

**RANCANG BANGUN SISTEM INFORMASI *TRY OUT* UNTUK SOAL
CPNS BERBASIS *WEB***

PROPOSAL SKRIPSI

Dibuat sebagai syarat pelaksanaan penelitian mahasiswa program sarjana

Oleh :

Rifan Alamsyah

NIM : 1606020



**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TINGGI TEKNOLOGI GARUT**

2020

LEMBAR PENGESAHAN

**RANCANG BANGUN SISTEM INFORMASI *TRY OUT* UNTUK SOAL
CPNS BERBASIS *WEB***

PROPOSAL SKRIPSI

Oleh:

Rifan Alamsyah

NIM: 1606020

Proposal Skripsi ini telah diperiksa dan disetujui oleh:
Pembimbing,

Asep Deddy S., M.Kom
NIDN: 0417026202

Mengetahui,
Ketua Program Studi Teknik Informatika,

Dede Kurniadi, S.Kom., M.Kom
NIDN: 0402098301

RINGKASAN

Ringkasan maksimum satu halaman. Kemukakan tujuan jangka panjang dan target khusus yang ingin dicapai serta metode yang akan dipakai dalam pencapaian tujuan tersebut. Ringkasan harus mampu menguraikan secara cermat dan singkat tentang rencana kegiatan yang diusulkan mengenai gambaran penelitian yang akan dilakukan kedepan mencakup ringkasan latar belakang permasalahan, tujuan, cara pemecahan masalah dan teknik/metode yang digunakannya, serta hasil yang ingin dicapai dan manfaatnya.

KATA PENGANTAR

Template ini dibuat untuk memudahkan mahasiswa program studi Informatika Sekolah Tinggi Teknologi Garut dalam penyusunan proposal skripsi. Bacalah penjelasan yang tertuang dalam template ini agar proposal sesuai harapan program studi selaku pelaksana penelitian bidang Teknik Informatika. Mahasiswa hanya perlu mengubah isi dari setiap bagian atau sub bagian; menambah dan menyisipkan sumber sitasi yang diperlukan; menambah sub bagian yang diperlukan; serta memperbaharui daftar isi, daftar gambar, daftar tabel, dan daftar pustaka. Pastikan sub bagian yang dibuat hanya sampai tiga tingkatan saja (misal : 1.1.1.1) serta ada jarak 1,5 spasi antara bagian dengan sub bagian dan antar sub bagian.

Proposal skripsi ini merupakan luaran tahap rencana penelitian, dan merupakan tiga dari lima bagian laporan Skripsi. Penelitian yang direncanakan harus sejalan dengan visi program studi, yakni “Menjadi Penyelenggara Program Studi Informatika Yang Unggul di Bidang Rekayasa Perangkat Lunak dengan Luaran Yang Berdaya Saing Global Berbasis Kearifan Lokal pada Tahun 2030”; dan misi penelitiannya, yakni “Menggalakkan penelitian untuk pengembangan ilmu pengetahuan dan teknologi berbasis *online* yang dapat meningkatkan kesejahteraan dan daya saing global”.

Garut, 2 Februari 2020

Penyusun

DAFTAR ISI

LEMBAR PENGESAHAN	
RINGKASAN	i
KATA PENGANTAR	ii
DAFTAR ISI.....	ivi
DAFTAR GAMBAR	vv
DAFTAR TABEL.....	vi
1. PENDAHULUAN	7
1.1. Latar Belakang	7
1.2. Masalah Penelitian	9
1.3. Tujuan Penelitian.....	9
1.4. Pertanyaan Penelitian	9
1.5. Cakupan Penelitian.....	10
1.6. Manfaat Penelitian.....	10
2. TINJAUAN PUSTAKA.....	Error! Bookmark not defined.
3. METODOLOGI PENELITIAN.....	Error! Bookmark not defined.
4. JADWAL RENCANA KEGIATAN PENELITIAN	10
DAFTAR PUSTAKA	11

DAFTAR GAMBAR

Gambar 2.1. Logo Sekolah Tinggi Teknologi Garut	Error! Bookmark not defined.
Gambar 3.1. Kerangka Pemikiran	6
Gambar 3.2. Contoh <i>Work Breakdown Structure</i> ..	Error! Bookmark not defined.
Gambar 3.3. Contoh Diagram Alur Aktivitas	8

DAFTAR TABEL

Tabel 2.1. Contoh Tabel.....	Error! Bookmark not defined.
Tabel 3.1. <i>Detail Activity</i>	9
Tabel 3.2. Sumber Daya Penelitian.....	9

1. PENDAHULUAN

1.1. Latar Belakang

Pegawai Negeri Sipil (PNS) menurut peraturan pemerintah nomor 11 tahun 2017 tentang manajemen pegawai negeri sipil adalah warga negara Indonesia yang memenuhi syarat tertentu, diangkat sebagai Pegawai ASN secara tetap oleh pejabat pembina kepegawaian untuk menduduki jabatan pemerintahan. Menurut data Badan Kepegawaian dan Diklat (BKD) Garut pada tahun 2019 terdapat 21.703 orang mendaftarkan diri sebagai Calon PNS (CPNS) akan tetapi formasi kebutuhan CPNS di lingkungan pemerintahan kabupaten Garut tahun anggaran 2019 sebanyak 838 formasi, dikarenakan kuota penerimaan formasi yang terbatas membuat seleksi CPNS menjadi hal yang sulit. Faktor jenis soal pada *test* CPNS juga membuat CPNS harus belajar materi yang beragam, menurut peraturan menteri pendayagunaan aparatur negara dan reformasi republik Indonesia *test* CPNS memiliki 3 jenis soal yakni meliputi Test Wawasan Kebangsaan (TWK) sebanyak 35 soal, Test Intelegensi Umum (TIU) sebanyak 30 soal dan Tes Karakteristik Pribadi (TKP) sebanyak 35 soal, oleh karena banyaknya soal yang harus dipelajari peserta calon PNS (CPNS) dituntut untuk memperbanyak latihan soal CPNS, baik dengan mengerjakan soal pada buku CPNS yang tersedia di toko-toko ataupun dengan mengerjakan soal dari aplikasi yang tersedia di paket pembelian buku yang memiliki sistem *Try Out* berbasis *Computer Asseted Test* (CAT) sehingga CPNS dapat belajar sesuai dengan *test* CPNS yang sesungguhnya. *Try out* adalah suatu mekanisme yang digunakan sebagai sebuah latihan bagi siswa sebelum melaksanakan ujian yang sesungguhnya (Aisah & Haryati, 2019). Dengan sistem informasi *Try Out* ini diharapkan dapat membantu CPNS untuk berlatih mengerjakan soal serupa yang di prediksi akan ada pada *test* CPNS sesungguhnya, sehingga nantinya memberikan kesempatan lebih besar untuk lulus.

Pada penelitian sebelumnya yang berjudul “Rancang Bangun Aplikasi *Bank Soal* Program Studi Teknik Informatika Sekolah Tinggi Teknologi Garut” menjelaskan bahwa hasil dari penelitian tersebut adalah aplikasi *Bank Soal* untuk menyimpan soal-soal hasil evaluasi (Azizah & Fitriani, 2016). Penelitian kedua yang berjudul “Perancangan Aplikasi Survei Kepuasan Mahasiswa Berbasis

Kuesioner *Online*” memiliki hasil Aplikasi Survei Kepuasan Mahasiswa berbasis kuesioner *online* untuk memfasilitasi mahasiswa dalam melakukan proses pengisian survei sehingga mahasiswa bisa mengisi survei dimana saja dan kapan saja tanpa ada batasan ruang dan waktu (Kurniadi & Islami, 2018). Pada penelitian ketiga yang berjudul “Aplikasi Tryout Ujian *Online* untuk SMA/SMK Sederajat (*Online Examination Tryout Application for Senior and Vocational High School Students*)”, membahas tentang program tryout yang digunakan sebagai sarana latihan siswa dalam menghadapi ujian nasional (Kumalasari, Erika, & Mustikasari, 2017). Pada penelitian keempat yang berjudul “Pembuatan Aplikasi Sistem Ujian *Online* Pada SMK Garuda Nusantara Bekasi”, pada penelitian ini membuat aplikasi ujian *online* sehingga memberi kemudahan bagi siswa, guru dan pengawas ujian dalam pelaksanaan dan pengolahan data dan hasil ujian (Riyadi, Hermaliani, & Utami, 2019). Pada penelitian terakhir yang berjudul “Aplikasi Learning Manajemen Sistem Dan Ulangan *Online* Berbasis Web” menjelaskan bahwa hasil penelitian menyediakan fitur-fitur untuk membantu guru dalam mengelola pengumuman, materi, tugas dan ulangan. Aplikasi ini juga membantu siswa dalam mengunduh materi, mengunggah pengumpulan tugas, melaksanakan ulangan secara *online* dan melihat nilai-nilai tugas dan ulangan (Ekaputri, Suryatiningsih, & Siswanto, 2016). Berdasarkan penelitian-penelitian yang dirujuk terlihat bahwa terdapat konsep seperti *bank* soal dan kuisisioner yang memiliki sistem tanya jawab yang akan berguna dalam pembuatan sistem penyimpanan dan struktur soal dan jawaban, serta pada penelitian yang dirujuk juga memiliki konsep *try out* dan ulangan berbasis *online* yang akan memudahkan pengguna dalam melakukan proses *try out* atau ulangan tanpa harus mengerjakan dalam kertas, dan soalpun dapat dinilai secara otomatis sehingga dapat memudahkan penilai atau guru. Pada penelitian yang dirujuk juga memiliki persamaan pembuatan basis aplikasi yakni berbasis *web*, oleh karena itu penulis akan membangun aplikasi berbasis *web* dengan alasan aplikasi akan lebih fleksibel untuk diakses dimana saja dan pada perangkat apa saja tanpa melakukan instalasi terlebih dahulu, serta aplikasi akan memiliki tampilan yang dapat menyesuaikan ukuran layar secara responsif sehingga pengguna dapat menggunakan aplikasi secara maksimal di perangkat masing-masing. Selain itu aplikasi *try out* yang akan dirancang di desain

menyerupai sistem CAT dengan fitur batas waktu pengerjaan, sehingga dapat melatih para peserta CPNS untuk mengelola waktu pengerjaan soal. Merujuk pada kelima penelitian sebelumnya, penulis mengambil judul penelitian Rancang Bangun Sistem Informasi *Try Out* Untuk Soal CPNS Berbasis *Web*.

1.2. Masalah Penelitian

Berdasarkan latar belakang serta jurnal rujukan yang diambil, terdapat beberapa masalah yaitu sebagai berikut:

1. Pada jurnal yang dirujuk, aplikasi belum fitur pembahasan setelah mengerjakan *try out* atau ulangan *online*.
2. Aplikasi belum memiliki informasi lanjutan dari nilai hasil *try out* atau ulangan.
3. Waktu pengerjaan *try out* atau ulangan pada aplikasi tidak fleksibel.

1.3. Tujuan Penelitian

Adapun tujuan dari penelitian ini dapat dirumuskan diantaranya adalah untuk:

1. Untuk membuat rancang bangun sistem informasi *try out* untuk soal CPNS berbasis *web*.
2. Untuk memudahkan peserta CPNS mendapatkan latihan soal sesuai dengan *test* CPNS yang sesungguhnya.

1.4. Pertanyaan Penelitian

Berdasarkan Masalah Penelitian yang ada diatas terdapat masalah yaitu, diantaranya :

1. Bagaimana membuat rancang bangun sistem informasi *try out* untuk soal CPNS?
2. Bagaimana membuat sistem informasi dengan latihan soal yang sesuai dengan *test* CPNS?

1.5. Cakupan Penelitian

Agar dalam pembahasan ini dapat mengarah pada tujuan yang telah direncanakan, untuk menghindari pembahasan yang bisa melebar, maka penulis membatasi laporan Skripsi ini kedalam beberapa cakupan penelitian ini, adalah sebagai berikut:

1. Aplikasi *Try Out* soal CPNS;
2. Metodologi yang digunakan pada perancangan sistem informasi *try out* untuk soal CPNS berbasis *web* ini menggunakan *Rational Unified Process* (RUP);
3. Dalam aplikasi menerapkan sistem *try out* yang memiliki batas waktu pengerjaan soal;
4. Memiliki bank soal yang disertai dengan penjelasan tiap soal;
5. Memiliki fitur review jawaban ketika selesai mengerjakan *try out*;
6. Pengguna aplikasi ini adalah masyarakat yang ingin belajar soal CPNS khususnya akan mengikuti ujian CPNS; dan
7. Bahasa pemrograman yang digunakan adalah PHP dan javascript serta MySQL sebagai penyimpanan *database*.

1.6. Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Bagi pengguna (masyarakat) selaku pengguna akhir dari aplikasi ini mendapatkan informasi seputar *try out* atau ulangan online soal CPNS yang dapat diakses dimana saja dan kapan saja; dan
2. Bagi pemerintahan, dengan adanya aplikasi ini diharapkan dapat membantu meningkatkan kualitas CPNS yang akan menjadi Aparatur Sipil Negara yang telah terlatih dan memiliki wawasan yang luas.

1.7. Sistematika

Proposal skripsi ini disusun dengan sistematika sebagai berikut:

1. **PENDAHULUAN**, berisi latar belakang, masalah, tujuan, pertanyaan, cakupan, dan manfaat penelitian, serta sistematika proposal;

2. **TINJAUAN PUSTAKA**, berisi analisis rinci terhadap penelitian sebelumnya dan hubungannya dengan penelitian yang akan dilaksanakan;
3. **METODOLOGI PENELITIAN**, berisi kerangka pemikiran, beserta penjelasan tentang tahapan penelitian berikut aktivitas dan tekniknya, serta waktu, tempat, dan sumber daya yang digunakan.
4. **JADWAL RENCANA KEGIATAN PENELITIAN**, berisi rencana jadwal penelitian secara detail disajikan dalam bentuk tabel.

2. TINJAUAN PUSTAKA

2.1. Penelitian Rujukan

Dalam penyusunan penelitian ini merujuk dari jurnal atau hasil penelitian sebelumnya, dimana terdapat beberapa keterkaitan dalam penelitian yang dilakukan baik secara langsung ataupun secara tidak langsung. Hasil penelitian tersebut sebagai berikut :

1. Penelitian rujukan pertama berjudul “Rancang Bangun Aplikasi *Bank Soal* Program Studi Teknik Informatika Sekolah Tinggi Teknologi Garut”. Pada penelitian ini menggunakan metodologi berorientasi objek dengan pendekatan *Unified Software Development Process* dengan pemodelan *Unified Modeling Language*. Untuk bahasa pemrograman yang digunakan adalah PHP dan untuk *database* menggunakan MySQL. Adapun hasil dari penelitian ini menghasilkan aplikasi bank soal di program studi teknik informatika mampu memberikan solusi dalam mengatur penyimpanan soal-soal dengan baik dan dosen dapat menggunakan soal-soal sebelumnya yang pernah dibuat untuk digunakan pada waktu berikutnya serta sejarah pembuatan soal-soal dapat dicari dengan mudah;
2. Penelitian rujukan kedua berjudul “Perancangan Aplikasi Survei Kepuasan Mahasiswa Berbasis Kuesioner Online”. Pada penelitian ini menggunakan metodologi *Unified Approach* yang meliputi tahapan *Object Oriented Analysis*, tahapan *Object Oriented Design* serta menggunakan *Component Based Development* pada tahapan Implementasi, dan untuk pemodelannya menggunakan *Unified Modelling Language*. Untuk bahasa pemrograman yang digunakan adalah PHP dan untuk *database* menggunakan MySQL. Adapun hasil dari penelitian ini berupa Aplikasi Survei Kepuasan Mahasiswa berbasis kuesioner online untuk memfasilitasi mahasiswa dalam melakukan proses pengisian survei sehingga mahasiswa bisa mengisi survei dimana saja dan kapan saja tanpa ada batasan ruang dan waktu;
3. Penelitian rujukan ketiga berjudul “Aplikasi Tryout Ujian *Online* untuk SMA/SMK Sederajat (*Online Examination Tryout Application for Senior and Vocational High School Students*)”. Metodologi yang digunakan adalah *Waterfall*. Untuk bahasa pemograman yang digunakan adalah PHP

dan MySQL sebagai penyimpanan *database*. Hasil dari penelitian ini berupa Aplikasi Tryout ujian online dan Manual book untuk ditujukan untuk semua pengguna aplikasi, baik guru, siswa, maupun admin aplikasi;

4. Penelitian keempat berjudul “Pembuatan Aplikasi Sistem Ujian *Online* Pada SMK Garuda Nusantara Bekasi”. Metodologi yang digunakan adalah *Waterfall*. Untuk bahasa pemrograman yang digunakan adalah PHP dan penyimpanan *database* yang digunakan adalah MySQL. Hasil dari penelitian ini adalah aplikasi ujian online sehingga memberi kemudahan bagi siswa, guru dan pengawas ujian dalam pelaksanaan dan pengolahan data dan hasil ujian pada SMK Garuda Nusantara Bekasi.
5. Penelitian terakhir berjudul “Aplikasi Learning Manajemen Sistem Dan Ulangan Online Berbasis Web”. Metodologi yang digunakan pada penelitian ini adalah Prototipe (Prototyping Model). Bahasa pemrograman yang digunakan adalah PHP dengan menggunakan *framework CodeIgniter* dan untuk penyimpanan *database* menggunakan MySQL. Hasil dari penelitian ini berupa aplikasi Learning Manajemen Sistem dan Ulangan Online, untuk membantu pihak sekolah dan siswa dalam penyampaian pengumuman, pembagian materi yang diajarkan, pemberian dan pengumpulan tugas siswa, pelaksanaan dan pemeriksaan ulangan, serta merekap nilai ulangan siswa.

2.2. Kesenjangan Penelitian

Berdasarkan penelitian yang telah dirujuk sebelumnya dan sudah dijelaskan, maka terdapat kesenjangan dan permasalahan diantaranya :

Tabel 2.1. Kesenjangan Penelitian

No	Judul Penelitian	Cakupan Penelitian	Kesenjangan Penelitian
1	Rancang Bangun Aplikasi <i>Bank Soal</i> Program Studi Teknik Informatika Sekolah Tinggi Teknologi Garut	Perancangan sistem informasi dari aplikasi pada penelitian berfokus pada penyimpanan soal.	Jika pada penelitian sebelumnya memberikan kemudahan dalam penyimpanan soal, maka pada penelitian ini konsep dari soal tersebut dapat diimplemtasikan sehingga

No	Judul Penelitian	Cakupan Penelitian	Kesenjangan Penelitian
	(Azizah & Fitriani, 2016)		dapat digunakan dalam <i>try out</i>
2	Perancangan Aplikasi Survei Kepuasan Mahasiswa Berbasis Kuesioner Online (Kurniadi & Islami, 2018)	Aplikasi yang dirancang memiliki sistem informasi yang hanya mencakup pertanyaan dan jawaban serta rekap nilai pada bobot jawaban yang dipilih serta jawaban yang berbentuk nilai dari 1 sampai 6	Jika perekapan di penelitian sebelumnya perekapan nilai didapatkan dari bobot jawaban yang dipilih, maka pada penelitian ini perekapan didapatkan dari kebenaran dari jawaban yang dipilih serta jawaban berbentuk <i>text</i> ataupun gambar
3	Aplikasi Tryout Ujian <i>Online</i> untuk SMA/SMK Sederajat (<i>Online Examination Tryout Application for Senior and Vocational High School Students</i>) (Kumalasari, Erika, & Mustikasari, 2017)	Pada sistem informasi <i>try out</i> yang dirancang sebelumnya, peserta hanya dapat mengerjakan <i>try out</i> apabila telah dimulai oleh admin sehingga waktu pengerjaan tidak fleksibel	Jika waktu pengerjaan <i>try out</i> pada penelitian sebelumnya memiliki waktu pengerjaan yang terbatas dan tergantung admin, maka pada penelitian ini waktu pengerjaan dapat dilakukan kapan saja serta dapat diulangi sebanyak apapun peserta mau
4	Pembuatan Aplikasi Sistem Ujian <i>Online</i> Pada SMK Garuda Nusantara Bekasi (Riyadi, Hermaliani, & Utami, 2019)	Pada penelitian sebelumnya sistem informasi <i>try out</i> , jawaban haruslah berbentuk <i>text</i>	Jika pada penelitian sebelumnya jenis jawaban haruslah berbentuk <i>text</i> , maka pada penelitian ini jawaban dapat berbentuk <i>text</i> atau gambar sesuai dengan soal CPNS yang ditentukan oleh pemerintah
5	Aplikasi Learning Manajemen Sistem Dan Ulangan	Pada perancangan sistem informasi <i>try out</i> sebelumnya,	Jika pada penelitian sebelumnya tidak adanya fitur <i>review</i> jawaban

No	Judul Penelitian	Cakupan Penelitian	Kesenjangan Penelitian
	Online Berbasis Web (Ekaputri, Suryatiningsih, & Siswanto, 2016)	sistem <i>try out</i> tidak memiliki fitur <i>review</i> jawaban setelah menyelesaikan <i>try out</i>	setelah menyelesaikan <i>try out</i> , maka pada penelitian ini akan diterapkan fitur <i>review</i> jawaban sehingga peserta <i>try out</i> dapat mengetahui kekurangan dari hasil <i>try out</i>

Berdasarkan kesenjangan penelitian sebelumnya yang telah dijelaskan, maka penelitian yang selanjutnya yaitu Rancang Bangun Sistem Informasi *Try Out* Untuk Soal CPNS Berbasis *Web*, sehingga aplikasi akan memiliki fitur *try out* meliputi pengerjaan *try out*, *review* jawaban dan unduh materi CPNS. Aplikasi ini akan membantu masyarakat yang berniat mengikuti *test* CPNS untuk belajar dan mengerjakan soal *try out* berbasis *Computer Asseted Test* (CAT), dimana sistem CAT ini digunakan pada *test* CPNS yang sesungguhnya.

2.3. Konsep Dasar Perancangan

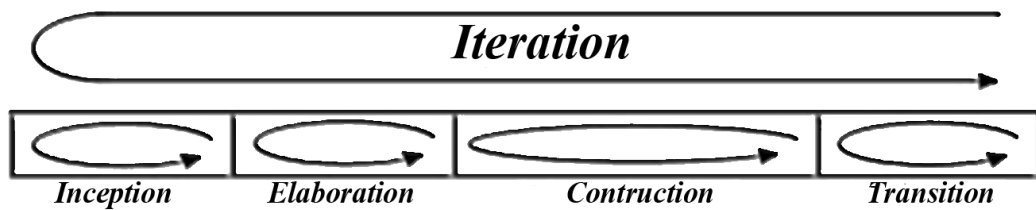
Perancangan atau rancang merupakan serangkaian prosedur untuk menerjemahkan hasil analisa dari sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem diimplementasikan (Buchari, Sentinuwo, & Lantang, 2015).

2.4. Aplikasi Web

Aplikasi merupakan penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output (Irawan & Setiyorini, 2017). Sedangkan aplikasi *web* merupakan sekumpulan file hypertext yang saling terhubung untuk menunjukan informasi-informasi tertentu dengan menggunakan teks-teks. Aplikasi web saat ini sedang berevolusi menjadi lingkungan komputasi yang canggih tidak hanya menyajikan fitur-fitur mandiri, fungsi-fungsi komputasi, dan isi untuk pengguna akhir, namun juga terintegrasi dengan sistem basis data yang dimiliki oleh perusahaan dan juga terintegrasi dengan aplikasi-aplikasi bisnis lainnya (Pressman R. , 2012).

2.5. Rational Unified Process (RUP)

Rational Unified Process (RUP) adalah pendekatan pengembangan perangkat lunak yang dilakukan secara berulang-ulang, fokus pada arsitektur, lebih diarahkan berdasarkan penggunaan kasus. *RUP* juga merupakan sebuah produk proses perangkat lunak yang dikembangkan oleh *Rational Software* yang diakuisisi oleh IBM pada bulan Februari 2003 (Sukamto & Shalahuddin, 2018). *RUP* memiliki empat buah tahapan yang dapat dilakukan secara iteratif.



Gambar 2.1. Tahapan Rational Unified Process

(Sukamto & Shalahuddin, 2018)

Berikut merupakan penjelasan untuk setiap tahapan pada *RUP* (Sukamto & Shalahuddin, 2018) :

1. Tahap pertama adalah *Inception* (permulaan) lebih pada pemodelan proses bisnis yang dibutuhkan (*business modeling*) dan mendefinisikan kebutuhan akan sistem yang akan dibuat (*requirements*). Berikut adalah tahapan yang dibutuhkan pada tahap ini:

- a. Memahami ruang lingkup dari proyek (termasuk pada biaya, waktu, kebutuhan, resiko dan lain sebagainya)
- b. Membangun kasus bisnis yang dibutuhkan.

Ditahap ini memetakan kebutuhan sistem, dengan menggunakan *Work Breakdown Structure (WBS)*.

2. Tahap kedua adalah *Elaboration* (perluasan/perencanaan), tahap ini lebih difokuskan kepada perencanaan arsitektur sistem. Tahap ini juga dapat mendeteksi apakah arsitektur sistem yang diinginkan bisa dibuat atau tidak. Mendeteksi resiko yang mungkin terjadi dari arsitektur yang dibuat. Tahap ini lebih pada analisis dan desain sistem serta implementasi sistem yang

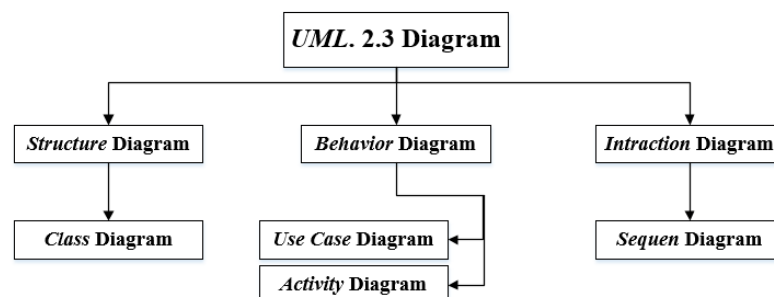
fokus pada purwarupa sistem (*prototype*). Pada tahap ini menggunakan pemodelan dan *UML*.

3. Tahap ketiga adalah *Construction* (konstruksi), tahap ini fokus pada pengembangan komponen dan fitur-fitur sistem. Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada implementasi perangkat lunak pada kode program. Tahap ini menghasilkan produk perangkat lunak. Pada tahap ini, mengimplementasikan kode program dengan XAMPP yang didukung dengan perangkat lunak lainnya.
4. Tahap keempat adalah *Transition* (transisi), tahap ini lebih pada instalasi sistem agar dapat dimengerti oleh *user*. Tahap ini menghasilkan produk perangkat lunak. Aktivitas pada tahap ini termasuk pada pelatihan *user*, pemeliharaan dan pengujian sistem apakah sudah memenuhi harapan *user*. Tahapan uji coba ini menggunakan pengujian *black-box*, yang bertujuan untuk menguji sistem dengan analisis atau tahapan yang ada pada metodologi *RUP*.

Akhir dari keempat fase ini adalah produk perangkat lunak yang sudah lengkap. Keempat fase pada *RUP* dijalankan secara berurutan dan iteratif dimana sistem interaksi dapat digunakan untuk memperbaiki interaksi berikutnya.

2.6. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah salah satu standar bahasa yang paling banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.



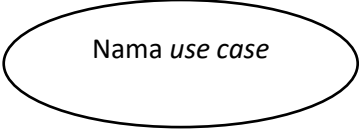
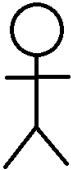

Gambar 2.2. Diagram *Unified Modeling Language*



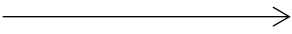

(Sukanto & Shalahuddin, 2018)

Adapun *UML* yang akan digunakan untuk penelitian ini meliputi diantaranya (Sukanto & Shalahuddin, 2018) :

1. *Use Case* atau diagram *use case*, merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi yang ada. Berikut adalah simbol-simbol yang ada didalam diagram *use case* :

Tabel 2.2. Simbol-Simbol Diagram *Use Case*


Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>.</p>
<p>Aktor/<i>actor</i></p> 	<p>Orang, proses, atau sistem lain yang berintraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.</p>
<p>Asosiasi / <i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> yang memiliki interaksi dengan aktor.</p>

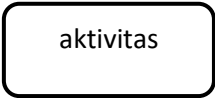
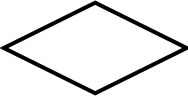


Simbol	Deskripsi
Ekstensi/ <i>extend</i> << <i>extend</i> >> 	Relasi <i>use case</i> tambahan menuju sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
Generalisasi/ <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>use case</i> dimana salah satu fungsi adalah fungsi yang lebih umum dari lainnya.
Menggunakan / <i>include</i> / <i>uses</i> << <i>include</i> >>  << <i>uses</i> >> 	Relasi tambahan ke sebuah <i>use case</i> , dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.

Sumber : Sukamto & Shalahuddin (2018)

2. *Activity Diagram*, menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu di perhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan oleh aktor, jadi aktivitas apa yang dilakukan oleh sistem.

Tabel 2.3. Notasi *Activity Diagram*


Simbol	Deskripsi
status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.



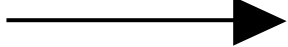
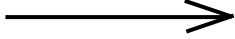

Simbol	Deskripsi
aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
percabangan / <i>decision</i> 	Asosiasi percabangan, dimana jika ada pilihan lebih dari satu aktivitas.
penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas, digabungkan menjadi satu aktivitas.
status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir dari aktivitas.


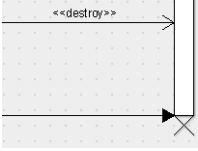
Sumber : Sukamto & Shalahuddin (2018)

3. *Sequence* Diagram, menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan kemudian diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek apa saja yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang dibuat menjadi objek itu.

Tabel 2.4. Notasi *Sequence* Diagram

Simbol	Deskripsi
aktor  Nama_aktor atau	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari <i>actor</i> adalah gambar orang, tapi <i>actor</i> belum tentu merupakan orang.

Simbol	Deskripsi
<div style="border: 1px solid black; padding: 2px; display: inline-block;">Nama_aktor</div> tanpa waktu aktif	
garis hidup / <i>lifeline</i> <div style="text-align: center;">  </div>	Menyatakan kehidupan suatu objek.
objek <div style="border: 1px solid black; padding: 2px; display: inline-block;">Nama_objek : nama kelas</div>	Menyatakan objek yang berinteraksi pesan.
Waktu aktif <div style="text-align: center;">  </div>	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.
Pesan tipe <i>create</i> <div style="text-align: center;"> <<create>>  </div>	Menyatakan suatu objek akan membuat objek yang lain, arah panah mengarah pada objek yang akan dibuat.
Pesan tipe <i>call</i> 1 : nama_metode() <div style="text-align: center;">  </div>	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri.
Pesan tipe <i>send</i> 1 : masukan <div style="text-align: center;">  </div>	Menyatakan bahwa suatu objek mengirimkan data/masukan informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.



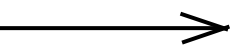
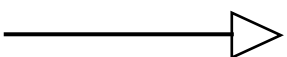
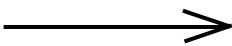

Simbol	Deskripsi
Pesan tipe <i>return</i> 1 : keluaran 	Menyatakan suatu objek yang telah menjalankan suatu operasi atau metode yang menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang akan menerima kembalian itu.
Pesan tipe <i>destroy</i> <<destroy>> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang akan diakhiri, sebaliknya jika ada <i>create</i> maka akan ada <i>destroy</i> .

Sumber : Sukamto & Shalahuddin (2018)

4. *Class Diagram*, menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sebuah sistem. Diagram kelas dibuat agar pembuat program membuat kelas-kelas sesuai dengan rancangan di dalam diagram kelas, agar antara dokumentasi perancangan dan perangkat lunak dapat sinkron. Kelas-kelas yang ada pada struktur sistem harus bisa melakukan fungsi-fungsi sesuai dengan kebutuhan sistem, sehingga pembuat perangkat lunak atau *programmer* dapat membuat kelas-kelas di dalam program perangkat lunak sesuai dengan perancangan diagram kelas yang telah dibuat.

Tabel 2.5. Simbol-Simbol Pada *Class Diagram*

Simbol	Deskripsi			
<table><tr><td>Nama_Kelas</td></tr><tr><td>+Atribut</td></tr><tr><td>+operasi()</td></tr></table> kelas	Nama_Kelas	+Atribut	+operasi()	Kelas pada struktur sistem.
Nama_Kelas				
+Atribut				
+operasi()				

<p>Antarmuka / <i>interface</i></p>  <p>Nama_<i>interface</i></p>	<p>Sama dengan konsep dalam <i>interface</i> dalam pemrograman berorientasi objek.</p>
<p>Asosiasi / <i>association</i></p> 	<p>Relasi antar kelas dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i>.</p>
<p>Asosiasi berarah <i>directed association/association</i></p> 	<p>Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>.</p>
<p>generalisasi</p> 	<p>Relasi antar kelas dengan makna / arti generalisasi-spesialisasi (umum - khusus).</p>
<p>Kebergantungan <i>/dependency association</i></p> 	<p>Relasi antar kelas dengan makna kebergantungan antar sebuah kelas.</p>
<p>agregasi / <i>aggregation</i></p> 	<p>Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>).</p>

Sumber : Sukamto & Shalahuddin (2018)

2.8. Software Pendukung

Adapun *tools* / alat bantu dalam pengimplementasian aplikasi *web* katalog berbasis *GIS* yaitu sebagai berikut :

1. *ArgoUML* adalah alat pemodelan *UML open source* terkemuka dan mencakup dukungan untuk semua diagram *UML* 1.4 standar. Ini berjalan pada *platform Java* apa saja dan tersedia dalam sepuluh bahasa (Tigris, 2019).
2. *Sublime Text*, merupakan sebuah perangkat lunak *text editor* yang digunakan untuk membuat atau meng-*edit* suatu aplikasi. *Sublime Text* mempunyai fitur *plugin* tambahan yang memudahkan para *programmer*. Selain itu, *Sublime Text* juga memiliki desain yang *simple* dan keren menjadikan *Sublime Text* terkesan elegan untuk sebuah *syntax editor* (Putratama, 2016);
3. Bahasa pemrograman, yang digunakan adalah sebagai berikut.
 - a. *HTML (Hyper Text Markup Language)*, yaitu bahasa pemformatan teks untuk dokumen-dokumen pada jaringan komputer yang sering disebut sebagai *world wide web*, yang kemudian dapat diakses untuk menampilkan berbagai informasi di dalam sebuah penjelajahan *web* internet / *browser* (Rerung, 2018);
 - b. *CSS (Cascading Style Sheet)*, adalah bahasa yang dapat digunakan untuk mendefinisikan bagaimana suatu bahasa *markup* ditampilkan pada suatu media dimana bahasa *markup* ini salah satunya adalah *HTML* (Rerung, 2018);
 - c. *PHP*, merupakan singkatan dari *Hypertext Preprocessor* adalah bahasa pemrograman *script server-side* yang didesain untuk pengembangan *web*. *PHP* disebut sebagai bahasa pemrograman server side karena *PHP* diproses pada komputer *server*. Hal ini berbeda dibandingkan dengan bahasa pemrograman *client-side* seperti *JavaScript* yang diproses pada *web browser (client)*. *PHP* dapat digunakan dengan gratis (*free*) dan bersifat *open source* (Lalompoh, 2018);
 - d. *Javascript*, adalah bahasa *scripting* yang populer diinternet dan dapat bekerja di sebagian besar *browser* populer seperti *Internet Explorer*

(IE), *Mozilla FireFox*, *Netscape* dan *Opera*. Kode *Javascript* dapat disisipkan dalam halaman *web* menggunakan *script* (Sunyoto, 2017).

4. *Framework*, adalah sebuah kerangka kerja. *Framework* juga dapat diartikan juga sebagai kumpulan dari *script* (terutama *class* dan *function*) yang dapat membantu *developer / programmer* dalam menangani berbagai masalah-masalah dalam pemrograman seperti koneksi ke *database*, pemanggilan *variabel*, *file*, dll. sehingga *developer* lebih cepat membangun aplikasi (Warsito, Yusup, & Yulianto, 2014). Pada pembuatan aplikasi, menggunakan *framework Laravel* dan *Bootstrap*.
 - a. *Laravel*, adalah sebuah *framework PHP* yang kuat dengan ukuran yang sangat kecil, dibuat untuk pengembang yang membutuhkan alat yang simpel dan elegan untuk membuat aplikasi web dengan fitur lengkap (CodeIgniter, 2020);
 - b. *Bootstrap*, adalah *framework front-end* yang *powerfull* untuk pengembangan sebuah aplikasi *web* agar lebih cepat dan mudah. *bootstrap* menggunakan *HTML*, *CSS*, dan *Javascript* (Enterprise, Pemrograman Bootstrap Untuk Pemula, 2016).
5. *Command Promp (CMD)*, untuk memakai fungsi dari *framework Laravel*, maka dibutuhkan *CMD*. Pengertian *Command Prompt* atau yang sering kita sebut dengan istilah *CMD* adalah suatu perintah *DOS* berbasis teks pada sistem operasi *Windows* (mastekno, 2017);
6. *XAMPP*, adalah perangkat lunak bantu yang dapat digunakan untuk melakukan pemrograman berbasis web tanpa harus melakukan instalasi paket perangkat lunak yang terpisah-pisah. *XAMPP* berfungsi sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL database* dan penerjemah bahasa yang ditulis kedalam bahasa pemrograman *PHP* dan *Perl* (Lalompoh, 2018).
 - a. *Apache*, sebuah nama *web server* yang bertanggung jawab pada *request-response HTTP* dan *logging* informasi secara detail. Selain itu, *Apache* juga disebut sebagai suatu *web server* yang mengikuti standar protokol *HTTP*, dan tentu saja sangat digemari (Irza, Zulhendra, & Efrizon, 2017);

- b. *MySQL*, adalah sebuah *server database open source* yang terkenal yang digunakan berbagai aplikasi terutama untuk *server* atau membuat *web*. *MySQL* berfungsi sebagai *SQL(Structured Query Language)* yang dimiliki sendiri dan sudah diperluas, umumnya bersamaan dengan *PHP* untuk membuat aplikasi *server* yang dinamis dan lebih *powerfull* (Lalompoh, 2018);
 - c. *PHPMyadmin*, adalah sebuah aplikasi *open source* / gratis yang berfungsi untuk memudahkan manajemen *MySQL*. *PhpMyAdmin* dapat membuat *database*, membuat tabel, memasukan, menghapus dan mengubah data dan terasa lebih mudah, tanpa perlu mengetikkan perintah *SQL* secara manual (Hidayat, Marlina, & Utami, 2017).
7. *GoogleChrome*, *google* meluncurkan web browser dengan meluncurkan *Chrome*, sebuah *web browser* yang memiliki kecepatan, kemudahan penggunaan yang baik. (Faiz, Umar, & Yudhana, 2017);