



PENGEMBANGAN APLIKASI ESTIMASI WAKTU DAN BIAYA DENGAN METODE USE CASE POINT YANG DIINTEGRASIKAN PADA SCRUM PROJECT MANAGMENT

¹Sarah Nurul Qudsiyah(12018026),²Herman Yuliansyah(0512078304)

1,2Program Studi Teknik Informatika Universitas Ahmad Dahlan

Jl. Prof. Dr. Soepomo, Janturan, Yogyakarta 55164

1E-mail : sarahnurulq@gmail.com

2E-mail : herman.yuliansyah@tif.uad.ac.id

Abstrak

Manajemen merupakan suatu proses atau orang yang bertanggung jawab untuk merencanakan, mendapatkan, mengendalikan, mengarahkan, memastikan, memandu, dan mengatur penggunaan sumber daya yang diperlukan dalam mencapai tujuan dari suatu bisnis secara tepat waktu. Manajemen tugas proyek (MTP) di Program Studi Teknik Informatika Universitas Ahmad Dahlan Yogyakarta dimana penyelenggaraan MTP bertujuan untuk membekali kemampuan mahasiswa untuk mengelola proyek dalam lingkup bidang Teknologi Informasi serta melatih kemampuan kerja dalam tim.

Metodologi yang digunakan dalam penelitian ini yaitu dengan melakukan pengumpulan data yang terkait topik penelitian melalui Studi Literatur dan wawancara. Sistem ini diuji dengan 2 metode yaitu pengujian Black-box test dilakukan oleh Merlinda Wibowo, S.T dengan hasil sudah sesuai dengan kebutuhan proses dan pengujian white box.

Hasil dari penelitian ini adalah sebuah Pengembangan aplikasi estimasi waktu dan biaya dengan metode use case point yang diintegrasikan pada scrum project management. Pengembangan Aplikasi web telah diuji dengan hasil yaitu pengujian pada aplikasi estimasi waktu dan biaya proyek di hasilkan sebuah Aplikasi yang membantu mahasiswa MTP (Manajemen Tugas Proyek) dalam menghitung waktu dan biaya proyek. Dari hasil diatas dapat disimpulkan bahwa aplikasi estimasi waktu dan biaya dapat berjalan dengan baik dan berfungsi sebagaimana mestinya untuk membantu mahasiswa MTP untuk menghitung estimasi waktu dan biaya proyek.

Kata Kunci : Effort Rate (ER), Use Case Point (UCP), Mahasiswa.

A. PENDAHULUAN

Manajemen merupakan suatu proses atau orang yang bertanggung jawab untuk merencanakan, mendapatkan, mengendalikan, mengarahkan, memastikan, memandu, dan mengatur penggunaan sumber daya yang diperlukan dalam mencapai tujuan dari suatu bisnis secara tepat waktu (Holmes dan Walsh, 2005). Sedangkan proyek di definisikan sebagai usaha tidak rutin dan satu kali yang dibatasi oleh waktu, sumber daya dan spesifikasi kinerja yang dirancang untuk memenuhi kebutuhan-kebutuhan pelanggan (Gray dan Larson, 2006). Manajemen proyek adalah seni menciptakan sesuatu untuk mendapatkan hasil dari serangkaian tindakan yang telah ditentukan secara sengaja sehingga memperoleh keuntungan (Kerzner, 2001), para ahli dan pemimpin bisnis menyatakan bahwa manajemen proyek adalah gelombang masa depan. Manajemen proyek menyediakan sekumpulan piranti yang berdaya guna bagi meningkatkan kemampuan individu dalam merencanakan, mengimplementasikan, dan mengelola berbagai aktivitas untuk mencapai tujuan-tujuan spesifik dari organisasi (Gray dan Larson, 2006).



Menurut M.Ochodek (2010) Peran perangkat lunak di masyarakat menjadi lebih besar dan lebih penting, menjadi sangat penting pengembang perangkat lunak untuk mengembangkan perangkat lunak berkualitas tinggi dengan harga yang efektif dalam waktu yang singkat. Untuk mencapai tujuan ini pengembangan perangkat lunak harus memiliki rencana yang matang baik dalam waktu pengerjaan dan biaya. Dalam tahap awal pengembangan perangkat lunak merupakan tantangan terbesar pengembang dikarenakan pengembang sangat sedikit yang diketahui tentang proyek yang diinginkan oleh pengguna dan ada ancaman bahwa proyek tidak akan bisa dilanjutkan untuk pengembangan lebih lanjut, sehingga sumber daya yang terbatas dapat menghabiskan estimasi usaha dan biaya. Dengan demikian, ada kesenjangan antara tingkat kesalahan estimasi dan sumber daya yang ditugaskan dalam kegiatan estimasi (semakin kecil kesalahan estimasi maka akan mendapatkan keuntungan yang lebih besar). Salah satu aspek kunci dari kesuksesan *projectmanagement* adalah estimasi perangkat lunak. Estimasi yang tidak realistis atau *overestimates*, maka dapat membahayakan proyek tersebut. *Underestimate* juga dapat berpengaruh terhadap komitmen tim karena dapat menyebabkan kekurangan dana dan waktu, sedangkan *overestimates* dapat berakibat penolakan terhadap proposal proyek.

Kebutuhan fungsional merupakan tahap awal pada pengembangan perangkat lunak dengan memberikan gambaran sistem yang diharapkan. Kebutuhan ini dapat digunakan sebagai estimasi sistem untuk memperkirakan upaya pengembangannya. Ide dari *functional size measurement* (FSM) diperkenalkan oleh Allan Albrecht yang mengusulkan sebuah metode yang disebut *Fungsi Analisis Point* (FPA). FPA ini merupakan FSM yang sangat terkenal di tahun 1986. Metode FPA Albrecht ini dikembangkan menjadi metode FSM lainnya, misalnya, Mark II Fungsi Tempat diusulkan oleh Symons (2009), COSMIC (1991), atau Use Case Points (UCP) diperkenalkan oleh Karner (1993). Menurut M.Ochodek (2010), metode UCP dianggap sangat bernilai pada pengukuran ukuran awal dan estimasi perangkat lunak.

Pemodelan *use case* merupakan teknik yang luas dalam menangkap proses bisnis dan persyaratan dari aplikasi proyek perangkat lunak. Pemodelan menggunakan *use case* memberikan ruang lingkup fungsional proyek, menganalisis isinya, memberikan pemahaman tentang upaya dan ukuran untuk merancang dan melaksanakan suatu proyek. Secara umum proyek besar memberikan *use case* yang rumit dan memberikan usaha yang besar untuk merancang dan melaksanakan proyek dari pada proyek kecil. Dalam menyelesaikan sebuah proyek dipengaruhi oleh hal berikut seperti jumlah proses untuk menyelesaikan *use case*, jumlah dan kompleksitas pengguna (*actors*), persyaratan teknik *use case* seperti *concurrency*, keamanan dan kinerja, berbagai faktor tambahan seperti pengalaman dan pengetahuan tim pengembang perangkat lunak.

Menurut Roy, K Clemons (2006) Sebuah estimasi memberikan yang memperhitungkan faktor-faktor diatas diawal siklus hidup pengembangan proyek akan memberikan perkiraan 20% dari waktu penyelesaian yang sebenarnya akan sangat membantu untuk penjadwalan proyek, biaya, dan alokasi sumber daya. *Use case point* (UCP) memberikan perkiraan jumlah jam/orang pada pengembangan perangkat lunak berdasarkan dari *use case*. Penggunaan metode *use case points* memberikan perkiraan *effort* mendekati perkiraan sebenarnya yang dihasilkan oleh pengembang perangkat lunak berpengalaman. Ini menandakan pengalaman tim proyek akan mempengaruhi tingkat akurasi estimasi yang dilakukan. Berdasarkan karya Gustav Karner metode *use case point* (UCP) menganalisa *Use Case Actors*, skenario dan berbagai faktor teknis dan abstrak yang dimasukkan menjadi satu bagian.



Pada proses pelaksanaan proyek, terdapat sebuah kerangka kerja yang dapat digunakan dengan pendekatan *Agile*. Kerangka kerja tersebut adalah kerangka kerja *Scrum*. *Scrum* merupakan sebuah kerangka kerja dimana orang-orang dapat menyelesaikan permasalahan kompleks yang senantiasa berubah-ubah, dimana pada saat bersamaan menghasilkan produk dengan nilai setinggi mungkin secara kreatif dan produktif (Sutherland & Schwaber, 2013). Sehingga estimasi proyek dapat digunakan sebagai perencanaan proyek dan kerangka kerja scrum dapat digunakan sebagai panduan pelaksanaan proyek. Pada scrum terdapat tahapan yaitu menentukan *backlog*. *Backlog* merupakan daftar fitur prioritas, yang berisikan deskripsi singkat semua fungsi yang diinginkan didalam produk. Ketika menggunakan kerangka kerja scrum tidak perlu untuk memulai proyek yang panjang, cukup dengan mendokumentasikan semua persyaratan yang diinginkan didalam produk dalam hal ini juga merupakan identifikasi awal pada perhitungan estimasi proyek pada UCP. Tim *Scrum* dan pemilik produk akan menuliskan segala sesuatu yang mereka bisa prioritaskan untuk produk yang akan dibuat. Sebuah *Backlog Scrum* ditulis dengan *user stories*. Cara utama tim scrum untuk mengekspresikan fitur produk adalah dalam bentuk cerita pengguna yang singkat dan deskripsi sederhana dari fungsi yang diinginkan diceritakan dari sudut pandang pengguna.

Dengan memanfaatkan metode-metode yang sudah ada pada penelitian mengenai *use case points* (UCP) dan digabungkan dengan scrum *framework* maka akan dikembangkan sebuah sistem untuk menghitung estimasi biaya yang dihitung berdasarkan *use case point* dari rancangan UML dan *user stories* di *scrum*. Diharapkan dengan adanya estimasi biaya proyek perangkat lunak ini akan membantu para pengembang perangkat lunak dalam menentukan waktu dan biaya dalam mengembangkan sebuah proyek perangkat lunak.

Hasil dari penelitian ini dapat dimanfaatkan sebagai sebuah alat bantu pelaksanaan Manajemen tugas proyek (MTP) di Program Studi Teknik Informatika Universitas Ahmad Dahlan Yogyakarta dimana penyelenggaraan MTP bertujuan untuk membekali kemampuan mahasiswa untuk mengelola proyek dalam lingkup bidang Teknologi Informasi serta melatih kemampuan kerja dalam tim. Berdasarkan wawancara yang dilakukan kepada mahasiswa MTP, disebutkan bahwa permasalahan yang ditemui oleh mahasiswa MTP saat ini adalah mahasiswa mempunyai kesulitan diantaranya menentukan waktu dan biaya untuk menyelesaikan suatu proyek. Sehingga hasil penelitian ini diharapkan dapat membantu mahasiswa MTP untuk mengestimasi waktu dan biaya proyek yang akan dikerjakan.

B. KAJIAN PUSTAKA

Penelitian ini mengacu pada penelitian terdahulu oleh Rick Kazman (2011), pada penelitian tersebut menghasilkan teknologi *interactive Use-Case Points (iUCP)*. Metode yang digunakan yaitu menggabungkan antara *Use Case Points (UCPs)* dan *Usage Centered design (UsageCD)*. Mereka membangun teknologi iUCP berdasarkan data statistika dari proyek UsageCD yang dikumpulkan selama bertahun-tahun. Akan tetapi, beberapa evaluasi yang sistematis memerlukan pengumpulan data yang lebih luas dan analisis dalam jangka waktu yang lama. Tujuannya pembuatan teknologi ini adalah untuk membantu pengembang *Software* dan desainer untuk lebih mudah dalam menerapkan heuristik yang cocok untuk aplikasi interaktif dan menjalankan pekerjaan secara konsisten pada sebuah proyek perangkat lunak. Sebuah Proyek yang umum di ambil dari poin-poin dari *use case* yang sudah tertera dan menghasilkan besaran jam dari keseluruhan proyek. Namun pada penelitian memiliki Kekurangan yaitu teknologi ini memiliki ketergantungan akan data

dari *UsageCD* yang dikumpulkan bertahun-tahun. Apabila data yang dikumpulkan tidak memenuhi syarat maka estimasi waktu yang dilakukan tidak akan menghasilkan estimasi yang tepat. Selain itu teknologi ini juga menghasilkan selisih antara SE (*Software Engineering*) dan HCI (*Human Computer Interaction*) yang cukup besar jika data yang dimasukkan belum sepenuhnya lengkap.

Pada penelitian terdahulu oleh Mudasir Manzoor Kirmani (2015), ini menghasilkan teknologi *Use case Point* dan menggunakan metode *Cost models, sizing methods (function Points Analysis (FPA))*. Dalam penelitian ini menganjurkan untuk menggunakan *Use Case Point* sebagai metode yang efektif untuk estimasi usaha *Software* di tahap awal proses pengembangan perangkat lunak. Selain itu mereka juga menekankan pada penggunaan penyulingan metode titik kasus estimasi untuk perangkat lunak pada pengembangan proyek perangkat lunak. Namun dalam rangka untuk memiliki metode UCP dalam menghitung estimasi waktu perlu adanya peninjauan dan perbaikan cara transaksi menggunakan kasus hitung. Oleh karena dalam menghitung jumlah waktu mereka menekankan untuk lebih menyempurnakan parameter-parameter yang digunakan dan faktor teknis dari metode UCP dengan metode yang lebih baik dan *Easy-Touse* untuk menghitung jumlah pengguna transaksi use case yang dapat menyebabkan model UCP yang lebih efektif dari estimasi upaya proyek-proyek pengembangan perangkat lunak, sehingga mendapatkan hasil yang lebih tepat dan akurat. Kekurangan dari metode ini adalah jika perhitungan transaksi yang terjadi pada use case salah maka estimasi yang dihasilkan tidak akan tepat, atau bahkan bisa merugikan pengembang. Karena perhitungan-perhitungan transaksi tersebut sangat mempengaruhi perhitungan estimasi waktu yang dihasilkan oleh teknologi ini.

Dalam penelitian lain yang dilakukan oleh Poonam Kumari(2015), ini menghasilkan teknologi *particle Swarm Optimization Algoritma (PSO)*. Pada penelitian ini menggunakan metodologi titik *use case* untuk upaya perhitungan estimasi waktu, metode *meta heuristic techniques dan soft computing technique*. Optimasi PSO juga dapat diterapkan untuk memperbaiki upaya pengembangan perangkat lunak. Teknologi ini meningkatkan akurasi upaya pengujian perhitungan estimasi waktu. Namun penelitian dari metode PSO ini memiliki kekurangan yaitu perhitungan yang terlalu rumit dari pada metode *use case* lainnya karena terlalu banyak komponen yang dihitung untuk menghasilkan estimasi waktu yang tepat.

Dalam penelitian yang dilakukan oleh M.Ochodek, J.Nawrocki, K.Kwarciak (2010), ini menghasilkan teknologi *Use case point* dan menggunakan metode *Functional Size Measurement (FSM)*, ukuran *Use Case Point Analysis (UCPA)* dan *Use Case Size Poin (USP)*. Namun penelitian ini memiliki Kekurangan yaitu perhitungan pada komponen-komponen dalam estimasi waktu harus tepat. Apabila ada beberapa faktor yang terlewat maka estimasi waktu menjadi tidak tepat.

Pada penelitian terdahulu oleh Ali Bou Nasif, Danny HO dan Luiz Fernandi C (2011), menghasilkan teknologi untuk menghitung estimasi waktu untuk pengerjaan proyek perangkat lunak. Dalam perhitungannya mereka menggunakan referensi *use case point* yang di kenalkan oleh Karner's Scheider's. Dengan menggunakan teknologi ini pengembang proyek lunak dapat mengetahui jumlah waktu yang diperlukan untuk menyelesaikan pengerjaan proyek perangkat lunak. Namun penelitian ini memiliki Kekurangan dari sisi teknologi yaitu estimasi waktu yang dilakukan pada beberapa proyek didapat hasil yang memuaskan untuk waktu pengerjaan di bawah 5000 jam, akan tetapi hasil berbeda didapat pada proyek besar yang waktu pengerjaannya di atas 5000 jam.



C. METODE PENELITIAN

1. Objek penelitian

Objek penelitian yang akan dibahas pada penelitian ini adalah “mahasiswa matakuliah Manajemen tugas proyek (MTP) yang ada di Universitas Ahmad Dahlan Yogyakarta. Mahasiswa matakuliah MTP tersebut yaitu mahasiswa angkatan 2012 dan mahasiswa angkatan 2013.

2. Metode Pengumpulan Data

Pengumpulan data dilakukan dengan studi literature dan wawancara. Studi literatur berasal dari berbagai macam buku, artikel, publikasi ilmiah. Wawancara dilakukan kepada mahasiswa MTP yang telah menyelesaikan matakuliah MTP.

3. Metode Pengembangan

Metode yang digunakan untuk membuat Aplikasi Web untuk *Scrum Project Management* ini ialah menggunakan kerangka kerja Scrum. Scrum merupakan salah satu kerangka kerja metode *Agile*.

Adapun tim *Scrum* yang terlibat adalah sebagai berikut :

1. *Product Owner* : Herman Yuliansyah, S.T., M.Eng.
2. *Scrum Master* : Adi Alamsyah
3. *Developer Team* :
 - a. Sarah Nurul Qudsiyah
 - b. Muhammad Reza
 - c. Pratama Setya Aji
 - d. Adi Alamsyah

4. Jalannya Penelitian

Adapun jalannya penelitian yang akan dilakukan pada skripsi ini adalah sebagai berikut :

- a. *Product backlog* ditentukan oleh *Product Owner*. *Product Owner* menentukan dengan mencatat semua kebutuhan yang dibutuhkan oleh aplikasi. Daftar *Product Backlog* langsung diurutkan sesuai prioritasnya. *Product Backlog* tidak dibatasi dan akan menjadi acuan atau sumber utama dari semua kebutuhan yang harus ada pada aplikasi ini. *Product Backlog* didokumentasikan menggunakan *Trello*. *Product Backlog* akan digunakan pada saat *Sprint Planning Meeting* untuk menentukan tugas-tugas yang akan dikerjakan sebagai *Sprint Backlog*. Perancangan arsitektur sistem untuk menggambarkan bentuk arsitektur pengembangan server dalam sistem tersebut.
- b. *Penentuan Sprint Backlog* ditentukan dengan mengacu pada *Product Backlog* yang telah ditentukan sebelumnya. *Developer Team* melaksanakan *Sprint Planning Meeting* untuk menentukan *Sprint Backlog* yang akan dikerjakan oleh *Developer Team*. Dalam *Sprint Planning Meeting* juga menentukan waktu *start* dan estimasi pengerjaan setiap *Sprint* oleh *Developer Team* dalam satuan jam serta menentukan *Sprint Goal* pada aplikasi ini. *Sprint Backlog* didokumentasikan di *Trello*. *Sprint Planning Meeting* akan menghasilkan *Sprint Backlog* yang siap untuk dikerjakan.
- c. *Implementasi dan Daily Scrum*
Setelah *Sprint Planning Meeting* selesai dan sudah menghasilkan *Sprint Backlog* serta *Sprint Goal* yang jelas,

selanjutnya *Developer Team* akan segera bekerja menyelesaikan *Sprint Backlog* yang telah diperoleh dari *Sprint Planning Meeting*.

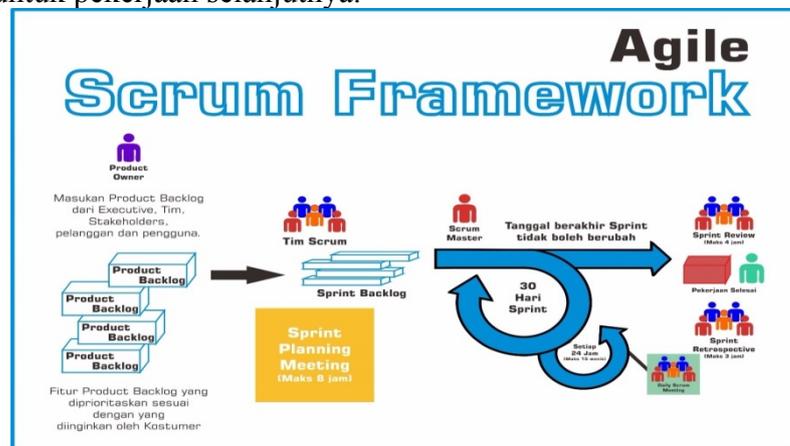
Daily Scrum dilakukan setiap hari dan di jam yang sama. *Daily Scrum* dilaksanakan di satu tempat atau dengan menggunakan media *messagingWhatsapp*. Waktu pelaksanaan *Daily Scrum* maksimal 15 menit dan hadirin yang terlibat dalam *Daily Scrum* adalah *Scrum Master* dan *Developer Team*.

d. *Sprint Review*

Sprint Review dilaksanakan diakhir *Sprint* untuk meninjau pekerjaan yang telah dikerjakan oleh *Developer Team*. *Sprint Review* dihadiri oleh tim *Scrum* dan *Stakeholder* dan waktu pelaksanaannya maksimal empat jam. Tim *Scrum* dan *Stakeholder* berkolaborasi untuk membahas *Sprint* yang telah dikerjakan dan diselesaikan oleh *Developer Team*.

e. *Sprint Retrospective*

Sprint Retrospective dilakukan untuk memberi kesempatan Tim *Scrum* meninjau dirinya sendiri dan membuat perencanaan untuk pekerjaan selanjutnya.



Gambar 1 :Skema dan alur dari *Scrum framework*

5. Pengujian Sistem

a. Pengujian Unit (White-Box Test)

White-box test adalah pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program secara procedural untuk membagi pengujian ke dalam beberapa kasus pengujian. Adapun metode yang akan digunakan dalam *white-box test* yaitu menggunakan *unit testing* yang disediakan oleh *framework PHP Codeigniter*.

b. Black-Box Test

Pengujian *Black-box* memungkinkan pembuat perangkat lunak untuk menentukan kondisi yang terjadi untuk suatu masukan yang akan menjalankan semua kebutuhan fungsional dari perangkat lunak yang dibuat.

Pengujian *black-box* dilakukan untuk menentukan beberapa macam kesalahan yaitu :

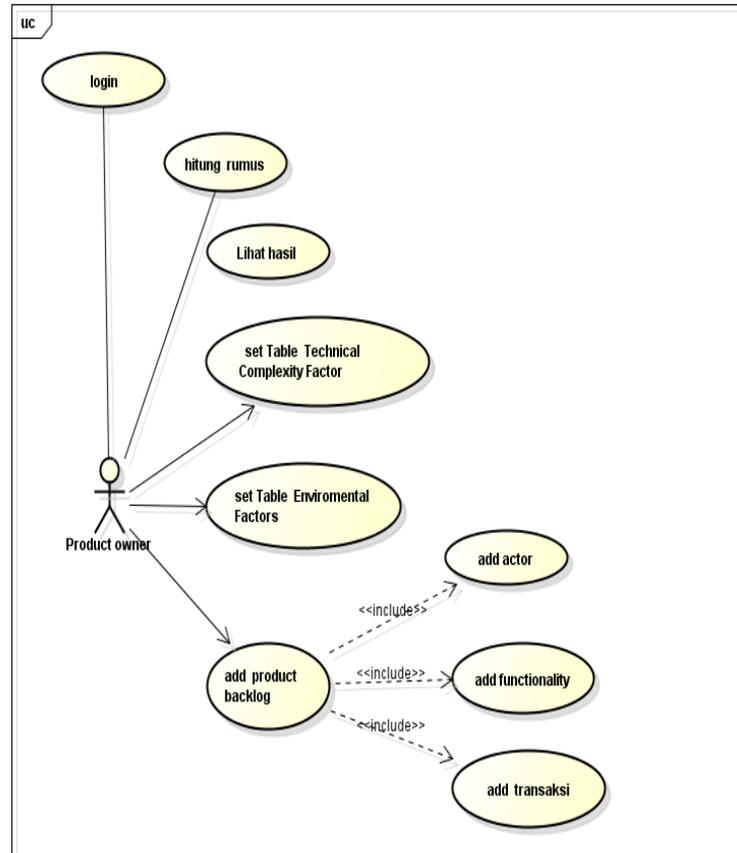
- Fungsi-fungsi yang tidak benar atau hilang.
- Kesalahan interface.
- Kesalahan dalam struktur data atau akses *database* eksternal.

d. Kesalahan kinerja

D. HASIL DAN PEMBAHASAN

1. Perancangan Sistem

a. Use Case Diagram

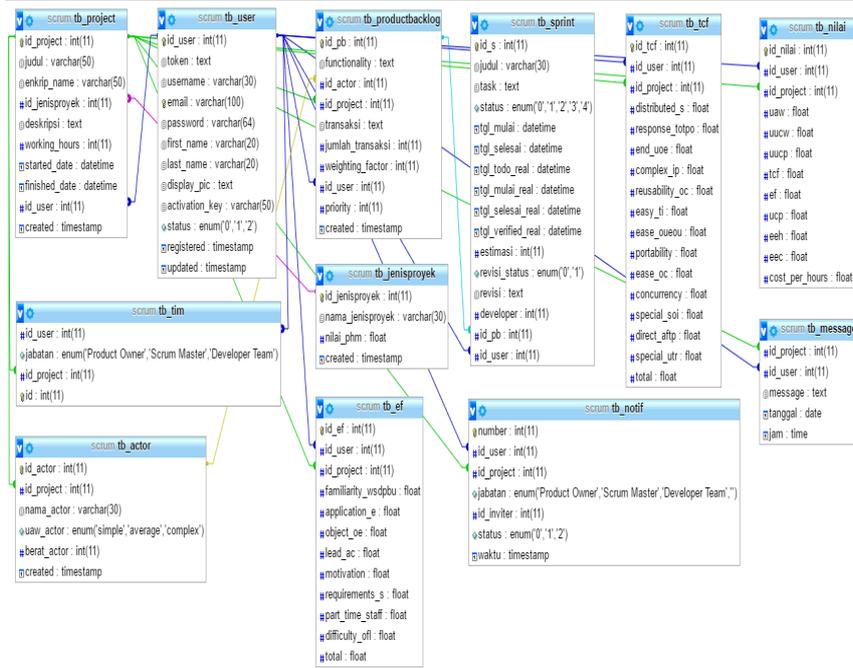


Gambar 2 . Use Case Diagram

Pada gambar 2, Untuk mendapatkan gambaran dari sistem yang akan dibuat maka digunakan proses dan data model dari sistem yang akan dimodelkan dengan *use case diagram*. Pembuatan diagram *use case* berdasarkan pada analisis kebutuhan dari pengguna, di dalam sistem yang akan dibangun yaitu *product owner* melakukan *create product backlog*, *product owner* melakukan *Add Product Backlog*, *product owner* memilih *actor* dan *Product Owner* juga bisa menambah *Actor*. Kemudian *Product Owner* juga bisa input *functionality*, setelah itu *Product Owner* akan memasukkan banyaknya jumlah transaksi. *Product owner* mengisi table *tcf Technical Complexity Factors* dan table *Envieronmental Factors*, kemudian *Product Owner* bisa menghitung nilai dan bisa melihat hasil estimasi waktu dan biaya merupakan penelitian yang sedang dilakukan.

b. Mapping Table

Setelah membuat ERD, tahap selanjutnya adalah memetakan entitas-entitas ataupun relasinya dalam bentuk mapping table seperti pada gambar 3 Berikut adalah mapping table berdasarkan ERD yang telah dibuat.



Gambar 3. Mapping Table

2. Software Testing

Pada tahap ini akan dilakukan pengujian terhadap aplikasi web untuk manajemen proyek berbasis *Scrum*, apakah sudah memenuhi persyaratan sebagai suatu aplikasi web. *Software Testing* yang dilakukan pada aplikasi yang akan dibuat pada penelitian ini ialah

a. Black Box Test

Pengujian *Black-box test* dilakukan oleh Merlinda Wibowo, S.T pengujian terhadap *Black-box test* tersebut dilakukan dengan menjawab daftar kuisioner mengenai apakah aplikasi sesuai dengan yang diharapkan.

b. White Box

Pengujian unit menggunakan teknik *white box*. Pengujian digunakan untuk mengetahui cara kerja suatu perangkat lunak secara internal, menjamin operasi-operasi internal sesuai dengan spesifikasi yang telah ditetapkan. Pengujian ini menggunakan alat / *tools Unit testing codeigniter* yang dikembangkan oleh codeigniter untuk otomatisasi pengujian. Pengujian dilakukan terhadap *3function* yang ada pada sistem *web* Estimasi waktu dan biaya proyek. Fungsi tersebut adalah fungsi update tcf, update ef, update rumus..

E. PENUTUP

a. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat disimpulkan sebagai berikut :



- a. Berdasarkan hasil pengujian pada aplikasi estimasi waktu dan biaya proyek di hasilkan sebuah Aplikasi yang membantu mahasiswa MTP (Manajemen Tugas Proyek) dalam menghitung waktu dan biaya proyek.
 - b. Berdasarkan hasil pengujian pada aplikasi estimasi waktu dan biaya dapat diketahui bahwa 100% testcase yang diuji pada aplikasi telah sesuai dengan Expected Result. Sehingga dapat disimpulkan bahwa aplikasi dapat berjalan dengan baik dan berfungsi sebagaimana mestinya untuk membantu mahasiswa MTP untuk menghitung estimasi waktu dan biaya proyek bidang
- b. Saran

Pengembangan aplikasi Estimasi waktu dan Biaya Proyek dengan Menggunakan metode *Use Case Point* Diintegrasikan pada *Scrum Project Management* ini merupakan tahap pengembangan sekaligus tahap pembelajaran pengembangan sehingga masih banyak bagian-bagian dari aplikasi yang perlu ditambahkan dan dikembangkan lagi. Adapun Saran yang dapat disampaikan dari hasil penelitian ini adalah sebagai berikut:

- a. Kekurangan pada sistem ini adalah tampilan antar muka atau *user interface* yang kurang menarik dan *user friendly*. Diharapkan untuk memperbaiki tampilan sehingga tampilan *web* lebih menarik dan *user friendly*
- b. Pengembangan aplikasi estimasi waktu dan biaya proyek dengan metode *use case point* yang diintegrasikan pada *scrum project management* selanjutnya dapat dikembangkan dengan honor perindividu.

F. DAFTAR PUSTAKA

- Ali Bou Nassif. 2011. *Regression Model for Software Effort Estimation Based on the Use Case Point Method*. IPCSIT vol.14 (2011) IACSIT Press, Singapore.
- A. Abran, J. Desharnais, S. Oigny, D. St-Pierre, C. Symons, The COSMIC Functional Size Measurement Method v3.0.1, Measurement Manual, May 2009.
- C.R. Symons, Software Sizing and Estimating: Mk II FPA (Function Point Analysis), John Wiley & Sons, Inc., New York, NY, USA, 1991.
- G. Karner, Metrics for objectory, Master's thesis, University of Linköping, Sweden, 1993
- Fadhil Rahmat. 2008. *Tingkat Kematangan Manajemen Proyek Pada Beberapa Industri Kopi di Aceh* (Riset). Institut Pertanian Bogor (IPB).
- Irine Dwi Kenestie, Sholiq 2011. *Penentuan Nilai Effort Rate (ER) Pada Metode Use Case Point (UCP) Untuk Estimasi Effort Proyek Pengembangan Perangkat Lunak di Bidang Pendidikan*. Jurnal Teknik Pomits.
- Ken Schwaber. 2004. *Agile Project Management with Scrum*, Microsoft Press.
- Kurniawan dkk.2013. *Penentuan Effort Rate Pada Estimasi Effort Menggunakan Metode Use Case Point Untuk Pengembangan Perangkat Lunak Website Pemerintahan*. Jurnal Sistem Informasi.(JSIKA Vol 2, No 2 (2013)/ ISSN 2338-137X).
- M. Ochodek dkk. 2010. *Simplifying Effort Estimation Based of Use Case Points* (jurnal). Poznan University of Technology, Institute of Computing Science, Poznan, Poland.
- Mudasir Manzoor Kirmani. 2015. *Use Case Point Method of Software Effort Estimation: A Review* . Volume 116 – No. 15, April 2015.



- Muhadi Hariyanto dkk.2015. Estimasi Proyek Pengembangan Perangkat Lunak dengan Fuzzy Use Case Points.Vol. 1, No. 1, April 2015
- Poonam Kumari.2015 .*Turing Of Use Case Points(UCP) Analysis Parameter Using PSO.*
- Rick Kazman dkk. 2011. *i UCP: Estimating InteractiveSoftware Project Size with Enhanced Use-Case Points. University of Hawaii.*
- Roy, K Clemons. 2006. *Project Estimation With Use Case Points* (Jurnal). Diversified Technical Services, Inc.
- Schwaber K dan Sutherland J. (2013).*Scrum Guide ID.* Tersedia pada <http://www.scrumguides.org/>