

Penerapan *Simple Agile Methodology* dalam Pengembangan Aplikasi Web

Muhammad Fariz Iqbal
Program Studi Sarjana Informatika
Universitas Islam Indonesia
Yogyakarta, Indonesia
18523114@students.uui.ac.id

Hanson Prihantoro Putro
Program Studi Sarjana Informatika
Universitas Islam Indonesia
Yogyakarta, Indonesia
hanson@uui.ac.id

Abstract—Sistem Informasi Magang diciptakan untuk memudahkan mahasiswa dalam mencari perusahaan yang membuka lowongan magang. Kendala yang sering ditemui dalam pengembangan Sistem Informasi Manajemen adalah tidak efektifnya alur selama pengembangan yang mengakibatkan pengguna merasa kurang puas terhadap hasil Sistem Informasi Manajemen yang telah dikembangkan. *Simple Agile Methodology* adalah metode yang digunakan dalam pengembangan sistem yang memiliki keunggulan terhadap durasi waktu yang cepat dalam pengembangannya sehingga menjadi lebih efisien dan fleksibel serta meningkatkan rasio kepuasan pelanggan. Hasil pekerjaan yang didapatkan dalam penelitian ini adalah Pengembangan Sistem Informasi Manajemen Jalur Magang dengan menggunakan metode *Simple Agile Methodology* yang bertujuan untuk memberikan fasilitas mahasiswa dalam mencari perusahaan yang membuka lowongan magang khususnya bagi mahasiswa Program Studi Sarjana Informatika. Dengan menerapkan *Simple Agile Methodology* dalam pengembangan Sistem Informasi Manajemen Jalur Magang, pengembangan sistem menjadi lebih efektif dikarenakan standar kode yang digunakan telah ditentukan oleh *developer*, rincian tugas terstruktur sesuai dengan tahapan pengembangan sistem, dan kebutuhan sistem lebih efisien dengan menggunakan skala prioritas.

Kata Kunci— *Simple Agile Method, Magang, Aplikasi Web Magang, Basis Data*

I. PENDAHULUAN

Kegiatan magang merupakan salah satu bentuk agar mahasiswa bersiap terjun di dunia kerja. Selain sebagai bentuk pengembangan diri mahasiswa, kegiatan magang juga meningkatkan hubungan antara perguruan tinggi dengan perusahaan. Kehadiran sistem informasi di dunia pendidikan dapat memudahkan mahasiswa dalam mencari perusahaan yang membuka lowongan program magang [1]. Program magang merupakan salah satu penjaluran kelulusan mahasiswa pada tahun keempat. Kegiatan magang berjalan selama kurun waktu dua semester atau kurang lebih enam bulan. Mahasiswa yang memilih penjaluran kelulusan jalur magang wajib mengikuti segala kegiatan di perusahaan yang diajukan. Kegiatan yang dilakukan pada akhir masa magang yaitu membuat laporan aktivitas magang dan publikasi ilmiah atau skripsi [2].

Sistem Informasi Magang memiliki tujuan yakni memudahkan mahasiswa dalam pencarian tempat magang. Namun, terdapat kendala yang dihadapi mahasiswa dalam mencari tempat magang yakni harus mencari secara manual perusahaan yang membuka lowongan untuk mahasiswa magang. Selain itu, *website* Sekawan yang ada di Program Studi Sarjana Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia hanya memiliki fitur pendaftaran penjaluran kelulusan dan daftar Dosen Pembimbing Akademik. Oleh sebab itu, perlu dibuat Sistem Informasi Manajemen Jalur Magang untuk mahasiswa Prodi Sarjana Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia yang tidak hanya memiliki fitur pendaftaran penjaluran kelulusan dan daftar Dosen Pembimbing Akademik, tetapi terdapat fitur tambahan yakni pencarian perusahaan yang membuka lowongan kerja untuk mahasiswa magang.

Dalam pengembangan sistem yang akan dikembangkan, perlu adanya metode pengembangan sistem yang dilakukan secara bertahap sesuai kebutuhan. Di sisi lain, metode yang digunakan perlu memiliki karakteristik efektif, efisien, kepuasan *user* menjadi prioritas, dan minim risiko kegagalan. Dengan menggunakan metode tersebut, pengembangan sistem yang dikembangkan dapat dilakukan secara efektif dan efisien karena berfokus pada prioritas fitur yang akan dikembangkan.

Simple Agile Methodology adalah metode pengembangan perangkat lunak yang terinspirasi dari *Scrum Programming* dan *Extreme Programming* dengan implementasi yang lebih sederhana dan lebih fleksibel [3]. *Scrum Programming* termasuk dalam metode *Agile* dengan mengedepankan pada kolaborasi tim, hasil produk, serta proses iterasi agar terwujud hasil yang maksimal [4]. Sedangkan *Extreme Programming* salah satu bagian dalam *Agile Software Development Methodologies* untuk pengembangan perangkat lunak yang berfokus pada penyederhanaan proses sehingga selaras dengan prinsip nilai berupa Komunikasi (*Communication*), Kesederhanaan (*Simplicity*), Umpan Balik (*Feedback*), Keberanian (*Courage*), dan Menghormati (*Respect*) [5].

Simple Agile Methodology memiliki keunggulan dan solusi waktu dalam waktu yang lebih singkat. Pembuatan konsep yang biasanya membutuhkan waktu 3-4 minggu, dengan menerapkan *Simple Agile Methodology* dapat dilakukan dalam waktu tiga

hari saja [6]. Hal ini yang menjadi acuan penggunaan *Simple Agile Methodology* dalam mengembangkan Sistem Informasi Manajemen Jalur Magang untuk mahasiswa Program Studi Sarjana Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia. Berdasarkan permasalahan tersebut, *Simple Agile Methodology* akan diimplementasikan dalam pengembangan Sistem Informasi Manajemen Jalur Magang yang memudahkan mahasiswa dalam pencarian perusahaan yang membuka lowongan magang.

Rumusan masalah yang diangkat berdasarkan latar belakang dalam penelitian ini yakni bagaimana penerapan *Simple Agile Methodology* dalam pengembangan aplikasi *web*. Batasan masalah dalam penelitian ini adalah pengembangan proyek berfokus pada lingkup program studi Sarjana Informatika, dengan proyek yang dikembangkan Sistem Informasi Manajemen Jalur Magang dalam bentuk aplikasi *web* dan pengembangan proyek ini menggunakan metode *Simple Agile Methodology*. Berdasar pada rumusan masalah dan batasan masalah yang telah dibahas, tujuan penelitian ini adalah melakukan pengembangan Sistem Informasi Manajemen Jalur Magang untuk mahasiswa Program Studi Sarjana Informatika dengan menerapkan *Simple Agile Methodology*. Harapannya dalam penelitian ini bermanfaat bagi *developer* untuk menambah wawasan terkait *Simple Agile Methodology* dan bermanfaat untuk mahasiswa Program Studi Sarjana Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia khususnya mahasiswa yang memilih penjurusan kelulusan jalur magang dalam mencari perusahaan yang membuka lowongan magang.

II. KAJIAN PUSTAKA

A. Magang

Magang kerja merupakan salah satu kegiatan mahasiswa pada tahun ke empat di masa perkuliahan dan menjadi syarat kelulusan. Kegiatan magang kerja atau Praktek Kerja Lapangan (PKL) merupakan program dari perusahaan atau instansi tertentu sebagai bentuk agar mahasiswa mencapai keahlian tertentu [7]. Kegiatan magang yang dilaksanakan mahasiswa harus mengikuti seluruh aktivitas di perusahaan tempat mahasiswa mengajukan tempat magang. Magang kerja dilaksanakan dalam kurun waktu dua semester atau sekurang-kurangnya enam bulan. Pada akhir kegiatan magang, mahasiswa membuat luaran berupa laporan aktivitas magang dan publikasi ilmiah yang berupa skripsi [2].

Era digitalisasi saat ini sistem informasi diperlukan di berbagai bidang salah satunya bidang pendidikan. Keberadaan sistem informasi berpengaruh dalam meningkatkan kinerja, efisiensi, efektivitas, dan produktivitas dalam pengelolaan magang [8]. Perguruan tinggi sebagai instansi di bidang pendidikan harus memiliki manajemen kegiatan magang kerja agar pelaksanaan kegiatan magang dapat berjalan lancar. Sistem informasi menjadi pilihan sebagai kelancaran kegiatan magang [9]. Kegiatan magang menjadi salah satu bentuk agar mahasiswa bersiap terjun di dunia kerja. Selain itu, kegiatan magang juga meningkatkan hubungan antara perguruan tinggi dengan perusahaan. Dengan adanya sistem informasi, mahasiswa dapat

dengan mudah mencari perusahaan yang menerima program magang [1].

Sistem Informasi Magang merupakan sistem yang memberikan informasi terkait kegiatan magang. Mulai dari pencarian lowongan tempat magang, profil perusahaan, log harian mahasiswa, lokasi tempat magang, prosedur pengajuan magang, dan jadwal ujian mahasiswa setelah melaksanakan magang. Selain itu, sistem informasi magang dapat membantu pihak dosen pembimbing untuk mengetahui perusahaan mana mahasiswa melakukan kegiatan magang [10]. Sistem Informasi Magang dibutuhkan di lingkungan universitas guna membantu proses penyelenggaraan, pengelolaan, dan pelaporan kegiatan magang kerja untuk mahasiswa tingkat 3 atau semester 6 [11].

B. Agile Method

Metodologi *Agile* merupakan metode pengembangan yang muncul pada tahun 1980-an oleh dua orang Jepang yakni Hirotaka Takeuchi dan Ikujiro Nonaka dalam artikel "*The New Product Development Game*". Pada awalnya kedua tokoh ini memberi istilah *Scrum* yang menggambarkan perilaku bisnis yang fleksibel dalam formasi tim dan bekerja secara kolektif [12]. Metodologi *Agile* merupakan metode manajemen proyek yang siklus pengembangannya berjalan dengan cepat, atau dapat disebut juga *Sprint*. Dalam metode *Agile* sendiri membagi beberapa bagian-bagian kecil dalam proses pengembangannya yang berjalan cepat dikarenakan proses utamanya mengedepankan pada *design-code-test once day* [13].

Metodologi *Agile* merupakan metodologi yang revolusioner dibandingkan dengan metode sebelumnya. Lahirnya metodologi *Agile* ini dikarenakan metodologi terdahulu memiliki banyak alur yang mengakibatkan proses pengembangan tidak sesuai dengan tuntutan *user*. Dengan hadirnya metodologi *Agile*, para *developer* mendapat banyak keuntungan diantaranya meningkatkan rasio kepuasan pelanggan, dapat melakukan *review* pelanggan mengenai *software* yang telah dikembangkan secara berkala, mengurangi risiko kegagalan implementasi *software* dari segi non-teknis maupun nilai kerugian baik secara material atau immaterial tidak terlalu besar apabila terjadi kegagalan [14].

Metodologi *Agile* sendiri memiliki kelebihan yakni: 1) Kepuasan klien merupakan tujuan dan prioritas dalam pengembangan sistem, 2) Waktu pengembangan sistem lebih cepat, 3) Minim risiko kegagalan implementasi *software*, 4) Apabila terjadi kegagalan pengembangan sistem kerugian biaya relatif kecil [15]. Di sisi lain metodologi *Agile* memiliki kelemahan yakni: 1) Masih jarang dipraktekkan secara langsung, 2) Terlalu sering interaksi dengan *user*, 3) Sulit diaplikasikan pada pengembangan proyek yang besar, 4) Metode ini membutuhkan anggota tim yang terlatih, 5) Lemah dalam perencanaan, 6) Keterbatasan waktu karena kebutuhan *user* selalu berubah [16]. Namun demikian, metode *Agile* ini menjadi metode yang baik untuk diterapkan. Diharapkan dengan penerapan metode *Agile* ini dapat menjadi solusi dalam pengembangan perangkat lunak yang cepat dan efisien tanpa mengurangi kualitas perangkat lunak dan sesuai dengan kebutuhan pengguna.

C. Simple Agile Method (SAM)

SAM adalah metodologi *Agile* yang terinspirasi dari *Scrum Programming* dan *Extreme Programming (XP)* tetapi dengan implementasi yang lebih sederhana dan lebih fleksibel [3]. *Scrum Programming* termasuk dalam metode *Agile* yang menekankan pada kolaborasi tim, hasil produk, serta proses iterasi agar terwujud hasil yang maksimal [4]. Sedangkan *Extreme Programming* salah satu bagian dalam *Agile Software Development Methodologies* untuk pengembangan perangkat lunak yang berfokus pada penyederhanaan proses sehingga selaras dengan prinsip nilai berupa Komunikasi (*Communication*), Kesederhanaan (*Simplicity*), Umpan Balik (*Feedback*), Keberanian (*Courage*), dan Saling Menghormati (*Respect*) [5].

Menurut Akash Agarwal “saat ini bagaimana TI menjadi gesit dan berorientasi pada hasil dan mendorong klien mencari solusi yang sederhana” [6]. Sebuah perusahaan layanan TI yang berada di Bangalore, Sahaj Soft telah menerapkan *Simple Agile Methodology* pada pengembangannya dan teruji memiliki keunggulan dan solusi dalam waktu yang lebih singkat. Pembuatan konsep yang biasanya membutuhkan 3-4 minggu, dengan menerapkan *Simple Agile Methodology* dapat dilakukan hanya dalam tiga hari saja.

Simple Agile Methodology memiliki prinsip-prinsip antara lain: 1) Responsif terhadap perubahan yang artinya memaksimalkan kemampuan adaptasi dalam tim berupa perubahan yang cepat dalam pengembangan proyek, 2) Mengutamakan kesederhanaan karena kesederhanaan akan mempermudah dalam pengembangan proyek dengan prinsip “*Simple is Best*”, 3) Pengelolaan skala prioritas merupakan hal penting karena kebutuhan pelanggan dan teknis menjadi prioritas, 4) Kebebasan memilih tugas yang bermakna para *developer* bebas memilih tugas sesuai efektivitas agar tetap mempertahankan visi misi dalam proyek, 5) Komunikasi dan saling menghargai karena komunikasi merupakan salah satu kunci keberhasilan proyek dan menggunakan pendekatan konstruktif dalam pembahasan masalah, 6) Kepemilikan kode secara kolektif artinya setiap anggota tim bertanggung jawab atas seluruh kode yang berarti anggota tim diperbolehkan mengubah bagian kode manapun, 7) Standarisasi penulisan kode artinya adanya aturan format yang ditaati *developer* agar konsisten dalam penggunaan kode tersebut dan meminimalisir adanya *bug* selama pengembangan proyek, 8) Integrasi yang berkelanjutan, artinya rutin dalam proses integrasi segala perubahan kode, 9) Serahan yang berkelanjutan artinya rutin dalam penyerahan *progress* yang memungkinkan mendapat *feedback* dari *user* [3].

Simple Agile Methodology memiliki beberapa aturan (*Role*) antara lain: *Project Manager* ada peran bersama antara Pimpinan Teknis (*Tech Lead*) dengan Pimpinan Hubungan Pelanggan (*Tech Lead + C.R.L.*), *Tech Lead* bekerja untuk menentukan prioritas tugas dan juga mengembangkan bagian kode sensitif, mempertahankan koherensi arsitektur proyek, dan memfasilitasi dialog dalam tim *developer*, *C.R.L. (Customer Relationship Lead)* bekerja dengan *Tech Lead* untuk menentukan prioritas tugas. *C.R.L.* juga berkomunikasi dan mengorganisir antara klien dengan *Test Team*.

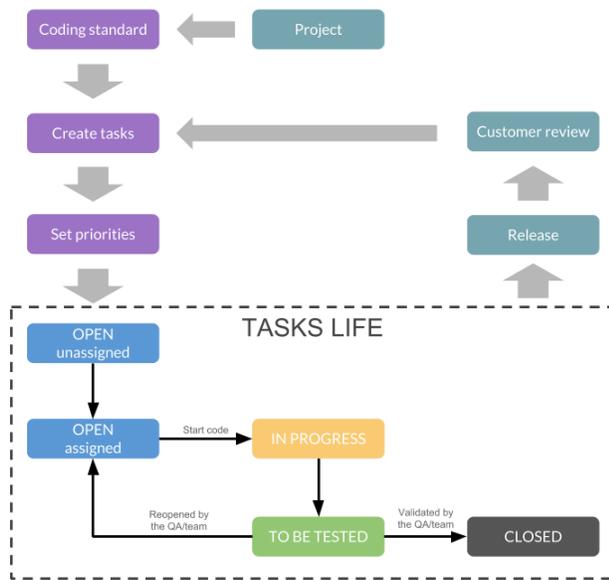
D. Alat Bantu Agile

Metode *Agile* memiliki beberapa alat bantu yang dapat digunakan dalam proyek. *Project Management Tools* merupakan alat yang membantu individu atau tim dalam mengatur dan mengelola proyek dan disesuaikan dengan kebutuhan tim dengan ukuran berbeda dan dengan tujuan berbeda. *Project Management Tools* mencakup fitur diantaranya: *Planning/Scheduling* yang berarti memungkinkan untuk merencanakan pekerjaan di dalam satu tempat yang terdiri dari tugas, subtugas, folder, *template*, alur kerja, dan kalender, *Collaboration* bermakna komunikasi tidak hanya menggunakan *email*, tetapi yang dapat membangun dan menciptakan cara yang lebih baik selama pengembangan proyek, *Documentation* bertujuan menghindari terjadinya file yang hilang dan memungkinkan untuk mengedit, membuat versi, dan menyimpan file, *Evaluation* bertujuan untuk menilai pekerjaan yang telah dilakukan agar menciptakan produktivitas dan pertumbuhan pada pengembangan seterusnya [17].

Project Management Tools sendiri memerlukan fitur yang dapat memudahkan *project manager* dan *developer* dalam melacak proyek dan bagaimana setiap proyek saling berinteraksi satu sama lain. Alat bantu dalam *Project Management Tools* yang digunakan memiliki karakteristik antara lain: Bersifat fleksibel baik dari sisi penggunaan maupun tampilan, Responsif terhadap perubahan dan memaksimalkan kemampuan adaptasi cepat, Multifungsi yang memudahkan dalam manajemen proyek, Kemampuan dalam mengelola skala prioritas, Kebebasan memilih tugas untuk menciptakan efektivitas para *developer*, Kemudahan berkomunikasi. Beberapa karakteristik tersebut menciptakan pilihan *tools* yang akan digunakan selama pengembangan proyek ini. *Tools* yang akan digunakan dalam manajemen proyek adalah Trello dan *tools* dalam pengembangan proyek menggunakan Github dan Gitlab.

III. METODOLOGI

Siklus hidup *Simple Agile Methodology* memiliki tahapan antara lain: 1) Menentukan standar kode, 2) Membuat rincian tugas, 3) Menentukan skala prioritas. Gambar 1 menunjukkan alur siklus hidup (*Lifecycle*) *Simple Agile Methodology*. Penjelasan detail alur *Lifecycle* dalam *Simple Agile Methodology* sebagai berikut: 1) Menentukan standar kode dan organisasi proyek yang fleksibel agar memudahkan pemahaman antar *developer*, 2) Membuat rincian tugas dengan fitur utama dan hindari penambahan fitur yang terlalu detail agar bisa fokus sepanjang pengembangan, 3) Menentukan skala prioritas. Apabila sudah ditentukan tugas prioritas, setiap *developer* memilih tugasnya masing-masing, Saat tugas sudah selesai, dilanjutkan pengujian oleh tim lain. Pengiriman setiap rilis dilakukan sesering mungkin kepada klien untuk mendapat umpan balik (*feedback*) dari pelanggan. Apabila mendapat tugas atau terdapat *bug* baru maka siklus kembali ke poin menentukan skala prioritas.



Gambar 1. Lifecycle dalam Simple Agile Methodology

IV. HASIL DAN PEMBAHASAN

Berikut adalah bagian hasil dari pengembangan proyek Sistem Informasi Manajemen Jalur Magang untuk Mahasiswa Program Studi Sarjana Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia yang masih dalam tahap pengembangan. Sesuai dengan metodologi *Simple Agile Method*, hasil dari setiap tahapan dijelaskan sebagai berikut:

A. Menentukan Standar Kode

Standar kode atau *coding-standard* merupakan hal krusial atau penting dalam pengembangan perangkat lunak. Standar kode atau *coding-standard* adalah seperangkat aturan dan praktik yang harus dipatuhi *developer* untuk memastikan kode tersebut mudah dipahami, mudah dipelihara, mudah diperluas, serta mengurangi risiko *bug* [18]. Dengan menggunakan *framework* laravel pada pengembangan sistem yang mempunyai standar kode pemrograman PHP, maka standar kode yang digunakan adalah PSR-1 (*PHP Standards Recommendation*)-1.

B. Membuat Rincian Tugas

Dalam membuat rincian tugas (*task*) selama pengembangan proyek, diperoleh dari berbagai aktivitas. Aktivitas yang dilakukan dalam pengembangan proyek adalah membuat sebuah *Work Breakdown Structure*. Dalam pembuatan rincian tugas ini terdapat tiga poin utama yakni: 1) Analisis yang berisi Analisis Proses Bisnis, Analisis *User Stories*, Analisis Fitur, dan Analisis Task. 2) Perancangan yang berisi Perancangan Antarmuka, Perancangan Antarmuka Setiap Halaman *User Stories*, Perancangan Data, dan Perancangan Panduan Implementasi. 3) Implementasi yang berisi Implementasi Basis Data, Implementasi Aplikasi, dan Implementasi Setiap Halaman *User Stories*.

User Stories merupakan bagian dari metode agile yang bertujuan untuk mengetahui kebutuhan dari perspektif

pengguna. Alur proses bisnis *User Stories* antara lain: 1) Studi Kelayakan (*Feasibility Study*) yang dilakukan untuk menilai apakah perangkat lunak yang dibuat berguna atau tidak, 2) Elisitasi dan Analisis Kebutuhan (*Requirements Elicitation and Analysis*) untuk menemukan kebutuhan sistem, 3) Spesifikasi Kebutuhan (*Requirements Spesification*) untuk menentukan standarisasi kebutuhan, 4) Validasi Kebutuhan (*Requirements Validation*) untuk memastikan apakah sudah sesuai dengan kebutuhan dan keinginan pengguna. Saat rincian tugas (*task*) sudah dibuat, langkah selanjutnya adalah dimasukkan ke Trello dan di-assign ke *developer* untuk pengembangan proyek.

C. Menentukan Skala Prioritas

Pada tahap penentuan skala prioritas dibagi menjadi dua yakni kriteria berdasarkan kepentingan dan kriteria berdasarkan urgensi. Pada bagian kriteria berdasarkan kepentingan memiliki dua elemen yakni *Important* dan *Not Important*. Pada elemen *Important* mengacu pada proses bisnis *User Stories* yang bermakna penting dan harus ada dalam proses bisnis. Pada elemen *Not Important* mengacu pada proses bisnis yang bersifat tambahan dalam proses bisnis. Sedangkan pada bagian kriteria berdasarkan urgensi memiliki tiga elemen yakni *High*, *Medium*, dan *Low*. Kriteria berdasarkan urgensi mengacu pada *timeline* dalam pengembangan proyek. Pada elemen *High* bermakna *task* harus dikerjakan terlebih dahulu, *Medium* bermakna *task* akan dikerjakan apabila *task* pada elemen *High* sudah dikerjakan, dan *Low* bermakna *task* akan dikerjakan apabila *task* pada elemen *Medium* sudah dikerjakan. Saat tahap penentuan skala prioritas sudah dikerjakan, tahap berikutnya adalah setiap *developer* memilih *task* mana saja yang akan dikerjakan dan di dalam Trello diberi label berdasarkan klasifikasi yang sudah dibuat untuk memudahkan *developer* selama pengembangan proyek.

Berdasarkan *User Stories* yang diterapkan menghasilkan 17 daftar *User Stories* yang terbagi menjadi dua kriteria yakni kriteria berdasarkan kepentingan dan kriteria berdasarkan urgensi. Pada bagian kriteria berdasarkan kepentingan berisi 27 *Important* dan 15 *Not important*. Pada bagian kriteria berdasarkan urgensi berisi 13 *High*, 25 *Medium*, dan 3 *Low*.

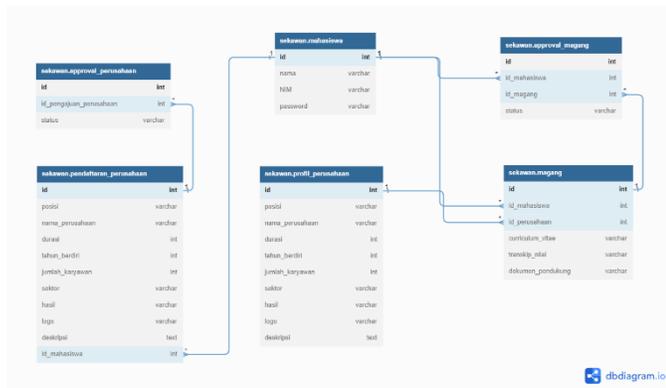
Tabel 1. Tabel Skala Prioritas

Kode	Task	Kategori	Kepentingan	Urgensi
2.1	Merancang tabel magang mahasiswa	Desain Basis Data	Important	High
2.2	Merancang tabel pendaftaran perusahaan magang	Desain Basis Data	Important	High
2.3	Merancang profil perusahaan magang	Desain basis data	Important	High
2.9	Merancang tabel approval perusahaan magang	Desain basis data	Important	High
2.12	Merancang tabel approval mahasiswa magang	Desain basis data	Important	High

Berdasarkan Tabel I, dapat disimpulkan skala prioritas didapatkan berdasarkan hasil dari proses *User Stories*. Tabel 1 didapat dari hasil diskusi *developer* dan *Project Manager* dalam menentukan *task* mana saja yang dikerjakan dalam pengembangan proyek dengan kriteria berdasarkan kepentingan pada elemen *Important* dan kriteria berdasarkan urgensi pada elemen *High*. Pada Tabel 1 menunjukkan yang akan dikerjakan dalam iterasi 1 *Agile*.

D. Memilih Tugas Developer

Pada tahapan pemilihan tugas yang akan dikerjakan, *developer* telah menentukan *task* yang dikerjakan selama pengembangan sistem yang berdasar pada alur *Lifecycle Simple Agile Methodology* yaitu: 1) Desain Antarmuka, Desain Basis Data, dan Implementasi Antarmuka. *Task* yang telah ditentukan kemudian di-assign ke Trello untuk menunjang *developer* dalam pengembangan sistem. Setelah *task* telah di-assign ke Trello, tahap selanjutnya adalah menentukan alat bantu selama pengembangan sistem antara lain: 1) Laravel sebagai *framework* untuk menunjang pemrograman standar kode yang digunakan yaitu PHP, 2) MYSQL sebagai *database server*, 3) PHPMyAdmin sebagai *tools* untuk mengelola database berbasis web. Hasil dari desain basis data terdapat pada Gambar 2.



Gambar 2. Rancangan *database*

Berdasarkan rancangan *database* pada Gambar 2, terdapat enam tabel yakni:

- Tabel mahasiswa berisi nama mahasiswa, NIM mahasiswa, dan *password* mahasiswa.
- Tabel magang berisi *id_mahasiswa* sebagai *foreign key* untuk berelasi dengan Tabel mahasiswa, *id_perusahaan* sebagai *foreign key* untuk berelasi dengan Tabel profil perusahaan, *curriculum_vitae*, *transkrip_nilai*, dan *dokumen_pendukung*.
- Tabel approval magang berisi *id_mahasiswa* sebagai *foreign key* untuk berelasi dengan Tabel mahasiswa, *id_magang* sebagai *foreign key* untuk berelasi dengan Tabel magang. Tabel approval magang bertujuan untuk menguji kelayakan mahasiswa dalam melaksanakan magang pada perusahaan yang akan dituju.

- Tabel profil perusahaan berisi posisi, nama_perusahaan, durasi, tahun_berdiri, jumlah_karyawan, sektor, hasil, logo, dan deskripsi.
- Tabel pendaftaran perusahaan berisi *id_mahasiswa* sebagai *foreign key* untuk berelasi dengan Tabel mahasiswa, posisi, nama_perusahaan, durasi, tahun_berdiri, jumlah_karyawan, sektor, hasil, logo, dan deskripsi.
- Tabel approval perusahaan berisi *id_pengajuan_perusahaan* sebagai *foreign key* untuk berelasi dengan Tabel pendaftaran perusahaan, dan status. Tabel approval perusahaan bertujuan untuk menguji kelayakan perusahaan yang diajukan oleh mahasiswa.

E. Melanjutkan Pengujian

Terdapat lima langkah dalam *Task Life Simple Agile Method* yakni: 1) *Open* yakni status pertama pada *task* setelah penentuan skala prioritas, 2) *Assigned* yakni status kedua yang berisi *task* diubah menjadi *prototype*, 3) *In Progress* adalah status ketiga yang mana *developer* melakukan *coding*, 4) *Tested* yakni status keempat yang berisi ujicoba sistem yang dilakukan oleh manajer, 5) *Closed* yakni status kelima yang mana sistem yang sudah diuji coba sudah siap untuk dirilis dan diuji coba oleh *user*.

Dari lima status dalam *Task Life Simple Agile Method*, proses pengembangan sistem masih dalam tahapan *In Progress* dikarenakan tahap implementasi *coding* sedang dalam proses pengembangan. Oleh karena itu akan dilanjutkan ke tahapan *Tested* yang berisi kegiatan ujicoba sistem yang akan dilakukan oleh manajer.

Dengan menggunakan metode *Simple Agile Methodolgy*, pengembangan sistem menjadi lebih efektif karena standar kode yang digunakan telah ditentukan, rincian tugas terstruktur sesuai dengan tahap pengembangan sistem, dan kebutuhan sistem lebih efisien dengan menggunakan skala prioritas. Terdapat sebuah penyedia aplikasi monitoring jaringan di Indonesia yaitu Netmonk yang menggunakan *Simple Agile Method* sebagai metode yang diterapkan di perusahaan tersebut dimana perusahaan tersebut mengedepankan prinsip bahwa proyek dan produk dapat berjalan beriringan sesuai prinsip dari *Simple Agile Method* yaitu kebutuhan *user* merupakan prioritas dan kemampuan adaptasi yang responsif terhadap perubahan yang cepat.

V. KESIMPULAN

Kesimpulan dari pengembangan proyek ini adalah telah berhasil dikerjakan Sistem Informasi Manajemen Jalur Magang sampai pada tahapan desain basis data yang akan digunakan. Pengembangan sistem dilakukan pada kriteria kepentingan *Important* dan kriteria urgensi *High*. Pengembangan proyek ini masih dalam tahapan *In Progress* pada *Task Life Simple Agile Methodology* dikarenakan tahap implementasi *coding* sedang dalam proses pengembangan. Oleh karena itu, tahapan selanjutnya adalah tahapan *Tested* yang berisi kegiatan ujicoba sistem yang akan dilakukan oleh manajer. Dengan menerapkan *Simple Agile Methodology* dalam pengembangan Sistem Informasi Manajemen Jalur Magang, pengembangan sistem

menjadi lebih efektif dikarenakan standar kode yang digunakan telah ditentukan oleh *developer*, rincian tugas terstruktur sesuai dengan tahapan pengembangan sistem, dan kebutuhan sistem lebih efisien dengan menggunakan skala prioritas.

Sebagai pekerjaan ke depan, selanjutnya akan dikerjakan *task* yang belum selesai agar tercipta Sistem Informasi Manajemen Jalur Magang yang berkualitas. Diharapkan sistem informasi manajemen tersebut dapat memudahkan mahasiswa yang memilih jalur kelulusan penjurusan magang khususnya mahasiswa Program Studi Sarjana Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.

REFERENSI

- [1] E. Purnomo, Y. R. Pratama, R. Rezani, J. Saputra, and A. G. Abdullah, "Design and implementation of web-based internship information system at vocational school Design and implementation of web-based internship information system at vocational school," 2018, doi: 10.1088/1757-899X/434/1/012301.
- [2] Jurusan Informatika UII, "Penjurusan Tugas Akhir," *Jurusan Informatika UII*, 2021. <https://informatics.uii.ac.id/sarjana/kurikulum-2020/jalur-tahun-keempat/>.
- [3] uBugtrack, "Simple Agile Methodology S.A.M. Programming," *uBugtrack*, 2021. https://ubugtrack.com/?page=simple_agile_methodology.
- [4] Aprilia D.L., "Metodologi Scrum," *Binus University*, 2021. <https://socs.binus.ac.id/2019/12/23/metodologi-scrum/> (accessed Dec. 22, 2021).
- [5] T. Gumelar, R. Astuti, and A. T. Sunarni, "Sistem Penjualan Online Dengan Metode Extreme Programming," *J. Telemat.*, vol. 9, no. 2, pp. 87–90, 2017.
- [6] ForbesIndia, "SimpleAgile: A Paradigm Shift for the Agile Methodology," *ForbesIndia*, 2021. <https://www.forbesindia.com/article/brand-connect/simpleagile-a-paradigm-shift-for-the-agile-methodology/53811/1> (accessed Dec. 22, 2021).
- [7] A. Lestari *et al.*, "SISTEM INFORMASI MAGANG BERBASIS WEBSITE PADA DINAS KESEHATAN PROVINSI JAWA TENGAH," vol. 4, no. Sens 4, pp. 95–100, 2019.
- [8] M. R. Baharuddin and U. Ulfah, "Pengembangan Sistem Informasi Manajemen Pelaksanaan Magang FKIP UNCP," *J. Literasi Digit.*, vol. 1, no. 1, pp. 34–41, 2021.
- [9] W. Hardyanto, A. Purwinarko, I. M. Sudana, and E. Suprptono, "Model Development of Management Information System of Internship," no. January, 2018, doi: 10.2991/iset-18.2018.41.
- [10] M. I. M. Wihelvi, "Jurnal wira pratika vol.7 - juli 2020," vol. 7, pp. 68–93, 2020.
- [11] A. Purwitasari, *Sistem Informasi Pengelolaan Magang guna Mempermudah Proses Administrasi (Studi Kasus: Jurusan Hama dan Penyakit Tumbuhan Universitas Brawijaya)*. Malang, 2015.
- [12] J. C. Goodpasture, *Project Management the Agile way: Making it Work in the Enterprise*. 2016.
- [13] A. Dimas, A. Rizki, and S. Hyde, "Inovasi Manajemen Proyek I-Learning Menggunakan Metode Agile Scrum Inovasi Manajemen Proyek I-Learning Menggunakan Metode Agile Scrum," *J. Nas. Inform. dan Teknol. Jar.*, vol. Vol.5, 2020.
- [14] U. Ependi, P. Studi, T. Informatikauniversitas, and B. Darma, "PENGEMBANGAN E-TRACE ALUMNI DENGAN MENGGUNAKAN PENDEKATAN METODE AGILE," vol. 2012, no. semnasIF, pp. 237–244, 2012.
- [15] Y. I. Chandra, "Perancangan Aplikasi Resep Makanan Tradisional Indonesia Menggunakan Pendekatan Agile Process Dengan Model Extreme Programming Berbasis Android.," 2016.
- [16] Fathiah, R. Zulaida, and N. Hayi, "Perbandingan Metodology Klasik dan Agile dalam Pengembangan Sistem Informasi," 2014.
- [17] wrike, "PROJECT MANAGEMENT GUIDE," *wrike*, 2022. <https://www.wrike.com/project-management-guide/faq/what-are-project-management-tools/> (accessed Jan. 11, 2022).
- [18] Adventures with Agile, "Coding Standards," *Adventures with Agile*, 2022. <https://www.adventureswithagile.com/2014/05/28/coding-standards/> (accessed Jan. 19, 2022).
- [19] Jsbinary, "Memahami Pentingnya Kualitas Coding dan Standar Coding," *jsbinary*, 2022. <https://www.binaryjs.com/memahami-pentingnya-kualitas-coding-dan-standar-coding/> (accessed Jan. 19, 2022).