

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Peneliti telah melakukan studi terhadap beberapa jurnal terdahulu yang berkaitan dengan tema penelitian ini. Berikut beberapa jurnal yang menjadi referensi.

Jurnal pertama berjudul “Perancangan Sistem Informasi Aplikasi *Tracer Study* Alumni Berbasis *Website*” dengan *output* berupa sistem informasi *tracer study* alumni di Universitas Nasional yang berfungsi untuk membantu pihak universitas dalam mengetahui persebaran alumni dan membantu para alumni untuk mengetahui lowongan pekerjaan yang tersedia melalui *website tracer study* [18]. Metode pengembangan sistem yang dilakukan pada penelitian jurnal pertama menggunakan metode *waterfall*. Perbedaan dengan penelitian ini adalah aplikasi yang peneliti bangun berupa *website tracer study* yang ditujukan untuk alumni di Insitut Teknologi Telkom Purwokerto tanpa dilengkapi dengan informasi lowongan pekerjaan serta metode pengembangan aplikasi yang dipakai adalah metode *scrum*, selain itu *website tracer study* yang peneliti bangun dilengkapi dengan peta digital persebaran alumni.

Jurnal kedua berjudul “Analisis dan Pengembangan Sistem Informasi *Tracer Study* Menggunakan Metode *Waterfall* Berbasis Web” dengan *output* sebuah aplikasi berbasis *website* yang dapat melakukan pengumpulan data, pengelolaan data dan juga penelusuran data para alumni dari Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Kanjuruhan yang dilengkapi dengan fitur lowongan pekerjaan [19]. Metode pengembangan sistem yang dilakukan pada penelitian jurnal kedua menggunakan metode *waterfall* serta menggunakan bahasa pemrograman PHP dan *database* MySQL. Perbedaan dengan penelitian ini adalah *website tracer study* yang peneliti bangun ditujukan untuk seluruh

alumni Insitut Teknologi Telkom Purwokerto tanpa dilengkap dengan fitur lowongan pekerjaan serta metode yang peneliti pakai adalah metode *scrum* dengan bahasa pemrograman golang dan javascript serta *database* MySQL.

Jurnal ketiga berjudul “Implementasi Metode *Prototype* dalam Perancangan Sistem Informasi *tracer study* pada Alumni Informatika UMM” yang menghasilkan *output* sebuah sistem *tracer study* berbasis *website* yang digunakan untuk melakukan pelacakan alumni Program Studi Informatika UMM yang dilengkapi dengan fitur informasi lowongan pekerjaan [20]. Metode pengembangan sistem yang dilakukan pada penelitian jurnal ketiga menggunakan metode *prototype* serta menggunakan *framework Codeigniter (CI)* dan *database* MySQL. Perbedaan dengan penelitian ini adalah *website tracer study* yang peneliti bangun ditujukan untuk seluruh alumni Institut Teknologi Telkom Purwokerto tanpa dilengkapi dengan informasi lowongan pekerjaan serta metode yang peneliti pakai adalah metode *scrum* dengan *framework* GIN serta *database* MySQL.

Jurnal keempat berjudul “Sistem *Traer Study* dan Persebaran Alumni Berbasis Web di Universitas Islam Syekh-Yusuf Tangerang” dengan hasil berupa sistem *tracer study* dan persebaran lulusan berbasis *website* di Universitas Islam Syekh-Yusuf Tangerang yang menyajikan informasi mengenai data lulusan, informasi lowongan pekerjaan, persebaran lulusan yang dapat memudahkan admin dalam melakukan pengelolaan data lulusan serta memberikan kemudahan informasi mengenai persebaran alumni dan pengisian kuesioner yang dapat diakses secara *online* [21]. Metode pengembangan sistem yang dilakukan pada penelitian jurnal keempat menggunakan metode *extream programming* serta menggunakan *framework Codeigniter (CI)*. Perbedaan dengan penelitian ini adalah *website tracer study* yang peneliti bangun ditujukan untuk seluruh alumni Institut Teknologi Telkom Purwokerto yang dibangun menggunakan metode *scrum* serta menggunakan bahasa pemrograman golang dan *library* reactjs.

Jurnal kelima berjudul “Perancangan dan Implementasi Website Tracer Study di SMA Maitreyawira Menggunakan Framework Scrum”

dengan *output* sebuah aplikasi *tracer study* berbasis *website* yang dapat membantu pihak SMA Maitreyawira dalam melakukan penelusuran alumni serta membantu dalam mengelola data alumni sehingga dapat menjadi lebih rapih dan terorganisir [22]. Metode pengembangan sistem yang dilakukan pada penelitian jurnal kelima menggunakan metode *scrum* serta menggunakan bahasa pemrograman PHP. Perbedaan dengan penelitian ini adalah *website tracer study* yang peneliti bangun ditujukan untuk alumni Institut Teknologi Telkom Purwokerto yang dibangun menggunakan metode *scrum* serta menggunakan bahasa pemrograman golang dan javascript serta dilengkapi dengan fitur persebaran alumni kedalam peta digital.

Jurnal terakhir berjudul “Perancangan Sistem Informasi Pemetaan Alumni STIKES Syedza Sainika Berbasis Web GIS” yang menghasilkan sebuah *website tracer study* yang menampilkan informasi pemetaan alumni STIKES Syedza Sainika yang didapatkan dari hasil pengolahan data spasial dan data atribut sehingga diperoleh *latitude* dan *longitude* untuk mengetahui posisi alumni untuk ditampilkan kedalam peta digital yang dibangun menggunakan bahasa pemrograman PHP dan *database* MySQL [23]. Perbedaan dengan penelitian ini adalah *website tracer study* yang peneliti bangun ditujukan untuk alumni Institut Teknologi Telkom Purwokerto yang memiliki fitur persebaran alumni dengan memanfaatkan bahasa pemrograman golang dan javascript.

Berdasarkan uraian enam penelitian terdahulu, penelitian ini memiliki kebaruan yaitu merancang dan membangun sistem *tracer study* untuk alumni Institut Teknologi Telkom Purwokerto yang dilengkapi dengan persebaran alumni ke dalam bentuk peta digital dengan menggunakan metode *scrum* serta memanfaatkan bahasa pemrograman golang dan javascript. Penjelasan mengenai penelitian terdahulu secara singkat terdapat pada Tabel 2.1 di bawah ini.

Tabel 2.1 Penelitian sebelumnya

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
1	Rizal Bagus Pambudi; Agung Triayudi; Andrianingsih [18]	Perancangan Sistem Informasi Aplikasi <i>Tracer</i> <i>Study Alumni</i> Berbasis <i>Website</i>	2020	Jurnal Media Informatika Budidarma	Penelitian ini menghasilkan sistem informasi <i>tracer study alumni</i> di Universitas Nasional Jakarta yang berfungsi untuk membantu pihak Universitas dalam mengetahui persebaran alumni dan	Aplikasi ini melakukan kegiatan <i>tracer study</i> di Universitas Nasional Jakarta dari mulai pengumpulan data sampai penyampaian data dalam bentuk laporan. Disamping itu aplikasi ini dilengkapi dengan informasi lowongan pekerjaan dan

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
					membantu para alumni untuk mengetahui lowongan pekerjaan yang tersedia melalui <i>website tracer study</i> yang dibangun menggunakan metode <i>waterfall</i> .	juga dikembangkan dengan metode <i>waterfall</i> . Perbandingan dengan penelitian ini adalah aplikasi yang peneliti buat hanya dapat melakukan kegiatan <i>tracer study</i> di Institut Teknologi Telkom Purwokerto dan penyampain hasil dari kegiatan <i>tracer study</i>

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
						dalam bentuk peta digital dan grafik tanpa dilengkapi informasi lowongan pekerjaan dan metode yang peneliti pakai adalah metode <i>scrum</i> .
2	Siti Atmini; Alexius Endy Budianto; Moh.Ahsan [19]	Analisis Dan Pengembangan Sistem Informasi <i>Tracer Study</i> Menggunakan Metode	2019	Jurnal Media Informatika Budidarma	Penelitian ini menghasilkan sistem informasi <i>tracer study</i> berbasis <i>website</i> yang dapat melakukan	Kegiatan <i>tracer study</i> dalam penelitian ini ditujukan untuk alumni dari program studi Teknik

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
		<i>Waterfall</i> Berbasis Web			<i>tracer study</i> untuk alumni program studi teknik informatika fakultas sains dan teknologi Universitas Kanjuruhan Malang saja dan teknologi Universitas Kanjuruhan Malang yang dibangun menggunakan metode <i>waterfall</i> dan juga bahasa pemrograman PHP dengan <i>database</i> MySQL.	informatika fakultas sains dan teknologi Universitas Kanjuruhan Malang saja bukan untuk seluruh alumni dari Universitas Kanjuruhan Malang serta dapat menampilkan informasi lowongan pekerjaan yang dibangun menggunakan metode <i>waterfall</i> dan juga bahasa

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
					<p>Sistem informasi ini dilengkapi dengan fitur untuk menampilkan informasi lowongan pekerjaan.</p>	<p>pemrograman PHP dengan <i>database MySQL</i>. Sedangkan kegiatan <i>tracer study</i> yang dilakukan oleh <i>website</i> yang peneliti bangun ditujukan untuk seluruh alumni Institut Teknologi Telkom Purwokerto. Selain itu sistem yang peneliti bangun</p>

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
						menggunakan metode <i>scrum</i> dengan bahasa pemrograman <i>golang</i> dan <i>javascript</i> tanpa dilengkapi dengan fitur <i>lowongan</i> pekerjaan.
3	Dwi Agustian; Evi Dwi Wahyuni; Ilyas Nuryasin [20]	Implementasi Metode <i>Prototype</i> dalam Perancangan Sistem Informasi <i>Tracer Study</i> pada Alumni	2022	Jurnal Repositor	<i>Website</i> aplikasi ini dapat melakukan kegiatan <i>tracer study</i> untuk alumni program studi informatika UMM dan dapat	Aplikasi ini melakukan <i>tracer study</i> hanya untuk alumni program studi informatika UMM dan dapat menampilkan informasi

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
		Informatika UMM			menampilkan informasi lowongan pekerjaan. <i>Website tracer study</i> yang dibuat pada penelitian ini dibangun menggunakan metode <i>prototype</i> dan <i>framework Codeigniter</i> serta <i>database</i> MySQL	lowongan pekerjaan yang dibangun menggunakan metode <i>prototype</i> dan <i>framework Codeigniter</i> serta <i>database</i> MySQL sedangkan aplikasi yang peneliti buat dapat melakukan kegiatan <i>tracer study</i> untuk seluruh alumni Institut Teknologi Telkom

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
						Purwokerto dan tidak menampilkan informasi lowongan kerja serta aplikasi dibangun menggunakan <i>framework</i> GIN serta <i>database</i> MySQL.
4	Qiswah Mardzotillah; Mohammad Ridwan [21]	Sistem <i>Tracer Study</i> dan Persebaran Alumni Berbasis Web di Universitas Islam Syekh-Yusuf	2020	Jutis (Jurnal Teknik Informatika)	Aplikasi ini dapat melakukan <i>tracer study</i> untuk alumni Universitas Islam Syekh-Yusuf	Aplikasi ini dibangun menggunakan <i>framework</i> Codeigniter yang dapat melakukan <i>tracer study</i> bagi alumni

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
		Tangerang			Tangerang serta dapat menampilkan informasi lowongan pekerjaan. Aplikasi ini dibangun menggunakan <i>framework</i> Codeigniter dengan bahasa pemrograman PHP serta menggunakan metode <i>extream programming</i> .	Universitas Islam Syekh-Yusuf Tangerang dan juga memberikan informasi mengenai lowongan pekerjaan yang dibangun menggunakan metode <i>extream programming</i> . Perbedaan dengan aplikasi yang peneliti buat adalah dalam <i>website</i> ini tidak menampilkan

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
						informasi lowongan pekerjaan dan teknologi yang dipakai untuk membangun <i>website</i> yaitu <i>library</i> Reactjs dan bahasa pemrograman Golang dengan menggunakan metode <i>scrum</i> .
5	Edwin Charley; Syaeful Anas Aklani [22]	Perancangan dan Implementasi <i>Website Tracer Study</i> di SMA	2022	Prosiding National Conference for Community Service	Aplikasi yang berhasil dibuat pada penelitian ini berupa aplikasi berbasis	Aplikasi yang dibuat berupa aplikasi berbasis <i>website</i> dengan fitur berupa <i>export</i> data hasil

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
		Maitreyawira Menggunakan <i>Framework Scrum</i>		Project (NaCosPro)	<i>website</i> yang digunakan untuk melakukan kegiatan <i>tracer study</i> alumni SMA Maitreyawira yang dilengkapi dengan fitur <i>export data</i> ke dalam <i>file excel</i> .	<i>tracer study</i> kedalam <i>file excel</i> . Perbedaan dengan aplikasi yang peneliti buat yaitu aplikasi peneliti selain dilengkapi dengan fitur <i>export data</i> ke dalam <i>file excel</i> , dilengkapi juga dengan fitur persebaran alumni yang disajikan dalam bentuk peta digital.

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
6	Herman Susilo [23]	Perancangan Sistem Informasi Pemetaan Alumni STIKES Syedza Saintika Berbasis Web GIS	2019	Nursing	Aplikasi berbasis <i>website</i> ini dapat melakukan <i>tracer study</i> dan menampilkan data hasil <i>tracer study</i> kedalam bentuk peta persebaran alumni. Aplikasi ini dibangun menggunakan bahasa pemrograman	Aplikasi pada penelitian ini dapat menampilkan informasi persebaran alumni menggunakan peta yang dibangun menggunakan bahasa pemrograman PHP dan DBMS MySQL. Sedangkan aplikasi yang peneliti akan dibangun

No	Nama Penulis	Judul	Tahun Terbit	Penerbit	Hasil	Perbandingan
					PHP dan DBMS MySQL.	menggunakan bahasa pemrograman Golang dan <i>library</i> Reactjs.

2.2 Dasar Teori

2.2.1 Sistem Informasi

Sistem merupakan suatu kesatuan yang terdiri dari dua atau lebih elemen yang saling bekerja sama untuk mencapai tujuan yang telah ditetapkan [24]. Elemen pada sistem dapat berupa manusia, mesin ataupun sebuah proses yang saling terkait satu sama lain. Informasi merupakan data yang telah diolah sehingga menjadi nilai yang dapat dipakai dalam pengambilan keputusan [24]. Nilai dalam sebuah informasi harus bersifat informatif dan berdasarkan fakta sehingga bermanfaat dalam pengambilan keputusan. Sistem informasi merupakan integrasi antara dua atau lebih elemen yang digunakan untuk mengumpulkan, mengolah dan menampilkan data menjadi sesuatu yang lebih informatif guna mendukung keperluan dalam melakukan pengambilan keputusan [25]. Banyak contoh sistem informasi, salah satunya adalah sistem informasi perpustakaan yang memberikan informasi berupa buku-buku yang ada di perpustakaan maupun pelayanan dalam peminjaman buku.

Sistem informasi dapat diimplementasikan ke dalam beberapa media, salah satunya adalah *website*. *Website* merupakan satu atau lebih halaman web yang saling berkaitan dan dapat disertai dengan menyisipkan gambar, dokumen, video, audio ataupun *file* lain yang dapat diakses oleh pengguna [4]. *Website* haruslah memiliki URL (*Uniform Resource Locator*) yang digunakan sebagai alamat *website* sehingga pengguna dapat mengunjungi *website* dengan cara memasukan URL. Salah satu komponen dalam *website* adalah tampilan *user interface* yang berguna sebagai media penghubung antara pengguna dengan sistem, sehingga keduanya dapat berinteraksi dengan mudah [26]. *User interface* dapat dibangun dengan menggunakan banyak teknologi maupun *library* salah satunya adalah *library* dari Javascript yaitu ReactJS.

Library merupakan sumber eksternal yang berisikan kode-kode siap pakai yang dapat membantu dalam pembuatan suatu program dengan cara kode yang ditulis akan memanggil kode dari *library* [27]. *Library* hanya dapat melakukan satu tugas saja, berbeda dengan *framework* yang merupakan kumpulan *library* sehingga dapat melakukan banyak tugas. Javascript merupakan bahasa pemrograman yang dieksekusi disisi *client*, dalam hal ini adalah *browser* seperti chrome dan mozilla yang berfungsi untuk memberikan fungsionalitas pada *website* [28]. Javascript juga dapat berdiri sendiri sebagai *file* javascript ataupun dapat juga disisipkan pada *file* lain seperti HTML.

ReactJS merupakan salah satu *library* Javascript yang bersifat *open source* yang dapat digunakan secara gratis untuk membuat komponen *user interface* yang dapat dipakai secara berulang. Manfaat dari komponen yang dapat dipakai secara berulang adalah tidak perlu memakan banyak waktu untuk membuatnya dari awal ketika akan menggunakan komponen yang sama di tempat yang berbeda [29]. Banyak perusahaan besar seperti Facebook, Instagram dan Airbnb yang telah menggunakan ReactJs dalam pembuatan aplikasi mereka.

2.2.2 LeafletJS

LeafletJS merupakan *library open source* dari Javascript yang memiliki fungsi sebagai penyedia layanan untuk membantu pembuatan peta digital yang interaktif [30]. LeafletJS banyak menyediakan *plugin* yang dapat dipakai dalam pembuatan peta digital. Tersedianya banyak *plugin* membuat pengguna dapat dengan mudah membuat peta digital sesuai dengan kebutuhan. Selain tersedianya banyak *plugin* seperti *marker* ataupun *popup*, LeafletJS juga mampu mengolah *file* geojson yang berisi informasi geografis [31].

2.2.3 *Tracer Study*

Tracer study atau penelusuran lulusan merupakan salah satu kegiatan yang dilakukan oleh perguruan tinggi yang pada umumnya dilakukan setiap satu sampai tiga tahun sekali dengan tujuan untuk memperoleh data tentang lulusannya berupa identitas pribadi alumni, riwayat pendidikan, status pekerjaan, *feedback* alumni terhadap metode pembelajaran selama di perkuliahan dan lain sebagainya [32]. Selain itu tujuan utama dari kegiatan *tracer study* adalah untuk mengetahui keterhubungan antara kurikulum yang diterapkan dengan kebutuhan pasar tentang tenaga kerja profesional. Data hasil dari *tracer study* akan diolah oleh pihak perguruan tinggi sehingga didapat informasi yang bisa dijadikan acuan pihak perguruan tinggi dalam melakukan evaluasi kurikulum dan juga bisa menjadi bahan acuan mahasiswa baru untuk bergabung bersama perguruan tinggi [33]. *Tracer study* menjadi salah satu komponen penting dalam menunjang keberhasilan akreditasi pada perguruan tinggi.

2.2.4 *Database*

Database merupakan kumpulan data yang saling berkaitan dan terstruktur yang mana data akan diproses dan dikelola sesuai dengan kebutuhan aplikasi [34]. Data pada *database* disimpan didalam tabel sehingga ketika akan melakukan manipulasi atau pengelolaan data, tabel tersebut akan menjadi rujukan. Pengelolaan *database* merupakan suatu kegiatan yang bertujuan untuk mengolah data sehingga data dapat disusun, diurutkan ataupun diambil sewaktu waktu dengan tujuan data hasil pengolahan akan menjadi informasi yang dapat ditampilkan kepada pengguna [35]. Saat ini banyak *database* yang dapat digunakan dalam mengelola data, salah satunya adalah *database* MySQL. *Database* MySQL memiliki beberapa kelebihan seperti penggunaannya yang terbilang cukup mudah karena didukung oleh *interface* yang ramah pengguna sehingga tidak perlu seseorang yang ahli dalam *coding* untuk dapat

mengoperasikannya, *database* MySQL juga bersifat *open source* sehingga gratis untuk dipakai oleh siapa saja [36].

MySQL merupakan DBMS (*Database Management System*) yang berfungsi menampung, mengolah dan menyimpan data pada *database* menggunakan bahasa SQL (*Structured Query Language*) [37]. SQL merupakan sebuah bahasa *scripting* yang digunakan untuk mengolah *database*, seperti memasukan data, mengambil data, mengubah data maupun menghapus data [38]. MySQL harus berjalan di atas *web server* agar bisa digunakan, salah satunya adalah *web server apache* yang sudah tertanam di *software* XAMPP. XAMPP merupakan sebuah *software open source* yang dapat dipakai oleh siapa saja secara gratis yang didalamnya sudah terpasang *web server apache*, *database* MYSQL, bahasa pemrograman PHP dan masih banyak lagi [39]. *Database* akan lebih mudah diintegrasikan dengan berbagai macam sistem dan akan lebih dinamis jika dikombinasikan dengan bahasa pemrograman *server side*, salah satunya adalah bahasa pemrograman Golang (*Go Language*).

Golang merupakan salah satu bahasa pemrograman *server side* yang bersifat *open source* yang dikembangkan di Google oleh Rob Pike, Robert Griesemer dan Ken Thompson beserta beberapa kontributor lainnya dalam komunitas pengembang *open source* yang mulai dipublikasikan pada bulan November 2009 [8]. Pada saat ini bahasa Golang mulai banyak digunakan oleh perusahaan-perusahaan besar ataupun *startup* yang bergerak dibidang teknologi karena Golang memiliki beberapa kelebihan seperti :

1. Mendukung konkurensi dengan cukup baik dan memiliki pengaplikasian yang tergolong mudah.
2. Golang merupakan bahasa yang bersifat *open source*.
3. Memiliki sistem *garbage collection* yang baik
4. *Reliable* dan cepat dalam skala besar.


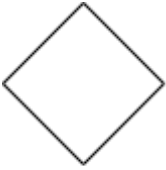
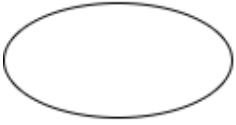

5. Memiliki sintaks yang bersih sehingga tidak mengotori sistem terlalu berlebihan.

Data dari *database* hasil pengolahan yang telah dilakukan oleh Golang akan disimpan ke dalam format JSON (*Javascript Object Notation*) sehingga dapat dikonsumsi oleh *library* ReactJs menggunakan API (*Application Programming Interface*) dengan arsitektur REST (*Representational State Transfer*). JSON merupakan salah satu format dengan struktur *key* dan *value* yang digunakan untuk melakukan pertukaran data yang memiliki keunggulan berupa mudah dibaca dan ditulis serta mudah diterjemahkan oleh mesin [40]. API merupakan sebuah *interface* yang memungkinkan terjadinya komunikasi atau pertukaran data antara dua atau lebih sistem yang berbeda secara bersama dengan tujuan untuk mempercepat proses pengembangan aplikasi [41]. REST merupakan arsitektur API yang menggunakan HTTP (*Hypertext Transfer Protocol*) sebagai media pertukaran data, sehingga *client* (pengguna) hanya perlu mengirim *request* kepada *server* (penyedia) menggunakan *method* *GET*, *POST*, *PUT* atau *DELETE* dengan memasukan alamat API sehingga *server* akan memberikan *response* berupa data yang telah diolah dalam bentuk JSON [42].

2.2.5 ERD (*Entity Relational Diagram*)

ERD (*Entity Relational Diagram*) merupakan diagram yang menggambarkan daftar tabel pada *database* beserta dengan detail relasi dan atributnya [43][44]. Relasi yang dimaksud menggambarkan hubungan keterkaitan antara satu tabel dengan tabel lain sedangkan atribut menggambarkan data dari dunia nyata seperti alamat, nama ataupun usia. ERD memiliki beberapa simbol diantaranya [45]:

Tabel 2.2 Simbol ERD (*Entity Relational Diagram*)

Simbol	Keterangan
 <p data-bbox="523 589 603 618">Entitas</p>	<p data-bbox="762 432 1337 622">Simbol yang digunakan untuk menggambarkan objek, sehingga dapat dibedakan antara objek dengan objek lain. Pada entitas akan melekat atribut.</p>
 <p data-bbox="523 864 603 893">Relasi</p>	<p data-bbox="762 663 1337 797">Simbol yang digunakan untuk menggambarkan hubungan atau keterkaitan anatar entitas.</p>
 <p data-bbox="523 1137 603 1167">Atribut</p>	<p data-bbox="762 936 1337 1182">Simbol yang digunakan untuk menjelaskan dengan detail terkait entitas. Atribut yang dimaksud dapat berupa nama, alamat, usia atau segala hal yang menjadi identitas dari entitas.</p>
 <p data-bbox="536 1305 590 1335">Alur</p>	<p data-bbox="762 1209 1337 1400">Simbol yang digunakan sebagai penghubung antara entitas dan atribut ataupun dipakai untuk menghubungkan relasi dengan entitas.</p>

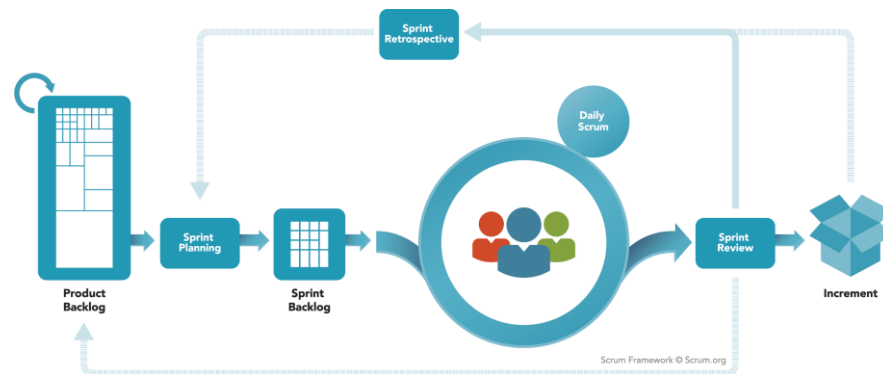
2.2.6 Black Box Testing

Black box testing merupakan metode pengujian perangkat lunak yang dilakukan untuk melihat hasil *input* dan *output* dari suatu fungsi yang telah dibuat tanpa harus mengetahui kode dari perangkat lunak tersebut [46]. Pengujian dilakukan menggunakan media *browser* seperti chrome, mozilla ataupun microsoft edge. *Black box testing* dapat membantu mendeteksi kesalahan pengkodean seperti fungsi yang salah, kesalahan pada *interface*, kesalahan struktur data atau kesalahan dalam mengakses *database* [47]. *Black box testing*

memiliki keunggulan seperti pengujian dilakukan dari sudut pandang pengguna dan penguji tidak perlu memiliki kemampuan *coding* [48].

2.2.7 Metode Scrum

Scrum merupakan kerangka kerja atau *framework* dari metode *agile* yang digunakan dalam pengembangan produk yang terbilang kompleks dengan mengimplementasikan pendekatan yang bertahap dan berkelanjutan untuk meningkatkan kemampuan dalam memprediksi dan mengendalikan resiko yang terjadi selama pengembangan berlangsung [49][50]. Metode *scrum* memiliki tujuh tahapan penting yaitu *product backlog*, *sprint planning*, *sprint backlog*, *daily scrum*, *sprint review*, *increment* dan *sprint retrospective* yang mana *sprint* tempat terjadinya *daily scrum* merupakan tahapan utama dimana para *developer* menulis *code* sesuai dengan desain dan ketentuan yang telah ditetapkan pada tahap sebelumnya, dengan durasi *sprint* memakan waktu selama satu sampai empat minggu [49][51]. Metode *scrum* memiliki keunggulan berupa mengutamakan kecepatan dalam melakukan pengembangan aplikasi dan dapat menerima perubahan yang dinamis, sehingga tidak terpaku dengan perencanaan awal [51]. Selain itu metode *scrum* tidak bersifat *sequential* artinya ketika ada salah satu tahap yang terjadi masalah maka tidak akan menjadi *blocking* terhadap tahap selanjutnya.



Gambar 2. 1 Metode *scrum* [14]

Gambar 2.1 merupakan tahapan yang harus dilalui dalam pengembangan menggunakan metode *scrum*. *Scrum* memiliki beberapa tahapan, diantaranya:

1. *Product backlog*

Product backlog merupakan daftar fitur yang akan dikembangkan selama masa *sprint* [52]. *Product backlog* menjadi inti pekerjaan bagi para *developer* selama *sprint* berlangsung. Fitur-fitur yang masuk kedalam *product backlog* dapat diatur tingkat prioritasnya sehingga akan memudahkan proses pada *sprint backlog* tentang *backlog* mana yang akan masuk *sprint* berdasarkan tingkat prioritasnya.

2. *Sprint Planning*

Sprint planning merupakan rapat para *scrum team* yang membahas dan menyusun pekerjaan atau *backlog* apa saja yang akan masuk kedalam tahap *sprint* berdasarkan prioritas yang telah ditetapkan pada tahap sebelumnya [52]. Pada tahap ini *developer* dapat memberikan estimasi waktu pengerjaan untuk setiap fitur salah satunya dengan mengacu pada bilangan fibonacci seperti 1,3,5,8 yang merepresentasikan jumlah hari pengerjaan. Tak hanya itu *developer* dapat menentukan DOD (*Definition of Done*) dari fitur yang akan dikerjakan. Pada tahap ini akan menghasilkan *output* berupa *sprint goal* dan *product backlog* yang dipilih untuk

masuk ke tahap *sprint* yang mana *output* ini disebut dengan *sprint backlog*.

3. *Daily Scrum*

Daily scrum terjadi pada tahap *sprint*. *Sprint* merupakan tahapan utama dalam *scrum* dimana semua *product backlog* yang telah dipilih akan dikembangkan dan diubah menjadi *product* yang sesuai dengan *sprint goal* [52]. *Sprint* berlangsung selama dua sampai empat minggu tergantung pada kesepakatan. Pada tahap *sprint* setiap *scrum team* akan mengadakan rapat setiap harinya selama kurang lebih 15 menit yang akan membahas kendala apa yang dialami pada hari kemarin dan pekerjaan apa yang akan diselesaikan pada hari ini, rapat ini biasa disebut dengan *daily scrum* atau *daily standup*. Tak hanya *daily scrum*, pengujian fitur juga akan dilakukan pada tahap *sprint*. Pada tahap ini akan menghasilkan *output* berupa *backlog* yang telah diimplementasikan kedalam bentuk kode program.

4. *Sprint Review*

Sprint review merupakan tahap mempresentasikan atau mendemokan fitur yang telah berhasil dikembangkan selama masa *sprint* dengan memastikan terpenuhinya DOD (*Definition Of Done*) [52]. DOD sangat penting keberadaannya karena ini akan mendefinisikan sebuah fitur sudah dikatakan selesai jika telah memenuhi DOD dari fitur tersebut. DOD biasanya ditetapkan sebelum tahap *sprint* dimulai. Ketika *backlog* atau fitur telah selesai dan telah memenuhi DOD maka akan dilakukan akumulasi total *backlog* yang telah memenuhi DOD pada *sprint* sekarang dan *sprint* sebelumnya, proses ini biasa disebut dengan *increment*. Apabila fitur tidak memenuhi DOD dan tidak lulus pengujian maka fitur akan masuk kembali ke *backlog* dan akan dikerjakan pada *sprint* selanjutnya.

5. *Sprint Retrospective*

Tahap ini merupakan tahap terakhir dari *scrum* yang akan membahas mengenai evaluasi terhadap *sprint* yang telah berlangsung [52]. Evaluasi tersebut mencakup apa saja yang berjalan baik selama *sprint*, apa saja yang berjalan tidak baik selama *sprint*, apa saja *improvement* yang dapat dilakukan untuk *sprint* selanjutnya dan bagaimana cara melakukan *improvement* tersebut. *Sprint retrospective* diadakan pada penghujung *sprint* dan dihadiri oleh semua *scrum team*. Pada tahap ini apabila *product backlog* telah kosong maka proses *scrum* akan berakhir, apabila sebaliknya maka proses *scrum* akan berulang ke tahap *sprint planning*.







2.2.8 UML (*Unified Modelling Language*)

UML merupakan sebuah bahasa standar yang digunakan untuk membuat gambaran, dokumentasi ataupun perencanaan pada pembangunan sistem perangkat lunak [53]. UML juga sering disebut sebagai *blueprint* dari perangkat lunak yang akan dibangun, *blueprint* ini akan menjadi panduan bagi *developer* dalam menyusun kode program. UML pada dasarnya berbentuk diagram yang memvisualisasikan perancangan dari sistem perangkat lunak, berikut beberapa diagram pada UML:

1. *Use Case Diagram*

Use case diagram merupakan sebuah diagram yang memperlihatkan bagaimana sebuah sistem berinteraksi dengan pengguna atau biasa disebut dengan aktor [53]. Diagram ini akan memperlihatkan siapa saja aktor yang dapat mengakses sistem dan apa saja yang dapat dilakukan oleh sistem. Ada beberapa simbol yang dipakai dalam membuat *use case diagram*, diantaranya sebagai berikut [54]:

Tabel 2.3 Simbol *use case diagram*




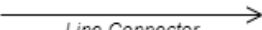
Simbol	Keterangan
 <p style="text-align: center;">Actor</p>	<p>Simbol yang digunakan sebagai pengganti manusia, alat ataupun sistem yang lain ketika berinteraksi dengan <i>use case</i>.</p>
 <p style="text-align: center;">Use Case</p>	<p>Simbol yang digunakan sebagai interaksi antara aktor dengan sistem.</p>
 <p style="text-align: center;">Association</p>	<p>Simbol yang digunakan untuk menghubungkan aktor dengan <i>use case</i>.</p>
 <p style="text-align: center;">Generalization</p>	<p>Simbol yang digunakan untuk menunjukkan penurunan sifat antara dua <i>use case</i> atau dua <i>actor</i>.</p>
 <p style="text-align: center;">Include</p>	<p>Simbol yang digunakan untuk menunjukkan bahwa satu <i>use case</i> merupakan bagian dari <i>use case</i> lainnya. Simbol ini akan mengikat dua <i>use case</i> atau lebih sehingga tidak dapat dipisahkan.</p>
 <p style="text-align: center;">Extend</p>	<p>Simbol yang digunakan untuk menunjukkan bahwa satu <i>use case</i> merupakan bagian tambahan dari <i>use case</i> lain. Simbol ini tidak mengikat <i>use case</i> sehingga proses dapat</p>


Simbol	Keterangan
	terpisahkan dan bersifat opsional.

2. Activity Diagram

Activity diagram merupakan sebuah diagram vertikal yang menjelaskan bagaimana suatu sistem bekerja dan berinteraksi dengan objek lainnya [53]. Objek pada *activity diagram* dapat berupa manusia, perangkat lunak, perangkat keras maupun *database*. Ada beberapa simbol yang dipakai dalam membuat *activity diagram*, diantaranya sebagai berikut [54]:

Tabel 2.4 Simbol *activity diagram*




Simbol	Keterangan
 <i>Initial Node</i>	Simbol yang digunakan sebagai penanda objek dibentuk dan aktifitas dimulai.
 <i>Activity</i>	Simbol yang digunakan untuk menggambarkan aktifitas yang terjadi pada objek.
 <i>Decision</i>	Simbol yang digunakan untuk menggambarkan tindakan yang harus diambil dalam keadaan tertentu.
 <i>Line Connector</i>	Simbol yang digunakan untuk menghubungkan satu <i>activity</i> dengan <i>activity</i> lainnya.


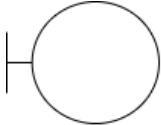
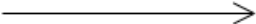
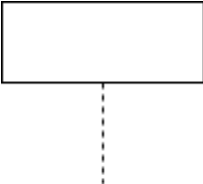
Simbol	Keterangan
 <i>Final Node</i>	Simbol yang digunakan sebagai penanda berakhirnya semua kegiatan dalam sistem.

3. *Sequence Diagram*

Sequence diagram merupakan sebuah diagram yang memperlihatkan secara detail interaksi antara objek pada sebuah sistem [53]. Objek pada diagram ini dapat berupa manusia atau aktor, *database* ataupun *software* yang terlibat dalam sebuah proses atau *use case*. Selain terjadinya interaksi antar objek, terjadi pula visualisasi pesan atau data apa saja yang dikirim oleh masing-masing objek. Ada beberapa simbol yang dipakai dalam membuat *sequence diagram*, diantaranya sebagai berikut [54][55]:

Tabel 2.5 Simbol *sequence diagram*

Simbol	Keterangan
 <i>Actor</i>	Simbol yang digunakan untuk menggambarkan pengguna yang akan berinteraksi dengan sistem.
 <i>Entity Class</i>	Simbol yang digunakan untuk menyimpan data atau informasi.
 <i>Activation</i>	Simbol yang digunakan untuk menggambarkan suatu aktivitas yang terjadi. Panjang kotak <i>activation</i> selaras dengan lamanya

Simbol	Keterangan
	aktivitas berlangsung.
 <p data-bbox="699 577 815 600"><i>Controll Class</i></p>	Simbol yang digunakan untuk mengatur perilaku sistem dan mengontrol alur kerja pada sistem
 <p data-bbox="699 801 815 824"><i>Boundary Class</i></p>	Simbol yang digunakan untuk menggambarkan interaksi antara aktor dengan sistem.
 <p data-bbox="719 976 802 999"><i>Message</i></p>	Simbol yang digunakan untuk menggambarkan pesan yang dikirim oleh objek.
 <p data-bbox="715 1308 799 1330"><i>Lifeline</i></p>	Simbol yang digunakan untuk menggambarkan durasi pemakaian objek. Didalam <i>lifeline</i> terdapat <i>activation</i> .