

BAB I

PENDAHULUAN

1.1. Gambaran Umum Objek Penelitian

Objek penelitian dalam penelitian ini adalah tim *technology* dari salah satu divisi di PT. XYZ.

1.1.1. Sejarah Singkat Divisi PT. XYZ

PT. XYZ adalah salah satu perusahaan terbesar dan terintegrasi di Indonesia yang bergerak di bidang media. Berawal pada tahun 1989, PT. XYZ adalah sebuah perusahaan sekuritas yang bergerak di bidang jasa keuangan dan kini telah berkembang menjadi grup investasi yang terkemuka di Indonesia. PT. XYZ kini memiliki fokus kepada tiga jenis investasi yang strategis, yaitu media, jasa keuangan, properti dan satu bisnis pendukung yang memiliki kaitan dengan sektor jasa keuangan, yaitu investasi keuangan. PT. XYZ didirikan di Surabaya, Jawa Timur pada tanggal 2 November 1989. Bisnis inti PT. XYZ kini terdiri dari penyiaran nasional dan bisnis konten. Bisnis pendukung PT. XYZ meliputi media cetak, *online* dan radio.

PT. XYZ terus melakukan berbagai inovasi dengan tujuan meningkatkan keunggulan kompetitifnya sesuai dengan komitmen PT. XYZ untuk menyediakan dan memfasilitasi layanan televisi yang lebih baik dengan jangkauan yang lebih luas. Kemudian, PT. XYZ membangun sebuah divisi yang bergerak dalam bidang media *streaming* sehingga masyarakat akan lebih mudah untuk menyaksikan acara dan program televisi dimanapun dan kapanpun. Divisi yang didirikan oleh PT. XYZ akan menghasilkan sebuah *platform* untuk *streaming* yang dapat diunduh oleh masyarakat Indonesia maupun di luar Indonesia melalui *PlayStore* untuk pengguna *device* Android atau pun *AppStore* untuk pengguna *device* iOS.

Aplikasi *streaming* baru yang diciptakan oleh PT. XYZ ini tidak hanya menyediakan layanan *streaming* acara dan program televisi saja, namun dapat digunakan untuk *streaming* berita, film, music, dan lainnya. Media *streaming* ini

adalah konten video ataupun audio yang dikirimkan dalam bentuk yang sudah terkompresi melalui internet dan dapat diputar sehingga para pengguna aplikasi ini tidak perlu mengunduh file untuk memutar acara atau program yang diinginkan. Produk baru berupa aplikasi *streaming* ini diluncurkan secara resmi ketika PT XYZ menginjak usia 30 tahun. Banyak tim baru yang diciptakan untuk mendukung pengembangan aplikasi streaming ini, salah satunya adalah tim *technology*.

1.1.2. Divisi ABC di PT XYZ

Divisi ABC yang didirikan oleh PT XYZ pada tahun 2019 ini bertujuan untuk mengembangkan layanan AVOD, yaitu layanan *Video On Demand* berbasis iklan dan pengguna dapat secara gratis menikmati layanan tersebut. Layanan tersebut dibungkus dalam bentuk aplikasi yang terdapat di *Play Store* untuk Android dan *App Store* untuk iOS. Selain layanan *Video On Demand*, divisi ini juga mengembangkan layanan *live streaming* dari empat stasiun televisi yang populer. Divisi ini terdiri dari 200 karyawan yang tersebar di beberapa tim. Penulis memiliki tim *technology* sebagai responden untuk penelitian ini, yaitu sebanyak 54 orang.

Tim *technology* menggunakan metode *scrum* dalam menjalankan proses pengembangan perangkat lunak. Tim menggunakan sprint yang berjalan selama dua minggu. Tujuan tim *technology* menggunakan metode *scrum* karena metode *scrum* memiliki pendekatan adaptif sehingga mampu melakukan perubahan yang cepat dan efisien. Metode *scrum* memisahkan antara prioritas yang tinggi dengan prioritas yang rendah. Metode *scrum* ini memiliki komunikasi antar tim yang berkelanjutan sehingga memiliki tingkat transparansi yang tinggi antara klien dan juga tim.

1.2. Latar Belakang Penelitian

Saat ini perkembangan teknologi sudah sangat pesat, sehingga pemanfaatan teknologi sudah mencakup hampir ke seluruh bidang kehidupan, termasuk organisasi maupun industri digital. Memilih dan menentukan suatu metodologi dalam merancang dan membangun perangkat lunak bukanlah hal yang mudah untuk dilakukan. Pemilihan model pengembangan perangkat lunak yang akan

menentukan kualitas dari produk yang akan dikembangkan juga menentukan biaya dan kebutuhan lainnya.

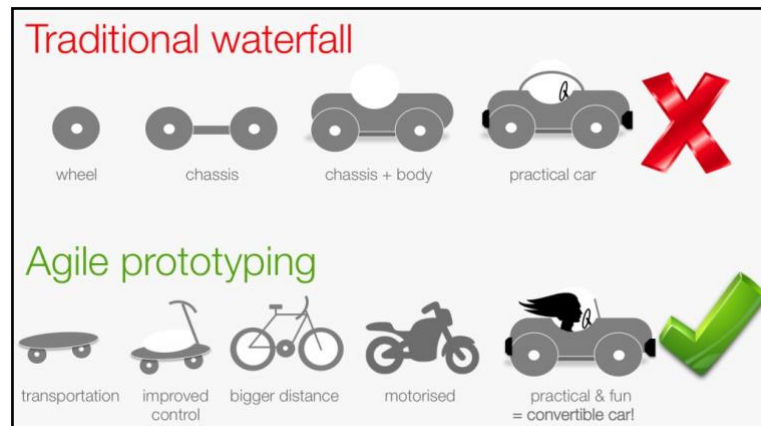
Pada awalnya, tim *technology* pada PT XYZ memilih model *Software Development Life Cycle* dengan metode *waterfall* untuk mengembangkan produk aplikasi karena dianggap mudah diimplementasi, menghasilkan kualitas sistem yang baik dan dokumen pengembangan sistem sangat terorganisir karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya. Metode *waterfall* merupakan *classic life cycle* yang menggunakan pendekatan sistematis, dimulai dari analisis kebutuhan hingga *maintenance* (Case Company Business Process Management Suite, 2012).

Berdasarkan hasil wawancara yang dilakukan kepada *project manager*, beberapa tim *backend engineer* dan *frontend engineer*, saat tim *technology* menggunakan metode *waterfall*, total durasi tim untuk menyelesaikan proyek pengembangan perangkat lunak tidak sesuai dengan estimasi yang sudah direncanakan karena proyek sering tidak mengikuti alur sekuensial, perubahan yang terjadi dapat menyebabkan perangkat lunak yang sudah selesai harus diubah seperti yang dapat dilihat pada lampiran 2. Klien harus menunggu produk hingga selesai dikembangkan dan tidak dapat mengubah *requirements* ditengah fase yang sedang berjalan. Pembagian *scope* tidak fleksibel dan terdapat waktu kosong bagi tim lain jika fasenya belum dimulai, misalnya fase yang sedang berjalan dilakukan oleh tim bisnis, maka tim lain menunggu dan ada waktu yang kosong. Tim menghasilkan dokumentasi terstruktur pada setiap fase yang dibutuhkan tim untuk pengembangan perangkat lunak selanjutnya atau melakukan *maintenance* lebih lanjut jika diperlukan.

Tim *technology* mengubah model pengembangan perangkat lunak menjadi metode *scrum*. Alasan tim *technology* mengubah model pengembangan perangkat lunak tersebut disampaikan oleh *project manager* bahwa metode *waterfall* berfokus pada kebutuhan yang sudah dibuat pada fase awal. Namun, metode *scrum* lebih fleksibel, yaitu melakukan pengembangan perangkat lunak dengan iterasi dengan waktu singkat dan berulang, melakukan penyesuaian kebutuhan dan perbaikan masalah pengguna tergantung dengan tingkat *priority* dan *severity* seperti yang

diilustrasikan pada Gambar 1.1. Iterasi dijalankan oleh tim selama dua minggu atau 10 hari kerja (Boebm, 2017).

Gambar 1.1. Ilustrasi Implementasi Metode *Waterfall* dan Metode *Scrum*



Sumber: Kniberg (2016)

Gambar 1.2 menunjukkan bahwa fase metode *scrum* dimulai dengan merancang kerangka perangkat lunak dan dilanjutkan dengan mengembangkan tampilan perangkat lunak. Metode ini memiliki kemampuan adaptasi yang cepat terhadap bentuk perubahan kebutuhan. Berjalan dengan iterasi, maka tidak ada waktu menunggu untuk para anggota tim seperti pada metode sebelumnya. Agenda yang akan dilaksanakan jika mengimplementasikan metode *scrum* yaitu *sprint planning*, *daily sprint*, *sprint review*, dan *retrospective*.

Gambar 1.2. Ilustrasi Implementasi Metode *Waterfall* dan Metode *Scrum*



Sumber: Dilek (2017)

Hasil wawancara kepada seluruh *project manager* dan beberapa anggota tim *technology* setelah satu tahun menggunakan metode *scrum* adalah terdapat beberapa kali iterasi yang tidak selesai sesuai dengan estimasi durasi yang sudah ditentukan pada *sprint planning* yang dapat dilihat pada lampiran 3. Seperti yang ditampilkan pada tabel di bawah ini, terdapat 12 dari 20 iterasi yang berjalan tidak sesuai rencana.

Tabel 1.1. Iterasi Metode *Scrum* Tidak Sesuai Rencana

No	Iterasi	Estimasi (hari kerja)	Realita (hari kerja)	Keterangan
1	Iterasi 5	10 hari kerja	13 hari kerja	Terlambat 3 hari
2	Iterasi 6	10 hari kerja	14 hari kerja	Terlambat 4 hari
3	Iterasi 7	10 hari kerja	15 hari kerja	Terlambat 5 hari
4	Iterasi 9	10 hari kerja	15 hari kerja	Terlambat 5 hari
5	Iterasi 12	10 hari kerja	13 hari kerja	Terlambat 3 hari
6	Iterasi 13	10 hari kerja	22 hari kerja	Terlambat 12 hari
7	Iterasi 14	10 hari kerja	15 hari kerja	Terlambat 5 hari
8	Iterasi 15	10 hari kerja	18 hari kerja	Terlambat 8 hari
9	Iterasi 16	10 hari kerja	14 hari kerja	Terlambat 4 hari
10	Iterasi 17	10 hari kerja	17 hari kerja	Terlambat 7 hari
11	Iterasi 18	10 hari kerja	13 hari kerja	Terlambat 3 hari
12	Iterasi 20	10 hari kerja	16 hari kerja	Terlambat 6 hari

Sumber: Data Sekunder (2021)

Dari tabel di atas, dapat dikatakan bahwa 60% dari iterasi yang sudah dilalui tidak berjalan sesuai dengan rencana yang sudah dibuat pada saat *sprint planning*. Berdasarkan data tersebut, teridentifikasi adanya permasalahan signifikan yang dialami oleh tim *technology*. Penelaah lebih lanjut dengan wawancara kepada anggota tim menunjukkan bahwa masalah tersebut terjadi karena adanya perubahan kebutuhan atau terjadinya kendala pada perangkat lunak yang dirasakan pengguna yang harus diselesaikan langsung di tengah iterasi tersebut, sehingga apa yang sudah dibangun oleh tim pengembang harus diubah kembali

mengikuti kebutuhan baru atau masalah pada perangkat lunak yang mendadak. Ketika permintaan perubahan kebutuhan yang terlalu banyak di tengah iterasi, terdapat penurunan produktivitas anggota tim karena terus melakukan perubahan pada sisi kode perangkat lunak. Masalah tersebut juga berdampak pada proses rilis ke market yang terlampau lama, sehingga tidak sedikit pengguna yang akhirnya memutuskan untuk melakukan *uninstall* perangkat lunak tersebut dikarenakan kendala pada perangkat lunak tidak kunjung diselesaikan dan dipublikasikan ke market.

Untuk menganalisis hasil survei, penulis menggunakan metode yang sama dengan penelitian yang telah dilakukan oleh Chow dan Cao (2008), yaitu *Critical Success Factor* (CSF). Chow dan Cao (2008) memberikan contoh penggunaan metode CSF dalam konteks manajemen proyek dan area proyek pengembangan perangkat lunak menggunakan model *Agile Software Development*. Dalam penelitian ini kami menggunakan metode identifikasi CSF yang sama untuk menganalisa pengaruh implementasi metode *scrum* terhadap efektivitas pengembangan perangkat lunak yang dilakukan oleh tim *technology* PT XYZ. Kemudian Subhas et. al melakukan review dalam upaya untuk memeriksa peringkat faktor keberhasilan penting yang dijelaskan sebelumnya dalam ulasan yang dilakukan oleh Chow dan Cao (2008). Hasil penelitian menunjukkan bahwa penggunaan metode CSF memiliki hubungan kritis terhadap keberhasilan manajemen proyek dan menyajikan sembilan faktor lagi yang memiliki pengaruh terhadap keberhasilan manajemen proyek, yaitu kerjasama klien, kepuasan klien, kewajiban klien, waktu pengambilan keputusan, budaya perusahaan, karakteristik orang, budaya masyarakat, dan mempersiapkan dan belajar. Sehingga penulis menggunakan *Critical Success Factor* untuk penelitian ini guna mendapatkan faktor mana saja yang memiliki pengaruh terhadap keberhasilan dan juga kegagalan dalam implementasi metode *scrum* untuk pengembangan perangkat lunak.

Dengan penerapan metode *scrum* pada tim *technology* PT XYZ ini, penulis tertarik untuk meneliti apakah metode *scrum* sudah berjalan efektif di tim *technology* PT XYZ. Penulis ingin mengukur pengaruh efektivitas *scrum* terhadap keberhasilan manajemen proyek TI yang diukur melalui dimensi *time*, *cost*, *quality*

dan *scope*. Keempat dimensi inilah diasumsikan dapat mewakili persepsi keberhasilan proyek. Selain itu, penelitian ini juga ingin mencari tahu faktor-faktor mana saja yang memiliki pengaruh signifikan terhadap efektivitas metode *scrum* di tim *technology* PT XYZ. Penelitian ini dilakukan dengan judul, “**Pengaruh Faktor-Faktor Kesuksesan Agile System Development Terhadap Efektifitas Pengembangan Produk Perangkat Lunak di PT XYZ dengan Menggunakan Metode Scrum**”.

1.3. Perumusan Masalah

Efektivitas dalam implementasi metode *scrum* dari *Agile Software Development* sebagai pengganti model *waterfall* menjadi fokus utama dari tim *technology* PT XYZ saat ini. Sesuai dengan perumusan masalah yang telah diuraikan, maka pertanyaan penelitian adalah sebagai berikut :

1. Apakah *Management Commitment* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?
2. Apakah *Organizational Environment* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?
3. Apakah *Team Environment* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?
4. Apakah *Team Capability* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?
5. Apakah *Customer Involvement* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?
6. Apakah *Project Management Process* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi

- (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?
7. Apakah *Project Definition Process* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?
 8. Apakah *Agile Software Techniques* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?
 9. Apakah *Delivery Strategy* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?
 10. Apakah *Project Nature* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?
 11. Apakah *Project Type* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?
 12. Apakah *Project Schedule* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*?

1.4. Tujuan Penelitian

Penelitian ini dilakukan dengan tujuan untuk menjawab pertanyaan penelitian yang telah diajukan sebelumnya, yaitu :

1. Mengetahui apakah *Management Commitment* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.
2. Mengetahui apakah *Organizational Environment* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi

- (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.
3. Mengetahui apakah *Team Environment* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.
 4. Mengetahui apakah *Team Capability* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.
 5. Mengetahui apakah *Customer Involvement* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.
 6. Mengetahui apakah *Project Management Process* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.
 7. Mengetahui apakah *Project Definition Process* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.
 8. Mengetahui apakah *Agile Software Techniques* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.
 9. Mengetahui apakah *Delivery Strategy* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.

10. Mengetahui apakah *Project Nature* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.
11. Mengetahui apakah *Project Type* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.
12. Mengetahui apakah *Project Schedule* memengaruhi efektivitas pengembangan perangkat lunak menggunakan metode *scrum* dilihat dari segi (a) *Perception Quality*, (b) *Perception Scope*, (c) *Perception Time*, dan (d) *Perception Cost*.

1.5. Manfaat Penelitian

Hasil dari penelitian ini diharapkan dapat menghasilkan Analisa pengaruh dari model *Agile Software Development*, khususnya metode *scrum*, terhadap efektivitas tim *technology* PT XYZ dalam mengembangkan perangkat lunak sehingga dapat memaksimalkan kelebihan dari metode *scrum* dan meningkatkan produktivitas tim *technology*. Penggunaan metode *scrum* diharapkan dapat memberikan manfaat untuk tim *technology* PT XYZ, yaitu pencapaian *project scope* yang semakin banyak oleh tim, menghasilkan produk dengan kualitas yang lebih baik karena dapat melakukan perubahan atau perbaikan sesuai dengan kebutuhan, kesesuaian waktu berjalannya proyek dengan rencana karena setiap iterasi akan ada penyesuaian *scope* perorang dengan waktu yang tersedia, dan banyaknya komunikasi antar tim sehingga mudah bagi *project manager* dalam melakukan pemantauan proyek kepada tim *technology*.

1.6. Sistematika Penulisan

Penyusunan tugas akhir ini menggunakan sistematika yang terbagi dalam uraian lima bab sebagai berikut :

1. BAB I Pendahuluan

Bab ini menguraikan tentang gambaran umum objek penelitian yaitu tim *technology* yang berada pada divisi ABC di PT XYZ dan juga latar belakang pemilihan judul.

2. BAB II Tinjauan Pustaka

Bab ini berisi teori-teori terkait penelitian dan penelitian terdahulu yang relevan dengan permasalahan dan penelitian. Bab ini menguraikan tentang *Agile Software Development*, metode *scrum*, PT XYZ dan lain-lain.

3. BAB III Metode Penelitian

Bab ini berisi karakteristik penelitian, alat pengumpulan data, variabel operasional, tahapan penelitian, populasi dan sampel.

4. BAB IV Hasil Penelitian dan Pembahasan

Pada bab ini dijelaskan mengenai analisis data dan pembahasan permasalahan yang sudah dirumuskan. Dalam bab ini dijabarkan hasil dari penelitian berupa analisa efektivitas model *Agile Software Development* menggunakan metode *scrum* terhadap tim *technology* yang ditimbang dari empat faktor, yaitu *quality*, *scope*, *timeliness* dan *cost*.

5. BAB V Kesimpulan dan Saran

Pada bab ini meliputi kesimpulan yang diambil berdasarkan hasil analisis data dan saran yang dapat dilakukan oleh peneliti selanjutnya.