

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Tinjauan Pustaka

Berikut beberapa kajian yang relevan dilakukan penulis pada beberapa penelitian terdahulu yang mengangkat topik penelitian tentang perancangan sistem informasi berbasis *web* dengan menggunakan metode *agile* dan semacamnya.

Penelitian yang dilakukan oleh R. Rahardian Dkk pada tahun 2022 menggunakan metode *agile* dalam merancang dan mendesain *website booking room* Witel Telkom Yogyakarta. *website* ini dibuat menggunakan bahasa pemrograman PHP dan kerangka kerja *Code Igniter* serta menggunakan metode *agile* karena metode ini lebih fleksibel dalam proses pengembangan. Penelitian ini dapat membantu pengguna dalam pemesanan ruangan secara *online* [7].

Penelitian selanjutnya dilakukan oleh K. Simatupang, Dkk pada tahun 2022 menggunakan metode *agile* dalam Merancang Sistem Informasi Pemesanan Fasilitas di Universitas Advent Indonesia. Sistem informasi ini diterapkan untuk mengatasi masalah kebutuhan sistem pemesanan ruangan di Universitas Advent Indonesia, Perancangan sistem informasi ini Mampu beroperasi secara optimal dan telah melewati pengujian dengan sukses menggunakan *black box testing* [8].

Penelitian selanjutnya dilakukan oleh M. Aldi, Dkk pada tahun 2021 menggunakan metode *agile*, Rancang Bangun Aplikasi Berbasis *Website* Manajemen Kost Pondok Salma. Berdasarkan hasil penelitian ini, didapatkan kesimpulan bahwa aplikasi manajemen kost pondok salma berbasis *website* ini berjalan dengan baik yang dibuktikan melalui pengujian *black box testing* dan ISO 9126, dengan indikator berhasil [9].

Dalam penelitian ini, penulis akan merancang sistem informasi kost sruntul berbasis *website* menggunakan metode *agile*, berbeda dengan penelitian sebelumnya yang berfokus pada perancangan sistem informasi *website booking room* witel telkom yogyakarta, merancang sistem informasi pemesanan fasilitas di universitas advent indonesia, dan rancang bangun aplikasi berbasis *website* manajemen kost pondok salma menggunakan metode *agile*, maka penelitian ini

akan berfokus pada pembangunan sistem informasi pada kost sruntul menggunakan metode *agile* serta dibangun menggunakan *framework laravel* dengan fitur-fitur yang akan memudahkan pengelolaan manajemen data anak kos di kost sruntul. *Website* ini akan diuji dengan menggunakan *Black Box Testing*. *Black box testing* merupakan pengujian aplikasi yang berfokus pada fungsional aplikasi. Pilihan menggunakan metode pengujian *backbox* dipertimbangkan dalam penelitian ini karena memiliki beberapa keunggulan, di antaranya adalah adalah penguji tidak harus memiliki kemampuan tentang bahasa pemrograman yang digunakan dalam perancangan aplikasi [10].

Tabel 2. 1. Tabel Kajian Pustaka dari Hasil Penelitian Terdahulu

No.	Penulis	Judul	Metode	Hasil dan Kesimpulan
1.	R.Rahardian, Dkk (2022) [7].	Desain <i>website booking room</i> Witel Telkom Yogyakarta.	<i>Agile</i>	Dari hasil penelitian yang dilakukan, <i>website</i> ini dibangun dengan bahasa pemrograman PHP dan kerangka kerja <i>Code Igniter</i> serta menggunakan metode agile karena metode ini lebih fleksibel dalam proses pengembangan. Penelitian ini dapat membantu pengguna dalam pemesanan ruangan secara <i>online</i> .
2.	K. Simatupang, Dkk (2022) [8].	Rancang Sistem Informasi Pemesanan Fasilitas di Universitas Advent Indonesia.	<i>Agile</i>	Dari hasil penelitian ini, ditemukan kesimpulan dalam perancangan sistem informasi pemesanan fasilitas Universitas Advent Indonesia menggunakan metode agile ini dapat dengan mudah diimplementasikan karena fleksibel terhadap perubahan-perubahan implementasi saat pengembangan. Sistem ini juga diuji dengan <i>black box testing</i> dimana indikator pengujian pada tabel menunjukkan 100% <i>valid</i> .
3.	M. Aldini, Dkk (2021) [9].	Rancang Bangun Aplikasi Berbasis <i>Website</i> Manajemen Kost Pondok Salma	<i>Agile</i>	Berdasarkan hasil penelitian ini, didapatkan kesimpulan bahwa aplikasi manajemen kost pondok salma berbasis <i>website</i> ini berjalan dengan baik yang dibuktikan melalui pengujian <i>black box</i>

				<i>testing</i> dan ISO 9126, dengan indikator berhasil.
4.	M.Yumna Majdina, Dkk (2020) [11].	Perancangan Aplikasi Manajemen Gudang Berbasis <i>Website</i> Menggunakan Metode Agile	<i>Agile</i>	Berdasarkan aplikasi yang diimplementasikan dengan menggunakan metode agile, serta diuji dengan <i>black box testing</i> berjalan dengan baik dengan tabel indikator keberhasilan status <i>valid</i> dan berhasil.
5.	N. Hasanah Dkk (2020) [12].	Rancang Bangun Sistem Informasi Pemasaran Berbasis <i>Web</i> untuk Rumah Kost.	<i>Waterfall</i>	Perancangan sistem informasi ini dibangun dengan menggunakan pendekatan metode <i>waterfall</i> , dimana terdapat beberapa kekurangan jika menggunakan metode ini, diantaranya adalah sulit dalam perubahan atau tidak fleksibel karena sistem dibuat linier dan hasil akhir harus sesuai dengan perancangan awal yang telah disepakati serta memakan banyak waktu perancangan sistem informasi ini juga diuji dengan <i>black box testing</i> dengan indikator valid dan sesuai dengan alur.

6.	C. Nizar Dkk (2021) [13].	Rancang Bangun <i>Website</i> Rumah Kost (E-Kost)	<i>Waterfall</i>	Berdasarkan penelitian ini, perancangan sistem informasi ini dapat memberikan solusi terhadap pencarian dan manajemen kost. Penelitian ini menggunakan metode <i>waterfall</i> yang berjalan baik sesuai dengan fungsionalitasnya, dan mendapatkan nilai pengujian dari beberapa responden menggunakan tabel pengujian kelayakan dengan rata-rata nilai PI 85%. Namun dengan menggunakan metode ini memiliki beberapa kelemahan, diantaranya adalah memakan banyak waktu, serta tidak fleksibel dalam proses pengembangan.
7.	N. Maulida Solihat Dkk (2021) [14].	<i>Website-Based Boarding House Application Design in Sumbawa Regency</i>	<i>Waterfall</i>	Berdasarkan hasil penelitian ini, aplikasi kost berbasis <i>website</i> ini berjalan sesuai harapan yang diuji menggunakan <i>black box testing</i> berdasarkan pengujian fungsionalitas. Aplikasi ini dibangun menggunakan metode <i>waterfall</i> dan <i>framework code igniter</i> . Terdapat beberapa kelemahan dalam penggunaan <i>framework code igniter</i> diantaranya adalah perancangan <i>website</i> dalam skala

				besar tidak dianjurkan dan <i>library</i> sangat terbatas.
8.	F. Anggiani Dkk (2021) [15].	Sistem Informasi Manajemen Berbasis <i>Web</i> Dengan <i>Framework</i> <i>Codeigniter</i>	<i>Waterfall</i>	Berdasarkan hasil penelitian ini, sistem informasi manajemen kost berbasis <i>web</i> ini berhasil diterapkan yang dapat membantu pemilik kost dalam manajemen data anak kost. Perancangan sistem informasi ini dirancang menggunakan metode <i>waterfall</i> , metode ini memiliki beberapa kelemahan, antara lain, tidak fleksibel dan harus terpaku pada proses perancangan data di awal kesepakatan, serta sulit dalam melakukan perubahan fitur.
9.	S. Kristianto (2022) [16].	Analisis Sistem Informasi Penyewaan Kost Di Kota Tangerang	<i>Waterfall</i>	Berdasarkan penelitian ini, sistem informasi ini dapat diterapkan untuk mempermudah pelanggan untuk mencari tempat kost. Perancangan sistem informasi ini dirancang menggunakan metode <i>waterfall</i> . Penelitian ini berfokus pada penerapan sistem informasi di kota Tangerang. Sistem informasi yang dirancang ini masih memiliki beberapa kekurangan, antara lain tampilan kurang mendukung user <i>friendly</i> , opsi pembayaran yang masih terbatas dan lain-lain.

10.	Y. Dwiputra (2021) [10].	<i>Development Of A Boarding House Search Information System Using The Waterfall Model</i>	<i>Waterfall</i>	Berdasarkan hasil penelitian yang dilakukan oleh Y. Dwiputra Dkk, penelitian ini menghasilkan <i>prototype</i> aplikasi pencarian kost menggunakan metode <i>waterfall</i> dengan menggunakan <i>framework laravel</i> .
-----	--------------------------	--	------------------	--

## 2.2 Dasar Teori

Pada bab ini akan dibahas teori-teori yang terkait dengan topik penelitian ini. Teori yang termasuk antara lain rumah kost, *website*, *laravel*, *Agile*, dan teori lainnya yang relevan.

### 2.2.1 Rumah Kost

Kost atau kontrakan merupakan jasa penyedia tempat tinggal sementara yang disediakan oleh pemilik kost, pada umumnya kost merupakan bangunan tempat tinggal yang berbentuk kamar atau ruangan untuk ditinggali. Kost berasal dari bahasa Belanda, yaitu "*In De Kost*" yang memiliki arti makan di dalam, kata ini dapat dikaji lebih dalam artinya menjadi "tinggal dan ikut makan" [1].

### 2.2.2 Website

*Website* merupakan suatu halaman yang saling berhubungan yang biasanya terletak atau disimpan pada *server* yang sama yang terdiri dari teks, audio, video, gambar, dan lain-lain. *Website* dibuat dengan tujuan yang beragam untuk menyampaikan informasi kepada khalayak ramai yang dapat diakses melalui internet [17].

*Website* dapat juga diartikan sebagai kumpulan lembaran digital yang didalamnya terkandung informasi berupa audio, teks, gambar, video yang terhubung ke internet. *Website* terdiri dari dua jenis, yaitu *web* statis dan dinamis. *Web* statis merupakan website yang disediakan untuk website lokal saja, atau isi konten *web* didalamnya tidak diperbaharui, sedangkan *web* dinamis merupakan website yang mempunyai isi diperbaharui sesering mungkin untuk menghasilkan informasi terbaru [18].

### 2.2.3 Agile

*Agile Modeling* Merupakan metode pengembangan *software* atau metode yang dipakai untuk manajemen proyek dengan menggunakan fase pengembangan yang singkat atau sering disebut "*sprint*". Dalam pengembangan perangkat lunak atau manajemen proyek, terdapat tujuh macam model, yaitu: *agile modelling*, *crystal*, *dynamic system development methodology*, *adaptive software development*, *feature driven development*, *extreme programming (XP)*, dan *scrum*.

*Agile* memiliki beberapa tahapan dalam proses perancangan dan pengembangan perangkat lunak, diantaranya adalah:

a. Perencanaan

Pada tahap ini pengembang melakukan tahapan persiapan dan perencanaan.

b. Rancangan / Implementasi

Pada tahap ini tim melakukan perancangan sistem dan melakukan pengkodean perangkat lunak.

c. Testing

Setelah membuat sistem, dilakukan pengujian dan pengecekan sistem.

d. Dokumentasi

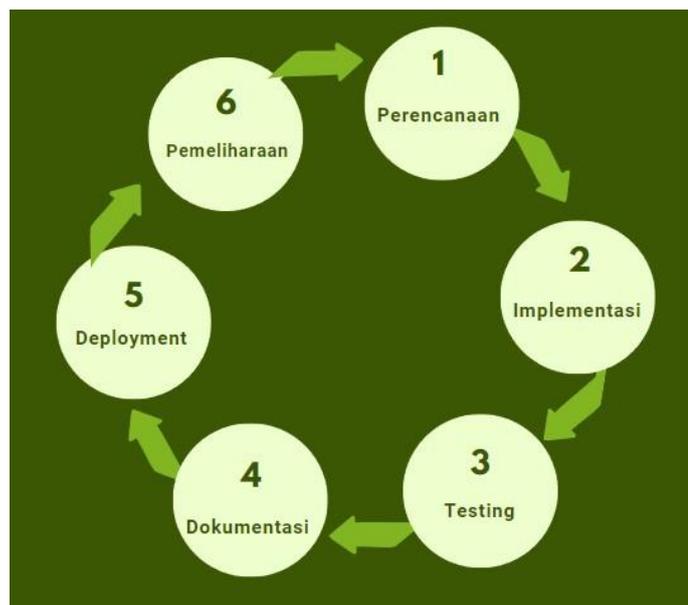
Setelah dilakukan *testing*, maka dilakukan proses dokumentasi dengan tujuan untuk mempermudah *maintenance* dikemudian hari.

e. Deployment

Tahap ini dilakukan untuk menjamin kinerja dan kualitas sistem yang telah dibuat, jika sistem sudah berjalan sesuai alur dan memenuhi persyaratan, maka aplikasi dapat disebarakan untuk dipakai.

f. Pemeliharaan

Tahap ini dilakukan untuk mengurangi atau menghilangkan kesalahan atau *bug* yang akan terjadi pada aplikasi yang dibuat. [19].



Gambar 2. 1. *Agile Modeling*

#### **2.2.4 Laravel**

*Laravel* merupakan kerangka kerja, *library*, ataupun *framework* berbasis bahasa pemrograman *PHP (Hypertext Preprocessor)* dengan menggunakan struktur *MVC (Model View Controller)* yang dikembangkan oleh Taylor Otwell, dimana terdapat banyak fungsi-fungsi kode yang diletakkan di dalam *library* kemudian diinstall di *laravel*. *Laravel* berfungsi untuk mempermudah dan mempercepat proses pembuatan *website*, *laravel* juga memiliki beberapa keunggulan seperti *template* yang ringan, tersedianya *library* yang berisi fungsi-fungsi kode program yang praktis untuk digunakan, modul yang tersedia bersifat individu dan independen, dan lain-lain [20].

#### **2.2.5 PHP (Hypertext Preprocessor)**

*PHP (Hypertext Preprocessor)* adalah suatu bahasa pemrograman atau skrip yang berjalan di sisi server yang dirancang untuk mengembangkan situs web. *PHP* disebut sebagai bahasa pemrograman sisi server karena *PHP* diproses di komputer *server*. Bahasa pemrograman *PHP* dapat digunakan diberbagai sistem operasi, seperti *windows*, *linux*, *MacOs* dan diberbagai sistem operasi lainnya [21].

#### **2.2.6 Sistem Informasi**

Sistem adalah himpunan cara atau metode kerja yang saling berhubungan untuk mencapai suatu tujuan. Sedangkan informasi adalah kumpulan data yang di olah untuk kepentingan penggunaanya, Dari pembahasan tersebut, dapat disimpulkan bahwa sistem informasi merupakan sebuah himpunan atau kumpulan yang terdiri dari berbagai komponen metode atau cara kerja yang memiliki kumpulan informasi dan diolah untuk kepentingan penggunaanya [22].

Sistem informasi adalah himpunan metode atau cara kerja yang terintegrasi untuk mendukung operasi dan manajemen serta menghasilkan informasi bagi pengambil keputusan [23].

#### **2.2.7 Black Box Testing**

*Black box testing* merupakan pengujian aplikasi yang pusat pada fungsional aplikasi tanpa harus mengetahui isi dari kode program yang dibuat, pengujian ini bertujuan untuk mengetahui dan mengatasi masalah atau *error*

maupun *bug* yang terjadi sebelum aplikasi diluncurkan dan dipakai. *Black box testing* memiliki tiga jenis pengujian, diantaranya adalah *functional testing* yang merupakan pengujian aplikasi secara fungsional pada fitur umum. Pengujian ini juga bisa digunakan untuk pengujian *software* secara keseluruhan. Selanjutnya terdapat pengujian *non-functional testing*, dan pengujian *regression testing* untuk menguji aplikasi jika aplikasi dilakukan *update* apakah terjadi *bug*. Pengujian *black box testing* memiliki beberapa teknik dalam pengujiannya, diantaranya adalah teknik *equivalence partitioning*, yaitu teknik yang dilakukan dalam pengujian sistem atau aplikasi dengan membuat *test case* terlebih dahulu untuk mengetahui menu atau fitur apa saja yang akan diuji nantinya. [24].

#### **2.2.8 CSS (Cascading Style Sheet)**

*CSS (Cascading Style Sheet)* berfungsi untuk mengatur gaya tampilan dari website yang memiliki lebih banyak kemampuan styling dari *HTML (Hyper Text Markup Language)* [25].

#### **2.2.9 Wireframe**

*Wireframe* adalah kerangka penataan suatu sistem untuk membangun aplikasi atau perangkat lunak, *wireframe* juga sering disebut sebagai rancangan kasar suatu aplikasi atau perangkat lunak, *wireframe* diterapkan pada saat awal atau sebelum pembuatan aplikasi dilakukan. *Wireframe* berfungsi untuk mempermudah tim dalam pengembangan atau perancangan aplikasi. *Wireframe* dapat dibuat dengan menggunakan *software Figma, Adobe XD, Balsamiq Wireframe* dan lain sebagainya [26].

#### **2.2.10 XAMPP**

XAMPP merupakan perangkat lunak web server yang menyediakan yang bersifat gratis yang dapat berjalan pada berbagai sistem operasi, seperti *MacOs, Linux, Windows, dan Solaris* [25].

#### **2.2.11 MYSQL**

MYSQL merupakan perangkat lunak manajemen *database* yang berfungsi untuk manajemen dan mengelola *database website* dengan

menggunakan bahasa SQL, MySQL dapat berjalan pada berbagai *platform*, sehingga dapat mempermudah tim dalam mengolah *database* [25].

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

*Entity Relationship Diagram* (ERD) adalah tahapan dasar dalam melakukan perancangan *database*. ERD digunakan untuk menggambarkan entitas yang saling ketergantungan dan terkait antara yang satu dengan yang lain. Dalam ERD, terdapat beberapa elemen dasar, yaitu relasi, atribut, entitas.

#### 1. Relasi

Relasi adalah hubungan antara entitas, misalnya antara satu entitas dengan entitas yang lain. Relasi terdiri dari antara himpunan entitas yang tidak sejenis.

#### 2. Atribut

Atribut merupakan informasi yang berada pada entitas, dimana atribut berfungsi untuk menjelaskan sifat dari entitas.

#### 3. Entitas

Entitas merupakan objek yang digambarkan secara unik dan berbeda, dimana entitas dapat berupa orang, benda, kondisi dan lain-lain [27].

### **2.2.13 Unified Model Language (UML)**

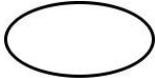
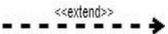
*Unified Model Language* (UML) merupakan cara untuk mendeskripsikan sistem. UML ditemukan oleh Grady Booch, Ivar Jacobson, dan James Rumbaugh. UML (*Unified Model Language*) terdiri dari lima belas diagram. Diagram ini dikelompokkan dalam *structure diagram* dan *behaviour diagram*. *Structure diagram* bertujuan untuk mendeskripsikan hubungan sistem informasi secara statis, sedangkan *behaviour diagram* berfungsi untuk mendeskripsikan saling keterkaitan antara objek dengan sistem informasi secara dinamis [28].

Berikut beberapa contoh *Unified Model Language* (UML):

#### a. *Use Case Diagram*

*Use Case Diagram* adalah pembentukan sesuatu sistem yang menggambarkan hubungan interaksi antara aktor dengan sistem yang akan dibuat [29]. Berikut simbol-simbol dan keterangannya:

Tabel 2. 2. *Use Case Diagram* [30].

Simbol	Nama	Keterangan
	Aktor	Merupakan orang atau aktor utama yang berhubungan dengan sistem
	<i>Use Case</i>	Penjelasan aksi yang dilakukan aktor
	<i>Generalization</i>	Penghubung generalisasi antara <i>use case</i> dengan aktor
	<i>Dependency</i>	Menggambarkan hubungan <i>independen</i> terhadap elemen mandiri.
	<i>Include</i>	Hubungan yang menyatakan suatu <i>use case</i> yang dibuat membutuhkan <i>use case</i> ini untuk menjalankannya.
	<i>Extend</i>	Sebuah <i>use case</i> yang dapat berdiri sendiri, tanpa ada pengaruh dari <i>use case</i> lain.

b. *Activity Diagram*

*Activity Diagram* merupakan susunan yang menggambarkan aliran kerja suatu sistem maupun menu-menu yang terdapat dalam perancangan suatu sistem, diagram ini juga menggambarkan aktivitas yang terjadi pada sistem. Berikut tabel penjelasan simbol-simbol yang ada pada *activity diagram*.

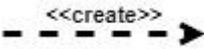
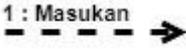
Tabel 2. 3. *Activity Diagram* [30].

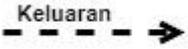
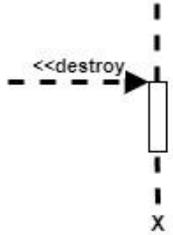
Simbol	Nama	Keterangan
	Status Awal	Simbol untuk memulai aktivitas
	Status Akhir	Simbol aktivitas selesai
	Aktivitas	Aktivitas yang dilakukan sistem
	Percabangan	Pengambilan keputusan
	Gabungan	Penggabungan lebih dari satu aktivitas
	<i>Swimlane</i>	Pemisah aktivitas yang terjadi dengan organisasi bisnis yang bersangkutan

c. *Sequence Diagram*

*Sequence Diagram* adalah diagram yang menggambarkan hubungan antara setiap objek. *sequence diagram* berfungsi untuk mengetahui hubungan antara suatu objek dengan pemiliknya serta interaksi dengan objek lain [29].

Tabel 2. 4. *Sequence Diagram* [31].

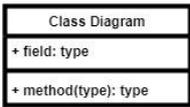
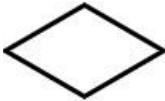
Simbol	Nama	Keterangan
	Aktor	Merupakan orang yang berhubungan dengan sistem.
	<i>Lifeline</i>	Menggambarkan objek aktif atau objek hidup.
	<i>Object</i>	Menggambarkan objek.
	Waktu Aktif	Menggambarkan objek yang sedang aktif dan berinteraksi.
	Pesan tipe <i>create</i>	Untuk membuat objek lain sesuai arah panah yang digambarkan.
	Pesan tipe <i>send</i>	Menggambarkan objek yang menginputkan data ke objek lain sesuai arah panah yang digambarkan.

	Pesan tipe <i>return</i>	Menggambarkan objek yang menghasilkan nilai return.
	Pesan tipe <i>destroy</i>	Menyatakan akhiran proses objek lain sesuai arah panah yang digambarkan.

d. *Class Diagram*

*Class Diagram* merupakan *blueprint* untuk mengklasifikasikan kumpulan objek yang memberikan gambaran umum sistem perangkat lunak yang akan dirancang atau dikembangkan [32].

Tabel 2. 5. *Class Diagram*

Simbol	Nama	Keterangan
	<i>class</i>	Simbol untuk menggambarkan kelas dan atribut kelas
	<i>activation</i>	Simbol untuk hubungan atau relasi antar kelas
	<i>lifeline</i>	Simbol yang menggambarkan ketergantungan hubungan antar kelas
	<i>Nary Association</i>	Simbol yang menggambarkan untuk menghindari asosiasi lebih dari dua objek

#### **2.2.14 Database**

*Database* merupakan kumpulan data yang tertata dengan tujuan agar mudah dikelola, disimpan, dimanipulasi, serta diakses oleh penggunanya. *Database* memiliki fungsi untuk kecepatan atau kemudahan dalam akses data, minimalisir ruang penyimpanan data, keakuratan dalam pencarian data, keamanan dalam pengelolaan data, dan kemudahan untuk penggunaan bersama (*sharing*) [25].