Formosa Journal of Science and Technology (FJST)
Vol.2, No.11, 2023: 3017-3034

Development of Electronic Registry Operating Record of Shoulder and Elbow

Cholid Fauzi[1*], Mochamad Nauval Ghassani[2], Chandriko Audito Rengganisworo[3], Silvia Nur Fadillah[4]
Politeknik Negeri Bandung
Corresponding Author: Cholid Fauzi cholid.fauzi@polban.ac.id

A R T I C L E I N F O

Keywords: Electronic Registry, E-Registry, Shoulders and Elbow Registry, Operating Record, Reverse Engineering

Received : 10, September
Revised : 12, October
Accepted: 15, November

A B S T R A C T

The register book is used by the doctor to keep a record of the disease that finds and what action should be taken. This research goal is to develop e-registry web application has been created for registering shoulder and elbow surgery based on existing web based application. According to the problems mentioned, it is proposed to apply an e-registry shoulder and elbow surgery mobile-based using Flutter technology. The mobile application will be connected to the e-registry web application using the API concept. This application development uses the Incremental method with sequential development activities in two iterations. The research focused on developing mobile-based using reverse engineering method, and the finding of this research, using reverse engineering is more practical in its use. The type of test used to test the result of this result which is the application is unit testing with a black box approach which serves to determine the suitability of the input and output produced.

INTRODUCTION

Computers and information technology have become more and more prevalent in many industries, including healthcare. The most important thing that any human being must take care of is their health (BPS-BKKBN, 2008). One of the things that must be thought about is maintaining one's health because it will impact both one's own and other people's performance and activities. Due to the many activities, they engage in today, many people consistently disregard their own health (Roy et al., 2020). As a result, there are currently numerous patients seeking treatment in hospitals, clinics, and health facilities for a wide range of ailments (Venkateswaran et al., 2022). A web-based electronic register application has been created for recording registers for shoulder and elbow surgery. This application handles shoulder and elbow surgery/surgery registers in three hospitals, namely: (1) Santosa Central Hospital; (2) RSKB Halmahera Alert; and (3) Dr. RSUP. Hasan Sadikin. However, clients still need a mobile-based application because the e-registry web display does not match the mobile display when accessed via a mobile device (Venkateswaran et al., 2018).

Apart from that, mobile devices are also used more frequently and are more practical to use, so it is necessary to develop a mobile-based shoulder and elbow surgery register application. The functionality of a mobile-based application will resemble the functionality of an e-registry web application [4], so it is necessary to understand the e-registry web application (Abdillah & Ihksan, 2021). It is hoped that the development of a mobile-based application can help in the implementation of recording by users in the future (Pagani, 2004). The problem formulation obtained is that users need a mobile-based electronic registration book application for shoulder and elbow surgery. The following are the points that we will analyze to develop an electronic register book application:

1. Is the suitability of the business process in the web application what is needed?
2. Are there any changes related to business rules in the application to be built.
3. Suitability of the form or available data according to needs.
4. Are there any additional roles for the application to be developed?
5. What features will be developed in the mobile application.

It is necessary to reverse engineer the e-registry web application to develop a mobile-based electronic registration book application for shoulder and elbow surgery. The type of test used to test the result of this result, which is the application is unit testing with a black box approach which serves to determine the suitability of the input and output produced.

THEORETICAL REVIEW

The most important thing that any human being must take care of is their health (BPS-BKKBN, 2008). One of the things that must be thought about is maintaining one's health because it will impact both one's own and other people's performance and activities. Due to the many activities, they engage in today, many people consistently disregard their own health (Roy et al., 2020).

As a result, there are currently numerous patients seeking treatment in hospitals, clinics, and health facilities for a wide range of ailments (Venkateswaran et al., 2022). A web-based electronic register application has been created for recording registers for shoulder and elbow surgery. This application handles shoulder and elbow surgery/surgery registers in three hospitals, namely: (1) Santosa Central Hospital; (2) RSKB Halmahera Alert; and (3) Dr. RSUP. Hasan Sadikin. However, clients still need a mobile-based application because the e-registry web display does not match the mobile display when accessed via a mobile device(Venkateswaran et al., 2018).

METHODOLOGY

Development of this research is using reverse engineering is an activity commonly called abstraction extraction. Reverse engineering evaluates old programs from their (often undocumented) source code and derives the specifications of the processes carried out, user interface applied, and the data structure of the program or database used(Larman, 2005; Pressman, 2015). The data collected and extracted from the extract abstraction process of reverse engineering. This is divided into several stages as explained in Figure II.1.

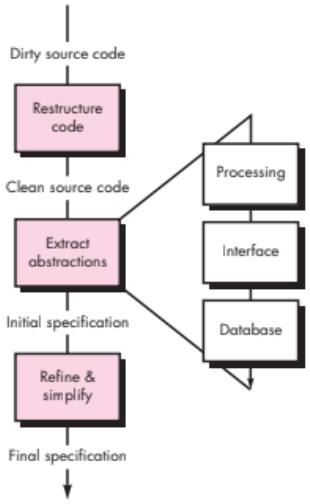

Figure 1. Stages of Reverse Engineering (Source: Pressman, 2010)

1. Restructure Code

The first level on reverse engineering is restructuring code (Albani et al., 2016)which is what is done at this stage clean code or restructuring the program so that the source code (source code) is easier to read and as a basis for activities reverse engineering further.

2. Extract Abstractions

This stage carries out abstraction extraction, which is divided into three parts, namely:

a. Processing

Each program that makes up an application system represents a functional abstraction at a high level of detail. At this stage, the

specifications of the source code. Block diagrams represent the interactions between these functional abstractions.

b. Interface
   Appearance user interface replacement may not reflect the interface of the old one or may be very different. This analysis can be used to change the image scale or add/subtract buttons on the user interface based on user needs.

c. Database
   Reverse engineering databases require an understanding of existing objects and their relationships. The database defines data objects and supports several methods for establishing relationships between objects.

3. Refine & Simplify

This level is a level final from reverse engineering which contains a simplification of the specifications of the system being analyzed to produce final specifications in software development activities and based on the goals of requirement why need to reverse engineering, the stages carried out are reverse engineering and application development(Horkoff et al., 2016). The development of an electronic register book application for shoulder and elbow-based surgery mobile has the stages or implementation flow described in Figure 2.

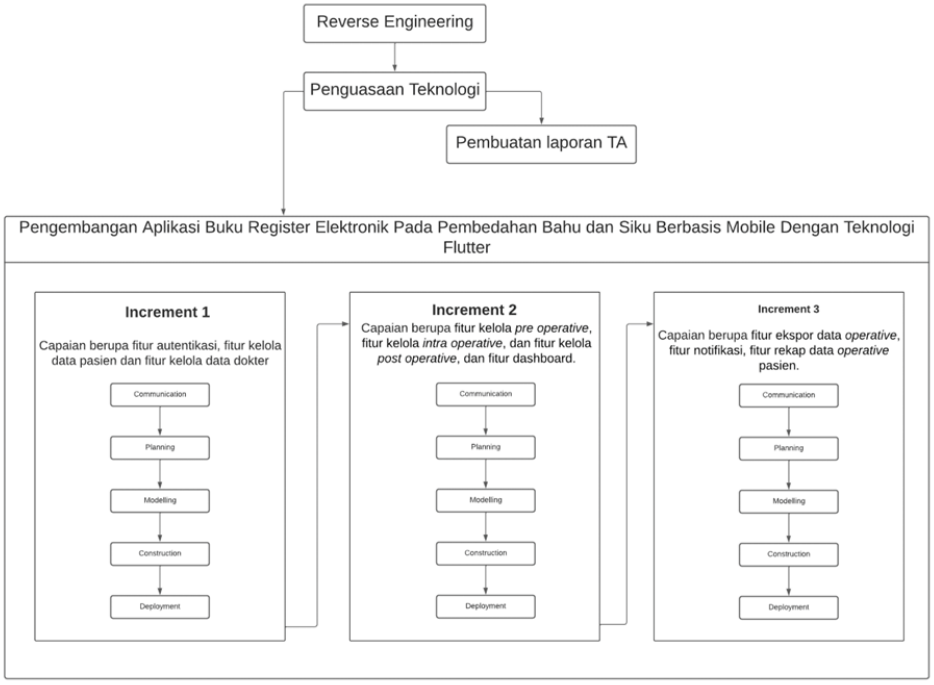

Figure 2. Implementation Flow

In each interation, a feature is determined to be achieved (Larman, 2005; Leff & Rayfield, 2001). Achievement of each feature increment on the development of electronic register books is explained in table below.

Table 1. Incremental Application Development

No	Increment	Stage Name	Activity	Artifacts
1	Increment 1	Analysis	1. Planning work on increment 2. Perform analysis stakeholder, analysis business process, analysis business rules, analysis user interface and data structure analysis based on authentication features, patient data management features and doctor data management features that will be built.	SRS Version 1 Document
		Design	Create class designs, database designs, interaction designs between objects and UI designs based on the results of the analysis that has been carried out.	
		Code	Implementation of authentication features, patient data management features and doctor data management features.	Product Increment 1
		Test	Testing features that have been implemented	Version 1 testing document
	Increment 2	Analysis	1. Planning work on increment 2. 2. Perform analysis stakeholder, analysis business process, analysis business rules, analysis user interface and data structure analysis based on management features pre-operative, management features intra operative, management features post-operative, the dashboard feature and the	SRS Version 2 Document

			supervising doctor feature validate the operative data that will be built	
		Design	Create class designs, database designs, interaction designs between objects and UI designs based on the results of the analysis that has been carried out.	
		Code	1. Fixing minor error / minor improvement from increment 1 2. Implementation of manage features pre-operative, manage feature intra operative, manage feature post-operative, dashboard features and supervising doctor features validate operative data	Product Increment 2
		Test	Testing features that have been implemented	Version 2 and UAT testing documents

.

RESULTS AND DISCUSSION

Using the method reverse engineering and analyze for the system to be built. From this analysis, conclusions are drawn which will be taken into consideration in designing the application to be developed (*Sample Unified Process Artifacts and Timing (s-Start; r-Refine)*, n.d.). The application to be developed is based on mobile (android) up to restructure code What is done is restructuring the code inward platform android. Restructuring is carried out into programming languages and framework which is different from web applications e-registry, namely inward framework Flutter with the Dart programming language. Explains the results of extracting abstractions on the web e-registry to obtain and understand the processes in the application (Venkateswaran et al., 2022). The results of this stage will then be communicated to the client for confirmation. Stages of abstraction extraction covers processing, interface, and database based on the features contained in the web application e-registry. The features that the abstraction will extract in the web application e-registry explained in table below.

Table 2. E-Registry Web Application Features

No	Feature	Description
1	Login	This feature functions to access applications according to role
2	Dashboard	This feature contains information about the overview operative every month, total operative (in the current month), total patients, total operative which have been validated, total operative which have not been validated and accumulated from Pre-Operative, Intra Operative, and Post-Operative.
3	Surgical Recording Calendar	This feature contains surgical schedule records based on data planned date on operative.
4	Manage Patient Data	This feature handles patient data management which includes adding, deleting, updating, searching, and displaying details of patient data.
5	Manage Doctor Data	This feature handles doctor data management which includes adding, deleting, and updating searches.
6	Manage Data Hospital	This feature handles hospital data management handled by doctors which includes additions, deletions and updates.
7	Manage Data Pre-Operative	This feature handles data management pre-operative which includes adding, deleting, updating, searching and displaying details of data pre-operative.
8	Manage Data Intra Operative	This feature handles data management intra operative which includes adding, deleting, updating, searching, and displaying details of data intraoperative.
9	Manage Data Post Operative	This feature handles data management post-operative which includes adding, deleting, updating, searching and displaying details of data post-operative.
10	Doctor Validates Data Operative	Supervising Doctor/Specialist will validate the data operative which has been input by the boarding doctor or resident doctor. Data is said to be valid when the data entered is in accordance with the physical evidence held by the Supervising Doctor/Specialist.
11	Export Data Recap Operative	This feature functions to download recaps operative into file in excel form. The entire data can be exported operative or just one of

		the operative. This feature can only be accessed by Supervising Doctors/Specialists.
12	Data Export Feature Operative Patient	This feature functions to download data operative into file in pdf format based on selected patients.

Processing

This sub-chapter explains the results of extracting abstractions to derive specifications for the covering applications business process, business rules, and stakeholders. The results of this stage will then be confirmed with the client for confirmation.

Stakeholder

Sub-chapter Analysis Stakeholder Explain about stakeholders or parties related to the web application e-registry. Analysis stages stakeholder this is done to find out the role of each stakeholder in the application. Stakeholder and its role in web applications e-registry are explained in table 3.

Table 3. Stakeholders Role

No.	Stakeholder	Role
1	Specialist/Supervising Doctor	Specialist/Supervising Doctors are application users with rights/access to all features in the application. Only Specialist/Supervising Doctors can validate all data operative that has been input.
2	Resident Doctor	Resident Doctors are application users with limited rights/access. The feature that cannot be used by Resident Doctors is the data validation feature operative and management features (adding, changing and deleting) doctor data.
3	Doctor Koas	Koas Doctors are application users with the same rights/access as Resident Doctors. In practice, Doctor Koas usually inputs patient data or data operative on the shoulder and elbow surgery register book.

Design of the application to be developed. The design was made considering the analysis in research problems.

Application Architecture Design

This sub-chapter explains the application architecture design. Application architecture is represented using package diagram. Making this package diagram refers to the domain model analysis in subsections IV.2.5 and design pattern used in application development is MVC. The application architecture design is explained in Figure below

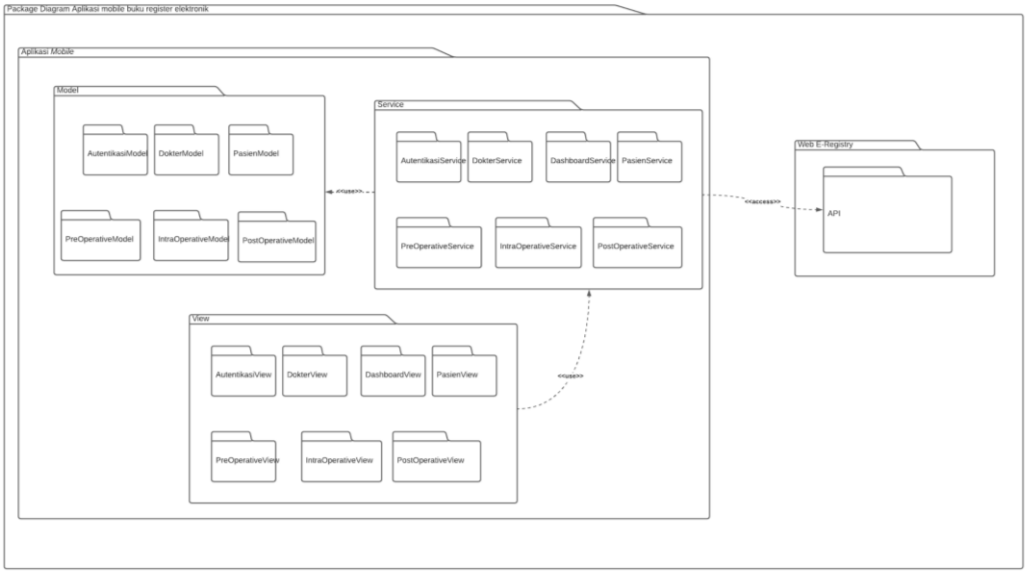

Figure 3. Package Diagram

Table 4. Package Diagram

No	Class	Type	Description
1	Abduction Degree Image Model	Model	A class that functions to accommodate information from the image abduction degree table
2	Domain Case Model	Model	A class that functions to accommodate information from the domain case model table
3	External Rotation 90 Abduction Image Model	Model	A class that functions to accommodate information from the external rotation 90 abduction image table
4	External Rotation Neutral Image	Model	A class that functions to accommodate information from the external rotation neutral image table
5	Forward Flexion Image Model	Model	Class that functions to accommodate information from the forward flexion image table
6	Hospital Model	Model	A class that functions to hold information from the hospital table
7	Internal Rotation Image Model	Model	A class that functions to hold information from the image rotation image table
8	Intra-Operative Model	Model	Class that functions to accommodate information from intra-operative tables
9	Open Reduction	Model	Class that functions to

	and Trauma Procedures Model		accommodate information from the open reduction and trauma procedures table
10	Pain Intervention Procedures Model	Model	Class that functions to accommodate information from the pain intervention procedures table
11	Patient Model	Model	Class that functions to hold information from the patient table
12	Post-Operative Model	Model	Class that functions to accommodate information from the postoperative table
13	Pre-Operative Model	Model	Class that functions to accommodate information from the preoperative table
14	Registration Model	Model	Class that functions to hold information from the registration model table
15	Shoulder Arthoplasty Procedures Model	Model	A class that functions to accommodate information from the shoulder arthroplasty procedures table
16	Shoulder Arthroscopy Procedures Model	Model	Class that functions to accommodate information from the shoulder arthroscopy procedures table
17	Shoulder Special Test Model	Model	Class that functions to accommodate information from the User shoulder special test table
18	User Model	Model	Class that functions to hold information from the User (Doctor) table
19	API	Data Provider	Classes that perform data retrieval functions and provide data from the API
20	Login Service	Service	Class that functions as a controller on the Login page
21	Doctor Service	Service	Class that functions as a controller on the Doctor menu page
22	Patient Service	Service	Class that functions as a controller on the Patient menu page
23	Pre-Operative	Service	Class that functions as a

	Service		controller on the pre-operative menu page
24	Intra-Operative Service	Service	Class that functions as a controller on the Intra Operative menu page
25	Post-Operative Service	Service	Class that functions as a controller on the post-operative menu page
26	Dashboard Service	Service	Class that functions as a controller on the Dashboard menu page
27	Login Screen	View	Class that functions to display the UI on the Login page
28	Doctor Screen	View	Class that functions to display the UI on the Doctor page
29	Patient Screen	View	Class that functions to display the UI on the Doctor page
30	Pre-Operative Screen	View	Class that functions to display the UI on the Pre-Operative page
31	Intra-Operative Screen	View	Class that functions to display UI on Intra Operative pages
32	Post-Operative Screen	View	Class that functions to display the UI on the Post-Operative page
33	Dashboard Screen	View	A class that functions to display the UI on the Dashboard page
34	Main Screen	View	The class that works to display the UI is chosen by the user
35	Shared Preference	Library	A library that functions to store data of primitive types (string, integer, Boolean) which will be stored in local data.

Class planning refers to the model domain found in sub-chapter IV.2.5, where class diagram developed from domain model the Class diagram used to describe classes, interface, and its associations(Larman, 2005; Pressman, 2015).

Implementation

Implementation Linkage Matrix

This table explains the sequence of implementation that has been carried out based on the results of the analysis and design results which will be shown in table below.

Table 5. Implementation Linkage Matrix

ID Requirement	Requirement	Planning			Deliverables
		Behavior	Process	User Interface	
REQ-F-01	The application can identify and confirm users who enter the system so they can perform functionality according to their role.	UC-01	SD-01	IM-01	Implemented
REQ-F-02	The application can display information regarding overview operative every month, total operative (in the current month), total patients, total operative which have been validated, total operative which have not been validated and accumulated from Pre-Operative, Intra Operative, and Post-Operative.	UC-25	SD-22	IM-20	Implemented
REQ-F-03	The application can save additional patient data database added by the doctor	UC-02	SD-02	IM-04	Implemented
REQ-F-04	The application can display all registered patient data	UC-02, UC-04, UC-05	SD-02, SD-03, SD-05	IM-03	Implemented
REQ-F-05	The application can handle changes to patient data and save these changes database.	UC-04	SD-03	IM-05	Implemented
REQ-F-06	The application can handle deletion of patient data	UC-03	SD-04	-	Implemented
REQ-F-07	The application can display detailed patient data as selected	UC-05	SD-05	IM-06	Implemented
REQ-F-08	The application can display patient data based on the keywords entered.	UC-26	SD-26	-	Implemented
REQ-F-09	The application can display all registered doctor data	UC-06, UC-08, UC-09	SD-06, SD-07, SD-09	IM-07	Implemented
REQ-F-10	The application can save data on adding doctors to database added by the Specialist/Supervising Doctor	UC-06	SD-06	IM-08	Implemented
REQ-F-	The application can handle	UC-08	SD-07	IM-09	Implement

011	changes to doctor's data and save these changes database.				ed
REQ-F-12	The application can handle the deletion of doctor data.	UC-07	SD-08	-	Implemented
REQ-F-13	The application can display detailed doctor data as selected.	UC-09	SD-09	IM-10	Implemented
REQ-F-014	The application can display doctor data based on the keywords entered	UC-27	SD-27	-	Implemented
REQ-F-015	The application can save additional data pre-operative into the database.	UC-10	SD-10	IM-12	Implemented
REQ-F-016	The application can display all registered preoperative data	UC-10, UC-12, UC-13	SD-10, SD-11, SD-13	IM-11	Implemented
REQ-F-017	The application can handle changes to preoperative data and save these changes into the file database	UC-12	SD-11	IM-13	Implemented
REQ-F-18	The application can handle preoperative data deletion.	UC-11	SD-12	-	Implemented
REQ-F-19	The application can display detailed preoperative data as selected.	UC-13	SD-13	IM-14	Implemented
REQ-F-20	The application can display pre-operative data based on the keywords entered.	UC-28	SD-28	-	Implemented
REQ-F-21	The application can save additional data operative into the database	UC-14	SD-14	IN 16	Implemented
REQ-F-22	The application can display all registered intra-operative data.	UC-14, UC-16, UC-17	SD-14, SD-15, SD-17	IN THE 15	Implemented
REQ-F-23	The application can handle intra-operative data changes and save these changes database.	UC-16	SD-15	IM-17	Implemented
REQ-F-24	The application can handle data deletion intra operative.	UC-15	SD-16	-	Implemented
REQ-F-25	The application can display detailed data intra operative according to the chosen one	UC-17	SD-17	IM-18	Implemented
REQ-F-26	The application can display intra-operative data based on the keywords entered	UC-29	SD-29	-	Implemented
REQ-F-	The application can save	UC-18	SD-18	IM-20	Implement

27	additional data post-operative into database.				ed
REQ-F-28	The application can display all the data post-operative registered.	UC-18, UC-20, UC-21	SD-18, SD-19, SD-21	IM-19	Implemented
REQ-F-29	Applications can handle data changes post-operative and save the changes in database.	UC-20	SD-19	IM-21	Implemented
REQ-F-30	The application can handle data deletion post-operative.	UC-19	SD-20	-	Implemented
REQ-F-31	The application can display detailed data post- operatives according to the chosen one.	UC-21	SD-21	IM-22	Implemented
REQ-F-32	The application can display post-operative data based on the keywords entered.	UC-30	SD-30	-	Implemented
REQ-F-33	The application can handle changes in preoperative data validation status when the data has been validated by the Supervising Doctor/Specialist.	UC-22	SD-25	IM-26	Implemented
REQ-F-34	The application can handle data validation state changes operative when the data has been validated by the Supervising Doctor/Specialist	UC-23	SD-26	IM-27	Implemented
REQ-F-35	The application can handle data validation state changes operative when the data has been validated by the Supervising Doctor/Specialist	UC-24	SD-27	IM-28	Implemented

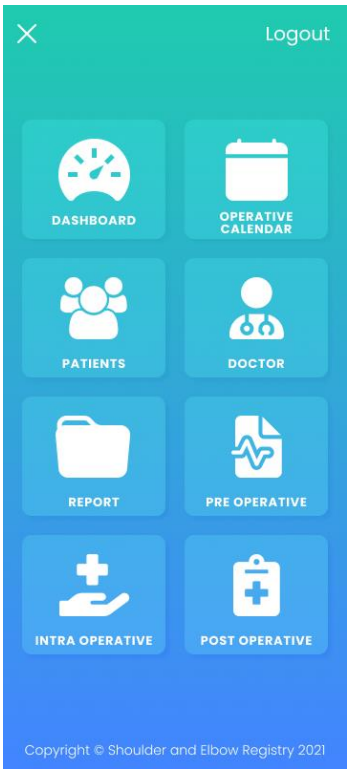

Figure 4. Impelementation FR01

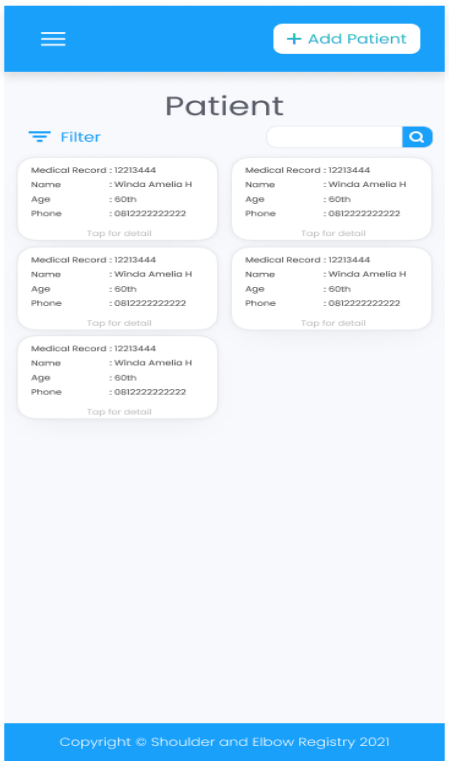

Figure 5. Impelementation FR03

CONCLUSIONS AND RECOMMENDATIONS

Conclusion of the achievements based on the objectives, results of analysis, design and implementation that have been carried out in the development of the Electronic Registry for Shoulder and Elbow, namely a mobile-based electronic register book application that has been successfully developed. The goals of this study were achieved using method reverse engineering, and the output are the 12-web e-registry features, only 8 features were developed for the mobile application. The features of the mobile-based application that have been built include:

A. Login Feature
B. Dashboard Features
C. Patient Data Management features, including adding data, deleting data, displaying data details and searching for data.
D. Doctor Data Management features, including adding data, deleting data, displaying data details and searching for data.
E. Pre-Operative Data Management features, including data deletion, displaying data details and data search.
F. Intra Operative Data Management features, including data deletion, displaying data details and data search.
G. Post-Operative Data Management features, including data deletion, displaying data details and data search.
H. The doctor feature validates operative data.

From the results of the application development, implementation of electronic medical records can prevent medical errors and support patient safety.

Also, further research with different study designs needs to be done to identify the effectiveness of this design and implementation using reverse engineering.

REFERENCES

Abdillah, N., & Ihksan, M. (2021). *Design of Web-Based Patient Registration Information System at Asy-Syifa Medika Clinic*.

Albani, A., Raber, D., & Winter, R. (2016). A Conceptual Framework for Understanding the Essence of Enterprise Engineering Methodologies. *Enterprise Modelling And Information Systems Architectures*, *11*(1), 1–26. https://doi.org/10.18417/emisa.11.1

BPS-BKKBN. (2008). *Indonesia Demographic and Health Survey: Vol. Desember*. Biro Pusat Statistik-Badan Kependudukan dan Keluarga Berencana Nasional.

Horkoff, J., Aydemir, F. B., Cardoso, E., Li, T., Maté, A., Paja, E., Salnitri, M., Mylopoulos, J., & Giorgini, P. (2016). Goal-Oriented Requirements Engineering: A Systematic Literature Map. *Proceedings - 2016 IEEE 24th International Requirements Engineering Conference, RE 2016*, 106–115. https://doi.org/10.1109/RE.2016.41

Larman, Craig. (2005). *Applying UML and patterns : an introduction to object-oriented analysis and design and iterative development*. Prentice Hall PTR.

Leff, A., & Rayfield, J. T. (2001). Web-application development using the Model/View/Controller design pattern. *Proceedings Fifth IEEE International Enterprise Distributed Object Computing Conference*, 118–127. https://doi.org/10.1109/EDOC.2001.950428

Pagani, M. (2004). Determinants of adoption of third generation mobile multimedia services. *Journal of Interactive Marketing*, *18*(3), 46–59. https://doi.org/10.1002/dir.20011

Pressman, R. (2015). Software Engineering Seventh Edition. *Metode Waterfall*.

Roy, D., Tripathy, S., Kar, S. K., Sharma, N., Verma, S. K., & Kaushal, V. (2020). Study of knowledge, attitude, anxiety & perceived mental healthcare need in Indian population during COVID-19 pandemic. *Asian Journal of Psychiatry*, *51*, 102083. https://doi.org/https://doi.org/10.1016/j.ajp.2020.102083

Sample Unified Process Artifacts and Timing (s-start; r-refine). (n.d.).

Venkateswaran, M., Mørkrid, K., Ghanem, B., Abbas, E., Abuward, I., Baniode, M., Norheim, O. F., & Frøen, J. F. (2018). eRegQual-an electronic health registry with interactive checklists and clinical decision support for improving quality of antenatal care: Study protocol for a cluster randomized trial. *Trials*, *19*(1). https://doi.org/10.1186/s13063-017-2386-5

Venkateswaran, M., Mørkrid, K., Rose, C. J., Frøen, J. F., Ghanem MPH, B., Abbas, E., Abu Khader MPH, K., Abu Ward, I., Awwad MPH, T., Isbeih, M., Hijaz MPH, T., Baniode, M., Frederik Frøen, J., Venkateswaran, M., Ghanem, B., Abbas, E., Abu Khader, K., Abu Ward, I., Awwad, T., … Rose, C. J. (2022). A digital health registry with clinical decision support for improving quality of antenatal care in Palestine (eRegQual): a pragmatic, cluster-randomised, controlled, superiority trial. In *Articles Lancet Digit Health* (Vol. 4). www.thelancet.com/