

BAB 1 PENDAHULUAN

1.1. Latar Belakang

User Acceptance Test atau yang sering disingkat menjadi UAT merupakan langkah akhir yang sangat penting karena bertugas untuk memastikan bahwa perangkat lunak sesuai dengan perencanaan *requirement* di awal proyek [1]. UAT dilakukan dengan melibatkan pengguna akhir dengan tujuan untuk memberikan gambaran nyata dari alur sistem [2]. Dari data laporan CHAOS, ditemukan bahwa dua faktor utama kegagalan proyek yaitu *requirement* yang tidak memadai (13.1 %) dan kurangnya keterlibatan pengguna akhir (12.4 %). Sementara itu, faktor keberhasilan proyek terbesar menurut responden (15.9 %) adalah dengan melibatkan pengguna akhir [1]. Namun, pelaksanaan UAT secara tradisional ini menimbulkan berbagai tantangan dari segi alokasi waktu, penggunaan sumber daya yang signifikan, serta sulitnya merekrut pengguna yang sesuai dengan target pengguna akhir aplikasi [1][2].

Salah satu gagasan yang berpotensi menjadi solusi adalah *Crowdsourcing Software Engineering* (CSE). CSE dilakukan dengan meng-*outsource* tugas-tugas kepada sekelompok besar individu melalui sistem panggilan terbuka [3]. Penerapan CSE memberikan manfaat seperti pengurangan biaya pengembangan, peningkatan kualitas solusi, dan penggunaan sumber daya yang optimal [3]. Dalam bidang pengujian perangkat lunak, CSE dikenal dengan istilah *Crowdsourced Software Testing* (CST). Seperti namanya dan definisi CSE, CST dapat disimpulkan sebagai penerapan teknik *crowdsourcing* dalam bidang pengujian perangkat lunak [4]. CST telah diterapkan dalam berbagai tipe pengujian, yaitu *usability testing*, *functional testing*, *compatibility testing*, dan *performance testing* [4]. Dengan tersedianya sumber daya yang terbatas, memanfaatkan *crowdsourcing* memungkinkan pengujian dilakukan oleh partisipan eksternal tanpa memerlukan tim internal yang besar. Hal ini tidak hanya membantu mendeteksi *defect* dan

memperluas keterlibatan pengguna akhir, tetapi juga berpotensi mengurangi biaya pelaksanaan pengujian [4]. Dengan demikian, CST berpotensi mengatasi hambatan UAT tradisional terkait biaya, ketersediaan partisipan, dan keterwakilan *end-user*.

Meskipun CST dapat mengurangi hambatan tersebut, pelaksanaannya masih dapat terkendala sinkronisasi waktu antara penguji dan penyelenggara. Untuk mengatasi hal ini, dapat diterapkan pendekatan asinkron sebagaimana pada metode *Asynchronous Remote Usability Testing* (ARUT). Dalam ARUT, pengguna dan evaluator dipisahkan baik secara waktu maupun lokasi, mengurangi kebutuhan fasilitator dan hambatan logistik selama proses pengujian [5][6]. Metode ini juga memungkinkan pengumpulan data dari partisipan yang beragam dengan tetap menjaga efisiensi waktu dan biaya [6]. Keberhasilan tersebut membuka peluang untuk menerapkan pendekatan asinkron pada CST dalam konteks UAT, guna menciptakan pengujian yang lebih fleksibel dan tetap mencerminkan pengalaman nyata pengguna akhir.

Dengan memperhatikan tantangan yang dihadapi UAT tradisional, seperti keterbatasan waktu, tingginya penggunaan sumber daya, dan sulitnya merekrut partisipan yang sesuai, penerapan CST berpotensi menjadi solusi dengan memanfaatkan partisipan eksternal secara terbuka. Hal ini dapat mengurangi ketergantungan pada tim internal berskala besar, menghilangkan kebutuhan fasilitas fisik untuk pengujian tatap muka, serta menekan biaya pelaksanaan melalui penggunaan infrastruktur daring yang memungkinkan partisipan bekerja dari lokasi masing-masing. Selain itu, CST juga dapat meningkatkan keterlibatan pengguna akhir, sebagaimana telah menjadi salah satu manfaat utama dari CSE. Pendekatan asinkron melengkapi CST dengan mengatasi hambatan sinkronisasi waktu dan logistik. Meskipun CST telah diterapkan dalam berbagai jenis pengujian perangkat lunak, penelitian sebelumnya belum secara spesifik mengeksplorasi penerapannya dalam konteks UAT. Oleh karena itu, sebuah pendekatan berbasis CST yang menempatkan pengguna akhir sebagai penguji utama, dengan penerapan

asinkron tanpa fasilitator, dapat menjadi solusi yang menjanjikan. *Framework* ini diharapkan mampu mengatasi keterbatasan UAT tradisional dengan memanfaatkan fleksibilitas CST untuk menghasilkan pengujian yang lebih mencerminkan pengalaman pengguna akhir. Tentunya dengan mempertimbangkan bagaimana hasil dari pengujian UAT oleh *end-user* dalam *framework crowdsourcing* dibandingkan dengan pengujian UAT konvensional.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, berikut adalah rumusan masalah yang akan menjadi fokus penelitian ini:

- Bagaimana pendekatan *crowdsourcing* dapat diadaptasi untuk mendukung pelaksanaan *User Acceptance Testing* (UAT) secara asinkron dan tanpa fasilitator?
- Bagaimana hasil dari pengujian UAT oleh *end-user* dalam *framework crowdsourcing* dibandingkan dengan pengujian UAT konvensional?

1.3. Tujuan dan Manfaat

Penelitian ini bertujuan untuk mengembangkan dan mengevaluasi *framework* UAT berbasis *crowdsourcing* yang dilaksanakan secara asinkron. *Framework* ini diharapkan dapat menjadi alternatif praktik UAT yang efektif dan tetap mempertahankan kualitas hasil pengujian. Keberhasilan *framework* akan diukur melalui dua aspek utama: kemampuan deteksi *bug* dan akurasi hasil pengujian. Evaluasi dilakukan dengan membandingkan performanya terhadap UAT konvensional melalui studi kasus pada aplikasi manajemen inventaris.

1.4. Batasan Masalah

Penelitian ini memusatkan perhatian pada pengujian aplikasi sederhana yang memiliki fungsi dasar dan antarmuka yang intuitif, seperti aplikasi manajemen inventaris. Dalam pelaksanaannya, partisipan pengujian berjumlah sepuluh

orang, jumlah yang dianggap memadai untuk mengidentifikasi sebagian besar permasalahan utama pada jenis aplikasi tersebut.

Lingkup pengujian difokuskan pada evaluasi hasil penerapan proses UAT berbasis *crowdsourcing*. Untuk mendukung proses ini, skenario uji dan *test case* telah disiapkan sebelumnya oleh tim pengembang. Oleh karena itu, penelitian ini tidak mencakup tahapan perancangan atau penyusunan *test case*, melainkan lebih menitikberatkan pada pelaksanaan dan efektivitas pengujian dengan pendekatan *crowdsourcing*.

1.5. Jadwal Pelaksanaan

Berisi jadwal pelaksanaan pengerjaan Tugas Akhir. Perlu ditetapkan beberapa *milestone* untuk menentukan pencapaian pekerjaan.

Jadwal pelaksanaan pengerjaan Tugas Akhir beserta *milestone* setiap tahap dapat dilihat pada Tabel 1.1, yang memuat rincian tahapan pekerjaan dari studi literatur hingga penyusunan laporan akhir.

Tabel 1.1. Jadwal pelaksanaan tugas akhir.

No.	Deskripsi Tahapan	Bulan 1	Bulan 2	Bulan 3	Bulan 4	Bulan 5	Bulan 6
1	Studi Literatur						
2	Perancangan Framework						
3	Implementasi Rancangan Pada Platform Website						
4	Penerapan Studi Kasus UAT Konvensional dan UAT Crowdsourcing						
5	Dokumentasi, Analisis Temuan, Serta Kesimpulan dan Saran.						
6	Penyusunan Laporan/Buku TA						