

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Studi Sebelumnya

Harlinda dan timnya melakukan penelitian mengenai masalah utama yang dihadapi, yaitu selama pandemi, pelatihan dan seminar dilaksanakan secara rutin untuk meningkatkan keterampilan dosen dan mahasiswa. Namun, sebagian besar pelatihan dan seminar masih dikelola melalui berbagai platform yang belum terintegrasi dengan baik. Oleh karena itu, diperlukan perancangan dan implementasi sistem informasi yang dapat mengatur pelatihan dan seminar tersebut. Metode scrum dipilih karena menekankan pada kualitas, fleksibilitas, dan keterlibatan pengguna dalam pengembangan perangkat lunak. Fitur-fitur yang diimplementasikan mencakup registrasi pengguna, login, pengingat kata sandi, manajemen pelatihan, manajemen sertifikasi, manajemen seminar, serta integrasi pembayaran melalui payment gateway. Penerapan metodologi Scrum dalam pengembangan aplikasi manajemen pelatihan dan seminar terbukti sangat efektif, sehingga sistem ini dapat meningkatkan efisiensi dan efektivitas dalam proses pendaftaran sertifikat di Fakultas Ilmu Komputer UMI[6].

Fikri dan timnya melakukan penelitian terkait implementasi tanda tangan digital dalam pembuatan surat keterangan di Fakultas Ekonomi Universitas Garut (FEKON UNIGA), Tanda tangan digital berfungsi sebagai metode matematis yang digunakan untuk mengidentifikasi dokumen secara elektronik., memastikan keaslian dan keabsahan informasi yang terkandung di dalamnya. Dengan penerapan tanda tangan digital, proses *administrasi* di lingkungan akademik diharapkan menjadi lebih efisien, terutama dalam layanan surat menyurat yang umum digunakan terkait kebutuhan untuk meningkatkan efektivitas pelayanan *administratif* di perguruan tinggi, di mana banyak dokumen digital perlu dikeluarkan dan diverifikasi keasliannya. Penelitian ini bertujuan untuk merancang sistem tanda tangan digital berbasis web yang dapat menerbitkan sertifikat digital untuk surat keterangan. Metodologi yang digunakan adalah *Scrum*, yang merupakan salah satu pendekatan *Agile* dalam pengembangan perangkat lunak. Proses penelitian dibagi menjadi beberapa tahap, termasuk inisiasi proyek, perencanaan dan estimasi, implementasi, tinjauan, dan rilis. Hasil survei

menunjukkan bahwa metodologi *Scrum* digunakan oleh 20,3% responden dalam pengembangan perangkat lunak, menegaskan relevansi dan efektivitas metode ini dalam konteks penelitian. Secara keseluruhan, penelitian ini bertujuan untuk memberikan solusi yang praktis dan efisien dalam verifikasi dokumen akademik melalui penggunaan tanda tangan digital [7].

Nurwahyuni dan timnya melakukan sebuah penelitian yang berlatar belakang Proses pengolahan data pendaftaran sertifikat di Badan Pertanahan Nasional mengalami kendala seperti penggunaan sistem yang tidak optimal dan penginputan data secara manual, yang menyebabkan antrian panjang dan waktu tunggu yang lama. Penelitian ini menggunakan pendekatan Extreme Programming (XP) untuk merancang sistem informasi manajemen sertifikat. XP dipilih karena fokusnya pada kualitas, fleksibilitas, dan keterlibatan pengguna dalam pengembangan perangkat lunak. Proses pengembangan dibagi menjadi beberapa tahapan: perencanaan, desain, coding, dan pengujian. Penerapan metodologi XP dalam pengembangan sistem informasi manajemen sertifikat berhasil meningkatkan efisiensi dan efektivitas proses pendaftaran sertifikat di Badan Pertanahan Nasional. Sistem ini tidak hanya mempercepat proses tetapi juga meningkatkan akurasi dalam pengelolaan data pendaftaran sertifikat, hasil pengujian fungsionalitas menunjukkan bahwa sistem memenuhi harapan dengan nilai 100% pada kuesioner *black box testing*. [8].

Eva Rahmawati dan timnya melakukan penelitian dengan latar belakang sebagai berikut. Di era digital yang semakin terhubung, webinar telah menjadi alat yang sangat penting untuk pendidikan, pelatihan, dan pertukaran informasi. Namun, proses pendaftaran dan distribusi e-certificate masih dilakukan secara semi-otomatis, yang sering menimbulkan berbagai masalah seperti kehilangan data, waktu penyelesaian yang lama, dan kebutuhan akan sumber daya manusia yang lebih banyak. Sistem ini bertujuan untuk meningkatkan kemudahan penggunaan dan efektivitas operasional, serta mengurangi beban kerja administratif. Penelitian ini menggunakan metode Extreme Programming (XP) dalam pengembangan sistem, karena metodologi ini mampu menangani perubahan kebutuhan dengan cepat dan efisien, serta memiliki fokus pada kualitas teknis dan kepuasan pelanggan [9].

Lucy dan timnya melakukan penelitian terhadap permasalahan utama pada akademi digital.id, yakni proses manual yang digunakan pada pembuatan sertifikat dapat menyebabkan keterlambatan dan kesalahan dalam pengelolaan data peserta. Penelitian ini bertujuan untuk merancang aplikasi sertifikat digital yang efektif dan mudah digunakan, Penelitian ini menggunakan metode Waterfall, yang meliputi beberapa tahap yaitu Identifikasi kebutuhan dan tujuan aplikasi, menentukan fitur yang diperlukan dan kebutuhan pengguna, implementasi menggunakan bahasa pemrograman PHP dan framework Laravel, melakukan pengujian untuk memastikan fungsionalitas aplikasi, dan memastikan aplikasi berjalan dengan baik di lingkungan produksi [10].

Tabel II.1 Studi Sebelumnya.

No.	Penulis	Tahun	Metode	Hasil
1	Harlinda, dkk [6]	2021	<i>Scrum</i>	Sistem yang dirancang memungkinkan pemantauan pendaftaran sertifikat secara efisien melalui aplikasi berbasis dashboard.
2	Fikri, dkk [7]	2023	<i>Scrum</i>	Sistem yang dibuat meningkatkan efisiensi. Sistem memungkinkan verifikasi integritas data secara matematis.
3	Nurwahyuni, dkk [8]	2022	XP	Sistem yang dirancang memungkinkan pendaftaran sertifikat secara lebih efisien melalui aplikasi berbasis <i>Dashboard</i> .
4	Eva Rahmawati, dkk [9]	2022	XP	Sistem diharapkan dapat meningkatkan pengalaman pengguna, mengurangi beban kerja <i>administratif</i> , dan meningkatkan kepuasan pelanggan.
5	Lucy Susannaiyah, dkk [10]	2024	Waterfall	Aplikasi yang dirancang memungkinkan pengguna untuk mengakses sertifikat secara <i>online</i> melalui tautan publik.

## 2.2 Dasar Teori

### 2.2.1 Pengembangan Sistem

Pengembangan sistem didefinisikan sebagai kegiatan untuk menciptakan sistem informasi berbasis komputer yang bertujuan menyelesaikan masalah organisasi atau memanfaatkan peluang yang muncul [11].

### 2.2.2 Website

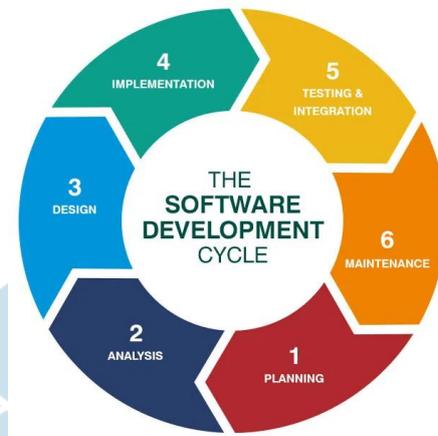
Website adalah sekumpulan halaman yang terdiri dari beberapa laman yang menyajikan informasi dalam bentuk data digital, termasuk teks, gambar, video, audio, dan animasi yang disediakan melalui alur koneksi internet (Abdullah et al., 2016). Sedangkan menurut (Yuhfizar, 2016) menyatakan “*Website* adalah keseluruhan halaman web yang terdapat dari sebuah domain yang mengandung informasi [12].

### 2.2.3 Software Development Life Cycle (SDLC)

Siklus Pengembangan Perangkat Lunak, yang sering disebut sebagai SDLC (*Software Development Life Cycle*), merupakan serangkaian langkah yang diambil dalam proses pengembangan sebuah sistem, dimulai dari tahap pra-produksi, fase produksi, hingga pasca-produksi. Siklus hidup sistem ini terdiri dari beberapa fase utama, seperti perencanaan, desain, pengembangan, dan pemeliharaan, di mana setiap tahap memiliki tujuan dan fungsi yang jelas [13].

Pada fase perencanaan, tim menentukan ruang lingkup proyek, mengumpulkan data dari *stakeholder*, dan membuat struktur tim, jadwal, serta anggaran. Di fase analisis, tim menentukan kebutuhan fungsional sistem dan mengidentifikasi masalah. Selanjutnya, pada fase desain, tim merancang arsitektur sistem dan antarmuka pengguna, serta memilih alat dan bahasa pemrograman yang tepat. Fase pengembangan melibatkan penulisan kode. Pada fase pengujian, sistem diuji untuk memastikan fungsionalitas, serta menemukan dan memperbaiki kesalahan. Fase implementasi bertujuan untuk menerapkan sistem ke lingkungan produksi, sedangkan fase pemeliharaan meliputi pemantauan dan perbaikan sistem setelah implementasi. Dengan mengikuti

tahapan SDLC yang terstruktur, tim dapat mengelola proyek dengan efisien dan memastikan sistem memenuhi kebutuhan pengguna [14].

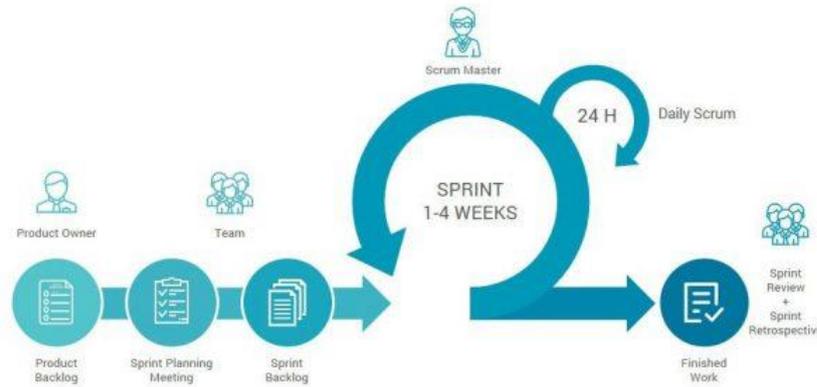


Gambar II.1 Software *Development* Life Cycle [15]

#### 2.2.4 *Scrum*

*Scrum* adalah sebuah kerangka kerja yang digunakan dalam pengembangan perangkat lunak, yang berfokus pada kolaborasi tim untuk meningkatkan komunikasi dan kerjasama antar anggota, sehingga dapat meningkatkan produktivitas di masa mendatang [16].

Kerangka kerja *scrum* mencakup serangkaian pertemuan, alat, dan peran yang dirancang untuk meningkatkan efisiensi pengiriman proyek. Praktik *scrum* yang mirip dengan tim olahraga yang berlatih untuk pertandingan besar, memungkinkan tim mengelola diri sendiri, belajar dari pengalaman, dan beradaptasi dengan perubahan. *Scrum* sering diterapkan dalam pengembangan perangkat lunak untuk menyelesaikan masalah kompleks secara efektif dan berkelanjutan. Prinsip dasar *scrum* meliputi transparansi, refleksi, dan adaptasi, yang membantu tim mengelola tantangan dan meningkatkan kualitas produk [17]. Gambar dibawah merupakan prosedur dari proses *scrum* yang telah ditetapkan.



Gambar II.2 Proses Scrum [18]

Dalam kerangka kerja *scrum*, terdapat tiga peran utama yang masing-masing memiliki tanggung jawab spesifik, yaitu *Scrum Master*, *Product Owner*, dan *Development Team*.

**Product Owner** bertugas untuk memaksimalkan nilai produk yang dihasilkan oleh tim pengembangan dengan mengelola *Product Backlog*, menyusun, dan memprioritaskan fitur serta tugas yang perlu dikerjakan. Mereka juga menetapkan visi produk dan berfungsi sebagai penghubung antara tim dan pemangku kepentingan untuk memastikan bahwa kebutuhan dan harapan mereka terpenuhi.

**Scrum Master** berperan sebagai fasilitator dan pelatih bagi tim *Scrum*, dengan tanggung jawab utama dalam memfasilitasi proses *Scrum* melalui pengorganisasian dan kepemimpinan pertemuan seperti *Sprint Planning* dan *Daily Scrum*. Mereka juga bertugas untuk mengatasi hambatan yang mengganggu kemajuan tim serta memberikan dukungan dan pelatihan agar prinsip-prinsip *Agile* dan *Scrum* dipahami dengan baik. [17]

**Development Team** adalah yang bertanggung jawab untuk mengembangkan dan menyelesaikan *Increment* produk dalam setiap *Sprint*. Mereka bekerja sama untuk menyelesaikan item-item dalam *Sprint Backlog*, mengambil inisiatif dalam merencanakan dan melaksanakan pekerjaan, serta beradaptasi dengan perubahan yang mungkin terjadi. Ketiga peran ini saling melengkapi dan berkontribusi pada keberhasilan tim *Scrum* dalam mencapai tujuan pengembangan produk yang efektif dan efisien [19].

### 2.2.5 Black Box Testing

*Black box testing* adalah suatu metode pengujian perangkat lunak yang menilai fungsionalitas aplikasi tanpa memperhatikan struktur internal atau rincian implementasinya. Dalam metode ini, penguji tidak memiliki pengetahuan tentang kode sumber atau cara kerja internal sistem yang diuji mereka hanya berinteraksi dengan aplikasi melalui antarmuka pengguna. Tujuan utama dari black box testing adalah untuk memastikan bahwa perangkat lunak memenuhi spesifikasi dan persyaratan yang telah ditentukan, serta berfungsi sesuai dengan ekspektasi pengguna. Pengujian ini dilakukan dengan merancang kasus uji berdasarkan kebutuhan dan spesifikasi perangkat lunak, sehingga penguji dapat memeriksa apakah output yang dihasilkan sesuai dengan input yang diberikan. Black box testing dapat diterapkan pada berbagai tingkat pengujian, termasuk pengujian unit, integrasi, sistem, dan penerimaan. Metode ini juga dikenal sebagai pengujian berbasis spesifikasi karena fokusnya terletak pada apa yang seharusnya dilakukan oleh perangkat lunak, bukan pada cara kerjanya[20].

## 2.3 Tabel Perbandingan

Tabel II.2 Tabel Perbandingan

<b>Aspek</b>	<b>Scrum</b>	<b>Waterfall</b>	<b>Kanban</b>
Pendekatan	Iteratif dan inkremental	Linear dan sekuensial	Berkelanjutan
Fleksibilitas	Sangat fleksibel terhadap perubahan	Kaku, sulit untuk mengakomodasi perubahan	Fleksibel, tetapi tidak terikat waktu
Waktu Pengembangan	Siklus pendek ( <i>sprint</i> ) 2-4 minggu	Fase panjang dengan waktu tetap	Tidak ada siklus waktu tetap
Peran Tim	Memiliki peran jelas ( <i>Scrum Master, Development Team</i> )	Tidak ada peran formal	Tidak ada peran formal

<b>Aspek</b>	<b>Scrum</b>	<b>Waterfall</b>	<b>Kanban</b>
Kelebihan	Adaptif, meningkatkan kolaborasi tim	Mudah dikelola dan dipahami	Fokus pada alur kerja yang efisien
Kompatibilitas	Memerlukan tim yang berpengalaman	Sulit untuk mengubah spesifikasi	Kurang terstruktur

Pemilihan metode Scrum dalam penelitian ini didasarkan pada keunggulannya yang signifikan dalam mendukung pendekatan iteratif dan inkremental, yang memungkinkan pengembangan produk secara bertahap melalui siklus *sprint*, kemudian metode ini juga fleksibel terhadap perubahan yang diminta dalam *sprint*, serta atmosfer yang berada pada institusi juga mendukung untuk metode penelitian scrum, seperti adanya ruangan khusus untuk *daily scrum* dan juga memiliki stakeholder yang mengurus bagian tim scrum nya masing-masing, seperti *scrum master*.