

## BAB II

### TINJAUAN PUSTAKA DAN LANDASAN TEORI

#### 2.1 Tinjauan Pustaka

Pemanfaatan *platform website* pada usaha *laundry* di Indonesia juga sudah banyak dilakukan untuk media promosi atau pemasaran, sebagai profil atau portofolio usaha, manajemen data, sebagai media transaksi *online*, ingin mendapatkan *feedback* pelanggan, dan masih banyak lagi. Seperti penelitian yang dilakukan oleh Desi Rizki Rahayu yang membuat **Sistem Informasi Laundry Berbasis Website** dengan bahasa pemrograman PHP, *CodeIgniter Framework*, dan menerapkan metode *Scrum* untuk merancang sebuah sistem informasi akuntansi pada MH Laundry dengan beberapa fitur seperti laporan laba rugi, neraca dan jurnal umum. Sehingga diharapkan pemilik MH Laundry dapat mengelola laporan keuangan dengan lebih mudah dan cepat sesuai dengan standar akuntansi, serta memaksimalkan perancangan sistem informasi *laundry* [4].

Penelitian serupa yang dilakukan Mohammad Fuad Farisi, Budi Praptono, dan Maria Dellarosawati Idawicaksati yang juga membuat **Perancangan Aplikasi Pencatatan Keuangan Studi Kasus Pada Amanah Laundry Bogor Menggunakan Agile Development Methods**, menyatakan jika ingin membuat sebuah aplikasi berbasis *website* dibutuhkan metode yang dapat menyelesaikan program yang tersusun rapi dengan cepat yaitu salah satunya dengan metode *Agile Development* menggunakan model *Scrum* [5]. Walaupun begitu, penelitian yang dilakukan oleh Doni Andriansyah dengan judul **Penerapan Model Waterfall Pada Sistem Informasi Layanan Jasa Laundry** menerapkan metode *Waterfall* pada sistem informasi berbasis *website* yang ia kembangkan, karena menurutnya dalam membantu menyelesaikan

pengembangan sistem, metode *Waterfall* dinilai sangat efektif untuk diterapkan, disertai dengan berbagai macam tahapan pengembangan yang terdapat pada metode *Waterfall*, dapat membantu pengembang dalam memahami kebutuhan pengguna dan juga sistem sehingga pengembangan sistem dapat diselesaikan dengan cepat dan efektif [6].

Selain itu, penelitian yang dilakukan Chayrul Arifin, Denny Sagita Rusdianto, dan Eriq Muhammad Adams Jonemaro yang berjudul **Pengembangan Sistem Pengelolaan *Laundry* berbasis *Website* (Studi Kasus: Senopati Jaya Card Laundry)** juga menerapkan metode *Waterfall* pada pengembangan sistem pengelolaan *laundry* mereka. Seluruh jenis pengujian yang diterapkan pada penelitian mereka menghasilkan nilai yang valid, karena peneliti sudah menyiapkan seluruh kebutuhan dengan jelas dan tidak akan diubah lagi, sehingga penerapan metode *Waterfall* dinilai lebih cocok dengan pengembangan sistem pengelolaan *laundry* mereka [7]. Namun berbeda dengan sebelumnya, pada penelitian yang dilakukan Andicha Eka Prastya dan Nurudin Santoso mereka membuat **Pengembangan Aplikasi Pengelolaan Usaha *Laundry* berbasis *Website*** yang menerapkan metode RAD (*Rapid Application Development*), karena menurut peneliti metode RAD secara umum mempunyai skala yang tidak terlalu besar dan memiliki durasi pengerjaan yang lebih pendek, dan RAD juga menerapkan model pengembangan dengan pendekatan yang difokuskan pada pengembangan sistem dengan waktu pengerjaan yang singkat dan cepat, sehingga mereka mendapatkan hasil yang memuaskan pada seluruh pengujian sistem baik secara *white-box* maupun *black-box* [8].

Selain untuk membuat *website* atau sistem informasi *laundry*, metode *Scrum* juga dapat diterapkan pada penelitian dengan studi kasus berbeda. Seperti penelitian yang dilakukan Novri Hadinata dan Muhammad Nasir dengan judul **Implementasi Metode *Scrum* Dalam Rancang Bangun Sistem Informasi Penjualan (Studi Kasus : Penjualan Sperpart Kendaraan)** yang ingin membangun sebuah sistem

informasi penjualan yang optimal dengan tujuan untuk memperkecil kesalahan yang ada. Sistem informasi penjualan ini dibuat menggunakan metode *Scrum*, yang bertujuan untuk membantu pengguna dalam melakukan transaksi penjualan maupun pembelian agar dapat merespon permintaan pelanggan dengan efektif dan memfasilitasi komunikasi antar divisi bagian gudang, admin dan pimpinan. Dengan demikian, sistem tersebut dapat membantu meningkatkan dan memperluas penjualan. Penerapan metode *Scrum* menunjukkan bahwa proyek pengembangan sistem sangat membutuhkan fleksibilitas yang tinggi untuk dapat memenuhi kepuasan pelanggan [9].

Pada penelitian dengan tema serupa yang dilakukan Aryanata Andipradana dan Kristoko Dwi Hartomo yang membuat **Rancang Bangun Aplikasi Penjualan Online Berbasis Web Menggunakan Metode Scrum** juga menerapkan metode *Scrum* untuk membuat aplikasi penjualan *online* berbasis *web* karena mereka percaya bahwa metode pengembangan model *Scrum* dapat membantu pemangku kepentingan dalam menyesuaikan perubahan pada kebutuhan sistem. Dengan sumber daya tim yang terbatas, *Scrum* dapat menyelesaikan aplikasi dengan mengandalkan kerjasama antar anggota tim untuk mencapai tujuan dari setiap fitur yang dibutuhkan pengguna[10].

Selain itu, penelitian lain yang dilakukan Dwi Wahyu Adi Nugroho dan Augie David Manuputty untuk membuat **Rancang Bangun Sistem Informasi Gelanggang Olahraga Berbasis Web dengan Metode Scrum** menerapkan metode *Scrum* untuk memudahkan peneliti melakukan *controlling* dalam pembuatan sistem setiap harinya karena selalu akan diadakan *meeting* secara berkala pada saat *daily scrum* karena penyelesaian sebuah sistem dilakukan dengan kerjasama antar tim, sehingga tercapai tujuan suatu sistem yang selaras. Maka dari itu, perancangan Sistem Informasi Gelanggang Olahraga (SIGORA) menjadi lebih optimal dan efektif ketika menggunakan metode *Agile Scrum* [11]. Namun penelitian yang dilakukan Rizki Wahyudi, Tri Astuti, dan Aep

Saprul Mujahid dengan judul **Implementasi *Extreme Programming* Pada Sistem Reservasi Tiket Travel Berbasis Android Dan Website** justru menggunakan metode *Extreme Programming* dalam pengembangan sistem reservasi tiket yang mereka bangun karena menurut peneliti metode *Extreme Programming* dinilai cocok untuk mengembangkan sistem yang menggunakan desain dan program yang berbasis *object oriented* seperti *Unified Modeling Language* dan bahasa pemrograman *Java*. Selain itu, metode *Extreme Programming* juga dapat meningkatkan komunikasi dengan pengguna dan memungkinkan proses pengembangan sistem dilakukan hanya dengan tim kecil saja. Sehingga melalui pengujian *Alpha Testing* menggunakan pendekatan *black-box* untuk memverifikasi kinerja fungsi sistem pada aplikasi *android* menghasilkan semua fitur pada aplikasi *android* yang dapat beroperasi sesuai dengan yang diinginkan. Setelah melaksanakan *Beta Testing* terhadap aplikasi *android* yang melibatkan konsumen PT Qyta Trans, menggunakan pendekatan kuesioner, diperoleh data bahwa 79,15% responden menganggap aplikasi tersebut pantas untuk diimplementasikan [12].

Penelitian lain yang dilakukan Rosa Delima, Halim Budi Santosa, dan Joko Purwadi dalam membuat ***Development of Dutatani Website Using Rapid Application Development*** menerapkan metode RAD (*Rapid Application Development*) yang mengkombinasikan pengembangan berulang (*iterative*) dengan *system prototyping* pada pengembangan *website* Dutatani, dikarenakan penerapan metode RAD dapat mempersingkat waktu pengembangan sistem sekaligus dapat memantau kecepatan pengerjaan dari setiap anggota tim, sehingga proses pengembangan *website* dapat dipantau dan diselesaikan dengan tepat waktu [13].

Sehingga dapat disimpulkan bahwa penerapan metode *Scrum* pada penelitian ini sangat cocok dikarenakan metode *Scrum* dapat diterapkan pada sistem yang mengandalkan *feedback* dari *client* dan memiliki

perubahan ditengah pengembangan, namun tidak akan berdampak buruk pada proses pengembangan sistem tersebut. Karena metode *Scrum* merupakan metode pengembangan yang fleksibel, sehingga tingkat keberhasilan pengembangan sistem akan jauh lebih tinggi dan sistem yang dikembangkan menjadi lebih optimal.

Tabel 2.1 Perbandingan dengan penelitian sebelumnya

No	Peneliti	Judul Penelitian	Metode	Deskripsi	Hasil Penelitian
1	Desi Rizki Rahayu	Sistem Informasi Laundry Berbasis Website	<i>Scrum</i>	Pada penelitian ini, PHP dan <i>CodeIgniter</i> digunakan untuk mengembangkan sistem informasi akuntansi laundry berbasis website. Tujuan utama dari pengembangan sistem ini adalah untuk menyederhanakan proses pencatatan keuangan yang terkomputerisasi serta memudahkan pengelola MH laundry dalam mencari data pemasukan dan pengeluaran melalui laporan jurnal umum, neraca dan laporan laba rugi.	Hasil pengujian <i>black box testing</i> yang telah dilakukan mengindikasikan bahwa fitur-fitur dalam sistem yang dibangun berjalan dengan normal. Sistem informasi laundry ini menyediakan berbagai menu termasuk profil, data informasi laundry, data pemasukan, data pengeluaran, data supplier, data akun, data pengguna, dan laporan keuangan. Dengan adanya kriteria tertentu dalam sistem, pengguna dapat dengan mudah memasukkan data untuk membuat laporan keuangan.
2	Mohammad Fuad Farisi, Budi Praptono, dan Maria Dellarosawati Idawicakwati	Perancangan Aplikasi Pencatatan Keuangan Studi Kasus Pada Amanah Laundry Bogor Menggunakan <i>Agile Development</i>	<i>Scrum</i>	Amanah Laundry saat ini masih menggunakan metode manual dalam pencatatan keuangan, yaitu melakukan pencatatan dengan buku keuangan. Sehingga metode tersebut dianggap kurang efisien dalam pengelolaan keuangan. Oleh karena itu, tujuan penelitian ini	Dari hasil penelitian dapat disimpulkan bahwa metode yang efektif dalam mengembangkan aplikasi berbasis website adalah melalui penggunaan metode <i>agile development</i> dengan model <i>scrum</i> . Metode tersebut

No	Peneliti	Judul Penelitian	Metode	Deskripsi	Hasil Penelitian
		<i>t Methods</i>		adalah untuk membantu Amanah Laundry dalam mengadopsi aplikasi berbasis <i>website</i> untuk membuat pencatatan dan laporan keuangan dengan menggunakan aplikasi berbasis <i>website</i> dengan pendekatan <i>agile development</i> dan model <i>scrum</i> .	memungkinkan program-program yang telah terstruktur dapat diselesaikan dengan rapi dan cepat. Selain itu, peran pengguna dalam memberikan <i>feedback</i> terhadap aplikasi yang telah dikembangkan juga sangat penting. Sistem pencatatan dan laporan keuangan berbasis <i>website</i> ini memberikan bantuan yang signifikan pada bagian keuangan Amanah Laundry. Melalui sistem ini, mereka dapat melakukan pencatatan keuangan, membuat laporan laba-rugi dan laporan neraca, serta dapat memantau proses keuangan Amanah Laundry melalui aplikasi ini.
3	Doni Andriansyah	Penerapan Model <i>Waterfall</i> Pada Sistem Informasi Layanan Jasa Laundry	<i>Waterfall</i>	D'cuci in <i>laundry</i> adalah sebuah usaha yang bergerak di bidang jasa pencucian/ <i>laundry</i> . Namun, saat ini seluruh kegiatan pada usaha tersebut belum terkomputerisasi. Sehingga peneliti ingin mengembangkan sebuah <i>website</i> yang dapat memberikan informasi mengenai paket layanan <i>laundry</i> yang tersedia dan juga memudahkan pelanggan dalam melakukan transaksi. Melalui <i>website</i> ini,	Metode <i>Waterfall</i> terbukti sangat efektif dalam membantu mendukung pengembangan sistem. Tahapan-tahapan pengembangan yang terstruktur sangat membantu pengembang untuk memahami kebutuhan sistem dan kebutuhan pengguna, sehingga proses pembangunan sistem dapat

No	Peneliti	Judul Penelitian	Metode	Deskripsi	Hasil Penelitian
				<p>pelanggan dapat melakukan transaksi secara <i>online</i> dan menggunakan layanan antar jemput. Perancangan sistem ini akan menggunakan metode <i>Waterfall</i> yang dianggap efektif dalam menyelesaikan pembangunan sistem sesuai dengan waktu yang ditentukan.</p>	<p>diselesaikan secara cepat dan akurat.</p>
4	<p>Chayrul Arifin, Denny Sagita Rusdianto, dan Eriq Muhammad Adams Jonemaro</p>	<p>Pengembangan Sistem Pengelolaan Laundry berbasis Website (Studi Kasus: Senopati Jaya Card Laundry)</p>	<p><i>Waterfall</i></p>	<p>Peneliti merancang sebuah sistem pengelolaan laundry berbasis web untuk meningkatkan pelayanan dan efisiensi pada Senopati Jaya Card Laundry. Sistem yang dikembangkan akan memberikan kemudahan bagi pengguna dalam melakukan pencatatan transaksi, melacak riwayat perawatan mesin cuci, mendapatkan informasi tentang cucian, dan memantau stok barang. Proses pengembangan sistem pengelolaan laundry ini menggunakan metode pengembangan <i>waterfall</i>. Sistem tersebut dibangun menggunakan pola MVC (<i>Model View Controller</i>) dengan menggunakan <i>Laravel</i> sebagai <i>framework</i>-nya. Untuk database menggunakan <i>HeidiSQL</i> dalam pengembangannya. Pengujian yang dilakukan meliputi pengujian unit, integrasi, validasi, dan juga pengujian</p>	<p>Semua kebutuhan fungsional dan nonfungsional telah berjalan dengan baik sesuai dengan definisi yang telah ditetapkan pada rekayasa kebutuhan. Hal ini dibuktikan melalui pengujian sistem yang meliputi pengujian unit, pengujian integrasi, pengujian validasi, dan pengujian kompatibilitas. Dalam pengujian unit dan integrasi dari tiga metode, metode <i>white-box testing</i> dengan menggunakan teknik basis <i>path testing</i> telah digunakan. Hasil pengujian dari ketiga metode tersebut menunjukkan bahwa semua nilai yang dihasilkan adalah valid. Pengujian validasi pada 29 kebutuhan fungsional dilakukan menggunakan metode <i>black box testing</i>, dan hasilnya juga valid. Pengujian</p>

No	Peneliti	Judul Penelitian	Metode	Deskripsi	Hasil Penelitian
				kompatibilitas.	kompatibilitas menggunakan aplikasi <i>SortSite</i> dan juga menghasilkan nilai yang valid.
5	Andicha Eka Prastya dan Nurudin Santoso	Pengembangan Aplikasi Pengelolaan Usaha <i>Laundry</i> berbasis <i>Website</i>	RAD ( <i>Rapid Application Development</i> )	Untuk mendukung operasional bisnis Ameera <i>Laundry</i> , diperlukan sebuah aplikasi yang dapat mengelola semua aspek bisnis agar mempermudah proses pengerjaannya. Aplikasi yang dikembangkan berbasis <i>website</i> menggunakan metode <i>Rapid Application Development</i> (RAD) karena metode tersebut memungkinkan pengembangan aplikasi yang cepat dan sesuai dengan kebutuhan yang diperlukan. Sistem ini memiliki 21 kebutuhan fungsional dan melibatkan 3 aktor dalam sistem. Pengembangan aplikasi dilakukan dengan menggunakan MERN <i>stack</i> , yang tersusun atas <i>MongoDB</i> , <i>Express JS</i> , <i>React JS</i> , dan <i>Node JS</i> , serta menerapkan model <i>design pattern</i> MVC ( <i>Model, View, Controller</i> ). Pengujian yang dilakukan mencakup pengujian unit, integrasi, dan sistem dengan menggunakan metode <i>black box</i> dan <i>white box</i> .	Secara umum, proyek atau perangkat lunak yang dikembangkan dengan metode RAD memiliki skala yang relatif kecil dan waktu pengerjaan yang singkat. Pendekatan RAD digunakan untuk pengembangan sistem yang berfokus pada kecepatan dan waktu yang terbatas. Ruang kerja dibagi menjadi beberapa bagian untuk memastikan fokus kerja dan pencapaian sesuai dengan tenggat waktu yang ditentukan. Implementasi proyek berhasil dilakukan dengan menerapkan metode RAD dengan menggunakan konsep MERN <i>stack</i> dan model MVC sesuai dengan peancangan yang telah ditentukan. Hasil pengujian menunjukkan bahwa semua kebutuhan sistem telah memperoleh hasil yang valid.
6	Novri	Implementas	<i>Scrum</i>	Proses pengolahan data	Implementasi

No	Peneliti	Judul Penelitian	Metode	Deskripsi	Hasil Penelitian
	Hadinata dan Muhammad Nasir	i Metode <i>Scrum</i> Dalam Rancang Bangun Sistem Informasi Penjualan (Studi Kasus : Penjualan Sperpart Kendaraan)		penjualan dan persediaan barang saat ini dicatat secara manual dalam bentuk lembaran kertas, dan data tersebut disimpan dalam biling arsip. Namun, perusahaan mengalami kesulitan dalam menentukan stok barang digudang, yang sering mengakibatkan penumpukan atau kekurangan <i>sparepart</i> mobil karena kurangnya stabilitas penyimpanan. Sehingga peneliti ingin mengembangkan sebuah sistem informasi penjualan yang dapat mengoptimalkan proses penjualan dan mengurangi kesalahan dengan menerapkan metode <i>scrum</i> . Metode <i>Scrum</i> mengandalkan <i>sprint</i> dan beban kerja yang didorong oleh <i>Prioritized Product Backlog</i> . <i>Prioritized Product Backlog</i> berisi fitur terbaru, perbaikan <i>bug</i> , dan hal lain yang berkontribusi pada produk akhir. Metode <i>Scrum</i> melibatkan pembagian waktu dalam bentuk <i>sprint</i> , dan memiliki tiga artefak, seperti <i>Prioritized Product Backlog</i> , <i>Sprint Backlog</i> , dan <i>Deliverable</i> .	metodologi <i>Scrum</i> telah membuktikan bahwa dalam pengembangan sistem, fleksibilitas yang tinggi merupakan hal yang penting untuk memenuhi kepuasan pelanggan. Hal ini terutama terjadi pada pengembangan aplikasi berbasis web, yang menjadi salah satu strategi untuk mencapai keunggulan berkompetisi. Sistem Informasi penjualan menggunakan model <i>scrum</i> yang dirancang dan dikembangkan dengan tujuan mempermudah pengguna dalam melakukan transaksi dalam penjualan dan pembelian. Dengan demikian, sistem ini dapat secara efektif menanggapi permintaan pelanggan dan memfasilitasi komunikasi antara divisi seperti bagian gudang, admin dan pimpinan. Dengan adanya sistem ini, diharapkan dapat membantu memperluas dan meningkatkan penjualan.
7	Aryanata Andipradana dan	Rancang Bangun Aplikasi	<i>Scrum</i>	Pembangunan aplikasi penjualan <i>online</i> berbasis web menjadi	Penerapan metode pengembangan sistem <i>scrum</i>

No	Peneliti	Judul Penelitian	Metode	Deskripsi	Hasil Penelitian
	Kristoko Dwi Hartomo	Penjualan <i>Online</i> Berbasis Web Menggunakan Metode <i>Scrum</i>		salah satu solusi untuk memberikan <i>platform</i> dan memperluas pemasaran produk UMKM melalui media digital. Dibangunnya aplikasi penjualan ini bertujuan untuk memberikan wadah bagi pemilik UMKM di kota Salatiga untuk menjalankan aktivitas penjualan secara <i>online</i> dan mempermudah berbagai proses bisnis penjualan. Dengan adanya aplikasi ini, fokus pemasaran dapat lebih terarah, sehingga diharapkan dapat meningkatkan angka penjualan. Metode yang akan digunakan dalam pengembangan aplikasi ini adalah <i>Agile software development</i> dengan menggunakan model <i>Scrum</i> . Selain itu, aplikasi ini akan dikembangkan dengan bahasa pemrograman PHP menggunakan kerangka kerja <i>CodeIgniter</i> , dan <i>database MySQL</i> .	membantu para pemangku kepentingan mengakomodasi perubahan kebutuhan sistem. Meskipun tim memiliki sumber daya terbatas, <i>scrum</i> mampu menyelesaikan pengembangan aplikasi dengan mengandalkan koordinasi yang baik antar anggota tim untuk mencapai tujuan dari setiap fitur yang diperlukan oleh pengguna. Penggunaan aplikasi berbasis web juga memberikan kemudahan bagi pengguna dalam mengoperasikan aplikasi tersebut.
8	Dwi Wahyu Adi Nugroho dan Augie David Manuputt y	Rancang Bangun Sistem Informasi Gelanggang Olahraga Berbasis Web dengan Metode <i>Scrum</i>	<i>Scrum</i>	Hingga saat ini, proses bisnis di Dinas Pemuda dan Olahraga (DISPORA) Salatiga seperti peminjaman gedung masih dilakukan secara manual. Hal ini menyebabkan beberapa kendala dalam penjadwalan pinjaman dan pembayaran. Sehingga peneliti membuat Sistem	Dalam mengimplementasikan metode <i>agile scrum</i> , kita dapat menyelesaikan pengembangan suatu produk sistem dengan koordinasi antar tim untuk mencapai tujuan produk yang sejalan. Dengan menggunakan metode tersebut peneliti juga dapat

No	Peneliti	Judul Penelitian	Metode	Deskripsi	Hasil Penelitian
				<p>Informasi Pusat Olahraga (SIGORA) berbasis web yang bertujuan untuk menghasilkan sebuah sistem informasi peminjaman gedung atau SIGORA yang menyediakan informasi gedung, jadwal peminjaman, transaksi pembayaran dengan metode transfer dan laporan bulanan secara otomatis, mempermudah dalam menjalankan proses bisnis pada DISPORA sehingga menjadi efisien dan menerapkan metode <i>scrum</i> untuk pengembangan sistem.</p>	<p>dengan mudah melakukan <i>controlling</i> dalam pembuatan sistem pada tiap harinya karena selalu akan mengadakan rapat atau <i>meeting</i> dengan rutin pada tahap <i>daily scrum</i>. Sehingga dapat disimpulkan bahwa dalam perancangan Sistem Informasi Gelanggang Olahraga (SIGORA) menggunakan metode <i>agile scrum</i> menjadi lebih optimal dan efektif.</p>
9	Rizki Wahyudi, Tri Astuti, dan Aep Saprul Mujahid	Implementasi <i>Extreme Programming</i> Pada Sistem Reservasi Tiket Travel Berbasis Android Dan <i>Website</i>	<i>Extreme Programming</i>	<p>Penelitian ini berfokus pada studi kasus PT. Qyta Trans Group, sebuah perusahaan rental mobil dan <i>travel</i> di Wilayah Jawa. Sistem yang dikembangkan memungkinkan konsumen untuk memilih jenis mobil yang ingin disewa dan posisi tempat duduk. Dalam pengembangan sistem ini, digunakan Metode <i>Extreme Programming</i> yang dinilai sesuai dengan desain dan pemrograman yang berbasis objek, seperti <i>Unified Modeling Language</i> dan bahasa pemrograman <i>Java</i>. Metode <i>Extreme Programming</i> juga memungkinkan</p>	<p>PT. Qyta Trans Group telah mengembangkan sistem reservasi berbasis <i>Android</i> dan <i>Website</i> dengan menggunakan Metode <i>Extreme Programming</i> sebagai solusi alternatif untuk reservasi tiket <i>travel</i> dan sewa mobil. Hasil pengujian <i>Alpha Testing</i> dengan metode <i>Black-box</i> menunjukkan bahwa semua fungsi yang ada pada aplikasi <i>Android</i> berjalan sesuai harapan. Selanjutnya, hasil pengujian <i>Beta Testing</i> terhadap aplikasi <i>Android</i> yang melibatkan konsumen PT. Qyta Trans menggunakan</p>

No	Peneliti	Judul Penelitian	Metode	Deskripsi	Hasil Penelitian
				komunikasi yang optimal dengan pengguna dan pengembangan sistem dengan tim kecil. Sistem yang dibangun memiliki basis Android dan <i>Website</i> . Pengujian dilakukan melalui dua metode, yaitu <i>Alpha Testing</i> dan <i>Beta Testing</i> .	metode kuesioner menunjukkan bahwa 79,15% dari responden setuju bahwa aplikasi ini sudah layak untuk diimplementasikan.
10	Rosa Delima, Halim Budi Santosa, dan Joko Purwadi	<i>Development of Dutatani Website Using Rapid Application Development</i>	RAD (Rapid Application Development)	IAIS merupakan sistem informasi pertanian yang memiliki beberapa subsistem yang terhubung dan terkait satu sama lain, dan juga memiliki basis data yang terintegrasi. Salah satu subsistem yang terintegrasi dengan IAIS adalah Dutatani <i>website</i> . <i>Website</i> Dutatani merupakan sebuah <i>platform website</i> bagian dari <i>Integrated Agriculture Information System</i> (IAIS) yang berfungsi sebagai halaman utama untuk pengelolaan data pengguna, mengelola konten, dan memuat tautan <i>website</i> agar terhubung dengan <i>website</i> lain yang dimiliki IAIS. Pengembangan <i>website</i> Dutatani menggunakan metode RAD yang mengkombinasikan pengembangan berulang ( <i>iterative</i> ) dengan <i>system prototyping</i> .	Berdasarkan penelitian yang dilakukan, peneliti berhasil mengembangkan <i>website</i> Dutatani dengan menerapkan metode RAD. Penerapan metode tersebut dilakukan dengan menggabungkan teknik pengembangan <i>iterative</i> dan <i>system prototyping</i> . Penerapan metode RAD dapat mempersingkat waktu pengembangan sistem sekaligus dapat memantau kecepatan pengerjaan dari setiap anggota tim, sehingga proses pengembangan <i>website</i> dapat dipantau dan diselesaikan dengan tepat waktu.

## 2.2 Landasan Teori

### 2.2.1 Website

*Website* adalah sekumpulan halaman web atau *file digital* yang disusun dengan HTML (*HyperText Markup Language*) dan disimpan di *Web Server* yang terhubung ke internet sepanjang waktu, sehingga dapat diakses secara publik oleh semua orang di dunia. Sebuah *website* dapat berisi konten dalam bentuk dokumen, gambar, video, atau aset digital lain, serta ditautkan dengan *hyperlink* dan *hypertext* [14].

### 2.2.2 Arsitektur Sistem

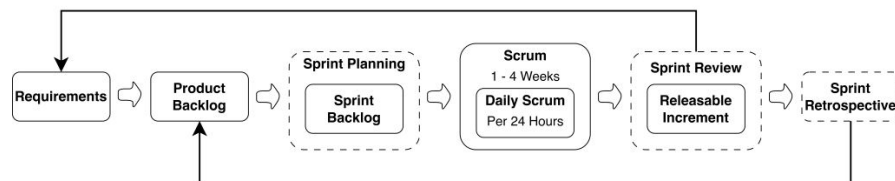
Arsitektur sistem adalah model konseptual yang menggambarkan struktur dan perilaku beberapa komponen dan subsistem seperti beberapa aplikasi perangkat lunak, perangkat jaringan, perangkat keras, dan bahkan mesin sistem lainnya seperti *Architecture Description Language (ADL)* yang membantu menggambarkan seluruh arsitektur sistem. Arsitektur sistem dapat dikategorikan secara luas menjadi organisasi arsitektur terpusat dan terdesentralisasi [15].

### 2.2.3 Backend

*Backend* merupakan komponen dari sebuah *website* yang tidak terlihat oleh pengguna, namun berperan penting dalam menjalankan fungsi-fungsi pada sebuah *website*. Bagian ini umumnya terhubung dengan *database* dan juga *Application Programming Interface (API)*. Setiap *backend* dari sebuah situs web terdiri dari tiga elemen utama yaitu *server*, *database*, dan aplikasi. Seorang *developer backend* bertanggung jawab dalam menulis kode yang memungkinkan interaksi dan kerjasama antara ketiga komponen tersebut, sehingga dapat melaksanakan fungsi-fungsi dan menyampaikan informasi yang dapat digunakan atau dilihat oleh pengguna [16].

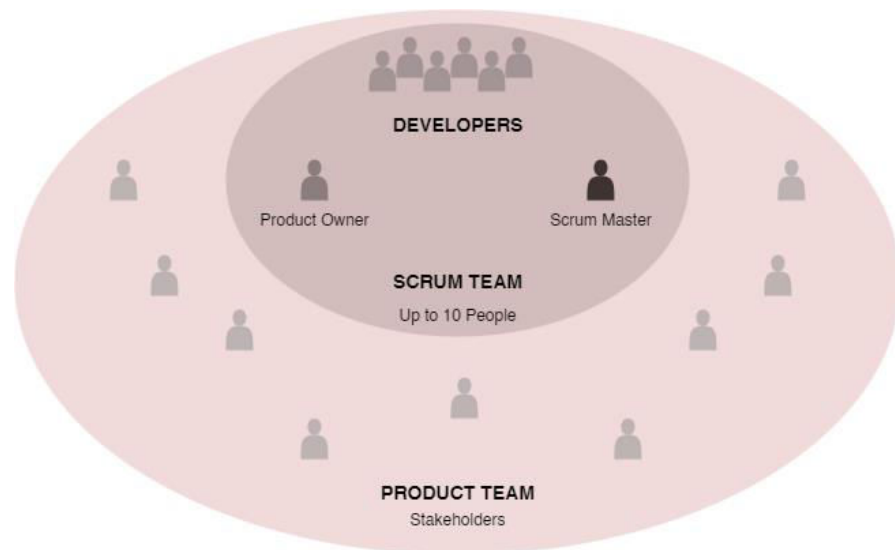
## 2.2.4 Metode Scrum

*Scrum* adalah *framework* atau kerangka kerja untuk mengatur pekerjaan, sedangkan *Agile* adalah kata sifat yang digunakan untuk mendeskripsikan berbagai praktik yang selaras dengan nilai-nilai *Agile Manifesto* dan 12 Prinsip *Agile*. *Scrum* dan *agile* tidak identik atau dapat dipertukarkan, tetapi sering terlihat digunakan secara bergantian di banyak sumber tertulis. *Scrum* adalah kerangka paparan berdasarkan empirisme, artinya orang yang menerapkan kerangka *scrum* mendapatkan pengetahuan dari pengalaman kehidupan nyata dan mengambil keputusan berdasarkan pengalaman tersebut. Baik proyek kecil ataupun besar, jika perlu menyelesaikan sesuatu, *scrum* menyediakan struktur untuk peningkatan efisiensi dan hasil yang lebih efektif. Proses *scrum* sederhana dan melingkar, dengan elemen konstan dan transparan inspeksi dan adaptasi [17]. Proses dan tahapan *scrum* dapat dilihat seperti pada Gambar 2.1



Gambar 2.1 Overview proses dan tahapan *scrum*

Pertama, susunlah daftar tugas yang biasa disebut *product backlog* dari proyek yang ingin diselesaikan. Kemudian tentukan masa atau periode tetap pengerjaan proyek tersebut yang disebut *sprint* dan tentukan bersama tim *scrum* yang telah dibentuk untuk menyusun waktu *sprint* dengan kesepakatan bersama. Lalu susun *backlog* yang ingin diselesaikan pada setiap *sprint* dan lakukan proses pengembangan sesuai dengan *sprint* yang sudah disepakati sebelumnya. Dan lakukan *review* pada akhir dari setiap *sprint* untuk memantau progress dan kinerja dari tim *scrum* [17]. Struktur dari sebuah *scrum team* dapat dilihat pada Gambar 2.2.



Gambar 2.2 Struktur sebuah *scrum team*

Dengan menggunakan proses *scrum*, anda dapat dengan cepat beradaptasi terhadap perubahan kekuatan pasar, kendala teknologi, peraturan, inovasi baru, preferensi keluarga, dan berbagai faktor lainnya yang dapat anda bayangkan. Kuncinya adalah proses kerja yang berkelanjutan pada item dengan prioritas tertinggi untuk diselesaikan. *Scrum* adalah kerangka kerja yang memberikan kejelasan tanggung jawab melalui peran, visibilitas melalui artefak, dan peluang untuk inspeksi dan adaptasi melalui peristiwa [17].

### 2.2.5 Database

*Database* adalah kumpulan *file* yang saling terkait dan berinteraksi melalui relasi yang ditunjukkan dengan kunci pada setiap *file*. Satu *database* merepresentasikan kumpulan data yang digunakan dalam suatu perusahaan atau instansi. Pengolahan *Database* merupakan suatu metode yang dilakukan terhadap beberapa *file* yang ada dalam suatu instansi, dimana *file* tersebut dapat diorganisir, diurutkan, diambil kapan pun diperlukan, dan ditampilkan dalam bentuk laporan agar data tersebut dapat disajikan dalam bentuk informasi yang terstruktur [18].

### 2.2.6 *PostgreSQL*

*PostgreSQL* adalah sistem basis data objek-relasional sumber terbuka yang kuat. Sistem ini menggunakan dan memperluas bahasa SQL serta dilengkapi dengan berbagai fitur untuk menyimpan dan menangani beban kerja data yang sangat kompleks dengan aman. *PostgreSQL* pertama kali dikembangkan pada tahun 1986 sebagai bagian dari proyek *POSTGRES* di *University of California* di Berkeley. Selama lebih dari 35 tahun, *PostgreSQL* telah mengalami pengembangan aktif pada *platform* ini. *PostgreSQL* telah membangun reputasi yang solid dalam hal arsitektur, keandalan, integritas data, rangkaian fitur yang kuat, kemampuan ekstensibilitas, dan dukungan yang kuat dari komunitas *open source*. Keunggulan ini memungkinkan *PostgreSQL* untuk secara konsisten menyediakan solusi yang inovatif dan performa yang tinggi. *PostgreSQL* dapat dijalankan di berbagai sistem operasi utama, dan dilengkapi dengan berbagai *add-on* yang kuat seperti *ekstender database geospasial PostGIS* yang populer. Oleh karena itu, *PostgreSQL* telah menjadi pilihan *database* relasional *open source* yang banyak digunakan oleh berbagai individu dan organisasi [19].

### 2.2.7 *Express JS*

*Express* adalah sebuah *framework* aplikasi web minimal dan fleksibel untuk *Node.js* yang menyediakan serangkaian fitur kuat untuk aplikasi web dan seluler. Dengan segudang metode utilitas HTTP dan *middleware* yang tersedia, *framework* ini memungkinkan pembuatan API yang kuat menjadi efisien dan mudah. *Express* memberikan lapisan tipis fitur dasar aplikasi web, tanpa mengaburkan fitur unggulan *Node.js* yang sudah biasa dikenal dan disukai banyak orang [20].

### 2.2.8 Node JS

*Node.js* juga dikenal sebagai *Node*, adalah *environment Javascript* yang berjalan secara *server side*. *Environment* ini didasarkan pada *runtime Google* dengan mengimplementasikan mesin yang diberi nama “V8”. Implementasi V8 dan *Node* sebagian besar menggunakan bahasa C dan C++, dengan fokus pada kinerja yang tinggi dan penggunaan memori yang rendah. Namun, V8 mendukung terutama *Javascript* di *browser* (terutama, *Google Chrome*), karena *Node* dirancang untuk mendukung proses *server* yang berjalan dalam jangka waktu yang lama. Perbedaan *Node* dari banyak *environment modern* lainnya adalah bahwa proses *Node* tidak bergantung pada *multithreading* untuk menangani eksekusi yang bersamaan. Pendekatan ini didasarkan pada model peristiwa I/O *asynchronous* [21]. *Node.js* sebagai *runtime Javascript* yang mengadopsi pendekatan peristiwa *asynchronous*, diciptakan dengan tujuan untuk membangun aplikasi jaringan yang dapat diskalakan [22].

### 2.2.9 REST

REST adalah seperangkat prinsip arsitektur yang memungkinkan transmisi data melalui antarmuka yang terstandarisasi seperti HTTP. Konsep REST mirip dengan aplikasi *website* biasa, dimana *client* dapat mengirim permintaan ke *server* melalui protokol HTTP dan menerima respons dari *server*. Roy Fielding, yang merupakan salah satu pendiri dari *Apache HTTP Server*, adalah orang yang mengembangkan konsep REST. Dalam arsitektur REST, *server* REST menyediakan sumber daya (*resources*) atau data, sedangkan *client* REST mengakses dan menampilkan sumber daya tersebut untuk penggunaan selanjutnya. Setiap sumber daya diidentifikasi oleh URI (*Universal Resource*

*Identifiers*) atau *global ID*. Representasi sumber daya dapat berupa format teks, JSON atau XML [23].

### 2.2.10 *Design Pattern* Perangkat Lunak

*Design Pattern* digunakan untuk mewakili beberapa praktik terbaik yang diadaptasi oleh *developer* perangkat lunak berorientasi objek yang berpengalaman. *Design Pattern* memberikan penamaan dan penjelasan secara sistematis mengenai pola desain umum yang membahas mengenai desain yang sering terjadi secara berulang dalam sistem yang berorientasi objek. Ini menggambarkan masalah, solusi, kapan menerapkan solusi, dan konsekuensinya, serta memberikan petunjuk dan contoh implementasi [24]. Salah satu model *design pattern* yang biasa digunakan adalah MVC. *Design Pattern* model MVC (*Model View Controller*) menetapkan bahwa aplikasi terdiri dari model data, informasi presentasi, dan informasi kontrol. *Pattern* atau polanya mengharuskan masing-masing dipisahkan menjadi objek yang berbeda. Pada bagian model hanya berisi data aplikasi murni, yang berisi penjelasan cara menyajikan data kepada pengguna. *View* menampilkan data model kepada pengguna. *View* mengetahui cara mengakses data model, tetapi tidak mengetahui apa arti data ini atau apa yang dapat dilakukan pengguna untuk memanipulasinya. Dan *controller* ada di antara *view* dan model, yang mendengarkan peristiwa (*event*) yang dipicu oleh *view* (atau sumber eksternal lainnya) dan mengeksekusi reaksi yang sesuai untuk peristiwa tersebut. Dalam kebanyakan kasus, reaksinya adalah memanggil metode pada model, karena *view* dan model terhubung melalui mekanisme notifikasi, hasil tindakan ini kemudian secara otomatis tercermin dalam *view* [25].

### 2.2.11 *Integration Testing*

Tujuan dari *integration testing* adalah untuk membangun sistem yang cukup stabil yang dapat menahan kerasnya pengujian tingkat sistem. Pengujian tingkat sistem mencakup spektrum pengujian yang luas, seperti pengujian fungsionalitas, pengujian keamanan, pengujian ketahanan, beban pengujian, pengujian stabilitas, pengujian stres, pengujian kinerja, dan pengujian keandalan. *Integration testing* pada sistem dilakukan secara bersama-sama oleh *developer* perangkat lunak dan *integration test engineer* dalam fase peningkatan kompleksitas untuk efisiensi dan efektivitas yang lebih baik. Pada fase pertama integritas antarmuka dan validitas fungsional dalam suatu sistem diuji. Pada fase kedua *end-to-end* dan dilakukan uji berpasangan. Terakhir, pada fase ketiga, *stress and endurance test* dilakukan. Setiap fase integrasi sistem diidentifikasi dalam rencana SIT (*System Integration Testing*) menggambarkan kategori fungsionalitas yang luas dalam struktur perangkat lunak, dan itu bisa terkait dengan *domain* tertentu dari struktur perangkat lunak sistem. Pengujian integrasi dikatakan selesai ketika sistem terintegrasi penuh bersama-sama, semua kasus uji telah dieksekusi, semua cacat parah dan sedang yang ditemukan telah diperbaiki, dan sistem diuji ulang [26].

### 2.2.12 *Black Box Testing*

*Black box testing* didasarkan pada persyaratan, karena itu teknik *black box* juga disebut teknik perilaku atau fungsional. Strategi pengujian *black box* mengabaikan struktur internal objek yang diuji. Dengan demikian, strategi *black box* tidak mempertimbangkan bagaimana caranya objek uji telah diimplementasikan, tetapi apa persyaratannya [27].