

REVERSE ENGINEERING SEBAGAI PENDEKATAN UNTUK MERANCANG ULANG APLIKASI MANAJEMEN PROYEK TEKNOLOGI INFORMASI PADA PT XYZ

Berliyanto

*Program Studi Teknik Informatika, FTI, Institut Teknologi Budi Utomo Jakarta,
berli@itbu.ac.id*

Abstrak

PT XYZ mengandalkan sebuah aplikasi manajemen proyek yang dimilikinya untuk mendukung operasional perusahaan. Aplikasi tersebut berjalan dengan baik secara fungsional namun tidak memiliki dokumentasi sistem karena dibuat secara *ad hoc* dan sporadis. Permasalahan muncul saat ada kebutuhan bisnis baru yang menuntut peningkatan kemampuan aplikasi. Misalnya dalam bentuk penambahan fitur dan perbaikan struktur data. Tanpa adanya dokumentasi aplikasi atau blueprint akan sangat sulit untuk meningkatkan sistem agar memenuhi kebutuhan bisnis yang baru. Terlebih lagi sebagian fitur dan data pada aplikasi saat ini masih perlu dipertahankan karena telah memenuhi sebagian kebutuhan operasional perusahaan. Penelitian ini mengusulkan solusi untuk permasalahan tersebut melalui pendekatan *reverse engineering*. Hasil yang didapat adalah kebutuhan aplikasi yang baru, spesifikasi rancangan aplikasi, dan purwarupa aplikasi yang telah melewati *unit testing* dan *scenario-based testing*.

Kata kunci: *reverse engineering*, perancangan *software*, *legacy system*, *system design*

1. PENDAHULUAN

Ketergantungan terhadap suatu aplikasi yang sudah lama dan usang oleh sebuah perusahaan adalah kondisi yang masih ditemukan saat ini. Salah satu perusahaan yang mengalami kondisi tersebut adalah PT XYZ. Perusahaan ini memiliki sebuah aplikasi utama untuk mendukung kegiatan manajemen proyek teknologi informasi yang dilakukannya sebagai kegiatan operasional harian. Aplikasi tersebut mampu mendukung kegiatan operasional PT XYZ selama ini. Namun demikian, saat ini terdapat beberapa kebutuhan bisnis baru yang menuntut fitur dan fungsionalitas tambahan dari aplikasi tersebut. Hal tersebut menjadi masalah karena aplikasi yang digunakan saat ini tidak memiliki dokumentasi sistem yang menjelaskan spesifikasi rancangannya. Sedangkan spesifikasi rancangan diperlukan untuk dapat meningkatkan fungsionalitas aplikasi tersebut dalam bentuk penambahan fitur atau perbaikan struktur data (Pressman & Maxim, 2020).

Aplikasi seperti yang dimiliki oleh PT XYZ dikenal dengan istilah *system legacy*. Jenis aplikasi ini umumnya memiliki karakteristik antara lain sebagai berikut: menggunakan teknologi lama, tidak memiliki dokumentasi atau *blueprint*, dan tidak

memenuhi kebutuhan bisnis yang akan datang (Dennis, Wixom, & Roth, 2018). *Legacy system* ini disarankan untuk dirancang ulang agar bisa menangani perubahan yang ada, terutama perubahan *requirement*. Proses yang seperti itu dinamakan dengan *reverse engineering*. Dan penelitian ini melakukan *reverse engineering* terhadap aplikasi manajemen proyek milik PT XYZ.

Poin permasalahan yang dirumuskan dalam penelitian ini ada tiga. Pertama PT XYZ masih bergantung pada aplikasi yang lama untuk mendukung proses bisnisnya. Kedua, proses peningkatan fungsionalitas terhadap aplikasi tersebut tidak dapat dilakukan secara langsung dikarenakan tidak adanya dokumentasi yang menjelaskan rancangan aplikasinya. Dan ketiga, terdapat kebutuhan bisnis baru yang belum dipenuhi oleh aplikasi tersebut. Permasalahan-permasalahan tersebut yang melatar belakangi dilakukannya penelitian ini. *Reverse engineering* merupakan sebuah pendekatan untuk mendapatkan rancangan suatu aplikasi yang sudah beroperasi. Secara teori, pendekatan ini dapat membantu permasalahan yang telah diuraikan sebelumnya. Dan berdasarkan permasalahan yang telah disebutkan sebelumnya dirumuskan tiga pertanyaan penelitian sebagai berikut:

1. Apa saja *requirement* baru yang belum terpenuhi oleh sistem yang ada saat ini?
2. Seperti apa rancangan aplikasi yang memenuhi *requirement* baru tersebut?
3. Sejauh mana rancangan aplikasi yang baru dapat diimplementasikan?

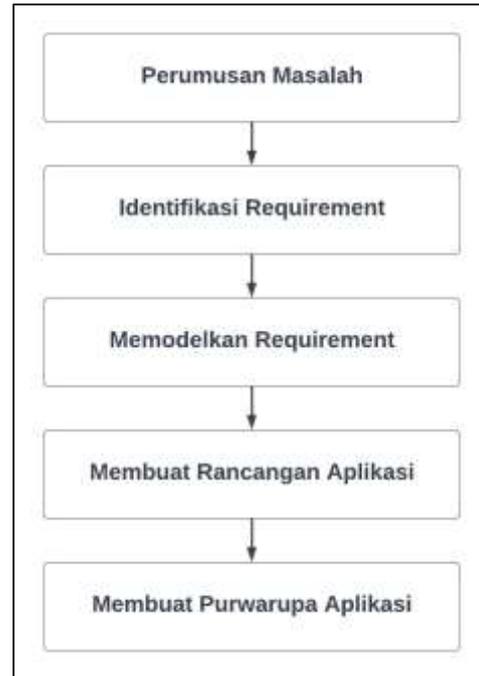
Dengan menjawab tiga pertanyaan penelitian tersebut maka didapatkan rancangan aplikasi baru yang memenuhi kebutuhan bisnis yang baru.

2. METODOLOGI

Penelitian ini merupakan penelitian kualitatif yang mengikuti urutan proses pada siklus hidup perangkat lunak (*software development life cycle*). Metode pengembangan sistem yang dilakukan pada penelitian ini menggunakan salah satu metode *Agile*, yaitu metode pendekatan *Extreme Programming (XP)*. Metode *Extreme Programming (XP)* memiliki empat tahapan, yaitu: *planning, design, coding, dan testing*. Pada penelitian ini, diterapkan proses *reverse engineering*. Gambar 1 menunjukkan langkah-langkah penelitian. Tahapan yang dilakukan pertama kali adalah mengumpulkan *source code* aplikasi sebagai data untuk dilakukan *reverse engineering*. Tujuannya adalah untuk mendapatkan gambaran kondisi aplikasi saat ini dan mengidentifikasi masalah-masalah yang ada di dalamnya.

Proses pengumpulan *requirements* dilakukan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian. Metode pengumpulan *requirements* yang digunakan pada penelitian ini, yaitu metode wawancara dan observasi. Dalam proses pengumpulan *requirements* diterapkan proses *reverse engineering* untuk mendapatkan dokumen kebutuhan pengguna atau bisnis.

Pada tahap pemodelan *requirement*, terdapat tiga jenis model yang dibuat. Pertama adalah model fungsional. Kedua adalah model proses. Dan ketiga adalah model data. Setiap model yang dibuat menggunakan notasi UML yang sudah terstandar.



Gambar 1 Langkah-langkah Penelitian
Sumber: Penelitian Mandiri, 2023

Pada tahap pembuatan rancangan aplikasi, rancangan antarmuka dibuat berdasarkan *requirements* yang sudah terkumpul dan berdasarkan rancangan antarmuka aplikasi saat ini. Selain membuat rancangan antarmuka, pada tahap ini juga membuat rancangan database yang dikembangkan dari database *existing* melalui proses *reverse engineering*. Tools yang digunakan adalah *Balsamic Wireframes*, dan *proto io* untuk membuat rancangan antarmuka.

Tahap terakhir adalah membuat purwarupa. Purwarupa adalah cara efektif untuk memastikan *requirement* yang dikumpulkan sudah valid atau belum (Sharp, Roger, & Preece, 2019). Pada tahap ini mulai dilakukan pengembangan purwarupa aplikasi berdasarkan *requirement*, dan *design model* yang telah didapat dan dibuat. Pembuatan purwarupa dilakukan oleh dua orang programmer secara *pair programming*. Teknik pemrograman secara *pair programming* disarankan apabila metode yang digunakan untuk mengembangkan aplikasinya adalah *Extreme Programming* seperti yang dilakukan dalam penelitian ini. Pada bagian akhir penelitian ini, purwarupa yang dihasilkan diuji fungsionalitasnya dengan teknik *black-box testing*.

3. HASIL DAN PEMBAHASAN

Setiap langkah penelitian yang dilakukan menghasilkan luaran-luaran yang merupakan hasil penelitian ini.

3.1 Kebutuhan Bisnis dan *User Story*

Kebutuhan bisnis (*business requirements*) dalam penelitian ini didapatkan dari wawancara dan observasi. Berdasarkan hasil pengumpulan kebutuhan bisnis tersebut, kemudian dibuatkan daftar kebutuhan bisnis dalam bentuk Tabel 1. Total ada 7 kebutuhan bisnis yang berhasil diidentifikasi.

Tabel 1 Daftar Kebutuhan Bisnis

Kode	Kebutuhan Bisnis
[BR01]	Divisi TI perlu mengelola <i>document number</i>
[BR02]	Divisi TI perlu melakukan monitoring pembayaran <i>document project</i>
[BR03]	Divisi TI perlu melakukan monitoring pembayaran <i>document maintenance</i> aplikasi
[BR04]	Divisi TI perlu melakukan monitoring pemakaian <i>mandays</i>
[BR05]	Divisi TI perlu mengelola dokumen <i>project, maintenance, dan mandays</i>
[BR06]	Divisi TI perlu mengelola data aplikasi yang dimiliki
[BR07]	Divisi TI perlu mendokumentasikan <i>document</i> yang sudah <i>full signed</i>

Sumber: Penelitian Mandiri, 2023

Dari tujuh kebutuhan bisnis yang ada, hanya tiga kebutuhan yang sepenuhnya telah terpenuhi oleh aplikasi yang digunakan saat ini. Tiga kebutuhan yang sudah terpenuhi sepenuhnya adalah [BR02], [BR03], dan [BR06]. Sementara itu, kebutuhan [BR01], [BR04], dan [BR05] baru dipenuhi sebagian oleh aplikasi tersebut. Sedangkan kebutuhan bisnis yang sama sekali belum dipenuhi adalah [BR07].

Kebutuhan bisnis yang disajikan pada tabel 1 secara tidak langsung menunjukkan peran penting dari aplikasi yang ada saat ini. Pertimbangan untuk tidak meninggalkan aplikasi tersebut dan membuat aplikasi baru merupakan keputusan yang wajar. Hal tersebut dikarenakan sebagian kebutuhan bisnis yang dimiliki oleh PT XYZ sudah terpenuhi. Ditambah lagi tiga kebutuhan bisnis lainnya sudah terpenuhi walaupun baru sebagian. Hanya ada satu kebutuhan bisnis

baru yang belum dipenuhi dengan aplikasi saat ini.

Dari kebutuhan bisnis kemudian disusun *user story* untuk memodelkan fungsionalitas dari aplikasi tersebut. Tabel 2 menunjukkan empat contoh *user stories* yang dihasilkan. Total ada sebanyak 42 *user stories* untuk rancangan aplikasi yang baru. Setiap *user story* yang ada memiliki keterkaitan dengan kebutuhan bisnis. Misalnya [US01] dan [US02] berkaitan dengan kebutuhan bisnis [BR01]. Sedangkan [US03] berkaitan dengan kebutuhan bisnis [BR07].

Tabel 2 Daftar *User Story*
Sumber: Penelitian Mandiri, 2023

Kode	User Story
[US01]	Sebagai <i>user</i> saya dapat membuat nomor dokumen untuk kebutuhan penomoran dokumen Divisi Teknologi Informasi.
[US02]	Sebagai user saya dapat mengubah deskripsi keterangan pada penomoran dokumen untuk kebutuhan apabila terdapat perubahan deskripsi.
[US03]	Sebagai user saya dapat melakukan upload dokumen yang sudah <i>full signed</i> ke dalam aplikasi untuk kebutuhan dokumentasi dokumen.
....
[US42]	Sebagai <i>user</i> saya dapat menerima notifikasi via email terkait data <i>maintennace</i> yang masa berakhirnya akan habis sehingga saya akan lebih aware

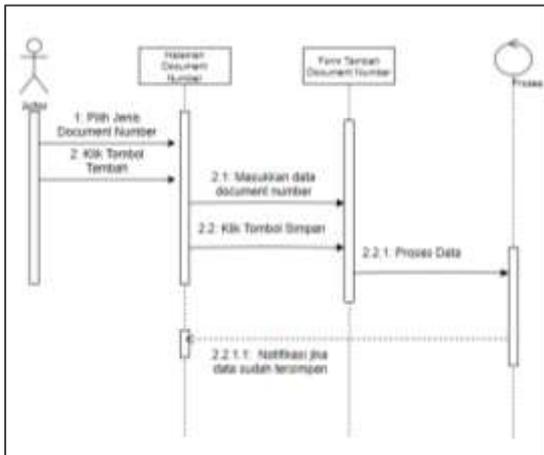
Sumber: Penelitian Mandiri, 2023

Proses validasi telah dilakukan untuk setiap *user story* yang didapatkan. Proses validasi dilakukan oleh tiga orang pengguna aplikasi. Cara melakukan validasinya adalah dengan meminta tiga orang pengguna tersebut membaca semua *user story* yang ada, lalu menanyakan apakah *user story* yang didapatkan relevan dengan kebutuhan mereka saat ini? *User story* dinyatakan valid apabila minimal ada dua pengguna yang menyatakan "iya" untuk pertanyaan tersebut. Dan semua *user story* yang didapatkan dinyatakan valid.

3.2 Model Proses Aplikasi

Pada penelitian ini, model proses dibuat dalam bentuk *sequence diagram* untuk menggambarkan interaksi antar objek di

dalam dan di sekitar sistem. Salah satu contoh model proses yang dibuat adalah proses menambah *document number*. Sequence diagram adalah salah satu teknik yang disarankan untuk digunakan ketika melakukan *reverse engineering* (Raharjana & Justitia, 2015). Gambar 2 menunjukkan *sequence diagram* tersebut.



Gambar 2 Sequence Diagram untuk Proses Menambah *Document Number*
Sumber: Penelitian Mandiri, 2023

Pembuatan nomor dokumen yang sudah berjalan saat ini baru hanya berupa formulir migrasi data. Hal tersebut dikarenakan fungsionalitas yang terbatas pada aplikasi yang berjalan saat ini. Proses pembuatan nomor dokumen dapat dilakukan oleh seluruh staff terkait, artinya seluruh staff tersebut merupakan pengguna aplikasi pada penelitian ini. Pengguna dapat membuat nomor dokumen dengan memilih dokumen yang dibutuhkan terlebih dahulu. Kemudian, pengguna mengisi informasi sesuai informasi yang ada pada sistem. Informasi yang sudah diisi disimpan ke dalam basis data. Pengguna dapat melihat langsung nomor dokumen yang dibuatkan oleh sistem berdasarkan inputan sebelumnya. Gambar 2 menunjukkan *sequence diagram* untuk proses tersebut dan hanya salah satu contoh dari model untuk proses yang dibuat dalam penelitian ini.

3.3 Rancangan Antarmuka dan Arsitektur

Rancangan dibuat pada penelitian ini adalah rancangan arsitektur dan rancangan antarmuka aplikasi. Rancangan antarmuka sistem dibuat untuk menjelaskan tampilan ketika pengguna berinteraksi dengan aplikasi ini. Rancangan antarmuka yang dibuat ini dapat menjadi panduan untuk programmer

ketika mengembangkan aplikasi sejenis. Rancangan antarmuka dibuat dalam bentuk mockup yang bisa disesuaikan dengan kebutuhan unik organisasi atau perusahaan. Gambar 3 menunjukkan rancangan antarmuka dashboard. Setiap rancangan antarmuka yang dibuat memperhatikan *Shneiderman 8 golden rules* (Shneiderman et al, 2016).



Gambar 3 Rancangan Antarmuka Dashboard
Sumber: Penelitian Mandiri, 2023

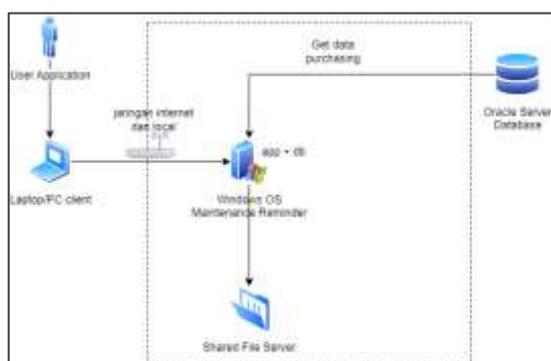
Salah satu halaman yang penting untuk aplikasi ini adalah halaman pengelolaan *document number*. Gambar 4 menunjukkan rancangan antarmuka untuk halaman tersebut. Sisi kiri antarmuka diisi dengan baris-baris menu utama. Dan 70% lebar di sisi kanan digunakan untuk konten dinamis. Disebut dinamis karena isinya akan menyesuaikan dengan halaman yang sedang diakses. Misalnya, saat mengakses halaman dashboard maka konten yang ditampilkan adalah ringkasan dari proyek TI yang sedang dilakukan (seperti pada gambar 3). Sedangkan ketika mengakses halaman pengelolaan *document number*, maka yang ditampilkan adalah informasi terkait dokumen proyek baik berupa tabel ringkasan, fitur pencarian, ataupun menu-menu lain yang terkait seperti yang ditampilkan pada Gambar 4.

Selain rancangan antarmuka, penelitian ini juga menghasilkan rancangan arsitektur aplikasi. Gambar 5 menunjukkan rancangan arsitektur tersebut. Rancangan arsitektur dibuat berdasarkan kebutuhan PT XYZ, namun demikian dapat digunakan sebagai acuan oleh organisasi lain. Aplikasi dirancang sebagai aplikasi berbasis website. Secara teori, jenis arsitektur yang sesuai adalah arsitektur client-server. Rancangan pada gambar 5 merupakan rancangan arsitektur client-server yang bisa memenuhi kebutuhan implementasi aplikasi ini dalam konteks penggunaan oleh PT XYZ.



Gambar 4 Rancangan Antarmuka Halaman Pengelolaan *Document Number*
Sumber: Penelitian Mandiri, 2023

Aplikasi yang berbentuk *webapp* ini membutuhkan *web browser* untuk dapat digunakan oleh pengguna. Rancangan yang dibuat menyarankan agar pengembang aplikasi memprioritaskan *browser* populer seperti *Google Chrome*, *Microsoft Edge*, dan *Mozilla Firefox*. Ketiganya merupakan *web browser* yang paling banyak digunakan oleh pengguna internet saat ini.



Gambar 5 Rancangan Arsitektur Aplikasi
Sumber: Penelitian Mandiri, 2023

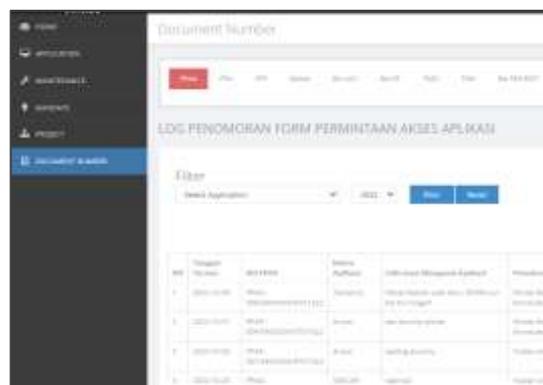
Spesifikasi untuk server disarankan untuk mengikuti apa yang sudah dimiliki oleh organisasi. Dalam penelitian ini server menggunakan sistem operasi *Windows Server 2016 standard 64 BIT*. Perangkat server yang digunakan memiliki processor *Intel Xeon E5-2620* dan *8GB DDR4 RAM*. Dengan spesifikasi tersebut, aplikasi ini diharapkan dapat menangani kebutuhan 100 pengguna yang mengakses aplikasi ini secara bersamaan. Kebutuhan tersebut sudah lebih dari cukup untuk *PT XYZ*.

Dari sisi server, arsitektur ini menggunakan tiga jenis server dengan fungsinya masing-masing. Ketiga server tersebut adalah *web server*, *database server*,

dan *file server*. *Web server* berperan untuk menjalankan aplikasi utama yang berbentuk *webapp*. Sedangkan *database server* menjalankan *database management system (DBMS)* yang menyimpan data. Sementara itu *file server* berfungsi untuk menyimpan dokumen-dokumen proyek yang diunggah ke dalam aplikasi. Pemisahan tiga server ini juga dilakukan untuk mengurangi risiko kegagalan aplikasi.

3.4 Purwarupa Aplikasi

Pengembangan purwarupa dilakukan pada tahap akhir penelitian ini. Tujuannya adalah untuk menguji apakah rancangan dan model kebutuhan yang dihasilkan dari penelitian ini dapat diimplementasikan menjadi aplikasi yang berjalan. Oleh karena itu lah pengembangan aplikasi dilakukan berdasarkan spesifikasi kebutuhan dan rancangan yang telah diuraikan pada bagian sebelumnya.



Gambar 6 Tampilan Salah Satu Halaman pada Purwarupa Aplikasi
Sumber: Penelitian Mandiri, 2023

Purwarupa dikembangkan menggunakan bahasa pemrograman *PHP* dengan *framework laravel*. Sementara itu bagian *frontend* aplikasi dibuat dengan *HTML* dan *CSS*. Programmer yang terlibat sejumlah dua orang. Pemrograman dilakukan dengan cara *pair programming*.

Purwarupa aplikasi dikembangkan dalam bentuk aplikasi berbasis *website*. Bahasa pemrograman yang digunakan untuk *backend* aplikasi adalah *PHP*. Sedangkan *frontend* aplikasi dikembangkan dengan *HTML*, *CSS*, dan *Javascript*. *Database Management System (DBMS)* yang digunakan oleh purwarupa ini adalah *MySQL*. Purwarupa ini dikembangkan oleh dua orang *programmer* secara *pair programming*.

Pengujian terhadap aplikasi dilakukan setelah purwarupa berhasil dikembangkan. Ada dua pengujian yang dilakukan yaitu *unit testing* dan *integration testing*. Unit testing dilakukan untuk memastikan setiap fungsi yang ada di dalam aplikasi dapat berjalan sebagaimana mestinya. Teknik yang digunakan untuk unit testing ini adalah *black-box testing* dan dilakukan oleh programmer yang memahami kode program aplikasinya. Sementara itu, *integration testing* dilakukan dengan *scenario-based testing*. Instrumen pengujian yang digunakan adalah *test case* yang dibuat berdasarkan *user story*. Hasil pengujian unit testing dan *integration testing* menunjukkan bahwa purwarupa yang dikembangkan dari spesifikasi rancangan yang diusulkan dapat berfungsi sesuai harapan.

4. KESIMPULAN

Kesimpulan penelitian ini dirumuskan setelah semua langkah dilakukan dan penelitian selesai. Tahap perumusan kesimpulan ini merupakan langkah terakhir yang dilakukan dalam penelitian ini. Secara umum, poin-poin dari kesimpulan menjawab setiap pertanyaan penelitian yang ada.

Penelitian ini berhasil mengidentifikasi sebanyak 7 (tujuh) kebutuhan bisnis. Empat di antaranya adalah kebutuhan yang belum dipenuhi oleh aplikasi yang ada saat ini. Dari tujuh kebutuhan bisnis tersebut kemudian diidentifikasi sebanyak 42 *user stories*. *User story* yang dihasilkan tersebut sekaligus menjadi model fungsional untuk aplikasi yang baru. Sementara itu model proses dengan *sequence diagram* juga dibuat sebagai pelengkap dari model fungsional tersebut. Tujuh kebutuhan bisnis dan 42 *user stories* beserta dengan model-model prosesnya merupakan jawaban untuk pertanyaan penelitian yang pertama.

Penelitian ini juga mengusulkan rancangan antarmuka dan rancangan arsitektur yang dapat memenuhi setiap kebutuhan yang diidentifikasi sebelumnya. Rancangan antarmuka dalam penelitian ini dibuat dalam

bentuk *mockup* sehingga dapat disesuaikan dengan kebutuhan spesifik organisasi. Sementara itu, rancangan arsitektur aplikasi dibuat dalam bentuk *client-server architecture*. Bentuk aplikasi yang diusulkan adalah aplikasi berbasis website. Rancangan antarmuka dan rancangan arsitektur tersebut merupakan jawaban untuk pertanyaan penelitian yang kedua.

Pada tahap terakhir penelitian ini dikembangkan sebuah purwarupa aplikasi. Purwarupa tersebut dikembangkan untuk menguji sejauh mana rancangan yang dihasilkan dapat diimplementasikan sebagai sebuah aplikasi yang berjalan. Hasil pengembangan purwarupa adalah sebuah aplikasi berbasis website yang secara fungsional sudah dapat berjalan dengan baik. Hal tersebut dibuktikan dengan hasil pengujian *unit testing* dan *scenario-based testing*. Purwarupa dan hasil pengujian aplikasi tersebut adalah jawaban untuk pertanyaan penelitian yang ketiga.

5. DAFTAR PUSTAKA

- Dennis, A., Wixom, B.H., Roth, R.M. (2018). *System analysis and design* (7th edition). Wiley
- Pressman, R.S., & Maxim, B.R. (2020). *Software Engineering A Practitioner's Approach Ninth Edition*. McGraw-Hill Education.
- Raharjana, I.K., & Justitia A. (2015). *Pembuatan Model Sequence Diagram dengan Reverse Engineering Aplikasi Basis Data pada Smartphone untuk Menjaga Konsistensi Desain Perangkat Lunak*. JUTI: Jurnal Ilmiah Teknologi Informasi.
- Sharp H, Rogers Y, & Preece J. (2019). *Interaction Design Beyond Human-Computer Interaction 5th Edition*. Wiley. 2019, 385-417
- Shneiderman B, Plaisant C, Cohen M, Jacobs S, Elmqvist N, & Diakopoulos N. (2016). *Designing the User Interface: Strategies for Effective Human-Computer Interaction 6th Edition*. Pearson.