



Available online at :

<https://ejournal.unsrat.ac.id/v3/index.php/IJIDS/index>
**IJIDS**

(Indonesian Journal of Intelligence Data Science)



## Sistem Informasi Apotek Berbasis Web Menggunakan Metode Personal Extreme Programming (XP Programming) Dengan Arsitektur MVC

Cindy E Ch Lahama<sup>1</sup>, Christie Ellyanne Juliet Clara Montolalu<sup>2</sup>, Edwin Tenda<sup>3</sup><sup>1,2</sup> Program Studi Sistem Informasi, Jurusan Matematika, FMIPA, UNSRATe-mail: <sup>1</sup>[lahamacindy@gmail.com](mailto:lahamacindy@gmail.com), <sup>2</sup>[christestelly.com](mailto:christestelly.com), <sup>3</sup>[edwin.phill@icloud.com](mailto:edwin.phill@icloud.com)

### ARTICLE INFO

#### History of the article:

Received January 1, 2024

Revised January 21, 2024

Accepted March 12, 2024

#### Keywords:

3 to 5

Keywords

#### Correspondece:

Christie Ellyanne Juliet

Clara Montolalu

E-mail:

[christestelly.com](mailto:christestelly.com)

### ABSTRAKSI

A web-based drug store data framework is outlined to encourage the productive administration and deals of solutions in a drug store with consistent integration. To realize this objective, the Individual Extraordinary Programming (XP) strategy has been received to oversee the development of this framework. Within the setting of this venture, the XP strategy is custom fitted to a littler advancement environment. The drug store data framework is outlined with the Model-View-Controller (MVC) engineering, which isolates commerce rationale, introduction, and client interaction into unmistakable components. This points to improve adaptability, versatility, and generally framework practicality. The results of this extend are anticipated to provide a web-based drug store data framework open through a user-friendly interface, proficient in pharmaceutical stock administration, and supporting exact installment frameworks. As a result, it is expected that operational proficiency and client benefit will altogether progress.

Keyword: Model-View-Controller (MVC) Architecture, Medicine Inventory Management, Personal Extreme Programming (XP), Pharmacy Information System.

### PENDAHULUAN

Menurut Permenkes RI No.9 Tahun 2017 tentang Apotek, apotek adalah sarana pelayanan kefarmasian tempat dilakukan praktek kefarmasian oleh Apoteker. Dalam menjalankan pekerjaan kefarmasian, apoteker harus menerapkan standar pelayanan kefarmasian yang bertujuan untuk meningkatkan mutu pelayanan kefarmasian, menjamin kepastian hukum bagi tenaga kefarmasian, dan melindungi pasien dan masyarakat dari penggunaan obat yang tidak rasional dalam rangka keselamatan pasien. [1]

Apotek Dira Farma merupakan salah satu Apotek yang ada di kota Manado yang sampai saat ini belum memanfaatkan sistem informasi dalam proses bisnis yang dijalankan. Apotek Dira Farma memiliki masalah dalam manajemen persediaan produk, pencatatan data produk dan pembukuan penjualan produk. Manajemen persediaan yang terpisah-pisah dan masih manual (ratusan item produk belum dipantau secara *up to date*), pencatatan stok yang dilakukan petugas masih manual, sehingga rawan terjadinya kesalahan pada akhirnya berdampak terhadap ketidakakuratan informasi yang tersedia. Mengingat begitu banyak data

yang harus dicatat, maka pencatatan secara manual tentu saja akan memakan waktu serta rawan terjadi kesalahan dalam menginput data dan pengecekan data, sehingga perlu adanya sistem yang terkomputerisasi yang dapat membantu petugas dalam mengelola data produk tersebut serta mempermudah petugas dalam melakukan pengecekan terkait waktu kadaluarsa produk dan juga dapat mempermudah pimpinan dalam melihat laporan-laporan yang dibutuhkan dengan cepat dan lengkap.

Sistem informasi apotek merupakan suatu sistem informasi yang dibuat untuk memudahkan dalam penyampaian informasi mengenai apotek secara meluas. Sistem ini mengumpulkan, menyimpan dan mengolah informasi mengenai obat-obatan dan mengatur penggunaan obat-obatan yang sejalan dengan perawatan pasien.

Berdasarkan latar belakang di atas, maka pada penelitian ini penulis akan mengembangkan suatu sistem informasi Apotek berbasis web menggunakan metode *Personal extreme programming* (PXP) dengan arsitektur *Model View* dan *Controller* (MVC) pada Apotek Dira Farma, dimana sistem informasi ini diharapkan dapat membantu dalam meningkatkan efektivitas dan kinerja apotek saat melakukan pengelolaan data produk, persediaan stok produk serta penjualan pada Apotek Dira Farma.

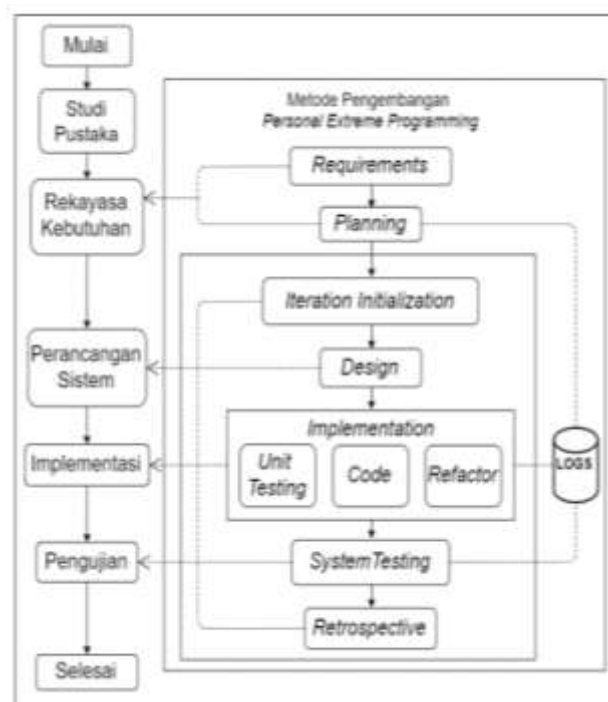
## METODE PENELITIAN

### A. Waktu dan Tempat Penelitian

Penelitian ini dimulai pada bulan Februari 2023, dengan tahap awal penyusunan proposal, pengumpulan data yang dilaksanakan pada Apotek Dita Farma, yang beralamat di Jl. 14 Februari, Teling Atas, Kec. Wanea, Kota Manado, Sulawesi Utara dan perancangan sistem web dengan pengujian sistem web. Perancangan sistem akan dilaksanakan di Laboratorium Komputer Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sam Ratulangi Manado.

### B. Metode Penelitian

Metode penelitian yang digunakan penulis pada penelitian kali ini terdiri dari beberapa tahapan. Tahapan tersebut bisa dilihat pada Gambar 1.



Gambar 1. Diagram Alir Penelitian. [2]

Pengembangan perangkat lunak ini menggunakan metode XP yang dapat diterapkan untuk menangani situasi proses pengembangan perangkat lunak dengan pemrogram tunggal, yang kemudian dinamakan dengan metodologi PXP. PXP terdiri dari beberapa fase, yaitu *requirements*, *planning*, *iteration initialization*, *design*, *implementation*, *system testing*, dan *retrospective*. [3]

#### 1. *Requirments* (Identifikasi kebutuhan sistem)

Pada tahap *requirements*, pengembang mengumpulkan kebutuhan yang diperlukan pada sistem yang akan dibangun. Pengumpulan kebutuhan dilakukan dengan cara melakukan wawancara kepada pimpinan, apoteker dan karyawan pada apotek Dira Farma. Pimpinan apotek menjelaskan bahwa segala proses yang terjadi pada penjualan, pembelian dan pengelolaan data produk di apotek Dira Farma masih dilakukan secara manual, dimana apoteker harus mencatat semua proses transaksi pembelian dan transaksi penjualan produk dalam buku untuk mendapatkan laporan penjualan serta proses pengecekan stok dan masa kadaluwarsa produk juga masih dilakukan manual setiap minggunya.

#### 2. *Planning*

Pada tahap ini pengembang akan melakukan serangkaian modul yang akan dikerjakan per iterasi berdasarkan *user story* yang telah didapatkan pada tahap sebelumnya.

#### 3. *Iteration Initialization*

*Iteration initialization* merupakan fase fungsionalitas yang sudah dibentuk dan diputuskan alur bisnis dari tahapan *planning* lalu dijabarkan dengan penggambaran diagram bahasa pemodelan standar menggunakan UML dimana diagram yang digunakan adalah *Usecase*, *Activity diagram*, dan *Sequence diagram*.

#### 4. *Design*

Tahapan sistem mulai didesain, mulai dari desain basis data menggunakan ERD dan desain tampilan pengguna (UI).

#### 5. *Implementation*

Pada tahapan ini dilakukannya pengkodean sistem dengan menggunakan bahasa pemrograman PHP dan *framework Laravel*. Kemudian untuk basis data akan menggunakan MySQL. Apabila kode selesai kemudian akan diuji dalam *unit testing*, jika kedapatan ada kesalahan maka dilakukan koreksi ulang atau *refactor* pada tahap dimana kesalahan tersebut bermula. Jika tidak ada kesalahan, maka dilanjutkan ke unit selanjutnya.

#### 6. *Sistem Testing*

Pada tahapan ini, dilakukan pengujian fungsionalitas sistem dengan metode *Black Box Testing*, dimana akan dilakukan dengan menjalankan unit atau modul dan diamati hasil dari unit. Hasil dari unit akan dilihat apakah sesuai atau tidak sesuai dengan proses bisnis yang diinginkan.

#### 7. *Retrospective*

*Retrospective* merupakan tahapan pengambilan kesimpulan terhadap sistem, apabila masih ada kesalahan maka akan dilakukan perbaikan mulai dari tahap *iteration initialization*. Fase *retrospective* berfungsi untuk

menganalisis hasil dari tiap tahapan yang dilalui. Dilakukan verifikasi mengenai kesesuaian waktu pengerjaan dengan rencana yang telah ditentukan sebelumnya.

## HASIL DAN PEMBAHASAN

### 4.1 Requirments

#### 1. Requirments

Kebutuhan sistem pada tahap ini didefinisikan melalui proses pengolahan hasil wawancara dengan pihak apotek dalam bentuk *user story*. Berdasarkan permasalahan yang telah diuraikan pada bab sebelumnya maka dibutuhkan suatu sistem website yang dapat memberikan alternatif bagi pihak apotek dalam menjalankan proses bisnis dengan efektif dan efisien. Kebutuhan tersebut dapat dituliskan dalam bentuk user story yang memiliki format seperti “*Sebagai ... saya ingin ... agar dapat ...*”. Setelah dilakukan fase kebutuhan, terdapat sepuluh user *stories* berdasarkan kebutuhan masing-masing actor yang dapat dilihat pada Tabel di bawah ini.

Tabel 1. Daftar User Story

| Kode User | User stories   |
|-----------|--|
| Story-01  | Sebagai admin, saya ingin memiliki akses untuk masuk kedalam system apotek dengan melakukan login dan logout sehingga karyawan tidak dapat mengakses system sebagai admin.   |
| Story-02  | Sebagai admin, saya ingin sistem ini dapat melakukan pengelolaan data produk antara lain dapat menambah, mengubah, dan menghapus data produk, sehingga admin tidak memerlukan buku untuk merekap stok produk yang ada. |
| Story-03  | Sebagai admin, saya ingin sistem ini mampu melakukan pengelolaan data supplier sehingga data supplier dapat tersimpan dalam sistem.  |
| Story-04  | Sebagai admin, saya ingin sistem ini mampu menjalankan transaksi pembelian produk dari supplier, sehingga memudahkan admin dalam proses transaksi dan pencatatan.  |
| Story-05  | Sebagai admin, saya ingin sistem ini mampu menjalankan transaksi penjualan produk untuk <i>customers</i> , sehingga memudahkan admin dalam proses transaksi dan pencatatan.  |
| Story-06  | Sebagai admin, saya ingin sistem ini dapat menampilkan laporan penjualan dan pembelian produk, sehingga mempermudah admin dalam pembukuan.   |
| Story-07  | Sebagai admin, saya ingin sistem ini dapat menampilkan pemberitahuan saat stok produk sudah dibawah 10 item, sehingga mempermudah admin dalam melakukan restok.  |

| Kode User | User stories   |
|-----------|--|
| Story-08  | Sebagai admin, saya ingin sistem ini dapat menampilkan pemberitahuan terkait masa expire produk setiap 3 bulan sebelum jatuh tempo expire produk, sehingga admin dapat dengan mudah melakukan restok.    |
| Story-09  | Sebagai kasir, saya ingin sistem ini mampu menjalankan transaksi penjualan produk untuk <i>customers</i> , sehingga memudahkan kasir dalam proses transaksi dan pencatatan.                              |
| Story-10  | Sebagai karyawan, saya ingin sistem ini dapat menyediakan fitur pencarian untuk melakukan cek data produk tanpa perlu melakukan login, sehingga mudah dalam melakukan cek data ketika melayani customer. |

Semua kebutuhan dari pihak apotek telah tercatat dalam user story berdasarkan actor yang nantinya akan menjalankan sistem yang dibuat. Selanjutnya user story inilah yang akan menjadi acuan oleh developer dalam membangun sistem website apotek.

## 2. Requirments

Setelah *user story* didapatkan, maka pada tahap ini *developer* akan menentukan perkiraan waktu untuk melakukan setiap iterasi. Dari sepuluh *user story*, terdapat tiga iterasi yang akan dilakukan. Pada iterasi pertama *developer* akan mengintegrasikan sistem login/logout pada sistem untuk actor admin, kemudian dilanjutkan dengan merancang sistem untuk dapat mengelola data produk dan supplier dengan waktu pengerjaan selama 3 minggu. Dilanjutkan dengan iterasi kedua dengan membuat sistem transaksi penjualan dan pembelian pada sistem website apotek serta fungsi untuk mengetahui laporan transaksi penjualan dan pembelian dengan estimasi waktu 5 minggu. Dan pada iterasi terakhir *developer* akan membuat sistem untuk menampilkan pemberitahuan terkait jumlah sisa stok produk min 10 item, *developer* juga akan membangun sistem dimana terdapat pemberitahuan terkait jatuh tempo kadaluwarsa produk setiap 2 bulan sebelum jatuh tempo, dan fitur yang terakhir dalam iterasi ini yaitu *developer* akan membuat fungsi pencarian untuk dapat menampilkan setiap produk yang cari tanpa perlu melakukan aktifitas login/logout. Tabel analisis kebutuhan sistem dapat dilihat pada Tabel dibawah ini :

Tabel 2. Analisis Kebutuhan sistem

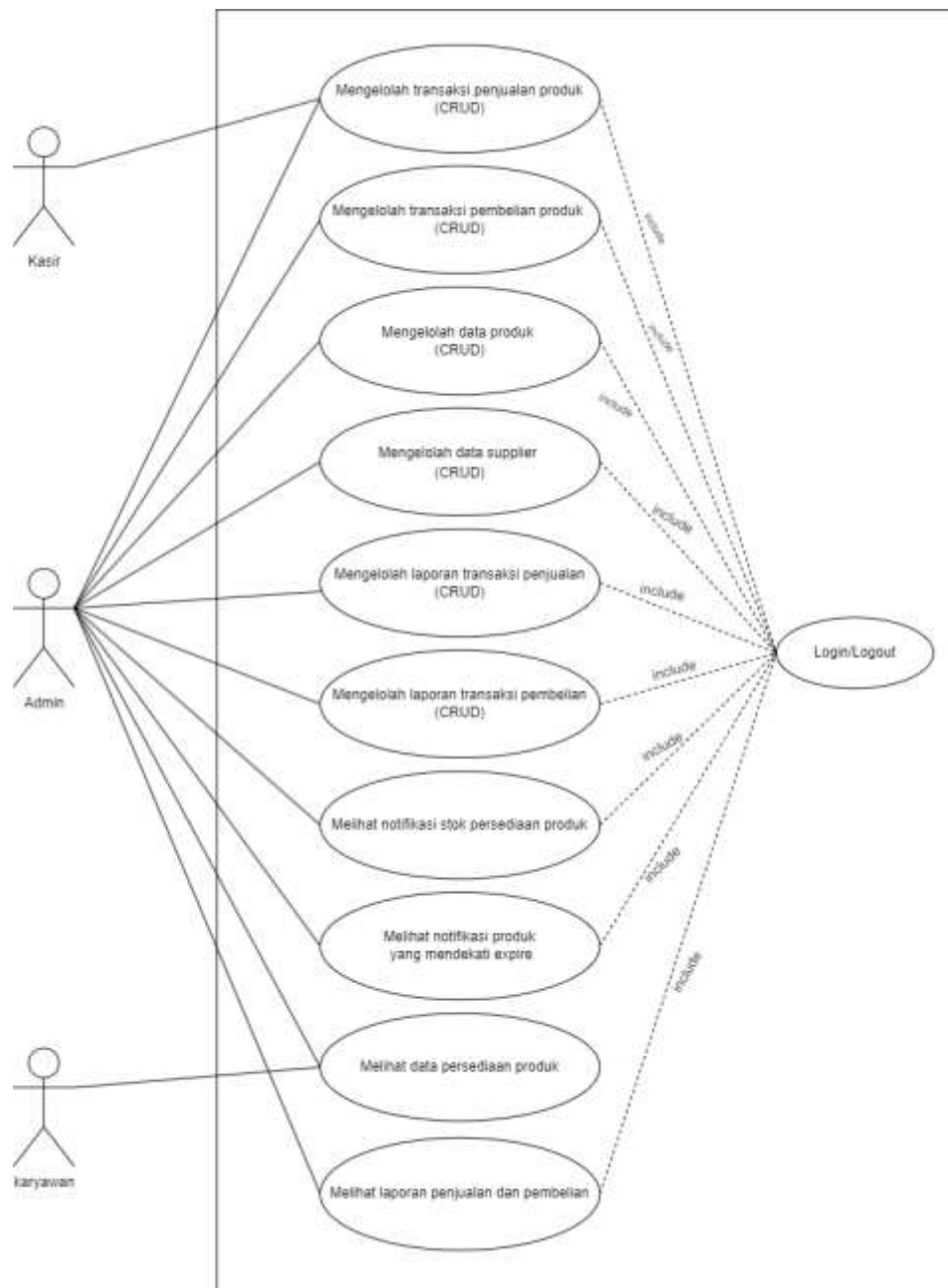
| No          | Kode User Story            | Deskripsi                                   | Estimasi (dalam minggu) |
|-------------|----------------------------|---|-------------------------|
| Iterasi – 1 |                            |   |                         |
| 1.          | US-01                      | Melakukan <i>Login</i> sistem               | 1 minggu                |
| 2.          | US-02                      | Kelola data produk                          | 1 minggu                |
| 3.          | US-03                      | Kelola data supplier                        | 1 minggu                |
| Iterasi – 2 |                            |   |                         |
| 4.          | US-04,<br>US-05<br>& US-09 | Melakukan transaksi penjualan dan pembelian | 3 minggu                |
| 5.          | US-06                      | Laporan penjualan dan pembelian             | 2 minggu                |
| Iterasi – 3 |                            |   |                         |
| 6.          | US-07                      | Sistem pemberitahuan sisa stok              | 1 minggu                |
| 7.          | US-08                      | Sistem pemberitahuan expire                 | 1 minggu                |
| 8.          | US-10                      | Menampilkan data lewat fitur pencarian      | 2 minggu                |

### 3. Iteration Initialization

Seperti yang telah dijelaskan pada bab sebelumnya, desain sistem pada penelitian ini meliputi *Usecase*, *Activity diagram*, dan *Sequence diagram*.

#### 1. Usecase diagram

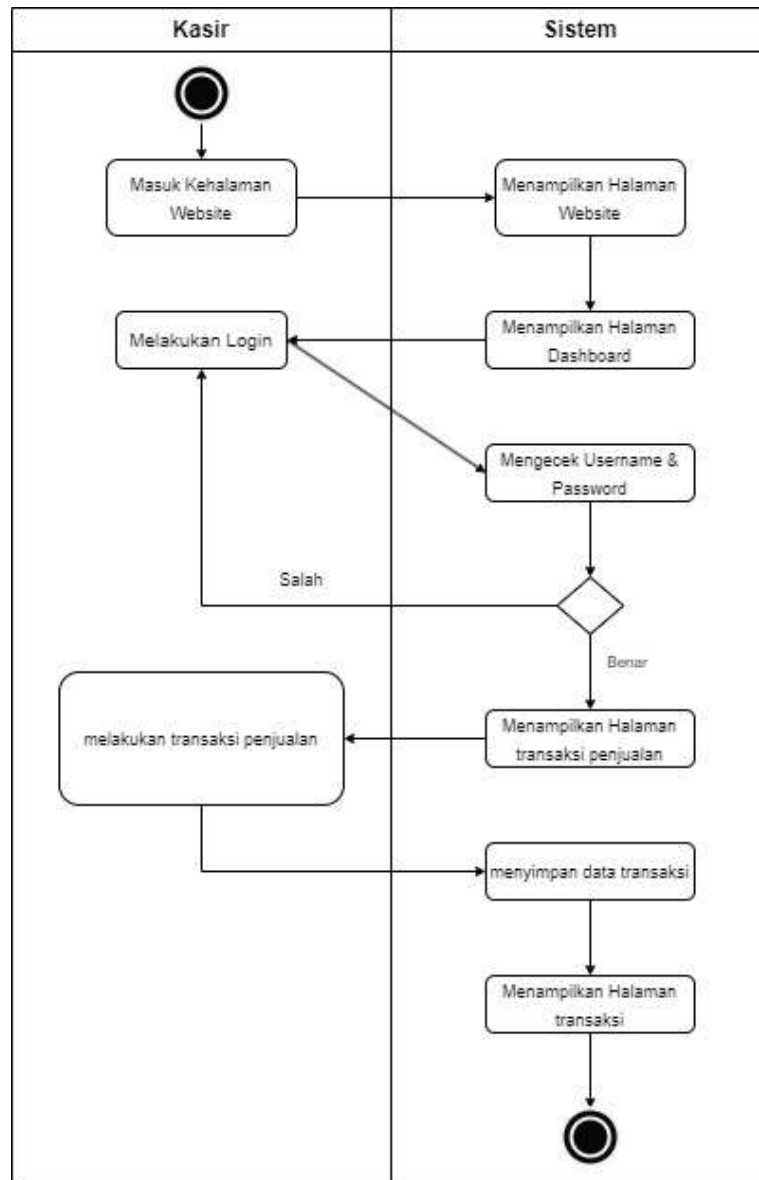
*Usecase diagram* yang ditampilkan akan digunakan untuk menjelaskan fitur-fitur yang dapat digunakan oleh pengguna atau *user*. Pada gambar dibawah merupakan *Use Case Diagram* yang terdapat aktor kasir, admin, dan karyawan dimana ketiga aktor mempunyai *usecase* masing-masing. Untuk aktor kasir memiliki satu usecase yaitu dapat mengelolah transaksi penjualan produk dan harus melakukan login ke sistem website. Untuk actor admin memiliki sepuluh usecase dimana admin dapat mengakses seluruh sistem yang ada dengan melakukan aktifitas login terlebih dahulu kedalam sistem, seperti mengelolah transaksi penjualan produk, mengelolah transaksi pembelian produk, mengelolah data produk, mengelolah data supplier, melihat dan mengelolah laporan transaksi penjualan dan pembelian, dapat melihat notifikasi stok persediaan produk dan notifikasi produk yang mendekati expire, admin juga dapat melihat data persediaan produk tanpa harus melakukan login ke sistem. *Use case diagram* pada sistem ini dapat dilihat pada Gambar 2:



Gambar 3. Usecase Diagram

## 2. Activity diagram

*Activity diagram* pada kasir menggambarkan aliran aktivitas atau aliran kerja yang dilakukan kasir di dalam sistem mulai dari awal kegiatan sampai kegiatan berakhir. Kasir akan membuka halaman website sistem akan menampilkan halaman dashboard, kasir akan melakukan login terlebih dahulu, jika login gagal maka sistem akan kembali ke halaman login, jika berhasil maka sistem akan langsung menampilkan halaman transaksi penjualan, dan jika kasir telah selesai melakukan proses transaksi maka sistem akan menyimpan bukti transaksi yang akan digunakan sebagai laporan transaksi penjualan. Activity diagram kasir dapat dilihat pada Gambar dibawah ini.

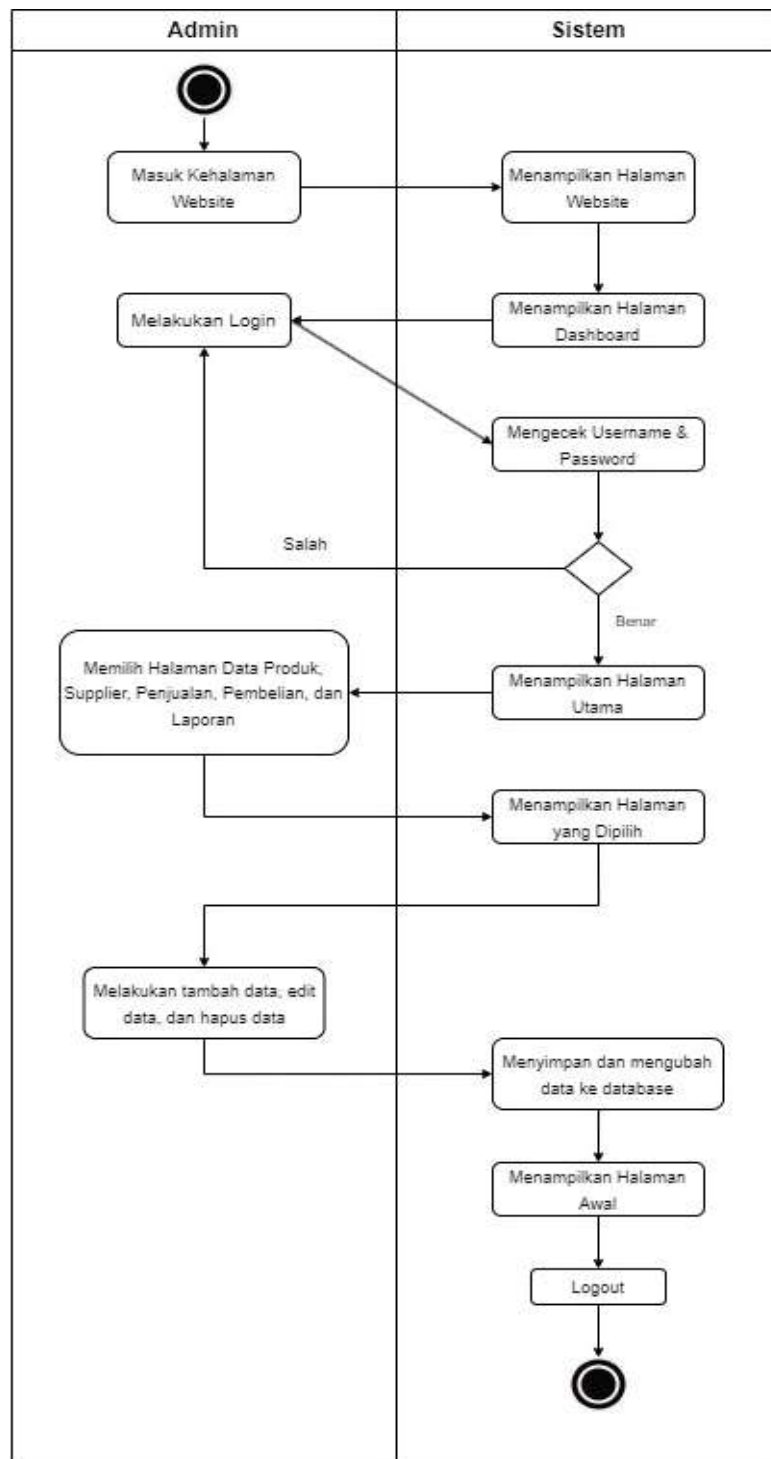


Gambar 4. Activity Diagram Kasir

### 3. Activity diagram

Pada admin menggambarkan aliran aktivitas atau aliran kerja yang dilakukan admin di dalam sistem mulai dari awal kegiatan sampai kegiatan berakhir. Admin akan membuka halaman website, sistem akan menampilkan halaman dashboard, admin akan melakukan login terlebih dahulu, jika login gagal maka sistem akan kembali ke halaman login, jika berhasil maka sistem akan langsung menampilkan halaman utama, dimana admin dapat memilih halaman yang akan dijalankan. Admin dapat mengakses seluruh menu yang dibuat pada sistem apotek, ada beberapa menu yang ditampilkan oleh sistem untuk admin yaitu, halaman produk, supplier, merk, kategori, penjualan, pembelian dan laporan. Ketika admin memilih menu yang akan digunakan, maka sistem akan menampilkan tampilan dari menu tersebut, selanjutnya admin dapat melakukan aktivitas edit data, tambah data dan hapus data, ketika selesai melakukan aktivitas edit data, tambah data dan hapus data maka sistem akan langsung memproses data tersebut ke database. Activity diagram admin dapat dilihat pada gambar dibawah ini.



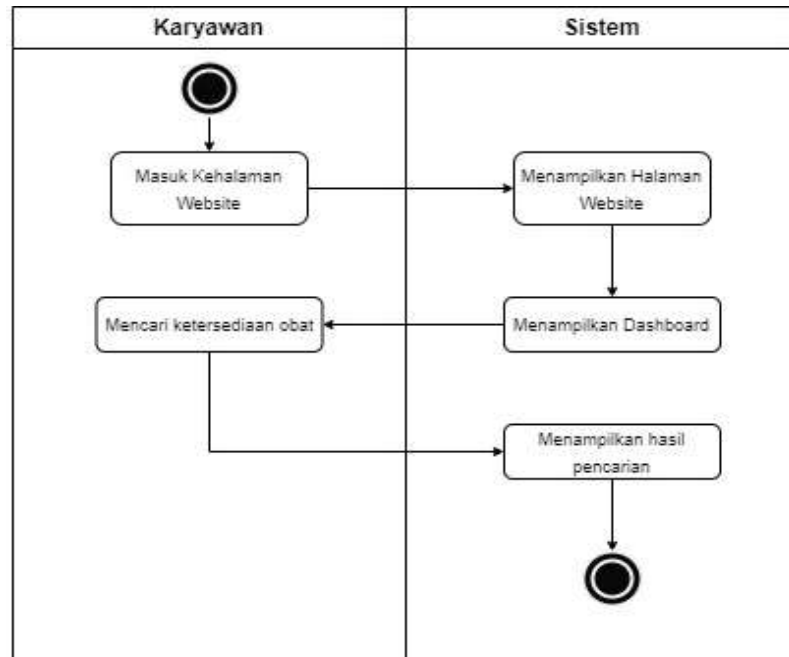


Gambar 5. Activity Diagram Admin

#### 4. Activity diagram

Pada karyawan menggambarkan aliran aktivitas atau aliran kerja yang dilakukan karyawan di dalam sistem mulai dari awal kegiatan sampai kegiatan berakhir. Karyawan akan membuka halaman website, sistem akan menampilkan halaman dashboard, karyawan hanya dapat mengakses tampilan dashboard, dimana karyawan dapat mengetahui stok dan data produk jika menggunakan fitur pencarian tanpa memerlukan login ke sistem website

dan sistem akan langsung menampilkan hasil dari data produk yang dicari. Activity diagram karyawan dapat dilihat pada Gambar dibawah ini.

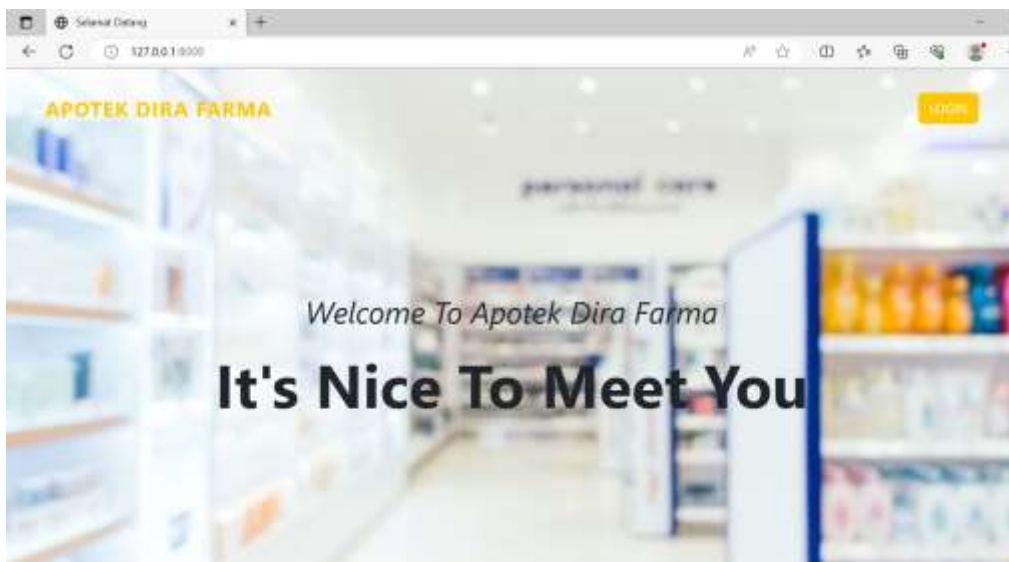


Gambar 6. Activity Diagram Karyawan

#### 4. Implementasi Program

##### 1. Halaman Landing Page

Halaman landing page memiliki beberapa fungsi yaitu, button untuk login sistem, terdapat fitur pencarian untuk mencari produk dan setiap user dapat mengakse halaman landing page.



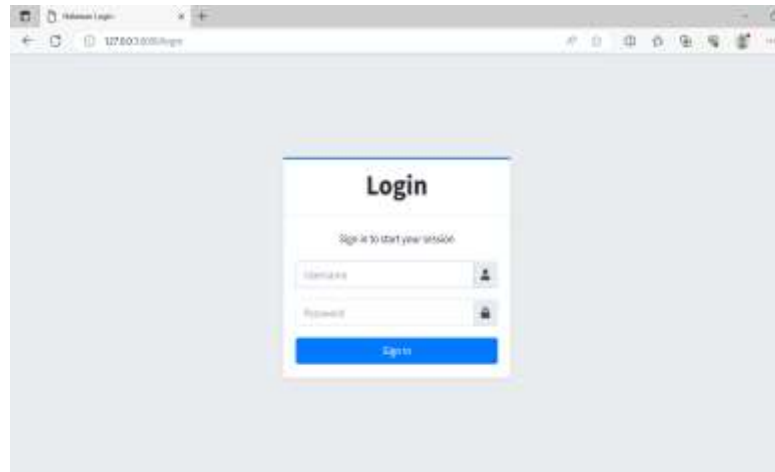
Gambar 6. Halaman Landing Page

##### 2. Menu Admin

Menu admin adalah menu yang hanya dapat diakses oleh pimpinan dan apoteker. Menu ini terdiri dari halaman login, halaman menu kategori, pabrik, produk, supplier, transaksi

penjualan, transaksi pembelian, dan laporan penjualan. Pada menu ini terdapat tombol menu logout yang berfungsi untuk keluar dari menu admin menuju ke halaman *landing page*.

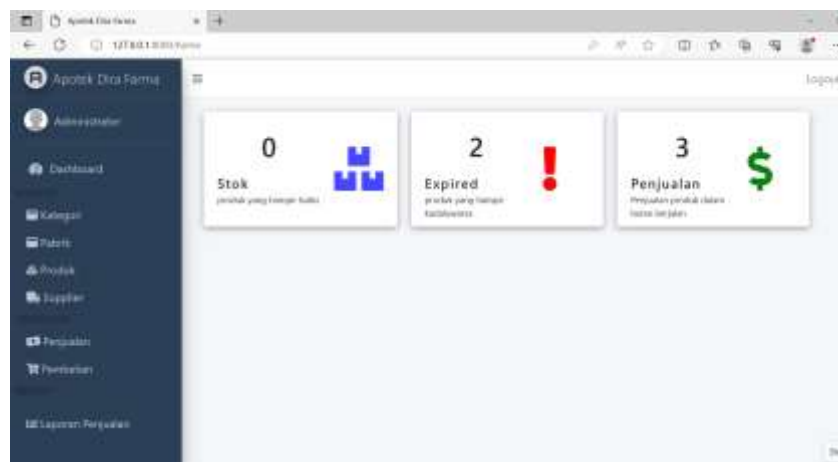
- a. Menu Admin adalah menu yang hanya dapat diakses oleh pimpinan dan apoteker. Menu ini terdiri dari halaman login, halaman menu kategori, pabrik, produk, supplier, transaksi penjualan, transaksi pembelian, dan laporan penjualan. Pada menu ini terdapat tombol menu Logout yang berfungsi untuk keluar dari menu Admin menuju ke halaman *landing page*.



Gambar 7. Halaman login

- b. Tampilan Dashboard Admin

Halaman dashboard pada Gambar 24 merupakan halaman awal yang ditampilkan sistem saat admin berhasil melakukan proses Login. Pada halaman ini terdapat tampilan cepat yaitu data stok produk yang hampir habis, data produk yang hampir kadaluwarsa dan data penjualan produk dalam bulan berjalan.



Gambar 8. Halaman Dashboard

## KESIMPULAN DAN SARAN

### 1. Kesimpulan

Sistem website Apotek Dira Farma menggunakan metode PXP dan arsitektur MVC ini boleh berhasil dikembangkan. Sistem ini mampu mengelola data produk, data penjualan dan pembelian, serta mampu memberikan laporan terkait transaksi.

### 2. Saran

Untuk pengembangan sistem yang akan datang, diharapkan peneliti selanjutnya dapat mengembangkan aplikasi berbasis *website* ini menjadi aplikasi berbasis *mobile* untuk memudahkan pengguna.

## DAFTAR PUSTAKA

- [1] Kemenkes RI. 2017. *Peraturan Menteri Kesehatan Republik Indonesia Nomor 9 Tahun 2017 tentang apotek*. Jakarta: Kementrian Kesehatan Republik Indonesia
- [2] Dzhurov, Y, I. Krasteva, and S. Ilieva, "*Personal Extreme Programming –An Agile Process for Autonomous Developers*," *Faculty of Mathematics and Informatics*, Sofia University, pp. 252-259, 2009
- [3] Kent Beck, (1999) "*Extreme Programming Explained: Embrace Change*", *AddisonWesley Professional*.