

BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi yang pesat membuat perusahaan dituntut untuk memiliki sebuah sistem yang dapat membantu kebutuhan bisnisnya, hal itu juga berlaku bagi lingkungan akademik seperti perguruan tinggi. Dalam dunia penelitian yang umumnya sangat penting dalam perguruan tinggi dibutuhkan kolaborasi yang baik antara dosen dan mahasiswa, namun seringkali perekrutan proyek penelitian dilakukan dengan cara lama seperti pengumuman lisan atau pengumuman pada papan. Cara-cara tersebut mengurangi potensi jangkauan yang bisa dicapai untuk mencari anggota tim penelitian. Oleh karena itu dibutuhkannya sistem manajemen yang dapat menangani masalah-masalah yang telah disebutkan. UMMRN merupakan sebuah sistem manajemen pakar yang memudahkan seorang peneliti (dosen) untuk melakukan perekrutan mahasiswa terhadap *project* penelitian yang mereka miliki yang akan dikembangkan menggunakan platform *Low-Code*. Sistem UMMRN ini nantinya akan memudahkan untuk merekrut anggota penelitian dan memudahkan untuk memantau jalannya sebuah proyek penelitian. UMMRN juga memiliki fitur komunitas yang dimana usernya dapat bergabung dengan komunitas yang mereka minati sehingga mereka akan lebih bersemangat ketika berinteraksi dan membahas suatu topik dengan orang-orang yang memiliki minat yang sama. Adapun stakeholder dari pengembangan sistem UMMRN ini dosen selaku pengguna utama dan pemilik dari proyek penelitian yang memanfaatkan fitur pembuatan proyek penelitian, perekrutan anggota penelitian, mengelola komunitas. Kemudian mahasiswa sebagai pengguna utama dan anggota tim penelitian riset dan anggota dari komunitas.

Pengembangan perangkat lunak seringkali menemukan kegagalan[1]. Menurut studi kasus yang dilakukan oleh *The Standish Group* pada tahun 2020 pengembangan proyek perangkat lunak memiliki tingkat kegagalan sebesar 66%. Hal ini menandakan bahwa proyek pengembangan perangkat lunak memiliki angka kegagalan yang cukup tinggi.

Untuk menjamin keberhasilan suatu pengembangan proyek perangkat lunak, diperlukan perencanaan yang matang sebelum pengembangan tersebut dilakukan. Perencanaan yang tidak tepat dapat menyebabkan keterlambatan dalam pengerjaan yang akan mempengaruhi terhadap biaya pengembangan menjadi semakin bertambah hingga akhirnya kegagalan dari pengembangan tersebut. Untuk mengatasi permasalahan tersebut dibuatlah banyak metode estimasi waktu dan biaya seperti contohnya *Function Point Analysis* (FPA) dan COCOMO (*Constructive Cost Model*).

Pengembangan perangkat lunak umumnya dilakukan dengan cara manual yaitu pengerjaan coding dari awal oleh tim pengembang. Tetapi perkembangan teknologi yang pesat baru-baru ini menghasilkan beberapa platform baru dalam pengembangan proyek perangkat lunak, yaitu dengan munculnya *Low-Code Development Platform* (LCDP). Seperti pada namanya yaitu "*Low-Code*" yang berarti minim/sedikit dalam penulisan kode, LCDP ini hanya membutuhkan pendekatan antarmuka visual dan konfigurasi *drag-and-drop*.

LCDP ini menjadi salah satu metode pengembangan perangkat lunak yang populer bagi perusahaan atau individu karena menawarkan efisiensi tinggi yang membuat pengerjaan suatu proyek perangkat lunak menjadi lebih cepat, membuat biaya pengembangan semakin murah, dan tidak membutuhkan keterampilan programming yang tinggi yang berarti pengguna non-teknis atau bisa disebut sebagai *Citizen Developers* bisa aktif ikut serta dalam pengembangan sebuah proyek perangkat lunak[2]. Dengan populernya pengembangan proyek pada platform *Low-Code* ini diperkirakan pasar untuk LCDP akan tumbuh sebesar USD 45.5 miliar pada 2025 ini[3].

Meskipun LCDP menawarkan kemudahan dalam penggunaannya, hal terpenting dalam pengembangan proyek perangkat lunak tetaplah estimasi dari biaya dan waktu. Kelemahan dari platform *Low-Code* ini adalah model estimasi tradisional seperti COCOMO tidak dapat diterapkan secara langsung pada LCDP karena ketergantungan model perhitungan tersebut pada kode yang biasanya digunakan dalam pengembangan tradisional. COCOMO akan mendapat hasil yang

akurat ketika memiliki jumlah *Lines of Code* yang jelas tetapi hal itu membuat metode tersebut membutuhkan banyak data dari proyek-proyek perangkat lunak sebelumnya sehingga membuat metode COCOMO ini tidak cocok digunakan di beberapa proyek[4] terutama pada proyek yang menggunakan platform *Low-Code* dalam model pengembangannya. Sedangkan *Function Point Analysis* merupakan metode estimasi proyek yang pendekatannya dilakukan dengan cara mengukur fungsionalitas pada sistem dan tidak membutuhkan LOC untuk perhitungan estimasinya. FPA juga merupakan metode estimasi yang dapat digunakan di awal pengembangan suatu sistem karena pendekatannya yang tidak memerlukan LOC sehingga memudahkan untuk menghitung estimasi pada proyek cepat seperti proyek dengan model *prototype*. FPA merupakan pendekatan fleksibel tidak bergantung pada bahasa pemrograman[5] sehingga membuatnya cocok digunakan sebagai metode estimasi sistem yang dikembangkan pada platform *Low-Code* dibandingkan dengan metode tradisional seperti COCOMO yang memerlukan LOC untuk perhitungannya.

Dari permasalahan diatas maka penelitian ini bertujuan untuk mengembangkan model estimasi biaya dan waktu pada proyek perangkat lunak pada platform *Low-Code* menggunakan metode *Function Point Analysis* (FPA).

1.2. Rumusan masalah

Berdasarkan latar belakang diatas rumusan masalah yang akan dibahas dalam penelitian ini adalah sebagai berikut :

1. Bagaimana cara menggunakan metode *Function Point Analysis* pada platform *Low-Code*.
2. Bagaimana hasil estimasi proyek perangkat lunak menggunakan metode *Function Point Analysis* pada platform *Low-Code*.

1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Mengetahui cara menggunakan metode *Function Point Analysis* pada platform *Low-Code*
2. Mendapat hasil estimasi proyek perangkat lunak pada platform *Low-Code*.

1.4. Cakupan masalah

Supaya pokok pembahasan dalam penelitian ini tidak meluas dan tetap fokus, maka diperlukan cakupan masalah sebagai berikut:

1. Penelitian hanya menggunakan platform LCDP seperti Mendix dan Appian.
2. Hanya menggunakan metode *Function Point Analysis* sebagai pendekatan estimasi.

