

Rancang Bangun Web Pengelolaan Hutan Rakyat Dengan Metode Personal Extreme Programming

Dipa Maulana Rifqi W.*¹, Evi Dwi Wahyuni ², Ilyas Nuryasin³

^{1,2,3}Universitas Muhammadiyah Malang

dipamaulana12@gmail.com*

Abstrak

Perangkat lunak mengalami perkembangan yang sangat pesat secara global. Berbagai metode digunakan untuk membangun proyek yang sesuai dengan kebutuhan dari klien. Tidak semua klien dapat mendeskripsikan kebutuhannya dengan jelas dan cenderung berubah-ubah mengikuti proses pengembangan perangkat lunak. hal ini tidak dapat dilakukan jika metode pengembangan perangkat lunak masih konvensional. Metode pengembangan perangkat lunak yang dapat mengikuti perubahan dari klien tersebut salah satunya adalah Extreme Programming. Bersifat fleksibel dan dapat memberikan hasil sesuai dengan yang dibutuhkan klien, Extreme Programming menjadi pilihan tepat guna pengembangan perangkat lunak yang membutuhkan waktu pengembangan relatif singkat dan hanya beranggotakan satu pengembang membuat metode ini dapat cocok digunakan dalam proyek skala kecil dengan durasi singkat. Dikombinasikan dengan metode MoSCoW untuk menentukan nilai prioritas akan lebih memudahkan pengembang untuk menentukan kebutuhan terbaik bagi klien.

Kata Kunci: Website Pengelolaan, Agile Development, Personal Extreme Programming, MoSCoW, Rehabilitasi Hasil Hutan dan Lahan

Abstract

In the last few years software has developed very rapidly around the world. Various methods are used to build the project according to the needs of client. Not all clients can clearly describe their needs and tend to change following the software development process. This development cannot be done if the software development method is still conventional. One of the software development methods that can follow changes from the client is Extreme Programming. Being flexible and able to provide results according to client requirements, Extreme Programming is the right choice for software development that requires relatively short development time and only one developer member makes this method suitable for use in small projects with short durations. Combined with the MoSCoW method to determine priority values, it will be easier for developers to determine the best needs for clients.

Keywords: Website Management, Agile Development, Personal Extreme Programming, MoSCoW, Forest and Land Product Rehabilitation

1. Pendahuluan

Teknologi media informasi berupa website dalam dunia pendidikan berfungsi untuk menunjang proses interaksi dengan user mereka agar dapat berjalan dengan baik. Peran website di Lembaga pemerintahan tidak sekedar sebagai media informasi namun juga salah satu barometer untuk mengukur kualitas lembaga tersebut. Menerapkan sistem informasi di lembaga pemerintahan dapat dijadikan solusi dalam penanganan masalah administrasi yang dialami pegawai, pengelolaan data pemerintahan, dan mengelola kegiatan pelaporan akhir data[1].

Lembaga pemerintahan Kementerian Kehutanan Provinsi Palangka Raya telah menerapkan teknologi sistem informasi berbasis web, namun hanya sebatas Sistem Informasi Hasil Hutan dimana sistem tersebut hanya menyediakan informasi tentang hasil hutan yang diproduksi oleh perusahaan saja[2]. Sistem yang dimaksud adalah Sistem Informasi Penatausahaan Hasil Hutan(SIPUHH) yang merupakan suatu sistem berbasis web yang berfungsi untuk mencatat, mempersiapkan, mengumpulkan, mengolah, menganalisis, menyimpan, dan menyebarkan informasi seputar penatausahaan hasil hutan kayu. Namun sistem ini terbatas pada pengguna yang merupakan staff dari Dinas Kehutanan, sehingga untuk masyarakat umum tidak memiliki akses menuju sistem ini[3]. Dengan menerapkan website sebagai media informasi Kehutanan

Palangka Raya berusaha memberikan layanan informasi yang lebih baik untuk menunjang keterbukaan, efisiensi serta keamanan yang diharapkan mampu menambah kualitas dari proses pendataan dan pengelolaan dan hutan rakyat.

Dari hasil observasi yang telah dilakukan, peneliti menemukan beberapa hal yang dapat menggambarkan bagaimana sistem ini kedepannya dapat lebih berguna bagi pihak Dinas Kehutanan Palangka Raya di mana proses yang berhubungan dengan pencakupan data hasil hutan rakyat masih manual. Proses perekapan yang harus ditempuh oleh staf kehutanan Palangkaraya masih menggunakan metode konvensional, dimana data kayu, pencetakan rekap, distribusi hasil rekap, serta berita acara terbaru semua harus dibagikan secara manual dalam bentuk hardcopy yang menjadikan hal tersebut tidak efisien. Proses distribusi data yang telah selesai pun masih harus dilakukan secara manual dan disimpan oleh masing-masing staff RHL (Rehabilitasi Hutan dan Lahan), dimana jika terjadi perubahan data maka tiap staff harus mengkomunikasikan dan mengirimkan data yang telah diubah melalui email ataupun flashdisk dalam bentuk softcopy.

Berdasarkan pada permasalahan diatas, penulis melakukan penelitian dengan judul "Rancang Bangun Web Pengelolaan Hutan Rakyat Dengan Metode Personal Extreme Programming". Personal Extreme Programming adalah metode pengembangan perangkat lunak yang dirancang untuk diterapkan oleh pengembang perangkat lunak secara individu[4]. Metode tersebut sangat tepat untuk diterapkan dalam pengerjaan aplikasi secara individual sesuai dengan kemampuan *programmer*. Proses pengembangan PXP bersifat berulang dan memungkinkan pengembang untuk lebih fleksibel dan responsive terhadap perubahan yang diinginkan, Penerapan PXP dapat digunakan untuk mengatasi masalah dalam pengembangan sistem berskala kecil yang membutuhkan kecepatan dalam pembuatan, mengurangi resiko dan biaya pada sistem[5].

Pada PXP terdapat metode pengumpulan kebutuhan klien yang disebut dengan *user story*. *User story* berisikan kebutuhan dari user yang nantinya akan diimplementasikan menjadi fitur didalam sistem yang dikembangkan, *user story* perlu untuk diurutkan berdasarkan kebutuhan yang paling mendasar hingga kebutuhan tambahan[6]. Untuk melakukan pengurutan *user story* penulis menggunakan metode prioritas MoSCoW. Metode prioritas MoSCoW bersifat mudah dan sederhana dalam penggunaannya, sehingga dapat mempermudah pihak pengembang dan klien untuk menentukan prioritas kebutuhan[7].

2. Metode Penelitian

2.1 Jenis Penelitian

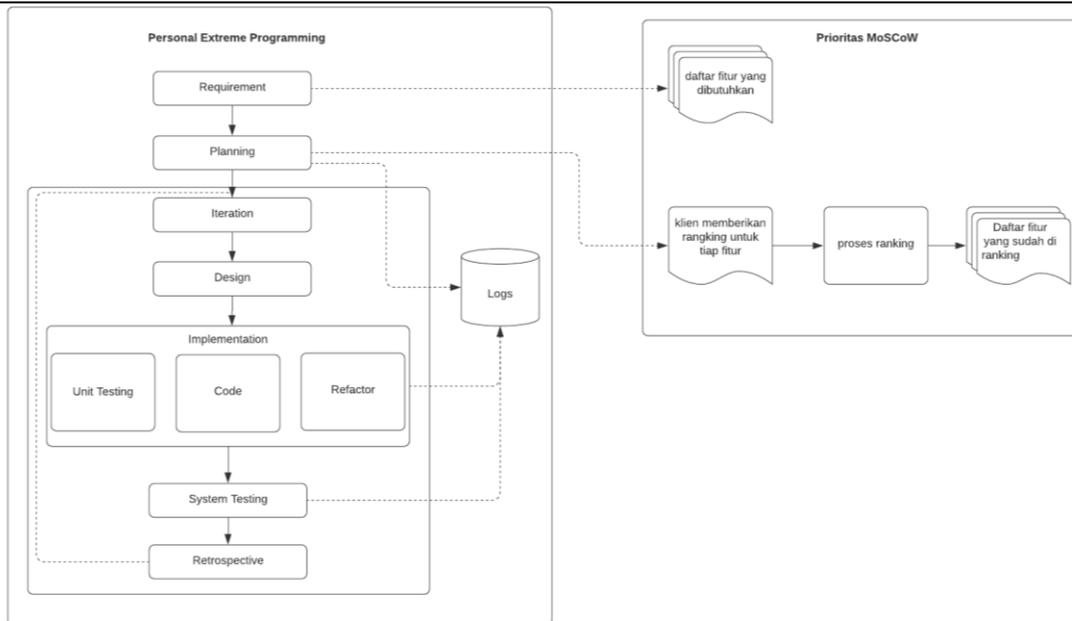
Jenis penelitian yang digunakan oleh penulis adalah penelitian kualitatif. Penelitian ini dilakukan dengan mengumpulkan data dari stakeholder untuk dapat menemukan kebutuhan yang akan diimplementasikan pada sistem yang akan dibangun.

2.2 Pengumpulan Data

Pengumpulan data yang dilakukan penulis adalah dengan menggunakan metode wawancara dan observasi untuk menemukan kebutuhan dari staff dan kepala bidang rehabilitasi hasil hutan yang akan diimplementasikan terhadap sistem yang akan dibangun. Pada pelaksanaan metode wawancara dilakukan terhadap salah satu staff dan kepala bidang dari seksi rehabilitasi hasil hutan untuk dapat mengetahui kebutuhan dan permasalahan yang ingin diselesaikan dari sistem yang sedang berjalan. Sedangkan metode observasi dilakukan dengan mengobservasi sistem yang telah berjalan pada bidang rehabilitasi hasil hutan.

2.3 Metode Pengembangan Sistem

Pada Gambar 1 menunjukan fase pengembangan aplikasi atau sistem menggunakan metode PXP dengan metode prioritas MoSCoW. Pada metode PXP dimulai dari tahap *requirement* hingga tahap *retrospective* dan pada metode MoSCoW dimulai dari menentukan daftar fitur, setelah fitur terkumpul, pada tahap planning dilakukan ranking pada setiap fitur yang dikerjakan bersama klien.



Gambar 1. Metode PXP menggunakan Prioritas MoSCoW

2.3.1 Requirement

Pada tahap *requirement*, dokumen yang akan mencakup kebutuhan fungsional dan nonfungsional dari sistem akan dibuat[8]. Pada tahap ini dilakukan wawancara dengan klien untuk mendapatkan kebutuhan apa saja yang akan diterapkan pada proyek. Hasil wawancara kemudian dimasukkan ke dalam *user story card* agar kebutuhan dapat terbagi dengan jelas[6].

2.3.2 Planning

Pada tahap *planning*, dilakukan penyusunan tugas berdasarkan hasil dari tahap *requirement* yang telah dijalankan. tahap ini bertujuan untuk menentukan tujuan untuk menentukan urutan prioritas *user story*, waktu yang dibutuhkan untuk implementasi *user story*, dan jumlah iterasi yang dibutuhkan hingga pengembangan sistem[8].

2.3.3 Iteration Initialization

Pada tahap *iteration initialization*, menandakan awal dimulainya setiap iterasi dalam sebuah proses *iteration development*. iterasi dimulai dengan memilih kebutuhan yang kedepannya akan menjadi fokus dari iterasi yang dijalankan[8]. Panjang dari iterasi bervariasi mulai dari 1 hingga 3 minggu tergantung pada proyek yang dikerjakan[9]. Penentuan banyaknya jumlah iterasi tergantung dari banyak kebutuhan yang didapat pada tahap *requirement*.

2.3.4 Design

Pada tahap *design*, dilakukan pemodelan suatu modul sistem dan kelas yang akan dilaksanakan dalam iterasi. Tahap ini bertujuan untuk merancang agar sistem memenuhi persyaratan klien tanpa mencoba menebak apa yang akan diperlukan pada masa mendatang. Desain yang dibuat adalah desain database yang akan digunakan pada kebutuhan atau proses bisnis yang dijalankan.

2.3.5 Implementation

Pada tahap *implementation*, penulis mengimplementasikan keseluruhan objek yang ada pada tahap *design* dan menguji objek tersebut. Ada tiga sub tahap yang terdapat pada *implementation* yaitu *unit testing*, *code*, dan *refactor*[8]. *Unit testing* merupakan pengujian kode program, *code* adalah proses pelengkapan kode yang akan digunakan agar sesuai setelah melalui tahap *unit testing*, dan *refactor* adalah proses yang dilakukan ketika kode program harus di optimasi[10].

2.3.6 System Testing

Pada tahap *system testing*, hasil dari tahap implementasi selanjutnya akan diuji. Proses pengujian akan dijalankan langsung oleh pihak klien dengan didampingi oleh pengembang. Hal tersebut guna memastikan sistem yang sudah dibuat sesuai dan memenuhi kebutuhan klien pada tahap *requirement* dan *planning*. Pengujian yang dilakukan menggunakan *User Acceptance Test*(UAT) [8].

2.3.7 Retrospective

Pada tahap *retrospective*, menandai akhir dari sebuah iterasi, Analisa data dikumpulkan selama tahap lain sedang dijalankan[8]. Pengembang lalu memverifikasi apakah perkiraan waktu pengerjaan sama dengan waktu sebenarnya, dan menjabarkan alasan jika ada potensi keterlambatan pengerjaan pada proyek masa depan.

3. Hasil Penelitian dan Pembahasan

3.1 Requirement

Pada tahap ini dilakukan proses pengumpulan data kebutuhan di bidang RHL dinas kesehatan Palangka Raya. Proses pengumpulan data dilakukan dengan cara diskusi bersama kepala seksi dan staff RHL. Selama proses diskusi, klien diminta untuk menyebutkan dan menjabarkan proses bisnis serta permasalahan yang terdapat pada bidang RHL. Permasalahan yang ditemukan adalah penyaluran data lahan masih bersifat manual, data hanya diketahui oleh staff yang bekerja di bidang tersebut, masyarakat tidak mengetahui area yang akan dan sudah dikelola, data rawan hilang. Berdasarkan masalah tersebut, pengembang melakukan pengarahan terhadap klien agar segala kebutuhan dari klien dapat diimplementasikan ke dalam fitur yang akan dibangun pada sistem informasi. Penjabaran tersebut menggunakan *user story*, terdapat beberapa kriteria penulisan dari *user story* yaitu, *negotiable*, *estimatable*, dan *testable*[6].

3.2 Planning

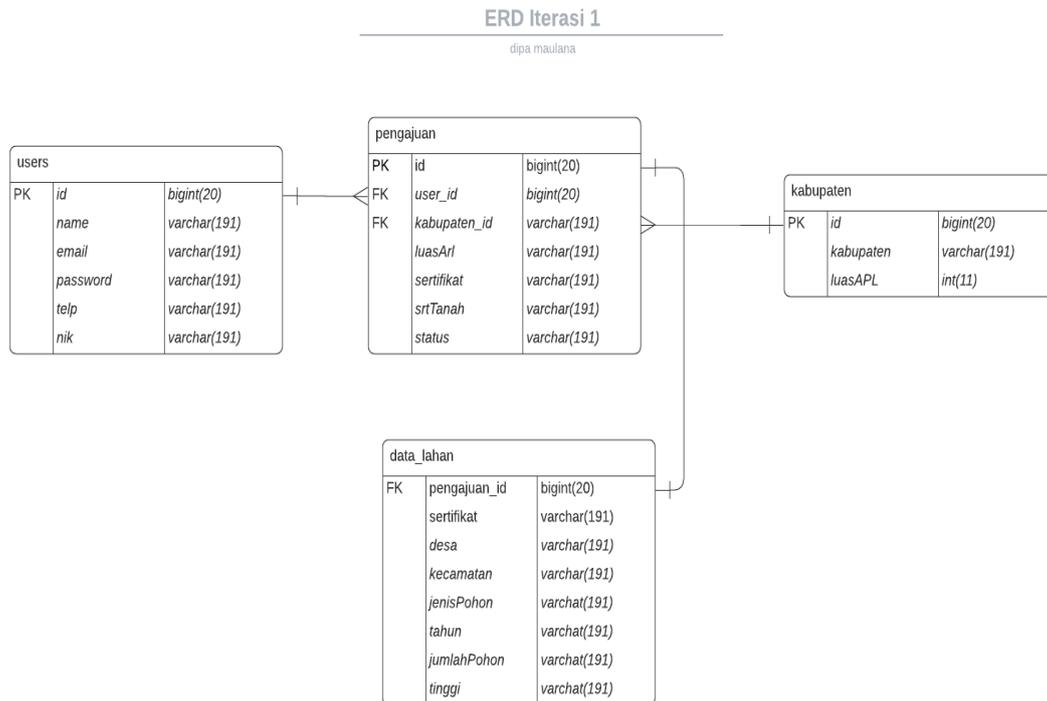
Pada tahap ini dilakukan *planning* pada masing-masing *user story* dengan cara diskusi bersama klien. Setelah proses diskusi maka terbentuklah beberapa tabel *user story* yang kemudian oleh pengembang dapat dilakukan proses estimasi nilai *story point* dan menentukan nilai *velocity*[9]. Selanjutnya pengembang menentukan prioritas dari tiap *user story* yang telah didapatkan kemudian dikelompokkan kedalam 4 kategori yaitu *Must have*, *Should have*, *Could have*, dan *Won't Have*[7]. Dari pembagian tersebut maka akan didapatkan hasil kebutuhan mana yang merupakan kebutuhan utama, dan kebutuhan mana yang bukan merupakan kebutuhan utama.

3.3 Iteration Initialization

Tahap ini menandakan permulaan dari iteration development. Pada tahap ini dilakukan pemilihan user story yang akan diimplementasikan selama proses iterasi. User story dipilih berdasarkan prioritas dengan urutan *Must Have*, *Should Have*, *Could Have*, dan terakhir *Won't have*.

3.4 Design

Pada Tahap design pengembang membuat design untuk semua user story yang telah dipilih pada tahap *iteration initialization*. Design dibuat dibuat sederhana untuk user story sesuai dengan iterasi yang sedang berlangsung sedangkan skema database yang akan ditampilkan merupakan *Entity Realtionship Diagram* (ERD) yang selanjutnya akan diimplementasikan ke tahap selanjutnya. Desain database ditunjukkan pada Gambar 2.



Gambar 2. Desain Entity Relationship Diagram

3.5 Implementation

Pada tahap ini dimulai dari pengujian *unit testing* dari setiap *user story* dengan menggunakan *library PHPUnit*. Hasil dari *unit testing* ditunjukkan pada Gambar 3.

```
PS D:\xampp\htdocs\blog> vendor/bin/phpunit --filter add_user_database
PHPUnit 9.5.13 by Sebastian Bergmann and contributors.

-                                                                    1 / 1 (100%)

Time: 00:03.325, Memory: 22.00 MB

OK (1 test, 1 assertion)
PS D:\xampp\htdocs\blog>
```

Gambar 3. Unit Testing

Setelah *unit testing* dijalankan maka proses selanjutnya adalah tahap *code*. Proses ini berguna untuk melengkapi kode program yang sudah melewati tahap *unit testing*. Selanjutnya tahap terakhir adalah *refactor*, dimana di tahap ini dilakukan proses optimasi kode jika diperlukan.

3.6 System Testing

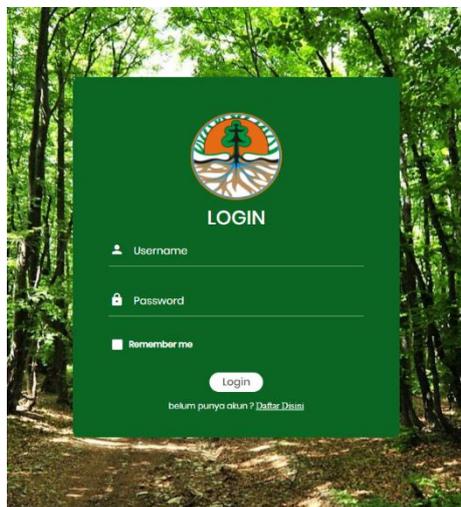
Pada tahap ini dilakukan *system testing* pada tiap iterasi menggunakan *User Acceptance Test* sesuai dengan iterasi yang dilakukan pada tahap *implementation*. Hasil dari *User Acceptance Test* ditunjukkan pada Tabel 1.

Tabel 1. User Acceptance Test

Iterasi ke-1			Hasil	
User Story	Prosedur Pengujian	MoSCoW	Sesuai	Tidak Sesuai
User Login	<ul style="list-style-type: none"> Pada halaman login dibagian bawah, tekan tulisan daftar disini Isikan data diri sesuai dengan form yang diminta 	Must Have	Ya	

	<ul style="list-style-type: none"> • Tekan tombol login pada navbar • Masukan <i>email</i> dan Password yang telah didaftarkan • Berhasil login 		Ya
	<ul style="list-style-type: none"> • Tekan tombol login pada navbar • Masukan <i>email</i> dan Password yang belum didaftarkan • Login gagal 		Ya
<i>User</i> Mengajukan Pengelolaan Lahan	<ul style="list-style-type: none"> • Tekan navbar bertuliskan Pengajuan Lahan • Masukan Nama, Nomor telfon, NIK, serta unggah file sertifikat dan surat tanah 	Must Have	Ya
<i>User</i> melihat Status Pengelolaan Lahan	<ul style="list-style-type: none"> • Tekan tombol ajukan • Tekan navbar bertuliskan Data Lahan • Tekan tombol bertuliskan lihat lahan saya 	Must Have	Ya

Dari hasil *User Acceptance Test* pada tabel diatas menunjukkan bahwa *user story* telah memenuhi kebutuhan dari *user*. Pada setiap *user story* mempunyai tampilan yang ditunjukkan pada gambar berikut.



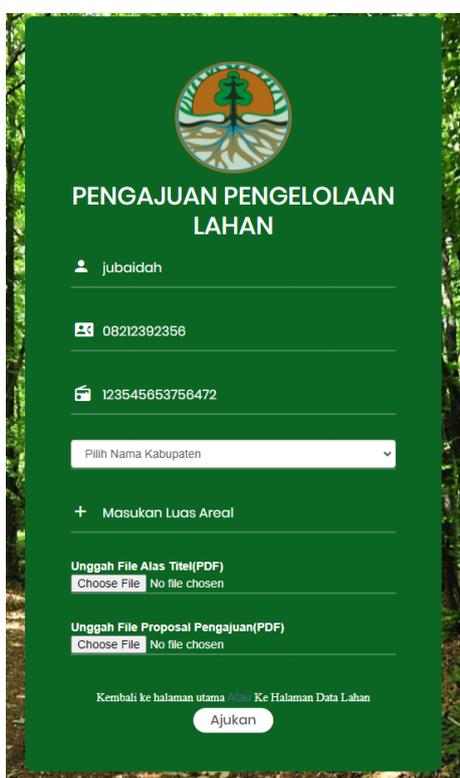
Gambar 4. Tampilan user login

Gambar 4. Merupakan gambar halaman yang akan ditampilkan ketika *user* menekan tombol login yang terdapat pada navbar, *user* memasukan email dan password yang sudah didaftarkan, jika belum mendaftarkan maka *user* menekan tulisan Daftar Disini untuk menuju ke halaman pendaftaran



Gambar 5. Tampilan daftar user

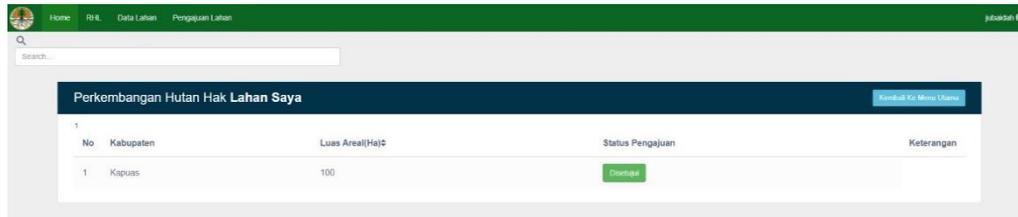
Gambar 5. Merupakan gambar halaman yang akan ditampilkan jika *user* menekan tulisan Daftar Disini yang terdapat pada halaman Login, pada halaman daftar *user* harus mengisi kolom yang tersedia sesuai dengan identitas *user*, setelah selesai melakukan pengisian *user* bisa menekan tombol Masuk untuk membuat akun dan masuk ke halaman utama.



Gambar 6. Tampilan form pengajuan lahan

Gambar 6. Merupakan gambar halaman pengajuan pengelolaan lahan jika *user* sudah melakukan login, jika belum melakukan login maka tombol pengajuan lahan tidak dapat ditekan.

User yang sudah login dapat melakukan pengajuan lahan dengan cara menekan tombol pengajuan lahan yang terdapat pada halaman utama.



Gambar 7. Tampilan status pengajuan lahan

Gambar 7. Merupakan halaman untuk pengaju lahan melihat status pengajuan lahannya, jika sudah disetujui maka status akan berubah menjadi disetujui jika menunggu proses maka status menjadi menunggu proses, dan jika ditolak maka status akan menjadi ditolak disertai dengan alasan penolakan.

3.7 Retrospective

Pada tahap ini, pengembang menunjukkan hasil dari tiap iterasi ke klien, verifikasi waktu pengerjaan lalu dilakukan terhadap waktu estimasi yang sudah ditetapkan pada tahap *planning*. Verifikasi tersebut ditampilkan pada Tabel 2.

Tabel 2 Verifikasi Waktu Pengerjaan

Iterasi 1				
User Story	MoSCoW	Story Point	Waktu Estimasi	Waktu Realisasi
User Login User	Must Have	2	4 Hari	4 Hari
Mengajukan Pengelolaan Lahan	Must Have	1	2 Hari	2 Hari
User Melihat Status Pengelolaan	Must Have	1	2 Hari	2 Hari
Velocity		4	8 Hari	8 Hari

4. Kesimpulan

Berdasarkan hasil pembahasan, pengujian, dan pengerjaan dari website pengelolaan hasil hutan yang sudah dibangun dapat disimpulkan bahwa penggunaan Metode PXP dapat berjalan lancar dalam perancangan sebuah sistem dengan lingkup yang tidak terlalu besar dan hanya beranggotakan satu orang. Menitik beratkan focus pada komunikasi antara klien dengan pengembang sehingga mempermudah pengembang dalam menangani setiap kebutuhan yang diajukan oleh klien. Proses pengumpulan kebutuhan pada tahap *requirement* membuat pengembangan pengembangan sistem berjalan dengan baik dikarenakan semua kebutuhan klien sudah didapatkan diawal. Setelah pengerjaan selesai, terdapat tahap *retrospective* yang juga sangat membantu dalam pengecekan fitur apakah sudah sesuai dengan kebutuhan awal atau belum. Penggunaan UAT pada penelitian ini memudahkan proses validasi sistem yang telah dikerjakan apakah telah memenuhi kebutuhan dari klien atau belum. pada tiap iterasi, UAT digunakan untuk memastikan apakah fitur yang dibutuhkan pada iterasi tersebut sudah memiliki hasil yang sesuai dengan apa yang dibutuhkan user.

Referensi

- [1] Dani Eko Hendrianto, "Pembuatan Sistem Informasi Perpustakaan Berbasis Website Pada Sekolah Menengah Pertama Negeri 1 Donorojo Kabupaten Pacitan," *Indones. J. Netw. Secur.*, vol. 3, no. 4, pp. 57–64, 2014.
- [2] M. A. Setiawan, Wahyudi, and Alpian, "Penerapan sistem informasi penatausahaan hasil hutan (SI - PUHH) online terhadap peredaran kayu Ilegal di Kalimantan Tengah," *J. Environment Manag.*, vol. 2, no. 2, pp. 106–113, 2021.

- [3] M. Simanjuntak, A. Daryanto, and T. Rusolono, "Forestry Company Satisfaction Towards Online Information System Implementation of Forest Product Management (Si-Puhh)," *Indones. J. Bus. Entrep.*, vol. 2, no. 1, pp. 12–22, 2016, doi: 10.17358/ijbe.2.1.12.
- [4] R. Fojtik, "Extreme programming in development of specific software," *Procedia Comput. Sci.*, vol. 3, pp. 1464–1468, 2011, doi: 10.1016/j.procs.2011.01.032.
- [5] S. A. Asri, I. G. M. A. Sunaya, P. M. Prihatini, and W. Setiawan, "Comparing Traditional and Agile Software Development Approaches: Case of Personal Extreme Programming," vol. 1, no. 1cst, pp. 561–563, 2018, doi: 10.2991/icst-18.2018.116.
- [6] V. Schetinger, C. Souza, L. M. Fontoura, and C. T. Pozzer, "User stories as actives for game development," *SBGames 2011*, pp. 1–4, 2011.
- [7] K. S. Ahmad, N. Ahmad, H. Tahir, and S. Khan, "Fuzzy-MoSCoW: A fuzzy based MoSCoW method for the prioritization of software requirements," *2017 Int. Conf. Intell. Comput. Instrum. Control Technol. ICICICT 2017*, vol. 2018-Janua, pp. 433–437, 2018, doi: 10.1109/ICICICT1.2017.8342602.
- [8] S. A. Asri, I. G. A. M. Sunaya, E. Rudiastari, and W. Setiawan, "Web Based Information System for Job Training Activities Using Personal Extreme Programming (PXP)," *J. Phys. Conf. Ser.*, vol. 953, no. 1, 2018, doi: 10.1088/1742-6596/953/1/012092.
- [9] W. Sadewo, "Perancangan Dan Implementasi Android-Mobile System Marketing Surveyor (a-Ms2) Menggunakan Metode Personal Extreme Programming (Pxp)," vol. 7, pp. 1–5, 2016.
- [10] E. Daka and G. Fraser, "A survey on unit testing practices and problems," *Proc. - Int. Symp. Softw. Reliab. Eng. ISSRE*, pp. 201–211, 2014, doi: 10.1109/ISSRE.2014.11.

