



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teori Umum

##### 2.1.1 Pengertian Komputer

Menurut Lubis dkk (2020) mengatakan bahwa komputer merupakan perangkat elektronika yang mempunyai sistem kerjanya harus dioperasikan berdasarkan dari perangkat lunak (*software*) dan perangkat keras (*hardware*) yang ada dan saling berkaitan.

Munazilin (2017:19) berpendapat bahwa komputer adalah mesin penghitung elektronik yang dapat menerima informasi digital, kemudian dapat melakukan pemrosesan yang sesuai dengan perintah yang tersimpan di memorinya, lalu dapat menghasilkan hasil yang sesuai dengan perintah berupa informasi.

Berdasarkan pendapat tersebut pengertian dari komputer adalah sebuah perangkat/mesin yang dapat menjalankan sebuah perintah berdasarkan informasi digital dan dapat menghasilkan sebuah hasil berupa data informasi.

##### 2.1.2 Pengertian Perangkat Lunak

Menurut Rosa dan Shalahuddin (2018:2) mengatakan bahwa Perangkat lunak adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).

Menurut Alda (2021:1) mengatakan bahwa perangkat lunak (*software*) adalah perintah (program komputer) yang bila dieksekusi memberikan fungsi dan unjuk kerja seperti yang diinginkan dimana memiliki struktur data yang memungkinkan program memanipulasi informasi secara proporsional dan mempunyai dokumen yang menggambarkan operasi dan kegunaan program.

Jadi, dapat disimpulkan bahwa perangkat lunak (*software*) merupakan sekumpulan serangkaian perintah yang dapat dimengerti oleh mesin komputer sehingga komputer dapat menjalankan apa yang kita inginkan.



### 2.1.3 Pengertian Sistem

Menurut Kristanto (2018:1), “Sistem adalah kumpulan dari elemen-elemen yang saling terkait dan saling bekerja sama untuk memroses suatu masukan (*input*) yang ditujukan untuk kepada suatu sistem tersebut, selanjutnya mengolah masukan tersebut hingga dapat menghasilkan suatu keluaran (*output*) sesuai yang diinginkan”.

Mawangga (2019:4) mengatakan bahwa suatu sistem adalah kumpulan dari elemen yang saling berintegrasi dan saling kebergantungan dalam sebuah lingkungan tersebut yang mempunyai tujuan tertentu untuk dicapai secara bersama-sama.

Berdasarkan pendapat tersebut dapat disimpulkan bahwa sistem merupakan kumpulan dari beberapa elemen seperti masukan (*input*) lalu diolah hingga mendapatkan suatu keluaran (*output*) yang saling terintegrasi untuk mencapai suatu tujuan tertentu.

### 2.1.4 Pengertian Informasi

Anggraeni dan Irviani (2017:12) mengatakan bahwa definisi informasi adalah suatu hasil dari pengolahan data yang diolah ke dalam bentuk yang lebih berguna yang menggambarkan suatu kejadian-kejadian yang digunakan untuk suatu pengambilan keputusan.

Menurut Suprihadi (2020:9) berpendapat bahwa informasi adalah sesuatu fakta yang dapat dimengerti oleh pemilik atau penggunanya.

Berdasarkan pendapat diatas dapat disimpulkan bahwa informasi adalah sekumpulan fakta yang telah diolah datanya sebelumnya oleh pemiliknya dalam mengambil suatu keputusan.

### 2.1.5 Pengertian Sistem Informasi

Sistem informasi adalah kumpulan-kumpulan dari perangkat keras, perangkat lunak serta perangkat manusia yang akan mengolah data di dalam komputer yang menggunakan perangkat-perangkat tersebut (Kristanto, 2018:12).

---



Menurut Rusmawan (dalam Susanto, 2019:34), “Sistem informasi adalah kumpulan dari sub-sub sistem baik dari non fisik maupun fisik yang saling berhubungan satu sama lain secara harmonis dalam hal bekerja sama untuk mencapai satu tujuan dengan mengolah data menjadi informasi yang berguna bagi pengguna”.

Jadi, dapat disimpulkan bahwa pengertian dari sistem informasi adalah kumpulan perangkat komputer yang akan mengolah data menjadi sebuah informasi yang berguna untuk pengguna (*user*).

### 2.1.6 Pengertian Basis Data

Lubis (2016:2) mengatakan bahwa pengertian basis data adalah gabungan dari file-file yang terbentuk karena hubungan yang masuk akal dan dapat diungkapkan dengan catatan serta bersifat tidak terikat.

Menurut Jayanti dan Sumiari (2018) berpendapat bahwa pengertian dari basis data adalah sekumpulan data yang saling terintegrasi atau terikat dari data satu dengan yang lainnya yang dikoordinirkan untuk memenuhi kebutuhan di dalam suatu organisasi tersebut.

Berdasarkan pendapat diatas, dapat disimpulkan bahwa basis data merupakan kumpulan data atau file yang telah dibentuk untuk memenuhi kebutuhan suatu organisasi tersebut yang mempunyai keterkaitan antara data satu dengan yang lainnya.

## 2.2. Teori Khusus

### 2.2.1 Pengertian *Unified Modeling Language* (UML)



Gambar 2.1 Logo *Unified Modeling Language* (UML)



Menurut Mulyani (2016:42), “ *Unified Modeling Language (UML)* merupakan sebuah teknik pengembangan sistem yang bahasanya menggunakan grafis yang digunakan sebagai alat untuk melakukan pendokumentasian serta dapat melakukan perincian pada sistem”.

Saputra dan Aprilian (2020:76) mengatakan bahwa defisini dari *Unified Modeling Language (UML)* adalah suatu bahasa pemodelan standar yang terdiri beberapa diagram yang saling terkait, kemudian dikembangkan untuk membantu pengembang sistem dan perangkat lunak untuk memvisualisasikan, menentukan, membangun, dan mendokumentasikan suatu sistem perangkat lunak.

Jadi, dapat disimpulkan bahwa pengertian dari *Unified Modeling Language (UML)* adalah suatu alat yang membantu dalam mengembangkan suatu sistem perangkat lunak.

Dengan menggunakan UML kita dapat membuat rancangan untuk semua jenis aplikasi perangkat lunak sesuai dengan rencana yang telah dibuat, dimana dapat ditulis di bahasa pemrogramana apapun, dan sistem operasi serta jaringan apapun. UML sintak mendefinisikan bagaimana bentuk yang telah dirancang dapat digabungkan. Terdapat beberapa diagram pada UML, namun berikut ini beberapa diagram yang sering digunakan oleh para pengembang, yaitu:

### **2.2.1.1 Pengertian Use Case Diagram**

Mulyani (2016:245) mengatakan bahwa *Use Case Diagram* merupakan kumpulan diagram dan teks yang saling bekerja sama untuk mengatur atau menyimpan dokumen untuk memperlihatkan bagaimana pengguna berinteraksi dengan sistem.

Menurut Habibi dan Aprilian (2019:89), “*Use Case* adalah model diagram UML yang digunakan untuk menggambarkan siapa yang akan menggunakan sistem”.

Menurut Muslihudin dan Oktafianto (2016) mengatakan bahwa terdapat komponen-komponen pembentuk diagram *use case* adalah :

Tabel 2.1 Simbol *Use Case*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menjelaskan interaksi atau peran yang pengguna kembangkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Keterikatan antara perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) yang akan mempengaruhi elemen yang bergantung pada elemen yang tidak mandiri ( <i>independent</i> ).
3		<i>Generalization</i>	Keterikatan antara objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
4		<i>Include</i>	Menjelaskan bahwa <i>use case</i> sumber secara terang-terangan ( <i>eksplisit</i> ).
5		<i>Extend</i>	Menjelaskan bahwa target <i>use case</i> memperluas perilaku dari <i>use case</i> itu sendiri yang merupakan sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menjelaskan dari paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Menjelaskan urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

Lanjutan Tabel 2.1 Simbol *Use Case*

9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah elemen-elemennya.
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Sumber : (Mulyani, 2016)

Jadi dapat disimpulkan bahwa pengertian dari *Use Case Diagram* adalah kumpulan dari teks dan diagram yang menggambarkan secara singkat siapa saja yang menggunakan sistem.

### 2.2.1.2 Pengertian Class Diagram

Haqi dan Setiawan (2019:78) berpendapat bahwa *Class Diagram* adalah salah satu diagram UML yang menampilkan kelas-kelas yang terdapat pada suatu sistem dimana sistem tersebut akan dijalankan.

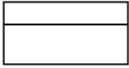




Menurut Mulyani (2016:247), “*Class Diagram* adalah diagram yang digunakan untuk menjelaskan kelas, komponen serta hubungan antar kelas tersebut”.

Tabel 2.2 Simbol *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
2		<i>Nary Association</i>	Upaya yang digunakan untuk menghindari asosiasi yang lebih dari 2 objek.

Lanjutan Tabel 2.2 Simbol *Class Diagram*



3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Sumber : (Mulyani, 2016)






Jadi, dapat disimpulkan bahwa pengertian dari *Class Diagram* adalah sebuah diagram yang digunakan untuk menampilkan kelas-kelas pada aplikasi yang akan dibuat.

### 2.2.1.3 Pengertian Activity Diagram

Mulyani (2016:249) berpendapat bahwa *Activity Diagram* merupakan diagram UML yang digunakan untuk menggambarkan alur dari suatu aplikasi.

Menurut Habibi dan Suryansah (2020:92) mengatakan bahwa pengertian dari *Activity Diagram* adalah suatu teknik yang menggambarkan sebuah logika dari aplikasi yang akan dibuat berupa prosedural atau *workflow* pada bisnis dengan kasus yang berbeda.

Tabel 2.3 Simbol *Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Sumber : (Mulyani, 2016)


Jadi, dapat disimpulkan bahwa pengertian dari *Activity Diagram* adalah gambaran aliran aktifitas yang akan dilakukan aplikasi pada satu proses.

#### 2.2.1.4 Pengertian Sequence Diagram

Menurut Haqi dan Setiawan (2019:77) mengatakan bahwa pengertian dari *Sequence Diagram* adalah salah satu dari jenis diagram UML yang menjelaskan interaksi atau aplikasi berdasarkan urutan waktu.

Mulyani (2016) berpendapat bahwa *Sequence Diagram* merupakan diagram UML yang menggambarkan hubungan antar objek.

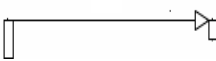
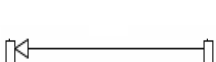
Tabel 2.4 Simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.





**Lanjutan Tabel 2.4 Simbol *Sequence Diagram***

2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

*Sumber : (Mulyani, 2016)*

Jadi, berdasarkan pendapat diatas bahwa pengertian dari *Sequence Diagram* adalah diagram UML yang menggambarkan hubungan antar objek berdasarkan urutan waktu.

### 2.2.1.5 Pengertian Kamus Data

Menurut Maniah dan Hamidin (2017) mengatakan bahwa pengertian dari kamus data yaitu sebagai katalog data dan kebutuhan-kebutuhan informasi dari suatu pemodelan sistem informasi.

Maniah dan Hamidin (2017) berpendapat bahwa kamus data mempunyai fungsi sebagai katalog data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi sehingga dapat membantu pengembang sistem agar dapat memahami aplikasi secara rinci.

Jadi, dapat disimpulkan bahwa kamus data merupakan sebagai pelaku sistem yang dapat membantu dalam memahami sebuah aplikasi secara detail dan teorganisir.

**Tabel 3.5 Kamus Data**

NO	SIMBOL	ARTI
1	=	Terdiri atas
2	+	And (dan)
3	()	Opsional
4	[]	Memilih salah satu alternatif
5	**	Komentar, Keterangan atau Catatan
6	@	Kunci (key field)
7		Pemisahaan alternatif simbol

## 2.3 Teori Judul

### 2.3.1 Pengertian Aplikasi

Menurut Alda (2020), “Aplikasi merupakan sebagai satu unit piranti lunak yang sengaja dibuat untuk memenuhi suatu kebutuhan akan berbagai aktivitas ataupun suatu pekerjaan seperti aktivitas periklanan, perniagaan, permainan, pelayanan masyarakat serta berbagai aktivitas lainnya”.

Menurut Jogiyanto HM (dalam suhartini (2017)) mengatakan bahwa aplikasi merupakan suatu penerapan, menyimpan suatu data, permasalahan, pekerjaan ke dalam suatu sarana atau media yang dapat digunakan menjadi suatu proses lainnya.

Jadi, dapat disimpulkan bahwa aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya dan siap pakai bagi pengguna.

### 2.3.2 Pengertian Persediaan

Pada umumnya, persediaan (inventory) merupakan barang dagangan yang utama dalam perusahaan dagang. Persediaan termasuk dalam golongan aset lancar perusahaan yang berperan penting dalam menghasilkan laba perusahaan. Secara umum istilah persediaan dipakai untuk menunjukkan barang-barang yang dimiliki untuk dijual kembali atau digunakan untuk memproduksi barang-barang yang



akan dijual. Dalam perusahaan dagang, persediaan merupakan barang-barang yang diperoleh atau dibeli dengan tujuan untuk dijual kembali tanpa mengubah barang itu sendiri. Sartono (2021) mengatakan bahwa Persediaan umumnya merupakan salah satu jenis aktiva lancar yang jumlahnya cukup besar dalam suatu perusahaan.

### 2.3.3 *Economic Order Quantity (EOQ)*

Pembelian bahan baku harus diperhitungkan dengan cermat. Dalam usaha perlu memperhatikan aliran masuk dan keluar dana yang digunakan untuk meningkatkan efektivitas dan efisiensi arus keuangan. Pemesanan bahan baku perusahaan harus melalui proses yang panjang, sesedikit mungkin dan seekonomis mungkin. Economic Order Quantity adalah salah satu metode yang digunakan dalam penentuan jumlah kuantitas pemesanan yang optimal (Syamsuddin, 2021). Heizer dan Render (2015), Economic Order Quantity (EOQ) adalah salah satu teknik kontrol pengendalian persediaan yang paling tua dan terkenal secara luas, metode pengendalian persediaan ini menjawab 2 (dua) pertanyaan penting, kapan harus memesan dan berapa banyak harus memesan :

- a. Biaya-biaya yang berubah-ubah sesuai dengan frekuensi pesanan, yang kini sering dinamakan set-up costs
- b. Biaya-biaya yang berubah-ubah sesuai dengan besarnya average inventory yang sering disebut carrying costs.

Perhitungan economic order quantity (EOQ) dapat dilakukan dengan rumus sebagai berikut:

$$EOQ = \sqrt{\frac{2 \cdot D \cdot S}{H}}$$

Keterangan :

D = kuantitas penggunaan per periode

S = biaya pemesanan per pesanan

H = biaya penyimpanan per unit



## 2.4 Teori Program

### 2.4.1 Pengertian HTML

Menurut Hidayatullah dan Kawistara (2015), “*Hypertext Markup Language* (HTML) adalah bahasa *standard* yang digunakan untuk menampilkan halaman web”.

Setiawan (2018:16) mengatakan bahwa pengertian dari HTML (*Hypertext Markup Language*) adalah sebuah bahasa pemrograman yang terstruktur yang dapat dikembangkan untuk membuat suatu halaman *website* yang dapat ditampilkan menggunakan *Web Browser*.

Jadi, dapat disimpulkan bahwa pengertian dari *HyperText Markup Language* (HTML) *HyperText Markup Language* (HTML) adalah Bahasa yang digunakan untuk menggambarkan struktur halaman Web.

### 2.4.2 Pengertian PHP



**Gambar 2.2** Logo PHP

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman *script server-side* yang di desain untuk pengembangan web karena diproses pada komputer server (Jannah dan Sarwandi : 2019).

Mundzir (2020) mengatakan bahwa pengertian dari PHP adalah suatu bahasa pemrograman umum yang digunakan untuk penanganan pembuatan dan pengembangan sebuah situs web yang bisa digunakan bersamaan dengan HTML.

Berdasarkan pengertian di atas, dapat ditarik kesimpulan bahwa PHP (PHP: *Hypertext Preprocessor*) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang



dapat dimengerti oleh komputer yang bersifat *server side* yang dapat ditambahkan ke dalam HTML.

### 2.4.3 Pengertian XAMPP



**Gambar 2.3** Logo XAMPP

Sidik (2020) mengatakan bahwa XAMPP merupakan paket *software* yang terdiri dari server web (Apache), database (MySQL – MariaDB), dan pengembangan aplikasi (PHP dan Perl).

Menurut Harani dan Sunandhar (2020), “XAMPP adalah sebuah perangkat lunak yang mendukung banyak dari sistem operasi dengan gabungan dari beberapa program”.

Menurut Harani dan Sunandhar (2020) bahwa XAMPP mempunyai kelebihan dan kekurangan, yaitu:

A. Kelebihan XAMPP :

1. *Database Storage Engine* ini banyak digunakan oleh programmer karena bersifat *free* (gratis).
2. Kemampuannya mempunyai kapasitas yang cukup mumpuni
3. Keamanan penyimpanan data pada XAMPP sudah terbilang cukup aman.
4. Kecepatannya menjadi kelebihan utama dari XAMPP.

B. Kekurangan XAMPP :

1. Tidak cocok untuk menangani data dalam jumlah yang besar, baik untuk menyimpan data maupun untuk memproses data.
2. Memiliki keterbatasan dalam kemampuan kinerja pada server ketika data yang disimpan telah melebihi batas maksimal kemampuan.



#### 2.4.4 Pengertian MySQL



Gambar 2.4 Logo MySQL

Menurut Fitri (2020), “MySQL adalah DBMS yang *Open Source* dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas).

Sehingga dapat disimpulkan bahwa MySQL merupakan suatu server database yang membantu pengolahan basis data dengan sangat cepat menggunakan *SQL Language*.

#### 2.4.5 Pengertian phpMyAdmin



Gambar 2.5 Logo phpMyAdmin

Yadhanto dan Prasetyo (2018) berpendapat bahwa *phpMyAdmin* adalah aplikasi web untuk mengelola database MySQL dan database MariaDB dengan mudah melalui antarmuka (interface) grafis.

Menurut Junus (2020), “*phpMyAdmin* adalah sebuah aplikasi berbasis web *Graphical User Interface* (GUI) yang digunakan bersamaan dengan DBMS MySQL.

Berdasarkan pendapat yang dikemukakan diatas dapat disimpulkan bahwa *phpMyAdmin* adalah sebuah aplikasi web yang digunakan untuk mengakses sebuah database.



## 2.5 Metode Pengembangan Sistem

### 2.5.1 Metode *eXtreme Programming*

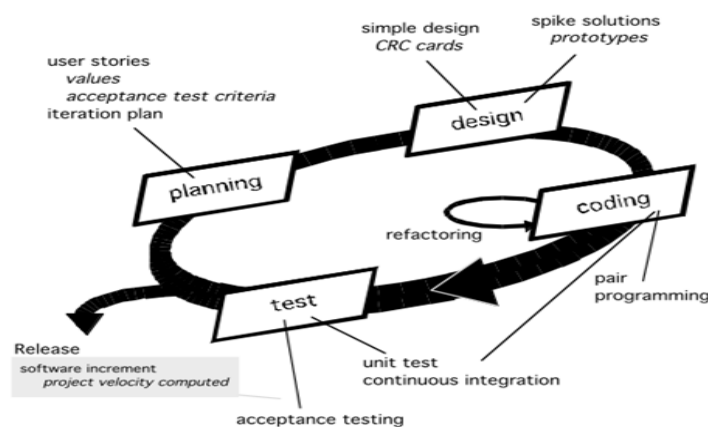
Menurut Carolina (dalam Supriyatna, 2019) mengatakan bahwa definisi dari metode *eXtreme Programming* (XP) adalah sebuah proses dari rekayasa piranti lunak yang condong menggunakan pendekatan berorientasi objek dan sasaran dari metode ini juga seimbang jika tim dihadapkan dengan persyaratan yang tidak jelas maupun terjadi perubahan persyaratan yang sangat cepat.

XP (*eXtreme Programming*) sangat cocok untuk pengembangan proyek yang memerlukan adaptasi cepat di dalam kondisi jika terjadi perubahan selama pengembangan aplikasi. XP juga cocok untuk anggota tim atau pengembang yang tidak terlalu banyak dan berada pada lokasi yang sama dalam pengembangan sistem.

#### 2.5.1.1 Tujuan Metode XP (*eXtreme Programming*)

Metodologi *eXtreme Programming* (XP) dikembangkan dengan tujuan untuk menghasilkan suatu piranti lunak yang mempunyai kuliatas tinggi yang cenderung lebih produktif. *eXtreme Programming* juga bertujuan dapat mengurangi biaya selama pengembangan piranti lunak dengan menggunakan tahapan yang lebih singkat. XP juga melihat suatu perubahan pada aplikasi secara alami maka dari itu suatu perubahan tak dapat terhindarkan selama proses pengembangan dari suatu piranti lunak.

#### 2.5.1.2 Kerangka Kerja *eXtreme Programming*



Gambar 2.7 Metodologi *eXtreme Programming*



Tahapan pengembangan perangkat lunak dengan XP meliputi :

1. **Planning/Perencanaan.** Tahap ini dimulai dengan pemahaman konteks suatu bisnis yang berasal dari aplikasi, mendefinisikan keluaran (*output*), fitur-fitur yang akan dibuat pada aplikasi, fungsi dari aplikasi yang dibuat, penentuan waktu dan biaya pengembangan aplikasi, serta alur dari pengembangan aplikasi tersebut.
2. **Design/Perancangan.** Tahap ini menekankan pada desain aplikasi secara sederhana. Alat untuk mendesain pada tahap ini dapat menggunakan kartu CRC (*Class Responsibility Collaborator*) yang dimana digunakan untuk pengembangan kelas yang akan digunakan pada diagram-diagram UML seperti *use case*, *class diagram*, *object diagram*, dan lain-lain.
3. **Coding/Pengkodean.** Hal utama dalam mengembangkan sebuah aplikasi dengan menggunakan metode XP adalah *pair programming* (dalam membuat program melibatkan 2 atau lebih pengembang).
4. **Testing/Pengujian.** Tahap ini berfokus pada pengujian dari fitur-fitur yang ada pada aplikasi sehingga tidak ada kesalahan dan aplikasi tersebut sesuai dengan proses bisnis yang telah direncanakan sebelumnya bersama *client*





*Politeknik Negeri Sriwijaya*

---