

## BAB II

### LANDASAN TEORI DAN TINJAUAN PUSTAKA

#### 2.1 Landasan Teori

##### 2.1.1 Ayat Al-Qur'an yang Berkaitan dengan Penelitian

Menikah merupakan salah satu anjuran yang dicontohkan oleh Rasulullah Muhammad SAW kepada umatnya. Ada banyak ayat di dalam kitab suci Al-Qur'an mengenai anjuran untuk menikah. Salah satu ayat yang sering dijadikan dasar untuk menikah karena setiap makhluk diciptakan berpasang – pasangan seperti yang tercantum pada Al – Qur'an. Berikut Ayat Pernikahan Dalam Islam, seperti surat An-Nisa Ayat 1 sebagai berikut :

يٰۤاَيُّهَا النَّاسُ اتَّقُوا رَبَّكُمُ الَّذِي خَلَقَكُمْ مِنْ نَفْسٍ وَاحِدَةٍ وَخَلَقَ مِنْهَا زَوْجَهَا وَبَثَّ  
مِنْهُمَا رِجَالًا كَثِيرًا وَنِسَاءً وَاتَّقُوا اللَّهَ الَّذِي تَسَاءَلُونَ بِهِ وَالْأَرْحَامَ إِنَّ اللَّهَ  
كَانَ عَلَيْكُمْ رَقِيبًا

“Hai sekalian manusia, bertakwalah kepada Tuhan-mu yang telah menciptakan kamu dari seorang diri, dan dari padanya Allah menciptakan isterinya; dan dari pada keduanya Allah memperkembang biakkan laki-laki dan perempuan yang banyak. Dan bertakwalah kepada Allah yang dengan (mempergunakan) nama-Nya kamu saling meminta satu sama lain, dan (peliharalah) hubungan silaturrahim. Sesungguhnya Allah selalu menjaga dan mengawasi kamu.” (QS. An-Nisa: 1)

### 2.1.2 Sistem Informasi

Secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung sama lain. (Fatta, 2007:3)

Berikut adalah karakteristik sistem yang dapat membedakan suatu sistem dengan sistem lainnya :

1. Batasan (*boundry*): penggambaran dari suatu elemen atau unsur mana yang termasuk di dalam sistem dan mana yang di luar sistem.
2. Lingkungan (*environment*): segala sesuatu di luar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.
3. Masukan (*input*): sumber daya (data, bahan baku, peralatan, energi) dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.
4. Keluaran (*output*): sumber daya atau produk (informasi, laporan, dokumen, tampilan layer komputer, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.
5. Komponen (*component*): kegiatan-kegiatan atau proses dalam suatu sistem yang mentransformasikan input menjadi bentuk setengah jadi (*ouput*).
6. Penghubung (*interface*): tempat dimana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.
7. Penyimpanan (*storage*): area yang dikuasai dan digunakan untuk penyimpanan dan tetap dari informasi, energi, bahan baku, dan sebagainya. (Fatta, 2007:5-6)

Informasi adalah data yang telah diolah menjadi bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau mendatang. (Fatta, 2007:9)

Sistem informasi adalah suatu sistem di dalam suatu organisasi, yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. (Putranto, 2013)

Berdasarkan beberapa referensi di atas, dapat disimpulkan bahwa sistem informasi adalah “suatu sistem yang dirancang agar dapat mempermudah transaksi di dalam perusahaan guna mencapai tujuan yang sudah ditetapkan”.

### **2.1.3 *Customer Relationship Management (CRM)***

bagi kalangan perusahaan yang bergerak di bidang teknologi informasi (IT), istilah CRM digunakan untuk mendeskripsikan berbagai aplikasi perangkat lunak yang digunakan untuk mengoptimisasi fungsi-fungsi pemasaran, penjualan, dan pelayanan. Pasar produk perangkat lunak CRM yang dewasa ini diramaikan oleh sekian banyak pemain itu sebenarnya dimulai pada tahun 1993 ketika Tom Siebel mendirikan perusahaan Siebel System Inc. Asal usul penggunaan istilah CRM dapat dilacak dari masa tersebut. (Buttle, 2004: 3).

*Customer Relationship Management (CRM)* merupakan rangkaian informasi yang menggunakan teknologi informasi untuk yang menyangkut metodologi, strategi, perangkat lunak dan keras yang mendukung proses kerja dan membantu perusahaan dalam mengelola hubungan dengan para konsumen. Tujuan utama dari CRM ini adalah agar konsumen menjadi lebih terpuaskan sehingga

konsumen menjadi seorang konsumen loyal dan konsumen yang selalu menggunakan produk kita secara terus-menerus (*Customer Retention*). CRM juga dijadikan sarana informasi terbaru tentang konsumen kita, sehingga perusahaan mampu mengimbangi kemauan dan harapan konsumen masa kini yang cerdas, berpotensi, dan pintar. CRM ini merupakan suatu strategi bagaimana membuat relasi kompleks yang saling memengaruhi antara strategi, taktik, proses, dan menata kemampuan serta teknologi informasi dalam memperoleh, mengkonsolidasi, dan menganalisis data konsumen dan kemudian digunakan kembali untuk berinteraksi dengan konsumen. (Wijayanti, 2018:197-198).

Sistem CRM ada 5 bagian penting, yaitu masukan, pengolahan, dan keluaran *plus feedback* atau komponen umpan balik, dan komponen pengendalian atau kontrol. Komponen *feedback* merupakan komponen data tentang kinerja sebuah sistem yang harus sesuai dengan tujuan CRM. Komponen kontrol digunakan untuk mengontrol proses pemantauan dan pengkajian apakah sistem CRM sudah berjalan sesuai dengan harapan atau sebaliknya. (Wijayanti, 2018:202).

Berdasarkan referensi diatas dapat disimpulkan bahwa CRM adalah strategi mengelola data konsumen untuk berinteraksi dan menjalin hubungan baik dengan konsumen sehingga perusahaan dapat mengetahui kebutuhan konsumen, dan untuk memberikan perhatian lebih kepada konsumen tersebut agar mendapatkan loyalitas konsumen.

Menurut (Buttle, 2004) Konsep *Customer Relationship Management (CRM)* dapat dipahami dalam tiga tataran, yaitu strategis, operasional, dan analitis.

*CRM* strategis terfokus pada upaya untuk mengembangkan kultur usaha yang berorientasi pada pelanggan atau *customer-centric*. Kultur ini ditujukan untuk merebut hati konsumen dan menjaga loyalitas mereka dengan menciptakan serta memberikan nilai bagi pelanggan yang mengungguli para pesaing.(Buttle, 2004: 4).

*CRM* operasional lebih terfokus pada otomatisasi cara-cara perusahaan dalam berhubungan dengan pelanggan. Berbagai aplikasi perangkat lunak *CRM* memungkinkan fungsi-fungsi pemasaran dan penjualan dapat berjalan secara otomatis.(Buttle, 2004: 6).

*CRM* analitis digunakan untuk mengeksploitasi data konsumen demi meningkatkan nilai mereka (dan nilai perusahaan). sistem ini dikembangkan berdasarkan informasi mengenai konsumen. Data pelanggan dapat diperoleh dari pusat-pusat informasi atau bank data yang dimiliki setiap perusahaan yang relevan, yakni data penjualan (riwayat pembelian barang atau jasa oleh pelanggan), data finansial (riwayat pembayaran atau skor kredit), data pemasaran (respon konsumen terhadap kampanye iklan, data skala loyalitas produk), dan data layanan.(Buttle, 2004: 13).

Alasan mendasar yang mendorong perusahaan membina hubungan dengan konsumen sesungguhnya sangat klasik, yakni motif ekonomi. Pundi-pundi perusahaan akan semakin gemuk jika mereka mampu mengelola *baseline* konsumen untuk mengidentifikasi, memuaskan, dan berhasil mempertahankan konsumen mereka yang paling menguntungkan. Itulah sesungguhnya tujuan utama yang dibidik oleh semua strategi CRM. (Buttle, 2004: 23).

#### **2.1.4 *Wedding Organizer***

*Wedding Organizer* (WO) adalah jasa pengorganisasian untuk acara pernikahan, baik dalam perencanaannya maupun pada saat hari pernikahan. *Wedding Organizer* hanyalah perantara antara klien dengan vendor dan antara klien dengan panitia. Selain itu WO juga mengatur keseluruhan acara dari konsep hingga acara penyelenggaraan pesta resepsi. (Fithrati 2014:75-76).

Berdasarkan referensi di atas dapat disimpulkan bahwa *Wedding Organizer* adalah sebuah tempat untuk vendor mempromosikan jasa yang mereka tawarkan, dan juga sarana bagi calon pengantin untuk merancang acara pernikahan dengan pertimbangan dana ataupun kebutuhan lainnya.

### **2.1.5 Metode Perancangan**

Teori yang berhubungan dengan teknik analisa yang digunakan meliputi Flowchart, DFD dan ERD.








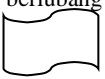

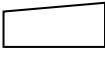







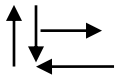



#### **2.1.5.1 Flowchart**

Definisi *Flowchart* Menurut Jogiyanto (2005:796) bagan alir sistem (*System Flowchart*) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem.

Dari berbagai pendapat para ahli dapat disimpulkan bahwa *flowchart* adalah teknik analitis bergambar yang digunakan untuk menjelaskan beberapa aspek dari sistem indaftarasi secara jelas, ringkas, dan logis serta bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem.

Berikut simbol bagan alir sistem (*systems flowchart*) dapat dilihat pada Tabel 2.1 :

Tabel 2.1 Simbol *Flowchart*

No	Simbol	Keterangan	No	Simbol	Keterangan
1	Dokumen 	Menunjukkan dokumen <i>input</i> dan <i>output</i> baik proses manual, mekanik atau <i>computer</i>	11	<i>Hard disk</i> 	Menunjukkan <i>input/ output</i> menggunakan <i>hard disk</i>
2	Kegiatan Manual 	Menunjukkan pekerjaan manual	12	Diskette 	Menunjukkan <i>input/ output</i> menggunakan diskette
3	Simpanan Offline 	File non-komputer yang diarsip urut angka ( <i>numerical</i> )	13	Drum magnetik 	Menunjukkan <i>input/ output</i> menggunakan Drum magnetik
4	Simpanan Offline 	File non-komputer yang diarsip urut angka ( <i>alphabetical</i> )	14	Pita kertas berlubang 	Menunjukkan <i>input /output</i> menggunakan Pita kertas berlubang
5	Simpanan Offline 	File non-komputer yang diarsip urut angka ( <i>cronological</i> )	15	Keyboard 	Menunjukkan <i>input/ output</i> menggunakan on-line keyboard
6	Kartu plong 	Menunjukkan <i>input/output</i> yang menggunakan kartu plong	16	Display 	Menunjukkan output yang tampil di komputer
7	Proses 	Menunjukkan proses dari operasi program komputer	17	Pita kontrol 	Menunjukkan penggunaan pita kontrol dalam <i>batch control total</i> untuk pencocokan di proses <i>batch processing</i>
8	Operasi luar 	Menunjukkan operasi yang dilakukan diluar proses operasi komputer	18	Hubungan komunikasi 	Menunjukkan proses transmisi data melalui <i>channel</i> komunikasi
9	Pengurutan <i>offline</i> 	Menunjukkan proses pengurutan data diluar proses komputer	19	Garis alir 	Menunjukkan arus proses
10	Pita magnetik 	Menunjukkan <i>input/ output</i> menggunakan pita magnetik	20	Penjelasan 	Penjelasandari suatu proses
			21	Penghubung 	Menunjukkan penghubung ke halaman yang masi sama atau ke halaman lain

(Sumber: Jogiyanto, 2005:796)

### 2.1.5.2 DFD (*Data Flow Diagram*)

Menurut Pressman (2012:225) DFD (beberapa penulis di Indonesia menyebutnya sebagai DAD atau Diagram Aliran Data) memperlihatkan gambaran tentang masukan-proses-keluaran dari suatu sistem/perangkat lunak.

Menurut Rosa A.S dan M.Shalahudin (2016:70) *Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*uotput*).

Menurut Rosa A.S dan M.Shalahudin (2016:70) berikut merupakan tahapan-tahapan perancangan dengan menggunakan DFD:


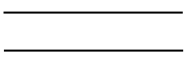
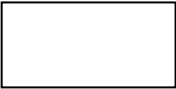
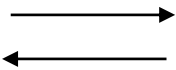
1. Membuat DFD level 0 atau sering disebut juga *Context Diagram*. DFD level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DVD level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.
2. Membuat DFD level 1. DFD level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD level 1 merupakan hasil *breakdown* DFD level 0 yang sebelumnya sudah dibuat.
3. Membuat DFD level 2. Modul-modul pada DFD level 1 dapat di-*breakdown* menjadi DFD level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk



di-*breakdown* lagi. Untuk sebuah sistem , jumlah DFD level 2 sama dengan jumlah modul pada DFD level 1 yang di-*breakdown*..

Notasi-notasi pada DFD ( Edward Yourdon dan Tom De Marco) adalah sebagai berikut:

**Tabel 2.2** *Simbol Data Flow Diagram*

Notasi	Keterangan
	Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program
	<i>File</i> atau basis data atau penyimpanan ( <i>storage</i> ); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data ( <i>Entity Relationship Diagram</i> (ERD), <i>Conceptual Data Model</i> (CMD), <i>Physical Data Model</i> (PMD))
	Entitas luar ( <i>external entity</i> ) atau masukan ( <i>input</i> ) atau keluaran ( <i>output</i> ) atau orang yang memakai/berinteraksi dengan perangkat lunak yang di modelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan
	Aliran data; merupakan data yang dikirim antar proses, dari menyimpan ke proses, atau dari proses ke masukan ( <i>input</i> ) atau keluaran ( <i>output</i> )

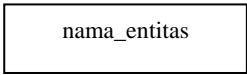
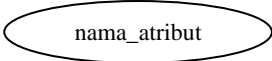
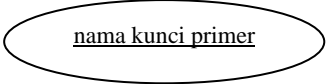
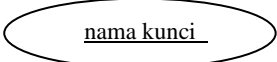
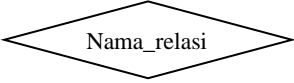
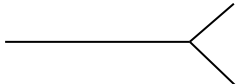
(Sumber: Rosa A.S dan M.Shalahudin, 2016:71)

### 2.1.5.3 ERD (*Entity Relationship Diagram*)

Menurut Menurut Rosa A.S dan M.Shalahudin (2016:50) Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data menggunakan OODBMS maka perancangan basis data tidak perlu menggunakan ERD. ERD memiliki beberapa aliran notasi seperti notasi chen (dikembangkan oleh Peter Chan), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow's Foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah

notasi dari Chen. Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi Chen:

**Tabel 2.3** Simbol ERD

Simbol	Deskripsi
Entitas / <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel
Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama)
Atribut multivalai / <i>multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
Asosiasi / <i>association</i> N 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian

(Sumber: Rosa dan Shalahuddin, 2016:50)

### 2.1.6 PHP (*Hypertext Preprocessor*)

Menurut dokumen resmi PHP, PHP singkatan dari PHP *Hypertext Preprocessor*, ia merupakan bahasa berbentuk skrip yang ditempatkan dalam *server* dan di proses di *server*. Hasilnyalah yang dikirim ke klien tempat pemakai menggunakan *browser*. Secara khusus, PHP dirancang untuk membentuk web dinamis. Artinya, ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah *skrip Perl*

yang dapat mengamati siapa saja yang melihat-lihat daftar riwayat hidupnya, yakni pada tahun 1994. *Skrip-skrip* ini selanjutnya dikemas menjadi *tool* yang disebut “*Personal Home Page*”. Paket inilah yang menjadi cikal-bakal PHP. (Kadir, 2002:2).

PHP/FI merupakan nama awal dari PHP. FI adalah Form Interface. Dibuat pertama kali oleh Rasmus Lerdoff. PHP awalnya merupakan program CGI yang dikhususkan untuk menerima input melalui form yang ditampilkan dalam *browser web*. *Software* ini disebar dan dilisensikan sebagai perangkat lunak *Open Source*. (Sidik, 2017:4).

#### **2.1.7 Database / Basisdata**

Basis data terdiri atas 2 kata, yaitu Basis dan Data. Basis kurang lebih diartikan sebagai markas atau gedung, tempat bersarang. Sedangkan Data adalah representasi fakta dunia nyata yang mewakili objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, peristiwa, konsep, keadaan, dan sebagainya, yang diwujudkan dalam angka, huruf, simbol, teks, gambar, bunyi, atau kombinasi. (Fathansyah, 2012:2).

Menurut Fathansyah (2012:2-3), sebagai satu kesatuan istilah, Basis Data (*Database*) sendiri didefinisikan dalam sejumlah sudut pandang seperti:

- Himpunan kelompok data (arsip) yang saling berhubungan diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan dengan cepat dan mudah
- Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundansi*) yang tidak perlu, untuk memenuhi berbagai kebutuhan

- Kumpulan *file/tabel/arsip* yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

### **2.1.8 Pengujian Sistem**

Pengujian sistem adalah serangkaian pengujian yang berbeda-beda yang tujuan utamanya adalah untuk sepenuhnya mewujudkan sistem berbasis-komputer. Meskipun masing-masing pengujian memiliki tujuan yang berbeda, semua pengujian tersebut dilakukan untuk memverifikasi bahwa semua elemen sistem telah terintegrasi dengan baik dan menjalankan fungsi yang telah ditetapkan. (Pressman, 2012:572).

Pengujian perangkat lunak merupakan persentase terbesar dari upaya teknis dalam proses perangkat lunak. Apapun jenis perangkat lunak yang anda bangun, strategi untuk perencanaan pengujian yang sistematis, pelaksanaan, dan kontrol dimulai dengan mempertimbangkan elemen-elemen kecil dalam perangkat lunak dan bergerak keluar terhadap program secara keseluruhan. Tujuan pengujian perangkat lunak adalah untuk menemukan kesalahan. (Pressman, 2012:580).

#### **2.1.8.1 Pengujian Kotak Hitam (*BlackBox Testing*)**

Peneliti menggunakan pengujian kotak hitam (*Blackbox*) sebagai pengujian sistem. Pengujian kotak hitam (*Black Box Testing*) berkaitan dengan pengujian-pengujian yang dilakukan pada antarmuka perangkat lunak. Pengujian kotak hitam mengkaji beberapa aspek fundamental dari suatu sistem/perangkat lunak dengan sedikit memperhatikan struktur logis internal dari perangkat lunak. (Pressman, 2012:587).

Pengujian *Black Box* atau Kotak Hitam berfokus pada persyaratan fungsional perangkat lunak. Artinya, teknik pengujian kotak hitam memungkinkan Anda untuk membuat beberapa kumpulan kondisi masukan yang sepenuhnya akan melakukan semua kebutuhan fungsional untuk program. Pengujian kotak hitam bukan teknik alternatif untuk kotak putih. Sebaliknya, ini merupakan pendekatan pelengkap yang mungkin dilakukan untuk mengungkap kelas kesalahan yang berbeda dan yang diungkap oleh metode kotak putih (Pressman, 2012:597).

Pengujian Kotak Hitam berupaya untuk menemukan kesalahan dalam kategori berikut :

- a. Fungsi yang salah atau hilang
- b. Kesalahan antarmuka
- c. Kesalahan dalam struktur data atau akses basis data eksternal
- d. Kesalahan perilaku atau kinerja
- e. Kesalahan inisialisasi dan penghentian

#### **2.1.8.1.1 Partisi Ekivalensi**

*Black box* mempunyai beberapa jenis pengujian diantaranya: metode pengujian *graph-based*, partisi ekivalensi, analisis nilai batas, dan pengujian perbandingan. Namun penelitian ini hanya menggunakan partisi ekivalensi sebagai jenis pengujian *black box*.

Partisi Ekivalensi adalah metode pengujian *black-box* yang membagi domain input dari suatu program ke dalam kelas data dari mana *test case* dapat dilakukan. *Test case* yang ideal mengungkap kelas kesalahan (misalnya, pemrosesan yang

tidak benar terhadap semua data karakter) yang akan memerlukan banyak kasus untuk dieksekusi sebelum kesalahan umum diamati. (Pressman, 2012: 556).

Desain *test case* untuk partisi ekivalensi didasarkan pada evaluasi terhadap kelas ekivalensi untuk suatu kondisi *input*. Bila serangkaian objek dapat di-link oleh hubungan yang simetris, transitif, dan refleksif, maka ada kelas ekivalensi. Kelas ekivalensi merepresentasikan serangkaian keadaan *valid* atau *invalid* untuk kondisi *input*. Secara khusus, suatu kondisi *input* dapat berupa harga numeris, suatu rentang harga, atau serangkaian harga terkait, atau sebuah kondisi *boolean*. (Pressman, 2012: 556).

## 2.2 Tinjauan Pustaka

Penelitian sebelumnya pernah dilakukan oleh Anatasha Onna Carissa et. al, tahun 2014 yang berjudul “PENERAPAN *CUSTOMER RELATIONSHIP MANAGEMENT (CRM)* SEBAGAI UPAYA UNTUK MENINGKATKAN LOYALITAS PELANGGAN”. Pada CRM Operasional dengan melakukan segmentasi pasar yaitu lebih pada kalangan remaja dan adanya otomatisasi layanan yaitu melayani pembelian secara online dan adanya contact center untuk menangani keluhan pelanggan. Serta pada CRM Analitis dengan menggunakan program Fashion Card untuk menyimpan data pelanggan.penerapan *Customer Relationship Management (CRM)* yang dilakukan Bandung Sport sebagai upaya untuk meningkatkan loyalitas pelanggan sudah baik dibuktikan dengan serangkaian proses diatas dan adanya tingkat pembelian kembali mencapai 25% dari total penjualan setiap bulan.

Penelitian selanjutnya pernah dilakukan oleh Indana Zulfa pada tahun 2017 yang berjudul “PENERAPAN *CUSTOMER RELATIONSHIP MANAGEMENT* DALAM MEMBERIKAN PELAYANAN PADA PT. BANK MUAMALAT INDONESIA KANTOR CABANG PEMBANTU PEUNAYONG BANDA ACEH”. Hasil penelitian menunjukkan bahwa penerapan CRM di PT. Bank Muamalat Indonesia Kantor Cabang Pembantu Peunayong Banda Aceh termasuk dalam kategori cukup baik dengan proses identifikasi, differensiasi, interaksi dan personalisasi. Hambatan dalam penerapan CRM meliputi hambatan internal dan eksternal. Hambatan internal merupakan hambatan yang timbulkan oleh pihak bank itu sendiri, sedangkan hambatan eksternal adalah hambatan yang muncul dari nasabah.

Penelitian selanjutnya pernah dilakukan oleh Ina Najiyah dan Suharyanto pada tahun 2017 yang berjudul “Sistem Informasi Wedding Planner Berbasis Web”. Pembuatan website wedding planner ini tujuannya untuk memudahkan vendor atau WO untuk mempromosikan produk di website dengan lebih efektif dan efisien serta dapat memberikan informasi lengkap untuk perencanaan pernikahan bagi calon pengantin. Dengan adanya sistem informasi wedding planner berbasis website ini memberikan gambaran kepada calon pengantin (member) untuk merencanakan pernikahannya dengan informasi-informasi yang tersedia di website wedding planner ini.

Penelitian selanjutnya dari M. Azam Ragil Tri Putranto pada tahun 2013 yang berjudul “Sistem Informasi Persewaan *Wedding Organizer* Berbasis Web”. Sistem informasi *Wedding Organizer* sangat membantu bagi orang-orang yang tidak mau repot dengan masalah perencanaan pernikahan mereka. Sistem WO ini

mempermudah dalam melakukan promosi untuk menjual paket pernikahan yang sesuai dengan jumlah undangan, catering, gedung, serta dekorasi pelaminan pengantin yang akan menentukan harga paket pernikahan.

Penelitian selanjutnya juga pernah dilakukan oleh Ade Wiwid Taniah dan Sri Harjunawati pada tahun 2017 yang berjudul “Perancangan Sistem Informasi Transaksi Penyewaan *Wedding Organizer* Pada CV. Denis Citra Mandiri Bekasi”. Aplikasi UML pada sistem ini dapat mempermudah dalam mendapatkan informasi yang dibutuhkan, diantaranya informasi paket pernikahan, pemesanan, laporan, serta informasi lain yang dibutuhkan. Penggunaan desain ini merupakan salah satu solusi untuk memecahkan permasalahan dari sistem berjalan.