

Rancang Bangun Website Perpustakaan Digital Untuk Alternatif Perpustakaan Konvensional Menggunakan Metode *Extreme Programming*

1st Kemal Indra Kusuma
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia

kemalindrak@student.telkomuniversity.
ac.id

2nd Rachmadita Andreswari
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia

andreswari@telkomuniversity.ac.id

3rd Dita Pramesti
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia

ditapramesti@telkomuniversity.ac.id

Abstrak—Perpustakaan merupakan tempat umum untuk mencari buku atau sekedar mencari referensi. Namun semenjak virus covid menyebar di seluruh dunia pada tahun 2019, pemerintah menutup semua tempat umum untuk mencegah penyebaran virus salah satunya perpustakaan. Dikarenakan oleh hal ini banyak masyarakat umum bingung bagaimana cara untuk membaca buku secara gratis dalam keadaan virus COVID-19 masih berkeliaran. Oleh karena itu diperlukan suatu solusi agar masyarakat masih dapat membaca buku secara gratis seperti pada perpustakaan umum. Solusi tersebut ialah membuat suatu website yang berisikan buku digital sehingga masyarakat umum dapat membaca buku kapan saja dan dimana saja tanpa harus keluar rumah. Metode pengembangan website akan menggunakan salah satu metode agile, yaitu extreme programming (XP). Metode extreme programming (XP) cocok diterapkan pada proyek berskala kecil dan hanya membutuhkan satu sampai tiga orang saja. Selain metode pengembangan diperlukan juga metode evaluasi untuk menguji kelayakan website. Metode evaluasi yang akan digunakan ada dua yaitu blackbox testing untuk menguji antar muka dan fungsi website dan load testing untuk menguji kapasitas website pada beberapa pengguna. Hasil dari pengujian tersebut akan dijadikan saran untuk pengembangan lebih lanjut agar website menjadi lebih efektif.

Kata kunci— web, extreme programming, blackbox testing, load testing

I. PENDAHULUAN

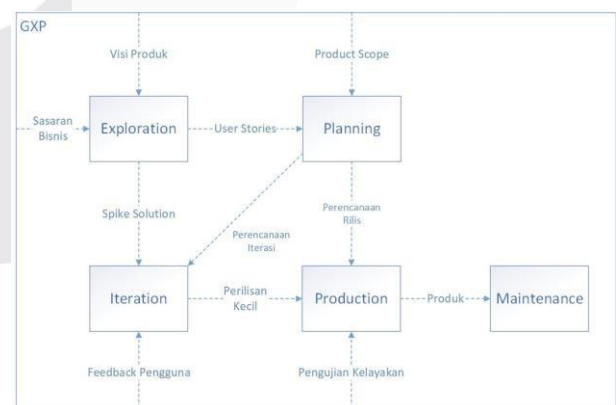
Pandemi COVID-19 sudah menyebar di seluruh dunia termasuk Kota Bandung serta sudah menjangkiti warga kota dan Kabupaten Bandung. Warga yang terinfeksi biasanya langsung diisolasi ke rumah sakit untuk mendapat perawatan intensif. Menurut data dari pusicov (covid19.bandung.go.id) pada tanggal 26 Oktober 2020 sudah terdapat 213 orang terinfeksi virus corona di Kota Bandung dan sebanyak 81 orang yang sudah meninggal dunia. Walaupun sempat turun, namun jumlah orang yang terinfeksi masih terus bertambah setiap 2 hari sehingga banyak tempat umum yang terpaksa ditutup sampai dengan pandemi ini berakhir. Upaya penutupan tersebut dilakukan untuk menekan angka

pertumbuhan pandemi COVID-19. Salah satu tempat umum yang terkena dampak pandemi COVID-19 adalah perpustakaan kota atau kabupaten yang masih bersifat konvensional atau tradisional. Oleh karena itu penulis mengusulkan sebuah solusi yakni membangun sebuah perpustakaan digital berbasis *website*. Perpustakaan digital adalah sebuah sistem yang memiliki berbagai layanan dan objek informasi yang mendukung akses objek informasi tersebut melalui perangkat digital[1]. Dalam pengembangan perpustakaan digital berbasis *website* ini diperlukan langkah – langkah atau sebuah metode untuk mempermudah proses pengembangan. Metode yang akan digunakan pada pengembangan *website* perpustakaan digital ini ialah *extreme programming* (XP).

II. KAJIAN TEORI

A. Extreme Programming

Extreme Programming merupakan salah satu metode *agile* yang memiliki beberapa langkah yang diharapkan dapat memberikan hasil yang maksimal dan dapat lebih membantu pada proses yang dibutuhkan [2]. Siklus *extreme programming* memiliki lima fase pengembangan seperti yang digambarkan pada Gambar 1, yaitu *exploration*, *planning*, *iteration*, *production* dan *maintenance* [3].



GAMBAR 1
FASE EXTREME PROGRAMMING

1. *Exploration*

Pada fase ini dilakukan pendefinisian fitur serta fungsi dari perangkat lunak yang akan menjadi gambaran dasar dari perancangan perangkat lunak yang akan dikembangkan. Hasil akhir dari fase ini ialah sekumpulan data kebutuhan pengguna yang akan digunakan pada fase selanjutnya.

2. *Planning*

Pada fase ini dilakukan pemilihan kebutuhan perangkat lunak sesuai permintaan pengguna yang nantinya akan menghasilkan rencana rilis serta iterasi. Rencana rilis menggambarkan fitur yang akan dikembangkan dan rencana iterasi merupakan tahapan yang akan dilakukan pada perancangan perangkat lunak tersebut.

3. *Iteration*

Fase ini merupakan pengembangan solusi dengan melakukan eksekusi dari fase perencanaan melalui pembuatan rancangan arsitektur perangkat lunak yang akan dibangun.

4. *Production*

Pada fase ini dilakukan pengujian perangkat lunak yang telah dikembangkan kepada target user dan melihat sejauh mana perangkat lunak itu membantu user berdasarkan hasil pada fase *exploration*.

5. *Maintenance*

Fase ini memfokuskan layanan perbaikan, jika terdapat kekurangan dari perangkat lunak yang telah dipublikasikan kepada user serta memastikan apakah kekurangan tersebut bisa diatasi.

B. UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) merupakan suatu pemodelan secara visual yang ditujukan untuk menjadi sarana perancangan suatu sistem berorientasi objek [4]. UML mempunyai sembilan *diagram* yang dapat dibangun pada saat pengembangan yaitu *class diagram*, *package diagram*, *use case diagram*, *sequence diagram*, *communication diagram*, *statechart diagram*, *activity diagram*, *componen diagram* dan *deployment diagram* [5].

C. Pengembangan *Website*

Web atau yang biasa disebut dengan *World Wide Web* merupakan suatu metode untuk menampilkan suatu atau beberapa informasi pada internet, baik berupa teks, gambar, suara maupun video [6]. Sedangkan menurut (Nur, Wijaya, & Bustami, 2020) pengembangan ialah suatu usaha untuk membuat atau memperbaiki sehingga dihasilkan produk yang lebih bermanfaat dan mempunyai mutu yang lebih baik [7].

Pada kedua penjelasan di atas dapat disimpulkan bahwa pengembangan web ialah suatu metode untuk membuat atau memperbaiki web yang berisikan sekumpulan informasi yang terhubung pada internet dengan tujuan untuk menghasilkan web yang lebih baik.

III. METODE

Untuk metode pengujian ini penulis menggunakan *blackbox testing* serta *load testing*. *Blackbox testing* merupakan pengujian perangkat lunak yang menguji spesifikasi fungsional perangkat lunak tersebut tanpa melihat pengujian pada desain serta kode program yang dibangunnya [8]. Sedangkan *load testing* adalah sebuah pengujian analisa perilaku sistem disaat sistem berada pada beban tertinggi [9].

Alasan pemilihan metode ini dikarenakan pada penelitian yang berjudul "PENERAPAN *EXTREME PROGRAMMING* DALAM MEMBANGUN APLIKASI PENGADUAN LAYANAN PELANGGAN (PaLaPa) PADA PERGURUAN TINGGI" menyimpulkan bahwa penerapan *extreme programming* pada pengembangan *website* dapat dilakukan secara cepat dengan jumlah anggota yang minim, serta hasil dari pengujian *website* memenuhi *acceptance criteria* pada *user story* [10]. Kemudian pada penelitian berjudul "Pengembangan Sistem Informasi Perpustakaan menggunakan Metode Extreme Programming (Studi pada: SMK 1 Muhammadiyah Malang)" menghasilkan tingkat kebenaran pada testing metode *white-box* sebesar 100% ditinjau dari hasil valid dari setiap *test case* yang diujikan dan tidak adanya jawaban "tidak" pada hasil pengujian *User Acceptance Testing* (UAT) [11].

IV. HASIL DAN PEMBAHASAN

A. *Exploration*

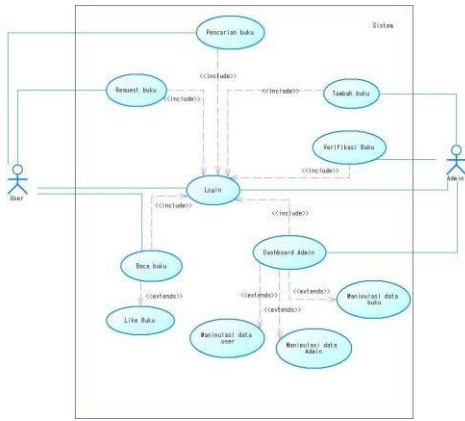
Pada tahap ini penulis merumuskan perumusan masalah melalui wawancara kepada pihak DISPUSIP mengenai perpustakaan konvensional serta data dari studi pustaka. Kemudian hasil dari kedua pencarian data tersebut tidak hanya menghasilkan rumusan masalah, namun juga menjadi *requirement* untuk penelitian ini.

B. *Planning*

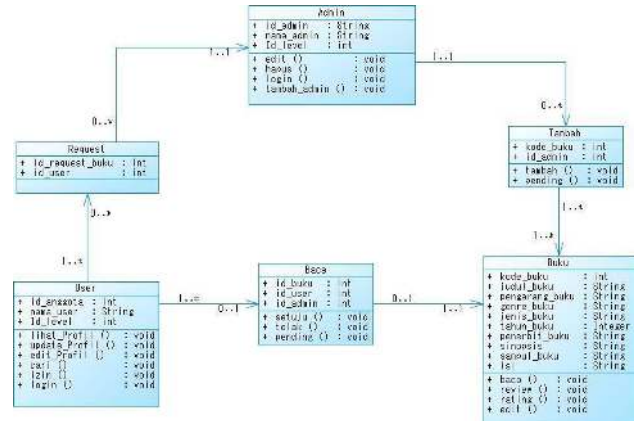
Pada tahap ini penulis membuat *planning* berupa fitur yang akan dimasukkan kedalam *website* yang dikembangkan serta bagaimana cara kerja fitur tersebut. Fitur utama yang dipakai pada *website* ialah peminjaman buku, permintaan penambahan buku dan penambahan buku.

C. *Iteration*

Pada tahap ini penulis mulai mengembangkan *website* berdasarkan perencanaan yang telah dibuat pada tahap *planning* sebelumnya. Tahapan awal yang dilakukan terlebih dahulu ialah *design*. *Design* yang dibuat terlebih dahulu ialah UML *diagram*. *Diagram* yang dipakai pada penelitian ini ialah *use case diagram*, *activity diagram*, *class diagram* dan *sequence diagram*. *Use case diagram* yang dibagi menjadi dua bagian yaitu pengguna dan admin.



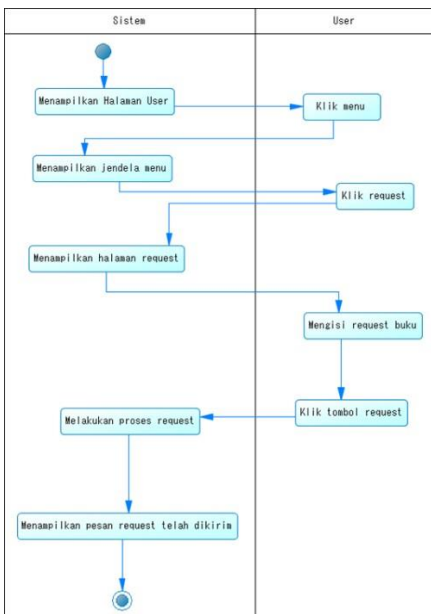
GAMBAR 2
USE CASE WEBSITE PERPUSTKAAN DIGITAL



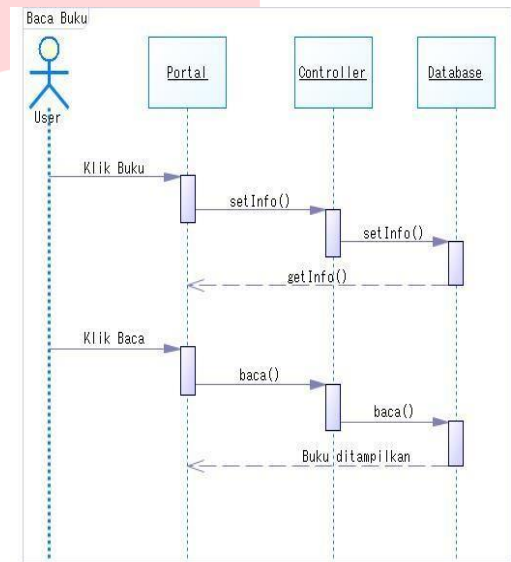
GAMBAR 4
CLASS DIAGRAM WEBSITE PERPUSTKAAN

Dari *use case diagram* yang digambarkan pada Gambar 2, dua aktor tersebut mempunyai beberapa *case* yang berbeda seperti pengguna yang dapat meminjam buku dan meminta penambahan buku serta admin yang dapat memanipulasi data buku dan menambahkan buku. Setelah itu juga terdapat 14 *activity diagram* yang telah dibuat pada tahap ini. Salah satu contoh *activity diagram* yang telah dibuat dapat dilihat pada Gambar 3.

Interaksi pada *class diagram* tersebut digambarkan lebih detail pada *sequence diagram*. Penelitian ini menghasilkan tujuh *sequence diagram*. Salah satu contoh *sequence diagram* yang dibuat dapat dilihat pada Gambar 5.



GAMBAR 3
ACTIVITY DIAGRAM REQUEST BUKU



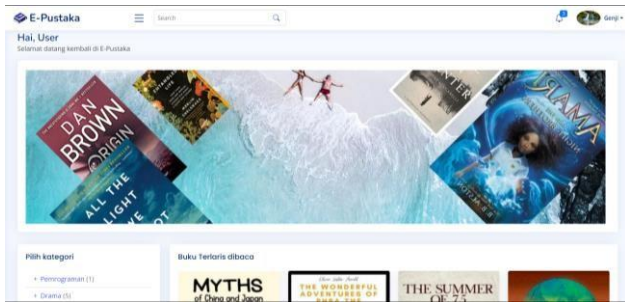
GAMBAR 5
SEQUENCE DIAGRAM BACA BUKU

Setelah perancangan UML *diagram*, maka pembangunan *website* dapat dijalankan mulai dari melakukan *coding* serta pembangunan tampilan *website*.

D. Production

Selanjutnya dibuatlah *class diagram* yang merupakan penggambaran konseptual dari *website* yang dikembangkan serta menjadi model untuk penyimpana data. *Class diagram* dibentuk dari 3 *class* utama yaitu admin, user, dan buku. Kemudian juga terdapat 3 *class* yang menjadi fungsi untuk mengambil data dari 3 *class* utama. Detail relasi dari setiap *class* tersebut dapat dilihat dari Gambar 4.

Pada tahap ini *website* telah dikembangkan dan siap untuk dilakukan *testing*. Selain fungsi yang sudah berjalan, tampilan *website* juga sudah dibuat agar *website* menjadi lebih menarik. Salah satu contoh tampilan *website* yang sudah dibangun dapat dilihat pada Gambar 6 yang merupakan *homepage website* tersebut.



GAMBAR 6
HOMEPAGE WEBSITE PERPUSTAKAAN

Setelah pembangunan baik dari *back-end* serta *front-end*, maka pengujian pun dilakukan mulai dari pengujian *blackbox testing* lalu *load testing*. Pada pengujian *blackbox testing* dilakukan pembuatan *test case* terlebih dahulu. Kemudian pengujian dilakukan berdasarkan *test case* tersebut. Hasil yang dikeluarkan akan dicatat untuk dijadikan kesimpulan. *Blackbox testing* ini dilakukan untuk menguji fungsionalitas dari *website* tersebut agar memastikan fungsi tersebut berjalan dengan baik. Dari 24 *test case* yang digunakan, hasil dari semua pengujian tersebut menghasilkan status sukses yang menandakan bahwa *website* berjalan sesuai dengan ekspektasi.

Selanjutnya *website* menjalani pengujian kedua yaitu *load testing*. Pada *load testing* dilakukan pengujian muatan pada sebuah halaman *website* dan akan di uji menggunakan pengguna virtual dengan jumlah yang berbeda. Kemudian hasil dari pengujian tersebut akan memperlihatkan berapa milidetik halaman tersebut menangani setiap pengguna serta jumlah *error* yang didapatkan.

Load testing ini dilakukan untuk mengetahui jumlah maksimal pengguna yang dapat ditampung pada suatu halaman *website* tersebut. Di uji sembilan halaman *website* dengan jumlah pengguna sebanyak 1000 pengguna. Hasil yang didapatkan jika semua halaman tersebut di totalkan, akan diperoleh total pada permintaan perdetik sebanyak 476.52 detik. Sedangkan pada jumlah waktu permintaan yang dilayani sebanyak 172820 milidetik atau 172.82 detik. Kemudian untuk total semua *error* pada halaman yang diuji cobakan hanya mendapatkan total 5 *error* yang ditemukan.

TABEL 1
HASIL LOAD TESTING

Label	Jumlah request	Permintaan per detik (rata – rata)	Jumlah waktu permintaan yang dilayani	Error
Landpage	1000	44,58	18970 ms	0
Homepage	1000	57,90	17206 ms	0
Detail buku (terkunci)	1000	37,06	24741 ms	1
Detail buku (proses)	1000	74,61	11193 ms	0
Detail buku (terbuka)	1000	34,98	26303 ms	0
Baca buku	1000	32,41	30694 ms	3

Daftar baca	1000	78,84	11251 ms	0
Form buku	1000	70,84	12740 ms	1
Daftar buku	1000	45,30	19722 ms	0
Total	995	476,52	172820 ms	5

Setelah melakukan percobaan pengujian kembali dengan memasukan 2000 permintaan yang bersamaan pada halaman *landpage* ditemukan 1 *error*. Kemudian dinaikan kembali sampai dengan 5000 permintaan dan ditemukan 2 *error*. Terakhir dinaikan kembali dengan jumlah yang besar yaitu 10000 permintaan dan mendapatkann *error* sebanyak 7. Hal tersebut menunjukkan bahwa sistem hanya dapat menangani permintaan secara maksimal yaitu 1000 permintaan. Untuk hasil yang lebih detail dapat dilihat melalui Tabel 1.

V. KESIMPULAN

Kesimpulan yang dapat ditarik dari penelitian ini ialah rancangan perpustakaan digital yang telah dikembangkan adalah rancangan perpustakaan yang dibangun dengan menggunakan metode *extreme programming* dalam bentuk *website* yang dapat diakses pada laptop dan Personal Computer (PC). Kemudian Alur proses bisnis utama pada perpustakaan digital menggunakan alur proses bisnis perpustakaan secara digital (*online*) melalui *website*. Alur ini dinilai lebih baik apabila dibandingkan dengan alur proses bisnis perpustakaan konvensional (*offline*). Sedangkan alur proses bisnis perpustakaan konvensional (*eksisting*) mengharuskan peminjam datang ke perpustakaan untuk meminjam dan mengembalikan buku koleksi secara fisik. Lalu Implementasi metode pengembangan *website extreme programming* perpustakaan digital dilakukan dengan menggunakan sumber daya mulai dari saat perancangan, pembangunan dan pengujian *website* yang menggunakan metode *extreme programming*. Jika dilihat dari jumlah tenaga kerja yang digunakan, implementasi *website* ini tidak terlalu memakai banyak tenaga kerja karena *extreme programming* hanya memerlukan 1 – 3 orang saja. Kemudian Uji fungsionalitas *website* perpustakaan digital menggunakan *blackbox testing* dan mendapatkan hasil yang sukses pada setiap fitur yang diuji. Kemudian pada *load testing*, diperoleh hasil bahwa *website* dapat menampung pengguna sebanyak 1000 pengguna secara bersamaan tanpa adanya *error*.

REFERENSI

- [1] S. Adrianto *et al.*, 'PERANCANGAN APLIKASI PERPUSTAKAAN DIGITAL'.
- [2] I. Carolina, A. Supriyatna, J. Kamal Raya No, and R. Barat Cengkareng Jakarta Barat, 'PENERAPAN METODE EXTREME PROGRAMMING DALAM PERANCANGAN APLIKASI PERHITUNGAN KUOTA SKS MENGAJAR DOSEN'.
- [3] R. Ferdiana, *Rekayasa perangkat lunak yang dinamis dengan global extreme programming*. CV Andi Offset, 2012.

- [4] P. P. Sistem Informasi Manajemen Keuangan Berbasis Web Studi Kasus Karya Swadaya Abadi Happy Anita Margaretha, M. N. Nababan, and H. Anita Margaretha, 'SAINTEK (Jurnal Sains dan Teknologi) PERANCANGAN SISTEM INFORMASI MANAJEMEN KEUANGAN BERBASIS WEB STUDI KASUS PT. KARYA SWADAYA ABADI'.
- [5] P. P. H. Widodo, *Menggunakan UML*. Kota Bandung: Informatika Bandung, 2011.
- [6] P. R. A. I. K. M. G. D. Krisnayani, 'Analisa Usability Pada Website UNDIKSHA Dengan Menggunakan Metode Heuristic Evaluation Vol.5, No.2', 2016.
- [7] R. Nur, I. Sani Wijaya, I. Bustami, D. Bangsa, and J. Ji Jendral Sudirman Thehok, 'EVALUASI DAN PENGEMBANGAN WEBSITE SEKOLAH MENGGUNAKAN METODE HEURISTIK SEKOLAH MTs NEGERI 2 KOTA JAMBI', 2020.
- [8] M.; S. R. A. Shalahuddin, *Rekayasa perangkat lunak : terstruktur dan berorientasi objek*. Kota Bandung: Informatika Bandung, 2014.
- [9] C. S. K. R. L. K. S. P. Q. S. Păsăreanu, 'Symbolic Execution and Recent Applications to Worst-Case Execution, Load Testing, and Security Analysis', in *Advances in Computers*, vol. 113, Academic Press, 2019, pp. 289–314.
- [10] R. A. R. A. Azdy, 'PENERAPAN EXTREME PROGRAMMING DALAM MEMBANGUN APLIKASI PENGADUAN LAYANAN PELANGGAN (PaLaPa) PADA PERGURUAN TINGGI', *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 5, no. 2, pp. 197–206, 2018.
- [11] R. E. Putra, S. A. Wicaksono, and I. Arwani, 'Pengembangan Sistem Informasi Perpustakaan menggunakan Metode Extreme Programming (Studi pada: SMK 1 Muhammadiyah Malang)', 2019. [Online]. Available: <http://j-ptiik.ub.ac.id>