Analisa *Usability* dan *Performance* pada Sistem Informasi Laboratorium Tata Hidang "SILABTATAHIDANG" Universitas Negeri Surabaya dengan Menggunakan Metode *Personal Extreme Programming* dan Pengujian *Black-box testing*Analisa

Muhammad Alif Hidayatullah[1], Dodik Arwin Dermawan[2]

Manajemen Informatika, Universitas Negeri Surabaya
Jl. Ketintang, Ketintang, Kec. Gayungan, Kota Surabaya, Jawa Timur 60231
[1]muhammad.19065@mhs.unesa.ac.id
[2]dodikdermawan@unesa.ac.id

Abstrak*— Penelitian ini bertujuan mengembangkan sistem informasi pengelolaan laboratorium kuliner yang efektif dan efisien untuk mendukung kegiatan akademik jurusan D4 Tata Boga dan S1 Pendidikan Tata Boga Universitas Negeri Surabaya. Fokus penelitian ini adalah meningkatkan aksesibilitas data dan pelayanan bagi mahasiswa dalam peminjaman alat dan bahan di laboratorium kuliner, serta meningkatkan produktivitas dalam pengembangan dan pemeliharaan sistem. Metode yang digunakan adalah Agile dengan pendekatan Personal Extreme Programming (PXP), dan pengujian menggunakan Black-Box testing. Hasil penelitian menunjukkan bahwa sistem informasi yang dikembangkan berhasil meningkatkan aktivitas peminjaman di laboratorium kuliner, terbukti dengan hasil uji usability dan performance yang signifikan dibandingkan sistem sebelumnya. Temuan ini menegaskan bahwa implementasi sistem informasi pengelolaan laboratorium kuliner baru memberikan manfaat nyata dalam efisiensi dan efektivitas pengelolaan laboratorium, serta meningkatkan pengalaman pengguna bagi mahasiswa.

***Kata Kunci*— Sistem Informasi, Laboratorium Kuliner, Aksesibilitas Data, Personal Extreme Programming (PXP), Black-Box testing.**

Abstract*— This research aims to develop an effective and efficient culinary laboratory management information system to support the academic activities of the D4 Culinary Arts and S1 Culinary Education departments at Surabaya State University. The focus of this research is to enhance data accessibility and services for students in borrowing equipment and materials from the culinary laboratory, as well as to improve productivity in system development and maintenance. The method utilized is Agile with the Personal Extreme Programming (PXP) approach, and testing is conducted using Black-Box testing. The research results indicate that the developed information system successfully enhances borrowing activities in the culinary laboratory, as evidenced by significant usability and performance test results compared to the previous system. These findings affirm that the implementation of the new culinary laboratory management information system provides tangible benefits in the efficiency and effectiveness of laboratory management, along with enhancing the user experience for students.

***Keywords*— Information System, Culinary Laboratory, Data Accessibility, Personal Extreme Programming (PXP), Black-Box Testing.**

I. PENDAHULUAN

Pada penelitian ini, peneliti akan mengembangkan Sistem Informasi Pengelolaan Laboratorium dalam bentuk *Website*. Menurut Sibero [1] *website* adalah suatu sistem yang berkaitan dengan dokumen yang berguna sebagai media untuk menampilkan gambar, teks, multimedia dan lainnya pada jaringan internet. Jumlah situs web pada tahun 2019, diperkirakan lebih dari 1,5 miliar *website* yang beredar di internet. Hal Ini juga sejalan dengan apa yang di katakan oleh Zhongguo [2] sejak tahun 2000-an, jumlah pengguna *internet*, *bandwidth*, dan jumlah situs web telah meningkat secara dramatis. Berdasarkan hal ini, peneliti berasusmsi bahwa pengguna situs web masih sangat aktif dan cenderung meningkat sampai sekarang. Sehingga dalam penelitian ini, peneliti bermaksud untuk mengembangkan sebuah sistem informasi berbasis web dengan nama "SIPELAKU" dengan menggunakan *Framework Laravel*.

Seiring dengan berjalannya waktu, pihak dari jurusan D4 Tata Boga dan S1 Pendidikan Tata Boga Universitas Negeri Surabaya menyadari bahwa *website* sistem informasi tersebut sudah layak untuk diperbarui agar lebih mudah digunakan oleh para pengguna. Berdasarkan pengamatan yang dilakukan oleh peneliti terhadap *website* sistem informasi "SILABTATAHIDANG", peneliti melihat yaitu masih menggunakan *PHP Native*, sebenarnya tidak ada salahnya menggunakan *PHP Native*, pada segi

performance kecepatan *website* seharusnya lebih unggul dari penggunaan *framework*. Dikarenakan konten dan ukuran projek pada *framework* jauh lebih banyak dan besar dibandingkan dengan *native*. Tetapi Ketika dilakukan *load* pada *website* SILABTATHIDANG masih terbilang cukup lambat. Hal ini dikarenakan terdapat *banner* gambar pada *home page* yang ukurannya cukup besar. Hal lain yang ditemukan oleh peneliti adalah Ketika pengguna ingin meminjam alat dan bahan, pengguna di haruskan melakukan *login* terlebih dahulu, peneliti juga tidak menemukan adanya halaman *forgot password*, hal ini cukup penting ketika membuat autentikasi pada sistem, karena mungkin saja pengguna memerlukan fitur tersebut jika pada saat itu password yang dimasukkan salah ataupun lupa.

Penelitian ini menggunakan model pendekatan *Agile Personal Extreme Programming* (PXP), Hal ini juga memastikan bahwa produk yang dikembangkan memenuhi kebutuhan dan standar kualitas yang ditentukan oleh klien. Peneliti juga melakukan komparasi dengan pendeketakan *agile* lainnya, hasilnya peneliti menemukan penelitian yang membandingkan antara *Extreme Programming* (XP) dan *Scrum* pada penelitian berikut [3]. Metode *Personal Extreme Programming* (PXP) memiliki beberapa keunggulan dibandingkan dengan metode *Scrum*. Pertama, XP memfokuskan pada kolaborasi dan komunikasi antar individu secara intensif dan terus menerus, sehingga memastikan bahwa produk yang dikembangkan sesuai dengan harapan klien. Hal ini membuat proses pengembangan produk lebih efisien dan efektif. Kedua, PXP memberikan fleksibilitas dan kebebasan bagi setiap individu dalam tim untuk mengejar solusi terbaik bagi setiap masalah yang muncul. Hal ini membuat tim lebih kreatif dan inovatif dalam menyelesaikan tugas mereka. Berbeda dari *Scrum* yang lebih berkonsentrasi pada proses dan memiliki jadwal yang kaku dan tidak fleksibel. Ketiga, PXP memfokuskan pada pengembangan produk yang berkesinambungan dan berkelanjutan, sehingga memastikan bahwa produk yang dikembangkan sesuai dengan harapan klien dan dapat beradaptasi dengan perubahan yang terjadi. Untuk pengujian peneliti menggunakan pengujian *black-box testing*.

Dengan menggabungkan pendekatan PXP dan pengujian *black-box testing*, peneliti berharap dapat mengatasi masalah performa dan kualitas pengalaman pengguna yang ditemukan pada sistem informasi sebelumnya. Pendekatan PXP memungkinkan kolaborasi yang intensif antara peneliti, memastikan bahwa setiap langkah pengembangan sistem berjalan sesuai dengan kebutuhan dan harapan pengguna. Dalam pengujian *black-box testing*, sistem akan diuji secara menyeluruh dari perspektif pengguna, memastikan bahwa fungsi-fungsi utama bekerja dengan baik dan memberikan pengalaman yang optimal. Dengan pendekatan dan pengujian ini, diharapkan sistem informasi SIPELAKU dapat memberikan perbaikan yang signifikan dan memenuhi harapan pengguna dalam pengelolaan laboratorium kuliner.

II. TINJAUAN PUSTAKA

A. Penelitian Terdahulu

TABEL I
Penelitian Terdahulu

No.	Peneliti	Review	
1.	Ardianzah	Judul	Pengembangan Sistem Pengelolaan Peminjaman Auditorium Universitas Muhammadiyah Malang Berbasis Web Menggunakan Metode *Personal Extreme Programming* [4].
		Hasil	Sistem pengelolaan peminjaman auditorium dikembangkan menggunakan metode *Personal Extreme Programming* (PXP) yang memenuhi kebutuhan klien.
		Perbedaan	Penelitian ini berfokus pada pengelolaan peminjaman auditorium dan menggunakan PXP sebagai metode pengembangan.
2.	Firmansyah	Judul	Rancang Bangun Sistem Informasi Penjualan Pada Toko Buku Kita Tasikmalaya Berbasis Web Menggunakan *Framework Laravel* 8 [5].
		Hasil	Sistem informasi penjualan untuk Toko Buku Kita dikembangkan menggunakan metode *Personal Extreme Programming* (PXP) dan *framework Laravel* 8.

		Perbedaan	Penelitian ini berfokus pada pengembangan sistem informasi penjualan untuk toko buku dengan menggunakan PXP dan *framework Laravel* 8.
3.	Ulfi	Judul	Implementasi Metode *Personal Extreme Programming* dalam Pengembangan Sistem Manajemen Transaksi Perusahaan (Studi Kasus: CV. Todjoe Sinar Group) [6].
		Hasil	Metode *Personal Extreme Programming* (PXP) diimplementasikan dalam pengembangan sistem manajemen transaksi perusahaan dan terbukti efektif.
		Perbedaan	Penelitian ini berfokus pada pengembangan sistem manajemen transaksi perusahaan menggunakan PXP.
4.	Asri	Judul	*Comparing Traditional and Agile Software Development Approaches: Case of Personal Extreme Programming* [7].
		Hasil	Hasil dari penelitian tersebut adalah pendekatan *agile* (PXP) memberikan kemudahan dan fleksibilitas untuk mengembangkan aplikasi.
		Perbedaan	Penelitian ini membandingkan antara pendekatan pengembangan perangkat lunak tradisional dengan pendekatan *Agile*, khususnya *Personal Extreme Programming* (PXP).

B. Dasar Teori

1) Personal Extreme Programming

Personal Extreme Programming (Personal XP) adalah metodologi pengembangan perangkat lunak yang diadaptasi dari *Extreme Programming* (XP) yang awalnya dikembangkan untuk digunakan oleh tim pengembang perangkat lunak. *Personal Extreme Programming* dirancang untuk digunakan oleh pengembang perangkat lunak individu yang bekerja sendiri atau dalam tim kecil. *Personal Extreme Programming* mempertahankan prinsip-prinsip *Extreme Programming* (XP), seperti pengembangan berbasis fitur, pengujian otomatis, dan integrasi berkelanjutan. Namun, *Personal Extreme Programming* menyesuaikan praktik-praktik ini untuk bekerja dalam lingkungan pengembangan perangkat lunak yang kurang terstruktur dan lebih mandiri.

Pada tesis berikut [8] Dalam *Personal Extreme Programming*, semua praktik inti dalam *Extreme Programming* (XP) yang secara mudah dapat diterjemahkan ke situasi satu orang dibiarkan tidak berubah. Praktik-praktik berikut tidak langsung dapat diaplikasikan oleh seorang individu dan oleh karena itu berbeda dari *Extreme Programming* (XP), antara lain:

a. *The Planning Game*
Jika pengembang (dalam hal ini peneliti) bekerja untuk diri sendiri atau memiliki cukup informasi untuk mewakili klien/ pihak terkait, maka pengembang hanya perlu bergantian dalam peran selama fase perencanaan dikarenakan faktor *user stories*.

b. *Pair Programming*
Pair programming adalah salah satu praktik dalam *Extreme Programming* (XP) yang melibatkan dua programmer yang bekerja sama dalam menulis kode, mereview, dan menguji satu sama lain. Manfaat dari *pair programming* jelas hilang ketika bekerja sendiri. Sebagai gantinya, seorang programmer yang bekerja sendiri dapat menerapkan teknik debugging yang disebut *rubber duck debugging*, meminta bantuan teman atau keluarga untuk pengingat, dan menerapkan informal *walkthrough* dan

filosofi uji terlebih dahulu sebagai pengganti praktik *pair programming*.

c. *On-site Customer*

On-site customer adalah salah satu praktik dalam *Extreme Programming* (XP) di mana seorang pengguna bekerja sama secara langsung dengan tim pengembang. Namun, dalam *Personal Extreme Programming* (PXP), praktik ini tidak selalu dapat diterapkan karena proyek yang dikerjakan oleh seorang pengembang sendiri mungkin tidak cukup besar untuk membenarkan kehadiran pengguna secara langsung. Sebagai alternatif, komunikasi dengan pengguna dapat dilakukan melalui email atau telepon, dan demo masih harus dipersiapkan dan diberikan pada akhir setiap iterasi, bahkan jika pengembang bekerja sendiri.

2) *Black-box testing*

Salah satu aktivitas pengujian yang dapat dilakukan dalam *Personal Extreme Programming* adalah pengujian perangkat lunak, dan *black-box testing* merupakan salah satu jenis pengujian yang dilakukan. Pengujian *Black-box testing* adalah gaya pengujian di mana penguji tidak memiliki pengetahuan tentang struktur dan kode dalam suatu aplikasi, dan karenanya harus membangun pemahaman mereka sendiri tentang aplikasi tersebut melalui analisis dan penyelidikan yang cermat [9]. Dengan melakukan *black-box testing* secara teratur pada *Personal Extreme Programming*, akan membantu dalam menjamin bahwa sistem yang dikembangkan dapat berjalan dengan baik dan memenuhi kebutuhan pengguna.

3) Sistem Informasi

Sistem informasi (SI) ialah kumpulan unsur yang berhubungan satu sama lain yang menghimpun, mengolah, menyimpan, serta mengkomunikasikan data dan informasi; SI menyajikan cara kembali informasi untuk mengawasi dan mengelola operasinya agar tetap sesuai tujuan dan targetnya. Mekanisme kembali informasi menjadi komponen signifikan dalam mendukung organisasi mencapai misi mereka, seperti meningkatkan keuntungan atau menaikkan pelayanan kepada pelanggan. [10].

4) PHP

PHP adalah bahasa skrip sisi server (*server-side*) yang memungkinkan pengguna memasukkan instruksi ke halaman web yang akan dijalankan oleh perangkat lunak server web sebelum mengirimkan halaman tersebut ke browser yang memintanya. Bahasa sisi server mirip dengan JavaScript karena memungkinkan pengguna untuk menyematkan konten yang dihasilkan secara dinamis ke dalam kode HTML pada halaman web, memberi pengguna kontrol lebih besar atas apa yang muncul di jendela browser daripada yang dapat disediakan oleh HTML saja. Perbedaan utama antara JavaScript dan PHP adalah tahap di mana kode dijalankan [11].

Setelah server web mengeksekusi kode PHP yang disematkan di halaman web, hasilnya menggantikan kode PHP di halaman tersebut. Yang dilihat browser hanyalah kode HTML standar saat menerima halaman. Itu sebabnya PHP disebut "*bahasa sisi server*": pekerjaannya dilakukan di server. Kode PHP ditulis dalam tag PHP. Seperti kebanyakan tag HTML, PHP memiliki tag awal dan tag akhir: <?php dan ?>. Apa pun di dalam tag PHP ini diperlakukan sebagai kode PHP dan berjalan di server.

5) *Laravel*

Laravel merupakan platform pengembangan PHP yang sumbernya terbuka (*open-source*), dengan struktur yang terorganisir dan mudah dipelajari. Dengan mengadopsi prinsip desain MVC (model-view-controller). Laravel memanfaatkan komponen-komponen dari berbagai kerangka kerja yang berbeda untuk membantu dalam konstruksi aplikasi *website*. Hasilnya adalah aplikasi web yang lebih teratur dan realistis. Laravel menyajikan serangkaian fitur yang menggabungkan elemen-elemen dasar dari kerangka kerja PHP seperti CodeIgniter, Yii, dan bahkan bahasa pemrograman lainnya seperti Ruby on Rails [12].

6) *MySQL*

MySQL merupakan sistem manajemen database relasional yang gratis dan *open-source*. *MySQL* berfungsi baik di web maupun di *server*. *MySQL* adalah sistem manajemen basis data server web yang dapat diskalakan, dan dapat diperluas dengan situs web. *MySQL* sejauh ini merupakan sistem manajemen basis data *SQL* sumber terbuka yang paling banyak digunakan, dikembangkan oleh Perusahaan *Oracle*. Tabel digunakan untuk menyimpan informasi dalam database *MySQL*. Tabel adalah sekumpulan kolom dan baris yang menyimpan informasi terkait. *MySQL* menyertakan *standalone clients* yang memungkinkan pengguna berkomunikasi langsung dengan database *MySQL* menggunakan *SQL*. Namun, *MySQL* lebih umum digunakan bersama dengan program lain untuk membuat aplikasi yang membutuhkan fungsionalitas basis data relasional [13].

7) XAMPP

XAMPP adalah lingkungan pengembangan PHP paling populer. XAMPP adalah distribusi Apache gratis dan mudah dipasang yang berisi *MySQL*, PHP, dan Perl. Jika pengguna belum pernah menginstal XAMPP pada perangkat, buka

https://www.apachefriends.org/download.html . Unduh dan instal. Perlu di perhatikan, Kita perlu mendownload dan menginstal versi XAMPP yang mendukung PHP 8 karena Laravel veris terbaru yaitu pada versi *Laravel* 9 membutuhkan PHP 8. Umumnya XAMPP pada Laravel digunakan untuk membuat server *local* dan *database MySQL* untuk menyimpan data yang diinputkan oleh pengguna nantinya [14].

8) AJAX

Asynchronous JavaScript and XML (AJAX) digunakan di sisi klien (*browser*) untuk mengirim dan menerima data dari server dengan memanfaatkan objek *XMLHttpRequest* dalam Bahasa pemrograman *JavaScript*. Terlepas dari namanya, penggunaan XML tidak diperlukan, dan JSON sering digunakan sebagai gantinya. Itu sebabnya pengembang/ *developer* hampir tidak pernah mengatakan AJAX lagi. Perlu diingat bahwa permintaan HTTP dapat dibuat secara sinkron, tetapi bukan praktik yang baik untuk melakukannya. Contoh permintaan sinkronisasi yang paling umum adalah penyertaan tag <script> [15].

9) Visual Studi Code

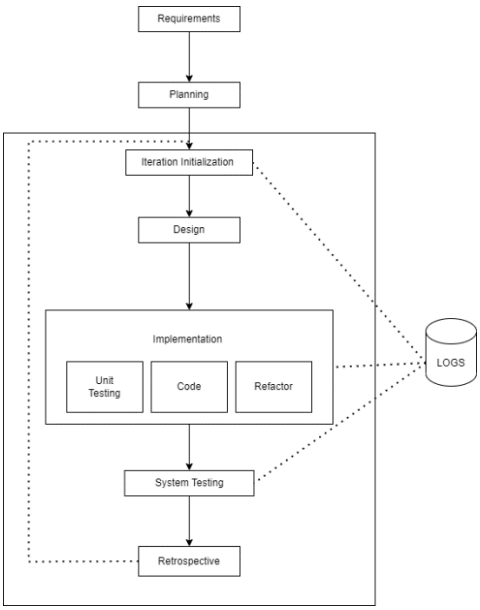
Visual Studio Code adalah editor kode yang dikembangkan oleh Microsoft untuk Windows, Linux, dan macOS. Ini adalah editor kode lintas platform yang ringan yang memberi pengembang/ *developer* berbagai alat untuk menulis dan men-debug kode. Salah satu fitur utama dari Visual Studio Code adalah dukungannya untuk berbagai bahasa pemrograman dan lingkungan pengembangan. Termasuk juga dukungan bawaan untuk debugging dan *IntelliSense*, yang menyediakan *intelligent code completion* dan fitur navigasi kode. Ini juga termasuk terminal bawaan dan dukungan untuk sistem kontrol versi seperti Git [16].

Visual Studio Code dirancang agar dapat disesuaikan, sehingga pengembang/ *developer* dapat memilih dari berbagai ekstensi dan tema untuk menyesuaikan editor dengan kebutuhan khusus dari masing-masing individu. Visual Studio Code juga open-source dan tersedia secara gratis, menjadikannya pilihan populer bagi pengembang/ *developer* di berbagai platform.

III. METODE PENELITIAN

A. Rancangan Penelitian

Penelitian ini dimulai dengan mengamati situs *website* SILABTATAHIDANG dan menganalisis Standar Operasional Prosedur (SOP) yang diterapkan dalam laboratorium kuliner, dengan tujuan untuk memahami kebutuhan yang harus diakomodasi oleh sistem informasi yang akan dibuat. Setelah tahap ini dilakukan, selanjutnya dilakukan telaah literatur guna mendukung perjalanan penelitian yang direncanakan. Setelahnya, dimulailah proses perancangan sistem dengan menerapkan pendekatan *Personal Extreme Programming* (PXP). Setelah sistem peminjaman berhasil dikembangkan, langkah berikutnya melibatkan penerapan dan pengujian sistem pada unit terbaru sistem informasi, yaitu SIPELAKU. Pengujian mencakup evaluasi kinerja dan kegunaan sistem, dan hasil dari tahap ini akan dijelaskan secara komprehensif dalam bentuk laporan.



Gambar.1 Skema Metode *Personal Extreme Programming*

1) *Requirements*

Tahap ini menggambarkan proses mengenali keperluan pengguna terkait sistem yang akan dikembangkan. Langkah berikutnya melibatkan peneliti dalam menghimpun kebutuhan yang diperlukan oleh sistem informasi yang bakal dibangun. Sesuai dengan penelitian sebelumnya [17], langkah pengumpulan keperluan atau *requirement* ini dijalankan dengan melakukan interaksi dan dialog bersama seorang dosen dari tata boga, juga dengan anggota laboran yang bertanggung jawab atas bantuan dalam aktivitas mahasiswa dan dosen di laboratorium.

2) *Planning*

Dalam tahap ini, peneliti akan mengorganisir rangkaian *task* atau modul yang akan dikerjakan dalam setiap iterasi, berdasarkan *User Stories*

yang terkumpul pada tahap pengumpulan kebutuhan (*Requirements*). Peneliti akan membuat daftar tugas tersebut dengan menggunakan teknik perencanaan yang disebut *practice planning game*. Dalam proses Planning Game, pengguna akan membuat *stories* pengguna berdasarkan kebutuhan dan keinginan mereka terhadap apa yang sistem dapat lakukan. Setelah mendapatkan cerita pengguna tersebut, peneliti akan memperkirakan waktu yang dibutuhkan untuk menyelesaikan tugas-tugas tersebut. Estimasi waktu pengerjaan untuk masing-masing *user stories* akan diwakili oleh *story points.*

3) *Iteration Initialization*
   Pada tahap *iteration initialization* merupakan tahapan yang memeperlihatkan dimulainya pelaksanaan *user stories* dari setiap iterasi. Daftar *user stories* yang akan dikerjakan dieperoleh dari hasil perencanaan pada tahap *planning.*
4) *Design*
   Pada tahap *Design*, peneliti memodelkan alur kerja sistem yang nantinya akan di implementasikan selama proses iterasi. *Design* alur kerja sistem tersebut hanya memenuhi kebutuhan dari klien/ pihak terkait tanpa adanya design tambahan untuk kebutuhan pengguna yang nantinya berubah. Ini juga dicocokan dengan SOP yang berlaku pada laboratorium tersebut. Adapun *design* lain yang dibuat adalah *design database*. *Design database* yang dibuat berupa skema tabel *design* untuk dijadikan sebagai *database* pada sistem nantinya. *Design database* ini didefinisikan sebagai proses bisnis oleh peneliti, dibuat dalam skema yang berisi tabel, atribut, dan *constraint* antar tabel.
5) *Implementation*
   Fase *Implementation* merupakan tahap di mana peneliti membuat kode pemrograman dan menerapkan objek-objek yang telah dirancang pada tahap *Design*. Tahap ini terdiri dari tiga sub-tahap, yakni *Unit Testing*, *Code Generation*, dan *Code Refactoring.*
6) *System Testing*
   Tahapan ini merupakan tahapan pengujian sistem yang telah dibuat sebelumnya untuk mengetahui kekurangan pada sitem yang telah dibuat. Pengujian fungsionalitas sistem ini menggunakan *Black-box testing*. *Black-box testing* merupakan metode pengujian perangkat lunak di mana pengujian dilakukan tanpa mengetahui detail bagaimana program bekerja di dalamnya. Pengujian dilakukan dari sudut pandang pengguna atau pengamat eksternal. Selama pengujian *Black-box testing*, *input* diberikan ke program dan *output* yang dihasilkan dievaluasi untuk memeriksa apakah hasilnya sesuai dengan yang diharapkan. Pengujian dapat dilakukan dengan input yang valid dan juga input yang tidak valid. Keuntungan dari *Black-box testing* adalah dapat memastikan bahwa program berfungsi dengan benar dari perspektif pengguna atau pengamat eksternal, sehingga dapat menemukan kesalahan yang tidak dapat dilihat dari perspektif pengembang.
7) *Retrospective*
   Pada tahap akhir dari proses iterasi, dilakukan *retrospective*. Pada tahap ini, peneliti menganalisis jalannya setiap fase pengembangan modul, kesesuaian estimasi waktu pengerjaan, penyebab terjadinya keterlambatan dalam proses pengembangan, serta mengambil langkah untuk mencegah hal tersebut terulang kembali pada iterasi selanjutnya.

B. Rancangan Sistem

1) Analisis Sistem
   Pada penelitian ini, peneliti melakukan analisis kebutuhan terhadap sistem informasi Pengelolaan Laboratorium Kuliner untuk Jurusan Tataboga Universitas Negeri Surabaya. Analisis kebutuhan ini dilakukan dengan melakukan analisis terhadap kebutuhan yang ada pada sistem informasi Pengelolaan Laboratorium Kuliner untuk Jurusan Tataboga Universitas Negeri Surabaya. Adapun lebih jelasnya dapat dilihat pada tabel berikut :

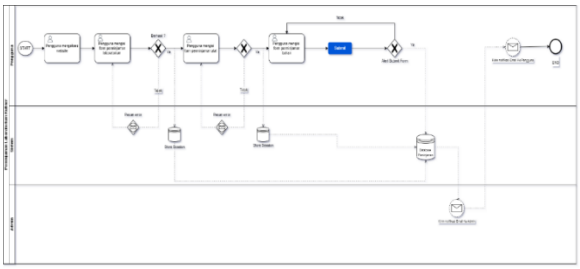
TABEL II
KEBUTUHAN SISTEM

No.	Nama	Kebutuhan Sistem
1.	Admin	Login, Logout, Input Data, Edit Data, Hapus Data, Cetak Laporan, Mengatur data peminjaman Lab, Mengatur data Alat dan Bahan, Mengatur Pengembalian.
2.	Mahasiswa	Membaca Petunjuk Penggunaan, Melakukan Peminjaman Lab, Melihat sisa dari alat dan bahan.

a. Perangkat Lunak
- XAMPP (PHP & MySQL)
- Composer
- Text Editor (Visual Studio Code, Sublime Text, Notepad ++ dll)
- Browser (Chrome, Microsoft Edge, Opera dll)
- OS Windows (Versi 7, 8 ,10, 11), MacOS, Linux

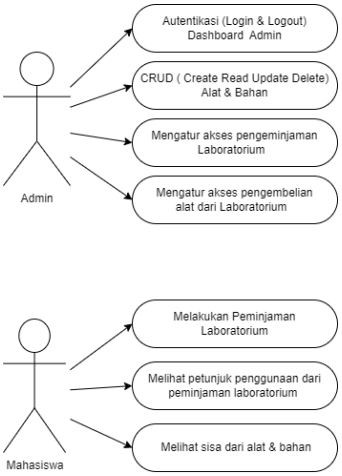
b. Perangkat Keras
- Device (Desktop/ Laptop)
- CPU Intel Core i5 Gen 8/AMD Ryzen 3000 series
- RAM (Random Access Memory) 8 Gigabyte DDR4
- Storage 1 Gigabyte

C. Alur Kerja Sistem



Gambar. 2 Alur Kerja Sistem

Pada Gambar 2 terdapat alur kerja sistem, dimana pada sistem tersebut terdapat 2 pengguna yaitu Admin dan Mahasiswa. Untuk melakukan peminjaman, terlebih dahulu mahasiswa harus terhubung ke internet, setelah itu mahasiswa untuk lebih memahami bagaimana alur peminjaman laboratorium tersebut, mahasiswa dapat pergi ke halaman petunjuk petunjuk penggunaan. Jika sudah maka mahasiswa bisa lanjut untuk melakukan peminjaman laboratorium, setelah itu jika sudah melakukan peminjaman laboratorium dan sudah mengecek data yang diinputan tidak ada yang salah, maka mahasiswa lanjut untuk melakukan peminjaman alat.



Gambar. 3 Use Case Diagram

Pada Gambar 3 terdapat sebuah ilustrasi use case diagram dengan terdapat 2 aktor yaitu admin dan mahasiswa. Aktor admin dapat melakukan login, logout, input data, edit data, hapus data, cetak laporan peminjaman, mengatur akses pengembalian. Sedangkan aktor mahasiswa dapat melakukan peminjaman laboratorium serta melihat petunjuk penggunaan dari peminjaman laboratorium. Hal lain yang bisa dilakukan oleh aktor mahasiswa adalah melihat sisa dari alat & bahan yang tersedia.

IV. HASIL DAN PEMBAHASAN

A. Planning

Pada tahap ini, peneliti merencanakan proses pengembangan sistem berdasarkan *requirements* yang telah didapatkan. Dalam tahap ini, akan menentukan prioritas, esrimasi waktu yang diperlukan untuk perngerjaan, dan urutan pengerjaan untuk setiap *user stories* serta mengevaluasi jumlah iterasi yang diperlukan selama pengembanga. Proses perencanaan ini meliputi estimasi *user stories*, menentukan prioritas *user stories*, dan merencanakan iterasi.

.

1) Estimasi *User Stories*

Pada tahap ini, peneliti akan menentukan estimasi waktu yang dibutuhkan untuk menyelesaikan setiap user stories yang ada. Estimasi waktu ini diukur dalam satuan story point yang berbanding lurus dengan durasi waktu yang dibutuhkan. Data Hasil estimasi waktu pengerjaan dapat dilihat pada tabel 4.1 berikut:

TABEL IIII
HASIL ESTIMASI USER STORIES

Kode *User Stories*	*Story Points*
ADM-01	3
ADM-02	3

MHS-01	6
MHS-02	1
MHS-03	3
MHS-04	2
MHS-05	3

Tabel III menunjukkan hasil estimasi waktu pengerjaan untuk masing-masing *user stories*. Estimasi waktu pengerjaan untuk 1 *story point* adalah 2 hari. Hal ini menunjukkan estimasi waktu yang dibutuhkan untuk pengerjaan pada *user stories* yang *memiliki story point* 1 adalah 2 hari, *story point* 2 adalah 4 hari, *story point* 3 adalah 6 hari, *story point* 4 adalah 8 hari, dan *user stories* yang memiliki *story points* 6 adalah 12 hari.

2) Penentuan Prioritas *User Stories*

Setelah menetapkan *story point* untuk setiap *user stories*, peneliti dan klien akan berdiskusi untuk menentukan urutan prioritas dari *user stories.* Proses penentuan prioritas dilakukan dengan menyortir *user stories* berdasarkan *value* dan *risk.* Untuk menentukan *value* dan *risk* dari *user stories*, digunakan tiga kategori nilai bisnis, yaitu *citical*, *significant business value*, dan *nice to have.*

TABEL IIIV
HASIL PRIORITAS USER STORIES

Kode *User Stories*	*Story Points*
ADM-01	*Critical*
ADM-02	*Critical*
MHS-01	*Critical*
MHS-02	*Critical*
MHS-03	*Critical*
MHS-04	*Critical*
MHS-05	*Critical*

Tabel IV menunjukkan hasil penentuan *value* untuk setiap *user stories*. Setelah *value* untuk setiap *user stories* telah ditetapkan, langkah selanjutnya adalah menentukan *risk* dari masing-masing *user stories*.

3) Penyusunan Daftar Iterasi

Setelah melakukan estimasi waktu dan menentukan prioritas untuk masing-masing *user stories*, tahap *planning* dilanjutkan dengan merencanakan iterasi dalam pengembangan menggunakan PXP yang berjalan secara berulang. Iterasi melibatkan penyelesaian *user stories* selama proses pengembangan berlangsung. Penentuan *user stories* yang akan diselesaikan dalam setiap iterasi dilakukan melalui diskusi dengan klien dan didasarkan pada prioritas *user stories* tersebut, hingga *velocity* yang diinginkan tercapai. *Velocity* adalah nilai yang digunakan oleh pengembang/peneliti untuk menentukan durasi iterasi. *Story point* dari setiap *user stories* akan dijumlahkan sampai totalnya mencapai nilai *velocity*, yang akan membentuk iterasi pertama. *User story* yang tidak termasuk dalam iterasi pertama akan dijumlahkan kembali untuk membentuk iterasi berikutnya. Proses ini dilakukan hingga semua *user stories* terakomodasi. Beberapa *user stories* yang diperoleh saling terkait dan bergantung satu sama lain, seperti yang dijelaskan dalam detail *user stories* dari klien. Hal ini menunjukkan bahwa *user stories* tersebut harus dipertimbangkan dalam pengembangan sistem informasi. Hasil perencanaan iterasi dapat dilihat pada tabel V.

TABEL V
HASIL PENYUSUAN DAFTAR ITERASI

Iterasi 1			
User Stories	Prioritas		Story Points
	Value	Risk	
MHS-01	*Critical*	3 (Medium)	6
Velocity			6
Iterasi 2			
User Stories	Prioritas		Story Points
	Value	Risk	
ADM-01	*Critical*	3 (Medium)	3
ADM-02	*Critical*	2 (Medium)	3
Velocity			6
Iterasi 3			
User Stories	Prioritas		Story Points
	Value	Risk	
MHS-02	*Critical*	1 (Low)	1
MHS-03	*Critical*	1 (Low)	2
MHS-04	*Critical*	4 (Medium)	3
Velocity			6
Iterasi 4			
User Stories	Prioritas		Story Points
	Value	Risk	
MHS-05	*Critical*	2 (Medium)	3
Velocity			3

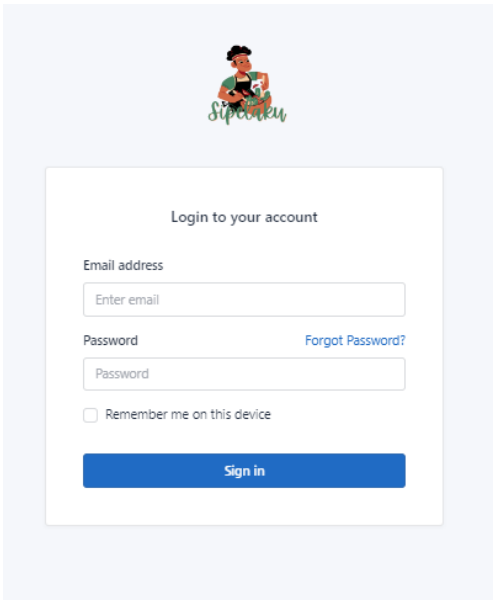
Dalam hal ini, setiap iterasi membutuhkan waktu 12 hari untuk diselesaikan dengan *velocity* 6 yang ditetapkan oleh pengembang, kecuali pada Iterasi

4 dengan *velocity* yang totalnya adalah 3 dimana hal tersebut hanya membutuhkan waktu 6 hari saja. Ada 7 *user stories* yang direncanakan untuk diselesaikan dalam waktu 42 hari. Dalam tahap ini, klien berhak mengajukan *user stories* baru atau mengubah yang sudah ada selama pengembangan sistem berlangsung. Pengembang akan menilai dan mengurutkan user stories tersebut, kemudian memasukkannya ke dalam iterasi saat ini atau iterasi selanjutnya.

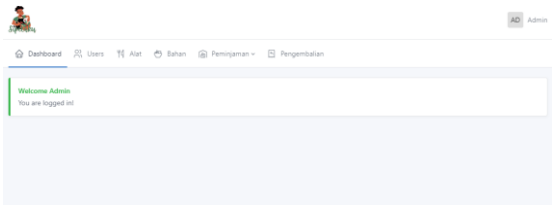
B. *Hasil Setiap Iterasi*

Setelah melakukan tahap *planning*, selanjutnya akan dilakuakan perencanaan iterasi untuk mengembangkan sistem yang terdiri dari 4 iterasi. Hasil dari setiap iterasi di tampilkan dengan dari *design* yang telah di buat dari *CRC Cards* dan *Prototype.*

1) Iterasi 1



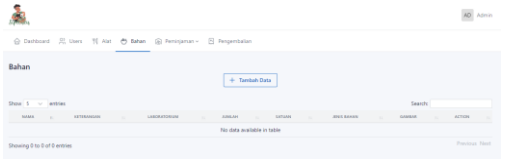
Gambar. 4 Tampilan *Login User Admin*



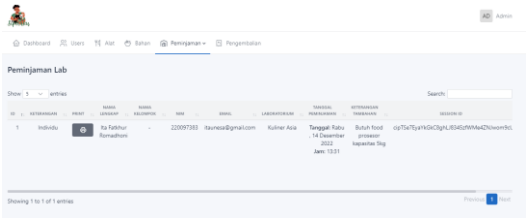
Gambar. 5 Tampilan *Dashboard Admin*

2) Iterasi 2

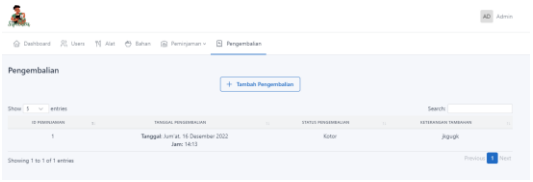
Gambar. 6 Tampilan *Dashboard* Alat



Gambar. 7 Tampilan *Dashboard* Bahan

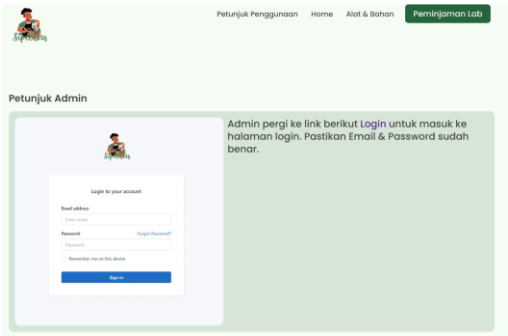


Gambar. 8 Tampilan *Dashboard* Peminjaman

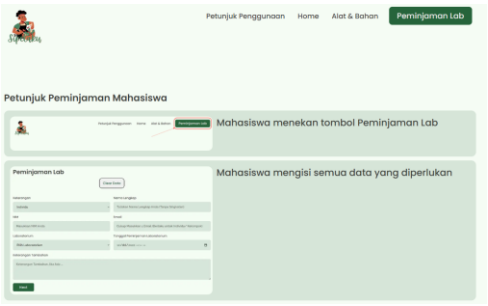


Gambar. 9 Tampilan *Dashboard* Pengembalian

3) Iterasi 3



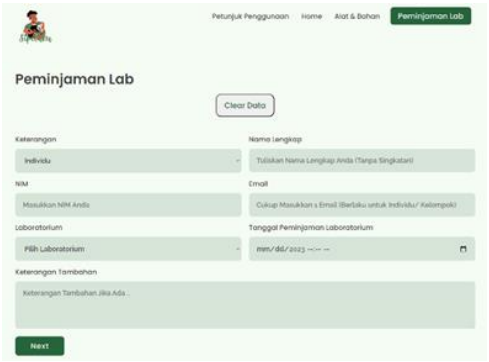
Gambar. 10 Tampilan Petunjuk Penggunaan *Admin*



Gambar. 11 Tampilan Petunjuk Penggunaan Mahasiswa

Gambar. 12 Tampilan Informasi Peminjaman 1

Gambar. 13 Tampilan Informasi Peminjaman 2



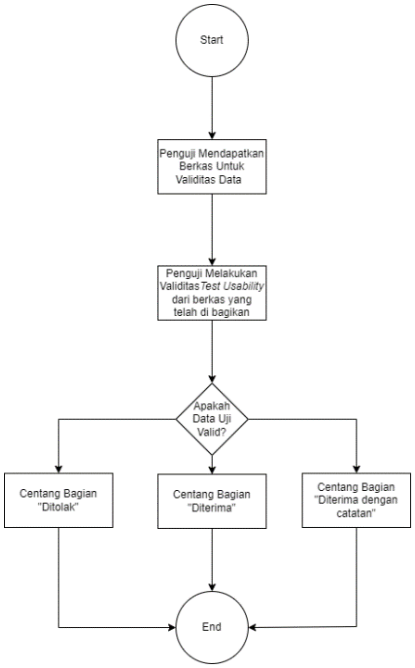
Gambar. 14 Tampilan Simplifikasi Peminjaman

4) Iterasi 4

Gambar. 15 Tampilan *User Interface Homepage*

C. *Analisa Usability*

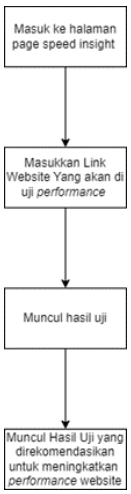
Pada analisa *usability*, terdapat *user acceptance test* yang telah dilakukan oleh pengembang pada iterasi-1 sampai dengan iterasi ke-4. Adapun *user acceptance test* tersebut juga pernah di lakukan oleh laboran serta *developer* yang memang ahli dalam bidang website sehingga hasil pengujian tersebut dapat memberikan perspektif yang komprehensif mengenai kelayakan dan kemudahan penggunaan sistem. Berikut adalah gambar unuk mekanisme dari pengujian *usability testing*.



Gambar. 16 Uji Validitas *Usability*

D. *Analisa Usability*

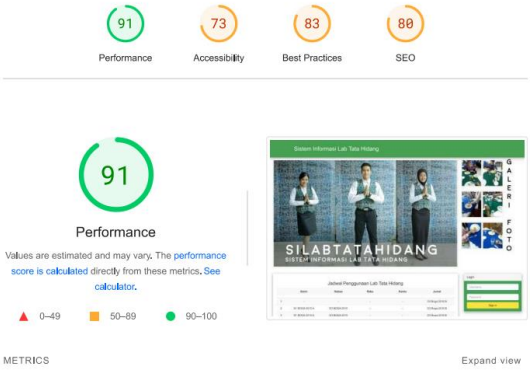
Sama halnya pada bagian usability, pada analisis performance, pengujian dilakukan untuk mengevaluasi kinerja sistem dalam hal kecepatan dan responsivitas. Berikut adalah gambar unuk mekanisme dari pengujian *performance testing*.



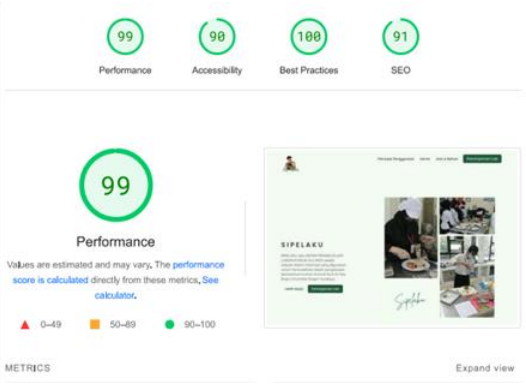
Gambar. 17 Uji Validitas *Performance*

Uji performance dilakukan dengan *tool PageSpeed Insights*. PageSpeed Insights adalah alat (*tool*) yang dikembangkan oleh Google untuk menganalisis performa sebuah halaman *web*site dan memberikan rekomendasi perbaikan yang dapat meningkatkan kecepatan dan kinerja halaman tersebut. Alat ini membantu pengembang web dan pemilik situs dalam memahami bagaimana halaman web dengan memuat dan memberikan saran untuk meningkatkan waktu muat dan pengalaman pengguna. Uji *performance* juga dilakukan dengan 2 kasus, yaitu yang pada tampilan *mobile* dan *desktop*. berikut adalah hasil dari pengujian tersebut.

1) Uji *Performance* Tampilan *Desktop*



Gambar. 18 Tampilan *Desktop* Silabtatahidang

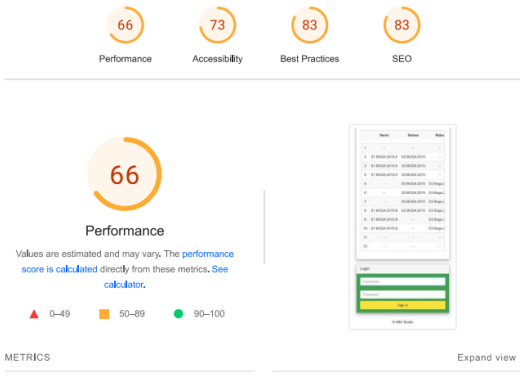


Gambar. 19 Tampilan *Desktop* Sipelaku

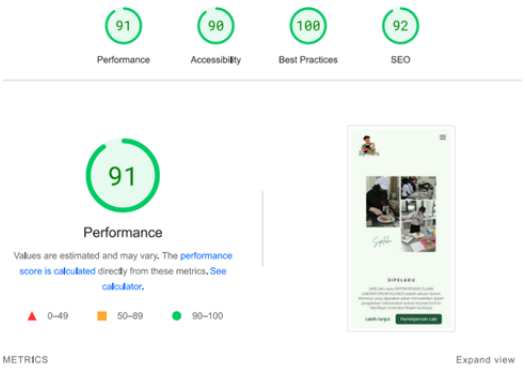
Terlihat bahwa dalam segi *performance* pada website silabtatahidang mendapatkan skor 91**,** sedangkan pada website sipelaku mendapatkan skor 99. Perbandingan skor kinerja ini memberikan gambaran tentang sejauh mana kedua website tersebut dapat diakses dengan cepat dan efisien pada perangkat desktop. Skor yang lebih tinggi menunjukkan bahwa website sipelaku memiliki performa yang lebih baik dalam memuat pada tampilan *desktop*.

Dengan hasil ini, dapat disimpulkan bahwa website sipelaku memiliki performa yang lebih baik dalam memuat pada tampilan desktop jika dibandingkan dengan website silabtatahidang. Skor 99 yang diperoleh oleh website sipelaku menunjukkan bahwa halaman web tersebut dapat diakses dengan cepat dan efisien, memberikan pengalaman pengguna yang lebih baik pada perangkat *desktop*.

2) Uji *Performance* Tampilan *Mobile*



Gambar. 20 Tampilan *Mobile* Silabtatahidang



Gambar. 21 Tampilan *Mobile* Sipelaku

Dalam gambar tersebut, terlihat bahwa pada segi *performance*, website silabtatahidang mendapatkan skor hanya 66, sedangkan website sipelaku mendapatkan skor 91. Perbandingan skor kinerja ini memberikan gambaran tentang sejauh mana kedua website dapat diakses dengan cepat dan efisien pada perangkat *mobile*.

V. KESIMPULAN DAN SARAN

A. *Kesimpulan*

Hasil penelitian selama pengembangan sistem informasi peminjaman laboratorium yang diberi nama SIPELAKU menunjukkan bahwa metode pengembangan *Personal Extreme Programming* dapat diterapkan dalam studi kasus ini.

1) Sistem informasi pengelolaan laboratorium kuliner yang dikembangkan dalam penelitian ini efektif dan efisien dalam mendukung kegiatan akademik jurusan D4 Tata Boga dan S1 Pendidikan Tata Boga Universitas Negeri Surabaya. Sistem ini memberikan kemudahan dalam proses peminjaman alat atau bahan di laboratorium kuliner, meningkatkan aksesibilitas data, dan mendukung kegiatan akademik yang lebih efisien.
2) Penelitian ini berhasil meningkatkan aksesibilitas data dan pelayanan bagi mahasiswa dalam proses peminjaman alat atau bahan di laboratorium kuliner. Mahasiswa dapat dengan mudah mengakses informasi terkait ketersediaan alat atau bahan, serta melakukan peminjaman dengan proses yang lebih efisien.
3) Integrasi antara sistem informasi pengelolaan laboratorium kuliner yang baru dengan sistem informasi yang sudah ada sebelumnya tercapai dengan sukses. Hal ini memungkinkan penggunaan yang lebih efisien dari sumber daya yang tersedia dan meningkatkan koordinasi antara Laboran dan Mahasiswa yang melakukan peminjaman.
4) Penggunaan metode *Personal Extreme Programming (PXP)* dan pengujian *Black-box testing* dalam pengembangan sistem ini berhasil meningkatkan usability dan performance. Metode PXP memungkinkan kolaborasi yang intensif antara tim pengembang dan klien, sementara pengujian *Black-box testing* membantu mengidentifikasi dan memperbaiki potensi kesalahan dalam sistem, sehingga menghasilkan sistem yang lebih baik dan lebih responsif terhadap kebutuhan pengguna.

B. *Kesimpulan*

Berdasarkan pengembangan yang dilakukan oleh peneliti pada penelitian ini, terdapat beberapa saran yang dapat diberikan untuk pengembangan dan pemeliharaan sistem informasi pengelolaan laboratorium kuliner yang lebih baik, antara lain:

1) Dalam mengembangkan sistem informasi pengelolaan laboratorium kuliner di masa depan, penting untuk terus mempertahankan dan memperbarui sistem agar tetap relevan dengan perkembangan kebutuhan pengguna dan teknologi terkini.
2) Perlu dilakukan pemantauan dan evaluasi secara rutin terhadap aksesibilitas data dan pelayanan bagi mahasiswa. Pengumpulan umpan balik dari pengguna sistem dapat membantu dalam meningkatkan dan memperbaiki pengalaman pengguna.
3) Dalam mengintegrasikan sistem informasi pengelolaan laboratorium kuliner dengan sistem yang sudah ada sebelumnya, perlu dilakukan analisis yang mendalam tentang kesesuaian antara kedua sistem tersebut. Perencanaan dan koordinasi yang baik diperlukan untuk

mengurangi potensi hambatan dan memastikan integrasi yang tepat.

4) Metode Personal Extreme Programming (PXP) dan pengujian Black-box testing dapat terus diterapkan dalam pengembangan sistem-sistem lainnya untuk meningkatkan usability dan performance. Penerapan metodologi pengembangan yang adaptif dan uji coba secara menyeluruh dapat membantu menghasilkan solusi yang lebih baik dan meminimalkan kesalahan

REFERENSI

[1] S. Supeni and L. A. Hakim, *MENGENAL POTENSI DESA WISATA DESA PLESUNGAN KECAMATAN GONDANGREJO KABUPATEN KARANGANYAR*, Pertama. Surakarta: UNISRI Press, 2020.

[2] M. Zhang, *Pornography, Ideology, and the Internet: A Japanese Adult Video Actress in Mainland China*. Maryland: Lexington Books, 2018.

[3] A. Akhtar, B. Bakhtawar, and S. Akhtar, "Extreme Programming VS Scrum: A Comparison Of Agile Models," *International Journal of Technology, Innovation and Management (IJTIM)*, vol. 2, no. 2, pp. 80–96, 2022, doi: 10.54489/ijtim.v2i1.77.

[4] D. H. Ardianzah, I. Nuryasin, and B. S. Wiyono, "Pengembangan Sistem Pengelolaan Peminjaman Auditorium Universitas Muhammadiyah Malang Berbasis Web Menggunakan Metode Personal Extreme Programming," *REPOSITOR*, vol. 4, no. 2, pp. 137–146, May 2022.

[5] M. A. Firmansyah, N. Ramsari, and A. D. Rachmanto, "Rancang Bangun Sistem Informasi Penjualan Pada Toko Buku Kita Tasikmalaya Berbasis Web Menggunakan Framework Laravel 8," *Jurnal FIKI*, vol. XII, no. 1, pp. 2087–2372, May 2022, [Online]. Available: http://jurnal.unnur.ac.id/index.php/jurnalfiki

[6] M. Ulfi, G. I. Marthasari, and I. Nuryasin, "Implementasi Metode Personal Extreme Programming dalam Pengembangan Sistem Manajemen Transaksi Perusahaan (Studi Kasus: CV. Todjoe Sinar Group)," *REPOSITOR*, vol. 2, no. 3, pp. 261–268, Mar. 2020.

[7] S. A. Asri, G. Made, A. Sunaya, P. M. Prihatini, and W. Setiawan, "Comparing Traditional and Agile Software Development Approaches: Case of Personal Extreme Programming," *Atlantis Highlights in Engineering (AHE) Proceedings of the International Conference on Science and Technology (ICST 2018)*, vol. 1, pp. 561–563, Dec. 2018, doi: https://doi.org/10.2991/icst-18.2018.116.

[8] M. van Deurzen, "The anatomy of the modern window manager A case study in developing a novel top-level reparenting list-based tiling window manager for X in an Agile manner," Thesis, Radboud University, Nijmegen, 2019.

[9] A. Hoffman, *Web Application Security Exploitation and Countermeasures for Modern Web Applications*, First Edition., vol. 1. Sebastopol, CA 95472: O'Reilly Media, Inc, 2020.

[10] R. M. Stair and G. W. Reynolds, *Principles of information systems*, Thirteenth Edition. Boston: Cengage Learning, 2018.

[11] T. Butler, *PHP & MySQL: Novice to Ninja*, Seventh Edition. Site Point, 2022.

[12] D. H. Thomas, *Laravel (8) PHP Framework Handbook | Start From Zero Knowledge.: Ideal programming book for anyone who wants to learn more about the implementation, practical use of Laravel 8.x*, First Edition. Prevent Care/ Independent, 2021.

[13] S. bin Uzayr, *Mastering MySQL for the Web: A Beginner's Guide*, First Edition. CRC Press, 2022. doi: https://doi.org/10.1201/9781003229629.

[14] D. Correa and P. Vallejo, *Practical Laravel Develop clean MVC web applications*, First Edition. Independently, 2022.

[15] A. Mardan, *Full Stack JavaScript Learn Backbone.js, Node.js, and MongoDB Second Edition*, Second Edition. San Francisco, California, USA: Apress Media LLC, 2018. doi: 10.1007/978-1-4842-3718-2.

[16] J. Sande and M. Galloway, *Dart Apprentice: Beginning Programming with Dart*, First Edition. Razeware LLC, 2021.

[17] H. I. Hasan, G. I. Marthasari, and I. Nuryasin, "Implementasi Metode Personal Extreme Programming Dalam Pengembangan Sistem Administrasi Pelayanan Desa (Studi Kasus: Desa Bulangan Barat Kec. Pegantenan Kab. Pamekasan)," *REPOSITOR*, vol. 3, no. 1, pp. 43–50, Jan. 2021.