

## **TUGAS AKHIR I**

### **IMPLEMENTASI ENTERPRISE RESOURCE PLANNING (ERP) MENGGUNAKAN ODOO PADA PEMBUATAN SISTEM LSP UNIVERSITAS DIAN NUSWANTORO SEMARANG**

Diajukan untuk memenuhi salah satu syarat mengerjakan dan  
menempuh ujian tugas akhir 2



Disusun Oleh:

Nama : Dwi Dian Ayu Puji Astuti

NIM : A11.2015.09001

Program Studi : Teknik Informatika – S1

**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DIAN NUSWANTORO  
SEMARANG**

**2019**

## **HALAMAN PERSETUJUAN TUGAS AKHIR I**

Nama : Dwi Dian Ayu Puji Astuti  
NIM : A11.2015.09001  
Program Studi : Teknik Informatika  
Fakultas : Ilmu Komputer  
Judul Tugas Akhir : IMPLEMENTASI ENTERPRISE RESOURCE  
PLANNING (ERP) MENGGUNAKAN ODOO PADA  
PEMBUATAN SISTEM LSP UNIVERSITAS DIAN  
NUSWANTORO SEMARANG

Tugas Akhir ini diperiksa dan disetujui,  
Semarang, 16 Januari 2019

Menyetujui:  
Pembimbing

**Dr. Guruh Fajar Shidik, S.Kom., M.Cs**

## **HALAMAN PENGESAHAN**

Nama : Dwi Dian Ayu Puji Astuti  
NIM : A11.2015.09001  
Program Studi : Teknik Informatika  
Fakultas : Ilmu Komputer  
Judul Tugas Akhir : IMPLEMENTASI ENTERPRISE RESOURCE  
PLANNING (ERP) MENGGUNAKAN ODOO PADA  
PEMBUATAN SISTEM LSP UNIVERSITAS DIAN  
NUSWANTORO SEMARANG

Tugas akhir ini telah diajukan dan dipertahankan dihadapan Dewan Penguji pada Sidang tugas akhir tanggal 16 Januari 2019. Menurut pandangan kami, tugas akhir ini memadai dari segi kualitas maupun kuantitas untuk tujuan penganugerahan gelar Sarjana (S. Kom)

Semarang, 16 Januari 2019

Ketua Penguji

**Dr M. Arief Soeleman, M.Kom**

## HALAMAN RINGKASAN

Lembaga Sertifikasi Profesi (LSP) merupakan lembaga pelaksanaan kegiatan sertifikasi profesi yang memperoleh lisensi dari Badan Nasional Sertifikasi Profesi (BNSP). Sertifikat kompetensi kerja yang diakui secara nasional dapat meningkatkan kredibilitas seorang professional di dunia kerja. Dengan memiliki sertifikat kompetensi kerja, seorang professional akan memiliki rasa kepercayaan diri yang lebih tinggi terkait dengan keterampilan yang dimiliki. Tugas utama LSP Udinus yaitu melaksanakan sertifikasi, dimana untuk melaksanakan tugas tersebut tentunya diperlukan sebuah sistem yang dapat mencakup dan terintegrasi dari fungsi-fungsi pendaftaran sampai penyerahan sertifikasi.

*Enterprise Resource Planning* (ERP) adalah istilah yang banyak digunakan di perusahaan – perusahaan besar. Tetapi penerimaannya yang luas dan mengarah pada penggunaan sistem semacam itu, menjadikan ERP dapat digunakan baik dalam skala menengah maupun kecil. Sistem ERP dapat menawarkan peluang besar untuk pengurangan biaya, peningkatan kualitas dan peningkatan efisiensi operasional.

Laporan tugas akhir I ini akan menjelaskan tentang alur proses bagaimana pembuatan sistem LSP Universitas Dian Nuswantoro, dari mulai analisis data, perancangan sistem, pembangunan sistem, implementasi hingga pengujian sistem. Pada tahap akhir pengembangan perangkat lunak, akan dilakukan evaluasi terhadap sistem. Hal-hal apa yang telah dilakukan dan apa yang belum dilakukan pada pengembangan perangkat lunak ini akan diulas pada bagian akhir tugas akhir.

## DAFTAR ISI

HALAMAN PERSETUJUAN TUGAS AKHIR I.....	i
HALAMAN PENGESAHAN .....	ii
HALAMAN RINGKASAN .....	iii
DAFTAR ISI .....	iv
DAFTAR TABEL .....	ivi
DAFTAR GAMBAR .....	vii
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang Masalah .....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Masalah .....	3
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian .....	4
BAB II LANDASAN TEORI .....	5
2.1 Tinjauan Studi .....	5
2.2 Tinjauan Pustaka .....	10
2.2.1 Lembaga Sertifikasi Profesi .....	10
2.2.2 Enterprise Resource Planning .....	10
2.2.3 Odoo .....	11
2.2.4 Python .....	12
2.2.5 Unified Modeling Language .....	12
2.2.6 Model Prototype .....	13
2.2.7 Black Box Testing .....	15
2.2.8 White Box Testing .....	17
2.2.9 Quality of Experience .....	19
2.2.10 Quality of Service .....	21
2.3 Kerangka Pemikiran .....	23
BAB III METODE PENELITIAN .....	254
3.1 Instrumen Penelitian .....	24
3.2 Metode Analisa Data .....	25

3.3 Tinjauan Pustaka .....	25
3.4 Metode Pengembangan Sistem .....	26
3.5 Pengujian Software .....	27
3.5.1 Black Box Testing .....	27
3.5.2 White Box Testing .....	28
3.5.3 Quality of Experience .....	28
3.5.4 Quality of Service .....	31
DAFTAR PUSTAKA .....	32

## DAFTAR TABEL

Tabel 2. 1 Contoh hasil kategori nilai skala .....	21
Tabel 2. 2 Contoh parameter QoS .....	22
Tabel 3. 1 Contoh Quisioner QoE .....	29
Tabel 3. 2 Contoh Evaluasi QoE .....	30

## DAFTAR GAMBAR

Gambar 2. 1 Model Pengembangan Prototyping .....	14
Gambar 2. 2 Proses pengerjaan teknik black box testing.....	16
Gambar 2. 3 Represent White box testing.....	18
Gambar 2. 4 Scale structure of the UEQ .....	20
Gambar 3. 1 Equivalence Partitioning black box.....	28
Gambar 3. 2 Contoh Diagram Evaluasi QoE .....	31



# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang Masalah**

Lembaga Sertifikasi Profesi (LSP) merupakan lembaga pelaksanaan kegiatan sertifikasi profesi yang memperoleh lisensi dari Badan Nasional Sertifikasi Profesi (BNSP). Lisensi diberikan melalui proses akreditasi oleh BNSP yang menyatakan bahwa LSP bersangkutan telah memenuhi syarat untuk melakukan kegiatan sertifikasi profesi yang nantinya akan mendapatkan sertifikat kompetensi kerja[1]. Sertifikat kompetensi kerja yang diakui secara nasional dapat meningkatkan kredibilitas seorang professional di dunia kerja. Dengan memiliki sertifikat kompetensi kerja, seorang professional akan memiliki rasa kepercayaan diri yang lebih tinggi terkait dengan keterampilan yang dimiliki[2].

Lembaga Sertifikasi Profesi Universitas Dian Nuswantoro (LSP Udinus) adalah lembaga independen dibawah naungan institusi induk Universitas Dian Nuswantoro (UDINUS) Semarang yang mana bertanggung jawab dalam pelaksanaan sertifikasi kompetensi kerja di lingkungan Universitas Dian Nuswantoro beserta jejaringnya berdasarkan Surat Keputusan Rektor No. 066/Kep/UDN-01/VII/2017 tentang Pembentukan Lembaga Sertifikasi Profesi (LSP) Universitas Dian Nuswantoro. LSP Udinus berfungsi untuk melaksanakan sertifikasi kompetensi kerja dengan tugas yaitu menyusun dan mengembangkan skema sertifikasi, membuat perangkat asesmen dan materi uji kompetensi, menyediakan tenaga pengusi (asesor), melaksanakan sertifikasi, melaksanakan surveilan pemeliharaan sertifikasi, menetapkan persyaratan, memverifikasi dan menetapkan TUK, memelihara kinerja asesor dan TUK, serta mengembangkan pelayanan sertifikasi[3]. Tugas utama LSP Udinus yaitu melaksanakan sertifikasi, dimana untuk melaksanakan tugas tersebut tentunya diperlukan sebuah sistem yang dapat mencakup dan terintegrasi dari fungsi-fungsi pendaftaran sampai penyerahan sertifikasi.

Dari permasalahan di atas penulis tertarik untuk mengimplementasikan *Enterprise Resource Planning* (ERP) untuk pembuatan sistem LSP Udinus. *Enterprise Resource Planning* (ERP) adalah istilah yang banyak digunakan di perusahaan – perusahaan besar. Tetapi penerimaannya yang luas dan mengarah pada penggunaan sistem semacam itu, menjadikan ERP dapat digunakan baik dalam skala menengah maupun kecil. Sistem ERP pada dasarnya adalah sistem dengan kompleksitas tinggi yang mengintegrasikan berbagai proses bisnis dan informasi yang terkait dalam suatu organisasi. Beberapa *market leaders* dalam pengembangan dan penerapan sistem ERP diantaranya adalah SAP, Oracle, Microsoft, Sage sistem dan lain-lain. Vendor tersebut menyediakan solusi ERP yang luas kepada pelanggan dengan biaya yang sangat tinggi. Pendekatan ini tentunya menyulitkan perusahaan kecil maupun menengah untuk berinvestasi pada sistem ERP. Hal ini menyebabkan munculnya sistem ERP berbasis *open source* yang lebih murah dibandingkan dengan sistem ERP *proprietary* dan memungkinkan kompatibilitas di berbagai platform, *tools*, *plugin* dan sebagainya[4].

Penelitian yang dilakukan sebelumnya [4][5], membahas tentang analisis beberapa solusi *open source* ERP dan juga CRM yang populer. Pada penelitian [5], dijelaskan bahwa *Open Source* ERP terus mendapatkan penerimaan dalam suatu organisasi bisnis. Hal ini dikarenakan mereka dapat menawarkan peluang besar untuk pengurangan biaya, peningkatan kualitas dan peningkatan efisiensi operasional, terutama untuk usaha kecil dan menengah yang biasanya harus mengatasi kesulitan besar karena sumber daya yang terbatas. Salah satu *Open Source* ERP yang telah berkembang pesat dan telah menjadi pemimpin dalam komunitas bisnis ukuran kecil hingga besar adalah OpenERP / Odoo[4]. Seperti yang telah disebutkan, Odoo memiliki semua kelebihan yang ditawarkan oleh sistem ERP *Open Source* seperti lebih murah, fleksibilitas, kepelikan mutlak, jaminan kualitas dan mudah diupgrade. Oleh karena itu, pada penelitian ini penulis akan melakukan implementasi pembuatan sistem menggunakan Odoo, dimana dapat disampaikan jika penulis akan membuat tugas akhir yang berjudul

## **“IMPLEMENTASI ENTERPRISE RESOURCE PLANNING (ERP) MENGGUNAKAN ODOO PADA PEMBUATAN SISTEM LSP UNIVERSITAS DIAN NUSWANTORO SEMARANG”.**

### **1.2 Rumusan Masalah**

Berdasarkan latar belakang masalah yang telah diuraikan, maka dalam penelitian ini permasalahan yang akan dirumuskan diantaranya adalah :

1. Bagaimana Quality of Experience implementasi Odoo pada Sistem LSP Universitas Dian Nuswantoro Semarang?
2. Bagaimana Quality of Service implementasi Odoo pada Sistem LSP Universitas Dian Nuswantoro Semarang?

### **1.3 Batasan Masalah (Ruang Lingkup)**

1. Menggunakan *Enterprise Resource Planning* sebagai integrasi sistem yang akan diterapkan
2. Menggunakan *Odoo* sebagai bahasa utama dalam pembuatan sistem.
3. Menggunakan Quality of Experience dan Quality of Service sebagai parameter keberhasilan sistem.
4. Objek penelitian menggunakan sistem LSP Universitas Dian Nuswantoro.

### **1.4 Tujuan Penelitian**

1. Mengetahui keberhasilan parameter Quality of Experience implementasi Odoo pada Sistem LSP Universitas Dian Nuswantoro Semarang.
2. Mengetahui keberhasilan parameter Quality of Service implementasi Odoo pada Sistem LSP Universitas Dian Nuswantoro Semarang.

### 1.5 Manfaat Penelitian

Beberapa manfaat yang didapatkan dari penelitian ini adalah :

1. Bagi Penulis

Penulis dapat mengetahui cara mengimplementasikan *Enterprise Resource Planning* menggunakan Odoo pada sistem LSP Universitas Dian Nuswantoro.

2. Bagi Universitas

- a. Sebagai titik acuan universitas kepada mahasiswa seberapa pemahaman dan penguasaan terhadap materi.
- b. Sebagai bahan referensi bagi mahasiswa yang akan mengembangkan permasalahan ataupun metode yang berbeda.

3. Bagi Peneliti Selanjutnya

Sebagai gambaran para peneliti yang akan melakukan penelitian dengan menggunakan Odoo untuk impelentasi *Enterprise Resource Planning* atau dengan sistem integrasi pengelolaan bisnis lain yang lebih baik.

## **BAB II**

### **LANDASAN TEORI**

#### **2. 1 Tinjauan Studi**

##### **2.1.1 State Of The Art**

Penelitian dalam bidang implementasi *Enterprise Resource Planning* dan *Odoo* sudah banyak dikembangkan sehingga menghasilkan beberapa produk yang baru dan bermanfaat bagi pengguna. *State of the art* atau penelitian sebelumnya, akan membahas mengenai persamaan dan perbedaan yang ada dalam penelitian sebelumnya. Dimana persamaan dan perbedaan akan dilihat dari objek yang diteliti, serta metode penelitiannya. Penyusunan data state of the art akan disusun dalam bentuk data matrik , sehingga dapat memperlihatkan perbandingan antara penelitian sebelumnya dengan penelitian yang akan dilakukan. Dan berikut beberapa referensi yang digunakan adalah:

1. Penelitian yang dilakukan Lerina [6]. Dalam penelitiannya yakni *Understanding Enterprise Open Source Software Evolution*. Hasil yang didapatkan yaitu mengusulkan analisis dan perbandingan beberapa solusi ERP dan CRM *open source* yang populer. Hasil penelitian ini menyoroti beberapa aspek penting yang harus ada pada ERP.
2. Penelitian yang dilakukan Samwel [7]. Dalam penelitian tersebut yaitu *Enterprise Resource Planning (ERP) System Implementation: A case for User participation*, penelitian tersebut menjelaskan kontribusi dan partisipasi apa saja yang dapat dilakukan oleh user ketika implementasi ERP.
3. Penelitian yang dilakukan Eli Hustada[8]. Pada penelitian *ERP and organizational misfits: An ERP customization journey* terhadap Perusahaan dan organisasi, didapat hasil bahwa ada

beberapa prosedur penting selama implementasi ERP, di mana kustomisasi ERP juga dapat dilakukan.

4. Penelitian yang dilakukan Moutaz [9]. Dalam penelitian ini dijelaskan sebagian besar sistem ERP saat ini termasuk modul dan aplikasi CRM, meski bagaimanapun, sistem CRM terpisah masih banyak diadopsi oleh organisasi.

N o	Penulis	Judul	Masalah	Metode	Hasil
1	Lerina Aversano, Marco Di Brino, Daniela Guardabascio, Marcella Salerno, Maria Tortorella	<i>Understanding Enterprise Open Source Software Evolution</i>	Kurangnya pemahaman tentang manfaat potensial yang berasal dari adopsi solusi proyek Open Source Software terutama dalam bidang <i>Enterprise Resource Planning</i> .	<i>ERP Opensource software / OpenERP</i>	Laporan hasil studi empiris yang berisi analisis evolusi sistem open source ERP yang paling relevan untuk digunakan.
2	Samwel Matende, Patrick Ogaob	<i>Enterprise Resource Planning (ERP) System Implementation : A case for User participation</i>	Kurangnya studi dan literatur tentang partisipasi pengguna dan kontribusi pengguna terhadap keberhasilan	<i>Enterprise Resource Planning</i>	Implementasi ERP berbeda dari pengembangan sistem tradisional di mana fokus utama telah bergeser dari penekanan

			implementasi sistem ERP.		berat pada analisis teknis dan pemrograman menuju desain proses bisnis dan elemen manusia.
3	Eli Hustada, Moutaz Haddarab, Baldwin Kalvenes	<i>ERP and organizational misfits: An ERP customization journey</i>	Kurangnya penyesuaian prosedur penting selama implementasi ERP, di mana kustomisasi ERP dilakukan untuk memastikan kepatuhan dengan proses dan persyaratan bisnis penting organisasi.	<i>ERP System misfit</i>	Hasil kami menunjukkan bahwa <i>tailoring decision</i> adalah proses yang dipengaruhi oleh keragaman keadaan sosial; resistensi terhadap sistem, transisi dari sistem lama, pengetahuan organisasi tentang sistem ERP, dan kematangan organisasi.
4	Moutaz Haddara,	<i>ERP II is Dead- Long</i>	Fakta bahwa sebagian besar	<i>Enterprise Resource</i>	Pengembangan empat

	Angelo Constantini	<i>Live CRM</i>	<p>sistem ERP saat ini termasuk modul dan aplikasi CRM, bagaimanapun, sistem CRM terpisah masih banyak diadopsi oleh organisasi. Dengan demikian, penelitian ini mencoba untuk menyelidiki alasan utama yang memotivasi organisasi untuk berinvestasi dalam sistem CRM yang berbeda daripada menggunakan modul terkait CRM dalam sistem ERP yang diperoleh</p>	<i>Planning</i>	<p>alasan utama untuk tolak ukur keberhasilan ERP, yaitu: pelingkupan selama implementasi ERP, biaya, fitur, keramahan pengguna dan kemudahan penggunaan. Makalah ini, pada akhirnya, memberikan ruang untuk penelitian lebih lanjut.</p>
--	--------------------	-----------------	--	-----------------	---



*State of the art* di atas membahas tentang persamaan dan perbedaan yang ada dalam penelitian sebelumnya, dan berikut perbandingan antara penelitian yang sebelumnya dengan penelitian yang akan dilakukan adalah sebagai berikut.

Perbedaan penelitian yang dilakukan oleh Lerina dengan penelitian yang akan di lakukan yakni dalam penelitian ini penulis menggunakan metode OpenERP terbaik dari perbandingan penelitian sebelumnya sebagai implementasi pada metode penelitian.

Perbedaan dari penelitian yang dilakukan oleh Samwel Mantende dengan penelitian yang akan di lakukan yakni dalam penelitian ini menjelaskan kontribusi dan partisipasi apa saja yang dapat dilakukan oleh user ketika implementasi ERP. Sedangkan penelitian yang akan dilakukan, penulis mengadopsi hasil dari penelitian yang bersangkutan untuk implementasi ERP.

Perbedaan dari penelitian yang dilakukan oleh Eli Hustada dengan penelitian yang akan di lakukan yakni dalam penelitian ini menjelaskan penyesuaian prosedur penting selama implementasi ERP. Sedangkan penelitian yang akan dilakukan, penulis mengadopsi penyesuaian prosedur tersebut pada implementasi ERP.

Perbedaan dari penelitian yang dilakukan oleh Moutaz Haddara dengan penelitian yang akan dilakukan adalah dalam penelitian ini menjelaskan tolak ukur keberhasilan implementasi ERP dan juga CRM. Sedangkan penelitian yang akan dilakukan menggunakan tolak ukur tersebut agar proses implementasi ERP nantinya dapat berjalan dengan baik.

## **2.2 Tinjauan Pustaka**

### **2.2.1 Lembaga Sertifikasi Profesi**

Lembaga Sertifikasi Profesi (LSP) merupakan lembaga pelaksanaan kegiatan sertifikasi profesi yang memperoleh lisensi dari Badan Nasional Sertifikasi Profesi (BNSP). Lisensi diberikan melalui proses akreditasi oleh BNSP yang menyatakan bahwa LSP bersangkutan telah memenuhi syarat untuk melakukan kegiatan sertifikasi profesi yang nantinya akan mendapatkan sertifikat kompetensi kerja[1]. Sertifikat kompetensi kerja yang diakui secara nasional dapat meningkatkan kredibilitas seorang professional di dunia kerja. Karena sertifikasi kompetensi merupakan sebuah pengakuan terhadap tenaga kerja yang mempunyai keterampilan dan kemampuan yang mumpuni sesuai dengan standar kerja yang sudah ditetapkan. Dengan memiliki sertifikat kompetensi kerja, seorang professional akan memiliki rasa kepercayaan diri yang lebih tinggi terkait dengan keterampilan yang dimiliki[2].

### **2.2.2 Enterprise Resource Planning**

Enterprise resource planning (ERP) adalah perangkat lunak manajemen proses bisnis yang memungkinkan suatu organisasi untuk menggunakan sistem aplikasi terintegrasi untuk mengelola bisnis dan mengotomatiskan banyak fungsi back office yang terkait dengan teknologi, layanan, dan sumber daya manusia.

Perangkat lunak ERP biasanya mengintegrasikan semua aspek operasi - termasuk perencanaan produk, pengembangan, produksi, penjualan dan pemasaran - dalam satu basis data, aplikasi, dan antarmuka pengguna.

Perangkat lunak ERP dianggap sebagai jenis aplikasi perusahaan, yaitu perangkat lunak yang dirancang untuk digunakan oleh bisnis yang lebih besar dan seringkali memerlukan tim khusus untuk menyesuaikan dan menganalisis data dan untuk menangani peningkatan dan penyebaran. Sebaliknya, aplikasi ERP Usaha Kecil

adalah solusi perangkat lunak manajemen bisnis ringan, sering kali disesuaikan untuk industri bisnis tertentu atau vertikal.

Saat ini sebagian besar organisasi menerapkan sistem ERP untuk menggantikan perangkat lunak lama atau untuk memasukkan aplikasi ERP karena saat ini tidak ada sistem. Faktanya, studi tahun 2016 oleh Panorama Consulting Solutions, LLC., Menunjukkan bahwa organisasi menerapkan ERP karena alasan berikut:

- Untuk mengganti perangkat lunak ERP yang kedaluwarsa (49%)
- Untuk mengganti sistem homegrown (16%)
- Untuk mengganti perangkat lunak akuntansi (15%)
- Untuk mengganti sistem non-ERP lainnya / tidak memiliki sistem (20%) [10]

### 2.2.3 Odoo

Odoo adalah sistem ERP open source yang sebelumnya dikenal sebagai OpenERP. Odoo dianggap sebagai aplikasi bisnis terinstal tertinggi di dunia dengan lebih dari 2.000.000 pengguna.

Odoo menawarkan sistem On-Premise dan Cloud ERP. Ini terdiri dari 30 aplikasi utama seperti (penjualan, e-commerce, faktur, akuntansi dan manajemen situs web pengguna). Selain itu, lebih banyak modul dan aplikasi telah diterbitkan oleh pengembang dari seluruh dunia.[11]

Odoo dikembangkan menggunakan Python8. Odoo menyediakan cara standar bagi pengembang untuk mengembangkan modul Odoo baru atau menyesuaikan dan memodifikasi modul yang sudah ada. Modul Odoo terdiri dari beberapa model yang berinteraksi satu sama lain dan dengan modul lain untuk mencapai tujuan modul yang dikembangkan.

Model *Inheritance* dan *View Inheritance* adalah fitur utama di Odoo yang memungkinkan pengembang untuk menambahkan fitur baru ke model atau melihat dan memodifikasi model atau tampilan yang ada.[12]

#### 2.2.4 Python

Python *is an interpreted, interactive, object-oriented programming language*. Ini mencakup modul, pengecualian, pengetikan dinamis, tipe data dinamis tingkat sangat tinggi, dan kelas. Python menggabungkan kekuatan luar biasa dengan sintaksis yang sangat jelas. Ini memiliki antarmuka ke banyak panggilan sistem dan perpustakaan, serta berbagai sistem jendela, dan dapat diperluas dalam C atau C ++. Ini juga dapat digunakan sebagai bahasa ekstensi untuk aplikasi yang membutuhkan antarmuka yang dapat diprogram. Akhirnya, Python portabel: ia berjalan pada banyak varian Unix, pada Mac, dan pada PC di bawah MS-DOS, Windows, Windows NT, dan OS / 2. Python adalah bahasa pemrograman tujuan umum tingkat tinggi yang dapat diterapkan ke banyak kelas masalah yang berbeda.

Python adalah bahasa pemrograman *high-level general-purpose* yang dapat diterapkan ke banyak kelas masalah yang berbeda. Bahasa ini dilengkapi dengan pustaka standar besar yang mencakup area-area seperti pemrosesan string (ekspresi reguler, Unicode, penghitungan perbedaan antar file), protokol Internet (HTTP, FTP, SMTP, XML-RPC, POP, IMAP, pemrograman CGI), rekayasa perangkat lunak (unit testing, logging, profiling, parsing kode Python), dan antarmuka sistem operasi (system calls, filesystems, TCP / IP sockets).[13]

#### 2.2.5 Unified Modeling Language

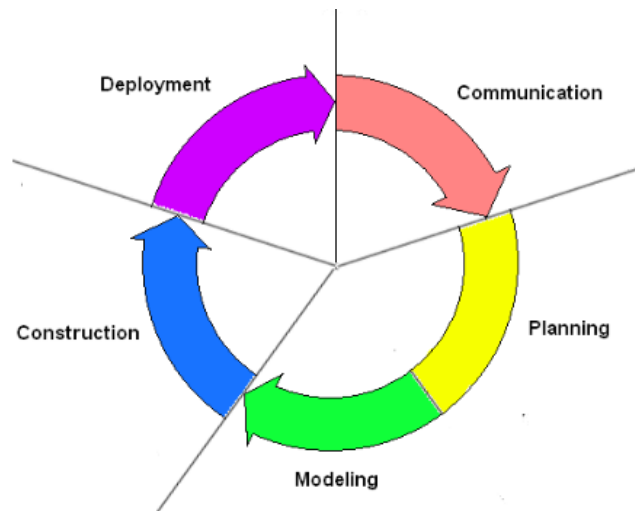
Unified Modeling Language (UML) adalah bahasa pemodelan *general-purpose*, pengembangan, di bidang rekayasa perangkat lunak, yang dimaksudkan untuk memberikan cara standar untuk memvisualisasikan desain sistem. Unified Modeling Language (UML) merupakan bahasa standar untuk menulis *blueprint* perangkat lunak. UML dapat digunakan untuk memvisualisasikan, menentukan, membangun, dan mendokumentasikan artefak dari sistem intensif perangkat lunak.

UML sesuai untuk sistem pemodelan mulai dari sistem informasi perusahaan hingga aplikasi berbasis Web terdistribusi dan bahkan ke sistem tertanam waktu nyata. Ini adalah bahasa yang sangat ekspresif, membahas semua pandangan yang diperlukan untuk pengembangan dan kemudian menggunakan sistem tersebut. Meskipun ekspresif, UML tidak sulit untuk dipahami dan digunakan. Belajar menerapkan UML secara efektif dimulai dengan membentuk model konseptual bahasa, yang membutuhkan pembelajaran tiga elemen utama: blok bangunan dasar UML, aturan yang menentukan bagaimana blok bangunan ini dapat disatukan, dan beberapa mekanisme umum yang berlaku di seluruh bahasa.

UML hanya bahasa dan hanya satu bagian dari metode pengembangan perangkat lunak. UML adalah proses yang independen, meskipun secara optimal harus digunakan dalam proses yang menggunakan use case, arsitektur-sentris, iteratif, dan inkremental. [14]

#### **2.2.6 Model Prototype**

Metode *prototyping* adalah yang berulang, di mana dalam setiap iterasi perangkat lunak prototipe dikembangkan. Pada iterasi pertama, hanya sebagian dari persyaratan yang ditentukan yang diterapkan, dan pada setiap iterasi berikutnya prototipe baru mengimplementasikan bagian tambahan dari persyaratan. Setiap prototipe diperiksa dan dievaluasi oleh pelanggan dan tim pengguna. Tuntutan mereka untuk perbaikan, perubahan, dan penambahan terkait dengan prototipe saat ini harus dipertimbangkan oleh pengembang dalam prototipe integrasi berikutnya. Iterasi ini akan berlanjut sampai semua persyaratan terpenuhi. [15]



*Gambar 2. 1 Model Pengembangan Prototyping*

Berdasarkan penelitian [16], berikut beberapa tahapan prototyping:

- a. *Planning* : Mendefinisikan format dan kebutuhan keseluruhan perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.
- b. *Modeling* : Segera membuat model, dan *quick design* fokus pada gambaran dari segi *software* apakah visible menurut customer.
- c. *Construction* : Membangun *prototyping* dengan membuat perancangan sementara yang berpusat pada penyajian kepada pelanggan (misalnya dengan membuat input dan contoh outputnya).
- d. *Deployment* : Dalam tahap ini *prototyping* yang sudah disepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.
- e. *Communication* : Komunikasi antara developer dan customer mengenai apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan.

Metode *prototyping*[15] telah ditemukan sangat berhasil untuk proyek-proyek kecil dan menengah. Ini juga berlaku untuk mengembangkan bagian dari proyek skala besar. *Prototyping* dapat

diterapkan dalam kombinasi dengan metodologi lain atau sebagai metodologi "berdiri sendiri". Dengan kata lain, luasnya prototyping dapat bervariasi dari mengganti satu SDLC (atau metodologi lain) di bagian proyek hingga prototyping lengkap dari keseluruhan sistem perangkat lunak.

Ada berbagai situasi tipikal ketika prototyping digunakan:

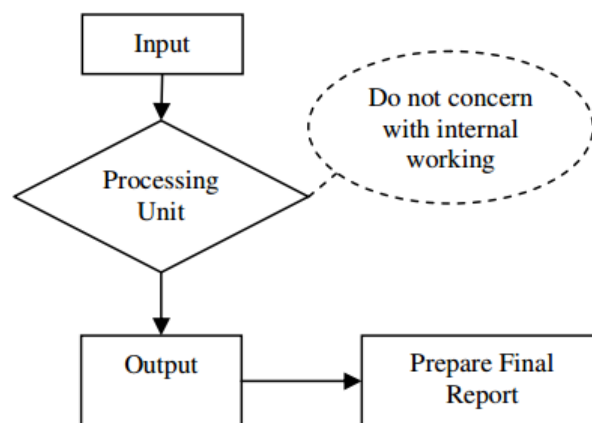
- Proyek dengan spesifikasi yang tidak jelas, di mana umpan balik dari pengguna pada prototipe berfungsi untuk mengeksplorasi persyaratan. Dalam banyak kasus ini, satu-satunya minat dalam proses prototyping adalah untuk secara akurat menentukan persyaratan sistem yang dihasilkan selama proses. Dengan demikian, perangkat lunak prototyping yang dihasilkan adalah perangkat lunak "sekali pakai" dan tidak digunakan.
- Sistem perangkat lunak yang direncanakan untuk layanan pendek dan untuk populasi pengguna yang terbatas, yaitu manajemen penelitian, penggunaan sementara sebagai sistem perangkat lunak perintis.
- Proyek untuk penggunaan reguler, ditandai dengan kompleksitas yang rendah dan populasi pengguna yang terbatas.

Kelebihan dan kekurangan utama dari *prototyping* atas manfaat SDLC lengkap dari keterlibatan intens pengguna dan pelanggan dalam proses pengembangan perangkat lunak. Keterlibatan seperti itu memfasilitasi pemahaman yang lebih baik tentang sistem dan kemungkinan kegagalan sistem yang lebih rendah. Namun, partisipasi intensif pelanggan dan pengguna dalam proses pengembangan membatasi kebebasan pengembang untuk memperkenalkan perubahan inovatif ke dalam sistem.

### 2.2.7 Black Box Testing

Pengujian *Black Box* adalah pengujian tanpa sepengetahuan kerja internal aplikasi yang sedang diuji (AUT). Juga dikenal sebagai pengujian fungsional atau pengujian input driven. Suatu teknik

pengujian perangkat lunak di mana cara kerja internal dari item yang diuji tidak diketahui oleh tester. Misalnya, dalam tes kotak hitam pada AUT, tester hanya mengetahui input dan apa hasil yang diharapkan dan bagaimana program sampai pada output tersebut. Penguji tidak pernah memeriksa kode pemrograman dan tidak memerlukan pengetahuan lebih lanjut tentang program selain spesifikasinya. Untuk alasan ini, penguji dan pemrogram dapat independen satu sama lain, menghindari bias pemrogram terhadap pekerjaannya sendiri. Metode desain pengujian ini berlaku untuk semua tingkat pengujian perangkat lunak: unit, integrasi, pengujian fungsional, sistem dan penerimaan.[17]



*Gambar 2. 2 Proses pengerjaan teknik black box testing*

Berikut adalah langkah-langkah yang menjelaskan proses kerja *Black Box Testing*[18] :

- Langkah 1 Input: Persyaratan dan spesifikasi fungsional sistem diperiksa. Dokumen desain tingkat tinggi dan kode sumber blok aplikasi juga diperiksa. Penguji memilih input yang valid dan menolak input yang tidak valid.
- Langkah 2 Unit Pemrosesan: Jangan khawatir dengan kerja internal sistem. Dalam unit pemrosesan, tester membuat kasus uji dengan input yang dipilih dan menjalankannya. Tester juga



melakukan pengujian beban, pengujian stres, tinjauan keamanan, dan pengujian globalisasi. Jika ada cacat terdeteksi itu akan diperbaiki dan diuji kembali.

- Langkah 3 Output: Setelah semua pengujian ini, tester mendapatkan output yang diinginkan dan menyiapkan laporan akhir.

### 2.2.8 White Box Testing

Pengujian *White Box*[17] dilakukan berdasarkan pengetahuan tentang bagaimana sistem diimplementasikan. *White Box Testing* meliputi analisis aliran data, aliran kontrol, arus informasi, praktik pengkodean, dan penanganan pengecualian dan kesalahan dalam sistem, untuk menguji perilaku perangkat lunak yang dimaksudkan dan tidak diinginkan. Pengujian *White Box* dapat dilakukan untuk memvalidasi apakah implementasi kode mengikuti desain yang dimaksudkan, untuk memvalidasi fungsionalitas keamanan yang diterapkan, dan untuk mengungkap kerentanan yang dapat dieksploitasi. Pengujian *White Box* memerlukan akses ke kode sumber. Meskipun Pengujian Kotak Putih dapat dilakukan kapan saja dalam siklus hidup setelah kode dikembangkan, praktik yang baik untuk melakukan Pengujian Kotak Putih selama fase pengujian unit.

Dalam pengujian *White Box*, tidak hanya memeriksa cara kerja, tetapi juga melengkapi pengujian *black box* karena tidak dapat membuat kasus pengujian lengkap karena sumber daya yang terbatas dan pertukaran biaya-manfaat. Ada fungsi / metode yang akan memberi output yang benar setelah memberikan input tetapi mungkin ada *test case* input yang tidak dipikirkan dan ini dapat menimbulkan masalah di sisi klien. Jadi jika terdapat output yang benar untuk input yang pasti, ini tidak berarti fungsi / metode yang telah diperiksa sudah lengkap. Untuk ini perlu adanya pengujian *white box* untuk sepenuhnya memahami mekanisme dan cara menghasilkan output menggunakan input tertentu.

Application Code

Pengujian *White Box*, dibagi menjadi dua langkah dasar:

Hal pertama yang akan sering dilakukan oleh seorang tester

- adalah mempelajari dan memahami *source code* aplikasi. Karena Pengujian *White Box* melibatkan pengujian cara kerja suatu aplikasi, tester harus sangat berpengetahuan dalam bahasa pemrograman yang digunakan dalam aplikasi yang mereka uji. Juga, orang yang diuji harus sangat menyadari praktik *secure coding*.

Keamanan sering menjadi salah satu tujuan utama pengujian perangkat lunak. Penguji harus dapat menemukan masalah keamanan dan mencegah serangan dari peretas dan pengguna naif yang mungkin menyuntikkan kode berbahaya ke dalam aplikasi baik secara sadar atau tidak sadar.

- Langkah 2) Buat *Test Case* dan *Execute*

- Langkah dasar kedua untuk Pengujian *White Box* melibatkan pengujian *source code* aplikasi untuk aliran dan struktur yang tepat. Salah satu caranya adalah dengan menulis lebih banyak kode untuk menguji *source code* aplikasi. Pengujian akan mengembangkan tes kecil untuk setiap proses atau serangkaian proses dalam aplikasi. Metode ini mensyaratkan

bahwa tester harus memiliki pengetahuan yang mendalam tentang kode dan sering dilakukan oleh pengembang.

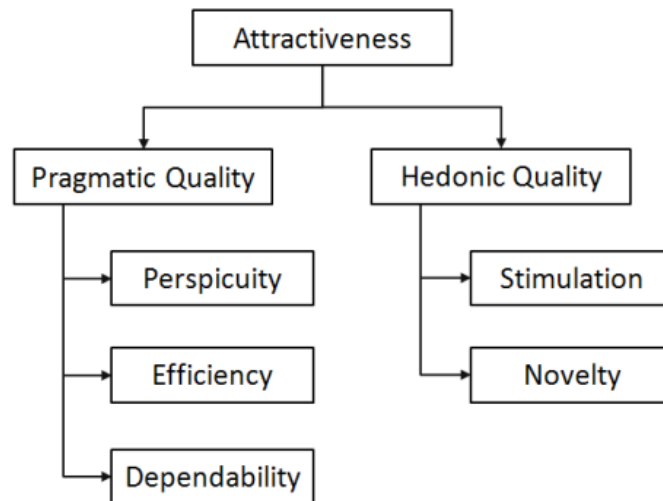
### 2.2.9 Quality of Experience

*The International Telecommunication Union (ITU)* mendefinisikan QOE sebagai penerimaan keseluruhan aplikasi atau layanan, sebagai perspektif subyektif oleh pengguna/user akhir/pelanggan/ (end user)[19]. *User experience* menggambarkan kesan subyektif pengguna terhadap suatu produk sebelum, selama dan setelah penggunaan (ISO 9241-210). Pengguna yang berbeda atau kelompok pengguna yang berbeda dapat menilai produk yang sama sangat berbeda mengenai pengalaman penggunaannya, misalnya karena mereka memiliki kebutuhan yang berbeda atau kemampuan atau keterampilan yang berbeda untuk menggunakan produk tersebut. Juga, pengguna yang lebih tua mungkin mengalami layanan Web publik yang sangat berbeda dibandingkan dengan pengguna yang lebih muda secara teknis lebih terampil. Kelompok sasaran suatu produk sangat penting.[20]

Pengujian Quality of Experience dapat dilakukan dengan beberapa cara, salah satunya yaitu dengan *User Experience Questionnaire (UEQ)*. Pada penelitian [21], dijelaskan bahwa tujuan utama UEQ adalah pengukuran UX yang cepat dan langsung. Kuesioner dirancang untuk digunakan sebagai bagian dari tes kegunaan normal, tetapi juga sebagai kuesioner *online*. Untuk penggunaan *online*, harus dimungkinkan untuk mengisi kuesioner dengan cepat, untuk menghindari peserta tidak menyelesaikannya. Jadi diferensial semantik dipilih sebagai format item, karena ini memungkinkan respons yang cepat dan intuitif. Setiap item UEQ terdiri dari sepasang istilah dengan makna yang berlawanan. Contoh :

*Not Understandable    o o o o o o    Understandable*  
*Efficient    o o o o o o    Inefficient*

Setiap item dapat dinilai pada skala Likert 7 poin. Jawaban untuk suatu item berkisar dari -3 (sepenuhnya setuju dengan istilah negatif) hingga +3 (sepenuhnya setuju dengan istilah positif). Setengah dari item mulai dengan istilah positif, sisanya dengan istilah negatif (dalam urutan acak).



Gambar 2. 4 Scale structure of the UEQ

Analisis ini menghasilkan kuesioner akhir dengan 26 item dikelompokkan ke dalam enam skala:

- a. Attractiveness: Kesan keseluruhan produk. Apakah pengguna suka atau tidak suka? Apakah itu menarik, menyenangkan atau menyenangkan?
  - 6 items: *annoying / enjoyable, good / bad, unlikable / pleasing, unpleasant / pleasant, attractive / unattractive, friendly / unfriendly.*
- b. Perspicuity: Apakah mudah mengenal produk? Apakah mudah dipelajari? Apakah produk mudah dimengerti dan jelas?
  - 4 items: *not understandable / understandable, easy to learn / difficult to learn, complicated / easy, clear / confusing.*
- c. Efisiensi: Dapatkah pengguna menyelesaikan tugas mereka tanpa usaha yang tidak perlu? Apakah interaksinya efisien dan cepat? Apakah produk bereaksi cepat terhadap input pengguna?

- 4 items: *fast / slow, inefficient / efficient, impractical / practical, organized / cluttered.*
- d. Dependability: Apakah pengguna merasa dapat mengendalikan interaksi? Bisakah dia memprediksi perilaku sistem? Apakah pengguna merasa aman saat bekerja dengan produk?
  - 4 items: *unpredictable / predictable, obstructive / supportive, secure / not secure, meets expectations / does not meet expectations*
- e. Stimulasi: Apakah mengasyikkan dan memotivasi untuk menggunakan produk? Apakah menyenangkan untuk digunakan?
  - 4 items: *valuable / inferior, boring / exciting, not interesting / interesting, motivating / demotivating.*
- f. Novelty (Baru): Apakah produknya inovatif dan kreatif? Apakah itu menarik perhatian pengguna?
  - 4 items: *creative / dull, inventive / conventional, usual / leadingedge, conservative / innovative.*

*Tabel 2. 1 Contoh hasil kategori nilai skala*

	<b>Att.</b>	<b>Eff.</b>	<b>Per.</b>	<b>Dep.</b>	<b>Sti.</b>	<b>Nov.</b>
Excellent	$\geq 1.75$	$\geq 1.78$	$\geq 1.9$	$\geq 1.65$	$\geq 1.55$	$\geq 1.4$
Good	$\geq 1.52$ $< 1.75$	$\geq 1.47$ $< 1.78$	$\geq 1.56$ $< 1.9$	$\geq 1.48$ $< 1.65$	$\geq 1.31$ $< 1.55$	$\geq 1.05$ $< 1.4$
Above average	$\geq 1.17$ $< 1.52$	$\geq 0.98$ $< 1.47$	$\geq 1.08$ $< 1.56$	$\geq 1.14$ $< 1.48$	$\geq 0.99$ $< 1.31$	$\geq 0.71$ $< 1.05$
Below average	$\geq 0.7$ $< 1.17$	$\geq 0.54$ $< 0.98$	$\geq 0.64$ $< 1.08$	$\geq 0.78$ $< 1.14$	$\geq 0.5$ $< 0.99$	$\geq 0.3$ $< 0.71$
Bad	$< 0.7$	$< 0.54$	$< 0.64$	$< 0.78$	$< 0.5$	$< 0.3$

#### 2.2.10 Quality of Service

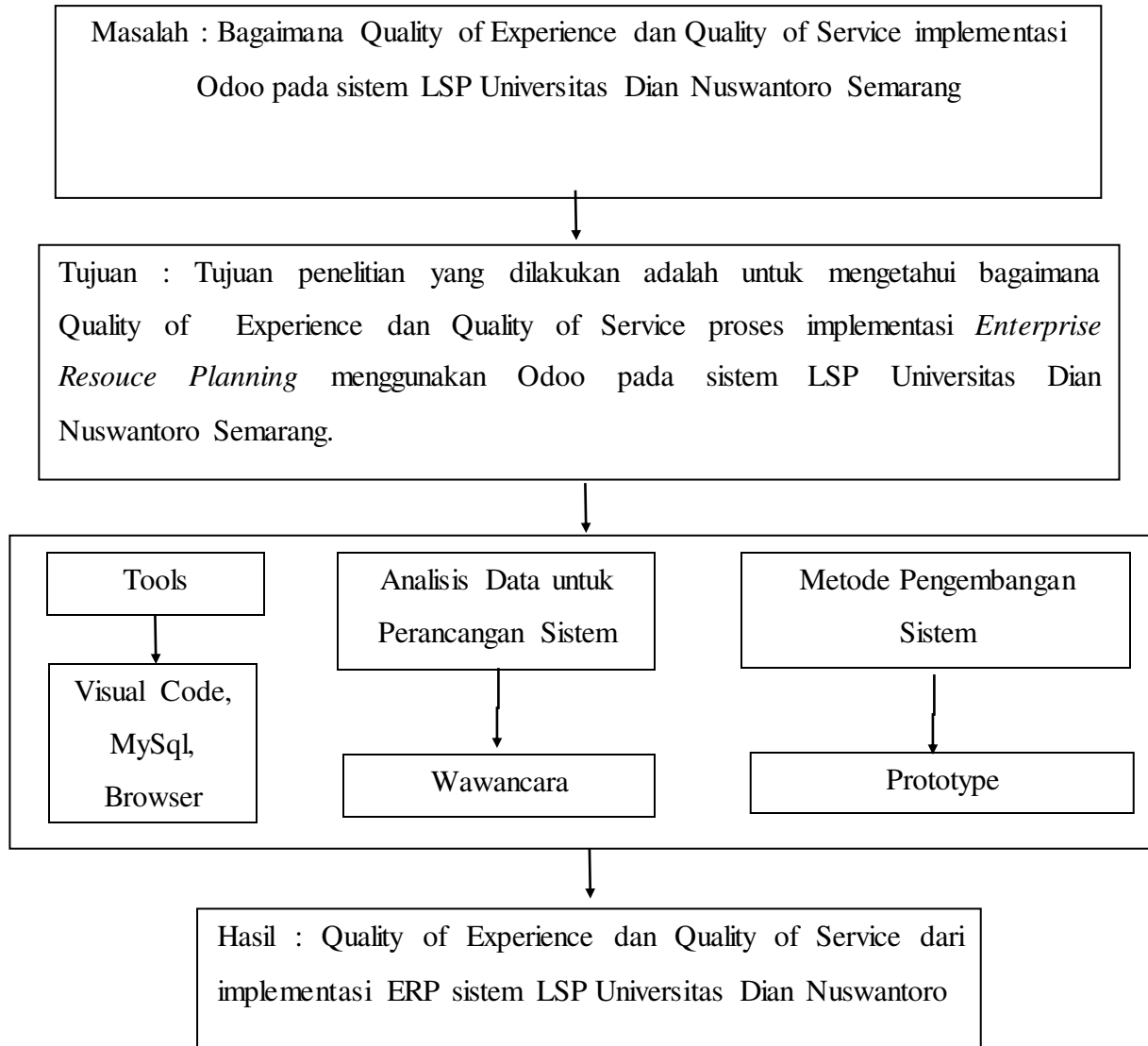
QOS [19] adalah ukuran kinerja pada tingkat paket dari perspektif jaringan dan kinerja perangkat lain yang terlibat dalam layanan ini. Selain itu, QOS mencakup kumpulan teknologi, yang juga dikenal sebagai mekanisme QOS, yang merupakan tools yang berguna di

tangan administrator jaringan karena dengan bantuannya dapat mengatasi adanya potensi kemacetan yang akan terjadi pada kinerja aplikasi. Dan terlebih lagi dengan mekanisme tersebut, dimungkinkan untuk menangani arus lalu lintas/ trafik jaringan yang berbeda secara spesifik atau sejumlah pengguna/user akhir dengan memberikan tingkat pelayanan lain yang lebih baik. Dalam istilah awam, kita dapat menggambarkan QOS sebagai kemampuan untuk menawarkan prioritas diantara pengguna/user akhir yang berbeda, aplikasi dan arus data yang berbeda. Selanjutnya, parameter yang digunakan untuk menjamin tingkat QOS tertentu adalah network centric seperti jitter, delay, bit error rate, bit rate, packet loss rate, dimana dapat mempengaruhi kualitas layanan tertentu yang dirasakan.

*Tabel 2. 2 Contoh parameter QoS*

QoS parameter and range			Note	
Parameter	Range	Score		
Delay (D)	70ms ~ 100ms	10	Y.1541, G.1010	
	100ms ~ 150ms	8		
	150ms ~ 200ms	6		
Jitter (J)	30ms ~ 50ms	10	Y.1541	
	50ms ~ 60ms	5		
	60ms ~ 70ms	3		
Packet loss rate (L)	$\sim 10^{-5}$	10	Y.1541	
	$10^{-5} \sim 10^{-4}$	7	DSL-	
	$10^{-4} \sim 10^{-3}$	3	Forum	
Packet error rate (E)	$\sim 10^{-6}$	10	Y.1541	
	$10^{-6} \sim 10^{-5}$	7	DSL-	
	$10^{-5} \sim 10^{-4}$	4	Forum	
Bandwidth (B)	Voice	256 ~ 128Kbps	10	G.1010 DSL- Forum WiMAX
		128 ~ 80Kbps	8	
		80 ~ 64Kbps	6	
	Video	3 ~ 2Mbps	10	
		2 ~ 1Mbps	8	
		1Mbps~512Kbps	5	
Call success rate (S)	100 ~ 99.9%		10	Undefined
	99.9% ~ 99.0%		9	
	99.0% ~ 98.0%		8	
Handover (H)	HO within the cell		N/A	Undefined
	HO between the cells		N/A	
	HO between the frequencies		N/A	
Channel	RF Power, RSSI,		N/A	Undefined

### 2.3 Kerangka Pemikiran



## **BAB III**

### **METODE PENELITIAN**

Metode penelitian yang digunakan dalam penulisan Tugas Akhir ini, sebagai berikut:

#### **3.1 Instrumen Penelitian**

Pada bab sebelumnya telah dibahas permasalahan yang telah diuraikan, maka diperlukan bahan dan peralatan yang dapat membantu dalam proses kelancaran pembuatan penelitian yang meliputi :

##### **3.1.1 Bahan**

Dalam penelitian ini bahan yang dibutuhkan dan digunakan adalah analisis data dan proses sertifikasi pada LSP Universitas Dian Nuswantoro.

##### **3.1.2 Peralatan**

Dalam penelitian ini memerlukan peralatan yang meliputi kebutuhan *hardware* dan kebutuhan *software* terdiri sebagai berikut :

###### **1.1.1.1 Identifikasi *Hardware***

Perangkat keras yang digunakan dalam penelitian Tugas Akhir dengan spesifikasi :

- Processor Intel Core i7-4700 HQ
- RAM 8 GB
- HDD 1 TB

###### **1.1.1.2 Identifikasi *Software***

Perangkat lunak yang digunakan dalam penelitian Tugas Akhir ini sebagai berikut :

###### **1. Sistem Operasi**

Sistem operasi yang digunakan dalam penelitian ini adalah Microsoft Winfows 10 Enterprise.

###### **2. Microsoft Office Word 2016**



Software ini digunakan untuk penulisan laporan hasil penelitian.

### 3. Visual Studio Code

Visual Studio Code digunakan dalam penyusunan kode program.

### 4. Module Odoo

Module Odoo merupakan modul utama untuk penyusunan kode program.

### 5. MySql

MySql berperan untuk pembuatan database.

### 6. Google Chrome

Google chrome digunakan untuk menampilkan hasil dari interface.

## 3.2 Metode Analisa Data

Metode yang digunakan dalam proses ini adalah dengan wawancara. Pada tahap wawancara kali ini, dilakukan langsung kepada pihak yang bersangkutan, yaitu pihak LSP Universitas Dian Nuswantoro Semarang. Hasil dari wawancara ini merupakan alur lengkap proses sertifikasi serta spesifikasi kebutuhan perangkat lunak yang nantinya data analisis ini dapat dijadikan acuan untuk membuat rancangan sistem.

## 3.3 Tinjauan Pustaka

Selain menggunakan metode analisa data berupa wawancara langsung dengan objek penelitian, ada juga metode pengumpulan studi pustaka yang digunakan untuk mencari sumber data dari jurnal, buku, dan media lain yang berhubungan dengan penelitian. Peneliti mencari di beberapa sumber seperti IEEE, Scopus, Sciencedirect dengan rentang tahun 2013-2018 dan keyword *Enterprise Resource Planning, Odoo, Enterprise Resource Planning + Odoo* , *White Box Testing, Black Box Testing, Quality of Service, Quality of*

*Experience, Quality of Service + Quality of Experience, User Experience Questionnaires.*

### 3.4 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan adalah *Prototype*. Adapun tahapan – tahapan dalam proses pengembanganya sebagai berikut :

1. *Planning*

Tahap *planning* adalah tahap awal, dimana pada tahap ini penulis mulai mendefinisikan spesifikasi kebutuhan perangkat lunak yang diperlukan dan mengumpulkan data analisis dari tahap wawancara.

2. *Modeling*

Pada tahap ini penulis mulai membuat rancangan - rancangan sistem LSP Universitas Dian Nuswantoro berdasarkan data analisis yang sudah didapat.

3. *Construction*

Pada tahap ini mulai dibuat interface – interface tiap modul sistem LSP, untuk kemudian di diskusikan dengan pihak LSP Universitas Dian Nuswantoro mengenai alur input dan output dari modul tersebut.

4. *Deployment*

Tahap ini merupakan implementasi dimana penulis mulai menerapkan tahapan *modeling* dan *construction* untuk diterjemahkan ke bahasa pemrograman yang telah ditentukan.

5. *Communication*

Tahapan terakhir dimana penulis mulai menguji coba sistem yang telah dibuat pada pihak LSP Universitas Dian Nuswantoro berdasarkan *Quality of Experience* dan *User Experience Questionnaires* yang telah ditentukan sebagai parameter keberhasilan sistem.

### 3.5 Pengujian Software

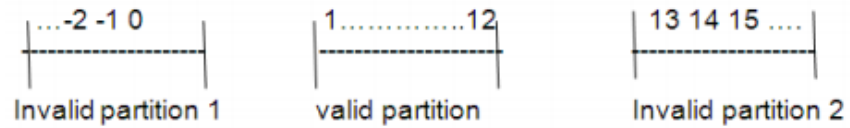
Dalam tahap proses pengujian sistem LSP Universitas Dian Nuswantoro, dilakukan dengan pengujian sebagai berikut :

#### 3.5.1 Black Box Testing

Pengujian *Black Box* pada perangkat lunak (*software*) dilakukan pada komponen antarmuka aplikasi. Teknik *black box* yang digunakan yaitu *Equivalence Class Partitioning*.

Equivalence partising (EP) adalah teknik berbasis spesifikasi atau *black box*. Gagasan di balik teknik ini adalah untuk membagi (yaitu mempartisi) satu set kondisi pengujian ke dalam kelompok atau set yang dapat dianggap sama (mis. Sistem harus menanganinya secara setara), karenanya partition mempartisi kesetaraan '. Dalam teknik partisi-ekivalensi kita perlu menguji hanya satu syarat dari setiap partisi. Ini karena penulis mengasumsikan bahwa semua kondisi dalam satu partisi akan diperlakukan dengan cara yang sama oleh perangkat lunak. Jika satu kondisi dalam suatu partisi berfungsi, penulis menganggap semua kondisi di partisi itu akan berfungsi. Demikian pula, jika salah satu kondisi dalam suatu partisi tidak berfungsi, maka penulis mengasumsikan bahwa tidak ada satu pun kondisi di dalam partisi tersebut yang berfungsi, sehingga tidak ada gunanya menguji lagi di partisi tersebut.

Ini mungkin paling baik dijelaskan dengan contoh yang mengambil parameter "bulan". Kisaran yang valid untuk bulan tersebut adalah 1 hingga 12, mewakili Januari hingga Desember. Kisaran yang valid ini disebut partisi. Dalam contoh ini ada dua partisi lebih lanjut dari rentang tidak valid. Partisi tidak valid pertama adalah  $\leq 0$  dan partisi tidak valid kedua adalah  $> 13$ .



Gambar 3. 1 Equivalence Partitioning Blackbox

### 3.5.2 White Box Testing

Pengujian *White Box* pada perangkat lunak (*software*) dilakukan menggunakan teknik *Basis Path*. Pengujian ini dilakukan bertujuan untuk mengukur kompleksitas logis desain procedural. Ada beberapa tahap dalam pengujian *basis path*, antara lain:

1. Transformasikan *source code* kedalam flow graph
2. Menghitung *cyclomatic complexity*
3. Menentukan jalur
4. Membuat desain *test case*
5. Graph matrix

### 3.5.3 Quality of Experience

Pada pengujian Quality of Experience ini, sistem LSP yang telah dibuat harus diuji kesesuaiannya sebagai tolak ukur keberhasilan sistem. Pengujian QoE ini ditujukan untuk membantu menemukan *error* yang tidak ditemukan oleh penguji. Proses pengujian QoE ini melibatkan pihak LSP Universitas Dian Nuswantoro dan calon user baru yang nantinya akan menggunakan sistem ini.

Pada tahap QoE, penulis menggunakan *User Experience Questionnaire* sebagai penunjang pengujian. Indikator yang digunakan pada pengujian ini menggunakan skala Likert 7 poin dengan 6 parameter keberhasilan QoE.

Tabel 3. 1 Contoh Kuisisioner QoE

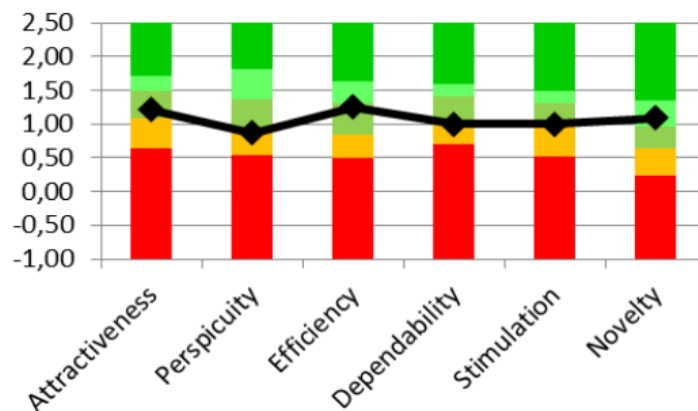
No	Pertanyaan	1	2	3	4	5	6	7
1	Apakah Sistem LSP Universitas Dian Nuswantoro dapat dioperasikan dengan mudah?							
2	Apakah semua menu dan fungsi pada sistem LSP Universitas Dian Nuswantoro dapat dijalankan dengan baik?							
3	Apakah sistem LSP Universitas Dian Nuswantoro dapat dijalankan dengan efisien?							
4	Apakah sistem LSP Universitas Dian Nuswantoro sudah cukup aman untuk digunakan?							
5	Apakah tampilan pada sistem LSP Universitas Dian Nuswantoro cukup menyenangkan untuk digunakan?							
6	Apakah sistem LSP Universitas Dian Nuswantoro sudah bisa dikatakan inventif?							

Setelah kuisioner diatas diberikan ke penguji dan data dari kuisioner telah diisi, maka data diolah untuk mendapatkan hasil penilaian *Quality of Experience*. Berikut adalah contoh table Evaluasi QoE.

*Tabel 3. 2 Contoh Evaluasi QoE*

No	Jawaban Penguji													
	1	%	2	%	3	%	4	%	5	%	6	%	7	%
1														
2														
3														
4														
5														
6														
7														
8														
9														
10														
Total														

Dari hasil penilaian dengan pengujian *Quality of Experience*, maka nantinya dapat ditarik kesimpulan apakah sistem LSP Universitas Dian Nuswantoro telah memenuhi keberhasilan parameter QoE atau tidak. Berikut adalah contoh Diagram Evaluasi QoE.



Gambar 3. 2 Contoh Diagram Evaluasi QoE

### 3.5.4 Quality of Service

Pengujian QoS merupakan tahapan pengujian ukuran kinerja pada tingkat paket dari perspektif jaringan dan kinerja perangkat lain yang terlibat dalam sistem. Parameter yang digunakan untuk menjamin tingkat QoS tertentu diantaranya yaitu *network centric* seperti *jitter*, *delay*, *bit error rate*, *bit rate*, *packet loss rate*, dimana dapat mempengaruhi kualitas layanan tertentu yang dirasakan.

Pada QoS ini penulis hanya mengambil 2 parameter, yaitu *delay* dan *packet loss*. Berikut rumus perhitungan 2 parameter tersebut :

a. Delay

$$\text{Rata-rata Delay} = \text{Total Delay} / \text{Total paket yang diterima}$$

b. Packet loss

$$\text{Packet loss} = \left( \frac{\text{data dikirim} - \text{paket data terima}}{\text{paket data kirim}} \right) \times 100\%$$

## DAFTAR PUSTAKA

- [1] “Badan Nasional Sertifikasi Profesi” [Online]. Available: <https://bnsf.go.id/utama>. [Accessed: 04 November 2018].
- [2] “Media Informasi Kinerja Perusahaan Indonesia” [Online]. Available: <http://annualreport.id/kiat-strategi/pentingnya-sertifikasi-profesi-bagi-karyawan>. [Accessed: 04 November 2018].
- [3] “Lembaga Sertifikasi Profesi Universitas Dian Nuswantoro” [Online]. Available: <http://lsp.dinus.ac.id/index.php/profil-lsp-udinus/>. [Accessed: 04 November 2018].
- [4] Ganesh A., K.N. Shanil, C. Sunitha, and A.M. Midhundas, “OpenERP/Odoo – An Open Source Concept to ERP Solution,” in *2016 IEEE 6th International Conference on Advanced Computing*, 2016.
- [5] Elragal A. and Haddara M., “The Future of ERP Systems: look backward before moving forward,” in *CENTERIS 2012 - Conference on ENTERprise Information Systems / HCIST 2012 - International Conference on Health and Social Care Information Systems and Technologies*, 2012.
- [6] Aversano L., Di Brino M., Guardabascio D., Salerno M., and Tortorella M., “Understanding Enterprise Open Source Software Evolution,” in *Conference on ENTERprise Information Systems / International Conference on Project MANagement / Conference on Health and Social Care Information Systems and Technologies, CENTERIS / ProjMAN / HCist 2015 October 7-9, 2015*, 2015.
- [7] Matende S. and Ogao P., “Enterprise Resource Planning (ERP) System Implementation: A case for User participation,” in *CENTERIS 2013 - Conference on ENTERprise Information Systems / PROjMAN 2013 - International Conference on Project MANagement / HCIST 2013 - International Conference on Health and Social Care Information Systems and Technologies*, 2013.
- [8] Hustad E., Haddara M., and Kalvenes B., “ERP and organizational misfits: An ERP customization journey,” in *Conference on ENTERprise*



- Information Systems / International Conference on Project MANagement / Conference on Health and Social Care Information Systems and Technologies, CENTERIS / ProjMAN / HCist 2016, October 5-7, 2016, 2016.*
- [9] Haddara M. and Constantini A., “ERP II is Dead- Long Live CRM,” in *CENTERIS - International Conference on ENTERprise Information System / ProjMAN - International Conference on Project MANagement / HCist - International Conference on Health and Social Care Information Systems and Technologies, CENTERIS / ProjMAN / HCist 2017, 8-10 November 2017, Barcelona, Spain, 2017*
  - [10] MHD Fawaz Enaya. 2016. “An Experimental Performance Comparison of NoSQL and RDBMS Data Storage Systems in the ERP System Odoo”. *Thesis*. University of Magdeburg, School of Computer Science.
  - [11] “Odoo” [Online]. Available: <https://www.odoo.com/>. [Accessed : 23 November 2018].
  - [12] Almugadam S.H., Bashir B.I., Hassan A.A., and Adam M.A., “Developing Tool For Odoo Platform,” in *2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE), Khartoum, Sudan, 2017.*
  - [13] “Python Documentation” [Online]. Available: <https://docs.python.org/2/faq/general.html>. [Accessed: 23 November 2018].
  - [14] Booch G., Rumbaugh J., and Jacobson I. 2005. *The Unified Modeling Language User Guide*. France: Addison-Wesley Professional.
  - [15] Galin, Daniel. 2018. *Software Quality: Concepts and Practice*, First Edition. United States: IEEE Computer Society Press.
  - [16] Susanto R., and Andriana A.D., “Perbandingan Model Waterfall dan Prototyping untuk Pengembangan Sistem Informasi,” in *Majalah Ilmiah UNIKOM Vol. 14*, 2010.
  - [17] Kumar M., Singh S.K., and Dwivedi Dr. R.K., "A Comparative Study of Black Box Testing and White Box Testing Techniques," in *International*

*Journal of Advance Research in Computer Science and Management Studies*, 2015.

- [18] Khan Mohd. E., "Different Approaches to Black Box Testing Technique for Finding Errors," in *International Journal of Software Engineering & Applications (IJSEA)*, Vol.2, No.4, October 2011, 2011.
- [19] Indarto E., Nugroho L. Edi, and Fauziati S., " Quality of Experience (QOE) Assessment pada layanan Broadband Internet personal, keluarga di lingkungan Urban dan Program Desa Internet (PLIK)," in *CITEE 2017*, 2017.
- [20] Schrepp M., Cota M.P., Goncalves R., Hinderks A., and Thomaschewski J., "Adaption of user experience questionnaires for different user groups," in *Universal Access in the Information Society Volume 16 Issue 3, August 2017*, 2017.
- [21] Schrepp M., Hinderks A., and Thomaschewski J., "Construction of a Benchmark for the User Experience Questionnaire (UEQ)," in *International Journal of Interactive Multimedia and Artificial Intelligence*, Vol. 4, No 4, 2017.