SDD Document
SEM I 2022/2023

PETAKOM Management System

Group Name

1. HAZIQ DANIAL BIN KHAIRUL AZIZY   [CB20140]
2. AMIR MUHAMMAD HAFIZ BIN OTHMAN   [CB20152]
3. HAZMI HAZIM BIN ZAIRUL FUAAD   [CB20117]
4. NABIL ABHAR BIN MOKHTAR   [CB20015]
5. WAN MOHD IKRAM BIN WAN MAHADI   [CB19085]
6. WALEED AHMED SALEH ALAMRI   [CB16177]

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Purpose

The purpose of this document is to provide a detail description of the design of a PETAKOM Management System. It is to include a detailed of the requirement of design of PETAKOM Management System and to assist the development team with good understanding on how the product is expected to be delivered. This document provides and describes an overview of the entire software and system with purpose, module overview, and references of the PETAKOM Management system that need to be developed.

1.2 Module Overview

The PETAKOM Management System have 6 modules which are:

- Manage User Registration
- Login
- Manage PETAKOM activities
- Manage calendar
- Generate report and activity proposal
- Manage committee election
- Manage PETAKOM and faculty bulletin

The system allows users to make a registration for the coordinator. Coordinator can register their account by entering their staff ID, email and password. Then, all users (Student, Lecturer, Dean, Coordinator, PETAKOM Committee, and Head of Student Development) need to log in to their account by using their ID and password registered as E-Community UMP System. As for the user profile, users can update, view and edit their profile such as change name, address, and mobile phone number. There is also a forgot password function where users can reset their password. Users need to enter their personal email address and the system will send the reset link to the email entered.

The second module is about managing the PETAKOM activities in the system. Students, lecturers and PETAKOM committee can propose their desired activities through the Activities

tab in the landing page. From here, they can fill all the necessary information about the activities that they proposed such as date and time, club name, advisor club, organizer, type of application (individual or club), type of activity (social interaction, festival, competition,etc..), total participants, venue, activity duration and budget. For an exception, only the PETAKOM committee can update and delete the activity proposal while others can only create and view the activity proposal. After completing the form, they can submit the form by clicking on the Submit button. Next, the proposed activities must be approved by the coordinator, dean and Head of Student Development.

Third module is about managing the calendar in the system. The PETAKOM committee updates semester dates, final exams, public holidays, annual activities, and adds and deletes dates. In addition to adjusting the dates after the approval of the proposed activities by the coordinator, dean, and head of student development, students and lecturers can follow the activities and dates of the events.

Fourth module will be focused on generating the report for the system. This module will be used by multiple users, such as the Coordinator, PETAKOM Committee, the Dean of the faculty, and the Head of the Student Development. This module will focus on the reporting aspects of the event by generating a feedback form that can be submitted by the participants of the event that is handled by the PETAKOM organisation. The PETAKOM Committee will be the one that created the form first and can be seen by the Coordinator, Dean, and the Head of Student Development. The PETAKOM Committee are the only users that can create and update the form.

Module number 5 enables students to conduct votes and elections securely. Students are empowered to suggest who should be the representative for the Petakom committee. The voting system will protect the integrity of our votes by prohibiting voters from voting multiple times. A secure voting tool that allows the Petakom committee and Coordinator to collect input from the election results and closely examine the results in real-time. The Head of the Student Development will approve the new representative after the voting session. In conclusion, students are participating in the democratic process by voting. Students elect leaders to represent them and their ideas, and the leaders advocate on their behalf.

The sixth module focuses on delivering the most recent PETAKOM and faculty bulletin

information. PETAKOM has a great deal of new material that must be shared with students and lecturers. In order to effectively convey the most recent information with students and lecturers, the system provides a platform for the committee to write and display bulletins on the homepage. The system provides a bulletin form that the committee must complete and confirm for publication. A list of created bulletins is provided for the committee to review, and they have the ability to update and delete bulletins at any time. On the homepage for students and lecturers, the system displayed all committee-created bulletins. Therefore, students and lecturers can readily get the most recent information regarding PETAKOM and faculty bulletin.

2. REFERENCE DOCUMENT

How Laravel implements MVC and how to use it effectively. (2018, May 24). Blog - Pusher. Retrieved from https://blog.pusher.com/laravel-mvc-use/

Figma: the collaborative interface design tool. (n.d.). Figma. Retrieved from https://www.figma.com/

PETAKOM. (n.d.). Faculty of Computing. Retrieved from https://fk.ump.edu.my/index.php/en/others/staff-student/petakom

Flowchart Maker & Online Diagram Software. (n.d.). Flowchart Maker & Online Diagram Software. Retrieved from https://app.diagrams.net/

Laravel - The PHP Framework For Web Artisans. (n.d.). Laravel - the PHP Framework for Web Artisans. Retrieved from https://laravel.com/

etc, W. D. (2022, October 23). *Laravel Naming Conventions - Web dev etc - my software development blog*. Laravel Naming Conventions - Web Dev Etc - My Software Development Blog. Retrieved from https://webdevetc.com/blog/laravel-naming-conventions/

3. GENERAL ARCHITECTURE

3.1 Application Layer
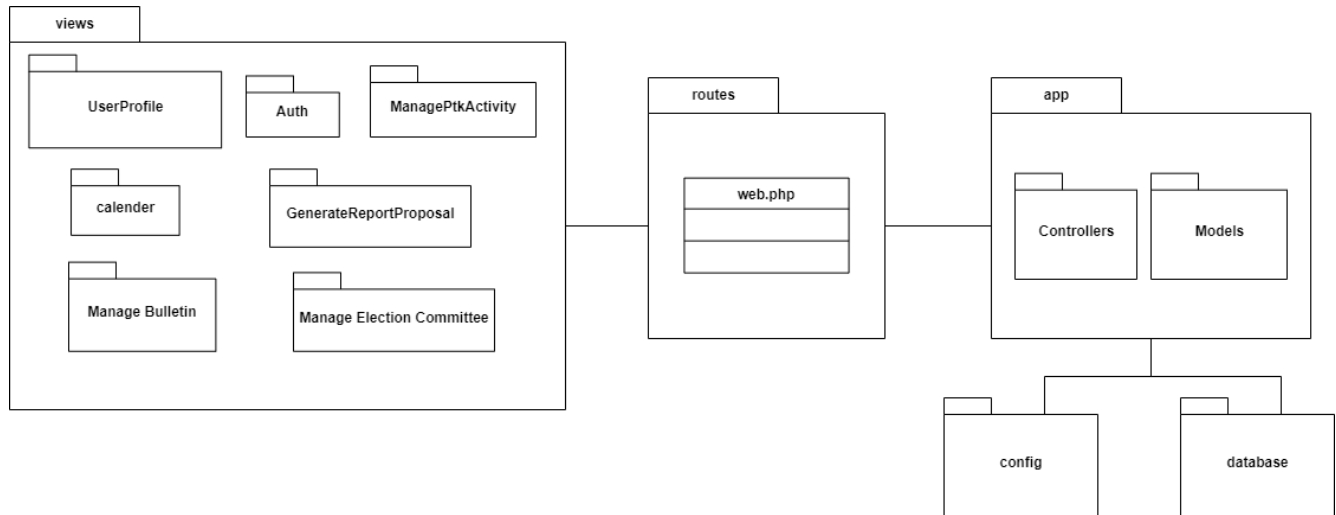
3.2 3-Tier Architecture Layer

3.1.1. Auth
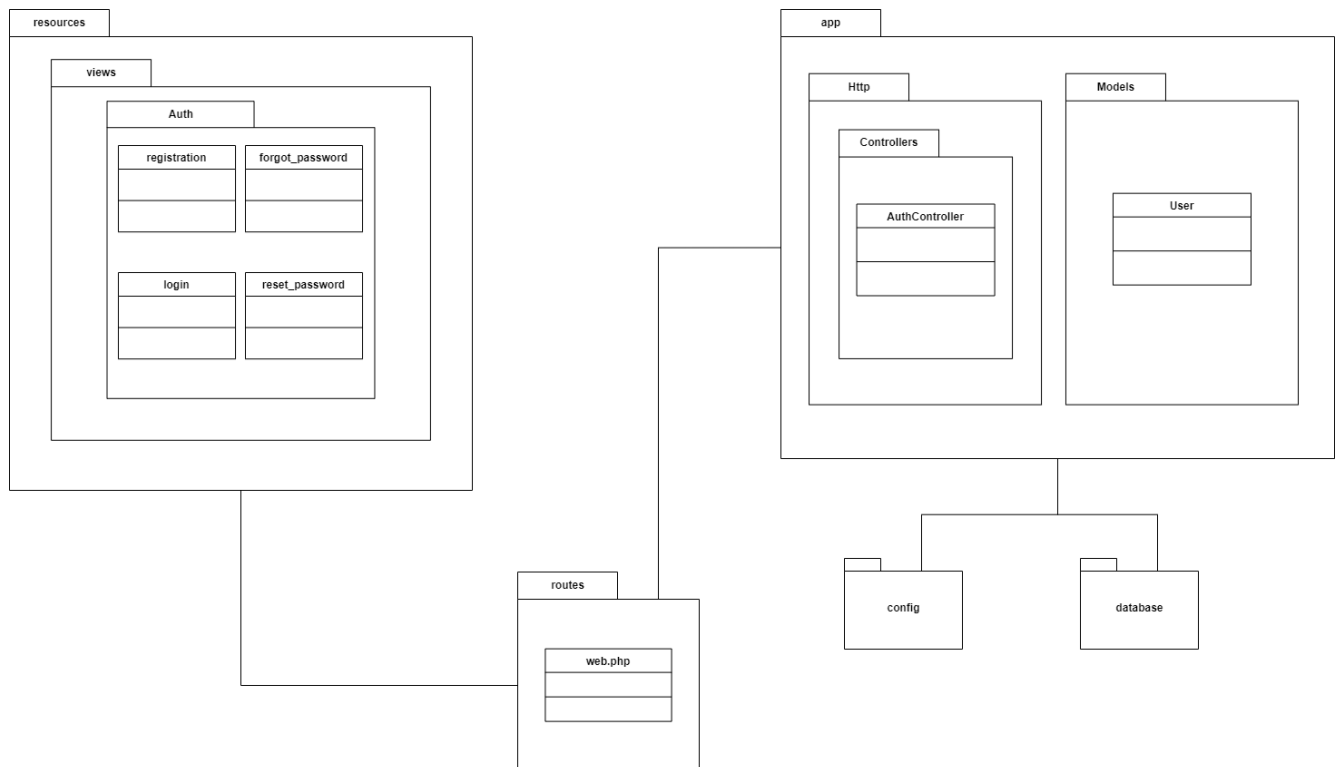
3.1.1.1 AuthInterface

Class Name	Description
registration	This interface allows coordinator to register their account by entering staff id, email and password.
forgot_passwrod	This interface allows user to get reset password link to their email by entering their email in the box given.
login	This interface allows user to login to their account by entering their id and password.
reset_password	This interface allows user to login to their account by entering their id and password.

3.1.1.2 App

Class Name	Description
AuthController	Controller that is used to manage registration function for registration interface.
User	Database that is used to create, retrieve, update or delete records of user record.

3.1.2. UserProfile
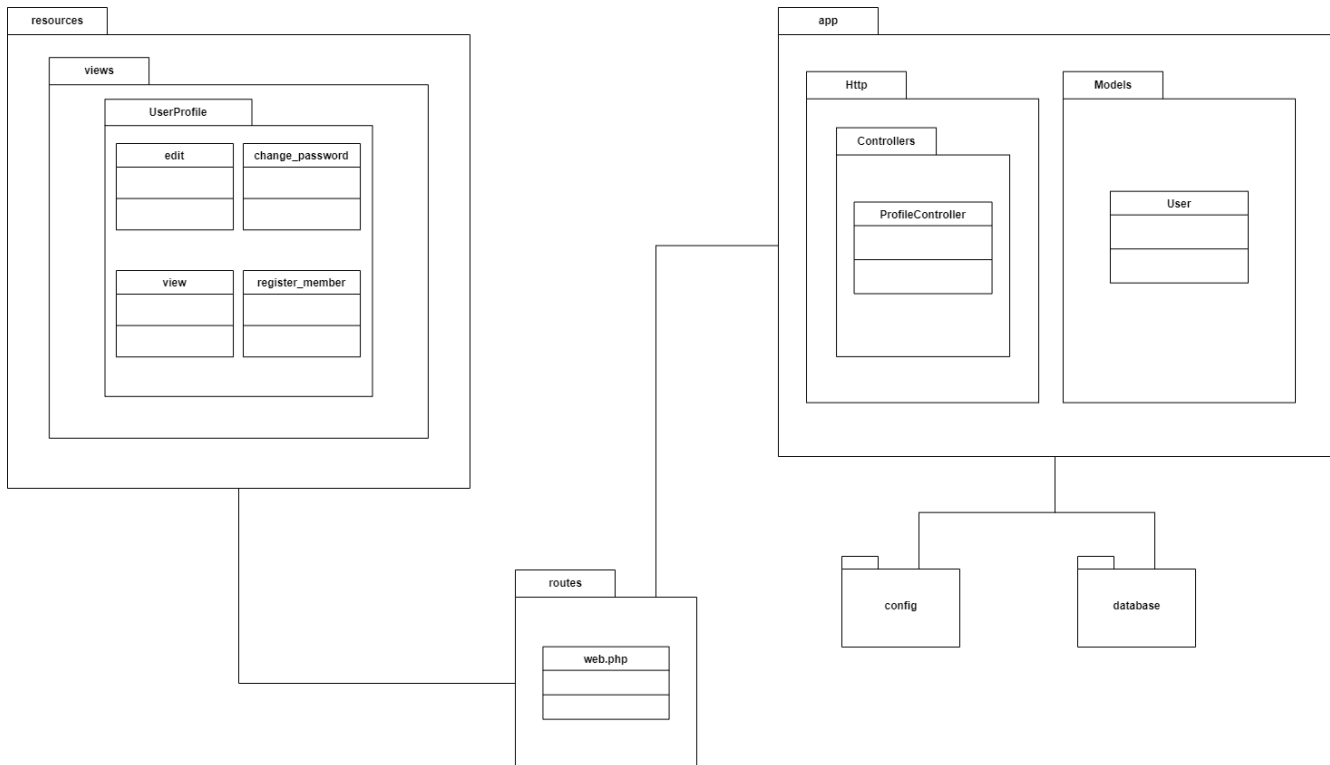
3.1.2.1 LoginInterface

Class Name	Description
edit	This interface allows user to update their name, phone number, email and address.
view	This interface allows user to view their detail.
register_member	This interface allows coordinator to register new member of PETAKOM Committee.
change_password	This interface allows user to change their password.

3.1.2.2 App

Class Name	Description
ProfileController	Controller that is used to manage function for the profile interface.
User	Database that is used to create, retrieve, update or delete records of user record.

3.1.3. Manage PETAKOM Activities
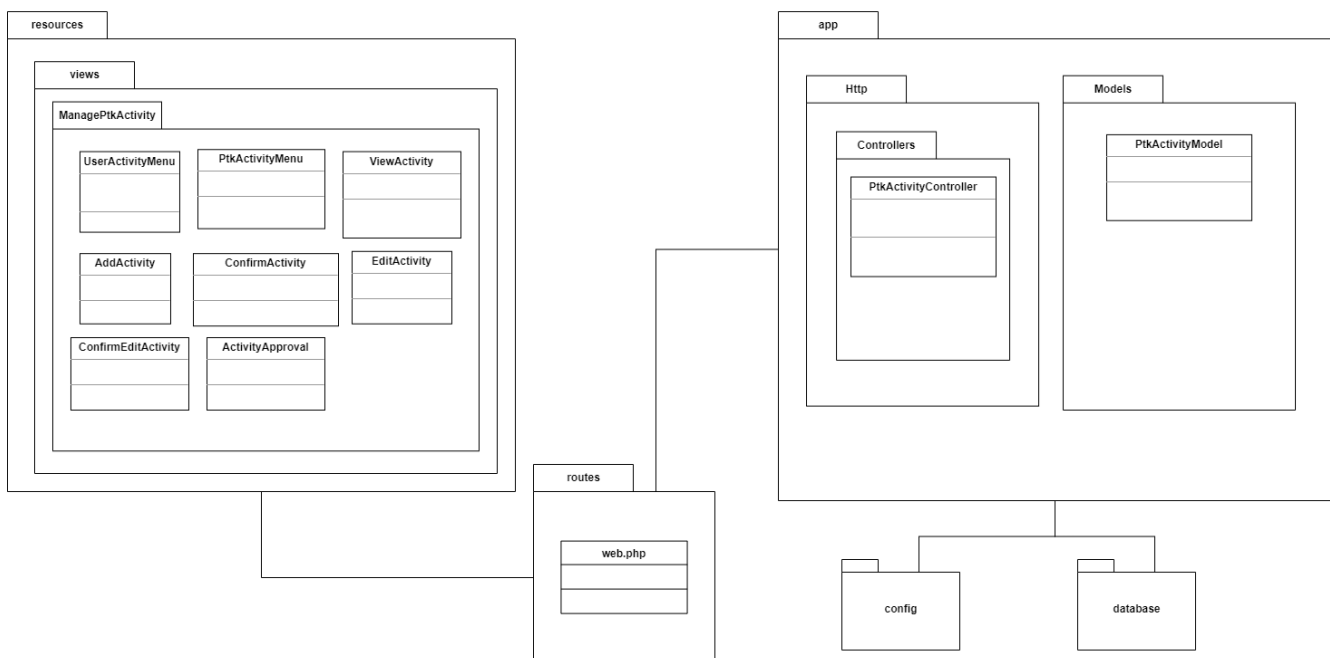
3.1.3.1 ManagePtkActivityInterface

Class Name	Description
UserActivityMenu	This interface allows user to view lists of PETAKOM and faculty activities.

PtkActivityMenu	This interface allows PETAKOM committee to view lists of activities, delete activities, start adding new activity, and edit any activity from the activity list.
ViewActivity	This interface allows user and PETAKOM committee to view selected activity.
AddActivity	This interface allows users and PETAKOM committee to add new activities.
ConfirmAddActivity	This interface allows users and PETAKOM committee to make confirmation on adding new activities.
EditActivity	This interface allows PETAKOM committee to edit selected activities.
ConfirmEditActivity	This interface allows users and PETAKOM committee to make confirmation on editing selected activities.
ActivityApproval	This interface allows dean, coordinator and Head of Student Department to make an approval on selected activities.

3.1.3.2 App

Class Name	Description
PtkActivityController	Controller that is used to manage PETAKOM activity function for PETAKOM Activity interface.
PtkActivityModel	Database that is used to create, retrieve, update or delete records of PETAKOM activity record.

3.1.4. Manage PETAKOM Calendar
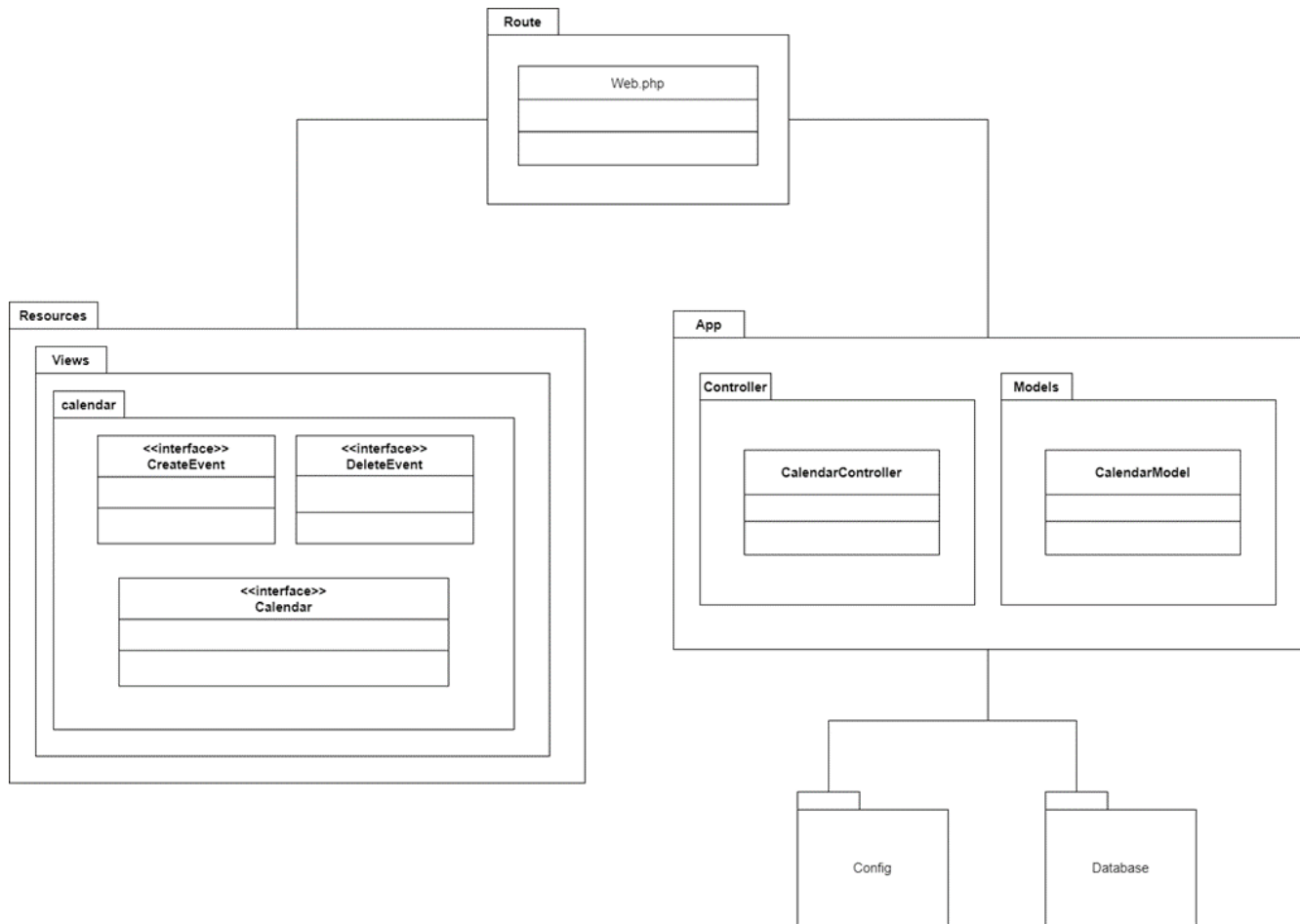
3.1.4.1 ManageCalendarInterface

Class Name	Description
Calendar	This class shows an event calendar with associated information.
CreateEvent	The Petakom committee can establish an event and add information using this class for a certain time period.
DeleteEvent	The Petakom committee may delete an event using this class.

3.1.4.2 App

Class Name	Description
CalendarController	Manage the calendar interfaces and communicate with the CalendarModel.
CalendarModel	Manage the calendar module's data between the system and the database.

3.1.5. Generate Report and Activity Proposal
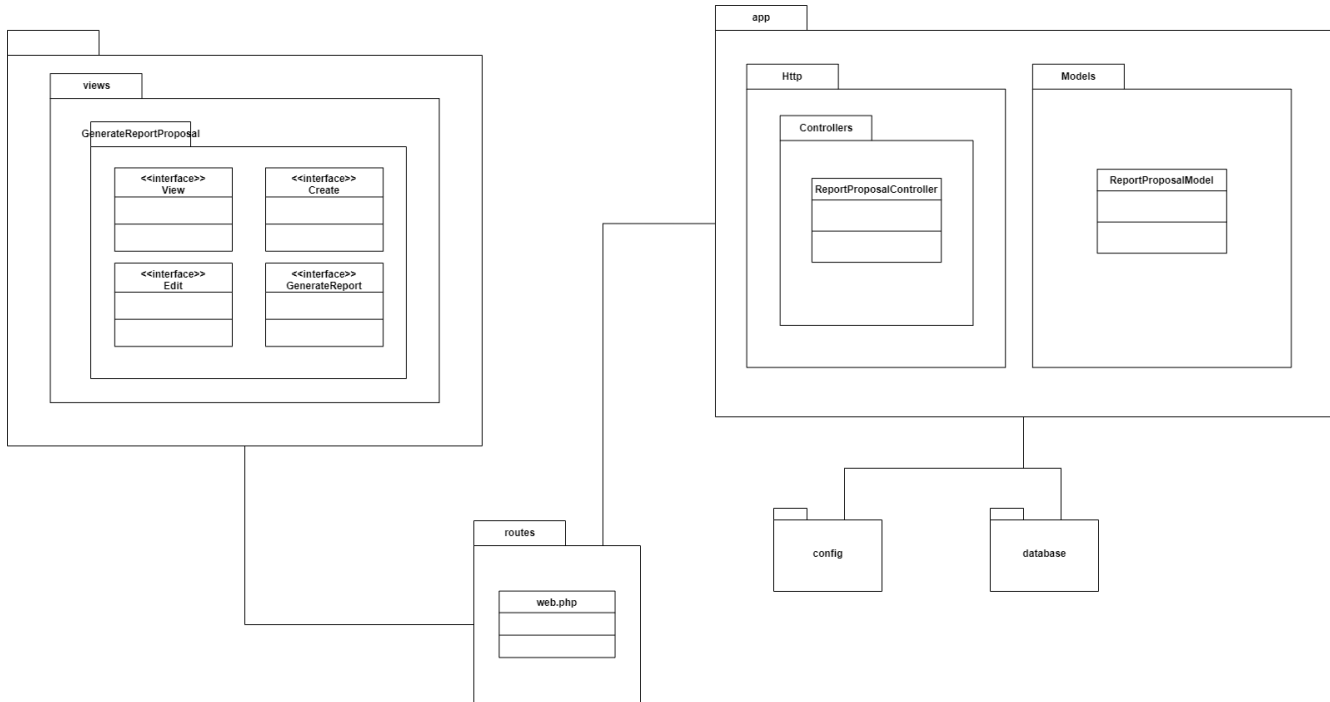
3.1.5.1 GenerateReportProposalInterface

Class Name	Description
View	This interface allows the Petakom Committee to view multiple forms that are created for each event.
Create	This interface allows the Petakom Committee to create the form for each event that is created.
Edit	This allows the Petakom Committee to edit the form.
GenerateReport	This allows all users to view the form's results for each event.

3.1.5.2 App

Class Name	Description
ReportProposalController	Controller that manages the model and the interface.
ReportProposalModel	Model that retrieves the data from the controller.

3.1.6. Manage Committee election
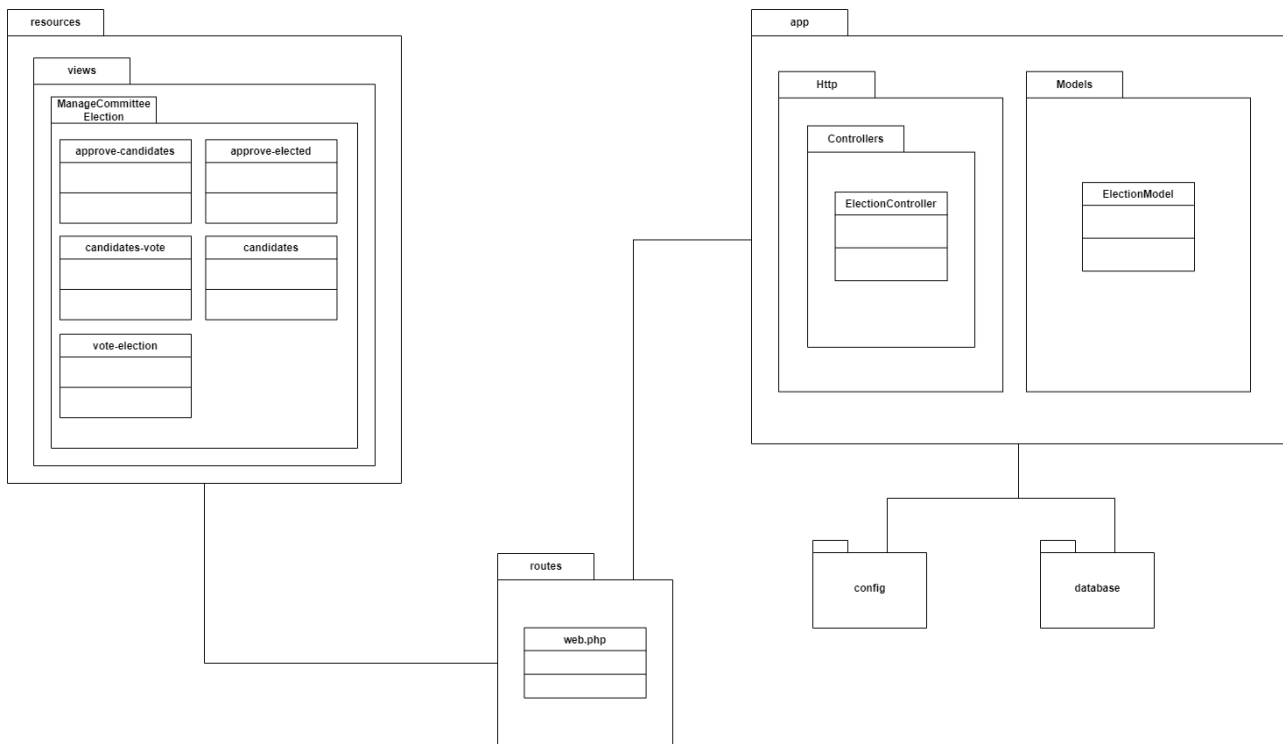
3.1.6.1 ManageElectionCommittee

Class Name	Description
candidates	This interface allows students to suggest suitable candidates become members of PETAKOM
candidate-votes	An interface that allows students to vote for PETAKOM candidates who are competing
approve-candidates	An interface that displays a list of candidate names suggested by students
vote-election	An interface that displays voting information and is responsible for vote counting
approve-elected	An interface that allows Hosd and Coordinator to approve the election winner.

3.1.6.2 App

Class Name	Description
ElectionController	Controller that is used to manage the function for manage election committee interface.
ElectionModel	To store and retrieve information from the manage election committee function

3.1.7. Manage PETAKOM and Faculty Bulletin
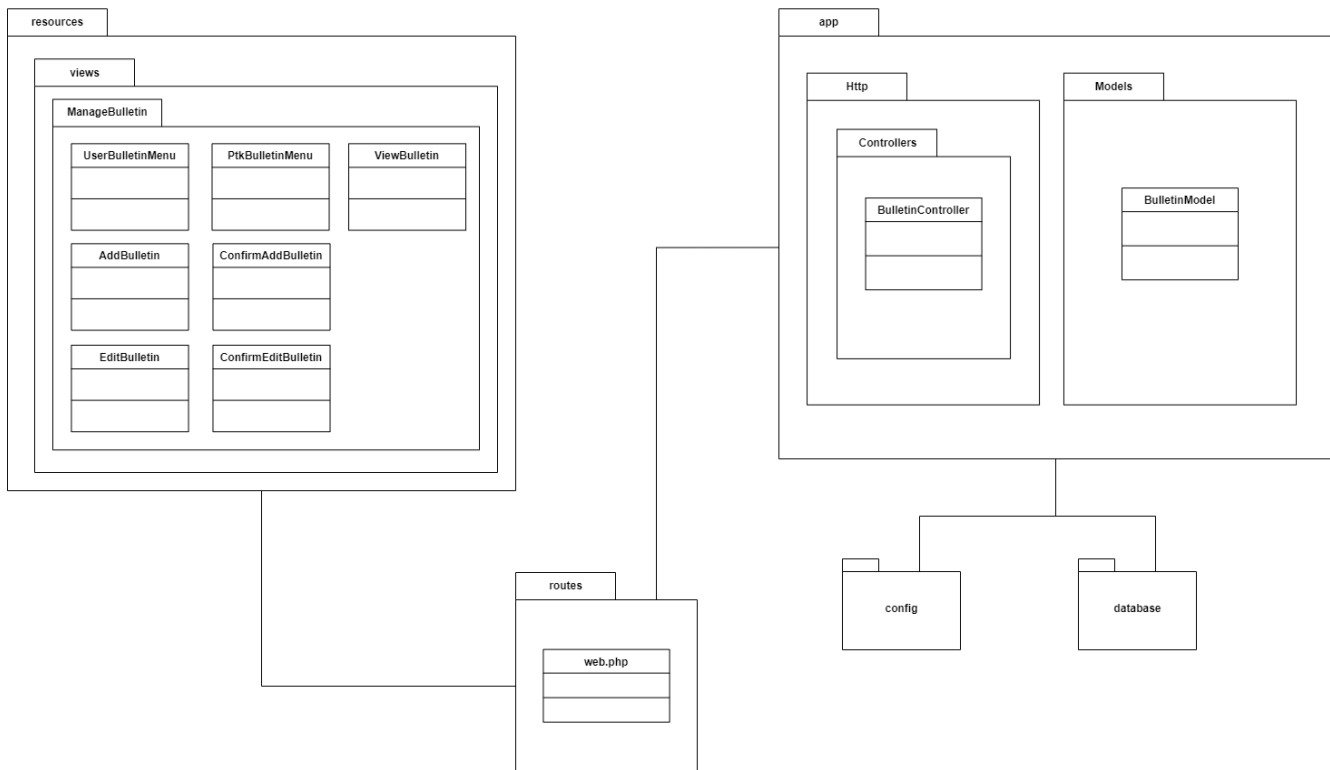
3.1.7.1 ManageBulletinInterface

Class Name	Description
UserBulletinMenu	This interface allows user to view lists of PETAKOM and faculty bulletins.
PtkBulletinMenu	This interface allows PETAKOM committee to view lists of bulletins, delete bulletin, start adding new bulletin, and edit any bulletin from the bulletin list.
ViewBulletin	This interface allows user and PETAKOM committee to view selected bulletin.
AddBulletin	This interface allows PETAKOM committee to add new bulletin.
ConfirmAddBulletin	This interface allows PETAKOM committee to make confirmation on adding new bulletin.
EditBulletin	This interface allows PETAKOM committee to edit selected bulletin.
ConfirmEditBulletin	This interface allows PETAKOM committee to make confirmation on editing selected bulletin.

3.1.7.2 App

Class Name	Description

BulletinController	Controller that is used to manage bulletin function for bulletin interface.
BulletinModel	Database that is used to create, retrieve, update or delete records of bulletin record.

3.1.2 Framework
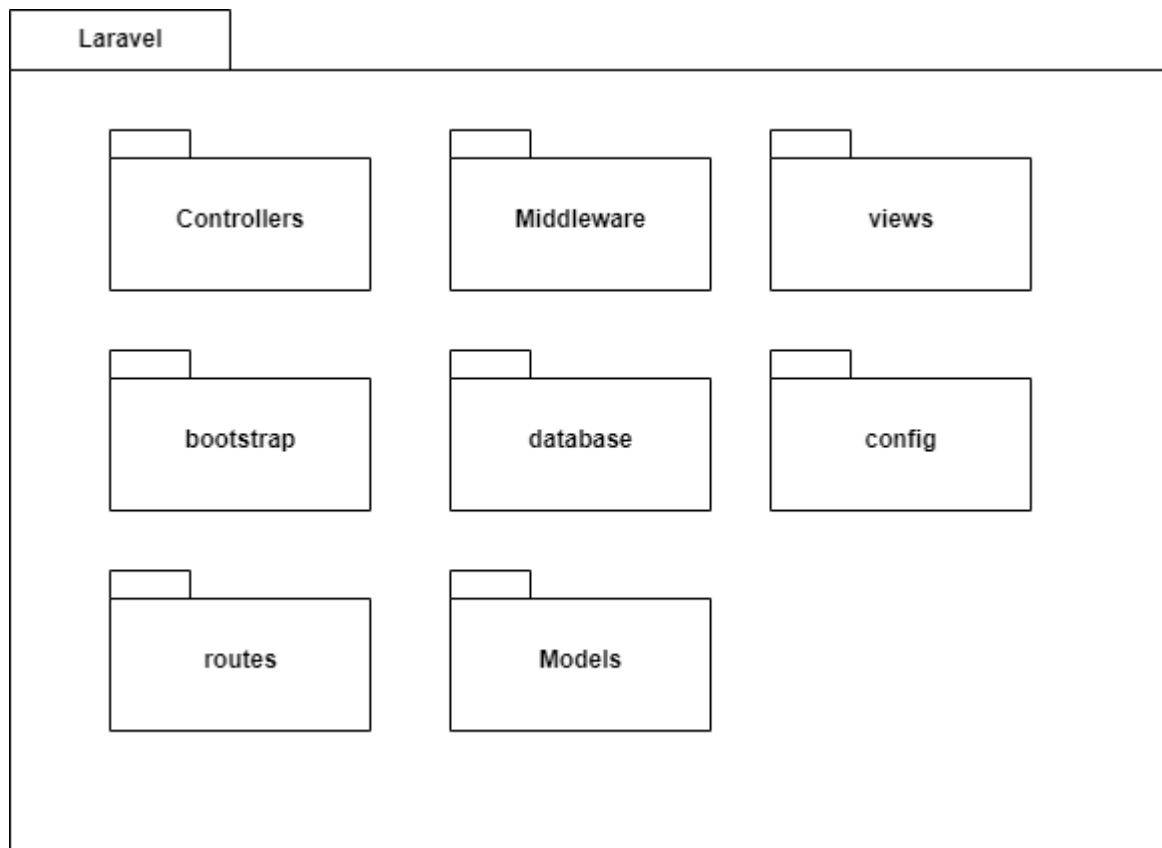
3.2. Package Module
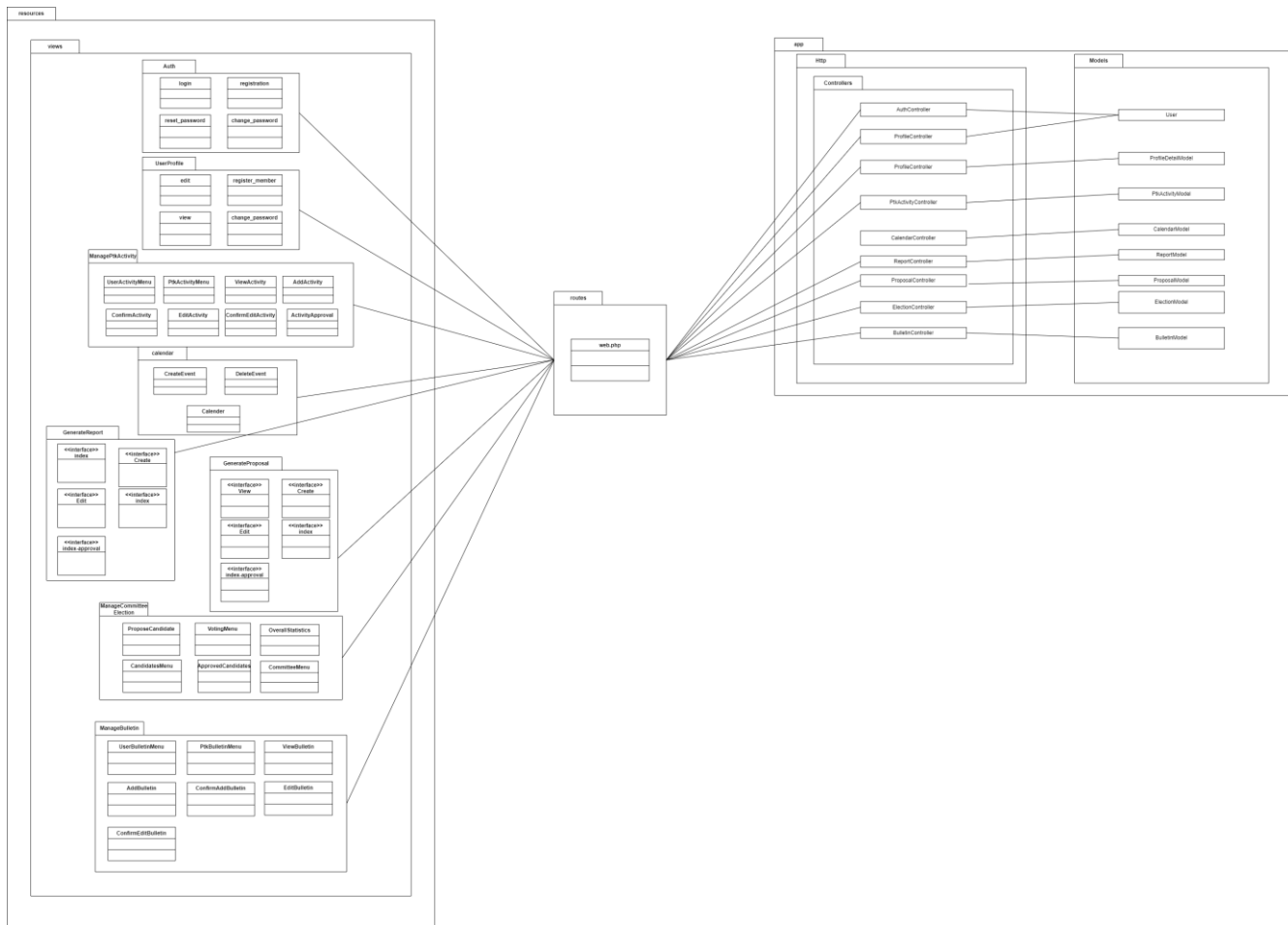
3.3. Entity Relationship Diagram (ERD)
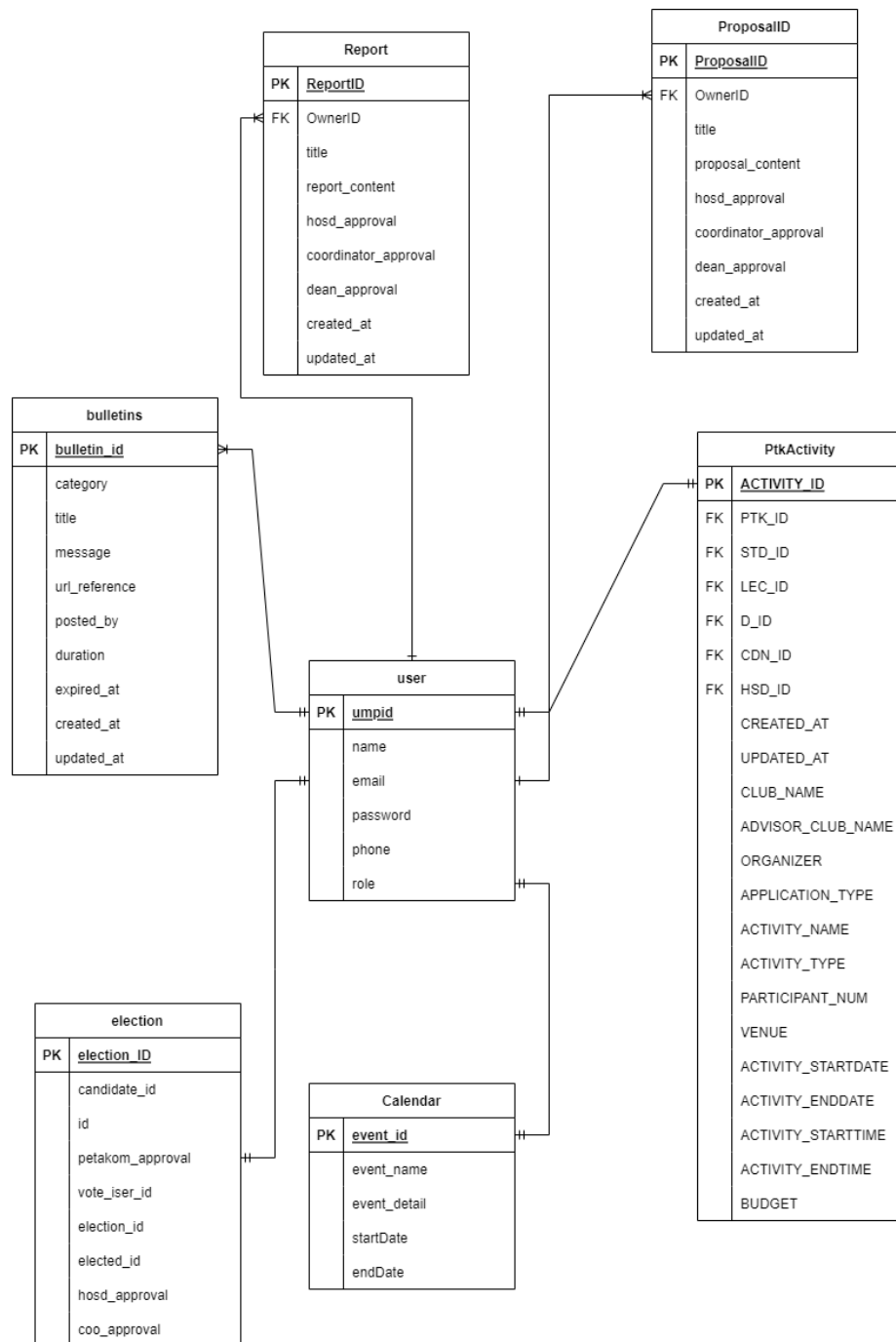
3.4. Data Dictionary

3.4.1. User

Field Name	Description	Data Type	Constraint
umpid	User ump id	VARCHAR (10)	PK
name	User name	VARCHAR (50)	
email	User email	VARCHAR (100)	
password	User password	VARCHAR (50)	
phone	User phone number	VARCHAR (15)	
role	User role	VARCHAR (25)	

3.4.2. Bulletin

Field Name	Description	Data Type	Constraint
bulletin_id	Bulletin id	INT	PK
category	Bulletin category	VARCHAR (255)	
title	Bulletin title	VARCHAR (255)	FK
message	Bulletin message	VARCHAR (255)	
url_reference	Bulletin url reference	VARCHAR (255)	
posted_by	PETAKOM committee name for add and edit bulletin	VARCHAR (255)	
duration	Bulletin duration	INT	
expired_at	Bulletin expired date	DATE	
created_at	Bulletin creation date	TIMESTAMP	
updated_at	Bulletin updated date	TIMESTAMP	

3.4.3. PtkActivity

Field Name	Description	Data Type	Constraint
ACTIVITY_ID	Activity ID	INT	PK
umpid	User id	VARCHAR (10)	FK
CREATED_AT	Activity creation date	DATE	
UPDATED_AT	Activity updated date	DATE	
CLUB_NAME	Club name	VARCHAR (50)	
ADVISOR_CLUB_NAME	Advisor's club name	VARCHAR (50)	

ORGANIZER	Organizer name	VARCHAR (50)	
APPLICATION_TYPE	Application type (individual or club)	VARCHAR (20)	
ACTIVITY_NAME	Activity name	VARCHAR (50)	
ACTIVITY_TYPE	Activity type	VARCHAR (20)	
PARTICIPANT_NUM	Number of participants	INT	
VENUE	Venue	VARCHAR (50)	
ACTIVITY_DURATION	Activity duration	VARCHAR (50)	
BUDGET	Activity budget	INT	

3.4.4. ManageCommitteeElection

Field Name	Description	Data Type	Constraint
election_id	Election id	NVARCHAR(5)	PK
vote_user_id	Voters information	NVARCHAR(5)	
candidate_id	Candidates information	NVARCHAR(5)	
petakom_approval	Petakom approval	BOOLEAN	
vote_user_id	Voting information	NVARCHAR(5)	
elected_id	Elected candidates information	NVARCHAR(5)	
hosd_approval	Head of Student Development approval	BOOLEAN	
coo_approval	Coordinator approval	BOOLEAN	

3.4.5. Calendar

Field Name	Description	Data Type	Constraint
event_id	Event ID	VARCHAR (10)	PK
event_name	Event Name	VARCHAR (50)	
event_detail	Details of The Event	VARCHAR (50)	
startDate	Start Date of Event	DATE	
endDate	End Date of Event	DATE	

3.4.6. Proposals

Field Name	Description	Data Type	Constraint
ProposalID	Primary key for proposal	VARCHAR(10)	PK
OwnerID	Foreign key for user table	VARCHAR(10)	FK
title	Title of the proposal	LONGTEXT	
proposal_content	Content of proposal	INT	
hosd_approval	Head of student development approval	INT	
coordinator_approval	Coordinator approval	INT	
dean_approval	Dean approval	INT	
created_at	timestamp created	TIMESTAMP	
updated_at	timestamp updated	TIMESTAMP	

3.3.7. Reports

Field Name	Description	Data Type	Constraint
ReportID	Primary key for report	VARCHAR(10)	PK
OwnerID	Foreign key for user table	VARCHAR(10)	FK
title	Title of the report	LONGTEXT	
report_content	Content of report	INT	
hosd_approval	Head of student development approval	INT	
coordinator_approval	Coordinator approval	INT	
dean_approval	Dean approval	INT	
created_at	timestamp created	TIMESTAMP	
updated_at	timestamp updated	TIMESTAMP	

4. DETAIL DESIGN

4.1   Auth [SDD-REQ-100] (HAZMI HAZIM CB20117)
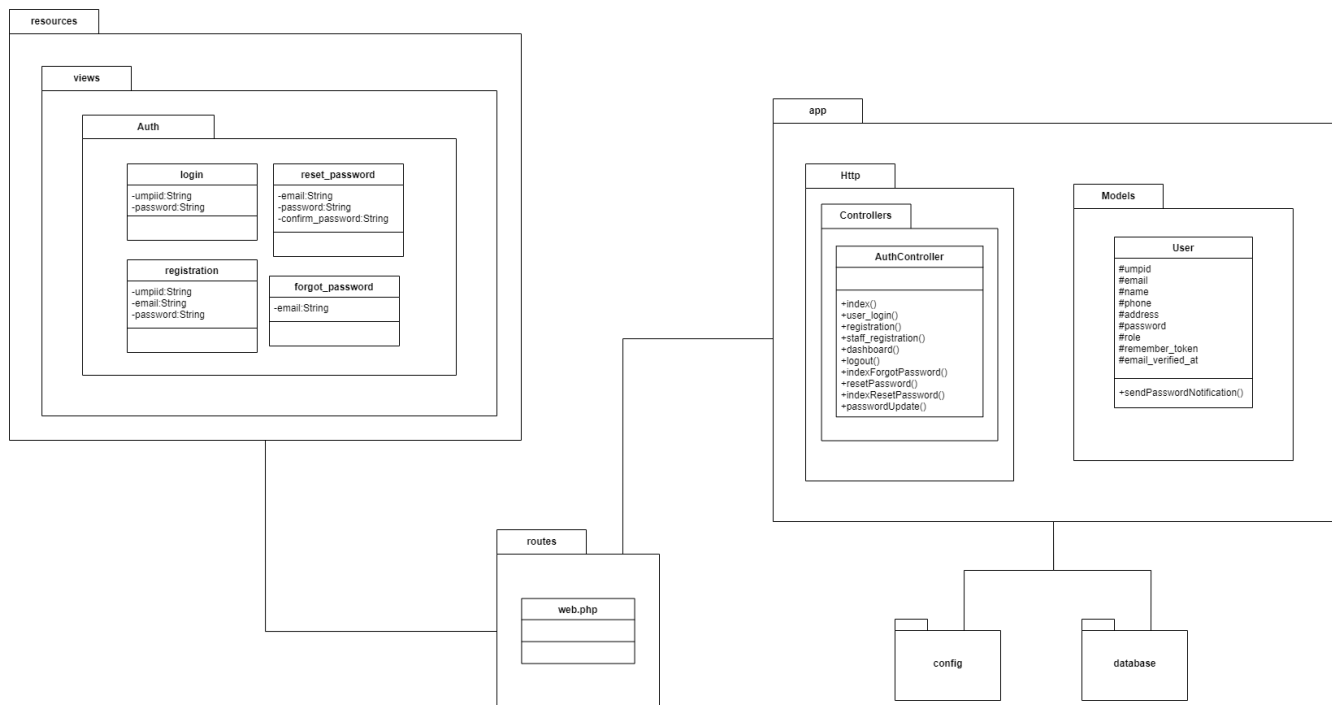
4.1.1 login [SDD-REQ-101]

Class Type	Boundary Class	
Responsibility	This class is responsible to display the login interface.	
Attributes	**Attributes Name**	**Attributes Type**
	umpid	String
	password	String
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.1.2 reset_password [SDD-REQ-102]

Class Type	Boundary Class	
Responsibility	This class is responsible to display the registration interface.	
Attributes	**Attributes Name**	**Attributes Type**
	email	String
	password	String

	Confirm_password	String
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.1.3 registration [SDD-REQ-103]

Class Type	Boundary Class	
Responsibility	This class is responsible to display the registration interface.	
	Attributes Name	**Attributes Type**
	umpid	String
Attributes	Email	String
	passowd	String
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.1.4 forgot_password [SDD-REQ-104]

Class Type	Boundary Class	
Responsibility	This class is responsible to display the forgot password interface.	
Attributes	**Attributes Name**	**Attributes Type**
	email	String
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.1.5 AuthController [SDD-REQ-105]

Class Type	Boundary Class	
Responsibility	This class is responsible to connect the registration interface with the User Model.	
Attributes	**Attributes Name**	**Attributes Type**
	Not applicable	Not applicable
Methods	**Method Name**	**Description**

	+index()	This function responsible to call login interface to be see by the user.
	+user_login()	This function responsible to validate input of user id and user password.
	+registration()	This function responsible to call registration interface to be see by the user.
	+staff_registration()	This function responsible create coordinator account by get input of coordinator id, email and password.
	+dashboard()	This function responsible to call dashboard interface to be see by the user.
	+logout()	This function responsible to end session of the user.
	+indexForgotPassword	This function responsible to call forgot password interface to be see by the user.
	+resetPassword()	This function responsible to send notification to the email input by the user.
	+indexResetPassword()	This function responsible to call reset password interface to be see by the user.
	+passwordUpdate()	This function responsible to update password of the user by new password.
Algorithm	**index()** START RETURN to login interface	

END

user_login()

START

   READ umpid, password

   If umpid != null && password != null

      REDIRECT to dashboard interface

   Else

      RETURN message

END

registration()

START

   REDIRECT to registration interface

END

staff_registration()

START

   READ umpid, email, password

   If email = null

     If password >= 6

       CREATE user account

     Else

       RETURN message

   Else

     RETURN message

END

darshboard()

START

   REDIRECT to dashboard

END

logout()

START

  Session flush

  REDIRECT login

END

indexForgotPassword()

START

  REDIRECT forgot_password

END

resetPassword()

START

  If email != null

    SEND password reset link

    RETURN message

  Else

    RETURN message

END

indexResetPassword()

START

  REDIRECT to reset_password

END

passwordUpdate()

START

  READ email, password, confirm_password

  If password >= 6

    If Confirm_password == password

      Update password

    Else

      RETURN message

	Else
	RETURN message
	END

4.1.6 User [SDD-REQ-106]

Class Type	Boundary Class	
Responsibility	This class is responsible to store data of the user.	
Attributes	**Attributes Name**	**Attributes Type**
	#umpid	String
	#email	String
	#name	String
	#phone	String
	#address	String
	#password	String
	#role	String
	#remember_token	String
	#email_verified_at	String
Methods	**Method Name**	**Description**
	+sendPasswordNotification()	This function responsible to set token to be send to email reset password link.
Algorithm	**sendPasswordNotification()**	
	START	
	GET token	
	SEND to email user	
	END	

4.2. UserProfile [SDD-REQ-200] (HAZMI HAZIM CB20117)
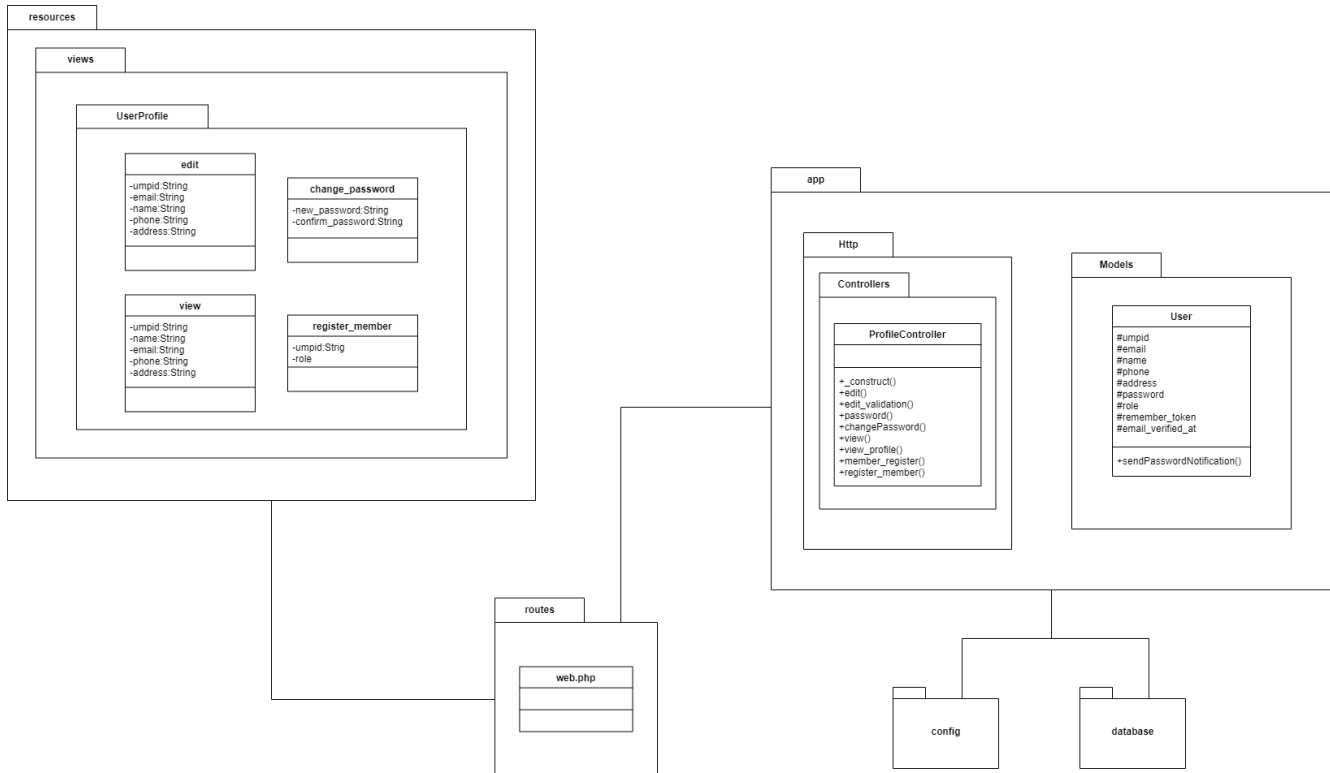
4.2.1 view [SDD-REQ-201]

Class Type	Boundary Class	
Responsibility	This class is responsible to display the view user profile interface.	
Attributes	**Attributes Name**	**Attributes Type**
	umpid	String
	email	String
	name	String
	phone	String
	address	String
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.2.2 edit [SDD-REQ-202]

Class Type	Boundary Class
Responsibility	This class is responsible to display the edit user profile interface.

Attributes	Attributes Name	Attributes Type
	umpid	String
	email	String
	name	String
	phone	String
	address	String
Methods	Method Name	Description
	Not applicable	Not applicable
Algorithm	Not applicable	

4.2.3 change_password [SDD-REQ-203]

Class Type	Boundary Class	
Responsibility	This class is responsible to display the change password interface.	
Attributes	Attributes Name	Attributes Type
	new_password	String
	confirm_password	String
Methods	Method Name	Description
	Not applicable	Not applicable
Algorithm	Not applicable	

4.2.4 register_member [SDD-REQ-204]

Class Type	Boundary Class	
Responsibility	This class is responsible to display the register member of PETAKOM committee interface.	
Attributes	Attributes Name	Attributes Type
	umpid	String
	role	String
Methods	Method Name	Description
	Not applicable	Not applicable
Algorithm	Not applicable	

4.2.5 ProfileController [SDD-REQ-205]

Class Type	Boundary Class	
Responsibility	This class is responsible to connect the user profile interface with the User model.	
Attributes	**Attributes Name**	**Attributes Type**
	Not applicable	Not applicable
Methods	**Method Name**	**Description**
	+_construct()	This function responsible to identify the role of user.
	+edit()	This function responsible to show edit interface of user profile.
	+edit_validation()	This function responsible to validate and update email, name, phone, and address user.
	+password()	This function responsible to show change password interface.
	+changePassword()	This function responsible to update new password user.
	+view()	This function responsible to show view interface of user profile.
	+view_profile()	This function responsible to get user id, email, name, phone and address to view in view interface.
	+member_register()	This function responsible to register new member of PETAKOM committee.
	+member_register()	This function responsible to show register_member interface.
Algorithm	**_construct()**	

```
START
    GET auth
END


edit()
START
    REDIRECT edit
END


edit_validation()
START
    GET email, name, phone, address
    RETURN message
END


password()
START
    REDIRECT change_password
END


change_password()
START
    If new_password >= 6
        If confirm_password = new_password
            UPDATE password
        Else
            RETURN message
    Else
        RETURN message
END


view()
START
```

	REDIRECT view
	END
	
	view_profile()
	START
	    GET umpid, email, name, phone, address
	END
	
	member_register()
	START
	    GET umpid, role
	    CREATE new member account
	    RETURN message
	END
	
	register_member()
	START
	    REDIRECT register_member
	END

4.2.6 User [SDD-REQ-106]

Class Type	Boundary Class	
Responsibility	This class is responsible to store data of the user.	
	Attributes Name	**Attributes Type**
	#umpid	String
	#email	String
	#name	String
	#phone	String
Attributes	#address	String
	#password	String
	#role	String
	#remember_token	String
	#email_verified_at	String

	Method Name	Description
Methods	+sendPasswordNotification()	This function responsible to set token to be send to email reset password link.

Algorithm	**sendPasswordNotification()**
	START
	   GET token
	   SEND to email user
	END	

4.3. Manage PETAKOM Activities [SDD-REQ-300] (WAN MOHD IKRAM HAMDI BIN WAN MAHADI CB19085)
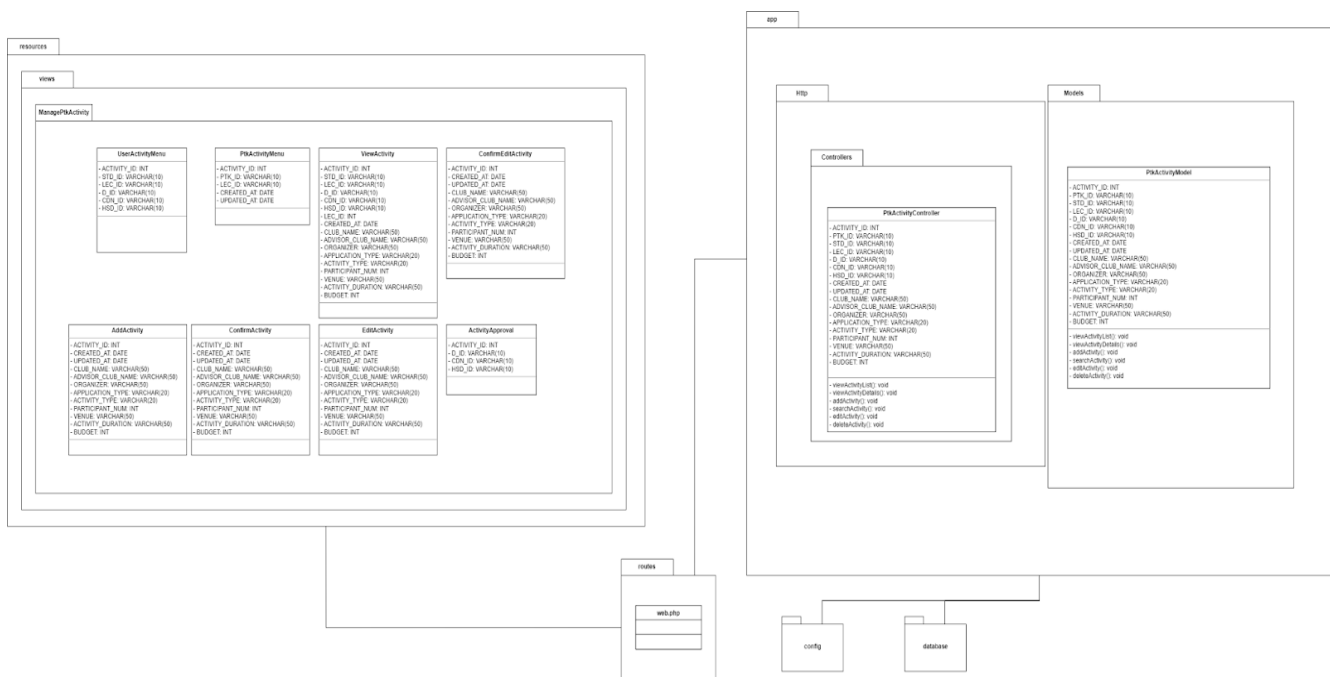
4.3.1 UserActivityMenu [SDD-REQ-301]

Class Type	Boundary Class	
Responsibility	This class is responsible for displaying the user activity menu interface.	
	Attributes Name	**Attributes Type**
Attributes	ACTIVITY_ID	INT
	STD_ID	VARCHAR(10)
	LEC_ID	VARCHAR(10)

	D_ID	VARCHAR(10)
	CDN_ID	VARCHAR(10)
	HSD_ID	VARCHAR(10)
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.3.2 PtkActivityMenu [SDD-REQ-302]

Class Type	**Boundary Class**	
Responsibility	This class is responsible for displaying the PETAKOM committee activity menu interface.	
	Attributes Name	**Attributes Type**
Attributes	ACTIVITY_ID	INT
	PTK_ID	VARCHAR(10)
	CREATED_AT	DATE
	UPDATED_AT	DATE
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.3.3 ViewActivity [SDD-REQ-303]

Class Type	**Boundary Class**	
Responsibility	This class is responsible for displaying the selected activity details interface.	
	Attributes Name	**Attributes Type**
Attributes	ACTIVITY_ID	INT
	STD_ID	VARCHAR(10)
	LEC_ID	VARCHAR(10)
	D_ID	VARCHAR(10)

	CDN_ID	VARCHAR(10)
	HSD_ID	VARCHAR(10)
	CREATED_AT	DATE
	CLUB_NAME	VARCHAR(50)
	ADVISOR_CLUB_NAME	VARCHAR(50)
	ORGANIZER	VARCHAR(50)
	APPLICATION_TYPE	VARCHAR(20)
	ACTIVITY_TYPE	VARCHAR(20)
	PARTICIPANT_NUM	INT
	VENUE	VARCHAR(50)
	ACTIVITY_DURATION	VARCHAR(50)
	BUDGET	INT
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.3.4 AddActivity [SDD-REQ-304]

Class Type	**Boundary Class**	
Responsibility	This class is responsible for displaying the add activity interface.	
	Attributes Name	**Attributes Type**
	ACTIVITY_ID	INT
	CREATED_AT	DATE
	UPDATED_AT	DATE
	CLUB_NAME	VARCHAR(50)
	ADVISOR_CLUB_NAME	VARCHAR(50)
Attributes	ORGANIZER	VARCHAR(50)
	APPLICATION_TYPE	VARCHAR(20)
	ACTIVITY_TYPE	VARCHAR(20)
	PARTICIPANT_NUM	INT
	VENUE	VARCHAR(50)
	ACTIVITY_DURATION	VARCHAR(50)
	BUDGET	INT
Methods	**Method Name**	**Description**

	Not applicable	Not applicable
Algorithm	Not applicable	

4.3.5 ConfirmActivity [SDD-REQ-305]

Class Type	Boundary Class	
Responsibility	This class is responsible for displaying the confirmation add activity interface.	
Attributes	**Attributes Name**	**Attributes Type**
	ACTIVITY_ID	INT
	CREATED_AT	DATE
	UPDATED_AT	DATE
	CLUB_NAME	VARCHAR(50)
	ADVISOR_CLUB_NAME	VARCHAR(50)
	ORGANIZER	VARCHAR(50)
	APPLICATION_TYPE	VARCHAR(20)
	ACTIVITY_TYPE	VARCHAR(20)
	PARTICIPANT_NUM	INT
	VENUE	VARCHAR(50)
	ACTIVITY_DURATION	VARCHAR(50)
	BUDGET	INT
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.3.6 EditActivity [SDD-REQ-306]

Class Type	Boundary Class	
Responsibility	This class is responsible for displaying the edit activity interface.	
Attributes	**Attributes Name**	**Attributes Type**
	ACTIVITY_ID	INT
	CREATED_AT	DATE
	UPDATED_AT	DATE
	CLUB_NAME	VARCHAR(50)
	ADVISOR_CLUB_NAME	VARCHAR(50)

	ORGANIZER	VARCHAR(50)
	APPLICATION_TYPE	VARCHAR(20)
	ACTIVITY_TYPE	VARCHAR(20)
	PARTICIPANT_NUM	INT
	VENUE	VARCHAR(50)
	ACTIVITY_DURATION	VARCHAR(50)
	BUDGET	INT
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.3.7 ConfirmEditActivity [SDD-REQ-307]

Class Type	**Boundary Class**	
Responsibility	This class is responsible for displaying the confirmation edit activity interface.	
	Attributes Name	**Attributes Type**
	ACTIVITY_ID	INT
	CREATED_AT	DATE
	UPDATED_AT	DATE
	CLUB_NAME	VARCHAR(50)
	ADVISOR_CLUB_NAME	VARCHAR(50)
Attributes	ORGANIZER	VARCHAR(50)
	APPLICATION_TYPE	VARCHAR(20)
	ACTIVITY_TYPE	VARCHAR(20)
	PARTICIPANT_NUM	INT
	VENUE	VARCHAR(50)
	ACTIVITY_DURATION	VARCHAR(50)
	BUDGET	INT
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.3.8 ActivityApproval [SDD-REQ-308]

Class Type	Boundary Class	
Responsibility	This class is responsible for displaying the approval activity interface.	
Attributes	**Attributes Name**	**Attributes Type**
	ACTIVITY_ID	INT
	PTK_ID	VARCHAR(10)
	CREATED_AT	DATE
	UPDATED_AT	DATE
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.3.9 PtkActivityController [SDD-REQ-309]

Class Type	Boundary Class	
Responsibility	This class is responsible to connect UserActivityMenu interface, PtkActivityMenu interface, ViewActivty interface, AddActivity interface, EditActivity interface, ConfirmAddActivity interface, ConfirmEditActivity interface and ActivityApproval interface with PtkActivityModel.	
Attributes	**Attributes Name**	**Attributes Type**
	ACTIVITY_ID	INT
	PTK_ID	VARCHAR(10)
	STD_ID	VARCHAR(10)
	LEC_ID	VARCHAR(10)
	D_ID	VARCHAR(10)
	CDN_ID	VARCHAR(10)
	HSD_ID	VARCHAR(10)
	CREATED_AT	DATE
	UPDATED_AT	DATE
	CLUB_NAME	VARCHAR(50)
	ADVISOR_CLUB_NAME	VARCHAR(50)
	ORGANIZER	VARCHAR(50)
	APPLICATION_TYPE	VARCHAR(20)

	ACTIVITY_TYPE	VARCHAR(20)
	PARTICIPANT_NUM	INT
	VENUE	VARCHAR(50)
	ACTIVITY_DURATION	VARCHAR(50)
	BUDGET	INT
	Method Name	**Description**
Methods	viewActivityList()	The user and the PETAKOM committee can view the list of activities.
	viewActivityDetails()	The user and the PETAKOM committee can view the selected activity details.
	addActivity()	The user and PETAKOM committee can add an activity.
	searchActivity()	The user and PETAKOM committee can search the activity.
	editActivity()	The PETAKOM committee can edit an activity.
	deleteActivity()	The PETAKOM committee can delete an activity.

Algorithm	**viewActivityList()**
	START
	redirect to ViewActivity interface
	retrieve data from table PtkActivity
	IF PETAKOM committee
	display PtkActivityMenu interface
	ELSE
	display UserActivitylMenu interface
	END IF
	END
	
	viewActivityDetails()	

START

click view button in PtkActivityMenu interface

read data from table bulletin with the chosen bulletin

display data

END

addActivity()

START

addActivity() is called

Create object service to connect to PtkActivityModel

RETURN to PtkActivityMenu interface

END

searchActivity()

START

enter input in search bar in PtkActivityMenu

read data from table PtkActivity with the input data as condition

RETURN to PtkActivityMenu interface

END

editActivity()

START

click delete button in PtkActivityMenu interface

enter club name, advisor club name, organizer, application type, activity type, participant number, venue, activity duration and budget

click confirm button in ConfirmEditActivity interface

alter data in table PtkActivity

display deletion success message

END

deleteActivity()

START

	click delete button in PtkActivityMenu interface
	delete data in table PtkActivity
	display deletion success message
	END

4.3.10 PtkActivityModel [SDD-REQ-310]

Class Type	Boundary Class	
Responsibility	This class is responsible to connect UserActivityMenu interface, PtkActivityMenu interface, ViewActivty interface, AddActivity interface, EditActivity interface, ConfirmAddActivity interface, ConfirmEditActivity interface and ActivityApproval interface with PtkActivityModel.	
Attributes	**Attributes Name**	**Attributes Type**
	ACTIVITY_ID	INT
	PTK_ID	VARCHAR(10)
	STD_ID	VARCHAR(10)
	LEC_ID	VARCHAR(10)
	D_ID	VARCHAR(10)
	CDN_ID	VARCHAR(10)
	HSD_ID	VARCHAR(10)
	CREATED_AT	DATE
	UPDATED_AT	DATE
	CLUB_NAME	VARCHAR(50)
	ADVISOR_CLUB_NAME	VARCHAR(50)
	ORGANIZER	VARCHAR(50)
	APPLICATION_TYPE	VARCHAR(20)
	ACTIVITY_TYPE	VARCHAR(20)
	PARTICIPANT_NUM	INT
	VENUE	VARCHAR(50)
	ACTIVITY_DURATION	VARCHAR(50)
	BUDGET	INT
Methods	**Method Name**	**Description**

	viewActivityList()	The user and the PETAKOM committee can view the list of activities.
	viewActivityDetails()	The user and the PETAKOM committee can view the selected activity details.
	addActivity()	The user and PETAKOM committee can add an activity.
	searchActivity()	The user and PETAKOM committee can search the activity.
	editActivity()	The PETAKOM committee can edit an activity.
	deleteActivity()	The PETAKOM committee can delete an activity.

Algorithm	**viewActivityList()**
	START
	redirect to ViewActivity interface
	retrieve data from table PtkActivity
	IF PETAKOM committee
	display PtkActivityMenu interface
	ELSE
	display UserActivitylMenu interface
	END IF
	END
	
	viewActivityDetails()
	START
	click view button in PtkActivityMenu interface
	read data from table bulletin with the chosen bulletin
	display data
	END	

addActivity()

START

addActivity() is called

Create object service to connect to PtkActivityModel

RETURN to PtkActivityMenu interface

END

searchActivity()

START

enter input in search bar in PtkActivityMenu

read data from table PtkActivity with the input data as condition

RETURN to PtkActivityMenu interface

END

editActivity()

START

click delete button in PtkActivityMenu interface

enter club name, advisor club name, organizer, application type, activity type, participant number, venue, activity duration and budget

click confirm button in ConfirmEditActivity interface

alter data in table PtkActivity

display deletion success message

END

deleteActivity()

START

click delete button in PtkActivityMenu interface

delete data in table PtkActivity

display deletion success message

END

4.4.   Manage Calendar [SDD-REQ-400] (Waleed Ahmed Al Amri CB16177)
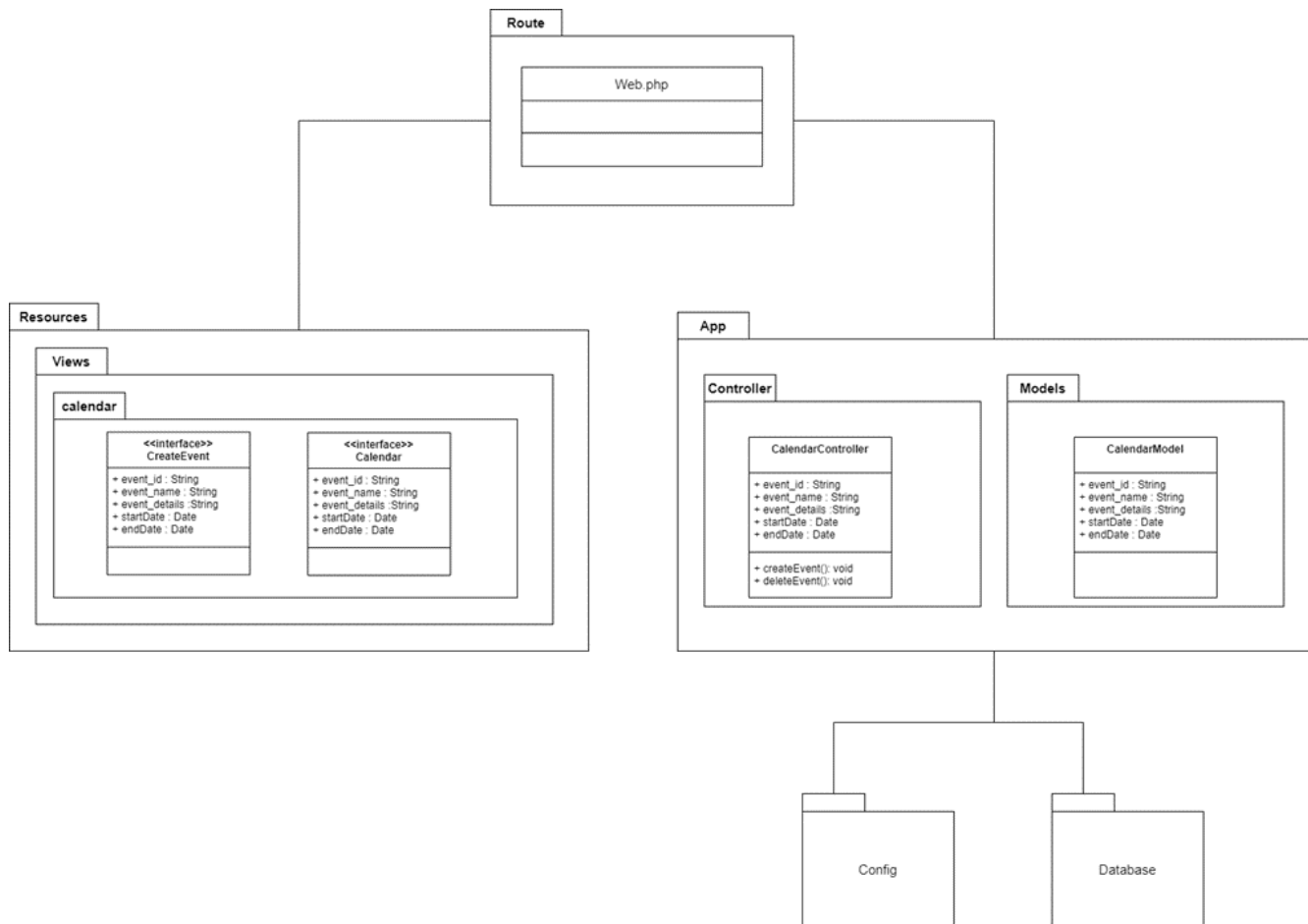
4.4.1 Calendar [SDD-REQ-401]

Class Type	Boundary Class	
Responsibility	Calendar's main page interface displays the yearly calendar along with all of the events and activities that took place during that time.	
Attributes	**Attributes Name**	**Attributes Type**
	event_id	String
	event_name	String
	event_details	String
	startDate	String
	endDate	String
Methods	**Method Name**	**Description**
	N/A	N/A

Algorithm	N/A

4.4.2 CreateEvent [SDD-REQ-402]

Class Type	Boundary Class	
Responsibility	An interface that enables the Petakom committee to enter details and generate events.	
Attributes	**Attributes Name**	**Attributes Type**
	event_id	String
	event_name	String
	event_details	String
	startDate	Date
	endDate	Date
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.4.3 CalendarController [SDD-REQ-403]

Class Type	Boundary Class	
Responsibility	Communicate with the CalendarModel and manage Calendar interfaces.	
Attributes	**Attributes Name**	**Attributes Type**
	event_id	String
	event_name	String
	event_details	String
	startDate	Date
	endDate	Date
Methods	**Method Name**	**Description**
	createEvent()	allows the user to create an event and add details
	deleteEvent()	allows the user to delete an event and it's details
Algorithm	**createEvent()** 1. START	

	2. INSERT INTO calendar (event_name , event_details, startDate, endDate) VALUE (submit)
	3. END
	deleteEvent()
	1. START
	2. DELETE FROM calendarWHERE event_id=$id
	END

4.4.4 calendarModel [SDD-REQ-404]

Class Type	Boundary Class	
Responsibility	Managing the calendarModel's data between the system and the database.	
Attributes	**Attributes Name**	**Attributes Type**
	event_id	String
	event_name	String
	event_details	String
	startDate	Date
	endDate	Date
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.5. Generate Report and Activity Proposal [SDD-REQ-500] (HAZIQ DANIAL BIN KHAIRUL AZIZY)
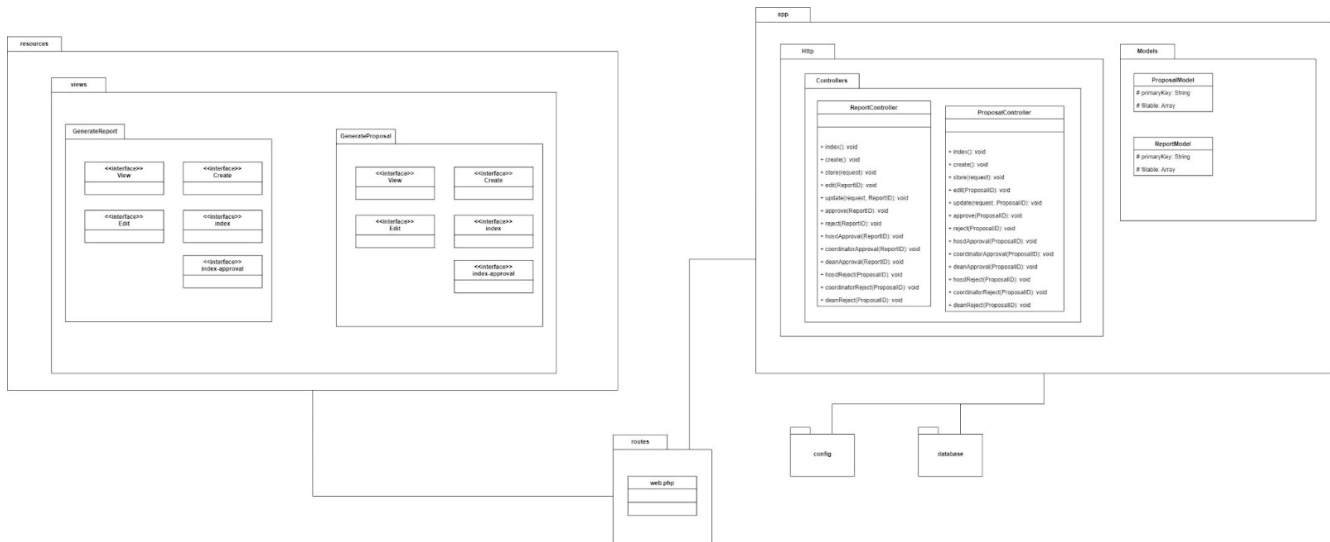
4.5.1 GenerateReport.View Page [SDD-REQ-503]

Class Type	Boundary Class	
Responsibility	Displays the content of the report.	
Attributes	**Attributes Name**	**Attributes Type**
	N/A	N/A
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.5.2 GenerateReport.Create Page [SDD-REQ-501]

Class Type	Boundary Class	
Responsibility	Display the page to create a report.	
Attributes	**Attributes Name**	**Attributes Type**
	N/A	N/A
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.5.3 GenerateReport.Edit Page [SDD-REQ-502]

Class Type	Boundary Class

Responsibility	Displays the page to edit the report.	
Attributes	**Attributes Name**	**Attributes Type**
	N/A	N/A
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.5.4 GenerateReport.index Page [SDD-REQ-503]

Class Type	**Boundary Class**	
Responsibility	Responsible for displaying the list of reports	
Attributes	**Attributes Name**	**Attributes Type**
	N/A	N/A
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.5.5 GenerateReport.index-approval Page
[SDD-REQ-507][SDD-REQ-504][SDD-REQ-505][SDD-REQ-506]

Class Type	**Boundary Class**	
Responsibility	Responsible for displaying the list of report applicants	
Attributes	**Attributes Name**	**Attributes Type**
	N/A	N/A
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.5.6 GenerateProposal.View Page [SDD-REQ-503]

Class Type	**Boundary Class**	
Responsibility	Displays the content of the proposal.	
Attributes	**Attributes Name**	**Attributes Type**

	N/A	N/A
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.5.7 GenerateProposal.Create Page [SDD-REQ-501]

Class Type	**Boundary Class**	
Responsibility	Display the page to create a proposal	
Attributes	**Attributes Name**	**Attributes Type**
	N/A	N/A
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.5.8 GenerateProposal.Edit Page [SDD-REQ-502]

Class Type	**Boundary Class**	
Responsibility	Displays the page to edit a proposal	
Attributes	**Attributes Name**	**Attributes Type**
	N/A	N/A
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.5.9 GenerateProposal.index Page [SDD-REQ-503]

Class Type	**Boundary Class**	
Responsibility	Responsible for displaying the list of proposals	
Attributes	**Attributes Name**	**Attributes Type**
	N/A	N/A
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.5.10 GenerateProposal.index-approval Page

[SDD-REQ-507][SDD-REQ-504][SDD-REQ-505][SDD-REQ-506]

Class Type	Boundary Class	
Responsibility	Responsible for displaying the list of proposal applicants	
Attributes	**Attributes Name**	**Attributes Type**
	N/A	N/A
Methods	**Method Name**	**Description**
	N/A	N/A
Algorithm	N/A	

4.5.11 ReportController [SDD-REQ-505]

Class Type	Boundary Class	
Responsibility	Responsible for the processes for the module, interacting between the GUI and the model.	
Attributes	**Attributes Name**	**Attributes Type**
	N/A	N/A
Methods	**Method Name**	**Description**
	index()	Displays a lists of reports that has created
	create()	Displays the create report form
	store(request)	Store the report data
	edit(ReportID)	Displays the edit the report form
	update(request, ReportID)	Update the report
	approve(ReportID)	Approve report
	reject(ReportID)	reject report
	hosdApproval(ReportID)	Head of Student Development approve report
	coordinatorApproval(ReportID)	Coordinator approve report
	deanApproval(ReportID)	Dean approve report
	hosdReject(ReportID)	Head of Student Development reject report

	coordinatorReject(ReportID)	Coordinator reject report
	deanReject(ReportID)	Dean reject report
| Algorithm | ```
FUNCTION index():
    GET all forms from ReportModel
    RETURN GenerateReport.index or GenerateReport.index-approval
 interface class to display the lists
END FUNCTION


FUNCTION create():
     RETURN GenerateReport.create form
 END FUNCTION


FUNCTION store(request):

    CREATE report data from ReportModel


    APPEND request data such as  title, and report_content to report
data


     SET approval to pending


    STORE report data to ReportModel


    REDIRECT GenerateReport.index

END FUNCTION


FUNCTION edit(ReportID):
     GET report data from ReportModel using ReportID
     RETURN GenerateReport.edit page with report data
 END FUNCTION




 FUNCTION update(request, ReportID):
     GET report data from ReportModel using ReportID


    APPEND request data such as  title, and report_content to report
data


     SET approval to pending


    STORE report data to ReportModel


     REDIRECT to GenerateReport.index page
 END FUNCTION
``` |

```
FUNCTION approve(ReportID):

    IF user is Head of Student Development:

        CALL hosdApproval(ReportID) function

    ELSE IF user is Coordinator:

        CALL coordinatorApproval(ReportID) function

    ELSE IF user is Dean:

        CALL deanApproval(ReportID) function

    END IF

     REDIRECT to GenerateReport.index-approval page

END FUNCTION


 FUNCTION reject(ReportID):

    IF user is Head of Student Development:

        CALL hosdReject(ReportID) function

    ELSE IF user is Coordinator:

        CALL coordinatorReject(ReportID) function

    ELSE IF user is Dean:

        CALL deanReject(ReportID) function

    END IF

    REDIRECT to GenerateReport.index-approval page

END FUNCTION

FUNCTION hosdApproval(ReportID):

    GET report data from ReportModel
```

```
        SET report hosd_approval to approved

END FUNCTION

FUNCTION coordinatorApproval(ReportID):

    GET report data from ReportModel

    SET report coo_approval to approved

END FUNCTION

FUNCTION deanApproval(ReportID):

    GET report data from ReportModel

    SET report dean_approval to approved

END FUNCTION


 FUNCTION hosdReject(ReportID):

    GET report data from ReportModel

    SET report hosd_approval to rejected

END FUNCTION

FUNCTION coordinatorReject(ReportID):

    GET report data from ReportModel

    SET report coo_approval to rejected

END FUNCTION

FUNCTION deanReject(ReportID):

    GET report data from ReportModel

    SET report dean_approval to rejected

END FUNCTION
```

4.5.12 ProposalController [SDD-REQ-505]

Class Type	Boundary Class	
Responsibility	Responsible for the processes for the module, interacting between the GUI and the model.	
Attributes	**Attributes Name**	**Attributes Type**
	N/A	N/A
Methods	**Method Name**	**Description**
	index()	Displays a lists of proposals that has created
	create()	Displays the create proposal form
	store(request)	Store the proposal data
	edit(ProposalID)	Displays the edit the proposal form
	update(request, ProposalID)	Update the proposal
	approve(ProposalID)	Approve proposal
	reject(ProposalID)	reject proposal
	hosdApproval(ProposalID)	Head of Student Development approve proposal
	coordinatorApproval(ProposalID)	Coordinator approve proposal
	deanApproval(ProposalID)	Dean approve proposal
	hosdReject(ProposalID)	Head of Student Development reject proposal
	coordinatorReject(ProposalID)	Coordinator reject proposal
	deanReject(ProposalID)	Dean reject proposal
| Algorithm | ```
FUNCTION index():
    GET all forms from ProposalModel
    RETURN GenerateProposal.index or GenerateProposal.index-approval
interface class to display the lists
END FUNCTION


FUNCTION create():

    RETURN GenerateProposal.create form

END FUNCTION
``` | |

```
FUNCTION store(request):

    CREATE proposal data from ProposalModel

    APPEND request data such as  title, and proposal_content to
proposal data

     SET approval to pending

    STORE report data to ProposalModel

    REDIRECT GenerateProposal.index

END FUNCTION


FUNCTION edit(ProposalID):
    GET report data from ProposalModel using ProposalID
    RETURN GenerateProposal.edit page with report data
 END FUNCTION


FUNCTION update(request, ProposalID):
    GET report data from ProposalModel using ProposalID

    APPEND request data such as  title, and proposal_content to
 proposal data

    SET approval to pending

    STORE report data to ProposalModel

    REDIRECT to GenerateProposal.index page
 END FUNCTION
```

```
 FUNCTION approve(ProposalID):

    IF user is Head of Student Development:

       CALL hosdApproval(ProposalID) function

    ELSE IF user is Coordinator:

       CALL coordinatorApproval(ProposalID) function

    ELSE IF user is Dean:

       CALL deanApproval(ProposalID) function

    END IF

     REDIRECT to GenerateProposal.index-approval page

END FUNCTION

FUNCTION reject(ProposalID):

    IF user is Head of Student Development:

       CALL hosdReject(ProposalID) function

    ELSE IF user is Coordinator:

       CALL coordinatorReject(ProposalID) function

    ELSE IF user is Dean:

       CALL deanReject(ProposalID) function

    END IF

    REDIRECT to GenerateProposal.index-approval page

END FUNCTION
```

```
FUNCTION hosdApproval(ProposalID):

    GET report data from ProposalModel

    SET report hosd_approval to approved

END FUNCTION

FUNCTION coordinatorApproval(ProposalID):

    GET report data from ProposalModel

    SET report coo_approval to approved

END FUNCTION

FUNCTION deanApproval(ProposalID):

    GET report data from ProposalModel

    SET report dean_approval to approved

END FUNCTION

FUNCTION hosdReject(ProposalID):

    GET report data from ProposalModel

    SET report hosd_approval to rejected

END FUNCTION

FUNCTION coordinatorReject(ProposalID):

    GET report data from ProposalModel

    SET report coo_approval to rejected

END FUNCTION

FUNCTION deanReject(ProposalID):

    GET report data from ProposalModel
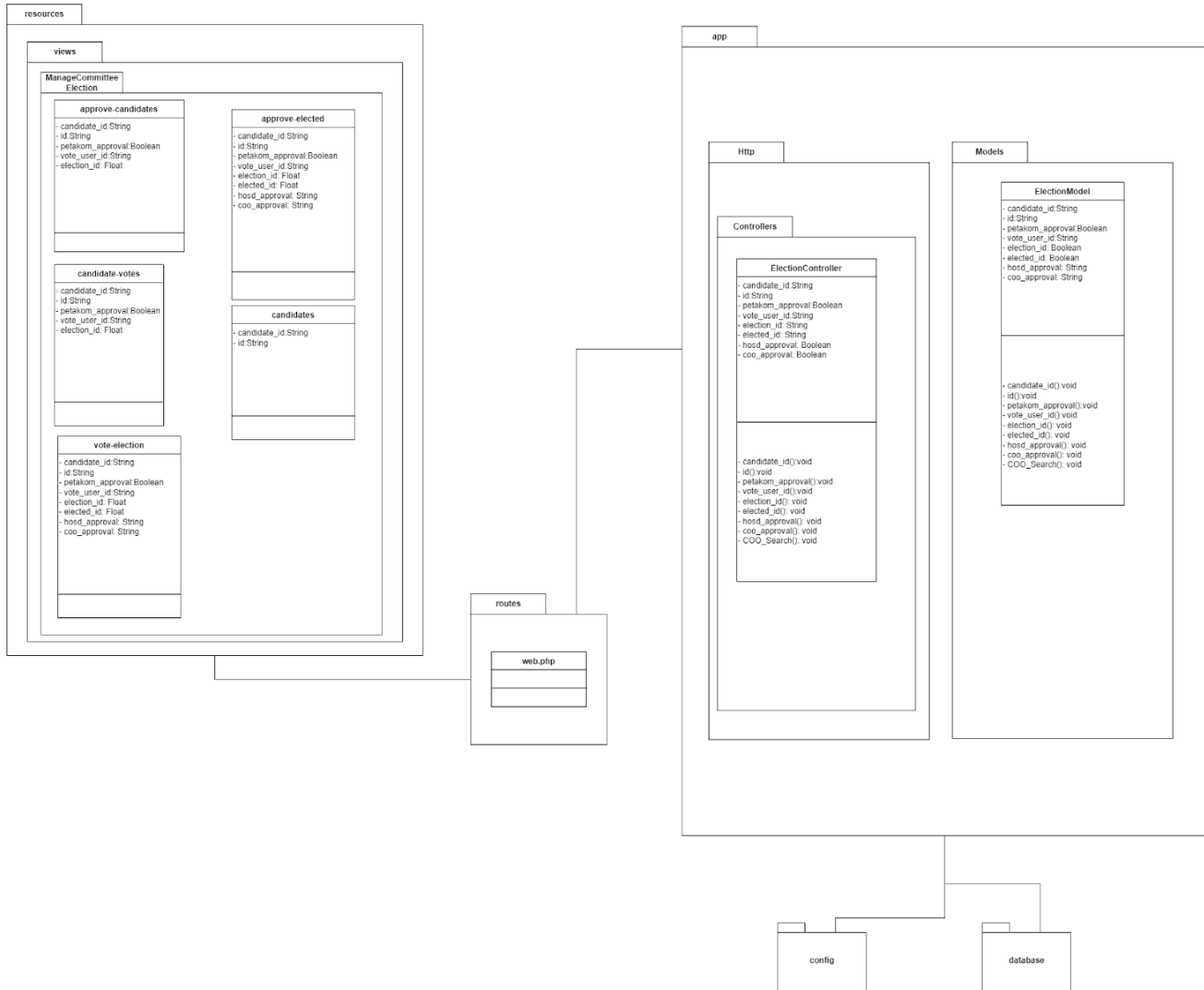
    SET report dean_approval to rejected
END FUNCTION
```

4.5.13 ReportModel [SDD-REQ-506]

Class Type	Boundary Class	
Responsibility	This class is responsible for data saving and data request retrieval from the database.	
	Attributes Name	**Attributes Type**
Attributes	primaryKey	String
	fillable	Array
	Method Name	**Description**
Methods	N/A	N/A
Algorithm	N/A	

4.5.14 ProposalModel [SDD-REQ-506]

Class Type	Boundary Class	
Responsibility	This class is responsible for data saving and data request retrieval from the database.	
	Attributes Name	**Attributes Type**
Attributes	primaryKey	String
	fillable	Array
	Method Name	**Description**
Methods	N/A	N/A
Algorithm	N/A	

4.6.   Manage Election Committee [SDD-REQ-600] (NABIL ABHAR BIN MOKHTAR CB20015)

4.6.1 candidates [SDD-REQ-601]

Class Type	Boundary Class	
Responsibility	This class is responsible for proposing suitable candidates to become members of PETAKOM	
Attributes	Attributes Name	Attributes Type
	Candidate_id	String
	id	String
	petakom_approval	Boolean
	vote_user_id	String
	election_id	String
	elected_id	String

	hosd_approval	Boolean
	coo_approval	Boolean
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.6.2 candidate-votes [SDD-REQ-602]

Class Type	**Boundary Class**	
Responsibility	This class is responsible for voting contesting candidates candidates who are competing	
	Attributes Name	**Attributes Type**
	Candidate_id	String
	id	String
	petakom_approval	Boolean
Attributes	vote_user_id	String
	election_id	String
	elected_id	String
	hosd_approval	Boolean
	coo_approval	Boolean
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.6.3 approve-candidates [SDD-REQ-601_2]

Class Type	**Boundary Class**	
Responsibility	This class is responsible for displaying approved candidates' information	
	Attributes Name	**Attributes Type**
	Candidate_id	String
	id	String
Attributes	petakom_approval	Boolean
	vote_user_id	String
	election_id	String

	elected_id	String
	hosd_approval	Boolean
	coo_approval	Boolean
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.6.4 vote-election [SDD-REQ-603_4]

Class Type	Boundary Class	
Responsibility	This class is responsible for displaying voting information	
	Attributes Name	**Attributes Type**
	Candidate_id	String
	id	String
	petakom_approval	Boolean
Attributes	vote_user_id	String
	election_id	String
	elected_id	String
	hosd_approval	Boolean
	coo_approval	Boolean
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.6.5 elected-approve [SDD-REQ-603_4]

Class Type	Boundary Class	
Responsibility	This class is responsible for approving the elected candidate who got the votes' majority	
	Attributes Name	**Attributes Type**
	Candidate_id	String
	id	String
Attributes	petakom_approval	Boolean
	vote_user_id	String
	election_id	String

	elected_id	String
	hosd_approval	Boolean
	coo_approval	Boolean
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.6.6 ManageCommitteeElectionController

Class Type	Boundary Class	
Responsibility	This class is responsible to connect ProposeCandidate, VotingMenu, CandidatesMenu, ApprovedCandidates, OverallStatistics, and CommiteeMenu interface with ElectionModel	
	Attributes Name	**Attributes Type**
Attributes	Candidate_id	String
	id	String
	petakom_approval	Boolean
	vote_user_id	String
	election_id	String
	elected_id	String
	hosd_approval	Boolean
	coo_approval	Boolean
Methods	**Method Name**	**Description**
	- candidate_id():void	Candidate's ID
	- id():void	ID
	- petakom_approval():void	list of proposed candidates that need to be reviewed and approved by the Petakom members

	- vote_user_id():void	Voting information of the contesting candidates
	- election_id(): void	ID that control the election process when the voting session is initiated
	- elected_id(): void	To display the candidates' details that are voted by students
	- hosd_approval(): void	Hosd approval of the elected winner that got the majority of the vote
	- coo_approval(): void	Coordinator approval of the elected winner that got the majority of the vote

Algorithm	**candidate_id()**
	START
	   candidate-id() is called
	   Create object student data to connect ElectionModel
	   RETURN to candidates interface
	 END
	
	id()
	START
	  id() is called
	  Create object student data to connect ElectionModel
	  RETURN to menu interface
	END
	
	petakom_approval()
	START
	  petakom_approval() is called
	  Create object student data to connect ElectionModel
	  RETURN to approve-candidates interface	

END

vote_user_id()

START

  vote_user_id() is called

  Create object student data to connect ElectionModel

  RETURN to candidate-votes interface

END

election_id()

START

  election_id() is called

  Create object student data to connect ElectionModel

  RETURN to candidate-votes interface

END

elected_id()

START

  elected_id() is called

  Create object petakom data to connect ElectionModel

  RETURN to vote-election interface

END

Hosd_approval()

START

  Hosd_approval() is called

  Create object petakom data to connect ElectionModel

  RETURN to approve-elected interface

END

Coo_approval()

START

  Coo_approval() is called

|	
	Create object petakom data to connect ElectionModel
	RETURN to approve-elected interface
	END

4.6.7 ManageCommitteeElectionModel

Class Type	Boundary Class	
Responsibility	To connect, store, retrieve, and manage the database of Student, Petakom member, Head of Student Development, and Coordinator information	
	Attributes Name	**Attributes Type**
Attributes	Candidate_id	String
	id	String
	petakom_approval	Boolean
	vote_user_id	String
	election_id	String
	elected_id	String
	hosd_approval	Boolean
	coo_approval	Boolean
	Method Name	**Description**
Methods	- candidate_id():void	Candidate's ID
	- id():void	ID
	- petakom_approval():void	list of proposed candidates that need to be reviewed and approved by the Petakom members
	- vote_user_id():void	Voting information of the contesting candidates
	- election_id(): void	ID that control the election process when the voting session is initiated
	- elected_id(): void	To display the candidates' details that are voted by students

	- hosd_approval(): void	Hosd approval of the elected winner that got the majority of the vote
	- coo_approval(): void	Coordinator approval of the elected winner that got the majority of the vote

Algorithm	**candidate_id()**
	
	START
	    READ candidate_id
	    IF candidate_id == candidate_id
	    THEN send data of candidate_id and id,
	  PRINT in candidates interface
	END
	
	id()
	START
	    READ if
	    IF id == id
	    THEN send data of users_id and id,
	  PRINT in menu interface
	END
	
	petakom_approval()
	START
	    READ petakom_approval
	    IF petakom_approval == petakom_approval
	    THEN approve data of candidate_id, and id
	  PRINT "The candidate is successfully approved"
	    END IF
	    RETURN to candidates interface page
	END
	
	vote_user_id()	

START

    READ vote_user_id

    IF vote_user_id == vote_user_id

    THEN send data of candidate_id, and id, petakom_approval

  PRINT "Candidates' list"

    END IF

    RETURN to candidates-votes interface

END

election_id()

START

    READ election_id

    IF election_id == election_id

    THEN send data of candidate_id, and id,

petakom_approval, vote_user_id, and election_id

  PRINT "The candidates is successfully voted"

    END IF

    RETURN to candidates-votes interface

END

elected_id()

START

    READ elected_id

    IF elected_id == elected_id

    THEN send data of candidate_id, and id,

petakom_approval, vote_user_id, and election_id, elected_id

  PRINT "Vote counts"

    END IF

    RETURN to vote-election interface

END

hosd_approval()

START

	READ hosd_approval
	IF hosd_approval == hosd_approval
	THEN send data of candidate_id, and id, petakom_approval, vote_user_id, and election_id, elected_id, hosd_approval
	PRINT "winner approved"
	END IF
	RETURN to approve-elected interface
	END
	coo_approval()
	START
	READ coo_approval
	IF coo_approval == coo_approval
	THEN send data of candidate_id, and id, petakom_approval, vote_user_id, and election_id, elected_id, coo_approval
	PRINT "winner approved"
	END IF
	RETURN to approve-elected interface
	END

4.7.   Manage Bulletin [SDD-REQ-700] (AMIR MUHAMMAD HAFIZ BIN OTHMAN CB20152)

4.7.1 UserBulletinMenu [SDD-REQ-701]

Class Type	Boundary Class	
Responsibility	This class is responsible for displaying the user bulletin menu interface.	
Attributes	**Attributes Name**	**Attributes Type**
	bulletin_id	INT
	title	VARCHAR(255)
	category	VARCHAR(255)
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.7.2 PtkBulletinMenu [SDD-REQ-702]

Class Type	Boundary Class	
Responsibility	This class is responsible for displaying the PETAKOM committee bulletin menu interface.	
Attributes	**Attributes Name**	**Attributes Type**
	bulletin_id	INT
	title	VARCHAR(255)
	category	VARCHAR(255)
	posted_by	DATE
	created_at	TIMESTAMP
	expired_at	TIMESTAMP
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.7.3 ViewBulletin [SDD-REQ-703]

Class Type	Boundary Class	
Responsibility	This class is responsible for displaying the selected bulletin details interface.	
Attributes	**Attributes Name**	**Attributes Type**
		INT
	bulletin_id	VARCHAR(50)
	title	INT
	message	VARCHAR(255)
	category	VARCHAR(255)
	url_reference	VARCHAR(255)
	posted_by	VARCHAR(255)
	created_at	DATE
		TIMESTAMP
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.7.4 AddBulletin [SDD-REQ-704]

Class Type	Boundary Class	
Responsibility	This class is responsible for displaying the add bulletin interface for PETAKOM committee.	
Attributes	**Attributes Name**	**Attributes Type**
	title	VARCHAR(255)
	category	VARCHAR(255)
	message	VARCHAR(255)
	url_reference	VARCHAR(255)
	duration	INT
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.7.5 ConfirmAddBulletin [SDD-REQ-705]

Class Type	Boundary Class	
Responsibility	This class is responsible for displaying the confirmation add bulletin interface for PETAKOM committee.	
Attributes	**Attributes Name**	**Attributes Type**
	title	VARCHAR(255)
	category	VARCHAR(255)
	message	VARCHAR(255)
	url_reference	VARCHAR(255)
	duration	INT
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.7.6 EditBulletin [SDD-REQ-706]

Class Type	Boundary Class	
Responsibility	This class is responsible for displaying the edit bulletin interface for PETAKOM committee.	
Attributes	**Attributes Name**	**Attributes Type**

	bulletin_id	INT
	title	VARCHAR(255)
	category	VARCHAR(255)
	message	VARCHAR(255)
	url_reference	VARCHAR(255)
	duration	INT
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.7.7 ConfirmEditBulletin [SDD-REQ-707]

Class Type	**Boundary Class**	
Responsibility	This class is responsible for displaying ththe confirmation edit bulletin interface for PETAKOM committee.	
	Attributes Name	**Attributes Type**
	bulletin_id	INT
	title	VARCHAR(255)
Attributes	category	VARCHAR(255)
	message	VARCHAR(255)
	url_reference	VARCHAR(255)
	duration	INT
Methods	**Method Name**	**Description**
	Not applicable	Not applicable
Algorithm	Not applicable	

4.7.8 BulletinController [SDD-REQ-708]

Class Type	**Boundary Class**	
Responsibility	This class is responsible to connect UserBulletinMenu interface, PtkBulletinMenu interface, ViewBulletin interface, AddBulletin interface, EditBulletin interface, ConfirmAddBulletin interface, and ConfirmEditBulletin interface with BulletinModel.	
Attributes	**Attributes Name**	**Attributes Type**
	bulletin_id	INT

	title	VARCHAR(255)
	message	VARCHAR(255)
	category	VARCHAR(255)
	url_reference	VARCHAR(255)
	posted_by	DATE
	created_at	TIMESTAMP
	updated_at	TIMESTAMP
	Method Name	**Description**
Methods	viewBulletinList()	The user and the PETAKOM committee can view the list of bulletins.
	viewBulletinDetails()	The user and the PETAKOM committee can view the selected bulletin details.
	addBulletin()	The PETAKOM committee can add a bulletin.
	searchBulletin()	The PETAKOM committee can search the bulletin.
	editBulletin()	The PETAKOM committee can edit a bulletin.
	updateBulletin()	The PETAKOM committee can update data to database.
	deleteBulletin()	The PETAKOM committee can delete a bulletin.

Algorithm	**viewBulletinList()**
	START
	   retrieve data from table bulletin where expired date > current date
	   IF not PETAKOM committee
	     Return to UserBulletinMenu interface
	   ELSE
	     Return to PtkBulletinMenu interface
	   END IF	

END

viewBulletinDetails()

START

read data from table bulletin with the chosen bulletin id

Return to view bulletin with the chosen bulletin data

END

addBulletin()

START

Create object service to connect to BulletinModel

Save object to insert data into the bulletin table

Redirect to index interface

END

searchBulletin()

START

Request search input

Retrieve data from bulletin data with search input condition

Return to PtkBulletinMenu interface

END

editBulletin()

START

Find data from the bulletin table with the selected bulletin id

Redirect to editBulletin interface with the selected bulletin data

END

updateBulletin()

START

Create object service to connect to BulletinModel with the
selected bulletin id

Save object to update data into the bulletin table

	Redirect to index interface
	END
	deleteBulletin()
	START
	Find data from the bulletin table with the selected bulletin id
	Save object to delete data from the bulletin table
	Redirect to index interface
	END

4.7.9 BulletinModel [SDD-REQ-709]

Class Type	Boundary Class	
Responsibility	This class will store input data and retrieve data of bulletin from the database	
Attributes	**Attributes Name**	**Attributes Type**
	BULLETIN_ID	INT
	BULLETIN_TITLE	VARCHAR(50)
	BULLETIN_CATEGORY	VARCHAR(20)
	PTK_ID	VARCHAR(10)
	PTK_NAME	VARCHAR(50)
	CREATED_AT	DATE
	UPDATED_AT	DATE
	BULLETIN_MESSAGE	VARCHAR(1000)
	BULLETIN_URL_REFERENCE	VARCHAR(1000)
	BULLETIN_DURATION	INT
	BULLETIN_EXPIRED_AT	DATE
Methods	**Method Name**	**Description**
	viewBulletinList()	The user and the PETAKOM committee can view the list of bulletins.
	viewBulletinDetails()	The user and the PETAKOM committee can view the selected bulletin details.

	addBulletin()	The PETAKOM committee can add a bulletin.
	searchBulletin()	The PETAKOM committee can search the bulletin.
	editBulletin()	The PETAKOM committee can edit a bulletin.
	deleteBulletin()	The PETAKOM committee can delete a bulletin.

Algorithm	**viewBulletinList()**
	
	   START
	   select table bulletin
	   read data in table bulletin
	END
	
	viewBulletinDetails()
	START
	   select table bulletin with specific bulletin
	   read data in table bulletin
	END
	
	addBulletin()
	START
	   get input from ConfirmAddBulletin interface
	   insert input data into table bulletin
	END
	
	searchBulletin()
	START
	   get input from PtkBulletinMenu interface search bar
	   select table bulletin with input data condition
	   read data in table bulletin
	END	

	editBulletin()
	
	START
	
	    get input from ConfirmEditBulletin interface
	
	    insert input data into table bulletin
	
	END
	
	deleteBulletin()
	
	START
	
	    read data in table bulletin
	
	    delete data in table bulletin
	
	END

5. REQUIREMENT TRACEABILITY

5.1 Authentication [UC-PTK-100] (HAZMI HAZIM CB20117)

Requirement ID	Description	Design ID
RQ-PTK-101	Users should be able to enter their id, email and password.	SDD-REQ-103 SDD-REQ-105
RQ-PTK-102	Users should be able to click <<Register>>button.	SDD-REQ-106
RQ-PTK-103	User should be able to click <<Login>> button.	SDD-REQ-101
RQ-PTK-104	User should be able to enter id and password.	SDD-REQ-105
RQ-PTK-105	User should be able to click <<Login>> button.	SDD-REQ-106
RQ-PTK-106	User should be able to click <<Register Member>> button.	SDD-REQ-204 SDD-REQ-205 SDD-REQ-106
RQ-PTK-107	User should be able to enter id to register member of PETAKOM.	
RQ-PTK-108	User should be able to select role of new member.	
RQ-PTK-109	User should be able to click <<Save>> button.	
RQ-PTK-110	User should be able to reset their password by click <<Forgot Password>> button and should be able to enter email for reset link.	SDD-REQ-104 SDD-REQ-103 SDD-REQ-105 SDD-REQ-106

5.2 UserProfile [UC-PTK-200] (HAZMI HAZIM CB20117)

Requirement ID	Description	Design ID
RQ-PTK-201	Users should be able to view their profile.	SDD-REQ-201
RQ-PTK-202	User should be able to click <<Edit>> button.	SDD-REQ-205 SDD-REQ-106
RQ-PTK-203	User should be able to enter email, name, phone number and address.	SDD-REQ-202 SFF-REQ-205 SDD-REQ-106
RQ-PRK-204	User should be able to click <<Save>> button.	
RQ-PTK-205	User should be to click <<User>> button.	
RQ-PTK-206	Users should be able to clicks <<Change Password>> button.	SDD-REQ-203 SDD-REQ-205

RQ-PTK-207	User should be able to enter new password and confirm password.	SDD-REQ-106
RQ-PTK-208	User should be able to click <<Save>> button.	

5.3 ManagePtkActivity [UC-PTK-300] (WAN MOHD IKRAM HAMDI BIN WAN MAHADI CB19085)

Requirement ID	Description	Design ID
RQ-PTK-301	The users (student, lecturer, PETAKOM committee) click <<Activities>> tab in the main page. From the Activities page, click add activity.	SDD-REQ-301
RQ-PTK-302	User enters their information in the activity form.	SDD-REQ-303 SDD-REQ-304
RQ-PTK-303	The users clicks <<Submit>> button.	SDD-REQ-305
RQ-PRK-304	User successfully submitted the form	SDD-REQ-305
RQ-PTK-305	Users manage their activity proposal by clicking the <<Edit Activity form>> button.	SDD-REQ-306
RQ-PTK-306	User enters their name, date and time, club name, advisor club,etc...	SDD-REQ-306
RQ-PTK-307	User clicks <<Save>> button.	SDD-REQ-307
RQ-PTK-308	For the approval, users (Dean, coordinator and Head of Student Development) will click <<Activity list for approval>> button.	SDD-REQ-308
RQ-PTK-308	From the lists, users can approve or reject the activity from here.	SDD-REQ-308

5.4 ManageCalendar [UC-PTK-400] (Waleed Ahmed AL Amri CB16177)

Requirement ID	Description	Design ID
RQ-PTK-401	Users are able to view the calendar	SDD-REQ-401 SDD-REQ-404
RQ-PTK-402	The Petakom Committee can plan an event for any day.	SDD-REQ-401 SDD-REQ-402 SDD-REQ-404

RQ-PTK-403	The software will show a form for adding an event.	SDD-REQ-406 SDD-REQ-407
RQ-PTK-404	Each event should have a unique colour created by the system.	SDD-REQ-401
RQ-PTK-405	System should display a successful message after adding an event or delete it	SDD-REQ-401 SDD-REQ-402
RQ-PTK-406	Any event may be deleted by the Petakom Committee.	SDD-REQ-401 SDD-REQ-404

5.5 GenerateReportAndProposal [UC-PTK-] (HAZIQ DANIAL BIN KHARUL AZIZY CB20140)

Requirement ID	Description	Design ID
RQ-PTK-501	Students, Lecturers, and PETAKOM Committee can create the form.	SDD-REQ-501
RQ-PTK-502	Students, Lecturers, and PETAKOM Committee can update the form.	SDD-REQ-502
RQ-PTK-503	Students, Lecturers, and PETAKOM Committee can see the final approval status result of the form.	SDD-REQ-503
RQ-PTK-504	Petakom Coordinator can change the approval status of the form.	SDD-REQ-504
RQ-PTK-505	Dean can change the approval status of the form.	SDD-REQ-505
RQ-PTK-506	Head of Student Development can change the approval status of the form.	SDD-REQ-506
RQ-PTK-507	Head of Student Development, Dean, Petakom Coordinator can see the list of the applied forms.	SDD-REQ-507

5.6 ManageCommitteeElection [UC-PTK-600] (NABIL ABHAR BIN MOKHTAR CB20015)

Requirement ID	Description	Design ID
RQ-PTK-600	**Propose a Candidate**	SDD-REQ-600
RQ-PTK-600_1	Students click the <<propose a candidate>> button to propose a candidate for the election	SDD-REQ-600_1

RQ-PTK-600_2	System display the proposal form	SDD-REQ-600_2
RQ-PTK-600_3	Students fill out the proposal form	SDD-REQ-600_3
RQ-PTK-600_4	System send the proposal to PETAKOM Committee and Coordinator	SDD-REQ-600_4
RQ-PTK-601	**Manage Candidates' List**	SDD-REQ-601
RQ-PTK-601_1	Coordinator and PETAKOM Committee click a <<proposed candidates>> to view the proposed candidates' list	SDD-REQ-601_1
RQ-PTK-601_2	System display the proposed candidates list	SDD-REQ-601_2
RQ-PTK-601_3	Coordinator and PETAKOM Committee check for each of the candidate's qualification and decided whether to approve or reject the proposal	SDD-REQ-601_3
RQ-PTK-601_4	System will save the approved candidates list	SDD-REQ-601_4
RQ-PTK-602	Manage Voting Session	SDD-REQ-602
RQ-PTK-602_1	Students click a <<view candidates>> button to view the candidate list	SDD-REQ-602_1
RQ-PTK-602_2	System display the list and the information about all the approved candidates' information	SDD-REQ-602_2
RQ-PTK-602_3	Students choose a candidate	SDD-REQ-602_3
RQ-PTK-602_4	System display the voting information for the student's affirmation	SDD-REQ-602_4
RQ-PTK-602_5	The student click the <<vote>> button	SDD-REQ-602_5
RQ-PTK-603	**Manage Voting Results**	SDD-REQ-603
RQ-PTK-603_1	Head of the Student Development click a <<button>> to view the election results	SDD-REQ-603_1
RQ-PTK-603_2	System display the election result	SDD-REQ-603_2
RQ-PTK-603_3	Head of the Student Development decided whether or not to approve the new PETAKOM representatives	SDD-REQ-603_3
RQ-PTK-603_4	The system will save the new representatives information if it were approved. The system will then display the election results and the new board of	SDD-REQ-603_4

	PETAKOM committee in some sort of notification center	

5.7 ManageBulletin [UC-PTK-700] (AMIR MUHAMMAD HAFIZ BIN OTHMAN CB20152)

Requirement ID	Description	Design ID
RQ-PTK-701	The user and PETAKOM committee are able to view the selected bulletin details.	SDD-REQ-703
RQ-PTK-702	The user and PETAKOM committee are able to view the list of bulletins.	SDD-REQ-701 SDD-REQ-702
RQ-PTK-703	The PETAKOM committee is able to add a new bulletin.	SDD-REQ-705
RQ-PRK-704	The PETAKOM committee is able to edit a bulletin.	SDD-REQ-707
RQ-PTK-705	The PETAKOM committee is able to delete a bulletin.	SDD-REQ-702
RQ-PTK-706	The PETAKOM committee is able to search for a bulletin.	SDD-REQ-702
RQ-PTK-707	The system displays input error messages if there is any input error in the add bulletin form.	SDD-REQ-704
RQ-PTK-708	The system displays input error messages if there is any input error in the edit bulletin form.	SDD-REQ-706