history* ketika *check in*
7. *Customer* menuju *reception* untuk melakukan *check out*
8. Melakukan *check out* di *reception*
9. Proses *check out customer*
10. *Reception* memberikan *invoice* kepada *customer*
11. *Customer* menerima *invoice* dari *reception*
12. *Customer* memilih layanan ambil barang dari aplikasinya
13. Bellboy mendapat notifikasi *inbox* pada aplikasinya
14. Bellboy membawa barang *customer* keluar lobby
15. *Customer* menerima barang dari bellboy
16. *Customer* di minta untuk memberikan *feedback* pelayanan yang ada pada aplikasi.

K. Perancangan Sistem

Adapun untuk perancangan ini dibuat menggunakan Unified Modelling Language (UML) yang terdiri dari *Use case Diagram*, *Activity Diagram*, *Sequence Diagram* dan *Class Diagram*. Dibawah ini terdapat gambar berikut penjelasannya dapat lihat masing-masing diagramnya.

1. Use case Diagram



Gambar 11. Use case Diagram

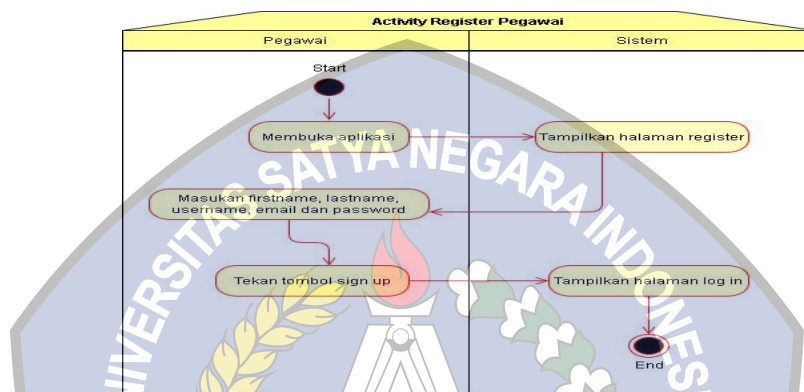
Berdasarkan gambar 11, dapat diuraikan kemampuan sistem rancangan adalah sebagai berikut :

1. Bellboy dapat mengelola data barang ketika mengambil barang di kamar tamu dan menyelesaikan tugas dengan menekan button (*complete*)
2. Bellcaptain dapat mengelola data barang ketika tamu *check-in*
3. Bellcaptain dapat akses menugaskan salah satu bellboy
4. Bellcaptain bisa mengelola nomer kamar dan laporannya.
5. Reception hanya bisa akses untuk input data customer pada saat *check-in*.

6. Admin dapat mengakses mengelola data *staff* dan kelola data *customer*.
7. *Customer* dapat mengakses pilih layanan dan mengisi *feedback* untuk pelayanan hotel.

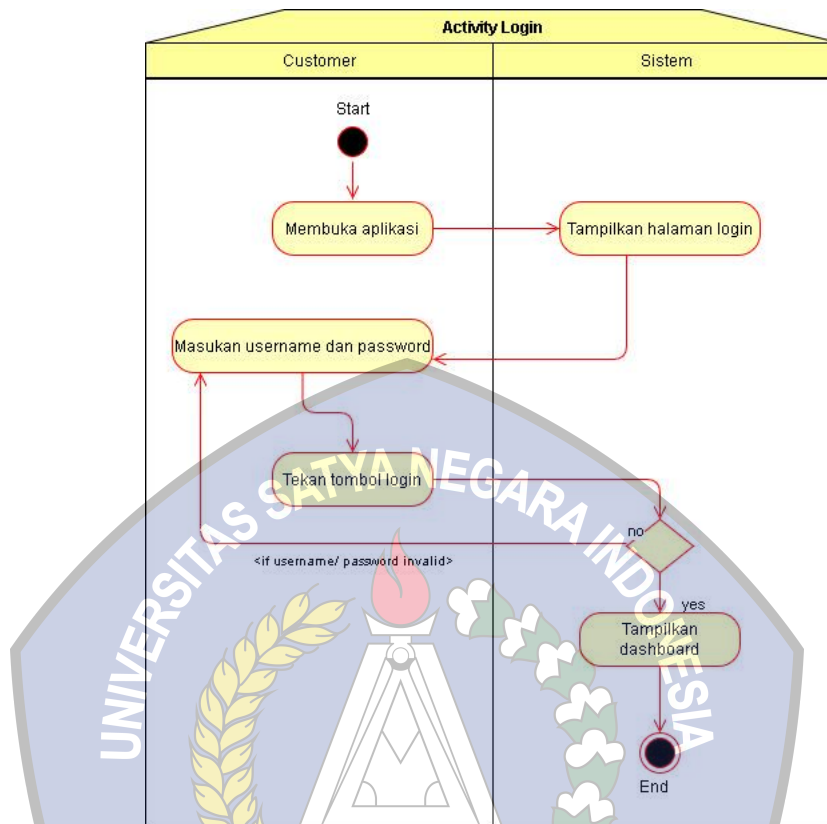
2. Activity Diagram

a. Activity register pegawai



Gambar 12. Activity Diagram Register Pegawai

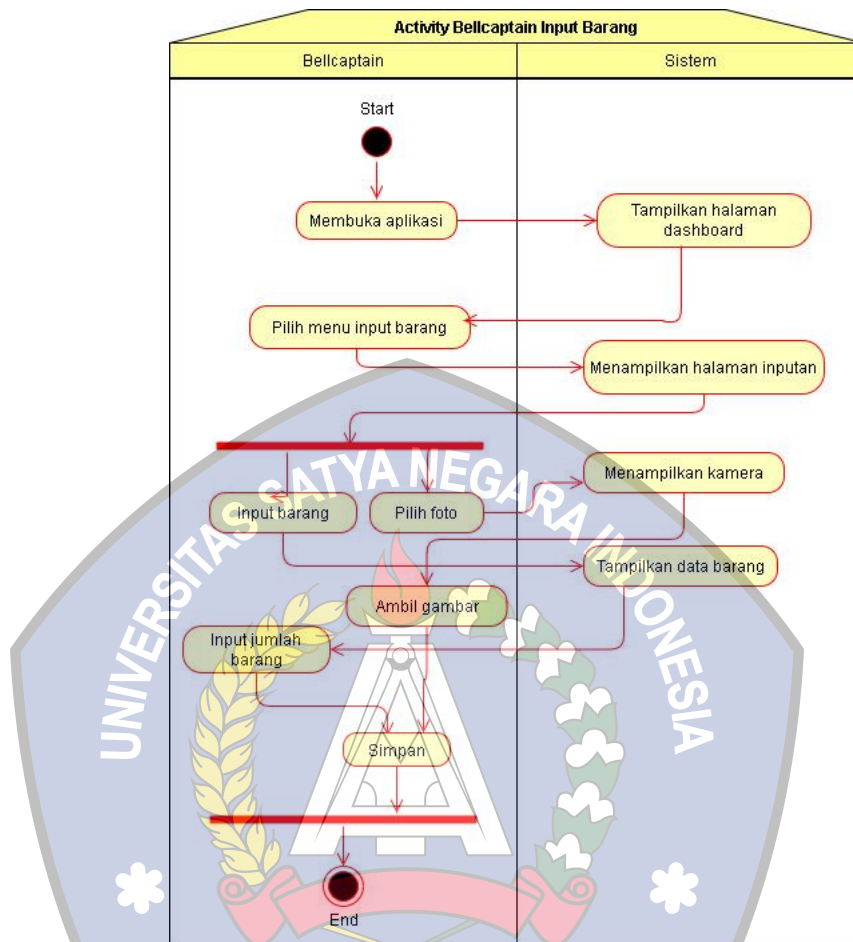
Pada gambar 12. Dijelaskan bagaimana *activity register* pegawai sebelum melakukan login ke aplikasi. Dimana pegawai harus mengisi tiap kolom yaitu *first name*, *last name*, email, *username* dan *password*. Jika sudah selesai maka selanjutnya akan menuju halaman login.

b. *Activity login*

Gambar 13. *Activity Login*

Pada gambar 13. Dijelaskan bagaimana *activity login* berjalan, dimana admin memasukan *username* dan *password*, apabila data sesuai, maka login berhasil dan akan masuk ke halaman *dashboard*, apabila salah memasukan *username* dan *password* maka sistem akan menampilkan pesan *error* dan akan kembali ke halaman login.

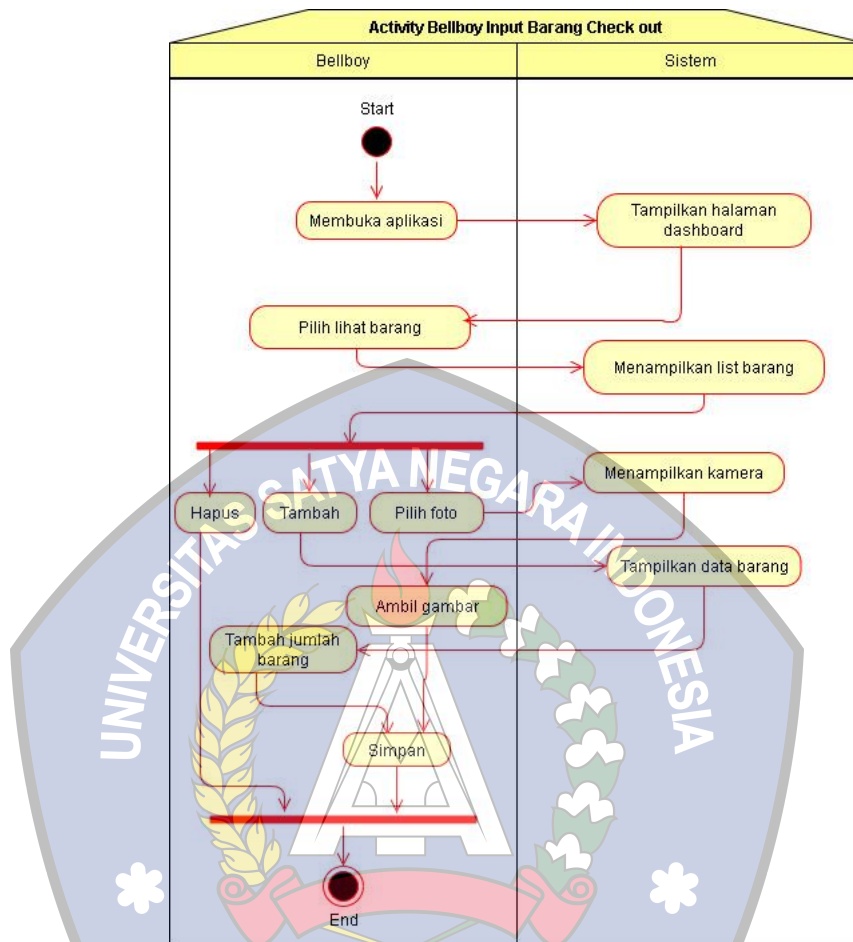
c. *Activity Bellcaptain input barang*



Gambar 14. Activity Diagram Bellcaptain Input Barang

Pada gambar 14. Dijelaskan bahwa bellcaptain dapat mengelola data barang ketika barang masuk ke *store* dengan menggunakan aplikasi yaitu input barang *customer* berdasarkan jenis dan jumlah barang yang dibawa, kemudian mengambil gambar barang tersebut sebagai file pendukung pada laporan transaksi pengiriman barang ke kamar *customer*.

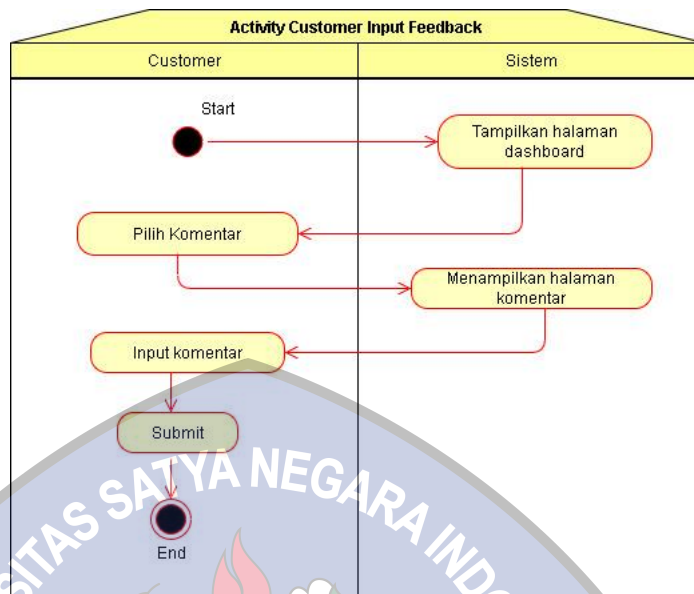
d. *Activity* Bellboy kelola barang pada saat *check out*



Gambar 15. *Activity* Bellboy kelola data barang

Pada gambar 15. Dijelaskan bahwa bellboy dapat melakukan kelola barang *customer* ketika *check out*. Kegiatan yang dilakukan yaitu menambah atau mengurangi jumlah barang sesuai jenis dan jumlah barang, kemudian mengambil gambar sebagai file pendukung dalam laporan pengarsipan hasil kerja dan selanjutnya di simpan dengan menekan *button save* pada aplikasi android.

e. *Activity Customer input komentar*

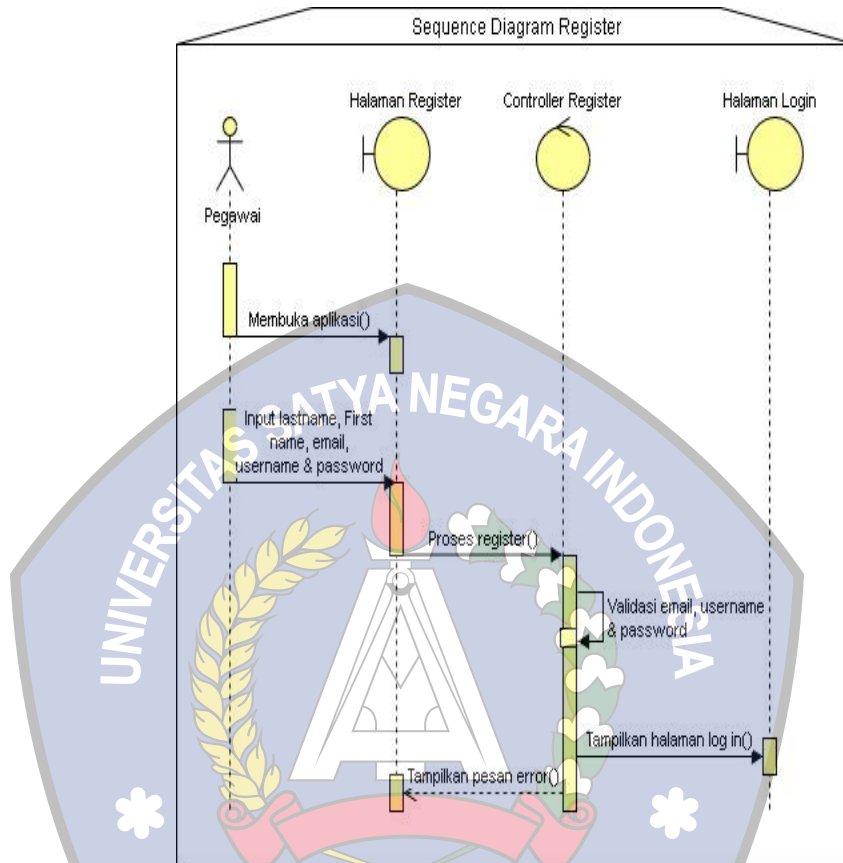


Gambar 16. *Activity customer input komentar*

Pada gambar 16. Dijelaskan bahwa aktivitas *customer* dapat memberikan *feedback* yaitu dengan mengisi kolom komentar yang ada pada aplikasi, komentar tersebut bisa berupa saran maupun kritikan mengenai pelayanan pada *customer* selama menginap setelah selesai mengisi pada kolom komentar kemudian di *submit*.

3. Sequence Diagram

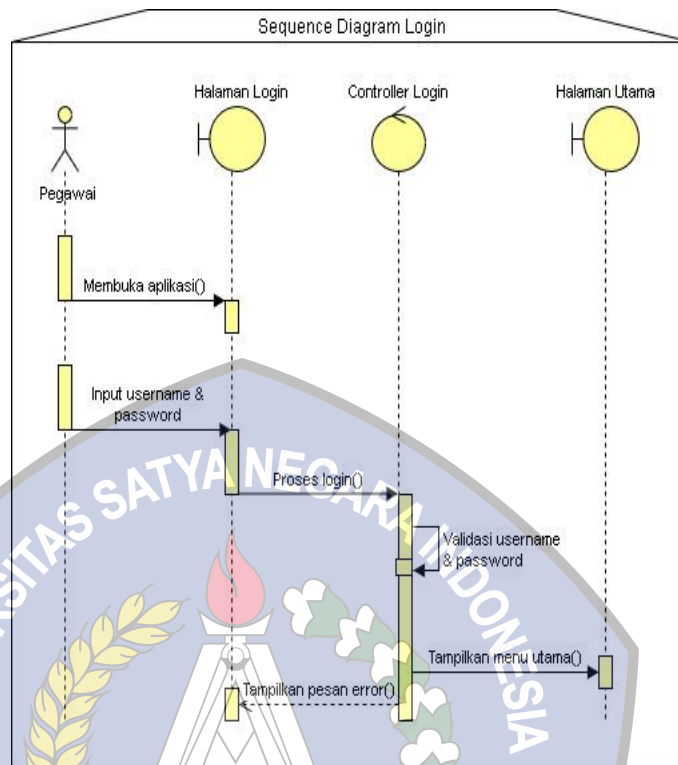
a. Sequence Diagram Register Pegawai



Gambar 17. Sequence diagram register pegawai

Pada gambar 17. Dijelaskan bahwa pegawai membuka aplikasi kemudian pilih menu *sign up* dan sistem menampilkan *form register* dan pegawai mengisi *lastname*, *firstname*, email, *username* dan *password* kemudian akan divalidasi oleh sistem selanjutnya akan menuju halaman *log in*.

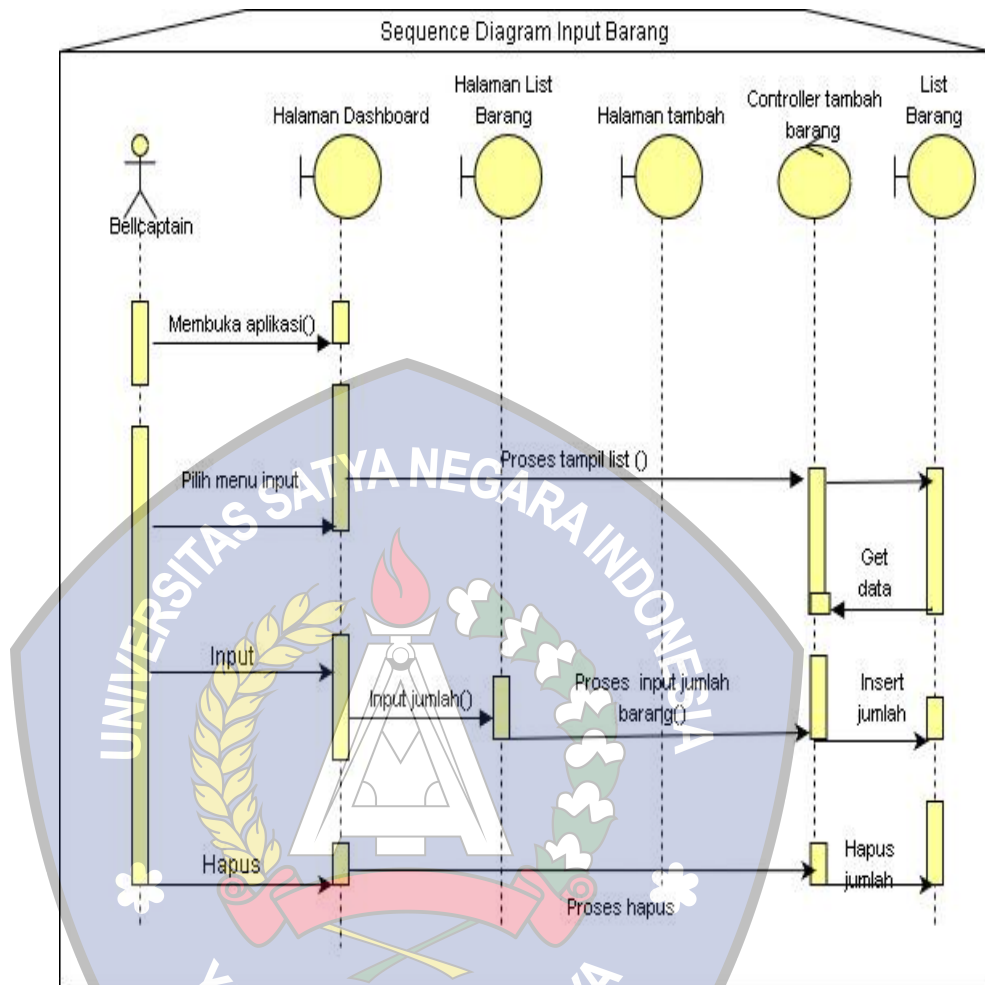
b. *Sequence Diagram Login*



Gambar 18. *Sequence diagram login*

Pada gambar 18. Menjelaskan bahwa pegawai sedang melakukan *login* dengan memasukan *username* dan *password*, sistem memvalidasi *username* dan *password* jika benar maka akan menuju halaman utama dan jika salah sistem akan menampilkan pesan *error* dan mengembalikan ke menu *login*.

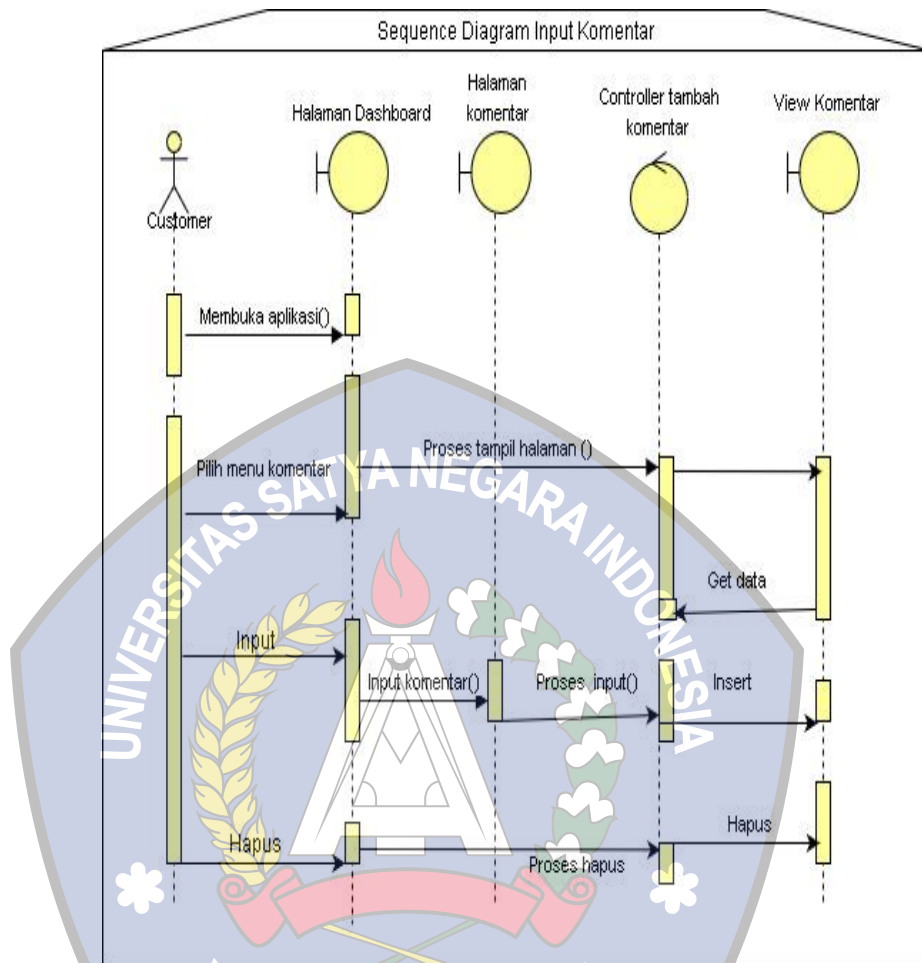
c. *Sequence Diagram List Barang*



Gambar 19. *Sequence diagram list input barang*

Pada gambar 19. Menjelaskan bahwa seorang bellcaptain membuka aplikasi untuk menginput list barang sesuai jenis dan jumlah barang yang dibawa oleh *customer*. Adapun ketika kelebihan jumlah barang bisa di hapus kembali. Aktifitas ini dilakukan ketika *customer check in*.

d. Sequence Diagram Input Komentar

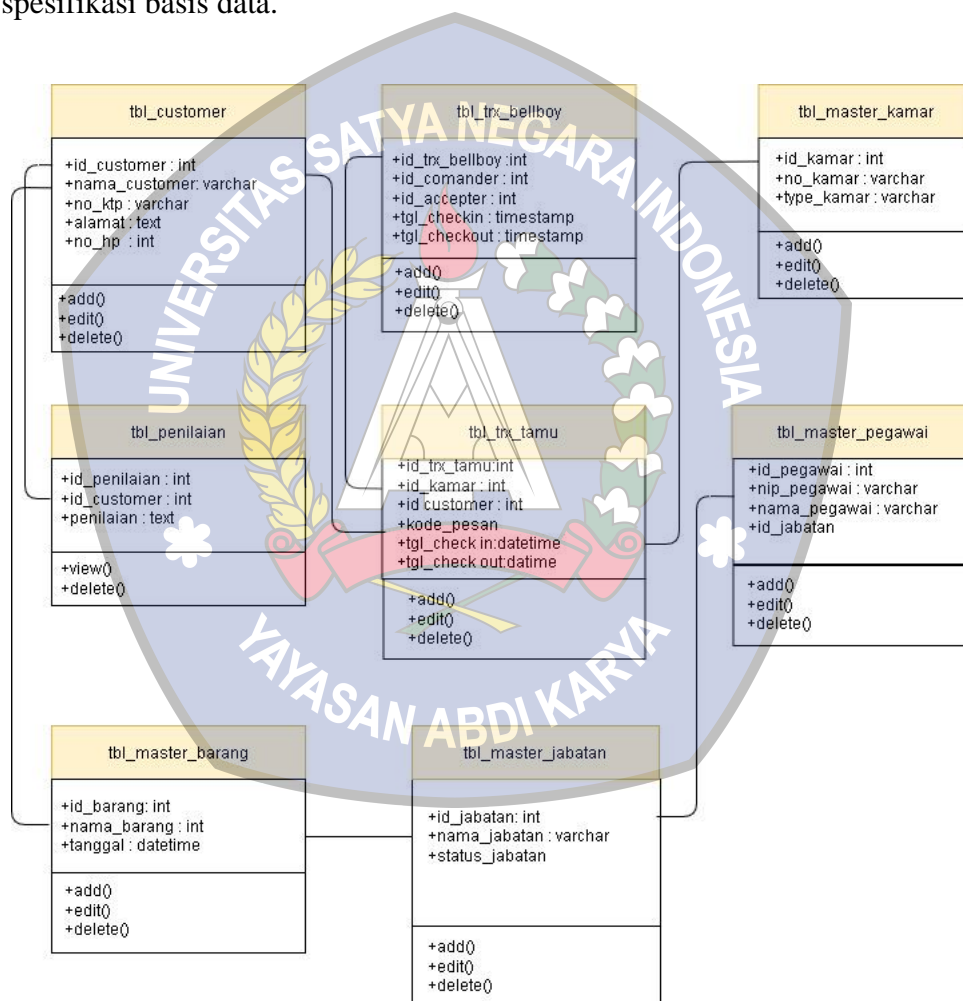


Gambar 20. *Sequence Diagram Input Komentar*

Pada gambar 20. Rancangan *sequence diagram* untuk *customer* mengisi komentar. Dimana ketika customer membuka aplikasi android dan telah selesai menginap maka halaman komentar akan muncul secara otomatis. Pada saat itu customer dapat memberi komentar kepada hotel mengenai pelayanannya selama menginap di Hotel Century Park.

4. Class Diagram

Class Diagram di bagian ini dibuat untuk menjelaskan hubungan antar *class* yang ada pada sistem dan bagaimana masing-masing *class* saling berhubungan. Pada gambar 21 terdapat 8 tabel yang saling berelasi, dan pada setiap tabel memiliki atribut yang dapat dilihat pada sub bab berikutnya yaitu spesifikasi basis data.



Gambar 21. *Class Diagram*

L. Spesifikasi Basis Data

Spesifikasi data merupakan uraian atau rincian tiap relasi antar tabel yang di sesuaikan pada software yang akan di gunakan pada saat implementasinya.

Berikut adalah spesifikasi data pada software yang di usulkan :

1. Nama tabel : master_kamar

Primary key : id_kamar

Foreign key : -

Tabel 3. Desain tabel Master kamar

No	Nama kolom	Type Data	Jenis/ panjang	Default	Kunci
1	Id_kamar	Int	11	Auto increment	Primary key
2	Kode_kamar	varchar	20	None	
3	Type_kamar	varchar	20	None	

2. Nama tabel : master_pegawai

Primary key : id_pegawai

Foreign key : id_jabatan

Tabel 4. Desain tabel Master pegawai

No	Nama kolom	Type Data	Jenis/ panjang	Default	Kunci
1	Id_pegawai	Int	11	Auto increment	Primary key
2	Nip_pegawai	varchar	20	None	
3	Nama_pegawai	varchar	100	None	
4	Alamat_pegawai	Varchar	100	None	
5	Email_pegawai	Varchar	50	None	
6	Password	Varchar	20	none	
7	Id_jabatan	Int	1	Not null	Foreign key

3. Nama tabel : Master_jabatan
 Primary key : id_jabatan
 Foreign key : _

Tabel 5. Desain tabel Master jabatan

No	Nama kolom	Type Data	Jenis/ panjang	Default	Kunci
1	Id_jabatan	Int	11	Auto increment	Primary key
2	Nama_jabatan	varchar	20	None	
3	Status_jabatan	Tinyint	1	None	

4. Nama tabel : customer
 Primary key : id_customer
 Foreign key : _

Tabel 6. Desain tabel customer

No	Nama kolom	Type Data	Jenis/ panjang	Default	Kunci
1	Id_customer	Int	11	Auto increment	Primary key
2	nama	varchar	20	None	
3	No_ktp	varchar	20	None	
4	Alamat	Varchar	100	None	
5	No_hp	Int	12	None	

5. Nama tabel : Master_barang

Primary key : id_barang

Foreign key : _

Tabel 7. Desain tabel master barang

No	Nama kolom	Type Data	Jenis/ panjang	Default	Kunci
1	Id_barang	Int	11	Auto increment	Primary key
2	Nama_barang	varchar	45	None	

6. Nama tabel : transaksi_bellboy

Primery key : id_trx_bellboy

Foreign key : _

Tabel 8. Desain tabel transaksi bellboy

No	Nama kolom	Type Data	Jenis/ panjang	Default	Kunci
1	Id_trx_bellboy	Int	11	Auto increment	Primary key
2	Id_commander	Int	11	None	Foreign key
3	Id_accepter	Int	11	None	
4	Tgl_trx	Date			
5	Status	Int	1		
6	Tgl_trx_mulai	Timestamp		Current_timestamp	
7	Tgl_trx_selesai	Timestamp		Yes , null	

7. Nama tabel : transaksi_tamu

Primary key : id_trx_tamu

Foreign key : -

Tabel 9. Desain tabel transaksi tamu

No	Nama kolom	Type Data	Jenis/ panjang	Default	Kunci
1	Id_trx_tamu	Int	11	Auto increment	Primary key
2	Id_kamar	Int	11	None	Foregin key
3	Id_customer	Int	11	None	Foreign key
4	Nip_pegawai	Varchar	20	None	
5	Kode_pesan	Varchar	20	None	
6	Tgl_checkin	Datetime		None	
7	Tgl_checkout	Datetime		None	

M. Perancangan Antar Muka

Perancangan antar muka atau biasa di sebut dengan *interface*, adalah perancangan yang di buat sebagai *prototype* atau gambaran aplikasi yang akan di buat. Dimana rancangan ini yang akan dilihat pertama kali ketika user membuka aplikasi. Rancangan dibuat sesuai kebutuhan user dengan desain yang simpel dan mudah digunakan oleh user.

a. Rancangan Pada Aplikasi Android

Berikut adalah Rancangan Sistem Informasi Pelayanan Concierge berbasis Android mobile. Desain yang mudah digunakan dan diaplikasikan oleh user agar dapat membantu dan mempermudah dalam pengoperasiannya. Rancangan ini dibuat pada setiap aktifitas sistem yang dibutuhkan sehingga menjadi satu kesatuan sistem yang saling berkaitan dan menghasilkan sebuah aplikasi Sistem Informasi Pelayanan Tamu pada Department Concierge Hotel Century Park.

1. Rancangan register pegawai



Gambar 21. Rancangan *register*

Pada gambar 21. Merupakan rancangan tampilan register yang mana akan muncul ketika seorang pegawai belum melakukan register. Pada halaman ini seorang pegawai harus mengisi tiap kolom yang sudah didesain yaitu nama depan dan nama belakang, username, email pegawai serta passwordnya kemudian konfirmasi passwordnya. Jika sudah terisi semua kolomnya selanjutnya tekan *button Sign Up* maka akan secara otomatis terdaftar pada database sistem android.

2. Rancangan *Log in*

Gambar 22. Rancangan *Log in*

Pada gambar 22. Merupakan rancangan tampilan login, yang mana terdapat *username* dan *password* sebagai syarat untuk bisa masuk ke halaman berikutnya. Pada halaman ini terdapat kolom username dan password. Ketika seorang pegawai sudah melakukan register maka langkah selanjutnya tekan *button Login*.

3. Rancangan List input barang

Form Input Barang

Suit Case	Jumlah Barang
Bag Pack	Jumlah Barang
Hand Bag	Jumlah Barang
Golf Bag	Jumlah Barang
Suit Box	Jumlah Barang
Shoes	Jumlah Barang
Hanger	Jumlah Barang
Travel Bag	Jumlah Barang
Lainnya	Jumlah Barang
Nama Barang	Jumlah Barang

Save

Gambar 23. Rancangan list input barang

Pada gambar 23. Adalah rancangan tampilan untuk bellcaptain input jumlah barang berdasarkan jenisnya. Kemudian jika nama barang tidak ada dilist maka bisa menambahkan nama dan jenis barang serta jumlah barangnya di kolom yang paling bawah. Setelah selesai input kemudian klik *button save* dan data yang telah dimasukan akan tersimpan dalam database sistem.

4. Rancangan List Bellboy



List Bell Boy

Herry	<input type="checkbox"/>
Deny	<input type="checkbox"/>
Bambang	<input type="checkbox"/>
Yudi	<input type="checkbox"/>
Ferei	<input checked="" type="checkbox"/>
Fauzy	<input type="checkbox"/>
Ringgo	<input type="checkbox"/>
Susanto	<input type="checkbox"/>

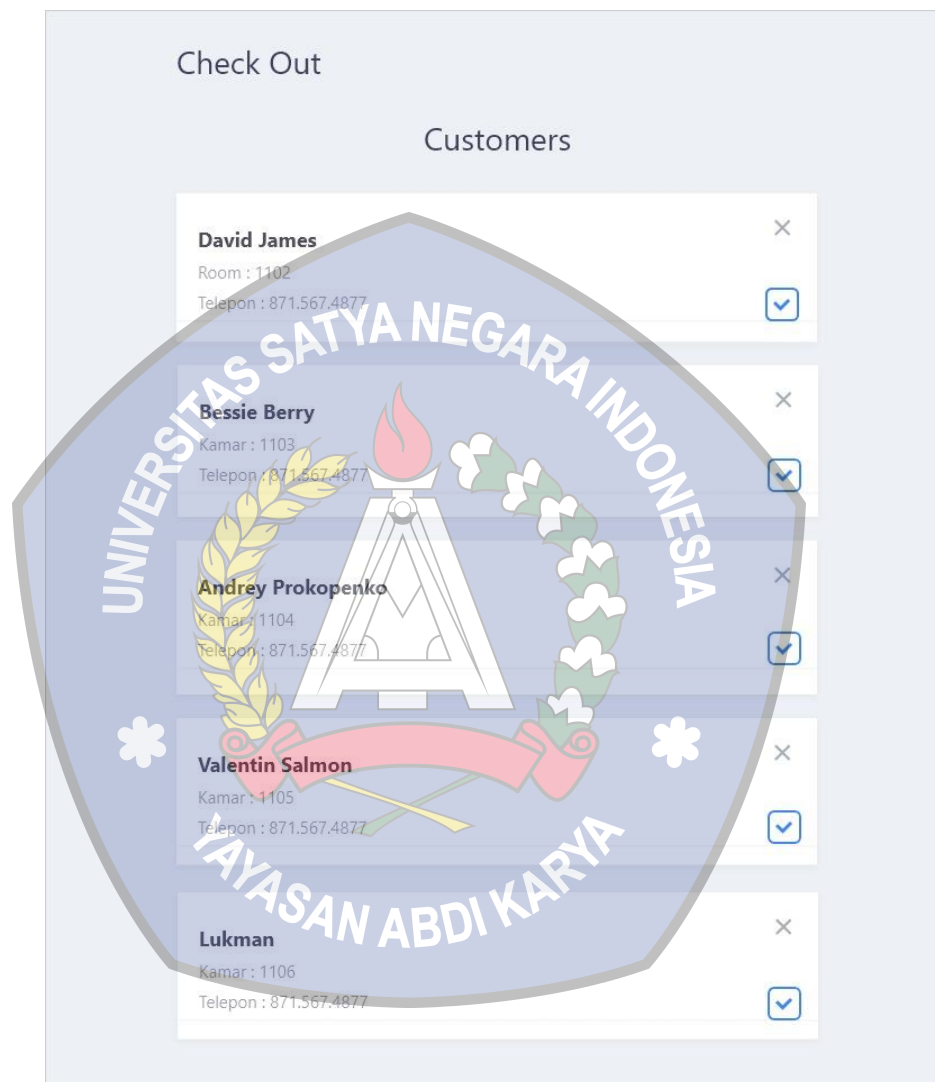
Save

Gambar 24. Rancangan list bellboy

Pada gambar 24. Merupakan rancangan untuk tampilan list bellboy, ini adalah tampilan untuk bellcaptain ketika akan mendelegasikan tugas maka harus di sesuaikan dengan bellboy yang *incharge*. Pada halaman ini terdapat list nama-nama bellboy dan kolom check list yang

digunakan untuk memilih bellboy yang akan mengerjakan tugas dari bellcaptain.

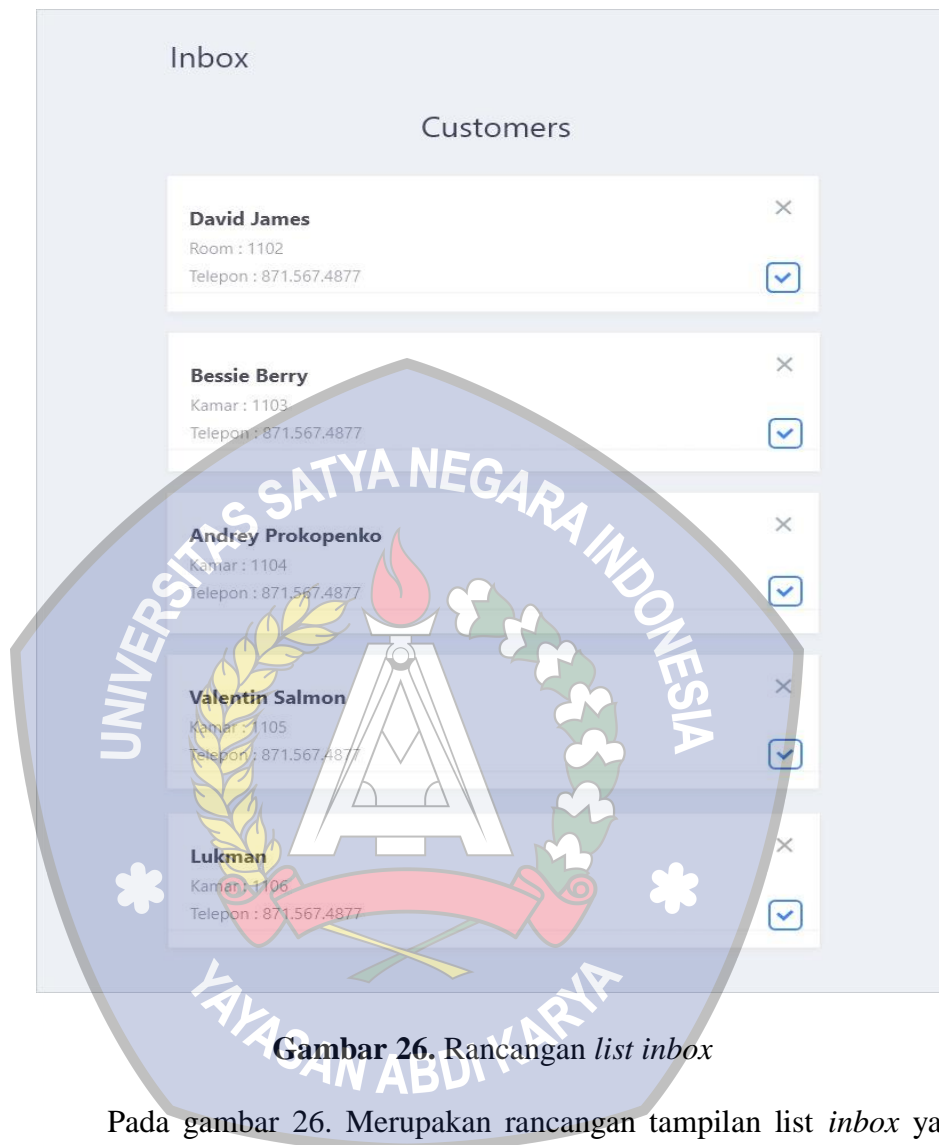
5. Rancangan *Inbox Check out*



Gambar 25. Rancangan list *check out*

Pada gambar 25. Adalah rancangan tampilan list *check out*, berisi nama *customer*, nomer kamar dan nomer telepon. Pada tampilan ini yang akan muncul pada aplikasi pegawai yang sedang bertugas.

6. Rancangan *Inbox Check in*



Gambar 26. Rancangan *list inbox*

Pada gambar 26. Merupakan rancangan tampilan *list inbox* yang berisi nama *customer*, nomer kamar dan nomer telpon. Secara keseluruhan hampir sama dengan *list check out* bedanya halaman ini ada pada *list inboxnya bellcaptain*. Sehingga bisa terlihat semua daftar customer yang akan meminta bantuan ke *concierge*.

7. Rancangan View Detail Barang

The screenshot shows a web form titled "View Detail Barang". It contains a list of items with corresponding input fields for their quantities. The items and their values are: Suitcase (4), Backpack (2), Handbag (2), Golfbag (4), Plastic bag (5), Shoes (1), Hanger (6), and Travelbag (0). A blue "Save" button is located at the bottom right of the form. A large, semi-transparent watermark of the Universitas Satya Negara Indonesia logo is overlaid on the form.

Item	Quantity
Suitcase	4
Backpack	2
Handbag	2
Golfbag	4
Plastic bag	5
Shoes	1
Hanger	6
Travelbag	0

Save

Gambar 27. Rancangan *view* detail barang

Pada gambar 27. Merupakan rancangan tampilan detail list barang hasil inputan dari bellcaptain dan inputan bellboy ketika *check out*. Berisi nama barang dan kolom jumlah barang kemudian *button* simpan.

8. Rancangan Form komentar



The image shows a web form for leaving a comment. At the top, there is a red stylized 'C' logo followed by the text 'CENTURY PARK Hotel'. Below this, a message reads 'Terimakasih Atas Kunjungannya'. The main part of the form consists of a large white rectangular input area with the placeholder text 'Input Comment'. To the right of the input area is a blue button labeled 'Submit'. A large, semi-transparent watermark of the Universitas Satya Negara Indonesia logo is overlaid on the form. The logo is a shield-shaped emblem with a torch in the center, surrounded by a wreath, and the text 'UNIVERSITAS SATYA NEGARA INDONESIA' and 'YAYASAN ABDI KARYA' around the perimeter.

Gambar 28. Rancangan halaman komentar

Pada gambar 28. Rancangan halaman komentar *customer* ketika selesai menginap. Halaman ini bagian dari aplikasi *customer* yang digunakan *customer* untuk memberikan feedback pelayanan hotel yang nantinya dapat memberikan masukan guna meningkatkan kualitas pelayanan nantinya.

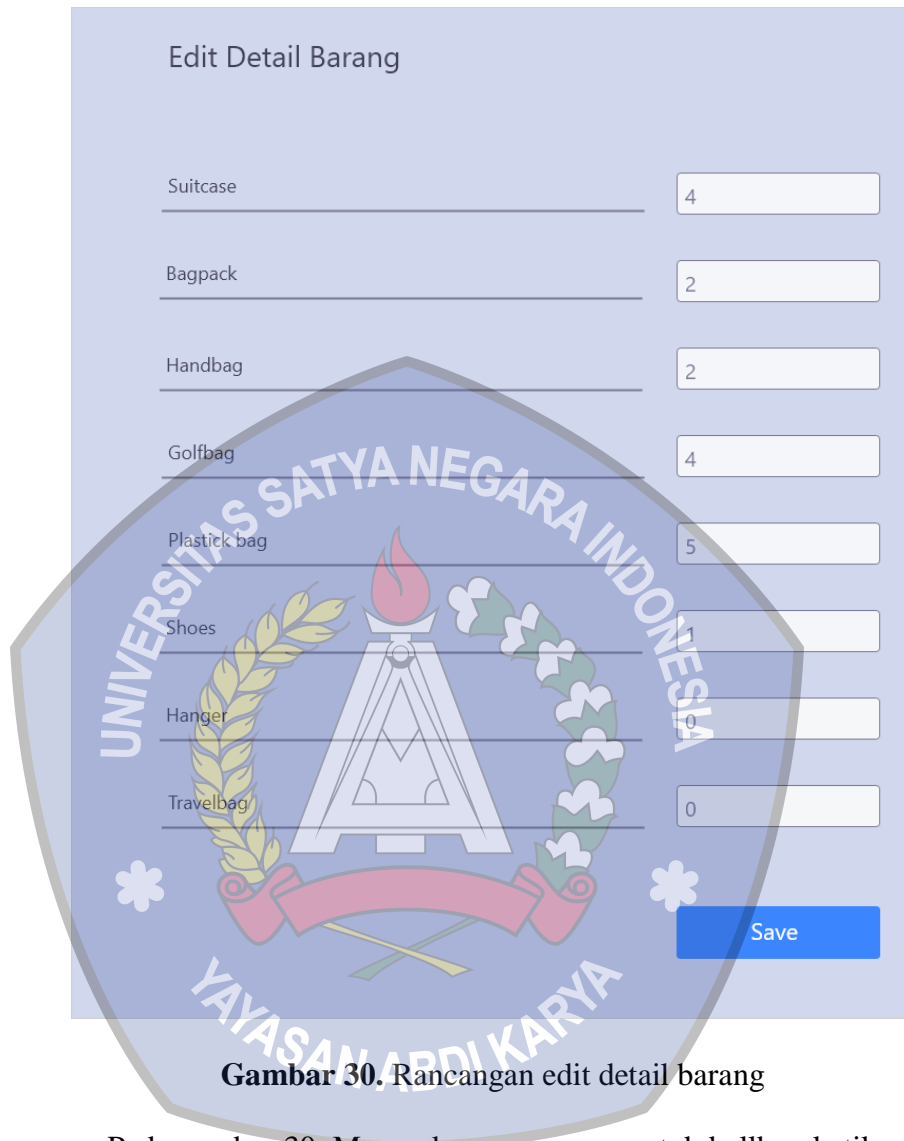
9. Rancangan Pilihan layanan



Gambar 29. Rancangan pilihan layanan

Pada gambar 29. Adalah rancangan halaman untuk *customer* ketika selesai *log in* kemudian dapat memilih antara layanan *check in* atau *check out*. Pada halaman ini juga terdapat informasi detail *customer* dan ucapan selamat datang.

10. Rancangan Edit Detail Barang



Item	Quantity
Suitcase	4
Backpack	2
Handbag	2
Golfbag	4
Plastic bag	5
Shoes	
Hanger	
Travelbag	0

Save

Gambar 30. Rancangan edit detail barang

Pada gambar 30. Merupakan rancangan untuk bellboy ketika ambil barang di kamar, maka halaman edit detail barang ini untuk menambah dan mengurangi jumlah barang pada list nya. Kemudian tekan tombol *save* untuk menyimpan data yang telah diinput pada sistem android.

b. Rancangan Pada Aplikasi Website

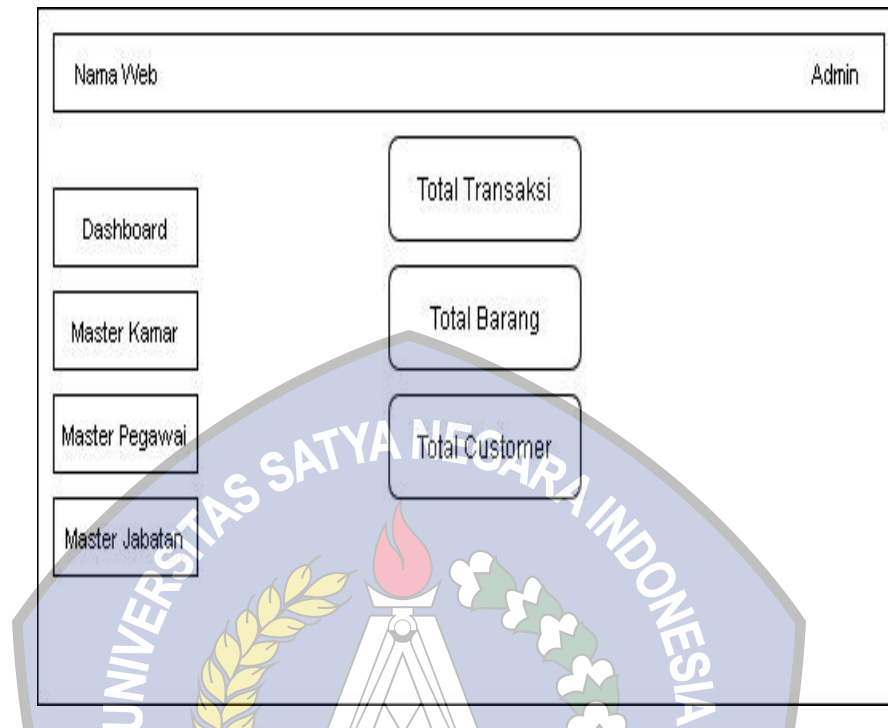
Berikut adalah rancangan *web desktop* sebagai pendukung aplikasi androidnya

1. Rancangan halaman *Log in*

Gambar 31. Rancangan halaman *Log in Web*

Pada gambar 31. Adalah rancangan halaman *log in* pada web aplikasi hotel century. Terdapat inputan username dan password. Disebelah kanan terdapat logo hotel century. Halaman *logi in* merupakan tampilan awal sebelum masuk ke halaman *dashboard*.

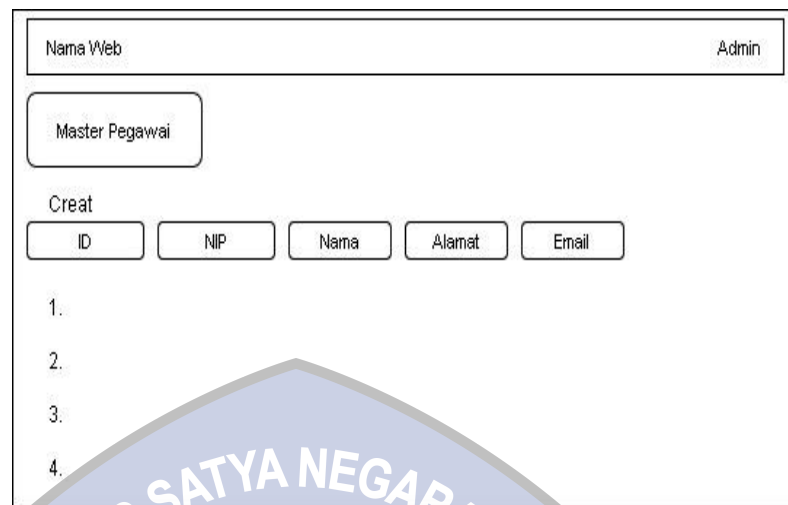
2. Rancangan halaman *Dashboard*



Gambar 32. Rancangan halaman *Dashboard*

Pada gambar 32. Merupakan tampilan awal setelah pegawai melakukan *log in*, terdapat informasi total transaksi, informasi total barang, dan informasi total *customer*. Dan disebelah kiri halaman terdapat informasi master kamar, master pegawai, dan master jabatan.

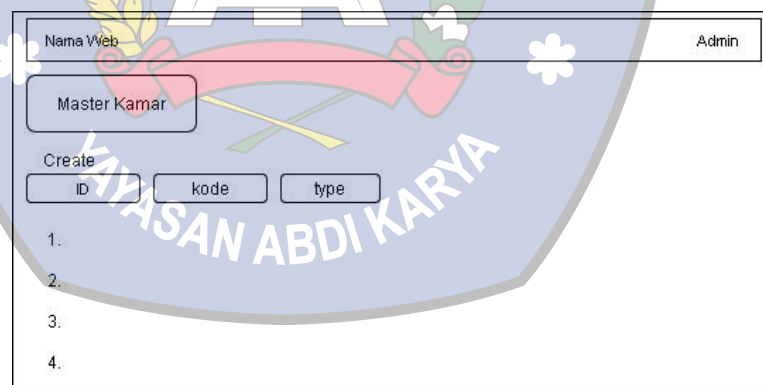
3. Rancangan layar Master Pegawai



Gambar 33. Rancangan Layar Master Pegawai

Pada gambar 33. Merupakan rancangan tampilan layar master pegawai untuk memasukkan pegawai baru oleh admin.

4. Rancangan layar Master Kamar



Gambar 34. Rancangan tampilan layar Master Kamar

Pada gambar 34. Adalah rancangan tampilan master kamar berisi kode kamar dan tipe kamar yang ada di Hotel Century Park.

5. Rancangan Halaman Bellboy Master

Gambar 35. Rancangan Halaman Bellboy Master

Pada gambar 35. Merupakan rancangan layar input untuk membuat nama bellboy baru pada admin. Diantaranya input id bellboy, nama dan nip bellboy.

6. Rancangan Halaman Transaksi Bellboy

Gambar 36. Rancangan Tampilan Transaksi Bellboy

Pada gambar 36. Adalah rancangan transaksi bellboy yang telah melakukan tugasnya yang ada pada tampilan admin.

BAB IV

IMPLEMENTASI DAN EVALUASI

A. Hasil dan Pembahasan

Berikut adalah implementasi hasil dari sistem informasi pelayanan tamu pada department concierge di Hotel Century Park

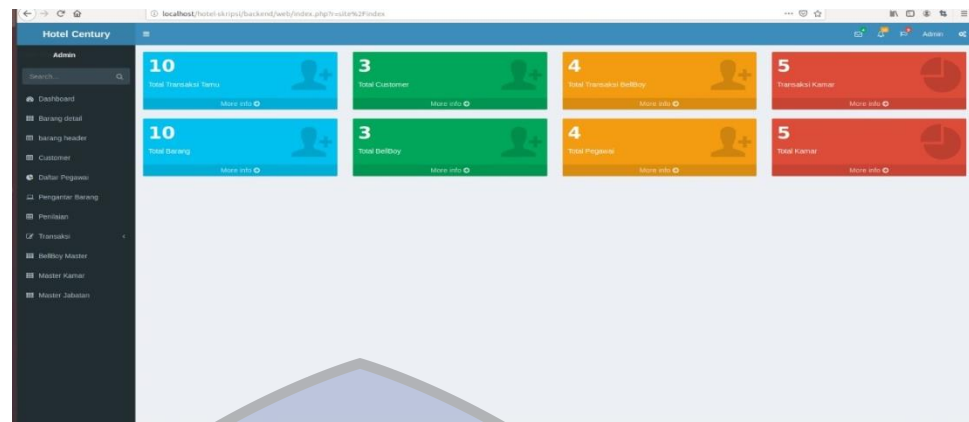
1. Tampil Menu Login



Gambar 37. Tampilan Menu *Log in*

Pada gambar 37, merupakan implementasi halaman *log in* aplikasi *web* sistem informasi untuk admin. Halaman *log in* merupakan tampilan awal yang digunakan oleh bagian Admin sebelum masuk ke halaman dashboard.

2. Tampilan Halaman *Dashboard*



Gambar 38. Tampilan Menu *Dashboard*

Pada gambar 38, merupakan hasil implementasi menu *dashboard*, yaitu tampilan utama pada aplikasi di admin . dalam *dashboard* merupakan halaman yang menampilkan total dari transaksi bellboy, total customer, dan total barang.

3. Tampilan Master Pegawai



Gambar 39. Tampilan Menu Master Pegawai

Pada gambar 39, merupakan implementasi dari menu inputan master pegawai dimana di dalamnya terdapat informasi id pegawai, nip, nama, alamat dan email pegawai.

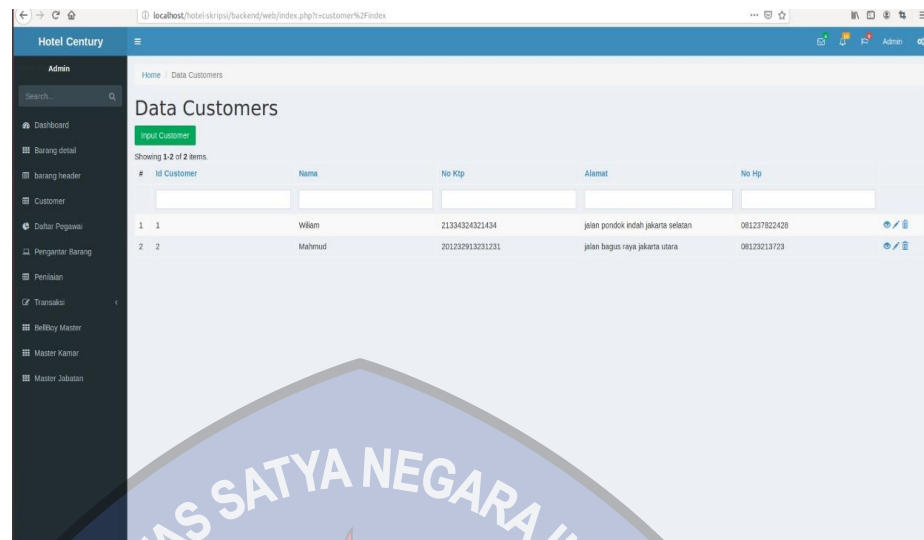
4. Tampilan Menu Bellboy Master



Gambar 40. Tampilan Menu Bellboy Master

Pada gambar 40, merupakan implementasi dari menu bellboy master yang di dalamnya terdapat informasi nama-nama bellboy, id dan nip bellboy.

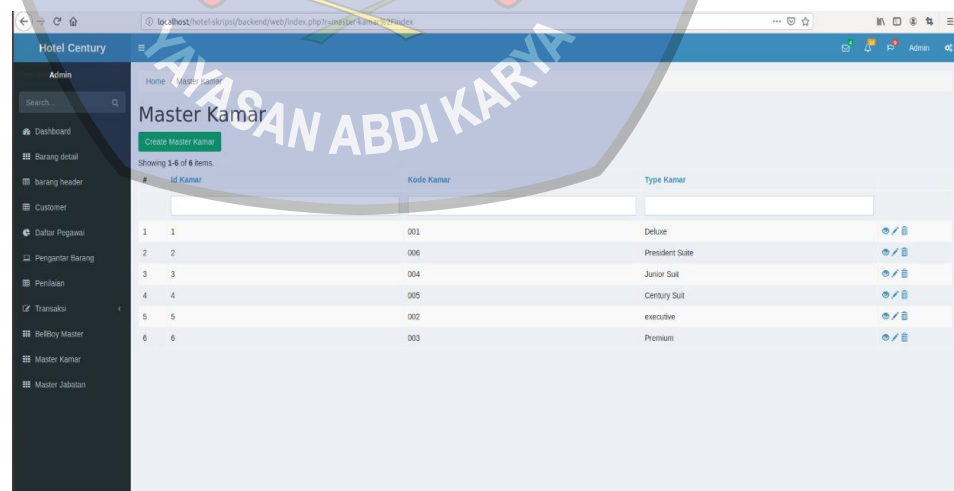
5. Tampilan Menu Input Customer



Gambar 41. Tampilan Menu Data Customer

Pada gambar 41, merupakan halaman input untuk customer baru, dimana ada beberapa informasi yang ada yaitu id customer, nama, no ktp, alamat dan no tlp customer.

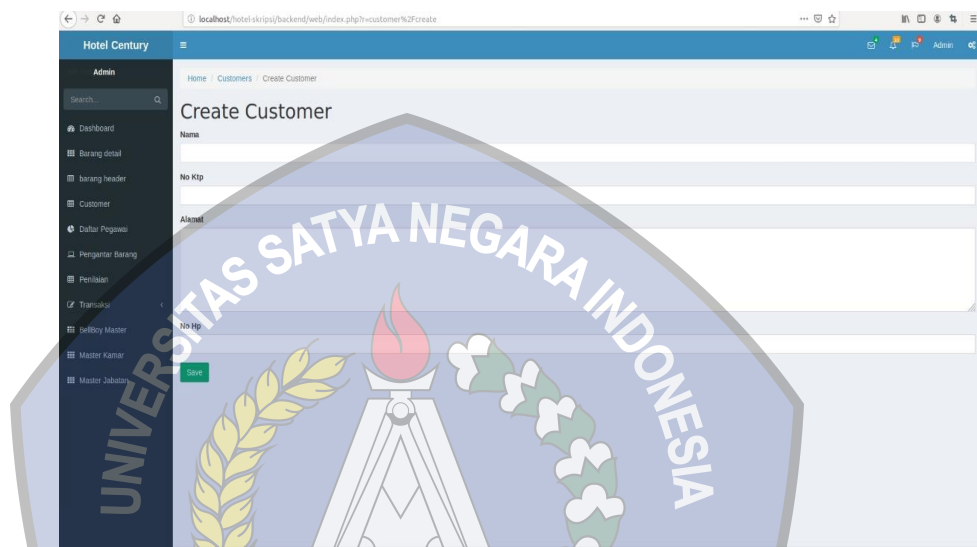
6. Tampilan Menu Master Kamar



Gambar 42. Tampilan Menu Master Kamar

Pada gambar 42, merupakan hasil implementasi menu master kamar yang ada di aplikasi admin, dimana ada beberapa informasi yaitu id kamar, type kamar dan kode kamar.

7. Tampilan Menu Create Customer



Gambar 43. Tampilan Menu *Create Customer*

Pada gambar 43, merupakan hasil implementasi menu *create customer* pada admin, dengan memasukkan data *customer* dari nama, no ktp, alamat, dan no hp *customer* setelah itu klik *save* dan akan tersimpan di database.

B. Evaluasi Sistem

Pada tahap evaluasi system, dibagi menjadi 2 yaitu evaluasi hasil uji coba sistem / *Software Testing* dan analisa hasil uji coba sistem / *Software Testing*. Hasil uji coba dilakukan kembali untuk menguji semua tahapan dan fitur dari aplikasi tersebut, untuk mengetahui apakah akan terjadi *error*, ketika melakukan uji coba pada aplikasi. Uji coba dilakukan pada setiap test *case* dari implementasi sistem, sehingga akan di dapatkan hasil evaluasi sistem yang sudah dilakukan uji coba sistem / *Software Testing* pada aplikasi ini. Hasil tes dan evaluasi dilakukan oleh responden yaitu saudara Johan yang melakukan pengujian dengan menggunakan metode BlackBox. Untuk hasil pengujian dapat dilihat pada tabel 10, dibawah ini.

Tabel 10. *Software Testing* menggunakan metode *BlackBox*

No	Test Case	Precondition	Test-Steps	Expected Result	Actual Result
1	Login	Pengguna harus membuka Xampp terlebih dahulu, dan klik action pada Apache dan MySQL	1-Mengakses alamat aplikasi: localhost/hotel-skripsi/index.php?r=Login 2-Masukan username : admin password : admin123 3-Klik tombol Login	Pengguna berhasil Login dan masuk ke menu dashboard admin	Berhasil [✓] Gagal []
2	Cari Data pegawai dan	1-Pengguna harus <i>Login</i> terlebih	1-pada menu dashboard pilih menu “daftar pegawai, untuk	Pengguna berhasil masuk ke	Berhasil [✓]

	menambah data pegawai	dahulu	menampilkan daftar nama pegawai untuk menampilkan semua daftar pegawai 2-klik “create master pegawai” Untuk menambah nama pegawai baru 3-klik “save” untuk menyimpan	menu daftar pegawai dan melakukan penambahan nama pegawai	Gagal []
3	Cari data customer	1-Pengguna harus Log in terlebih dahulu 2-pengguna telah memasukkan data customer	1-Pilih menu customer 2-Ketik nama customer dan tekan enter 3-Klik “print” untuk mencetak data customer	Pengguna berhasil menampilkan data customer dan mencetak data customer	Berhasil [✓] Gagal []
4	Input daftar barang	1-Pengguna harus Log in terlebih dahulu 2-Pengguna telah masuk pada halaman dashboard	1-Pilih menu “create barang” 2-masukan nama barang dan jumlahnya 3-klik “save” untuk menyimpan	Pengguna berhasil menginput nama barang	Berhasil [✓] Gagal []
5	Log in halaman reception	Pengguna harus membuka	1-Mengakses alamat aplikasi: localhost/hotel-	Pengguna berhasil Login dan	Berhasil [✓]

		Xampp terlebih dahulu, dan klik action pada Apache dan MySQL	skripsi/index.php?r=Login 2-Masukan username : reception password : 123456 3-Klik tombol Login	masuk ke menu dashboard reception	Gagal []
6	Input nama jabatan	1-Pengguna harus Log in terlebih dahulu 2-Pengguna telah masuk pada halaman dashboard	1-Pilih menu “master jabatan” 2-Pilih menu”create” untuk memasukan nama jabatan 3-Pilih “save” untuk menyimpan nama jabatan baru	Pengguna berhasil menambah nama jabatan dan menyimpan data jabatan baru	Berhasil [✓] Gagal []
7	Tambah data customer	1-Pengguna harus Log in terlebih dahulu 2-Pengguna telah masuk pada halaman dashboard	1-Pilih menu “customer” 2-Pilih menu”create” untuk membuat nama customer baru, masukan nama, alamat, no ktp, no telpon. 3-Pilih “save” untuk menyimpan nama customer baru	Pengguna berhasil menambah nama customer dan menyimpan data customer baru	Berhasil [✓] Gagal []
8	Melihat type kamar	1-Pengguna harus Log in terlebih dahulu 2-Pengguna telah masuk	1-Pilih menu “master kamar” 2-Pilih menu”view” untuk melihat detail nama type kamar	Pengguna berhasil melihat detail nama type kamar dan update	Berhasil [✓] Gagal []

		pada dashboard		nama kamar detail	
9	<u>Hapus data barang</u>	1-Pengguna harus Log in terlebih dahulu 2-Pengguna telah masuk pada dashboard	1-Pilih menu “master barang” 2-Pilih nama barang yang akan di hapus 3-Pilih tanda “hapus” untuk mengurangi data barang	Pengguna berhasil menghapus detail nama barang	Berhasil [✓] Gagal []
10	Melihat daftar Nilai	1-Pengguna harus Log in terlebih dahulu 2-Pengguna telah masuk pada dashboard admin	1-Pilih menu “nilai” pada dashboard 2-Pilih “view” yang akan di lihat	Pengguna berhasil melihat daftar nilai dari customer	Berhasil [✓] Gagal []
11	Log in aplikasi android	1-Pastikan aplikasi terhubung dengan internet 2-Pengguna sudah memiliki aplikasi android	1-Buka aplikasi android 2-masukan username ” email” 3-Masukan “password”	Pengguna berhasil Login dan masuk ke aplikasi android	Berhasil [✓] Gagal []
12	Register di	1-Pastikan	1-Buka aplikasi	Pengguna	

	Aplikasi android	aplikasi terhubung dengan internet 2-Pengguna sudah memiliki aplikasi android	android 2-masukan "Firstname" 3-Masukan "lastname" 4-Masukan "email" 5-Masukan "password" 6-Masukan "Confirm password" 7-Klik <i>button Register</i>	berhasil register dan masuk ke halaman <i>login</i>	Berhasil [✓] Gagal []
13	Input Barang	1-Pastikan aplikasi terhubung dengan internet 2-Pengguna sudah memiliki aplikasi android	1-Pengguna buka aplikasi 2-masuk menu input barang 3-masukan jumlah barang sesuai nama barang 4-ketik manual jika tidak ada di list nama barang	Pengguna berhasil masuk ke input barang	Berhasil [✓] Gagal []
14	Tampilkan list nama Bellboy	1-Pastikan aplikasi terhubung dengan internet 2-Pengguna sudah memiliki aplikasi android	1-Pengguna buka aplikasi 2-masuk menu list bellboy 3-klik " <i>view</i> "	Pengguna berhasil masuk ke halaman list bellboy	Berhasil [✓] Gagal []

BAB V

KESIMPULAN DAN SARAN

Berdasarkan analisa yang telah dilakukan, maka penulis dapat menarik kesimpulan dan memberikan saran yang dapat dipergunakan untuk pengembangan sistem yang telah di hasilkan dari penelitian ini.

A. Kesimpulan

Kesimpulan dari penelitian ini adalah berhasil merancang aplikasi dan menerapkan sistem pelayanan tamu pada Department Concierge di Hotel Century Park, serta dapat memberikan rekomendasi kepada Manajemen maupun pengusaha untuk dapat menggunakan sistem informasi pelayanan tamu ini sehingga dapat meningkatkan kualitas pelayanan tamu yang menginap di Hotel Century Park. Dan diharapkan dengan menerapkan sistem informasi pelayanan tamu dapat meningkatkan keuntungan (*revenue*) hotel maupun pengusaha.

B. Saran

Untuk kemajuan sistem yang digunakan dimasa mendatang, maka penulis memiliki saran agar penggunaan sistem informasi ini sebagai bagian dari SOP, karena alasan keamanan data barang maupun *customer* lebih terjaga dengan adanya pengarsipan yang sudah terintegrasi dengan database sistem.

DAFTAR PUSTAKA

Asropudin. 2013. *Kamus Teknologi Informasi Komunikasi*. Bandung: CV Titian.

Arief, M.Rudianto. 2011. *Pemrograman Web Dinamis Menggunakan Php dan Mysql*.

Yogyakarta : CV ANDI OFFSET.

Budi Raharjo, Imam Heryanto, dan Arif Haryono, 2010, *Tuntunan Pemrograman*

Java untuk Handphone dan Alat Telekomunikasi Mobile. Bandung. Penerbit

Informatika.

Hansun, Seng, dkk. 2018. *Pemrograman Android dengan Android Studio IDE*.

Yogyakarta: Andi.

Haryanti, Tuti. 2017. *Perancangan Sistem Informasi Pelayanan Hotel Menggunakan*

Metode Waterfall. Karawang.

Kustiyaningsih, Yeni. 2011. *Pemrograman Basis Data berbasis Web Menggunakan*

PHP dan MySQL. Yogyakarta: Graha Ilmu.

Nugroho, Adi. 2014, *Rekayasa Perangkat Lunak Menggunakan UML dan Java*.

Yogyakarta: Andi Publisher.

Nuzuliarini Nuris dan Eka Rini Yulia, 2018. *Rancang bangun sistem pelayanan*

tamu pada hotel Rensa Jakarta. Jakarta.

Prakoso, Aji. (2017). *Front Office Praktis.* Yogyakarta: Gava Media. Ruslan,

Rosady.

Safaat H. Nazaruddin. 2011. *Android Pemrograman Aplikasi Mobile Smartphone*

dan Tablet PC Berbasis Android. Informatika. Bandung. Rosa A.S, M.

Suprpto, F. 2018. *Rekayasa Perangkat Lunak.* Jakarta: Lentera Ilmu Cendekia.

Tohari, Hamim. 2014. *Analisis Serta Perancangan Sistem Informasi Melalui.*

Pendekatan UML. Andi Offset , Yogyakarta. Rexnowati. 2015.

Yudhanto, Yudha, dan Ardhi Wijayanto. 2017. *Mudah Membuat dan Berbisnis*

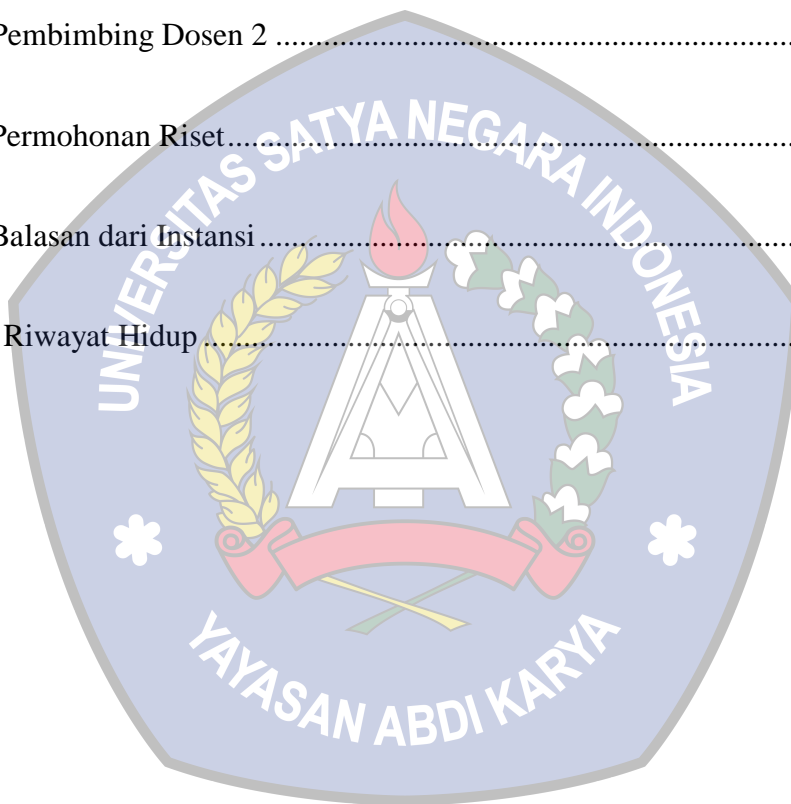
Aplikasi Android dengan Android Studio. Jakarta: Elex Media

Komputindo.

Wicaksono, SR. 2017. *Rekayasa Perangkat Lunak.* Malang: Seribu Bintang

DAFTAR LAMPIRAN

Lampiran Koding	L1
Form Pembimbing Dosen 1	L10
Form Pembimbing Dosen 2	L11
Surat Permohonan Riset.....	L12
Surat Balasan dari Instansi	L13
Daftar Riwayat Hidup	L14



LAMPIRAN KODING

```

<!-- Logo Century -->

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

<androidx.appcompat.widget.LinearLayoutCompat
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center">

    xmlns:app="http://schemas.android.com/apk/res-auto"
    <androidx.appcompat.widget.AppCompatImageView
        xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="120dp"
        android:layout_height="120dp"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="center"
        android:gravity="center"
        tools:context=".LoginActivity">
        <LinearLayout
            android:layout_width="match_parent"

            android:layout_height="match_parent"
            android:gravity="center"
            android:orientation="vertical">

            <!-- Tutup untuk Logo Century -->

            <!-- Label Login Pesan Welcome -->

```

<LinearLayout	
	android:layout_width="match_parent"
android:layout_width="match_parent"	android:layout_height="wrap_content"
android:layout_height="wrap_content"	android:layout_margin="20dp"
android:layout_margin="5dp"	android:gravity="center">
android:gravity="center">	
<androidx.appcompat.widget.AppCompatTextView	<androidx.appcompat.widget.AppCompatEditText
android:layout_width="wrap_content"	android:id="@+id/ed_email_id"
android:layout_height="wrap_content"	android:layout_width="match_parent"
android:layout_gravity="center"	android:layout_height="wrap_content"
android:src="@mipmap/logo_century_park"	android:layout_gravity="center"
	android:hint="Email"
android:text="@string/label_pesan_login"	android:src="@mipmap/logo_century_park"
android:textSize="14sp"></androidx.appcompat.widget.AppCompatTextView>	android:textColorHint="#3D3C3C"
	android:textSize="14sp"
</LinearLayout>	>
<!-- -->	</androidx.appcompat.widget.AppCompatEditText>
<LinearLayout	

```

</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:gravity="center">
    <androidx.appcompat.widget.AppCompatEditText
        android:id="@+id/ed_passwrod_id"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:hint="Password"
        android:inputType="textPassword"
        android:src="@mipmap/logo_century_park"
        android:textColorHint="#3D3C3C"
        android:textSize="14sp"
    >
    </androidx.appcompat.widget.AppCompatEditText>
    <androidx.appcompat.widget.AppCompatTextView
        android:id="@+id/tv_forgot_id"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:gravity="right"
        android:text="Forgot Password ?"
        android:textColorHint="#3D3C3C"
        android:textSize="12sp">

```

```

</androidx.appcompat.widget.AppCo
mpatTextView>

</LinearLayout>

<LinearLayout


android:layout_width="match_parent"

android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:gravity="center"
    android:orientation="horizontal">
<androidx.appcompat.widget.AppCom
patButton
    android:layout_width="0dp"
    android:id="@+id/button_login_id"
    android:layout_margin="10dp"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:gravity="center"
    android:background="@color/colorBu
ttonSecondary"
    android:text="SIGN Up"
    android:layout_margin="10dp"
    android:textColor="#3D3C3C"
    android:textSize="12sp">

    android:gravity="center"
    android:text="LOGIN"
    android:textColor="#ffffff"

</androidx.appcompat.widget.AppCo
mpatButton>
    android:background="@color/colorBu
ttonPrimary"
    android:layout_weight="0.5"
    android:textSize="12sp"/>
<androidx.appcompat.widget.AppCom
patButton
    android:id="@+id/button_signup_id"
    android:layout_width="0dp"
    android:layout_weight="0.5"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:gravity="center"
    android:background="@color/colorBu
ttonSecondary"
    android:text="SIGN Up"
    android:layout_margin="10dp"
    android:textColor="#3D3C3C"
    android:textSize="12sp">

```



```

</LinearLayout>

</LinearLayout>

</androidx.appcompat.widget.LinearLayoutCompat>

package com.example.century;

import
androidx.appcompat.app.AppCompatActivity;

import
androidx.appcompat.widget.AppCompat
  atButton;

import
androidx.appcompat.widget.AppComp
  atEditText;

import
androidx.appcompat.widget.AppComp
  atTextView;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.Toast;

import
com.example.century.API.ApiInterfac
  e;

import
com.example.century.API.RetrofitClie
  nt;

import
com.example.century.GSON.GSONRe
  sponseLogin;

import retrofit2.Call;

import retrofit2.Callback;

import retrofit2.Response;

public class LoginActivity extends
  AppCompatActivity {

    AppCompatEditText edEmail,
    edPassword = null;

    AppCompatButton btnLogin,
    btnSignup = null;

    AppCompatTextView tvForgot =
    null;

    ApiInterface apiInterface=null;

    @Override

    protected void onCreate(Bundle
    savedInstanceState) {

```

```

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_login);

declareComponent();

btnLogin.setOnClickListener(new
View.OnClickListener() {

    @Override
    public void onClick(View v) {

        // Rest api di proses di sini
        beserta inputan email dan password

        Log.d("test_email",
edEmail.getText().toString().trim());

        Log.d("test_password",
edPassword.getText().toString());

startRequestLogin(edEmail.getText().t
oString().trim(),
edPassword.getText().toString());

    }

});

private void declareComponent(){

        edEmail =
        findViewById(R.id.ed_email_id);

        edPassword=
        findViewById(R.id.ed_passwrod_id);

        btnLogin =
        findViewById(R.id.button_login_id);

        btnSignup =
        findViewById(R.id.button_login_id);

        tvForgot =
        findViewById(R.id.tv_forgot_id);

        apiInterface =
        RetrofitClient.getClient().create(ApiInt
erface.class);

        private void
startRequestLogin(String email, String
password){

            if(email==null){

                showMessage("Email belum
diisi");

            }

            else if(password == null){

                showMessage("Password
belum diisi");

            }

            else if(email ==null &&
password==null){

```

```

        showMessage("Form
        authenticakan belum diisi");

    }

    else{

        Call<GSONResponseLogin>
        postAuthServer=
        apiInterface.postAuth(email,password)

        postAuthServer.enqueue(new
        Callback<GSONResponseLogin>() {

            @Override
            public void
            onResponse(Call<GSONResponseLog
            in> call,
            Response<GSONResponseLogin>
            response) {

                if(response.isSuccessful()){

                    showMessage("Anda
                    berhasil login");

                }

                else{

                    if(response.code()==500){

                        showMessage("Response server
                        mati");

                    }

                }

            }

            @Override
            public void
            onFailure(Call<GSONResponseLogin
            > call, Throwable t) {

                if(call.isCanceled()){

                    showMessage("Kegagalan server");

                }

            }

        }

    }

    private void showMessage(String
    message){

        runOnUiThread(new
        Runnable() {

            @Override
            public void run() {

                if(message!=null){

```

```

    }

    Toast.makeText(getApplicationContext()
    t(),message,
    Toast.LENGTH_SHORT).show();

    }

    });

    }

}

package com.example.century.GSON;

import
com.google.gson.annotations.Expose;

import
com.google.gson.annotations.Serialize
dName;

public class GSONResponseLogin {

    @SerializedName("message")
    @Expose

    private String message;

    @SerializedName("code")
    @Expose

    private Integer code;

    public String getMessage() {

        return message;

    }

    public void setMessage(String
    message) {

        this.message = message;

    }

    public Integer getCode() {

        return code;

    }

    public void setCode(Integer code) {

        this.code = code;

    }

}

package com.example.century.API;

import
com.example.century.Shared.Param;

import retrofit2.Retrofit;

import
retrofit2.converter.gson.GsonConverte
rFactory;

public class RetrofitClient {

```



```

private static Retrofit retrofit = null;

public static Retrofit getClient() {

    if (retrofit==null) {

        retrofit = new
Retrofit.Builder()

.baseUrl(Param.BASE_DOMAIN)

.addConverterFactory(GsonConverterFactory.create())

        .build();
    }

    return retrofit;
}

package com.example.century.API;

import
com.example.century.GSON.GSONRe
sponseLogin;

import retrofit2.Call;

import retrofit2.http.Field;

import retrofit2.http.FormUrlEncoded;

import retrofit2.http.POST;

public interface ApiInterface {

    @FormUrlEncoded
    @POST("auth")

    Call<GSONResponseLogin>
postAuth(@Field("email") String
email,

@Field("password") String password);
}

```

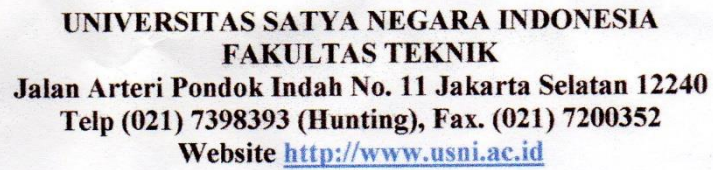


UNIVERSITAS SATYA NEGARA INDONESIA
FAKULTAS TEKNIK
Jalan Arteri Pondok Indah No. 11 Jakarta Selatan 12240
Telp (021) 7398393 (Hunting), Fax. (021) 7200352
Website <http://www.usni.ac.id>

KARTU BIMBINGAN SKRIPSI/TUGAS AKHIR
FAKULTAS TEHNIK

Nama : Lukman Amim
No Mhs : 011605503125040 Prodi : Sistem Informasi
Dosen Pembimbing I : Kiki Kusumawati, ST., MMSI.
Dosen Pembimbing II : Prionggo Hendradi, S.Kom., MMSI
Judul : Perancangan Sistem Informasi Pelayanan Tamu pada
Department Concierge di Hotel Century Park Berbasis Android

No	Tanggal	Catatan Pembimbing I	Ttd dosen pembimbing
1	01/10	Diskusi Tentang Judul	
2	8/10	Penyesuaian Judul tugas akhir	
3	15/10	konsultasi Proposal TA	
4	3/12	Review BAB I - III	
5	10/12	perbaikan BAB I - III	
6	7/2/2020	konsultasi Program & BAB III	
7	11/2/2020	Review Program	



Nama : Lukman Amim
No Mhs : 011605503125040 Prodi : Sistem Informasi
Dosen Pembimbing I : Kiki Kusumawati, ST., MMSI.
Dosen Pembimbing II : Prionggo Hendradi, S.Kom., MMSI
Judul : Perancangan Sistem Informasi Pelayanan Tamu pada
 Department Concierge di Hotel Century Park Berbasis Android

L 11



UNIVERSITAS SATYA NEGARA INDONESIA
FAKULTAS TEKNIK

Jalan Arteri Pondok Indah No. 11, Jakarta Selatan 12240
Telp. (021) 739 8393 (Hunting), Fax. (021) 720 0352
website <http://www.usni.ac.id>

Jakarta, 11 Oktober 2019

Nomor : 624/D-TEK/X/2019
Lampiran :
Perihal : Permohonan Riset

Kepada
Yth,

Front Office Manager
Hotel Century Park, Jakarta
Jl. Pintu Sate Senayan
Jakarta Pusat

Bersama ini kami dari Fakultas Teknik Universitas Satya Negara Indonesia (USNI) memberitadukan dan mohon dengan hormat hal-hal sebagai berikut:

1. Dalam rangka mengakhiri pendidikan Program Strata Satu (S-1) Fakultas Teknik USNI, salah satu syaratnya adalah Mahasiswa diwajibkan menulis skripsi.
2. Sehubungan dengan butir 1 (satu) di atas salah seorang Mahasiswa Fakultas Teknik USNI, yaitu:

Nama : LUKMAN AMIM
NIM : 011605503125040
Fak/Prodi : Teknik/Sistem Informasi (S-1)

Memilih instansi yang Bapak/Ibu pimpin sebagai objek penelitian untuk keperluan penyusunan skripsi.

3. Hasil penelitian yang dilakukan Mahasiswa tersebut hanya untuk keperluan ilmiah semata, dan tidak akan disebarluaskan.

Demikian permohonan ini kami ajukan, atas kesediaannya kami ucapkan terima kasih.

Hormat Kami,
Dekan

Ir. Nurhayati, M.Si



CENTURY PARK
Hotel

SURAT KETERANGAN

Nomor: 001/HR/II/2020

Atas nama HRD Hotel Century Park menerangkan bahwa :

Nama : Lukman Amim
NIM : 011605503125040
Fak/Prodi : Teknik/Sistem Informatika (S-1)
Universitas : Universitas Satya Negara Indonesia

Telah melaksanakan Penelitian/Riset di Hotel Century Park dalam rangka menyusun Skripsi
Demikian Surat Keterangan ini dibuat untuk di gunakan sebagaimana mestinya

Jakarta, 14 Oktober 2019

Hotel Century Park


CENTURY PARK
Imam Safar Hotel

Training and Development Coordinator

DAFTAR RIWAYAT HIDUP

A. Biodata Mahasiswa

N.I.M : 011605503125040

Nama Lengkap : Lukman Amim

Tempat & Tanggal Lahir : Pemalang, 04 February 1984

Alamat Lengkap : Jl Pondok Aren II No 73,
RT 001/001, Kel. Pondok Betung
Kec. Pondok Aren,
Tangerang Selatan

B. Riwayat Pendidikan

1. SD Negri 02 Padek, Ulujami, Pemalang : Lulus Tahun 1997
2. SMP Negri 02 Pamutih, Ulujami, Pemalang : Lulus Tahun 2000
3. SMK Nusantara 1 Comal, Comal, Pemalang : Lulus Tahun 2003

Jakarta, 17 Februari 2020

Saya yang bersangkutan

Lukman Amim