

Pengembangan Frontend Website Pada Platform Survey Online Menggunakan Agile Scrum

Sugiyanto*, Citra Fauzan Alqodri

Program Studi Teknik Informatika, Fakultas Teknik Elektro dan Teknologi Informasi
Institut Teknologi Adhi Tama Surabaya

*Penulis korespondensi. E-mail: sugianto@itats.ac.id

ABSTRACT

This research explores the use of the Agile Scrum methodology in frontend development using the React.js framework, emphasizing its application for enabling continuous iteration in web application development. This methodology was selected for its advantages in providing flexibility and adaptability during feature development for web applications. The results indicate that applying Agile Scrum in frontend development with React.js enhances team efficiency, improves the aesthetic quality of websites, and ensures regular updates. However, challenges include difficulties in accurately estimating the time needed for each feature in an iteration and maintaining the high level of discipline required from the team.

Kata Kunci

Agile Scrum;
Kerangka Kerja React.js;
Pengembangan Frontend;
Pengembangan Software;
Pengujian Black Box

Keywords

Agile Scrum;
Black Box Testing;
Frontend Development;
React.js Framework;
Software Development

ABSTRAK

Penelitian ini mengeksplorasi penggunaan metodologi *Agile Scrum* dalam pengembangan *frontend* menggunakan *framework React.js*, dengan menekankan penerapannya untuk memungkinkan iterasi berkelanjutan dalam pengembangan aplikasi web. Metodologi ini dipilih karena keunggulannya dalam menyediakan fleksibilitas dan kemampuan beradaptasi selama pengembangan fitur untuk aplikasi web. Hasil penelitian ini menunjukkan bahwa penerapan *Agile Scrum* dalam pengembangan *frontend* dengan *React.js* dapat meningkatkan efisiensi tim pengembang, meningkatkan kualitas estetika situs web, serta memastikan pembaruan informasi secara berkala. Namun, tantangannya meliputi kesulitan dalam memperkirakan waktu yang dibutuhkan untuk setiap fitur dalam satu iterasi secara akurat dan mempertahankan tingkat disiplin tinggi yang dibutuhkan dari tim.

PENDAHULUAN

Pengembangan *frontend* dalam sebuah situs *web* memainkan peran krusial dalam memberikan pengalaman pengguna yang optimal. *Frontend* merupakan antarmuka yang berinteraksi langsung dengan pengguna, sehingga kualitasnya sangat mempengaruhi kepuasan dan loyalitas pengguna. Model proses rekayasa perangkat lunak yang efisien dan adaptif sangat penting untuk memastikan bahwa situs *web* dapat berkembang sesuai dengan kebutuhan dan umpan balik pengguna. Penelitian ini adalah pengembangan *frontend website* pada *platform survey online* menggunakan metodologi *Agile Scrum* untuk mendukung adaptabilitas dan skalabilitas pengembangan *frontend website*.

Beberapa penelitian sebelumnya telah menunjukkan berbagai model proses rekayasa perangkat lunak yang digunakan untuk mengembangkan aplikasi berbasis *web*. Model *Waterfall* digunakan untuk pembuatan aplikasi sistem informasi simpan pinjam koperasi dan perancangan *frontend website* [1], [2]. Model ini memiliki tahapan yang jelas dan terstruktur, mulai dari perencanaan hingga pemeliharaan. Salah satu keunggulan model ini adalah proses pengembangan yang terorganisir dan mudah dipahami, namun kelemahannya adalah sifatnya yang linear dan kurang fleksibel terhadap perubahan. Model *Waterfall* juga digunakan untuk implementasi QR Code pada *android* [3], yang menunjukkan bahwa meskipun model ini memungkinkan perencanaan yang matang, pengujian yang dilakukan setelah implementasi selesai menyebabkan masalah sulit diatasi secara dini.

Metode *User Centered Design* (UCD) dalam pengembangan *frontend website* perpustakaan [4] mengutamakan interaksi pengguna dan umpan balik untuk memastikan bahwa situs tersebut sesuai dengan kebutuhan pengguna. Model *Prototype* memungkinkan perbaikan dan penyesuaian berdasarkan umpan balik dari user sejak tahap awal pengembangan, meskipun iterasi awal mungkin terlalu sederhana dan tidak mencerminkan semua aspek fungsionalitas akhir [5], [6]. Penggunaan model prototyping dalam implementasi *React.js* untuk sistem informasi digital printing [7] memungkinkan perbaikan dari awal untuk memenuhi kebutuhan *user*. Pada perancangan *web portal* lomba [8], metode *Feature Driven Development* (FDD) memungkinkan manajemen yang lebih terstruktur terhadap kompleksitas aplikasi, tetapi kurang fleksibel untuk perubahan kebutuhan implementasi.

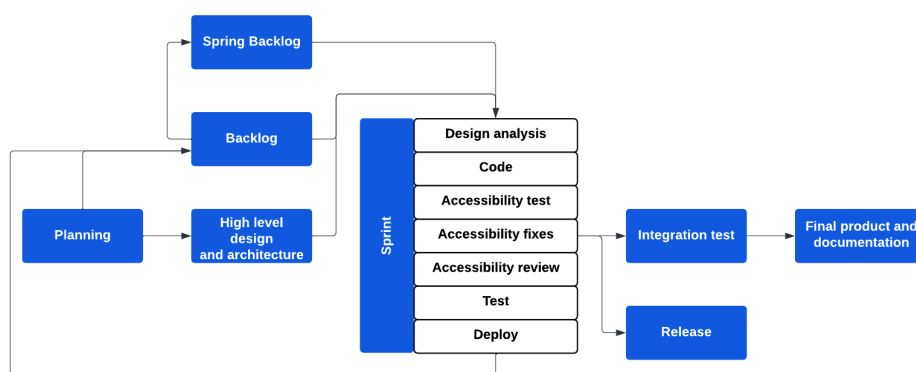
Metodologi *Agile Scrum* menawarkan fleksibilitas dan adaptabilitas yang lebih baik. Metodologi ini memungkinkan pengembangan fitur secara bertahap dan memanfaatkan umpan balik dari pengguna untuk perbaikan di iterasi berikutnya [9], [10]. Salah satu tantangan dalam metodologi ini adalah memperkirakan waktu dan fitur untuk setiap iterasi. Selain itu, metodologi ini membutuhkan disiplin yang tinggi dari tim untuk mengikuti rutinitas *Scrum*. Implementasi *React.js* dalam pengembangan *frontend website* [10] menunjukkan bahwa metodologi *Agile Scrum* memiliki kemampuan untuk membuat tampilan *website* yang menarik dan informasi yang lebih *up-to-date*, meskipun intensitas pertemuan dan *deadline* yang ketat dapat menyebabkan kelelahan tim.

TINJAUAN PUSTAKA

Metodologi *Agile Scrum*

Metodologi *Agile* adalah salah satu model proses rekayasa perangkat lunak yang efektif dan fleksibel. Metodologi ini tidak mengatur secara rinci prosedur tentang bagaimana membuat model yang telah ditentukan, meskipun ada cara untuk menjadi *modeler* yang efektif. Pendekatan *Agile* untuk pengembangan *website* mengutamakan pendekatan iteratif, di mana tim bekerja secara mandiri dan berkolaborasi untuk membangun perangkat lunak [11].

Scrum adalah pengembangan metodologi *Agile* yang digunakan untuk mengelola pengembangan produk kompleks dengan tujuan meningkatkan nilai produk secara kreatif dan efisien. *Scrum* bukanlah suatu proses, teknik, atau metodologi tertentu; sebaliknya, itu adalah kerangka kerja yang mencakup berbagai elemen dari proses, teknik, dan metodologi tersebut. *Scrum* telah digunakan sejak awal tahun 1990 [12]. Salah satu contoh model proses dalam implementasi *Agile Scrum* dapat dilihat pada Gambar 1.



Gambar 1. Model proses metodologi *Agile Scrum* [13]

Berikut adalah proses utama dalam *Scrum*:

1. *Sprint*

Sprint terdiri dari serangkaian tugas yang harus diselesaikan untuk memenuhi kebutuhan yang telah ditetapkan, dengan batas waktu yang telah ditetapkan sebelumnya untuk tim. Pada tahap ini, pengembangan aplikasi dimulai sesuai dengan perencanaan *sprint* [12].

2. *Sprint Planning*

Sprint planning adalah proses perencanaan untuk menyelesaikan *task* dalam satu *sprint*. Pada tahap ini, tugas dibagi ke masing-masing individu. Selain itu, *sprint planning* juga menghitung jumlah waktu yang diperlukan untuk membangun fitur dalam setiap *sprint* [12].

3. *Development*

Development merujuk pada proses atau kegiatan pengembangan atau pengerjaan tugas atau aktivitas tertentu dalam konteks pengelolaan proyek atau pengembangan produk. Dalam model seperti *Scrum* atau metodologi pengembangan lainnya, *development* mengacu pada pembuatan dan pelaksanaan tugas-tugas untuk mencapai tujuan proyek atau *sprint* yang telah ditetapkan [12].

4. *Daily Scrum*

Daily Scrum merupakan kegiatan harian dalam *sprint* yang dilakukan oleh tim *Scrum* untuk mengevaluasi pekerjaan yang sudah dilakukan serta potensi hambatan dalam proyek. Tim *Scrum* menggunakan *Daily Scrum* sebagai alat untuk meningkatkan kemajuan produk menuju pencapaian *sprint* [14].

5. *Sprint Review*

Sprint review adalah kegiatan evaluasi yang dilakukan oleh tim *Scrum* setelah selesai satu periode *sprint*. Pada tahap ini, tim meninjau pekerjaan yang telah selesai, memperlihatkan hasil kepada *stakeholders*, dan mendapatkan umpan balik untuk memastikan bahwa produk sesuai dengan ekspektasi dan kebutuhan yang telah ditetapkan [12].

Black Box Testing

Black Box Testing adalah cara untuk menguji fungsionalitas aplikasi tanpa memperhatikan struktur dan cara kerja program. Untuk melakukan pengujian, deskripsi perangkat lunak, seperti spesifikasi, persyaratan, dan desain, digunakan saat mengembangkan kasus uji. Kasus uji dapat fungsional atau non-fungsional. Selama proses ini, evaluator menentukan output yang seharusnya dihasilkan oleh aplikasi dan menguji dengan input yang valid dan tidak valid [15].

Penelitian Terdahulu

Beberapa penelitian terdahulu mencakup pembuatan aplikasi sistem informasi simpan pinjam koperasi berbasis *website* menggunakan model *Waterfall* [1]. Penelitian ini mengembangkan sebuah *website* dengan tahapan yang jelas dan terstruktur, mulai dari perencanaan hingga pemeliharaan. Meskipun model ini membuat proses pengembangan lebih terorganisir dan mudah dipahami, kelemahannya terletak pada sifatnya yang linear dan kurang fleksibel terhadap perubahan. Selanjutnya, ada penelitian yang membahas perancangan *frontend website* dengan model yang sama [2], menekankan tahapan terstruktur dari awal hingga akhir proyek, tetapi menghadapi tantangan serupa terkait fleksibilitas terhadap perubahan kebutuhan. Sebuah studi lain mengeksplorasi implementasi *QR Code* pada *platform Android* menggunakan model *Waterfall* [3]. Meskipun model ini memungkinkan perencanaan yang matang, pengujian yang dilakukan setelah implementasi sering kali menyulitkan penanganan masalah yang muncul.

Penelitian mengenai pengembangan *frontend website* perpustakaan dengan menggunakan metode *User Centered Design* (UCD) [4] menunjukkan pentingnya partisipasi aktif pengguna dalam proses desain. Metode ini melibatkan wawancara dan evaluasi iteratif untuk memastikan kesesuaian dengan kebutuhan pengguna. Model *prototyping* yang digunakan untuk mengembangkan sistem informasi persediaan barang [4] memungkinkan perbaikan iteratif berdasarkan umpan balik pengguna, meskipun prototipe awal seringkali sederhana dan belum mencakup semua fungsionalitas akhir. Implementasi *progressive web app* dan penggunaan *React.js* untuk sistem informasi digital *printing* [7] dengan model ini menunjukkan fleksibilitas penyesuaian dari awal, meski menghadapi tantangan dalam menetapkan prioritas pengembangan yang memenuhi kebutuhan pengguna [6]. Metode *Feature Driven Development* (FDD) digunakan pada penelitian yang fokus pada pembagian pengembangan perangkat lunak berdasarkan fitur,

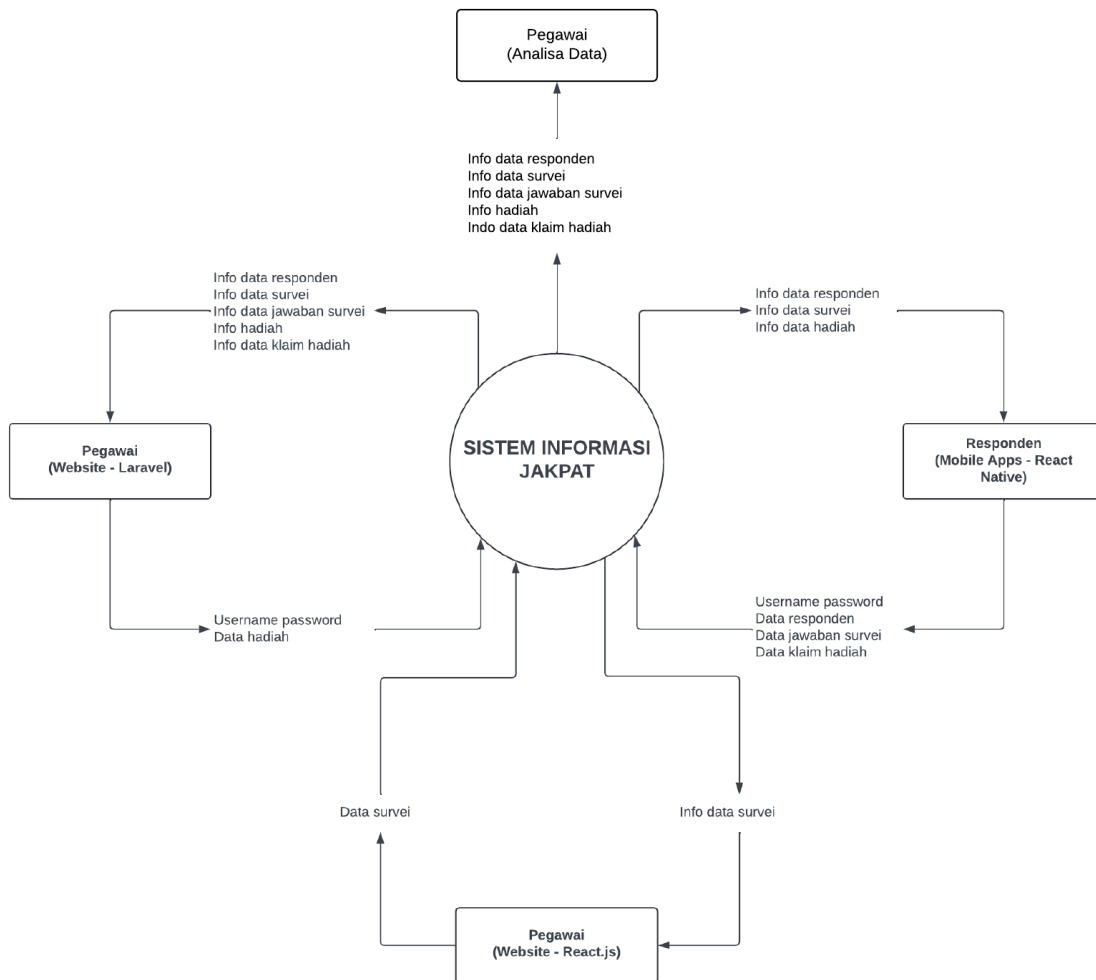
memungkinkan manajemen yang lebih terstruktur namun kurang fleksibel terhadap perubahan persyaratan implementasi [8].

Studi tentang pengembangan aplikasi *marketplace* dengan metodologi *Agile Scrum* menyoroti keunggulan pengembangan fitur secara bertahap dan adaptabilitas terhadap perubahan, meskipun menghadapi tantangan dalam estimasi waktu dan fitur setiap iterasi [9]. Implementasi *React.js* untuk *frontend website* [10] juga menggunakan metodologi ini, menghasilkan tampilan yang lebih menarik dan informasi terkini, tetapi menghadapi tantangan dalam disiplin tim dan manajemen waktu. Metodologi *Agile Scrum* mencakup *sprint planning*, *development*, *daily scrum*, *testing*, evaluasi kode, dan *sprint review* dengan *stakeholders* dan tim di setiap akhir *sprint*. Perbandingan metode pengembangan pada penelitian terdahulu dapat dilihat pada Tabel 1.

METODE

Penerapan *Scrum* dan Perancangan Sistem

Pekerjaan dilakukan pada proyek *Platform Survey Online* berbasis *website* menggunakan *frontend framework React.js* dengan metodologi *Agile Scrum* yang masih berada dalam tahap pengembangan di mana proyek ini berfokus pada menu pembuatan pertanyaan pada survei seperti judul pertanyaan, gambar pada pertanyaan, kolom jawaban, dan gambar pada kolom jawaban. Sistem informasi yang menghubungkan keseluruhan proyek pada penelitian ini dapat dilihat pada Gambar 2 dan Gambar 3.



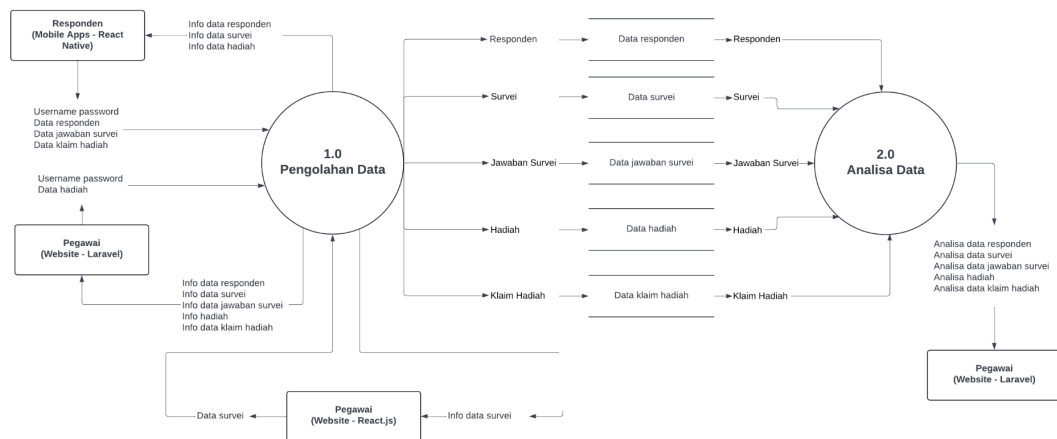
Gambar 2. Diagram Konteks

Tabel 1. Perbandingan penelitian terdahulu

No.	Perangkat Lunak Yang Diteliti	Metode		
		Nama	Keunggulan	Kekurangan
1	Sistem Informasi Simpan Pinjam Koperasi Berbasis Web [1]	Waterfall	<i>Waterfall</i> memiliki tahapan jelas, diantaranya perencanaan, analisis, desain, implementasi, pengujian, dan pemeliharaan. Setiap fase menghasilkan dokumen penting untuk referensi dan pemeliharaan sistem di masa depan.	Bersifat linear, sulit menangani perubahan kebutuhan, dan masalah sering terdeteksi terlambat, memerlukan biaya besar untuk diperbaiki. Tidak efektif untuk proyek yang tidak jelas.
2	Frontend Website Profile [2]	Prototype	Penetapan tahapan awal hingga akhir membantu perencanaan dan estimasi. Dokumentasi di setiap fase memudahkan pemahaman dan pelacakan. Penyampaian yang jelas membuat proyek mudah dipahami.	Sulit mengakomodasi perubahan kebutuhan atau pasar yang cepat. Evaluasi di akhir siklus dapat mendeteksi kesalahan terlambat. Tidak cocok untuk proyek kompleks atau yang membutuhkan <i>RnD</i> .
3	Otentikasi dengan Teknologi QR-Code Berbasis Android [3]	Waterfall	Model <i>Waterfall</i> mengikuti tahapan terstruktur dari analisis hingga validasi, memastikan perencanaan matang dan pemahaman yang baik terhadap kebutuhan proyek.	Perubahan sulit setelah analisis dan desain selesai, biaya pengujian setelah implementasi meningkat jika ada kesalahan, dan kesalahan awal sulit diperbaiki.
4	Frontend Website Perpustakaan [4]	UCD	Sistem dirancang sesuai kebutuhan pengguna dengan proses wawancara dan kuesioner untuk detail kebutuhan. Desain antarmuka dievaluasi dan diperbaiki berdasarkan umpan balik pengguna.	Wawancara, kuesioner, dan evaluasi membutuhkan waktu dan sumber daya. Keberhasilan tergantung pada partisipasi aktif pengguna dan kompleksitas umpan balik.
5	Sistem Informasi Pengelolaan Persediaan Barang [5]	Prototype	Wawancara dan observasi lalu kebutuhan pengguna diintegrasikan dalam desain sistem. <i>Prototyping</i> memungkinkan penyesuaian berdasarkan <i>feedback</i> pengguna, sementara iterasi kedua dapat menambahkan fitur lain.	Prototipe awal terlalu sederhana dan tidak mencakup semua fungsionalitas, kurang memenuhi kebutuhan pengguna. Keputusan desain dari prototipe yang belum lengkap dapat mengecewakan pengguna.
6	Website E-Reses DPRD Kabupaten Sukabumi [6]	Prototype	Memfasilitasi perbaikan berdasarkan <i>feedback stakeholder</i> menggunakan <i>wireframe</i> dan <i>mockup</i> . Mengurangi risiko pengembangan dan evaluasi langsung dari <i>prototype</i> untuk memastikan kesesuaian.	Prototipe awal tidak mencakup semua fitur kompleks, perlu iterasi tambahan dan investasi sumber daya yang signifikan. Bergantung pada respons <i>stakeholder</i> dan komunikasi antara ekspektasi awal dan akhir.

Tabel 1. (lanjutan)

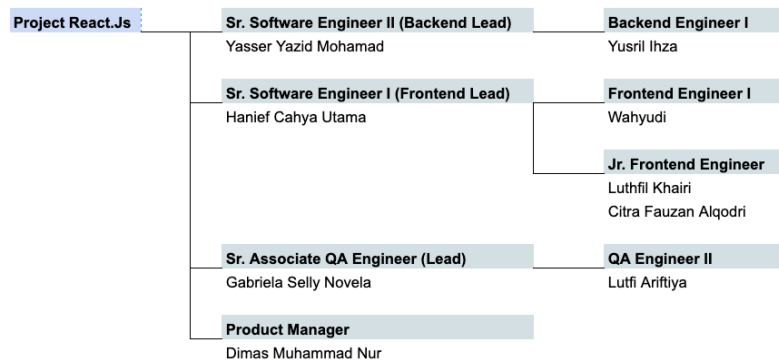
No.	Perangkat Lunak Yang Diteliti	Metode		
		Nama	Keunggulan	Kekurangan
7	Sistem Informasi Digital Printing Berbasis Website [7]	RAD & Prototype	Prototype unggul dalam iterasi cepat yang dapat mempercepat pengembangan perangkat lunak, melibatkan pengguna untuk memastikan kebutuhan dan meminimalisir kesalahpahaman.	Fokus pada kecepatan pengembangan bisa mengabaikan skala, keamanan, dan efisiensi. Pengembangan iteratif berisiko tidak mencerminkan kebutuhan akhir.
8	Web Portal Lomba [8]	Agile FDD	Pembagian pengembangan perangkat lunak berdasarkan fitur untuk manajemen yang terstruktur. Memungkinkan progres perangkat lunak yang cepat dan berkualitas tinggi, dengan proyeksi jelas terhadap kebutuhan pengguna.	Tidak cocok untuk proyek besar atau yang membutuhkan fleksibilitas ekstrim dalam perubahan. Memerlukan tim yang mahir dalam mendefinisikan dan mengelola setiap fitur, perubahan mendadak bisa menjadi tantangan.
9	Aplikasi Marketplace Ikan Di Kabupaten Probolinggo [9]	Agile Scrum	Responsif terhadap perubahan pasar dan kebutuhan. Pengembangan fitur secara bertahap dengan uji coba langsung dan implementasi masukan pengguna dalam setiap iterasi.	Memerlukan partisipasi aktif dari pengguna dan <i>stakeholder</i> , serta tantangan dalam memperkirakan waktu dan fitur untuk setiap iterasi.
10	Frontend Website PT XYZ [10]	Agile Scrum	Tim bekerja kolaboratif dan adaptif untuk hasil <i>website</i> yang menarik. Menggunakan <i>sprint</i> terstruktur dengan untuk meningkatkan efektivitas dan kualitas tim.	Memerlukan disiplin tinggi dalam rutinitas seperti <i>daily scrum</i> , <i>sprint planning</i> , dan <i>retrospective</i> , yang menguras waktu. Pembagian tugas dan jadwal yang ketat bisa menyebabkan kelelahan tim.
11	Sistem Informasi Disabilitas di Kosta Rika [13]	Agile Scrum	Beradaptasi dengan menambahkan tugas dalam prosesnya, interaksi dengan tim validasi memungkinkan masalah diselesaikan tepat waktu.	Memerlukan disiplin tinggi dalam rutinitas, tantangan dalam memperkirakan waktu dan fitur untuk setiap iterasi.
12	Penelitian ini: Frontend Website Pada Platform Survey Online	Agile Scrum	Pengembangan iteratif menjadi responsif terhadap perubahan kebutuhan. Proses pengembangan lebih terstruktur dengan <i>sprint planning</i> , <i>development</i> dan <i>daily</i> , dan <i>sprint review</i> .	Pembagian tugas dan jadwal yang ketat bisa menyebabkan kelelahan tim.



Gambar 3. DFD (Data Flow Diagram) Level 1

Struktur Organisasi Tim

Struktur Organisasi Tim dapat dilihat dengan lengkap pada Gambar 4.



Gambar 4. Struktur Organisasi Tim

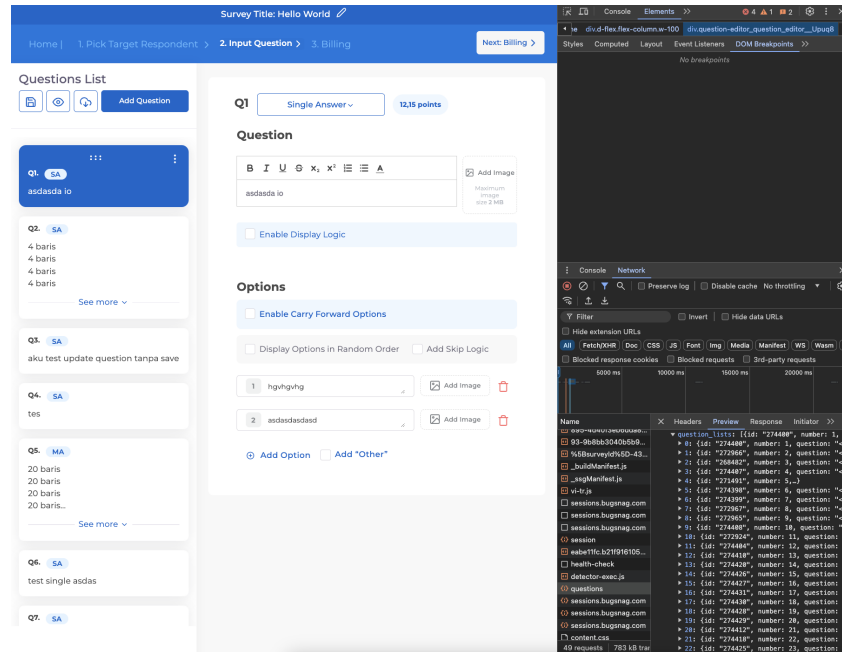
Pembagian Tugas

Tim pada *project React.js* terdiri dari satu orang *Project Manager* yang bertugas dalam mengatur manajemen *Scrum* termasuk penjadwalan fitur apa yang akan dibangun pada setiap sprint, dua orang *Backend Engineer* yang bertugas dalam membuat *Application Programming Interface* (API) untuk keperluan manajemen data, empat orang *Frontend Engineer* yang bertugas dalam membangun antarmuka *website*, dan dua orang *Quality Assurance Engineer* yang bertugas dalam melakukan pengujian dari fitur yang dikerjakan oleh *Backend Engineer* dan *Frontend Engineer* menggunakan metode *Black Box Testing*.

HASIL DAN PEMBAHASAN

Load All Question

Load All Question adalah sebuah fungsi yang dikembangkan untuk mendapatkan seluruh pertanyaan pada satu laman utama agar saat *user* mengakses halaman utama, *user* dapat melihat seluruh data pertanyaan. Hasil dari fungsi *Load All Question* dapat dilihat pada Gambar 5.



Gambar 5. Hasil *Load All Question*

UI Component Single Answer Form

UI Component Single Answer Form adalah sebuah komponen yang dikembangkan sebagai fitur kepada user mengisi pemilihan opsi pada pertanyaan dengan tipe *Single Answer* di survey yang dikembangkan. Hasil dari UI Component Single Answer Form dapat dilihat pada Gambar 6.

Options

☐ Enable Carry Forward Options

☐ Display Options in Random Order ☐ Add Skip Logic

1 Option 1

2 Option 2

☒ Add Option ☐ Add "Other"

Gambar 6. Hasil UI Component Single Answer Form

UI Component Grid Single Answer Form

UI Component Grid Single Answer Form adalah sebuah komponen yang dikembangkan sebagai fitur kepada user mengisi pemilihan opsi pada pertanyaan dengan tipe *Grid Single Answer* di survey yang dikembangkan. Hasil dari UI Component Grid Single Answer dapat dilihat pada Gambar 7.

Row

☐ Enable Carry Forward Options

☐ Display Row in Random Order ☐ Add Skip Logic

1

Row 1

☒ Add Image

2

Row 2

☒ Add Image

☒ Add Row ☐ Add "Other"

Column

☐ Enable Carry Forward Options

☐ Display Column in Random Order ☐ Add Skip Logic

1

Column 1

☒ Add Image

2

Column 2

☒ Add Image

☒ Add Column ☐ Add "Other"

Gambar 7. Hasil UI *Component Grid Single Answer Form*

UI Component Action Form

UI *Component Action Form* adalah sebuah komponen yang dikembangkan sebagai fitur kepada *user* mengisi pemilihan opsi pada pertanyaan dengan tipe *Action* disurvey yang dikembangkan. Hasil dari UI *Component Action Form* dapat dilihat pada Gambar 8.

[illegible]

Gambar 8. Hasil UI *Component Action Form*

UI Component Scale Form

UI *Component Scale Form* adalah sebuah komponen yang dikembangkan sebagai fitur kepada *user* mengisi pemilihan opsi pada pertanyaan dengan tipe *Scale* di survey yang dikembangkan. Hasil dari UI *Component Scale Form* dapat dilihat pada Gambar 9.

Scale

 Respondent will be asked to select in a scale. Question Example: How satisfied are you?
Min. scale 0, Max. scale 20.

Min. Scale Max. Scale

Gambar 9. Hasil UI *Component Scale Form*

UI Component Add Other

UI Component Add Other adalah sebuah komponen yang dikembangkan sebagai fitur kepada user mengisi pemilihan opsi lainnya pada pertanyaan dengan tipe *Single Answer* di survei yang dikembangkan. Hasil dari UI Component Add Other dapat dilihat pada Gambar 10.

Gambar 10. Hasil UI Component Add Other

Hasil Black Box Testing

Seluruh pengembangan yang dilakukan diuji menggunakan metode *Black Box Testing* dan seluruh hasil uji dapat dilihat pada Tabel 2.

Tabel 2. Pengujian *Black Box Testing*

No.	Skenario Pengujian	Hasil Yang Diharapkan	Hasil
1	Fungsi <i>Load All Question</i>	User mengakses halaman utama dan seluruh <i>question</i> dapat dilihat	Lolos
2	UI Component <i>Single Answer Form</i>	UI Component <i>Single Answer Form</i> sesuai dengan <i>mockup</i>	Lolos
3	UI Component <i>Single Answer Form</i> mengganti nilai <i>component option</i>	User dapat mengubah nilai dari <i>option</i>	Lolos
4	UI Component <i>Grid Single Answer Form</i>	UI Component <i>Grid Single Answer Form</i> sesuai dengan <i>mockup</i>	Lolos
5	UI Component <i>Grid Single Answer Form</i> mengganti nilai <i>row</i>	User dapat mengubah nilai dari <i>row</i>	Lolos
6	UI Component <i>Grid Single Answer Form</i> mengganti nilai <i>column</i>	User dapat mengubah nilai dari <i>column</i>	Lolos
7	UI Component <i>Action Form</i>	UI Component <i>Action Form</i> sesuai dengan <i>mockup</i>	Lolos
8	UI Component <i>Action Form</i> mengganti tipe <i>action</i>	User dapat mengubah tipe dari <i>option</i>	Lolos
9	UI Component <i>Action Form</i> mengganti nilai kolom <i>form</i>	User dapat mengubah nilai dari kolom <i>form</i>	Lolos
10	UI Component <i>Form Input Question</i>	UI Component <i>Form Input Question</i> sesuai dengan <i>mockup</i>	Lolos
11	UI Component <i>Form Input Question</i> memilih <i>component Min. Scale</i>	User dapat mengubah nilai dari <i>Min. Scale</i>	Lolos
12	UI Component <i>Form Input Question</i> memilih <i>component Max. Scale</i>	User dapat mengubah nilai dari <i>Max. Scale</i>	Lolos
13	UI Component <i>Add Other</i>	UI Component <i>Add Other</i> sesuai dengan <i>mockup</i>	Lolos
14	UI Component <i>Add Other</i> mengganti nilai <i>checkbox</i>	User dapat mengubah nilai dari <i>checkbox</i> Component <i>Add Other</i>	Lolos

KESIMPULAN

Setelah dilakukan pengembangan frontend menggunakan metodologi *Agile Scrum*, penelitian ini menghasilkan beberapa fitur dengan pengembangan yang berulang. Proses pengembangan tersebut dapat membantu dalam meningkatkan kualitas *website* secara bertahap dan memastikan bahwa setiap iterasi membawa peningkatan yang signifikan. Selain itu, metodologi ini juga memungkinkan tim untuk beradaptasi dengan perubahan kebutuhan secara cepat dan efisien, serta meningkatkan kolaborasi dan komunikasi antar anggota tim. Hasil penelitian ini menunjukkan bahwa metodologi *Agile Scrum* merupakan pendekatan yang efektif.

DAFTAR PUSTAKA

- [1] Handayani, D., & Salam, M., "Aplikasi Sistem Informasi Simpan Pinjam Koperasi Berbasis Website Menggunakan Metode Waterfall," *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 3, no. 5, p. 425-434, 2023.
- [2] Aklani, S. A., & Liling, J. R., "Perancangan Front End Website Profile Builder Menggunakan Framework React Js Pada PT. Pundi Mas Berjaya," *The 5th National Conference for Community Service Project 2023*, vol. 5, no. 1, p. 863-872, 2023.
- [3] Parlika, R., N. R. Sandyca, Andreanto, B., R. M. Ihsanur, & M. A. Fahri, "Implementasi Otentikasi Dengan Teknologi QR-Code Berbasis Android Menggunakan Codeigniter Dan React Native," *e-NARODROID*, vol. 5, no. 2, p. 56-67, 2019.
- [4] Prawastiyo, C. A., & Hermawan, I., "Pengembangan Front-End Website Perpustakaan Politeknik Negeri Jakarta Dengan Menggunakan Metode UCD (User Centered Design)," *Journal Information Science and Library*, vol. 1, no. 2, p. 1-11, 2020.
- [5] Nursaid, F. F., Brata, A. H., & Kharisma, A. P., "Pengembangan Sistem Informasi Pengelolaan Persediaan Barang Dengan ReactJS Dan React Native Menggunakan Prototype (Studi Kasus : Toko Uda Fajri)," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 4, no. 1, p. 46-55, 2020.
- [6] Ramdhani, M. D., Asriyanik, A., & Pambudi, A., "Implementasi Progressive Web App Pada Website E-Reses DPRD Kabupaten Sukabumi dengan Metode Prototipe," *Jurnal SAINTIKOM (Jurnal Sains Manajemen Informatika dan Komputer)*, vol. 22, no. 2, p. 260-269, 2023.
- [7] Nuraminudin, M., Dewi, M. M., Suharsono, S., Dahlan, A., & Lukman, L., "Implementasi Reactjs Pada Pembuatan Sistem Informasi Digital Printing Berbasis Website," *Information System Journal (INFOS)*, vol. 6, no. 1, p. 25-32, 2023.
- [8] Permana, P., & T. I., "Implementasi Metode Feature Driven Development Pada Perancangan Web Portal Lomba," *Kohesi: Jurnal Sains dan Teknologi*, vol. 1, no. 6, p. 1-10, 2023.
- [9] Yaqin, M. A., "Pengembangan Aplikasi Marketplace Ikan Di Kabupaten Probolinggo Berbasis Frontend Backend Menggunakan React Js," *Nusantara Journal of Computers and Its Applications*, vol. 8, no. 2, p. 63-73, 2023.
- [10] Kinanti & Widiyari, L., "Implementasi React JS dalam Mengembangkan Front-End Website PT XYZ," *Infoman's : Jurnal Ilmu-Ilmu Informatika Dan Manajemen*, vol. 17, no. 2, p. 1-10, 2023.
- [11] Ariesta, A., Dewi, Y. N., Sariasih, F. A., & Fibriany, F. W., "Penerapan Metode Agile Dalam Pengembangan Application Programming Interface System Pada PT XYZ," *Jurnal CoreIT*, vol. 7, no. 1, p. 38-42, 2021.
- [12] Prastio, C. E., "Aplikasi Self-Service Menu Menggunakan Metode Scrum Berbasis Android," *PETIR*, vol. 11, no. 2, p. 203-220, 2018.
- [13] Chacon, V. R., Camacho, H., R., González, J., R., Blanco, A., G., & Rivas, M., C., "Adapting SCRUM methodology to develop accessible web sites," *2019 International Conference on Inclusive Technologies and Education (CONTIE)*, p. 112-116, 2019.
- [14] Andipradana, A., & Hartomo, K. D., "Rancang Bangun Aplikasi Penjualan Online Berbasis Web Menggunakan Metode Scrum," *Jurnal Algoritma*, vol. 18, no. 1, p. 161-172, 2021.
- [15] Putra, A. P., Andriyanto, F., Karisman, K., Harti, T. D. M., & Sari, W. P., "Pengujian Aplikasi Point Of Sale Menggunakan Blackbox Testing," *Jurnal Bina Komputer*, vol. 2, no. 1, p. 74-78, 2020.