

Developing Review Websites Using Feature Driven Development (FDD)

Marchelin Fau Hariono, Albert, Arvin Vinsensius, Adhi Kusnadi

Program Studi Teknik Informatika, Universitas Multimedia Nusantara (UMN) Tangerang, Indonesia

Adhi.kusnadi@umn.ac.id

Diterima 24 Oktober 2014

Disetujui 11 Desember 2014

Abstract – In this paper discussed a case study of the development of a website review . The discussion of this case study is limited to the implementation and use of the five steps in the development FDD review website . Moreover it is not a discussion of such budget , personal development , the success rate of software , and others . That reviews are steps in the development of the system , ie develop an overall model of phase , build a feature list , plan by features , design by features , build by feature . Conclusions that can be drawn from this discussion is the Feature Driven Development successfully used for software development carried out by the team , which will produce a good system . FDD will also make the user comfortable , where there is an attractive feature to assist companies in doing business . FDD does not focus on how to design a system that is considered costly and time . However , FDD prioritize how to execute a project with a faster and more efficient despite the implementation of the use of this method requires a lot of qualified human resources.

Index Terms - website review, feature driven development, FDD

I. PENDAHULUAN

Teknologi yang semakin maju dan berkembang mendorong industri teknologi untuk terus berinovasi dan mencari metode-metode terbaik untuk mengembangkan proyek dalam berbagai model. Keinginan klien akan cepatnya suatu proyek dikerjakan membuat banyak pengembang memilih untuk menggunakan *agile process* dalam pengerjaan proyeknya. *Feature Driven Development* (FDD) merupakan salah satu *agile process* yang mengacu pada dokumen kode dalam bentuk fitur yang memudahkan para pengembang dan klien mendapatkan gambaran akan sistem yang sedang dikerjakan.

Pembahasan pada tulisan akan menggunakan sebuah studi kasus tentang pengembangan sebuah *website review*. *Review* yang dimaksud di sini adalah bersifat umum atau general seperti melakukan

review terhadap sebuah produk atau jasa. Tujuan dari pembahasan ini adalah memberikan gambaran dasar bagi khususnya mahasiswa atau pemula yang ingin menggunakan metode FDD dalam pembuatan sebuah *website review* sehingga mereka dapat mengetahui apa saja yang harus dipersiapkan dan diperlukan dalam menjalankan proyek mereka.

Pembahasan mengenai studi kasus ini terbatas hanya pada implementasi dan penggunaan lima langkah FDD dalam pengembangan *website review*. Segala bentuk jenis yang terkait lainnya di luar penjelasan (anggaran, personal development, tingkat keberhasilan *software*, dan sebagainya) tidak termasuk dalam lingkup bahasan.

II. FEATURE DRIVEN DEVELOPMENT

Feature Driven Development (FDD) merupakan proses yang didesain dan dilaksanakan untuk menyajikan hasil kerja secara berulang-ulang dalam waktu tertentu dan dapat diukur. FDD merupakan pendekatan yang mengacu pada pembuatan sistem menggunakan metode yang mudah dimengerti dan diimplementasikan, teknik *problem solving*, dan pelaporan yang mudah dimengerti dan dikontrol oleh *stakeholders*[6]. Dengan metode FDD, pemrogram diberikan informasi yang cukup dan alat bantu untuk menyelesaikan proyek. Pemimpin proyek dan manajer diberikan informasi berdasarkan waktu mengenai tim dan proyek yang sedang berjalan untuk menghindari risiko yang terjadi. Pelaporan menjadi lebih mudah, tidak membebani, periodik, dan akurat. Pengguna juga dapat secara langsung melihat bagian mereka sebagai hasil *progress* proyek dan memberikan *feedback* dalam tahap pengembangan yang bertujuan untuk menciptakan sistem sesuai keinginan klien.

Terdapat lima kelebihan yang dimiliki oleh metode *Feature Driven Development* yang menjadi nilai tambah bagi para pengembang sistem saat mengerjakan proyeknya. Pertama, *tangible results*

rather than process pride. Proses dalam metode FDD mengutamakan sesuatu yang dapat diukur ketimbang proses-proses perancangan yang rumit dan menghabiskan waktu, sumber daya, dan biaya. Pada saat merancang proyek, penjadwalan langsung diarahkan ke dalam bentuk fitur. Kedua, *a system for building system is necessary.* Sistem yang dibangun harus rapi dan kuat agar para pengembang dapat bekerja dan menghasilkan sebuah sistem yang diharapkan oleh klien. Ketiga, *simple is better.* Perancangan dibuat sesederhana mungkin, tetapi dapat memenuhi semua persyaratan yang diberikan oleh klien sehingga mereka mendapatkan gambaran sistem dengan mudah. Keempat, *process steps should be obviously valuable to each team member.* Setiap langkah atau proses selama pengembangan sistem harus dapat dinilai dan terukur bagi para pengembang sistem. Selain itu, desain kode juga menjadi lebih mudah dan efektif pada saat pemeriksaan. Terakhir, *good processes move to the background.* Semua proses lebih baik dikerjakan di belakang sehingga pengerjaan dengan metode FDD terlihat lebih sederhana.

Beberapa kekurangan yang dimiliki oleh metode *Feature Driven Development* yang diperoleh dari hasil studi analitik adalah sebagai berikut. Jumlah pekerja dalam proyek banyak (10-250 orang) yang dibagi ke dalam beberapa divisi. Semakin banyak pekerja, biaya yang dikeluarkan semakin tinggi. Selain itu, FDD juga lebih menekankan pengembang dengan keterampilan tinggi dan sulit untuk mencari pengembang dengan kriteria tersebut. Dalam FDD terdapat *class owner* yang menangani setiap fitur. Masalahnya adalah ketika fitur A memiliki dependensi terhadap fitur B dan fitur B mengalami perubahan, fitur A harus menunggu kepastian dari fitur B yang menyebabkan mundurnya jadwal proyek. Pada intinya, FDD bersifat individual, maka dapat terjadi saling tunggu antar fitur yang terkait. FDD membatasi penambahan fitur kurang dari 10% dari total waktu, biaya, dan kualitas. Perubahan fitur hanya dilakukan pada sebuah proses dan keterlibatan klien hanya pada beberapa tahapan. Jumlah jam kerja FDD tidak terikat sehingga dapat menyebabkan para pengembang tidak bekerja secara optimal di pertengahan, tetapi pada akhir jadwal mereka akan bekerja lebih ekstra.

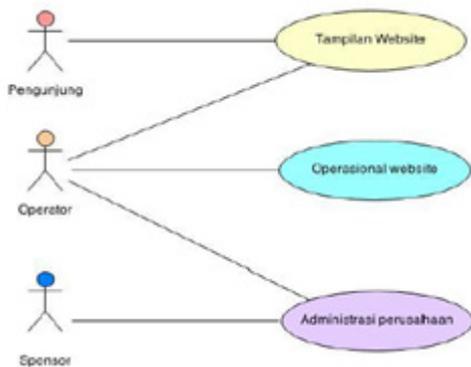
III. CARA KERJA DAN IMPLEMENTASI

Sebelum dimulai, dalam studi kasus ini akan dilihat bagaimana cara membangun sebuah *website review* dengan metode FDD. Berikut adalah studi kasusnya, seorang klien berencana akan membangun sebuah perusahaan yang bergerak di

bidang *review* produk elektronik (*gadget*) dan karena era internet yang sudah menjamur, klien tersebut ingin usahanya juga bergerak melalui *online*. Terpikirlah untuk menciptakan sebuah *website review*. Akhirnya klien ini mencari sebuah perusahaan profesional untuk dibuatkan sebuah *website review*. Setelah berkonsultasi dengan kepala proyek, dihasilkan beberapa ide atau kebutuhan yang diperlukan klien pada *website review*nya antara lain klien bersedia mengeluarkan budget yang besar, tetapi kualitas yang dihasilkan harus istimewa dan interaktif. Waktu yang disepakati untuk menyelesaikan proyek ini oleh pihak pengembang adalah dua bulan. Singkat cerita, sang pemimpin proyek mengambil kesimpulan bahwa mereka akan menggunakan metode FDD dalam proyek kali ini sebab klien meminta untuk menciptakan sebuah *website* yang interaktif atau kaya dengan *feature*. Langkah paling awal, sang pemimpin akan membagi anggotanya ke dalam beberapa kelompok (*grup*) dalam hal pembagian tugas pelaksanaan proyek ini. Pada proyek ini mereka akan bekerja dengan keseluruhan anggota berjumlah enam orang termasuk pemimpin proyek. Mereka semua dalam kelompok besar akan dikumpulkan bersama untuk membahas cara kerja mereka menggunakan metode FDD. Oleh karena itu, mereka semua akan menjalani beberapa tahap seperti di bawah ini.

A. *Develop an Overall Model Phase*

Pada tahap awal ini, seluruh anggota pengembang wajib sudah mengetahui dasar dan cara pengembangan dengan *agile process* khususnya dengan metode FDD. Kemudian mereka diminta untuk setiap grup memikirkan, merancang, dan mengajukan apa saja yang mereka harapkan dan perlukan dalam membuat sebuah *website review* yang baik. Setelah semua hasil dikumpulkan, maka mereka akan menggabungkan semuanya itu ke dalam sebuah gambaran secara garis besar yang mencakup atau menggambarkan keseluruhan sistem yang akan mereka kembangkan. Biasanya mereka dapat menggunakan *tools* yang tersedia baik *online* atau *offline* untuk membuat sebuah diagram tentang keseluruhan proses mereka. Diagram yang biasa diperlukan di sini adalah *Use Case Diagram*.

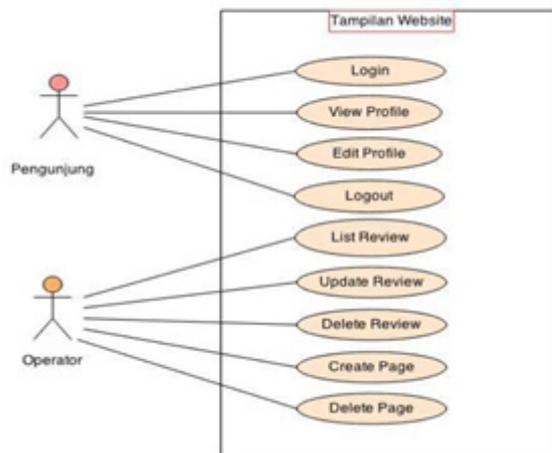


Gambar 1. *Use Case Diagram* “Utama”

Use case pada gambar 1 merupakan use case diagram “Utama” yang merupakan tujuan utama yang akan dikembangkan. Berdasarkan gambar 1, itulah yang akan dicapai dalam tahap pertama ini (*develop an overall*) di mana semua hal yang akan dikembangkan akan disatukan dan membentuk sebuah perencanaan yang matang secara garis besar.

B. *Build a Feature List*

Feature list adalah apa yang dilihat klien untuk validitas dan kelengkapan sistem. Fitur dalam langkah ini berbasis customer bukan teknologi. Bahasa yang digunakan sesederhana mungkin agar klien paham. Pada tahap selanjutnya setelah menentukan keseluruhan rangkaian sistem, kini para pengembang harus mengidentifikasi fitur-fitur apa saja yang dapat di jadikan *list* pada setiap modul yang dihasilkan. Pada kasus ini sistem memiliki modul “*Tampilan Website*”. Jika dijabarkan modul ini memiliki beberapa feature seperti *Login*, *View Profile*, *Edit Profile*, *Logout*, *List Review*, *Update Review*, *Delete Review*, *Create Page*, *Delete Page*. Pada tahap ini pemimpin proyek (kepala) menargetkan untuk menghasilkan minimal 25 feature untuk proyek ini. Tahap ini bisa diimplementasikan juga dengan menggunakan *Use Case Diagram*. Pembuatan *Use Case* bisa dilakukan dengan menggunakan *software* seperti tahap pertama atau penulisan secara manual (tulisan tangan).

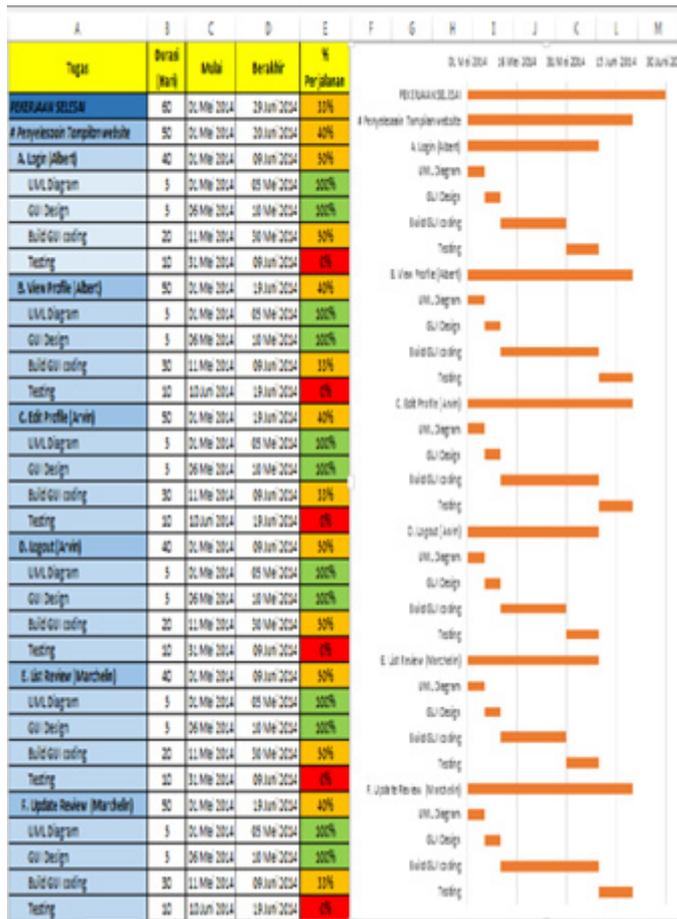


Gambar 2. *Use Case Diagram* “*Tampilan Website*”

Use case pada gambar 2 merupakan hal-hal apa saja yang akan dikerjakan dan dikembangkan oleh kelompok (*tampilan website*) dan menjelaskan apa saja feature yang akan dikembangkan pada setiap modul. Hasil ini merupakan penjabaran yang dilakukan oleh setiap kelompok sesuai dengan modul yang mereka kembangkan. Pada kesempatan kali ini, kami hanya menampilkan penjabaran dari *Use Case* “*Tampilan Website*” saja. Untuk kedua *Use Case* lainnya (*operasional dan administrasi*) cara penjabaran dan pembuatannya juga akan sama dengan gambar 2.

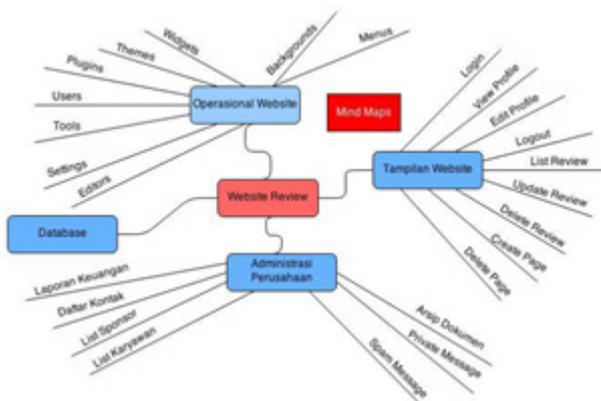
C. *Plan by Features*

Pada tahap ketiga ini merupakan tahap yang paling penting karena semua perencanaan pengembangan harus ditentukan di sini. Semua kelompok harus membuat dokumentasi terhadap apa saja yang telah mereka buat dalam modul. Setiap modul harus ditentukan waktu yang dibutuhkan menyelesaikannya dengan penjabaran masing-masing feature. Pemimpin proyek akan mengumpulkan semua estimasi dari masing-masing kelompok dan kemudian akan membuat sebuah *list* atau agenda waktu secara keseluruhan. Hal itu dapat dilakukan dengan menggunakan *grantt chart* atau *mind maps* atau keduanya. Fungsi dibuatnya *grantt chart* dan *mindmaps* adalah untuk membantu para pengembang melihat keseluruhan progres yang telah berjalan lebih baik.



Gambar 3. Gantt Chart

Bisa kita lihat dari kedua gambar di atas, *ganttt chart* dan *mind maps* akan membantu para pengembang lebih mudah untuk mengetahui tugas mereka masing-masing. Setiap modul digambarkan dengan jelas beserta fitur-fitur yang dibutuhkan.

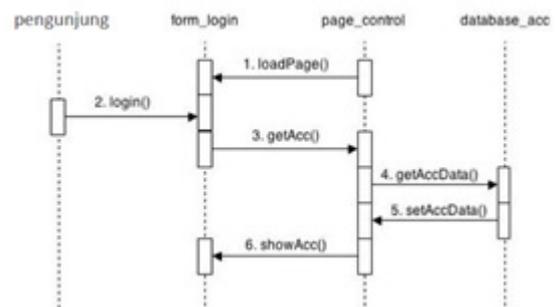


Gambar 4. Mind Maps

D. Design by Features

Setiap fitur dibuatkan *sequence diagram* dan *class diagram* untuk menunjukkan kepada klien bagaimana sebuah sistem bekerja sehingga jika ada kebingungan dan ketidaksetujuan dapat ditanggung para pengembang pada awal pengerjaan sistem.

Sebagai contoh, kami akan memberikan gambar *sequence diagram* untuk *user*: pengunjung saat mereka melakukan *login*. Untuk *sequence diagram* lainnya serupa dengan diagram berikut ini,



Gambar 5. Sequence Diagram

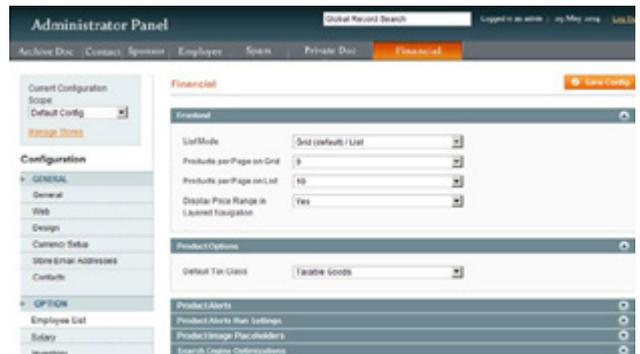
Untuk menggambarkan atribut dan metode apa saja yang dimiliki sistem *website review* dapat menggunakan *class diagram*. Contoh *class diagram* direpresentasikan sebagai berikut.



Gambar 6. Class Diagram

E. Build by Feature

Pada akhir tahapan FDD, pengembang membangun sistem yang sudah dirancang dengan menggunakan bahasa pemrograman dan *tools* yang sesuai. Mereka juga membuat *user interface* dari sistem tersebut dan membangun *server*.



Gambar 7. User Interface

IV. SIMPULAN DAN SARAN

A. Simpulan

Simpulan yang dapat diambil dari pembahasan ini adalah *Feature Driven Development* berhasil digunakan untuk pengembangan *software* yang dilakukan oleh tim yang akan menghasilkan sebuah sistem yang baik. Ini sangatlah berguna untuk klien dalam membantu pekerjaan, hal ini yang disebut sebagai 'fitur'. FDD berfokus pada pengembang untuk rekapitulasi hasil kerja setiap dua minggu. FDD juga akan membuat nyaman pemakai, dimana ada sebuah fitur yang atraktif untuk membantu perusahaan dalam menjalankan bisnisnya. FDD tidak berfokus pada bagaimana merancang sebuah sistem yang dianggap memakan banyak biaya dan waktu. Namun, FDD lebih mengutamakan bagaimana mengeksekusi sebuah proyek dengan waktu yang lebih cepat dan efisien meskipun pada implementasinya penggunaan metode ini membutuhkan banyak sumber daya manusia yang *qualified*.

B. Saran

Dapat dicari lagi metode lain untuk mengatasi kelemahan FFD yaitu lebih mengutamakan kecepatan dibanding biaya.

DAFTAR PUSTAKA

- [1] Arimbawa, I Wayan Agus. 2013. "*FeatureDriven Development (FDD), Apakah Bisa Disebut Agile?*". Dalam <http://agusarimbawa.files.wordpress.com/>. Diakses tanggal 24 oktober 2014
- [2] Danik Wahyuningsih.2012." Pengaruh Strategi Pembelajaran Aktif *Mind Maps* Terhadap Pemahaman, Sikap dan Keterampilan dari Hasil Belajar Biologi Siswa Kelas XI IPA".
- [3] Firdaus, Adila, Imran Ghani, dkk. 2013. "*Developing Secure Websites Using Feature Driven Development (FDD): A Case Study*". Dalam *Journal of Clean Energy Technologies*, Vol. 1, No. 4, Oktober 2013.
- [4] Goyal, Sadhna. 2007. "*Agile Techniques for Project Management and Software Engineering*". Dalam www.csis.pace.edu
- [5] Kamarujung. 2013. "*Pengertian FDD*". Dalam <http://kamarujung.blogspot.com/>. Diakses 24 oktober 2014
- [6] Palmer, Stephen R. dan John M. Felsing, *Practical Guide to Feature-Driven Development*, Prentice Hall, 2001.
- [7] Sari, Mutia. 2014. "Copy of FDD-SPI". Dalam <http://prezi.com/>.
- [8] Danik Wahyuningsih.2012." Pengaruh Strategi Pembelajaran Aktif *Mind Maps* Terhadap Pemahaman, Sikap dan Keterampilan dari Hasil Belajar Biologi Siswa Kelas XI IPA".