

BAB II

KAJIAN LITERATUR

2.1 SCRUM

Sebuah kerangka kerja dimana orang-orang dapat mengatasi masalah kompleks adaptif, dimana pada saat bersamaan mereka juga menghantarkan produk dengan nilai setinggi mungkin secara produktif dan kreatif (Ken Schwaber dan Jeff Sutherland: 2017).



Gambar 1 Alur kerja SCRUM

Scrum adalah kerangka kerja proses yang telah digunakan untuk mengelola pengembangan produk kompleks sejak awal tahun 1990-an. Scrum bukanlah sebuah proses, teknik, ataupun metodologi. Akan tetapi Scrum adalah sebuah kerangka kerja yang menggunakan bermacam proses dan teknik di dalamnya. Kerangka kerja Scrum terdiri dari Scrum Team dan peran-peran, acara-acara, artefak-artefak dan aturan-aturan terkait. Scrum mengekspos ketidak-efektifan dari manajemen produk dan teknik kerja,

sehingga dapat secara terus-menerus meningkatkan kinerja produk, tim, dan lingkungan kerja.

Scrum dibangun di atas teori proses kontrol empiris atau bisa disebut empirisme. Empirisme menyatakan bahwa pengetahuan datang dari pengalaman dan pengambilan keputusan didasari oleh apa yang telah diketahui hingga saat ini. Scrum menggunakan pendekatan yang bertahap dan berkelanjutan untuk mengoptimalkan kemampuan prediksi dan mengendalikan risiko.

Tiga pilar yang memperkokoh setiap implementasi dari proses kontrol empiris adalah:

1) Transparansi

Aspek signifikan dari sebuah proses harus dapat dilihat oleh orang-orang yang bertanggung jawab terhadap dampaknya. Transparansi membutuhkan aspek-aspek tersebut ditentukan oleh standar baku sehingga para pengamat memiliki pemahaman yang sama terhadap apa yang sedang ditinjau.

2) Inspeksi

Pengguna Scrum harus sering menginspeksi artefak Scrum dan perkembangan menuju Sprint Goal agar mereka dapat mendeteksi adanya variasi hasil yang tidak diharapkan.

3) Adaptasi

Jika pemeriksa menemukan bahwa ada satu hal atau lebih dari proses yang menyimpang di luar ambang batas yang bisa diterima yang dapat menyebabkan produk tidak bisa diterima, maka proses atau materi yang sedang diproses harus diubah. Perubahan harus dilakukan secepatnya untuk meminimalkan penyimpangan yang semakin jauh.

2.1.1 Scrum Team

Bentuk tim dalam Scrum dirancang untuk mengoptimalkan fleksibilitas, kreativitas dan produktivitas. *Scrum Team* menghantarkan produk secara

iteratif dan inkremental guna memaksimalkan peluang untuk mendapatkan umpan balik. Penghantaran produk “Selesai” secara inkremental dilakukan guna memastikan versi produk yang berpotensi untuk digunakan selalu siap tersedia.

1) Product Owner

Product Owner adalah orang yang bertanggung jawab untuk memaksimalkan nilai bisnis dari produk yang dihasilkan oleh Development Team. Cara melakukannya sangat bervariasi antar organisasi, Scrum Team dan individu. *Product Owner* adalah satu-satunya orang yang bertanggung jawab dalam pengelolaan *Product Backlog*.

2) Development Team

Development Team terdiri dari para ahli profesi yang bekerja untuk menghantarkan Increment “Selesai” yang berpotensi untuk dirilis di setiap akhir Sprint. Increment “Selesai” wajib tersedia pada saat Sprint Review. Hanya anggota dari Development Team yang membuat Increment ini. *Development Team* dibentuk dan diberikan wewenang oleh organisasi untuk menyusun dan mengelola pekerjaan mereka sendiri. Hasil sinergi dari tim ini akan mengoptimalkan efisiensi dan efektivitas *Development Team* secara keseluruhan.

4) Scrum Master

Scrum Master bertanggung jawab untuk mengenalkan dan menyokong penggunaan Scrum sebagaimana dijelaskan di dalam Panduan Scrum ini. Scrum Master melakukan ini dengan membantu orang-orang agar dapat memahami teori, praktik-praktik, aturan-aturan dan tata nilai Scrum.

Scrum Master adalah pemimpin yang melayani Scrum Team. Scrum Master membantu orang-orang di luar Scrum Team untuk dapat memahami interaksi mana yang bermanfaat dan tidak bermanfaat. Scrum Master membantu orang-orang untuk mengubah interaksi ini guna memaksimalkan nilai bisnis yang dihasilkan oleh Scrum Team.

2.1.2 Acara Scrum

Acara-acara wajib dalam Scrum diselenggarakan guna terciptanya kerutinan dan mengurangi pertemuan lain yang bukan merupakan bagian dari Scrum. Setiap acara memiliki batasan waktu, artinya setiap acara memiliki durasi waktu maksimal.

1) **Sprint**

Jantung dari Scrum adalah *Sprint*, yaitu sebuah batasan waktu dengan durasi satu bulan atau kurang, dimana terdapat proses pembuatan *Increment* yang “Selesai”, dapat digunakan dan berpotensi untuk dirilis. *Sprint* memiliki durasi yang konsisten sepanjang daur hidup pengembangan produk.

2) **Sprint Planning**

Pekerjaan yang akan dikerjakan di *Sprint* direncanakan pada saat *Sprint Planning*. Perencanaan ini dilakukan secara kolaboratif oleh seluruh anggota *Scrum Team*. *Sprint Planning* memiliki batasan waktu maksimal delapan jam untuk *Sprint* yang berdurasi satu bulan. *Scrum Master* memastikan acara ini diselenggarakan dan peserta memahami tujuannya. *Scrum Master* mengedukasi *Scrum Team* untuk menjaganya di dalam batasan waktu.

3) **Sprint Goal**

Sprint Goal adalah sebuah objektif untuk *Sprint* yang dapat dicapai lewat pengimplementasian *Product Backlog*. *Sprint Goal* merupakan panduan bagi *Development Team* untuk menjawab pertanyaan mengapa mereka mengembangkan *Increment*. *Sprint Goal* dibuat pada pertemuan *Sprint Planning*. *Sprint* memberikan ruang fleksibilitas mengenai fungsionalitas yang akan diimplementasikan di dalam *Sprint*. *Product Backlog* item terpilih merupakan satu fungsi yang selaras yang bisa jadi *Sprint Goal*. *Sprint Goal* bisa saja menghubungkan pekerjaan yang tidak memiliki keterkaitan agar *Development Team* tidak bekerja dari instruksi kerja yang berbeda-beda.

4) Daily Scrum

Daily Scrum adalah acara untuk Development Team yang memiliki batasan waktu 15 menit. Acara ini dilakukan setiap hari selama Sprint berlangsung. Di acara ini, Development Team membuat rencana kerja untuk 24 jam ke depan. Acara ini mengoptimalkan kolaborasi dan performa dari tim dengan melakukan inspeksi pada pekerjaan yang dilakukan semenjak Daily Scrum sebelumnya dan melakukan prakiraan terhadap pekerjaan selanjutnya di dalam Sprint. Daily Scrum dilakukan di waktu dan tempat yang sama setiap harinya untuk mengurangi kompleksitas.

5) Sprint Review

Sprint Review diselenggarakan di akhir Sprint untuk menginspeksi Increment dan mengadaptasi *Product Backlog* bila diperlukan. Pada saat Sprint Review, Scrum Team dan pemegang kepentingan berkolaborasi untuk meninjau apa yang sudah diselesaikan di Sprint. Berdasarkan hasil tinjauan tersebut dan perubahan terhadap *Product Backlog* di dalam Sprint, hadirin berkolaborasi untuk menentukan pekerjaan selanjutnya yang dapat dilakukan untuk mengoptimalkan nilai bisnis. Ini adalah pertemuan informal, bukan pertemuan laporan status, dan presentasi Increment dilakukan guna mendapatkan umpan balik dan mengembangkan kemampuan kolaborasi.

6) Sprint Retrospective

Sprint Retrospective adalah sebuah kesempatan bagi Scrum Team untuk menginspeksi dirinya sendiri dan membuat perencanaan mengenai peningkatan yang akan dilakukan di Sprint berikutnya. Sprint Retrospective terselenggara setelah Sprint Review dan sebelum Sprint Planning berikutnya. Acara ini diselenggarakan paling lama tiga jam untuk Sprint yang berdurasi satu bulan. Untuk Sprint yang lebih singkat, durasi acara ini biasanya lebih singkat.

Scrum Master memastikan acara ini terselenggara dan setiap peserta memahami tujuannya.

Scrum Master memastikan bahwa acara ini bernuansa positif dan produktif. Scrum Master mendukung peserta untuk menjaganya di dalam batasan waktu. Scrum Master turut berpartisipasi sebagai rekan kerja sejawat yang bertanggung gugat terhadap proses Scrum di pertemuan ini.

2.1.3 Artefak-artefak Scrum

Artefak Scrum merepresentasikan pekerjaan atau nilai bisnis guna terciptanya transparansi dan kesempatan untuk menginspeksi dan mengadaptasi. Artefak-artefak yang dijabarkan Scrum dirancang sedemikian rupa untuk memaksimalkan transparansi informasi utama agar setiap orang memiliki pemahaman yang sama mengenai artefak tersebut.

1) *Product Backlog*

Product Backlog adalah daftar terurut semua hal yang telah diketahui hingga saat ini harus ada di dalam produk. *Product Backlog* adalah satu-satunya sumber kebutuhan untuk semua perubahan yang perlu diberlakukan terhadap produk. *Product Owner* bertanggung jawab terhadap *Product Backlog*, termasuk isi, ketersediaan dan urutannya.

2) *Sprint Backlog*

Sprint Backlog adalah daftar *Product Backlog* item yang terpilih untuk *Sprint* ditambah perencanaan untuk menghantarkan Increment dan mencapai *Sprint Goal*. *Sprint Backlog* adalah prakiraan dari Development Team mengenai fungsionalitas yang akan masuk ke dalam Increment berikutnya dan pekerjaan yang perlu dikerjakan untuk menghantarkan fungsionalitasnya menjadi Increment yang “Selesai”.

3) *Increment*

Increment adalah manifestasi dari *Product Backlog* item yang diselesaikan dalam Sprint dan total nilai bisnis Increment dari seluruh Sprint yang lalu. Di akhir Sprint, Increment yang baru harus “Selesai”, yang artinya Increment tersebut harus berada pada kondisi yang dapat digunakan dan sesuai dengan definisi “Selesai” milik Scrum Team. Sebuah Increment adalah hasil pekerjaan yang bisa diinspeksi dan telah selesai guna mendukung empirisme di akhir Sprint. Increment adalah sebuah langkah kecil menuju sebuah visi ataupun tujuan. Increment harus bersifat dapat digunakan terlepas apakah *Product Owner* memutuskan untuk merilisnya atau tidak.

2.2 Laporan Keuangan

Menurut M. Sadeli (2002:2), pengertian laporan keuangan adalah hasil dari proses akuntansi dan informasi histories yang di dalamnya terdapat proses identifikasi, pengukuran, dan laporan informasi ekonomi sebagai bahan pertimbangan dalam mengambil keputusan yang tepat.

Menurut Sofyan S. Harahap (2006:105), pengertian laporan keuangan adalah suatu laporan yang menggambarkan kondisi keuangan dan hasil usaha suatu perusahaan pada jangka waktu tertentu.

Menurut Munawir Sjadzali (2010:5), pengertian laporan keuangan adalah proses akuntansi yang dapat digunakan sebagai alat untuk mengkomunikasikan data keuangan.

Menurut Kasmir (2013:7), laporan keuangan adalah suatu laporan yang menunjukkan kondisi keuangan perusahaan saat ini atau periode berikutnya. Kesimpulannya laporan keuangan adalah hasil akhir dari proses pencatatan transaksi keuangan suatu perusahaan yang menunjukkan kondisi keuangan perusahaan tersebut pada satu periode akuntansi dan merupakan gambaran umum mengenai kinerja suatu perusahaan.

2.3 Android

Android adalah sebuah sistem operasi yang dirancang oleh Google yang berbasis kernel Linux dan berbagai perangkat lunak *Open Source* lainnya serta biasa digunakan untuk perangkat dengan layar sentuh seperti smartphone dan tablet.

Walau Android adalah milik Google, Google tidak memungut biaya bagi siapapun yang ingin menggunakan Android karena Android adalah perangkat lunak *open source*. Hal ini sangat bertolak belakang dengan sistem operasi Windows 10 Mobile milik Microsoft, dimana perusahaan harus membayar royalti jika ingin menggunakan sistem operasi tersebut. Sedangkan iOS milik Apple hanya bisa digunakan di perangkat iPhone dan iPad saja dan Apple tidak merilisnya untuk perangkat lain.

Lembaga penelitian di Amerika Serikat, Pew Research Center menerbitkan laporan tentang negara dengan orang dewasa terbanyak yang menggunakan smartphone, seperti dilansir laman *weforum*, beberapa hari yang lalu. Indonesia termasuk salah satu negara yang disurvei. Survei dilakukan Pew Research Center dengan responden 30.133 orang di 27 negara yang dilakukan pada 14 Mei sampai 12 Agustus 2018.

Dalam survei untuk melihat perbandingan kepemilikan smartphone dan telepon seluler biasa di antara orang dewasa tersebut, terungkap posisi Indonesia berada di urutan ke-24 dari 27 negara. Dari seluruh orang dewasa pemilik HP di Indonesia, 42 persen memiliki smartphone, 28 mempunyai HP biasa, dan 29 persen tidak memiliki HP.

Pertumbuhan pengguna smartphone di Indonesia lumayan tinggi. Untuk pemakai muda (18-34 tahun) kepemilikan smartphone meningkat dari 39 persen menjadi 66 persen dari 2015-2018. Sedangkan untuk pengguna HP berusia di atas 50 tahun, pemakai smartphone juga naik dari 2 persen pada 2015 menjadi 13 persen pada 2018.

2.4 Android Studio

Android Studio adalah IDE pemrograman Android resmi dari Google yang dikembangkan dari IntelliJ. Sebelum ada Android Studio, programmer Android telah menggunakan Eclipse. Eclipse adalah IDE pemrograman Android sebelum munculnya Android Studio. Bisa dibilang Google telah berpaling dari Eclipse dan menjadikan Android Studio sebagai IDE resminya. Dikarenakan sudah meresmikan Android Studio pada tanggal 16 Mei 2013, Google menghentikan support ADT ke Eclipse tak lama kemudian dan ADT resmi hanya didapatkan oleh Android Studio.

Platform android terdiri dari Sistem Operasi berbasis Linux, sebuah GUI (Graphic User Interface), sebuah web browser dan Aplikasi Studio End-User yang dapat di download dan juga para pengembang bisa dengan leluasa berkarya serta menciptakan aplikasi yang terbaik dan terbuka untuk digunakan oleh berbagai macam perangkat.

2.5 Kotlin

Java	POJO	Kotlin	M
<pre>class Person { private String name; public Person(String name) { this.name = name; } public String getName() { return name; } public void setName(String name) { this.name = name; } // toString... // hashCode... // equals... // copy... }</pre>		<pre>data class Person(val name: String)</pre>	
Java	Code	Kotlin	M
<pre>public void createAndPrintPerson() { String name = "Pieter"; Person person = new Person(name); printName(person.getName()); // Prints: Pieter Otten }</pre>		<pre>fun createAndPrintPerson() { val name = "Pieter" val person = Person(name) printName(person.name) // Prints: Pieter Otten }</pre>	

Gambar 2 Perbedaan pada java dan kotlin

Kotlin adalah bahasa pemrograman yang merupakan “penyempurnaan” dari bahasa pemrograman Java untuk pengembangan aplikasi Android. Kotlin awalnya dikembangkan oleh JetBrains, perusahaan dibalik IntelliJ IDEA. Setelah melalui banyak perkembangan, JetBrains merilis Kotlin secara *open source* dan kini setelah perkembangannya semakin maju, Google mendukung penuh Kotlin untuk pengembangan aplikasi Android.

Beberapa fitur dan kelebihan kotlin yaitu :

- **Open Source**

Kotlin didistribusikan di bawah Lisensi Apache, Versi 2.0. Kompiler (kompiler Kotlin), plugin IntelliJ IDEA, dan library-library java semuanya open source.

- **Easy To Learn**

Bahasa Pemrograman Kotlin mudah dipelajari, Hal ini dipengaruhi oleh Java, Scala, Groovy, C #, JavaScript dan Gosu. Belajar Kotlin mudah dilakukan jika Anda sudah familiar dengan salah satu bahasa pemrograman ini. Sangat mudah dipelajari jika Anda mengenal Java.

- **Safe**

SKotlin menjamin setiap *syntax* yang kita tulis secara proses kompilasi dapat mencegah kemungkinan terjadinya error, contohnya mampu mencegah terjadinya **NullPointerException** ketika kita menulis kode-kode menggunakan bahasa Java.

- **Consice and Expressive**

Proses kompilasi dan kerumitan pada saat coding menggunakan java akan berkurang ketika menggunakan kotlin.

- **Interoperable With Java and Android**

Kotlin dapat membaca kode lama atau *library* yang kita gunakan saat coding dengan bahasa Java dan begitupun sebaliknya.

2.6 Kolaborasi

Dalam pembuatan aplikasi YukAmal terdiri dari 4 tim yang masing-masing mengerjakan modul yang berbeda-beda sehingga kami kami menggunakan 2 tools untuk saling berkomunikasi yaitu Trello dan Github

2.6.1 Trello

Trello adalah sebuah aplikasi yang memudahkan kamu untuk bekerja sama dengan orang lain dalam sebuah tim saat melakukan proyek bersama. Trello akan membantumu menyusun rencana kerja dan menyelesaikannya secara bersama-sama.

Fitur-fitur pada trello:

1. *Boards* atau papan adalah sebuah penanda pekerjaan yang sedang dikerjakan saat ini
2. *List* adalah tempat membuat daftar langkah kerja atau tahapan yang harus dilakukan untuk menyelesaikan pekerjaan
3. *Card* berguna untuk memasukkan ide atau tugas yang harus diselesaikan agar proyek yang dijalankan dapat berhasil

Gambar di bawah ini, merupakan gambaran umum alur kerja pada trello.

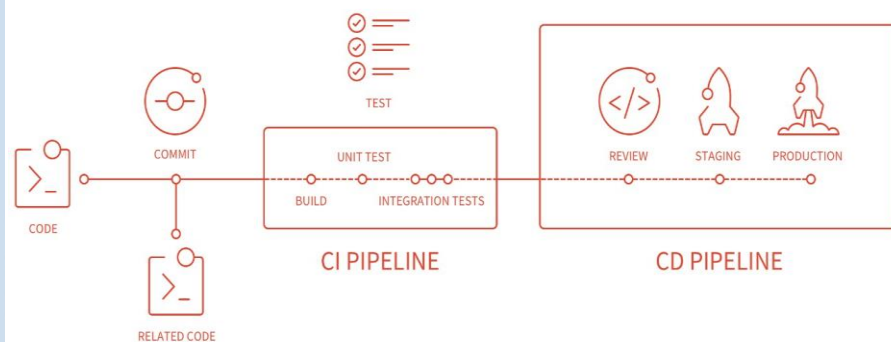


Gambar 3 Alur kerja trello

2.6.2 Github

Jika Git adalah jantung, maka Hub adalah jiwa GitHub. Sistem Hub yang ada pada GitHub berfungsi untuk mengubah baris perintah (*command line*), seperti Git, menjadi jaringan media sosial terbesar bagi para developer.

Selain berkontribusi dalam proyek tertentu, GitHub juga memungkinkan usernya untuk berkomunikasi dengan orang-orang yang memiliki kesamaan visi dan misi. Anda bahkan bisa *follow* mereka dan melihat proyek yang dilakukan atau bahkan mencari tahu siapa saja yang terhubung dengan mereka.



Gambar 4 Alur kerja Gitlab

Gitlab memberikan fitur lain diantaranya sebagai berikut:

1. Project, Kita dapat mengatur status project kita akan menggunakan fitur public atau private, Jika kita menggunakan fitur public maka project kita dapat dilihat, diubah, dan diunduh oleh siapa saja. Namun jika kita menggunakan fitur private maka project yang sedang kita kerjakan hanya dapat dilihat dan diubah oleh kita saja. Tetapi fitur private ini merupakan fitur berbayar tidak seperti fitur public yang diberikan secara gratis
2. Dokumentasi, termasuk memberikan file README secara otomatis dalam berbagai format Markdown

3. Issue Tracking, yaitu fitur pelacakan masalah yang terjadi dalam konten yang sedang dikembangkan. fitur ini dibekali dengan labels, milestone, assignees(penerimaan pengalihan hak), dan search engine
4. Commits history, dengan adanya fitur ini perubahan yang kita lakukan akan tetap ada rekam jejaknya. Contohnya jika kita telah melakukan project beta 1, kemudian kita melakukan perubahan kembali dengan beta 2 maka project beta 1 akan tetap ada
5. Notifikasi Email, Kita akan mendapatkan notifikasi melalui email tentang apa sedang terjadi dalam Github
6. Men-subscribe seseorang, kita dapat mengikuti aktivitas pengguna lain yang kita anggap sangat membantu kita dalam melakukan pengembangan konten yang kita kerjakan
7. Emoji, agar diskusi kita terasa lebih nyata kita dapat menggunakan emoji untuk berekspresi pada konten diskusi
8. Wiki, untuk merekam sejumlah catatan atau panduan terkait proyek yang bersangkutan

2.7 UAT

User Acceptance Testing (UAT) merupakan proses verifikasi bahwa solusi yang dibuat dalam sistem sudah sesuai untuk pengguna. Proses ini berbeda dengan pengujian sistem (memastikan software tidak crash dan sesuai dengan dokumen permintaan pengguna), melainkan memastikan bahwa solusi dalam sistem tersebut akan bekerja untuk pengguna (yaitu, tes bahwa pengguna menerima solusi di dalam sistem)

UAT umumnya dilakukan oleh klien atau pengguna akhir, biasanya tidak fokus pada identifikasi masalah sederhana seperti kesalahan ejaan, maupun di cacat showstopper, seperti crash perangkat lunak. Penguji dan pengembang mengidentifikasi dan memperbaiki masalah ini selama tahap awal pengujian fungsionalitas, pengujian saat integrasi dan pada tahap sistem testing

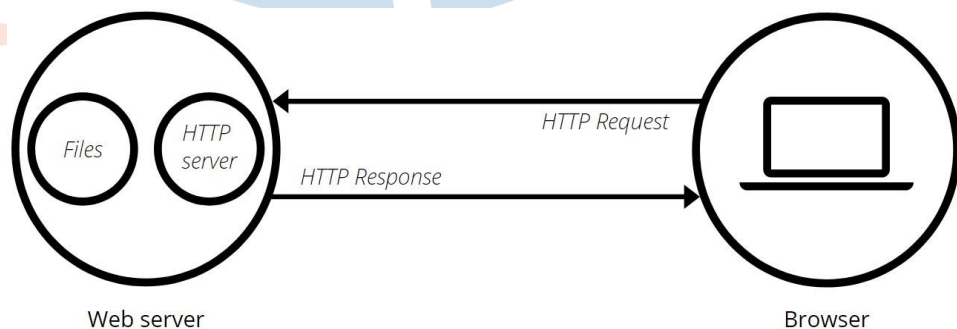
2.8 Web Service

Web service adalah suatu sistem perangkat lunak yang dirancang untuk mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. *Web service* digunakan sebagai suatu fasilitas yang disediakan oleh suatu web untuk menyediakan layanan (dalam bentuk informasi) kepada sistem lain, sehingga sistem lain dapat berinteraksi dengan sistem tersebut melalui layanan-layanan (*service*) yang disediakan oleh suatu sistem yang menyediakan *web service* (Predede, dkk: 2013). *Web service* menyediakan standar komunikasi di antara berbagai aplikasi *software* yang berbeda-beda, dan dapat berjalan di berbagai *platform* maupun *framework* (Hartono, dkk, 2013).

2.9 Web Server

Web server merupakan program komputer yang ditanamkan pada komputer yang diakses publik (*server*) sehingga komputer tersebut mampu menerima dan memberikan tanggapan pada pengaksesnya (*client*). Aksi dan reaksi antara *web server* dan *client* ini diatur oleh protokol yang telah disepakati bersama, dan umumnya protokol yang digunakan adalah HTTP (Madcoms, 2016:4).

Gambar di bawah ini menjelaskan cara kerja web server.



Gambar 5 Cara kerja web server

Pada saat mengakses halaman situs web, browser mengirimkan permintaan ke server yang kemudian diproses oleh web server. HTTP *request* dikirimkan ke web server. Sebelum memproses HTTP *request*,

web server juga melakukan pengecekan terhadap keamanan. Pada web server, HTTP *request* diproses dengan bantuan HTTP server. Kemudian web server mengirimkan HTTP *response* ke browser dan memprosesnya menjadi halaman situs web.

2.10 Table Penelitian

Tabel 1 Penelitian Terkait

No	Judul	Tahun	Hasil
1	Sistem Informasi Pengelolaan Masjid Ibaadurrahman Berbasis Web (Dodyk Kristanto)	2018	Membangun aplikasi informasi seputar masjid dan pelaporan keuangan menggunakan metode waterfall berbasis web
2	Perancangan dan Implementasi Aplikasi ikhtisar Kas Masjid Istiqomah Berbasis Desktop (Amarudin, Agung sofiandri)	2018	Aplikasi ikhtisar kas keuangan masjid menggunakan metode prototype berbasis dekstop
3	Pengembangan System Informasi Keuangan Masjid Babul Muttaqien Parang Tambung Kota Makassar (A.Mulia Bulaeng, Hasrul Bakri, Udin Sidik Sidin)	2019	Membangun Aplikasi laporan keuangan menggunakan metode prototype berbasis web
4	Sistem Informasi Pengelolaan Penyaluran dan Pengalokasian ZISWAF Berbasis Android (Yuni Astuti)	2018	Membangun aplikasi pengelolaan keuangan ZISWAF (Zakat, Infaq, Shodaqoh, Wakaf) berbasis android

Pada table 1 diatas tentang penelitian terkait belum ada yang membahas pengembangan sistem laporan keuangan dan donasi masjid berbasis android dengan menggunakan metode scrum



2.11 Posisi Penelitian

Tabel 2 Posisi Penelitian

Android	Laporan Keuangan	Metode Scrum
	<p><u>Dodyk Kristanto - 2018</u></p> <p>Sistem Informasi Pengelolaan Masjid Ibaadurrahman Berbasis Web</p>	
	<p><u>Amarudin, Agung Sofiandri - 2018</u></p> <p>Perancangan dan Implementasi Aplikasi ikhtisar Kas Masjid Istiqomah Berbasis Desktop</p>	
	<p><u>A.Mulia Bulaeng, Hasrul Bakri, Udin Sidik Sidin – 2019</u></p> <p>Pengembangan System Informasi Keuangan Masjid Babul Muttaqien Parang Tambung Kota Makassar</p>	
<p><u>Yuni Astuti - 2018</u></p> <p>Sistem Informasi Pengelolaan Penyaluran dan Pengalokasian ZISWAF Berbasis Android</p>		
<p><u>Muhammad Yahya Abdullah – 2019</u></p> <p>Perancangan Aplikasi Donasi dan Pelaporan Keuangan Masjid Berbasis Mobile Dengan Menggunakan Metode SCRUM</p>		