



## **APLIKASI MOBILE UNTUK MONITORING TASK DAN KOMUNIKASI DAILY SCRUM PADA SCRUM PROJECT MANAGEMENT**

**<sup>1</sup>Muhammad Reza(12018180),<sup>2</sup>HermanYuliansyah(0512078304)**

1,2ProgramStudi Teknik Informatika Universitas Ahmad Dahlan  
Jl. Prof.Dr.Soepomo,Janturan,Yogyakarta 55164  
1E-mail :ejhapoo@gmail.com  
2E-mail : herman.yuliansyah@tif.uad.ac.id

### **ABSTRAK**

*Selama masa pengembangan dan pelaksanaan perlu dilakukan monitoring dan pengendalian untuk mengetahui apakah proyek tersebut sudah berjalan dengan lancar dan sesuai dengan sasaran dan tujuan yang diinginkan. Berdasarkan salah satu prinsip dari agile manifesto, yaitu "Metode yang paling efisien dan efektif untuk menyampaikan informasi ke dan dalam tim pengembangan adalah percakapan tatap muka". Offshore outsourcing development tampaknya menentang scrum dalam beberapa hal, salah satunya adalah karena scrum mengharuskan komunikasi tatap muka antar anggota tim scrum, namun definisi offshore development adalah jarak yang jauh antar anggota tim. Tujuan pembuatan aplikasi ini adalah untuk menghasilkan aplikasi mobile yang dapat mengadopsi komunikasi daily scrum dalam framework scrum dan monitoring tugas proyek.*

*Dalam pembuatan aplikasi mobile untuk monitoring task dan daily scrum pada scrum project mangement,menggunakan salah satu kerangka kerja dari metodologi agile, yaitu Scrum. Scrum memiliki 4 acara utama, yaitu Sprint Planning Meeting, Implementasi dan Daily Scrum,Sprint Review, dan Sprint Retrospective. Implementasisistem menggunakan Java Android.Sistem diuji dengan 2 metode, yaitu Junit test, Black-box test, dan Usablity test.*

*Hasil dari penelitian ini aplikasi mobile untuk monitoring task dan daily pada scrum project mangement. Sistem telah diuji dengan hasil 15 fungsi dan 9user storiesberjalan dengan sesuai.Dari hasil tersebut dapat disimpulkan bahwa aplikasi sudah dapat digunakan denganbaik.*

**Kata Kunci** : Scrum, Daily Scrum, OffShore, Monitoring..

### **A. PENDAHULUAN**

Proyek didefinisikan 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). Dalam membuat proyek, dibutuhkan suatu pengelolaan atau manajemen agar mencapai tujuan yang memiliki kualitas. Manajemen adalah 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). Manajemen proyek adalah seni menciptakan sesuatu untuk mendapatkan hasil dari serangkaian tindakan yang telah

ditentukan secara sengaja sehingga memperoleh keuntungan (Kerzner, 2001). 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).

Pelaksanaan proyek diharapkan dapat dilakukan dengan baik dan terarah. Karena itu selama masa pengembangan perlu dilakukan *monitoring* dan pengendalian untuk mengetahui apakah proyek tersebut sudah berjalan dengan lancar dan sesuai dengan sasaran dan tujuan yang diinginkan. Jika terdapat penyimpangan, diharapkan dapat segera dideteksi dan dapat segera diambil langkah perbaikan (antisipasi). Hal itu karena jika tidak maka akan menghambat pelaksanaan proyek dan menyebabkan waktu dan biaya tidak sesuai dengan estimasi yang diharapkan. Dengan pengawasan yang baik, dapat dilakukan tindakan pencegahan (*preventive*) terhadap kecenderungan penyimpangan yang mungkin dapat dan akan terjadi. Pelaksanaan proyek itu sendiri juga harus terukur dengan baik, pengukuran dapat berdasarkan waktu tempuh pelaksanaan tugas, realisasi anggaran, dan pencapaian sasaran tugas. Hal itulah yang menentukan pencapaian sasaran proyek secara keseluruhan (Noerlina, 2008).

*Scrum* adalah sebuah kerangka kerja untuk mengembangkan dan mengelola produk kompleks, dimana pada saat bersamaan menghasilkan produk dengan nilai setinggi mungkin secara kreatif dan produktif. Didalam *scrum* dapat dimasukkan beragam proses dan teknik untuk mengembangkan produk. *Scrum* akan memperlihatkan seberapa besar efektifitas perkembangan dari manajemen produk dan proses pengembangan yang sedang tim jalani, dengan begitu tim dapat melakukan peningkatan pada bagian yang dirasa kurang (Ken Schwaber, 2013).

*Scrum* menyediakan empat acara formal, yang dibungkus didalam *Sprint*, untuk inspeksi dan adaptasi, seperti *Sprint Planning*, *Daily Scrum*, *Sprint Review*, dan *Sprint Retrospective*. *Sprint Planning* merupakan pekerjaan yang akan dilaksanakan didalam *Sprint* direncanakan pada saat *Sprint Planning*. Perencanaan ini dibuat secara kolaboratif oleh seluruh anggota tim *scrum*. *Daily Scrum* adalah diskusi dengan batasan waktu maksimum selama 15 menit agar Tim pengembang dapat mensinkronisasikan pekerjaan dan membuat perencanaan untuk 24 jam kedepan. Ini dilakukan untuk meninjau pekerjaan dari *Daily Scrum* terakhir dan memperkirakan pekerjaan yang dapat dilakukan sebelum *daily scrum* berikutnya (*Scrum Guide ID*, 2013).

*Sprint Review* merupakan kegiatan yang diadakan diakhir *Sprint* untuk meninjau inkremen dan merubah *Product Backlog* bila diperlukan. Tim *Scrum* dan stakeholder berkolaborasi untuk membahas apa yang telah dikerjakan dalam *Sprint* yang baru usai. *Sprint Retrospective* adalah sebuah kesempatan bagi Tim *Scrum* untuk meninjau seluruh kegiatan tim *scrum* pada *Sprint* sebelumnya dan membuat perencanaan mengenai peningkatan yang akan dilakukan pada *Sprint* berikutnya. *Sprint* ini diadakan setelah *Sprint Review* dan sebelum *Sprint Planning* berikutnya (*Scrum Guide ID*, 2013).

Salah satu prinsip dasar metodologi perangkat lunak *scrum* adalah pentingnya komunikasi antara berbagai orang yang terlibat dalam pengembangan perangkat lunak. *Scrum* menekankan peningkatan komunikasi melalui komunikasi tatap muka. Berdasarkan salah satu prinsip dari agile manifesto, yaitu "Metode yang paling efisien dan efektif untuk menyampaikan informasi ke dan dalam tim pengembangan adalah percakapan tatap muka". *Trend* lain yang telah menyebar didunia pengembangan perangkat lunak baru-baru ini adalah berpindahnya pengembangan perangkat lunak ke arah *offshore outsourcing development*, dimana banyak pengembangan perangkat lunak

dikerjakan dengan biaya lebih murah karena pengembangan dilakukan oleh beberapa pengembang dari benua berbeda. *Offshore outsourcing development* tampaknya menentang scrum dalam beberapa hal, salah satunya adalah karena scrum mengharuskan komunikasi tatap muka antar anggota tim scrum, namun definisi *offshore development* adalah jarak yang jauh antar anggota tim (Martin fowler, 2003).

*Framework* scrum lebih banyak menekankan komunikasi antar *stakeholder* yang terlibat, seperti dalam kegiatan *meeting*, *review*, dan *Sprint Retrospective*. Cara komunikasi yang umum digunakan yaitu pertemuan secara langsung/*meeting*. Pertemuan langsung/*meeting* memiliki kelebihan untuk dapat lebih memperkuat ikatan antar *stakeholder* dalam pengerjaan proyek serta dapat lebih memotivasi satu sama lain sehingga proyek dapat berjalan dengan baik.

Perpaduan scrum dengan *offshore outsourcing* membutuhkan cara lain karena lokasi anggota tim scrum yang berjauhan dan tidak memungkinkan untuk pertemuan langsung. Cara yang dapat digunakan yaitu media aplikasi *chatting*. Namun pada media *chatting* hanya dapat melakukan “*chatting*”. Tanpa adanya fitur untuk memilah-milah acara formal yang terdapat pada scrum, yaitu *Sprint Planning*, *Daily Scrum*, *Sprint Review*, dan *Sprint Retrospective*.

Pada program studi Teknik Informatika Universitas Ahmad Dahlan Yogyakarta, terdapat salah satu matakuliah Manajemen Tugas Proyek (MTP) dimana MTP merupakan salah satu matakuliah wajib yang terdiri dari 1 sks teori dan 2 sks pelaksanaan proyek yang bertujuan diselenggarakannya untuk membekali kemampuan mahasiswa untuk mengelola proyek dalam lingkup bidang Teknologi Informasi serta melatih kemampuan kerja dalam tim.

Dari hasil wawancara dengan mahasiswa matakuliah MTP, selama menempuh matakuliah MTP, mahasiswa mengalami beberapa kesulitan diantaranya mahasiswa kesulitan dalam mengawasi tugas-tugas mahasiswa saat MTP. Mahasiswa MTP juga membutuhkan media komunikasi yg lebih dari sekedar *messenger* biasa. Karna pengerjaan manajemen tugas proyek (MTP) berlangsung selama satu semester (beberapa bulan), maka akan sulit melakukan dokumentasi komunikasi selama pengerjaan proyek berlangsung jika hanya menggunakan *messenger* biasa.

Karena mahasiswa MTP kesulitan dalam melakukan pengawasan terhadap proyek dan kesulitan dokumentasi komunikasi selama pengerjaan proyek berlangsung, dan dengan *monitoring* yang baik, dapat dilakukan tindakan pencegahan (preventive) terhadap kecenderungan penyimpangan yang mungkin dapat dan akan terjadi (Noerlina, 2008). Untuk itu perlu dibuat aplikasi sebagai media komunikasi serta dokumentasi komunikasi selama pengerjaan manajemen proyek yang juga berguna sebagai sarana *monitoring* proyek berbasis *mobile*.

Teknologi *mobile* yang sedang berkembang saat ini adalah Android. Android merupakan sistem operasi bergerak (*mobile*) yang menggunakan versi modifikasi dari kernel Linux. Sistem ini memiliki berbagai keunggulan sebagai *software* berbasis kode komputer yang bisa didistribusikan secara terbuka (*open source*), sehingga *programmer* bisa membuat aplikasi baru didalamnya. Hal tersebut menjadikan android cocok untuk mendukung *monitoring* dalam scrum. Sampai saat ini, ada beberapa aplikasi android yang sudah mendukung *monitoring* dalam scrum. Diantaranya Planbox, dan TargetProcess.

## 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 Point* (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 Qudsiah
  - b. Muhammad Reza
  - c. Pratama Setya Aji
  - d. Adi Alamsyah

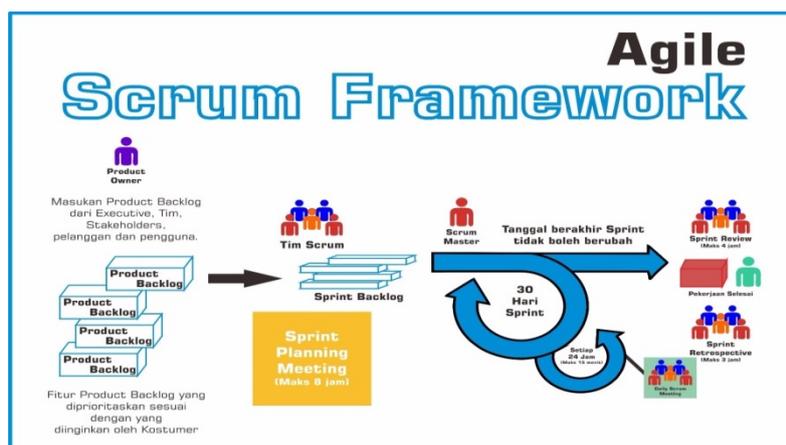
#### 4. Jalannya Penelitian

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. *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.

- b. **Penentuan Sprint Backlog:** *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 menggunakan *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*. Waktu pengerjaan setiap *Sprint* untuk mencapai *Sprint Goal* maksimal dikerjakan selama 30 hari. Setiap *Sprint Backlog* dikerjakan oleh satu orang anggota *developer Team*. Pekerjaan yang dilakukan meliputi desain *user interface*, desain basis data, pembuatan kode program sampai membuat *test* skenario dan melakukan *testing*.
- d. ***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

Unit testing merupakan pengujian yang dilakukan untuk memastikan bahwa sistem dapat berfungsi dengan benar (Perry, 2006). Dari definisi diatas, unit testing merupakan pengujian fitur atau modul yang merupakan bagian kecil dari sistem yang diuji secara terpisah untuk memastikan fitur atau modul tersebut dapat berfungsi dengan baik dan benar.

### 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 :

- 1) Fungsi-fungsi yang tidak benar atau hilang.
- 2) Kesalahan interface.
- 3) Kesalahan dalam struktur data atau akses *database* eksternal.
- 4) Kesalahan kinerja

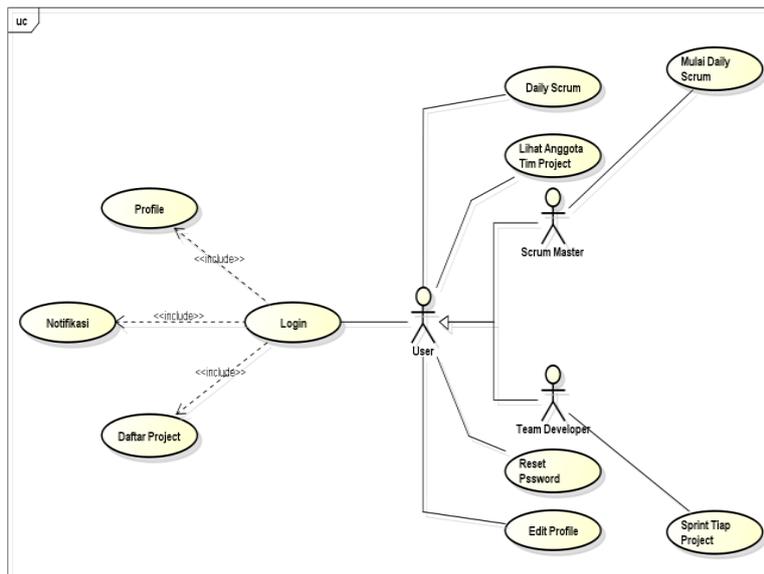
### c. Usability Testing

*Usability* merupakan analisa kualitatif yang menentukan seberapa mudah user menggunakan antarmuka suatu aplikasi. *Usability Testing* yang akan digunakan adalah Metode SUS (System Usability Scale). Metode ini diaplikasikan dengan menggunakan 10 pernyataan berbentuk kuisoner yang diikuti dengan skala Likert 5 tingkat sebagai jawaban untuk setiap pernyataan, mulai dari sangat setuju hingga sangat tidak setuju

## D. HASIL DAN PEMBAHASAN

### 1. Perancangan Sistem

#### a. Use Case Diagram

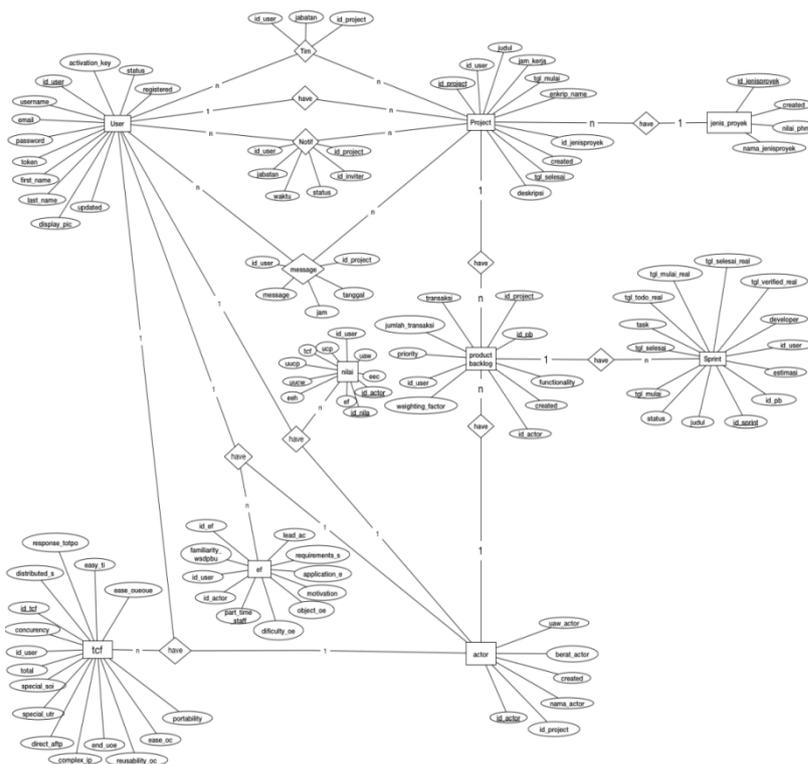


Gambar 2 . Use Case Diagram

Gambar diatas merupakan *UsecaseDiagram* yang terbentuk dari *product backlog*. Dan nanti akan diimplementasikan tiap usecase menjadi fungsi pada pengembangan aplikasi.

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. Hasil Sprint

Penentuan *sprint* ditentukan saat mengadakan *Sprint Planning Meeting* yang diadakan di Ruang Lab. Jaringan Kampus 3 UAD jalan Janturan. Dilaksanakan pada tanggal 4 Mei 2016 pukul 10.00 WIB dan diadakan selama 3 jam. Adapun *sprint* yang dihasilkan saat *SprintPlanningMeeting* tersebut adalah sebagai berikut :

Tabel Sprint Backlog

No	Product Backlog	Sprint Backlog	Estimasi
1	Log in mobile	Tampilan Log in	10 jam
		Activity Log in	15 jam
		Testing Log in	5 jam
2	Reset password mobile	Tampilan Reset Password	10 jam
		Activity Reset Password	20 jam
		Testing Reset Password	10 jam
3	Edit profile mobile	Tampilan Profile	10 jam
		Tampilan Form Edit Profile	5 jam
		Activity profile	26 jam
		Testing	15 jam
4	Dashboard project mobile	Tampilan Dashboard	10 jam
		Activity Dashboard	15 jam
		Testing	7 jam
5	Notifikasi invite tim Scrum mobile	Tampilan notifikasi invite project	16 jam
		Activity notifikasi invite	16 jam
6.	Daily Scrum Mobile	Tampilan History Daily	10 jam
		Activity History Daily	14 jam
7.	Chat Mobile	Tampilan Chat Daily Mobile	25 jam
		Activity Chat Daily Mobile	42 jam
		Activity Chat By Date	33 jam

		Mobile	
8.	Info tim Scrum Mobile	Tampilan info tim Scrum mobile	13 jam
		Activity info tim scrum	19 jam
9.	Sprint Mobile	Tampilan Sprint Todo	15 jam
		Activity Sprint Todo	20 jam
		Tampilan Sprint Doing	10 jam
		Activity Sprint Doing	11 jam
		Tampilan Sprint Done	5 jam
		Activity Sprint Done	10 jam
		Tampilan Sprint Verified	5 jam
		Activity Sprint Verified	25 jam
Total Estimasi			447 Jam

### 3. 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. Unit Testing

Unit testing dilakukan oleh programmer yang menuliskan suatu modul/unit tertentu. Unit testing yang akan digunakan adalah Junit karena bahasa pemrograman yang dipakai adalah bahasa Java.

Berikut adalah *Script* pengujian 6 method pada aplikasi Scrum Project Management

#### b. Black Box Test

Pengujian *Black-box* dilakukan oleh Merlinda Wibowo, S.T., pengujian terhadap Black-box test tersebut dilakukan dengan menjawab daftar kuesioner mengenai apakah aplikasi sesuai dengan yang diharapkan dengan memberikan tanda ( V ) apabila sesuai dan tanda ( - ) jika tidak sesuai

#### c. Usability Testing

Pengujian SUS (System Usability Scale) dilakukan oleh end-user yang menggunakan aplikasi, yaitu mahasiswa MTP Teknik Informatika, Universitas Ahmad Dahlan, Yogyakarta.

## E. PENUTUP

### a. Kesimpulan

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

- a. Telah dihasilkan sebuah Aplikasi *mobile* yang dapat membantu mahasiswa matakuliah Manajemen Tugas Proyek (MTP) diprodi Teknik Informatika Universitas Ahmad Dahlan untuk memantau tugas-tugas pada manajemen proyeknya serta mendokumentasikan komunikasi selama manajemen tugas proyek berlangsung dengan *daily scrum*
- 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

Saran dari pengembang adalah agar aplikasi ini dapat menggunakan tampilan yang lebih baik lagi terutama pada *listview*. Serta perlu ditambahkan push notification yang lebih baik dengan menggunakan *Firebase Cloud Messaging*.

## F. DAFTAR PUSTAKA

- Fowler, Martin.(2006). Using an Agile Software Process with Offshore Development. Diunduh dari [www.it.uu.se/edu/course/homepage/acsd/ht03/Fowler.pdf](http://www.it.uu.se/edu/course/homepage/acsd/ht03/Fowler.pdf)
- Gray, Clifford F. and Larson, Eric W. (2006). Proyek Management: The Managerial Process 3th Edition, Singapore: The McGraw-Hill Companies. Inc.
- Holmes, Steve J. and Walsh, Robert T. (2005).Conducting Effective Proyek Management Maturity Assessment Interviews.IMSI TECH 2005.Integrated Management Systems, Inc., Ann Arbor, MI. Diunduh dari [www.imsi-pm.com/home/library/ conducting\\_assessment.pdf](http://www.imsi-pm.com/home/library/ conducting_assessment.pdf).
- Kerzner, Harold. (2001). Proyek Management: A Systems Approach to Planning, Scheduling, and Controlling - 7th Editions. Canada: John Wiley & Sons.
- Khan, A.R. et al., 2013. Web based Software Proyek Management using Android Mobile. *International Journal Of Next Generation Computer Applications*, 1(7), pp.1–5. Available at: <http://ijngca.com/Papers/IJNGCA25032013.pdf>.
- Noerlina. (2008). Perencanaan Manajemen ProyekSistem Informasi Dan Teknologi Informasi Online Bisnis, *Jurnal Piranti Warta*Vol.11 No.3 Agustus 2008: 440-450
- Schwaber K dan Sutherland J. (2013).*Scrum Guide ID*. Tersedia pada <http://www.scrumguides.org/>