NOWLEDGE REPOSITORY AND MANAGEMENT FOR AADK

MUHAMMAD AKMAL HAZIM BIN MOHD ZAKI

DEPARTMENT OF SOFTWARE ENGINEERING

AND INFORMATION SYSTEM

FACULTY OF COMPUTER SCIENCE AND

INFORMATION TECHNOLOGY

UNIVERSITI PUTRA MALAYSIA

2022/2023

Supervisor Initiation

Knowledge Repository and Management for AADK

Has been prepared by:

Muhammad Akmal Hazim bin Mohd Zaki (201075)

Has been checked and approved by:

Dr. Jamilah Din

Check and approved by:

Dr. Jamilah Din

Project Supervisor

Department of Software Engineering and Information System

Faculty of Computer Science and Information Technology

Universiti Putra Malaysia (UPM)

Date:

Student's Declaration

I hereby confirm that this thesis is my original work with the available information getting from sources mentioned in this thesis; quotations, illustrations and citations have been duly acknowledged. This thesis also has not been submitted previously or concurrently for any bachelor students of Universiti Putra Malaysia or other institutions.

Muhammad Akmal Hazim bin Mohd Zaki

Date:

Acknowledgement

First and foremost, this thesis has benefited from the blessings of Allah SWT, the Almighty and Merciful for being so kind to me.

My appreciation will forever be to my supervisor, Dr. Jamilah Din for her patience, motivation, enthusiasm, and immense knowledge. Her guidance helped me in all the time of project and writing of this thesis. I could not have imagined having a better advisor and mentor for my final year project. I would especially want to express my gratitude for her perseverance, advice, precise information, and insightful remarks despite her obligations, administrative work, and hectic schedule.

Besides my advisor, I would like to thank both of my parents Mohd Zaki Mohd Zin and Roshaliza Mohamad for their prayers, caring and supporting me mentally and physically to complete the project. I would not achieve what I have today without both of my loving parents. My gratitude also sent to my siblings and family that helps and believe in me to complete this thesis.

Finally, I am also thankful to all my friends especially Syafiqah who have been my great supporter and directed me a lot. Also, my sincere thanks also go to my fellow batchmates Hakim, Haris, Ammar, Addin and Hafizan throughout degree.

Abstract

Agensi AntiDadah Kebangsaan (AADK) is responsible for determining all the country's effort to combat the drug threat. e-Parlimen is a web-based application that provided knowledge management and repository for Agensi AntiDadah Kebangsaan (AADK) to improve existing work processes about question and answer that relate with drugs in Parliament. Nowadays, there is no online platform that use by AADK to collect, record, search and manage the answer collection in the Parliament session and only manual method use to record and manage the answer from the Parliament session. Objectives for this project are design a repository to store all information regarding Parliament and develop e-Parlimen system based on technique identified.

Methodology used in the project is Rapid Application Development (RAD), a sort of Agile software development technique. At the end of this project, the system has been fully function and will be use as soon as possible.

Abstrak

Agensi AntiDadah Kebangsaan (AADK) bertanggungjawab menentukan segala usaha negara untuk memerangi ancaman dadah. e-Parlimen adalah aplikasi berasaskan web yang menyediakan pengurusan maklumat dan repositori untuk AADK bagi menambak baik proses kerja sedia ada berkaitan soal jawab yang berkaitan dadah di parlimen. Pada masa kini, tiada platform dalam talian yang digunakan oleh AADK untuk mengumpul, merekod, mencari dan mengurus jawapan dalam sidang Parlimen dan hanya menggunakan kaedah manual untuk merekod dan mengurus jawapan daripada sidang Parlimen. Objektif projek ini adalah mereka bentuk repositori untuk menyimpan semua maklumat mengenai Parlimen dan membangunkan sistem e-Parlimen berdasarkan teknik yang dikenal pasti.

Metodologi yang digunakan dalam projek ini ialah Rapid Application Development (RAD), sejenis teknik pembangunan perisian Agile. Pada akhir projek ini, sistem telah berfungsi sepenuhnya dan akan digunakan secepat mungkin.

Table of Content

List of Figures

List of Tables

CHAPTER 1: INTRODUCTION

Manual filing system is a system that save file or information by hand and without using a computer system. Manual filing system are the paper-based filing and is not efficient for searching especially when need to get the file or information immediately because need to open the file and revise the document in detail. Manual filing system also increases error because all tasks will be done by human and will lead to wrongly saved or loss of file or information and will affect other's trust. Recently, many of manual filing system already change to computerized system to save time and avoid human error.

Agensi AntiDadah Kebangsaan (AADK) is responsible for determining all the country's effort to combat the drug threat. AADK also responsible to show the statistics about the drug in Parliament. Nowadays, there is no online platform that use by AADK to collect, record, search and manage the answer collection in the Parliament session. The manual method use is to record and manage the answer from the Parliament session. Therefore, the e-Parlimen is an initiative from AADK to fully use online platforms to collect, record, search and manage the answer to improve existing work processes from manual to electronic filing. This initiative is in line with the government's move towards e-Government.

1.1 Problem Statement

Currently, the recording and managing process for the answer from Parliament conducted manually. This means the staff that responsible for this task will record and manage the task by hand or traditional way. This will lead to wrongly managed and lost the answer from the Parliament.

Next, the staff that responsible for the record will manually search and references the record. The staff will take long time to find the document because need to open the file and check the document in detail. This method is not efficient especially if the document request from many people.

Furhermore, the access of the answer is only through the staff incharged. The answer of Parliament is confidential, and the Parliament staff need to check the status of people requesting the documents. This method will increase the time to get the document and will lead to unavailability if the staff incharged is not available.

1.2 Objectives

The objectives for e-Parlimen system are:

1. Design a repository to store all information regarding parliament.

2. Identify technique or tool to search the information in the repository.

3. Develop e-Parlimen system based on the design and technique or tool identified.

1.3 Scopes

Scopes for this project divided into two:

User scope

I. The system should be able to store and manage the documents in repository.
II. The system should allow user searching the documents.
III. The system should allow authorized users to view documents.

System scope

I. This project focus on the development of the front-end and back-end of the web system.
II. Only authorized users can access this system.

1.4 Thesis Organisation

This thesis is divided into 6 chapters and is structured as follow: introduction in chapter 1 followed by literature review in chapter 2, methodology in chapter 3, system analysis and design in chapter 4, implementation and testing in chapter 5 and finally, discussion in chapter 6.

The project background, problem statement, objectives and scope of this project have been explained earlier in this chapter.

Chapter 2 covers the literature review that explain tools or techniques identified for searching documents.

Chapter 3 focuses on the methodology used which explains the processes to develop and implement the system.

Chapter 4 consists of the system analysis and design which includes the system requirements, functions, use case and database designs.

Chapter 5 discusses the implementation and testing of system such as user interface with the functions and implementation of whole technologies used in the system.

Chapter 6 concludes all the processes from the earlier stages and proposal for future work of this project.

CHAPTER 2: LITERATURE REVIEW

For this project, the main concern is on repository and searching technique to use in this system. This is because, e-Parlimen system is related to data repository and searching the data that have been store. For e-Parliamen system, the process to store data is mapping to Knowledge Management to make sure the document can be store in more efficient way. Knowledge management is process to gather, organize, analyse, and share the information in more efficient way. From the input gather from, it will organize the input and analyse the input using some specific algorithm and will make the system boost the decision-making ability or in simple words to retrieve the data in more efficient way.

2.1 Knowledge Management

Knowledge management can be divided into 2 main types which are implicit and explicit, as shown in Figure 2.1. We need to distinguish these types of knowledge because this help to choose the suitable tools or technologies that can be used for this project.

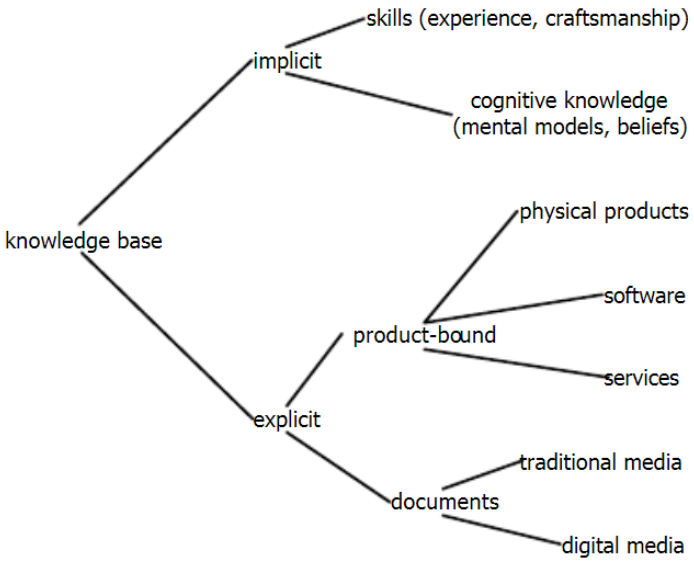

Figure 2.1 Types of Knowledge Management

The implicit knowledge is the practical knowledge. Implicit knowledge can provide a variety of options for carrying out a task as well as potential outcomes, resulting in a careful process to decide on the optimal course of action.

Another type is explicit knowledge, which is knowing as basic form of knowledge and easy to pass along because it is written and accessible. Explicit knowledge can be determined once the data has been organised, analysed, and interpreted. It can be expressed, recorded, shared, and stored with ease.

2.2 Search Engines

A search engine is a computer software that looks through documents for a list of keywords and documents that match those keywords. Although they are a general class of programmes, search engines can also be sourced from databases and data warehouses.

	Elasticsearch	Solr	Sphinx
Search features	Medium (between 5 and 7 features)	High (more than 7 features)	High (more than 7 features)
Performance	High (Read more than 500 queries per second)	High (Read more than 500 queries per second)	High (Read more than 500 queries per second)
Scalability	High (up to 1TB)	High (up to 1TB)	High (up to 1TB)
Can read rich documents	Yes	Yes	No
Visualization of Data	Yes	Yes	No

Table 2.1 Search Engine Tools Comparison

Table 2.1 indicates the comparison between the search engine tools between Elasticsearch, Solr and Sphinx. From the table we can see Elasticsearch have medium search features not many as Solr and Sphinx because Elasticsearch is still a young technology. Both of Elasticsearch and Solr can read rich documents such as word and PDF file and give them a huge advantage compared to Sphinx. Also, both tools allowed users to have visualization of data. The similarities of these tools are real time indexing, high performance, and high scalability.

2.3 pdf-Parser

From the user requirement, this application need to read all the detail of the PDF files so pdf-Parser is a technique that can be implement to extract raw data from PDF file. pdf-Parser is a Command-line program that parses and analyses PDF documents. It provides features to extract raw data from PDF documents, like compressed images. pdf-parser can deal with malicious PDF documents that use obfuscation features of the PDF language.

2.4 Summary

Throughout the literature review process, some of the techniques and tools have been found that can be use to the system to increase process to gather, organize, analyse, and share the information in more efficient way. At the end of literature review, pdf-Parser have been used to the system because pdf-Parser can extract all data from PDF file and increase searching processes.

CHAPTER 3 METHODOLOGY

This chapter will be describing the methodology that will be used for this project. This project was developed by following Rapid Application Development (RAD) methodology. This chapter discusses about the RAD and phases involved during development.

3.1 Rapid Application Development

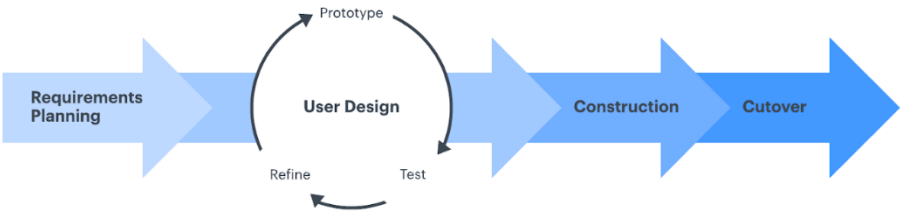

Figure 3.1 Rapid Application Development cycles

Figure 3.1 above shows Rapid Application Development cycles. RAD start with define and finalize project requirements planning. Then, building prototypes and gather user feedback from the prototypes. Development of this application was in construction phase based on user feedback earlier.

Date	Purpose
22 July 2022	First meeting with stakeholders.
12 August 2022	Second meeting to confirm user requirements.
22 September 2022	Gain feedback for module 1.
31October 2022	Gain feedback for module 2.
16 November 2022	Demo all function for module 1 and 2.
17 January 2023	Gain feedback for module 3.
7 February 2023	Demo all function for the system

Table 3.1 Meeting held with stakeholders

Rapid Application Development need to have constant meetings with users. Table 3.1 shows the meeting held with stakeholders throughout this project development.

3.1.1 Requirement Planning

Sprint	Module
Sprint 1	Manage User
Sprint 2	Manage Question and Answer
Sprint 3	Dashboard

Table 3.2 Sprint requirements

For requirement planning I already met users to get the user requirements as shown in Table 3.1 on 22 July 2022. Then, after I gathered requirement from users, I met users for the second meeting on 12 August to confirm user requirements. From the requirement planning we decided to have 17 functions

that divided into three modules as shown in Table 3.2. From the first and second meeting we also have decided to user Laravel as the framework, PHP as the programming language and MySQL as database for this project.

3.1.2 User Design

Each modules will start with building prototypes and gather user feedback from the prototypes. As shown in Table 3.2 each module has been shown to the stakeholders to gain feedback for each module. For module 2 we have decided to user pdf-parser as technique to extract data from PDF files for the searching.

3.1.3 Construction

After gain feedback from user at design phases, development for all modules occur using PHP as programming language, Laravel as framework and MySQL as database. Demonstrate of each module have been presented to the user because for this methodology users can still suggest changes or improvements as the software is being developed. Some improvements were made for the interface to increase user experience.

3.1.4 Cutover

Cutover phase was the final phase in Rapid Application Development. After demonstrated all the working functions on the last meeting users decided to have future enhancements that explain in 6.2 and this application will be deploy as soon as possible.

3.2 Summary

At the end of this Rapid Application Development methodology constant feedback from user have been gained to increase user experience using this system. Also, programming language, framework, technique and database that been used at the construction were suitable to use for this system.

CHAPTER 4: SYSTEM ANALYSIS AND DESIGN

This chapter has two sub-sections: system analysis, and system architecture and detailed design. System analysis section discusses system use cases with their description. While in system architecture and detailed design section discuss a few diagrams of the system architecture design.

4.1 Use Case Model

This section will explain all the use cases for the system. All the use cases will be divided into three modules that are manage user, manage question and answer and dashboard.

4.1.1 Manage User Module

a) Use Case Diagram

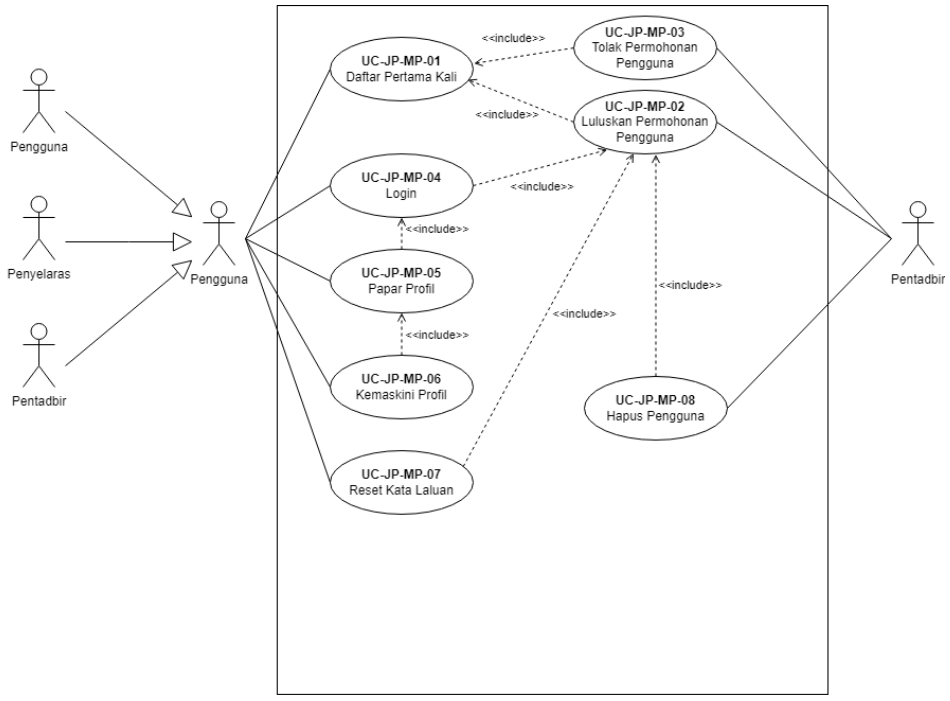

Figure 4.1 Manage User Use Case Diagram

Figure 4.1 shows the use case diagram for manage user module. *Pengguna, Penyelaras* and *Pentadbir* are combined as *Pengguna* and can *Daftar Pertama Kali, Login, Papar Profil, Kemaskini Profil* and *Reset Kata Laluan*. Also for *Pentadbir, Pentadbir* can *Terima Permohonan Pengguna,Tolak Permohonan Pengguna* and *Hapus Pengguna.*

b) Use Case Description

Table 4.1 shows the description for each of use case in figure 4.1.

Use Case ID	**Use Case Name**	**Description**
UC-JP-MP-01	*Daftar Pertama Kali*	New user must register before user is eligible to access the system.
UC-JP-MP-02	*Luluskan Permohonan Pengguna*	*Pentadbir* will approve the application of eligible users to use the system.
UC-JP-MP-03	*Tolak Permohonan Pengguna*	*Pentadbir* will reject the application of users who are not eligible to use the system.
UC-JP-MP-04	*Login*	Users need to login using their email and password to access the system.
UC-JP-MP-05	*Papar Profil*	User can display their user profile.
UC-JP-MP-06	*Kemaskini Profil*	Users can update their profile.

UC-JP-MP-07	*Reset Kata Laluan*	Users can reset the password by entering user's email.
UC-JP-MP-08	*Hapus Pengguna*	*Pentadbir* can delete users from using the system

Table 4.1 Manage User Use Case Description

4.1.2 Manage Question and Answer Module

a) Use Case Diagram

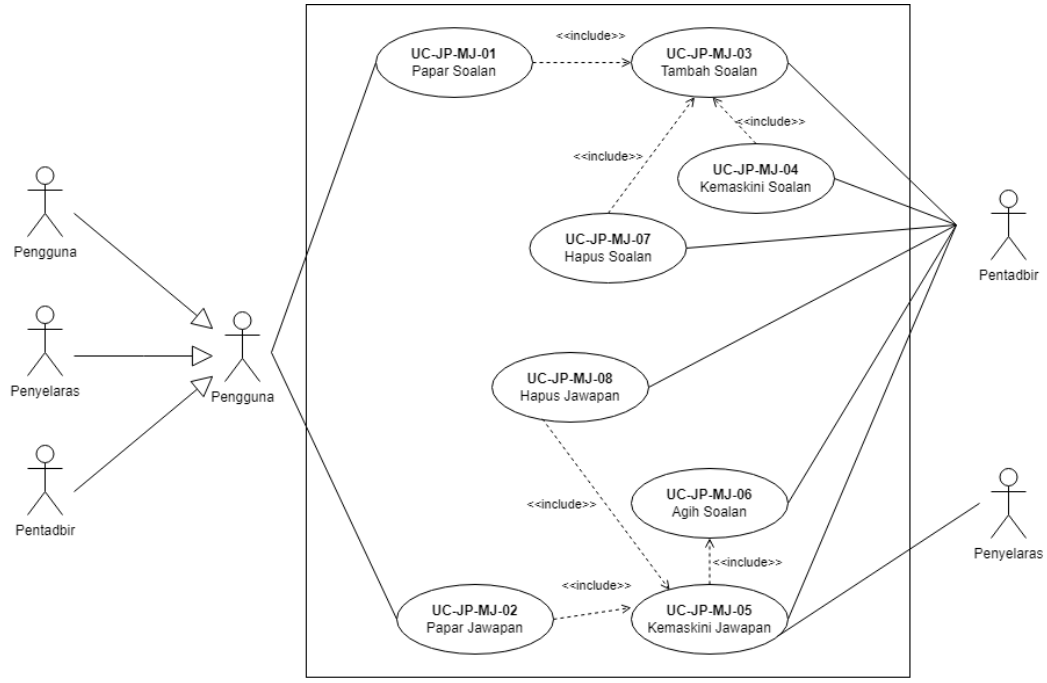

Figure 4.2 Manage Question and Answer Use Case Diagram

Figure 4.2 shows the use case diagram for manage question and answer module. *Pengguna, Penyelaras* and *Pentadbir* as *Pengguna* can *Papar Soalan* and *Papar Jawapan. Penyelaras* also can *Kemaskini Jawapan*. For *Pentadbir, Pentadbir* can *Tambah Soalan, Kemaskini Soalan, Kemaskini Jawapan, Agih Soalan* and

b) Use Case Description

Table 4.2 shows the description for each of use case in figure 4.2.

Use Case ID	**Use Case Name**	**Description**
UC-JP-MJ-01	*Papar Soalan*	Users can view the question.
UC-JP-MJ-02	*Papar Jawapan*	Users can view the answer.
UC-JP-MJ-03	*Tambah Soalan*	*Pentadbir* can add new question in the system.
UC-JP-MJ-04	*Kemaskini Soalan*	*Pentadbir* can edit question in the system.
UC-JP-MJ-05	*Kemaskini Jawapan*	*Pentadbir* and *Penyelaras* can edit answer in the system.
UC-JP-MJ-06	*Agih Soalan*	*Pentadbir* can share the question to *Penyelaras*.
UC-JP-MJ-07	*Hapus Soalan*	*Pentadbir* can delete question in the system.
UC-JP-MJ-08	*Hapus Jawapan*	*Pentadbir* can delete answer in the system.

Table 4.2 Manage Question and Answer Case Description

4.1.3 Module Dashboard

a) Use Case Diagram

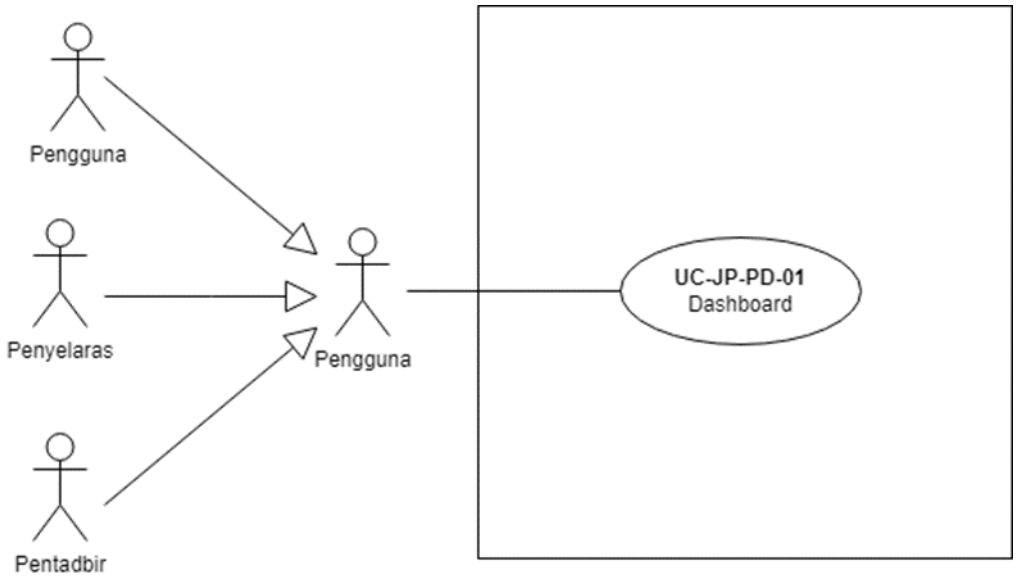

Figure 4.3 Dashboard Use Case Diagram

Figure 4.3 shows all the users that are *Pengguna, Penyelaras* and *Pentadbir* can view Dashboard.

b) Use Case Description

Table 4.3 show the description for use case in figure 4.3.

Use Case ID	**Use Case Name**	**Description**
UC-JP-PD-01	Dashboard	Users can view dashboard of the system that show table about *jenis pertanyaan.*

Table 4.3 Dashboard Use Case Description

4.2 Non-Functional Requirements

Non-functional ID	Requirements	Description
NF-AS-01	Modularity	The system developed must be modular so that it is easy to maintain
NF-AS-02	Scalability	The system must be able to accommodate unlimited number of users
NF-AS-03	Response time transaction	The response time of a transaction is lower than 5 seconds.

Table 4.4 Non-functional Requirements Description

Table 4.4 shows all the non-functional requirement description. A non-functional requirement is a requirement that specifies how the system should behave.

4.3 System Design and Architecture

IEEE defines architectural design as the process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system. So, this section can help to understand how all these components can be integrated to form a system.

4.3.1 Architectural Diagram

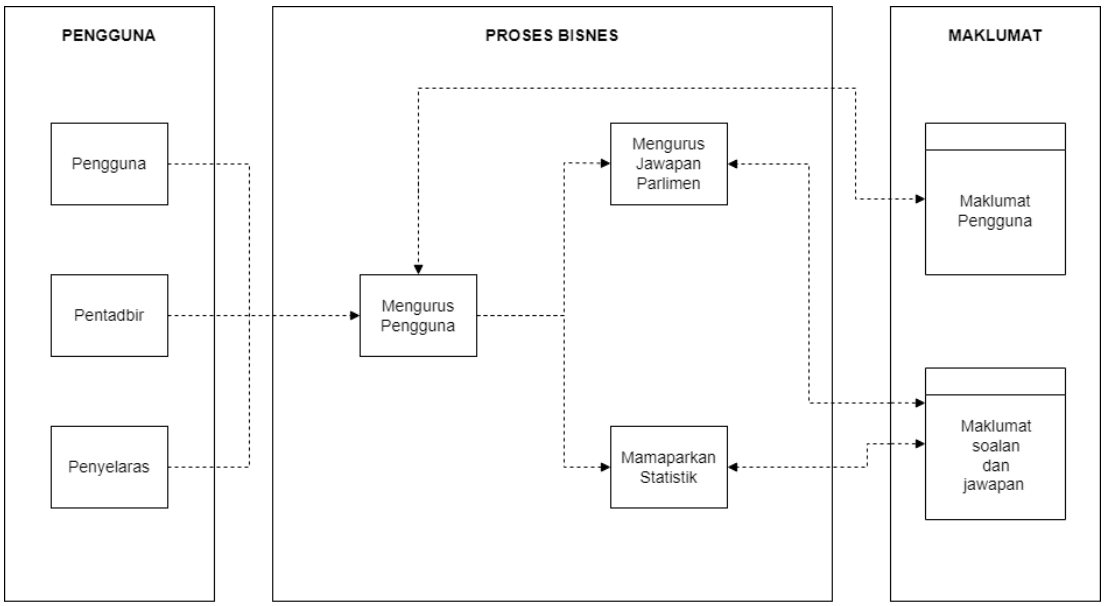

Figure 4.4 Architectural Diagram

Figure 4.4 shows the architectural diagram for the system development. It contains the connection between *pengguna, proses bisnes* and *maklumat* that required.

4.3.2 Entity-Relationship-Diagram (ERD)

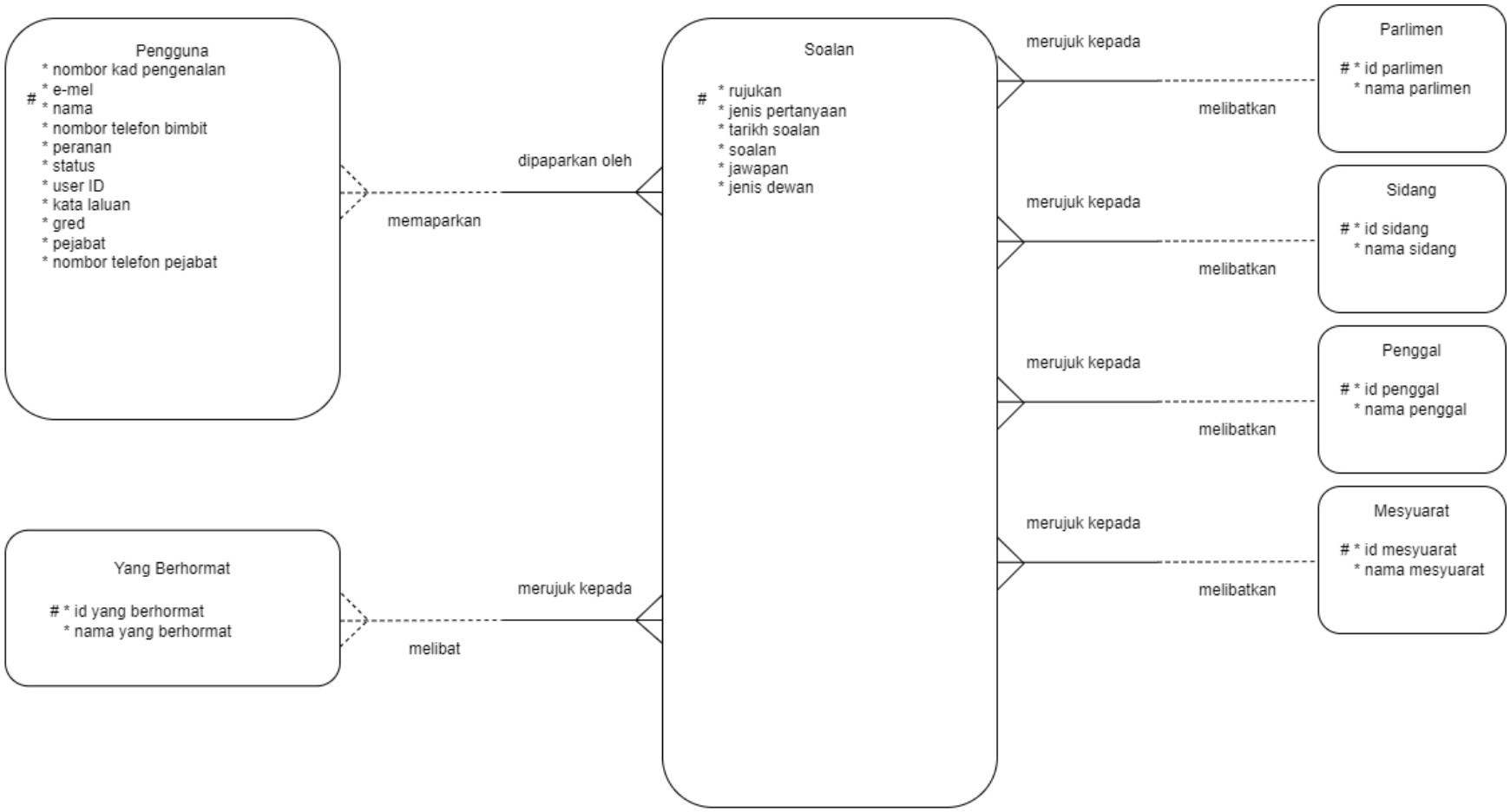

Figure 4.5 Entity-Relationship-Diagram (ERD)

Figure 4.5 shows the Entity-Relationship-Diagram for this system. It has two main tables that are *Pengguna* and *Soalan. Soalan* have reference tables that are *Parlimen, Sidang, Penggal, Mesyuarat* and *Yang Berhormat*. For *Pengguna* table, the primary key is *e-mel* and *e-mel* must be unique. For *Soalan* table*, Rujukan* is the primary key and must be unique. The detail about ERD diagram has been discussed in Appendix B.

4.4 Summary

For this project, there were 3 users for this project that are *Pengguna, Penyelaras* and *Pentadbir*. All the data were saved in Soalan and Pengguna tables and this information will be retrieved in three different modules that are Manage User, Manage Question and Answer, and Dashboard.

CHAPTER 5: IMPLEMENTATION AND EVALUATION

This chapter shows the implementation phase of the application. The results are displayed by describing the user interface designed during the design phase. The implementation phase is conducted using Laravel Framework and PHP as the programming language. The user interface is created using HTML, CSS, and JavaScript. The application is connected to the build in MySQL database for storing the application's data. The testing report then illustrates the application's actual result either the application passes the requirements or fail.

5.1 User Interface Design

a) Module Manage User

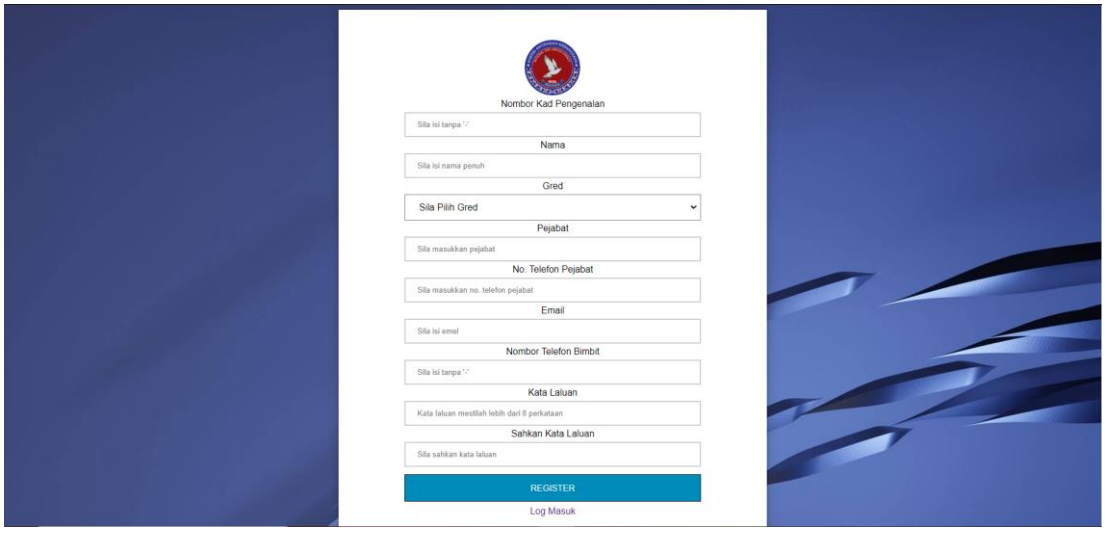

Figure 5.1 Registration Page

Figure 5.1 shows the Registration page for the system. New user need to insert all information in the registration form that are *nombor kad pengenalan, nama, gred,*

pejabat, no telefon pejabat, emel, nombor telefon bimbit, kata laluan and *sahkan kata laluan*. After user register, user needs to wait for *Pentadbir* to approve.

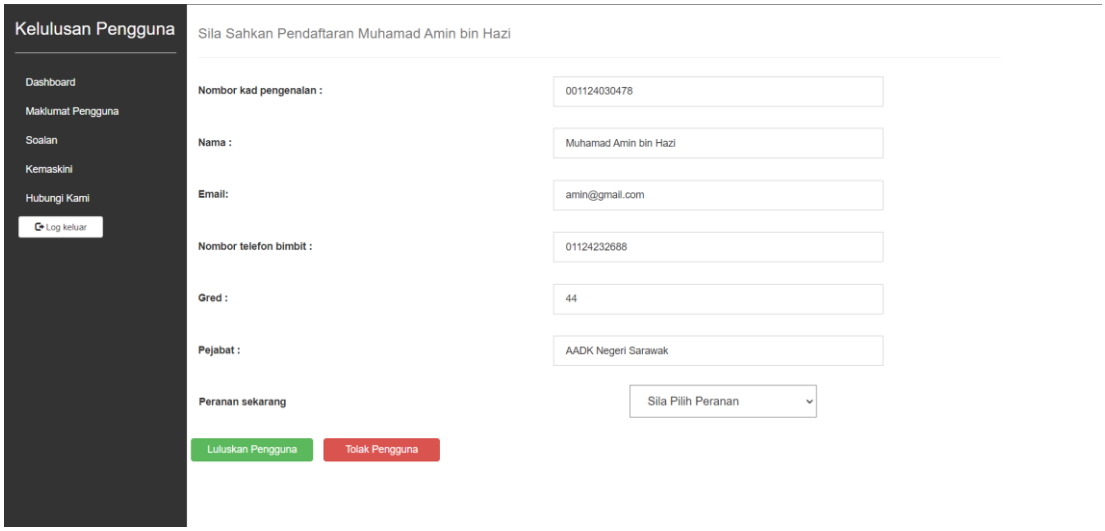

Figure 5.2 User Approval Page

Figure 5.2 shows User Approval Page. After user register, *Pentadbir* need to approve or reject the user registration. If *Pentadbir* want to approve user, *Pentadbir* need to click *Luluskan Pengguna* or if *Pentadbir* want to reject user, *Pentadbir* need to click *Tolak* Pengguna. User will receive email of approval or rejection after the approval process.

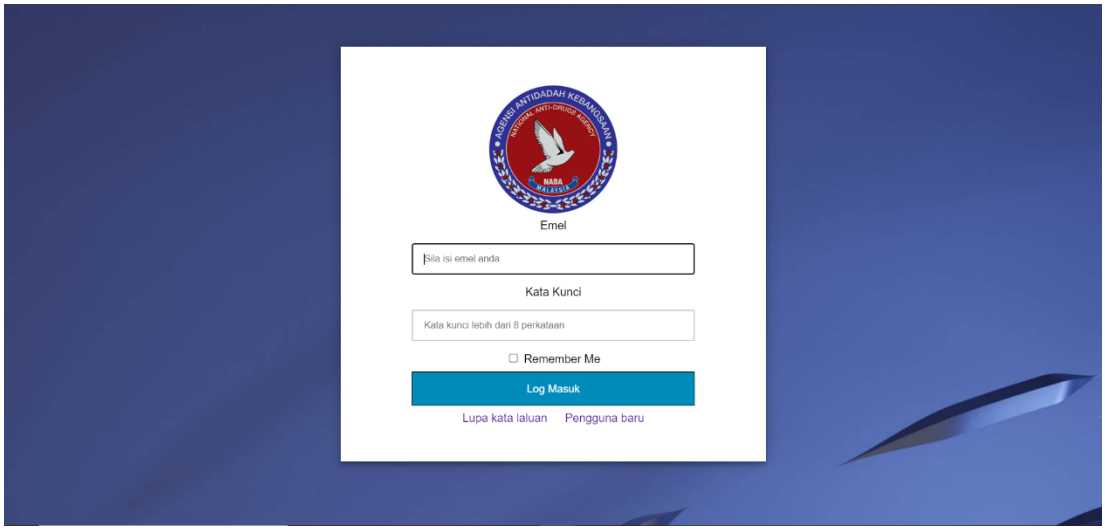

Figure 5.3 Login Page

Figure 5.3 shows Login Page. After users have been approved, users can enter the system by fill their email and password.

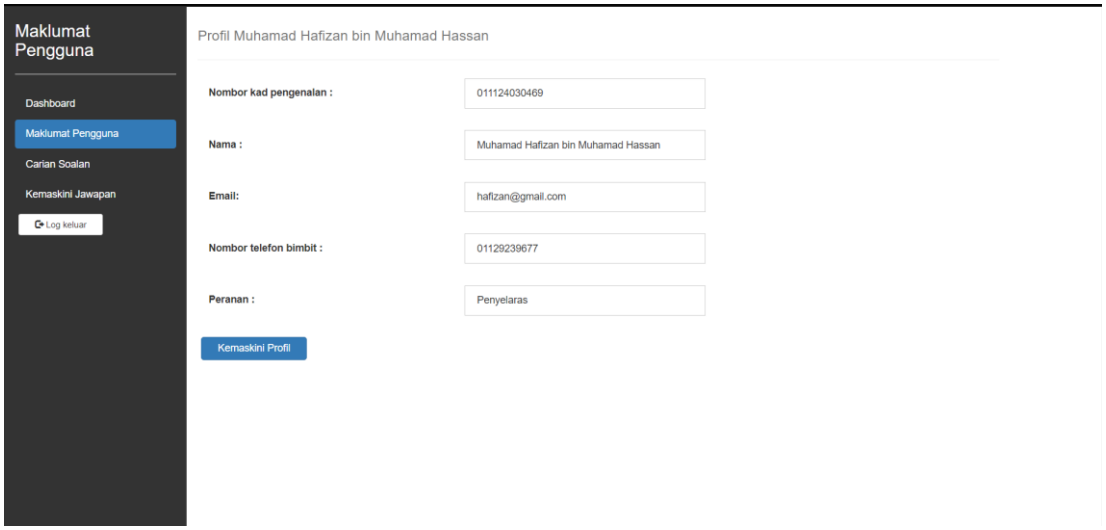

Figure 5.4 Profile Page

User can view their information by click *Maklumat Pengguna* and Figure 5.4 Profile Page will display. User can change their information by click *Kemaskini Profil.*

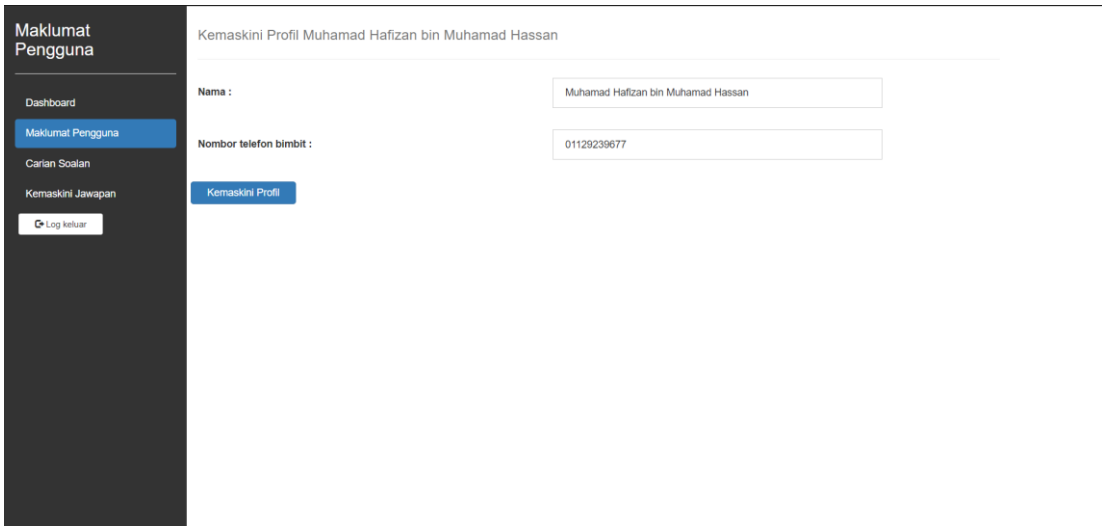

Figure 5.5 Edit Profile Page

If user click *Kemaskini Profil* in Figure 5.4, user will be redirect to Figure 5.5 that is Edit Profile Page. In this page, user can their *nama* and *nombor telefon bimbit.*

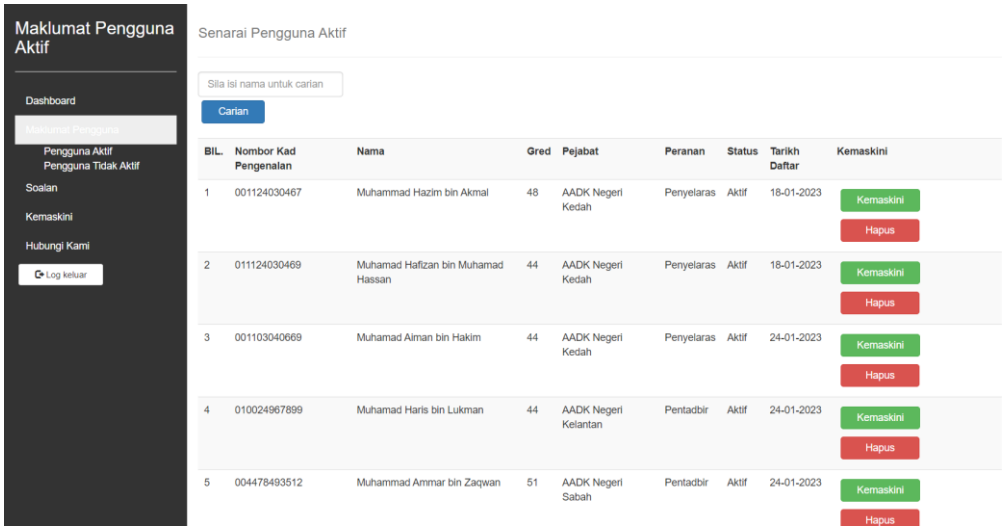

Figure 5.6 Edit or Delete User Page

Pentadbir can view information about active user by clicking on *Pengguna Aktif* and Figure 5.6 will be display. *Pentadbir* can change user information by clicking *Kemaskini* or can delete user from using system by clicking *Hapus*.

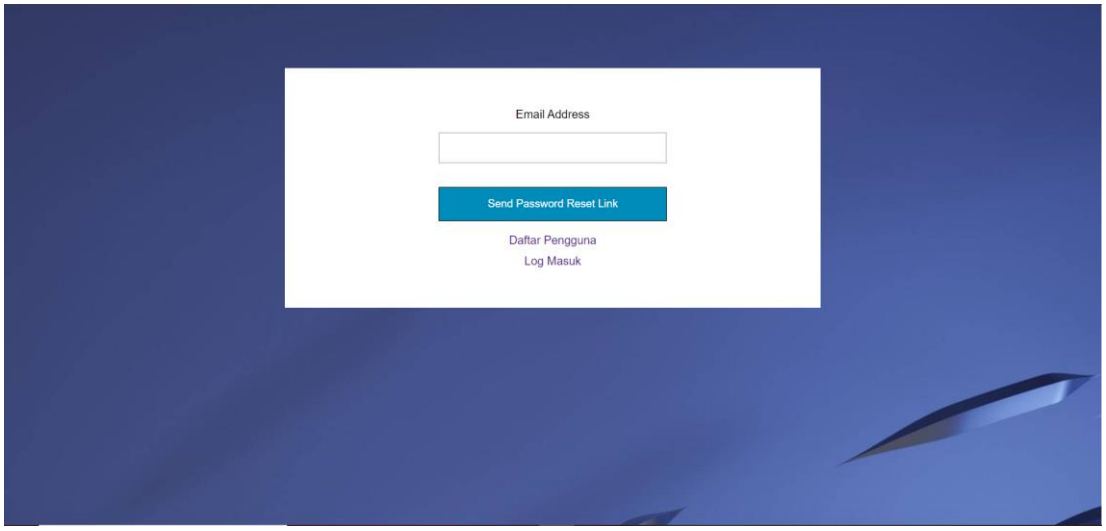

Figure 5.7 Edit or Delete User Page

User that forgets password can click *Lupa Kata Laluan* at Figure 5.3 and Figure 5.7 will be display. User need to enter email and reset password link will be send to user's email.

.

b) Module Manage Question and Answer

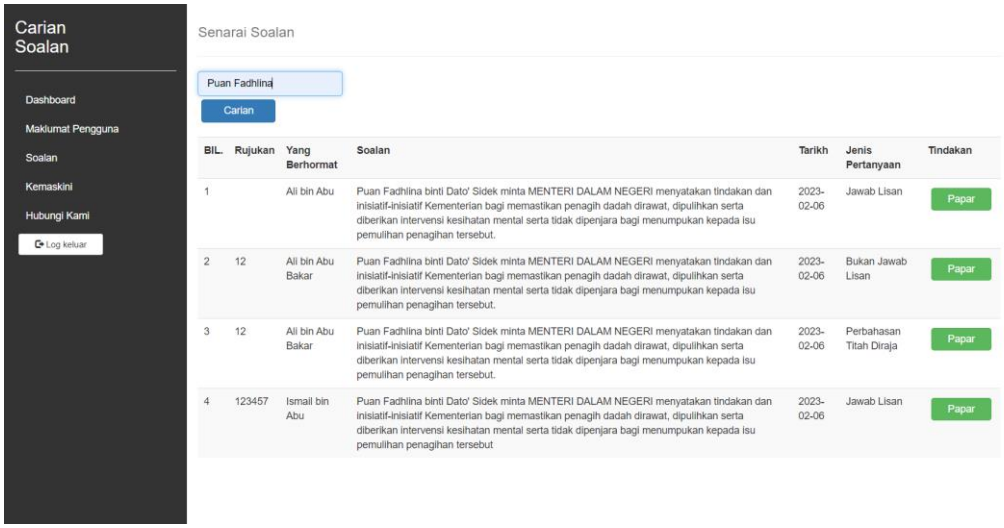

Figure 5.8 Search Question and Answer Page

Figure 5.8 will be display after user click *Soalan*. User can search keywords in question and answer by fill the keywords in the form and click *Carian*. The question and answer that contain the keywords will be display and user can click *Papar* to view the answer in PDF file.

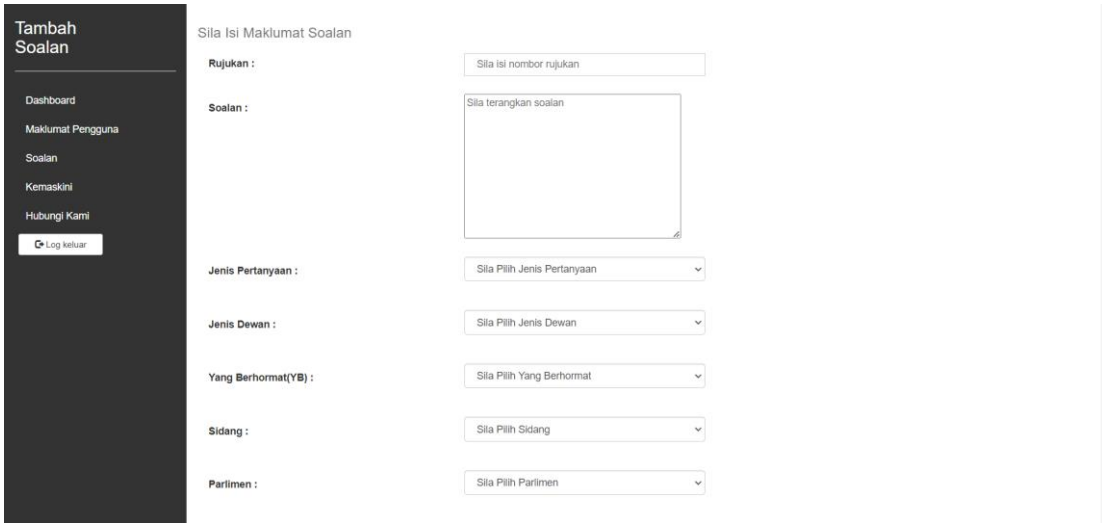

Figure 5.9 Add Question Page

Pentadbir can add new Question by click *Soalan* and then click *Tambah Soalan. Pentadbir* need to fill all the form in Figure 5.9 that are *Rujukan, Soalan, Jenis Pertanyaan, Jenis Dewan, Yang Berhormat(YB), Sidang, Parlimen* and *Tarikh.*

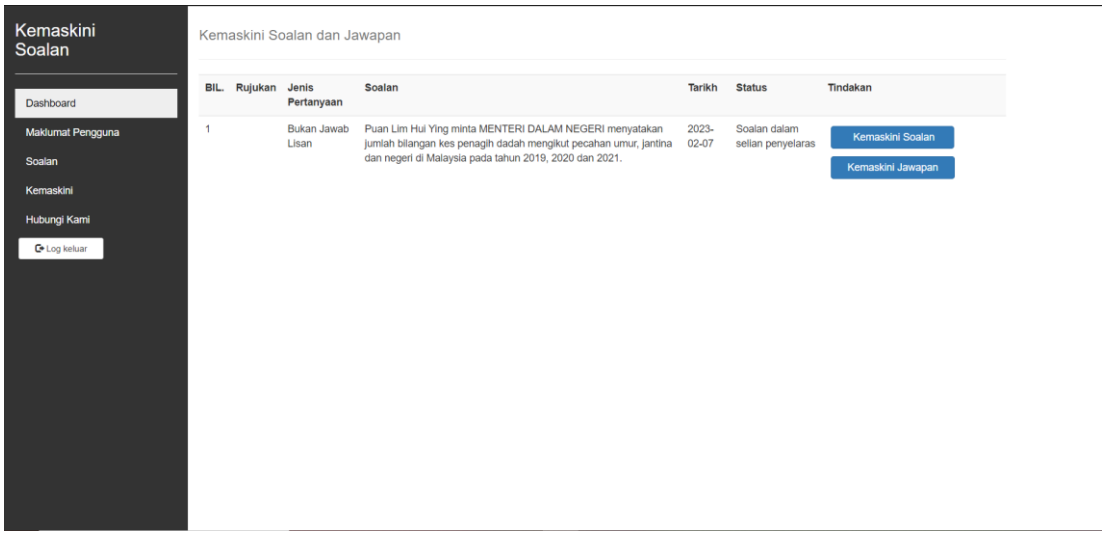

Figure 5.10 Edit Question and Answer Page

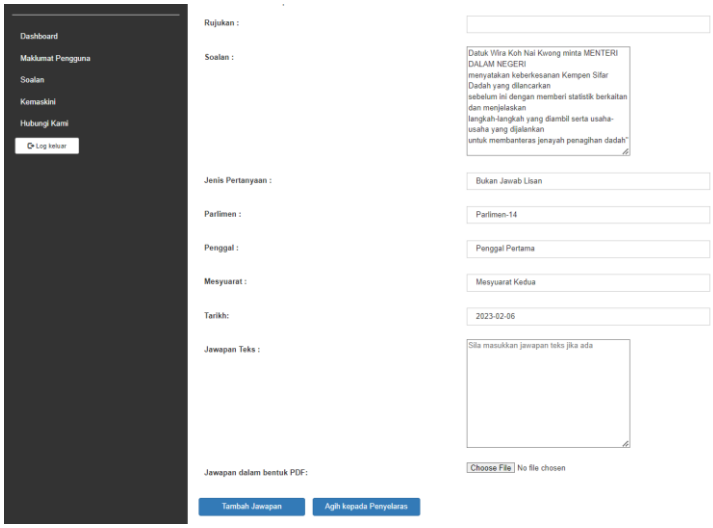

Figure 5.11 Edit Question and Answer Page

All new questions added will be display in Figure 5.10. *Pentadbir* can edit the question, add answer or share the question to the *Penyelaras*. Question that been share to *Penyelaras* can be answered by *Penyelaras.* If *Pentadbir* click *Kemaskini Jawapan* Figure 5.11 will be display. *Pentadbir* can add answer or can share the question to *Penyelaras* by click *Agih Kepada Penyelaras.*

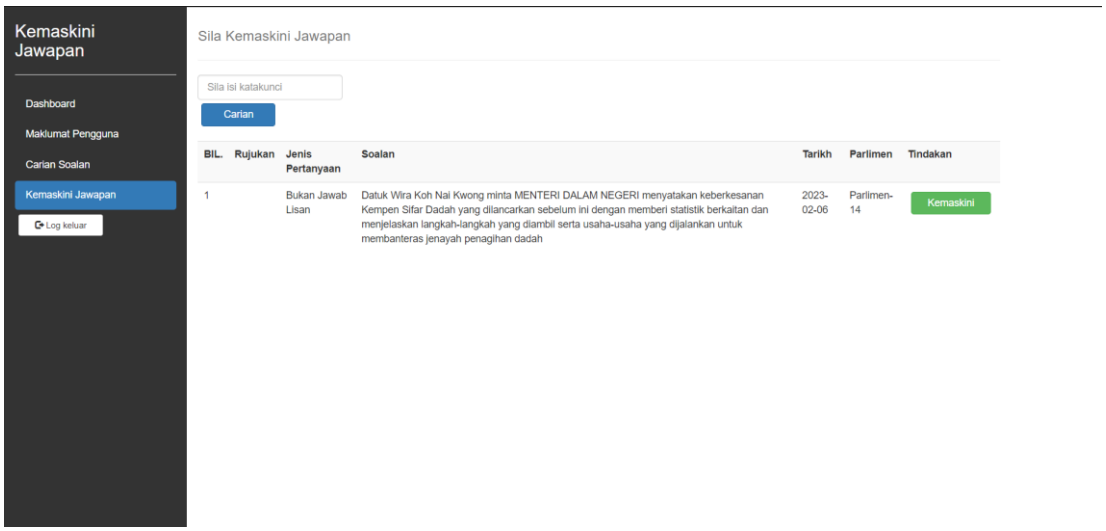

Figure 5.12 Penyelaras Add Answer Page

Question that has been share to *Penyelaras* can be view by *Penyelaras* in Figure 5.12 by clicking *Kemaskini Jawapan*. *Penyelaras* need to click *Kemaskini* to add new answer for the question.

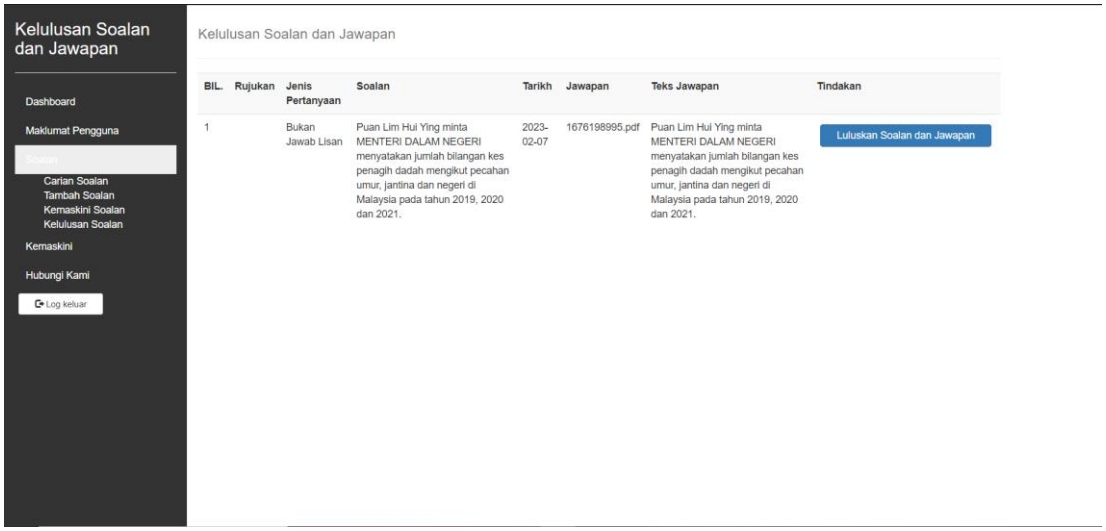

Figure 5.13 Question and Answer Approval Page

Question that has been add by *Pentadbir* or *Penyelaras* will be display in Figure 5.13. *Pentadbir* need to click *Luluskan Soalan dan Jawapan* to approve question and answer before the question and answer can be view by user.

c) Module Dashboard

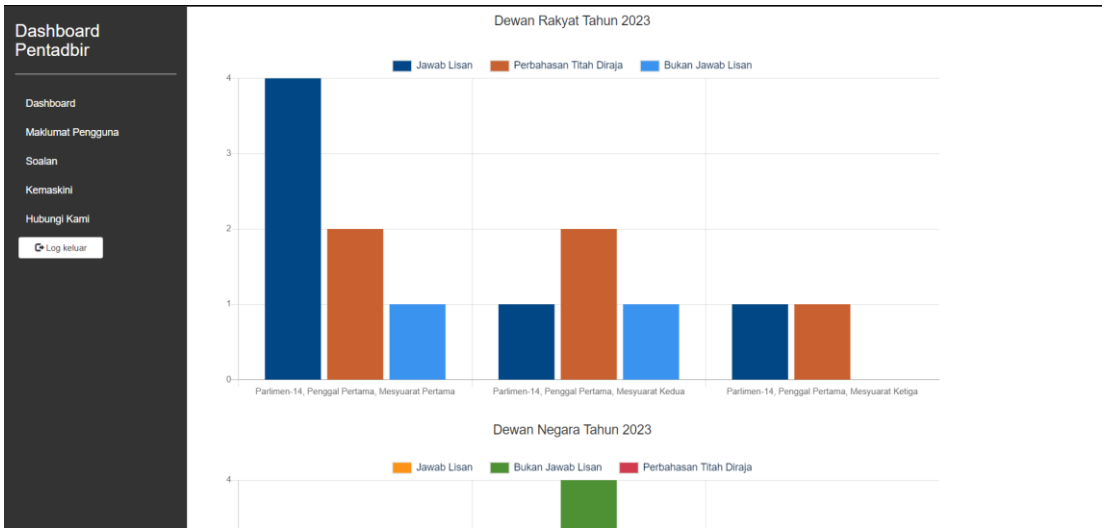

Figure 5.14 Dashboard Page

After login or by clicking *Dashboard*, system will navigate user into dashboard page as shown in Figure 5.14. Dashboard show the statistics of Jenis Pertanyaan for 2023 in graph table.

5.2 Testing Phase

After the implementation phase, testing was conducted to make sure e-Parlimen system fulfil all the user requirements.

5.2.1 Unit Testing

Unit testing has been conducted on this system before proceeding to the User Acceptance Test. Unit testing is important to make sure all the function works as expected. All of the test produce the expected results. The test condition that has conducted mention in Table 5.1 below.

Test ID	Test Condition
TC01	New user can register to the system.
TC02	*Pentadbir* can approve the application of eligible users to use the system.
TC03	*Pentadbir* can reject the application of users who are not eligible to use the system.
TC04	Users can login using their email and password to access the system.
TC05	User can display user profile.
TC06	Users can update their profile.
TC07	User can reset the password by entering user's email.
TC08	*Pentadbir* can delete users from the system
TC09	Users can view the question.
TC10	Users can view the answer.
TC11	*Pentadbir* can add new question in the system.
TC12	*Pentadbir* can edit question in the system.
TC13	*Pentadbir* and *Penyelaras* can edit answer in the system.
TC14	*Pentadbir* can share the question to *Penyelaras.*
TC15	*Pentadbir* can delete question in the system.
TC16	*Pentadbir* can delete answer in the system.
TC17	Users can view dashboard of the system that show table about *jenis pertanyaan.*

Table 5.1 Test Conditions

5.2.2 User Acceptance Test

User Acceptance Test (UAT) have been conducted with users from Agensi AntiDadah Kebangsaan for each module to gain feedback of the system before moving into the next module. All the users required to test the system by following the task provided as guideline. Feedback of the testing will come from multiple aspects such as system efficiency, smoothness, and user interface.

a) Task Description

All of the participants already test this application. For this task, participants need to do so some tasks to make sure all of them understand and know how to do all the function. All the task and description for the test mentioned in Table 5.2 below.

No	Task
1	Register an account by inserting IC, *nama, email, nombor telefon bimbit, gred, pejabat, nombor telefon pejabat*, password and confirm password.
2	Login to the system by filling email and password.
3	Approve and reject new register user.
4	Search question and answer using searching
5	Add new question
6	Edit question
7	Add answer for the question or share question to *Penyelaras*
8	Approve question and answer

Table 5.2 Task Description

b) Task Completion Rate

Usability Testing with Agensi Antidadah Kebangsaan		
Respondents	**Number of Task Completed**	**Task Completed Rate**
1	8/8	100%
2	8/8	100%
3	8/8	100%
4	8/8	100%
5	8/8	100%

Table 5.3 Task Completion Rate

Table 5.3 above shows the completion task for all the respondent for task in table 5.2. As shown in the table, all the respondents successfully completed all the task given with 100% rate.

5.3 Summary

This project has been showed and tested by user from AADK. Interfaces for *Pengguna, Penyelaras* and *Pentadbir* are different from each other because each user will have different role. From the user feedback this application was fully functional and follow all the user requirements that discuss earlier.

CHAPTER 6: CONCLUSION

6.1 Conclusion

The project of e-Parlimen system is an initiative from AADK to move towards e-Government. All the function agreed at initial project have successfully been implemented in the system. All the objectives have been achieved and solve the problem statement mentioned in Chapter one. The implementation of e-Parlimen system will bring a lot of benefits to AADK. This system also will ease the organization to save and search documents in efficient way compared to the manual filing system. This also will increase the people trust to AADK because all the documents will not easily miss place or missing.

6.2 Future Enhancements

There are some enhancements for the future to resolve some absences of feature and improvement in terms of usability and maintenance of the system. The enhancements stated as below:

a) *Gred, Pejabat* in *Pengguna* table and *Jenis Pertanyaan* and *Jenis Dewan* in *Jawapan* table need to be reference tables.

This enhancement needs to make sure database for this system is easy to maintain and be more reusable.

b) Create an interface to add, edit and delete *Gred, Pejabat* and *Parlimen.*

These interfaces need to be implemented to make sure this system can be use although if *Gred, Pejabat* and *Parlimen* are increase. Also, this will ease the work for AADK ICT because this functionality can be done by user itself.

6.3 Summary

This system already implemented all the functions from the user requirements that discuss before development process. Furthermore, this system already been show to the user and have been approved and evaluated by user itself. From the feedback of user, this system needs to have some future enhancements and will be use as soon as possible.

REFERENCES

1. Chris, K. (2021). *What is php? The PHP programming language meaning explained*. freeCodeCamp.org. Retrieved July 4, 2022, from https://www.freecodecamp.org/news/what-is-php-the-php-programming-language-meaning-explained/#:~:text=PHP%20is%20an%20open%2Dsource,Graphical%20User%20Interfaces%20(GUIs).

2. Dymond, A., (2002), The knowledge warehouse: the next step beyond the data warehouse, Procs of SUGI 27, Orlando, Florida, www2.sas.com/proceedings/sugi27/Proceed27.pdf.

3. *Features*. Features - Apache Solr. (n.d.). Retrieved July 5, 2022, from https://solr.apache.org/features.html

4. *Free and open search: The creators of Elasticsearch, Elk & Kibana*. Elastic. (n.d.). Retrieved July 5, 2022, from https://www.elastic.co/

5. Klimenko, A. (2022). *Solr vs Elasticsearch vs Sphinx: Search platforms comparison*. Greenice. Retrieved July 4, 2022, from https://greenice.net/elasticsearch-vs-solr-vs-sphinx-best-open-source-search-platform-comparison/

6. Valamis. *(2022) Knowledge management: Importance, benefits, examples.* Retrieved July 4, 2022, from https://www.valamis.com/hub/knowledge-management

7. *Murad, Roslinda & Hussain, Hanizan & Abd Samad, Nor Hafiza & Tahir, Wan & Ali, Hanif. (2018). Collaborative Research Management Method in Knowledge Management: Conceptual Foundations and Research Issues. 10.1109/ICCOINS.2018.8510583.*

8. *Open source search engine*. Sphinx. (n.d.). Retrieved July 5, 2022, from http://sphinxsearch.com/

9. Perkins, K. (2022). *Top mysql gui tools*. Database Journal. Retrieved July 4, 2022, from https://www.databasejournal.com/features/top-10-mysql-gui-tools/

10. *Tools and techniques for knowledge and information management*. (n.d.). Retrieved July 3, 2022, from https://www.researchgate.net/publication/259574106_Tools_and_techniques_for_knowledge_and_information_management