



Universiti
Malaysia
PAHANG

Engineering • Technology • Creativity

SDD Document SEM II 20222023

e-Munakahat System (EMUN)

Group Name

1. Muhammad Amir Bin Mohamed Ali [CB21060]
2. Ahmad Suffian Bin Md Noor Suhaime [CB21137]
3. Chua Kian Pheng [CB21106]
4. Nik Alia Syafiqah Binti Nik Azuri [CB21035]
5. Shammene A/P Muneesvaran [CB21018]



Quantum Corp

TABLE OF CONTENTS

1. INTRODUCTION	3
1.1 Purpose	3
1.2 System Identification	4
1.3 System Overview	5
1.4 Referenced Document	7
2. DATA DESIGN	9
2.1 Entity Relationship Diagram (ERD)	9
2.2 Data Dictionary	10
3. GENERAL ARCHITECTURE	11
3.1 Layered Architecture	11
3.1.1 Application Layer	11
3.1.2 Business Services Layer	15
3.1.3 Middleware Layer	19
3.2 MVC Package Relationship	20
4. DETAIL DESIGN	21
4.1 Manage Incentive [SDD-REQ-100]	21
4.2 Manage Incentive Controller [SDD-REQ-200]	24
4.3 Manage Incentive Model [SDD-REQ-300]	26
5. REQUIREMENT TRACEABILITY	27
5.1 Manage Incentive [SRS-REQ-900]	27
APPENDIX A	29
APPENDIX B	30
APPENDIX C1	31
APPENDIX C2	32
APPENDIX C3	33
APPENDIX C4	34

1. INTRODUCTION

1.1 Purpose

This software design document (SDD) serves as a blueprint for the development of the e-Munakahat System (EMUN). It outlines the software design process, including the software's architecture, modules, interfaces, and data structures. The primary purpose of an SDD is to ensure that the software development team and stakeholders understand the design of the software application and how it will function.

The software design document (SDD) is a communication tool between the development team and stakeholders, ensuring everyone is on the same page regarding the software's design and implementation. It will also help the stakeholders to understand any limitations or assumptions that may impact the system's performance or usability, which will help the stakeholders to make informed decisions and better plan for the project. This will also help to ensure that the final product meets the needs of the users and the jurisdiction.

1.2 System Identification

The Software Design Document (SDD) belongs to the “e-Munakahat System” (EMUN).

Table 1.1 Document Identity.

System title	e-Munakahat System												
System abbreviation	EMUN												
System identification number	SDD_EMUN_QC_2023												
Subsystem Title	IkatanCinta												
Subsystem Abbreviation	IC												
Package ID	<div>SDD-REQ-100</div> <div>Meanings for terms use:</div> <table><tr><td>SDD</td><td>Software Design Document</td></tr><tr><td>REQ</td><td>Requirement</td></tr><tr><td>100</td><td>Package number</td></tr></table>			SDD	Software Design Document	REQ	Requirement	100	Package number				
SDD	Software Design Document												
REQ	Requirement												
100	Package number												
Use Case ID	<div>UC101-EMUN-001</div> <div>Meanings for terms used:</div> <table><tr><td>UC</td><td>Use Case</td></tr><tr><td>1</td><td>Number of the system module</td></tr><tr><td>01</td><td>Number of use case within a module in the subsystem</td></tr><tr><td>EMUN</td><td>e-Munakahat System (System name)</td></tr><tr><td>001</td><td>Document release number</td></tr></table>			UC	Use Case	1	Number of the system module	01	Number of use case within a module in the subsystem	EMUN	e-Munakahat System (System name)	001	Document release number
UC	Use Case												
1	Number of the system module												
01	Number of use case within a module in the subsystem												
EMUN	e-Munakahat System (System name)												
001	Document release number												
Requirement Traceability ID	<div>UC101-EMUN-001</div> <div>Meanings for terms used:</div> <table><tr><td>UC</td><td>Use Case</td></tr><tr><td>1</td><td>Number of the system module</td></tr><tr><td>01</td><td>Number of use case within a module in the system</td></tr><tr><td>EMUN</td><td>e-Munakahat System (System name)</td></tr><tr><td>001</td><td>Document release number</td></tr></table>			UC	Use Case	1	Number of the system module	01	Number of use case within a module in the system	EMUN	e-Munakahat System (System name)	001	Document release number
UC	Use Case												
1	Number of the system module												
01	Number of use case within a module in the system												
EMUN	e-Munakahat System (System name)												
001	Document release number												

Document Identification System	SDD_EMUN_IC_2023_V1.0
-----------------------------------	-----------------------

Symbol/Number	Meaning
EMUN	It represents the system name which is e-Munakahat System (EMUN)
SDD	It represents the software design document (SDD)
QC	It represents the company name which is Quantum Corp SDN. BHD.
2023	It represents the year where the system has been released.
V1.0	V represent the word "Version" of the system while 1.0 is the general version of the system. The number will increase by 0.1 if there any updates or bugs fixed.

1.3 System Overview

Our system is a web-based system to support the Pahang Wedding management system. The purpose of the system is to provide a solution for managing wedding registration and operations efficiently. Users can use this system for marriage registration and information retrieval. It also eases the government to collect granular data on weddings in Pahang and store the information for future use. This system is handled by the Quantum Corp company. This system contains various functionalities such as user registration, marriage application, marriage registration, marriage preparation course, proof of payment, request for marriage, marriage card, marriage consultation, and special incentives for the bride and groom.

There are five modules in this system which are:

1. Registration and handle user profile

The initial stage of using the system is applicant registration, which is required for applicants to enter the system. It contains private information such as an Identity Card number, phone number, email address, and password. Since registration is based on the user's IC number, a unique number with no duplicates, each user has just one account for the system. Once the applicant's registration is successful, they can log in to access the system's contents after completing the registration process. The admin

can register new staff through the admin dashboard. Once registration for staff is successful, they will be granted access to the system. The system also allows users (applicants & staff) to modify their profile details to update their information. In case the applicant has forgotten their password, they will be able to reset their password. Apart from that, the admin can update staff and applicant details through the admin dashboard.

2. Attend the marriage preparation course with proof of payment and to request a marriage.

After users register and login into the account, the users can proceed to the next step where they can apply for the marriage registration course. As a marriage registration course is an important step for the e-Munakahat system will update the name list of applicants. The system will show if the user has already paid for the course or not. So, then the staff can print the proof of payment and send the receipt to the applicants. The system shall allow the admin to manage the marriage preparation course details such as date, time, and where the course will be conducted. While the staff will handle the things that are related to the user or applicants. Approve the applications in the system and print the applicants' name list for the course for every session. If the applicant presents on the course day, the staff can approve their marriage course certificate. After the course ends, the applicants will be represented with the certificate. The staff can update the applicant's name from the system for obtaining the marriage course certificate. Next step for the applicant is request their marriage from E-Munakahat system.

3. An application to register a marriage within or outside the country, as well as a voluntary marriage, and to produce the marriage card or certificate with proof of payment.

Marriage registration is split into two processes which are voluntary and authorized registration. User needs to pass a pre-marriage course to be able to register their marriage. Voluntary registration is for the older generation who live far away from town and did not register when they married. For authorized registration, the user must prepare various forms and information in which the template is provided in the system. After staff from JAIP has approved the marriage registration, it will notify the user so that they know their marriage registration has been approved.

4. Register for a marriage consultation with a service advisor.

A consultation module is a module where the user can make an appointment to seek advice or to get a divorce. This module requires the users to enter their personal information and also their spouse information including their marriage information and upload related documents as proof. Staff can view the application made by the user and will decide whether to approve the appointment and responsible to assign consultant and slot to the consultation session. Staff are also responsible for managing the list of consultants and their information in this module. The staff can edit, add, and delete the consultants' information in the system.

5. Application for special incentive for the bride and groom

The marriage registration system features a special incentive module that enables users to apply for incentives, provided they meet the necessary prerequisites. This module allows individuals to input their personal information and upload necessary documents, making the process more streamlined. Additionally, the staff side of the platform allows for the review of applications and ensures that the process is efficient and user-friendly. The special incentive module in the marriage registration system also includes a notification feature that keeps applicants informed about the status of their applications. Once the staff reviews the application, the applicant will receive a notification, either confirming their approval or outlining the reasons for the denial. This ensures that applicants are aware of the status of their application in a timely and efficient manner."

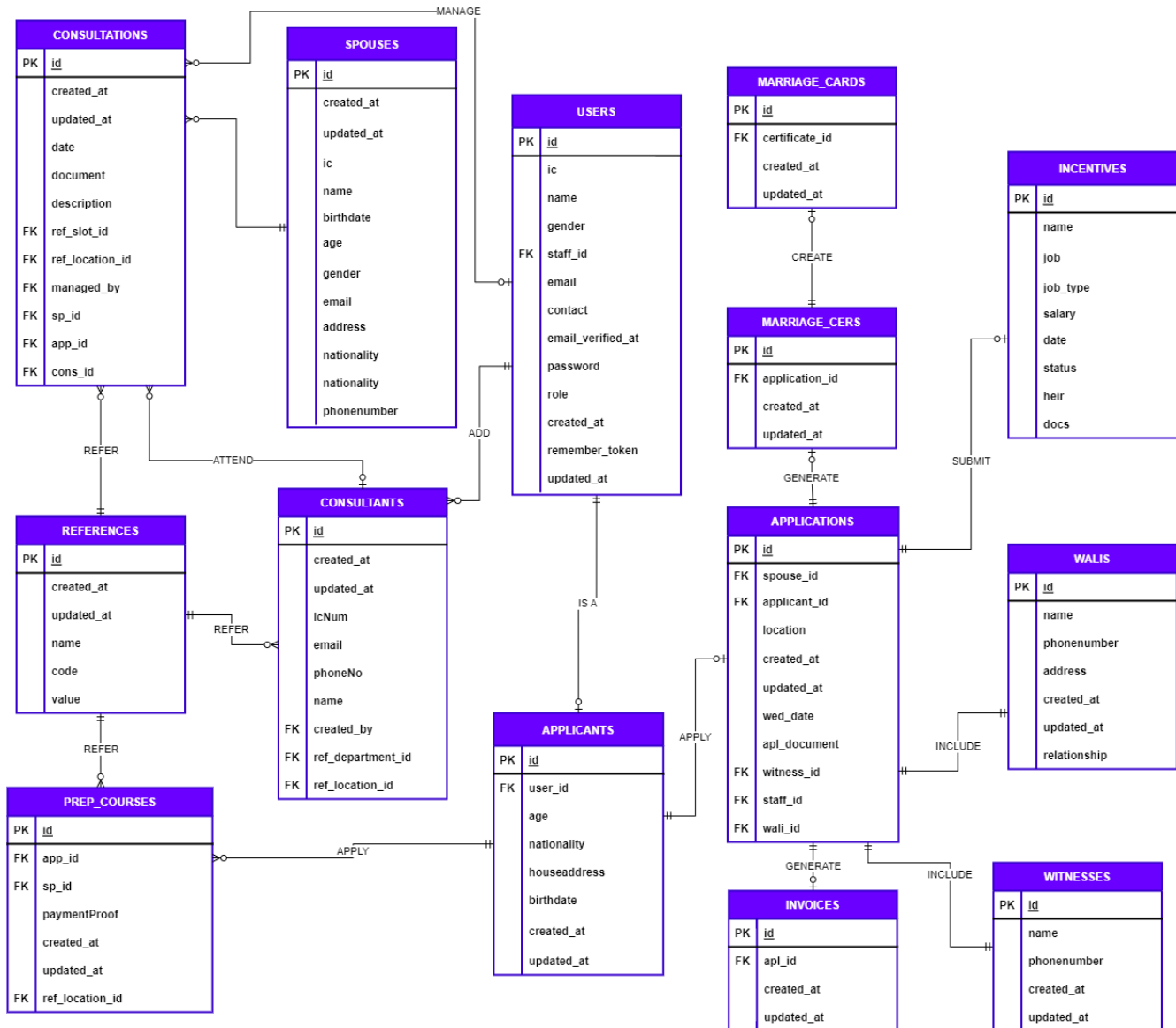
1.4 Referenced Document

1. Azma, B. A., et al (2023) BCS2243 Final Assessment Question Sem II 20222023.
2. Amir.A, et al (2023) SOFTWARE REQUIREMENT SPECIFICATION (SRS).
3. Layered Architecture. (2021, November 11). Retrieved May 2, 2023, from <https://www.baeldung.com/cs/layered-architecture>.
4. How to build a simple PHP MVC framework. (2021, May 30). Retrieved May 28, 2023, from <https://www.giuseppemaccario.com/how-to-build-a-simple-php-mvc-framework/>.
5. Simple PHP MVC Framework Example. (2023, May 26). Retrieved May 28, 2023, from <https://phpflow.com/php/simple-mvc-architecture-example-in-php/>.

6. What Is Middleware? Definition, Architecture, and Best Practices. (2022, February 18). Retrieved May 28, 2023, from <https://www.spiceworks.com/tech/cloud/articles/what-is-middleware/>.

2. DATA DESIGN

2.1 Entity Relationship Diagram (ERD)



2.2 Data Dictionary

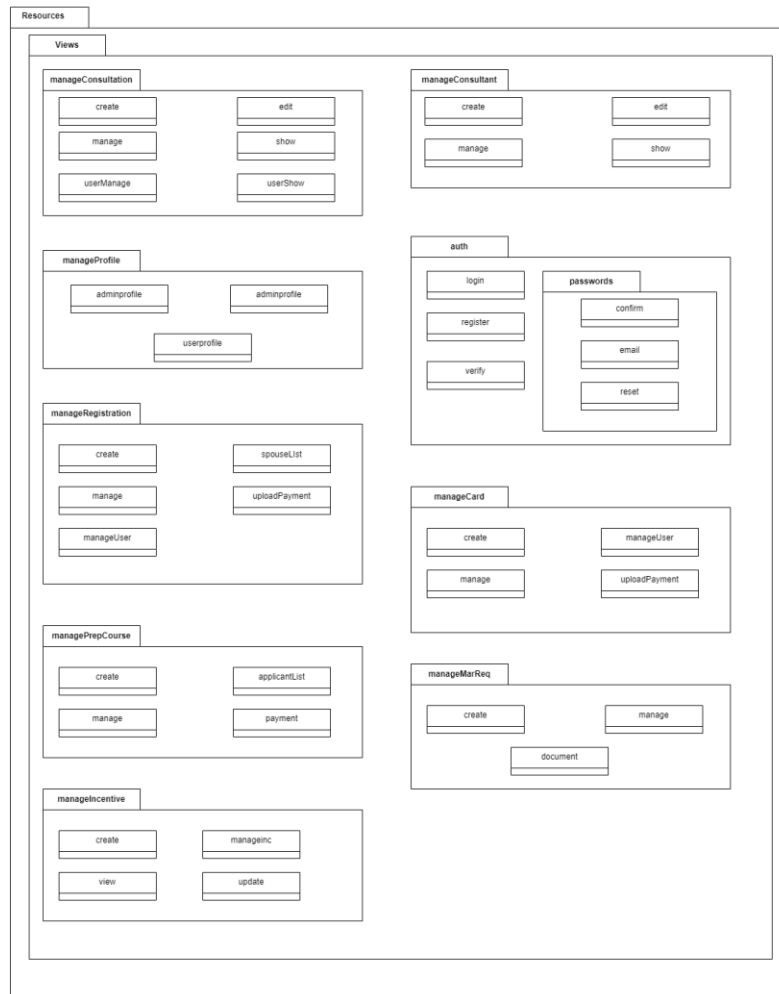
2.2.9 INCENTIVE

Field Name	Description	Data Type	Constraint
<u>id</u>	Incentive id	INT	PK
applicant_name	Incentive Applicant name	INT	
job	Job	VARCHAR (255)	
job_type	Job Type	VARCHAR (255)	
salary	Salary	DOUBLE	
date	Incentive Application Date	DATE (50)	
status	Approve status	VARCHAR (255)	
heir	Heir	VARCHAR (255)	
docs	Incentive Documents	BLOB	
created_at	Incentive created time	TIMESTAMP	NOT NULL
updated_at	Incentive updated time	TIMESTAMP	

3. GENERAL ARCHITECTURE

3.1 Layered Architecture

3.1.1 Application Layer



3.1.1.1 Manage Profile [SDD-REQ-100]

Class Name	Description
adminprofile	A view class for admin to register staff in the EMUN system
staffprofile	A view class for staff to manage their profile in the EMUN system
userprofile	A view class for user to manage their profile in the EMUN system

3.1.1.2 Auth [SDD-REQ-200]

Class Name	Description
login	A view class for user, staff and admin login into the EMUN system
register	A view class for user to register into the EMUN system
verify	A view class for user to verify the account in the EMUN system
confirm	A view class for user to confirm the account in the EMUN system
email	A view class for user to enter the email to reset password in the EMUN system
reset	A view class for user to reset password in the EMUN system.

3.1.1.3 Manage Preparation Course [SDD-REQ-300]

Class Name	Description
create	A view class for applicant to register the marriage preparation course by fill in the form
manage	A view class for applicant display the information regarding to marriage preparation course
applicantList	A view class for JAIP staff to see the list of applicants that apply marriage preparation course
payment	A view class for applicant to submit proof of payment

3.1.1.4 Manage Marriage Request [SDD-REQ-400]

Class Name	Description
create	A view class for applicant to register the marriage application by fill in the form
manage	A view class for applicant edit their information
document	A view class for applicant to submit document

3.1.1.5 Manage Registration [SDD-REQ-500]

Class Name	Description
spouseList	A view class that displays the spouse's information and allow the user to search, edit, delete, print and submit
create	A view class that allows the user to create a marriage registration application
manage	A view class that displays the user's marriage registration application and allow staff to accept or reject
manageUser	A view class that displays the marriage registration application
uploadPayment	A view class that displays payment receipt request and allow the user to upload their receipt

3.1.1.6 Manage Card [SDD-REQ-600]

Class Name	Description
create	A view class allow the user to request for the marriage card
manage	A view class that displays the user's marriage card application and allow staff to accept or reject
manageUser	A view class that displays the marriage card application
uploadPayment	A view class that displays payment receipt request and allow the user to upload their receipt

3.1.1.7 Manage consultation [SDD-REQ-700]

Class Name	Description
create	A view class to display form to allow the user to fill information details
show	A view class to allow JAIP staff to view the information of the application
edit	A view class to allow JAIP staff to fill user appointment detail and approve their application
userManage	A view class to allow user to check the list of consultation applications made

userShow	A view class to allow user to view the information of the application made
manage	A view class to display the list of application and allow JAIP staff to manage the consultation application

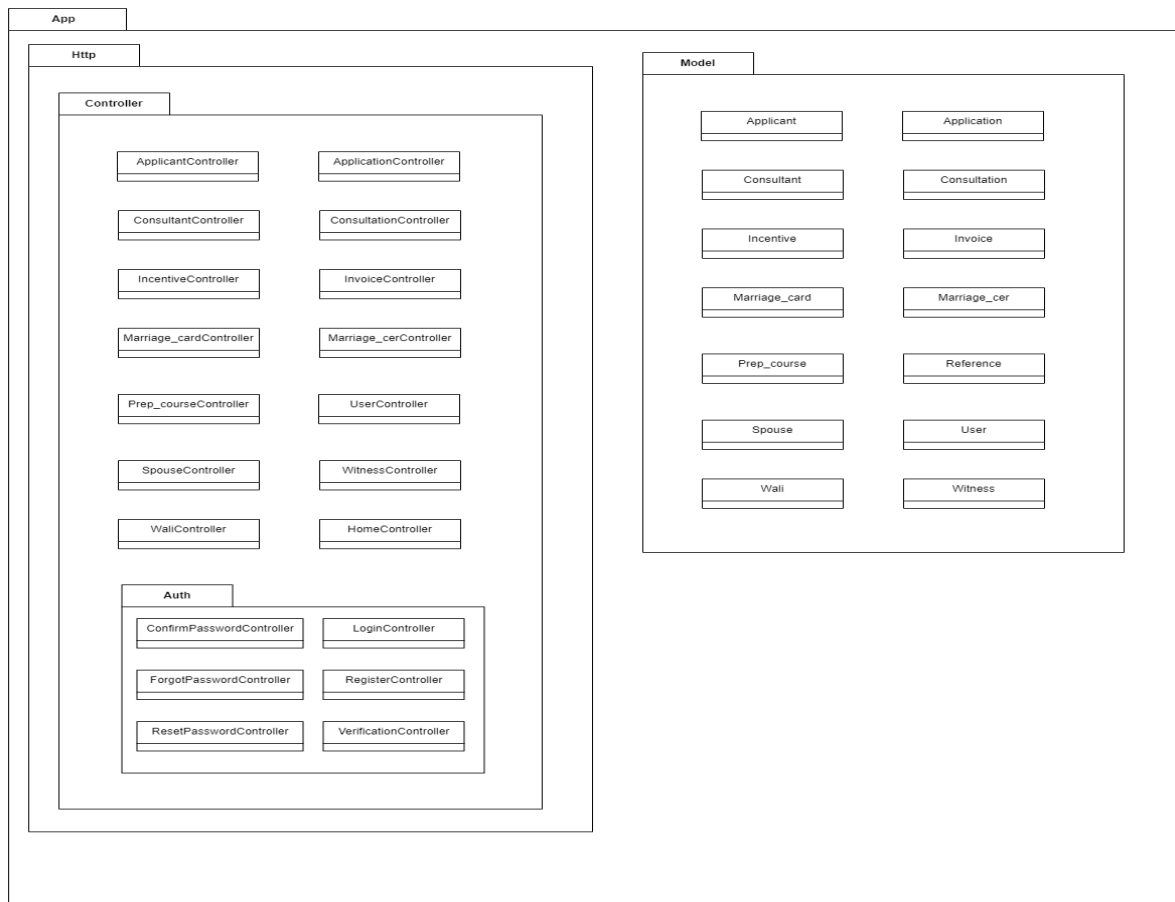
3.1.1.8 Manage consultant [SDD-REQ-800]

Class Name	Description
edit	A view class to allow JAIP staff to update the consultant's information
manage	A view class to allow JAIP staff to manage the list of consultants
create	A view class to allow JAIP staff to enter information of new consultant
show	A class that handles the consultant information

3.1.1.9 Manage Incentive [SDD-REQ-900]

Class Name	Description
apply	A view class that displays the apply incentive page upon request which includes an application form
view	A view class that displays the user's incentive application details, hence information in a specified format
update	A view class that displays the user's information and allows the user to edit their information

3.1.2 Business Services Layer



3.1.2.1 Controller [SDD-REQ-1000]

Class Name	Description
ApplicantController	Handles the logic and actions related to managing applicants in the application. Contains methods for creating, updating, and deleting applicant records, as well as retrieving applicant information.
ApplicationController	Manages the application-related functionality within the application. Contains methods for handling the creation, editing, deletion, and retrieval of application records.
ConsultantController	Responsible for handling the actions and operations related to managing consultants in the application. Contains methods for creating, updating, deleting, and retrieving consultant information.

ConsultationController	Handles the actions and operations associated with managing consultations in the application. Contains methods for scheduling consultations, updating consultation details, and retrieving consultation information.
IncentiveController	Manages the logic and actions related to incentives in the application. Contains methods for creating, updating, deleting, and retrieving incentive records.
InvoiceController	Handles the actions and operations related to managing invoices within the application. Contains methods for creating, updating, deleting, and retrieving invoice records.
Marriage_cardController	Manages the functionality associated with marriage cards within the application. Contains methods for creating, updating, deleting, and retrieving marriage card records.
Marriage_cerController	Handles the actions and operations related to managing marriage ceremonies in the application. Contains methods for creating, updating, deleting, and retrieving marriage ceremony records.
Prep_courseController	Responsible for managing the functionality related to preparation courses in the application. Contains methods for creating, updating, deleting, and retrieving preparation course records.
SpouseController	Manages the functionality and operations related to spouses within the application. Contains methods for creating, updating, deleting, and retrieving spouse records.
UserController	Handles the actions and operations related to managing users within the application. Contains methods for user registration, authentication, updating user information, and retrieving user details.
WaliController	Manages the logic and actions associated with managing walis (guardians) within the application. Contains methods for creating, updating, deleting, and retrieving wali records.
WitnessController	Handles the actions and operations related to managing witnesses within the application. Contains methods for creating, updating, deleting, and retrieving witness records.

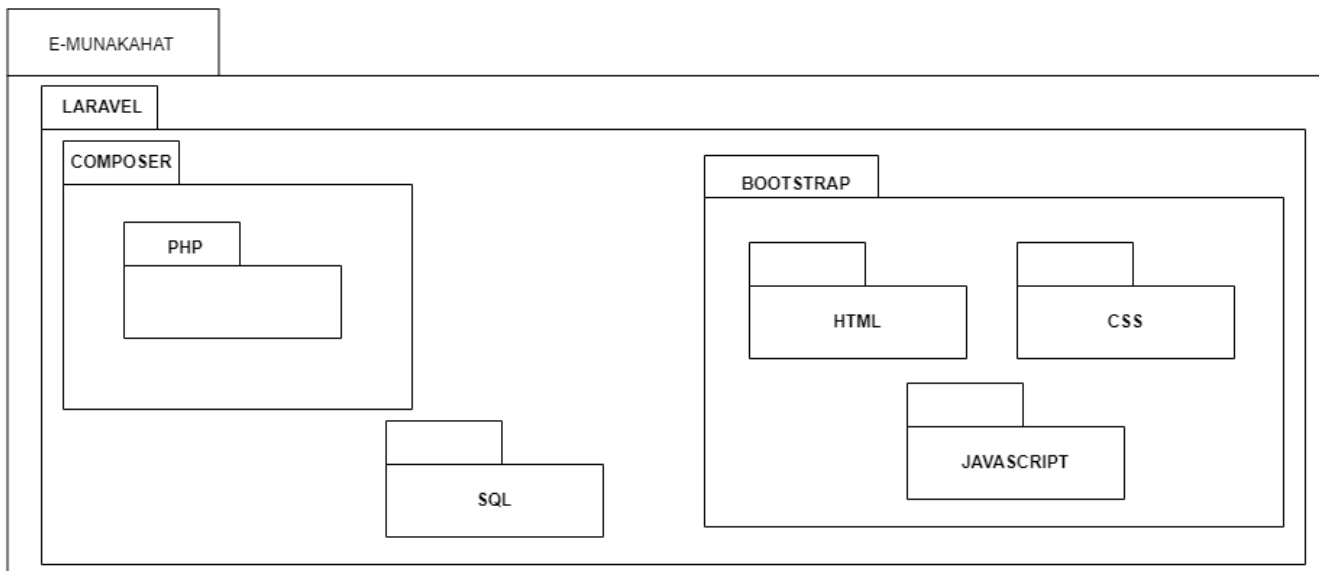
HomeController	Handles the main functionality and actions related to the home page or main dashboard of the application. Contains methods for retrieving and displaying relevant information on the home page.
ConfirmPasswordController	This controller is responsible for handling password confirmations and uses a simple trait to include the behaviour
ForgotPasswordController	This controller is responsible for handling password reset emails and includes a trait that assists in sending these notifications from your application to your users.
LoginController	This controller handles authenticating users for the application and redirecting them to your home screen.
RegisterController	This controller handles the registration of new users as well as their validation and creation.
ResetPasswordController	This controller is responsible for handling password reset requests and uses a simple trait to include this behaviour.
VerificationController	This controller is responsible for handling email verification for any user that recently registered with the application.

3.1.2.2 Model [SDD-REQ-1100]

Class Name	Description
Applicant	This class retrieve and store applicant details
Application	This class retrieve and store application details
Consultant	This class retrieve and store consultant details
Consultation	This class retrieve and store consultation details
Incentive	This class retrieve and store incentive details
Invoice	This class retrieve and store invoice details

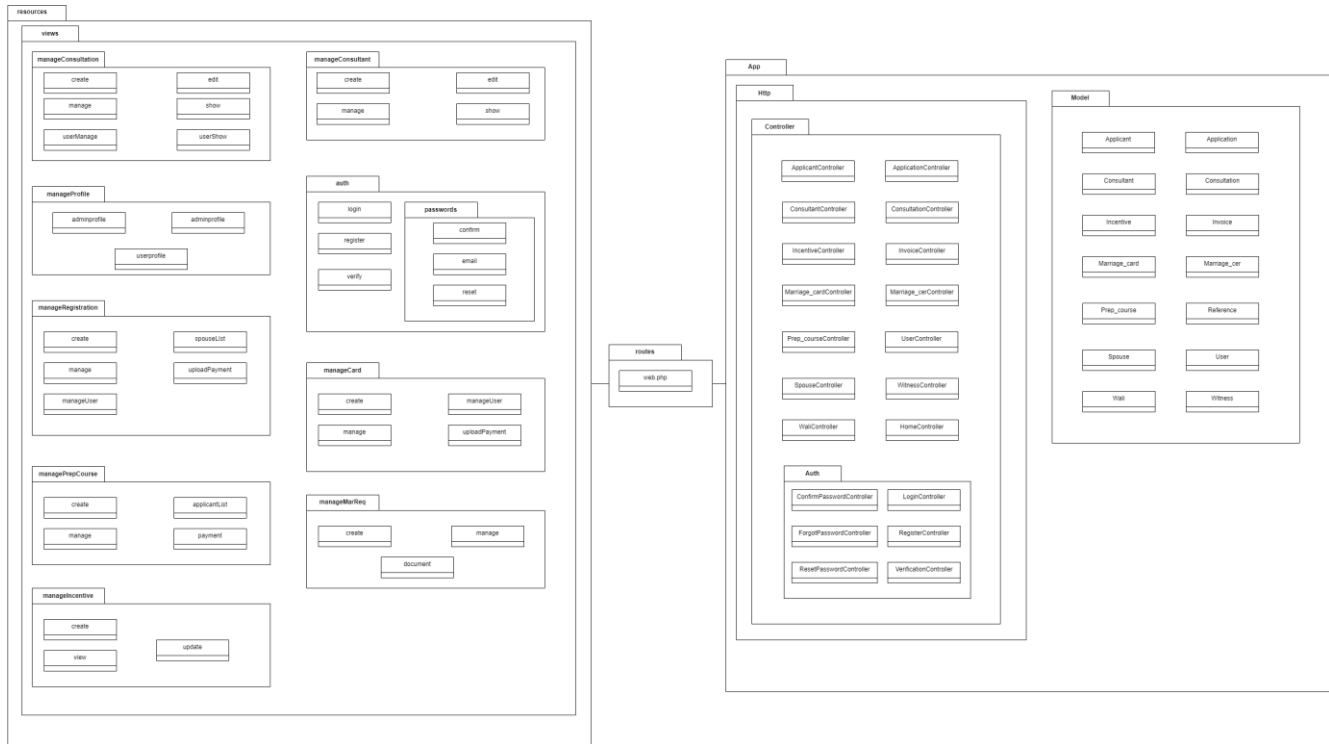
Marriage_card	This class retrieve and store marriage card details
Marriage_cer	This class retrieve and store marriage certificate details
Prep_course	This class retrieve and store preparation course details
Reference	This class retrieve and store reference details
Spouse	This class retrieve and store spouse details
User	This class retrieve and store user details
Wali	This class retrieve and store wali details
Witness	This class retrieve and store witness details

3.1.3 Middleware Layer



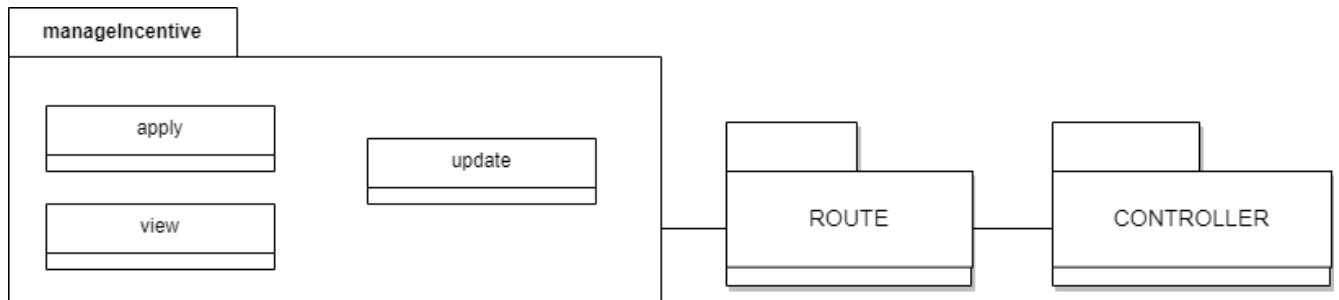
Package Name	Description
HTML	The package that contains of creating and design a website page
CSS	The package that contains of styling of the HTML package
JAVASCRIPT	The package that contains of enhancing the interactivity and functionality of website pages
LARAVEL	PHP web application framework. It contains various packages and classes for building web application.
PHP	The package that contains of code that builds dynamic web applications
SQL	The package that contains the database management

3.2 MVC Package Relationship



4. DETAIL DESIGN

4.1 Manage Incentive [SDD-REQ-100]



4.1 apply [SDD-REQ-101]

Class Type	Boundary Class	
Responsibility	An interface that provides the capability to the user to apply incentives for marriage	
Attributes	Attributes Name	Attributes Type
	id	INT
	name	STRING
	job	VARCHAR (255)
	job_type	VARCHAR (255)
	salary	DOUBLE
	date	DATE (50)
	status	VARCHAR (255)
	heir	VARCHAR (255)
	docs	BLOB
	created_at	TIMESTAMP
	updated_at	TIMESTAMP
Methods	Method Name	Description
	create ()	To apply an incentive by entering the required data into the system.
Algorithm	None	

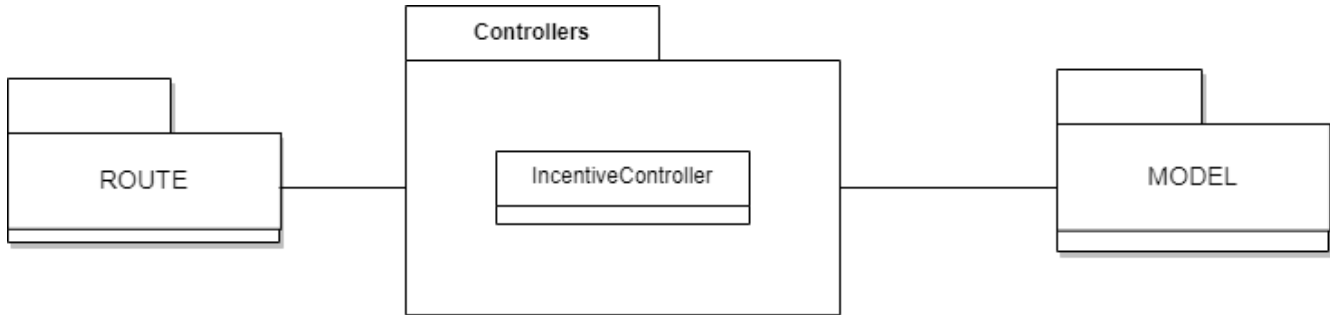
4.1.2 view [SDD-REQ-102]

Class Type	Boundary Class	
Responsibility	An interface that provides the capability for the user to view incentive application status and download reports.	
Attributes	Attributes Name	Attributes Type
	id	INT
	name	STRING
	job	VARCHAR (255)
	job_type	VARCHAR (255)
	salary	DOUBLE
	date	DATE (50)
	status	VARCHAR (255)
	heir	VARCHAR (255)
	docs	BLOB
	created_at	TIMESTAMP
	updated_at	TIMESTAMP
Methods	Method Name	Description
	view ()	To view the status of the incentive application report
Algorithm	None	

4.1.3 update [SDD-REQ-102]

Class Type	Boundary Class	
Responsibility	An interface that provides the capability for the user to update information of the incentive application.	
Attributes	Attributes Name	Attributes Type
	id	INT
	name	STRING
	job	VARCHAR (255)
	job_type	VARCHAR (255)
	salary	DOUBLE
	date	DATE (50)
	status	VARCHAR (255)
	heir	VARCHAR (255)
	docs	BLOB
	created_at	TIMESTAMP
	updated_at	TIMESTAMP
Methods	Method Name	Description
	update ()	To modify the incentive application details in the system.
Algorithm	None	

4.2 Manage Incentive Controller [SDD-REQ-200]

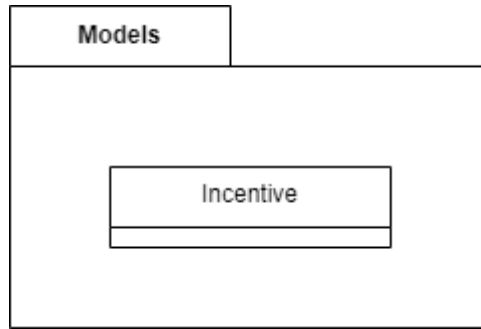


4.2.1 IncentiveController [SDD-REQ-201]

Class Type	Controller Class	
Responsibility	Manages the logic and actions related to incentives in the application. Contains methods for creating, updating, deleting, and retrieving incentive records.	
Attributes	Attributes Name	Attributes Type
	id	INT
	name	STRING
	job	VARCHAR (255)
	job_type	VARCHAR (255)
	salary	DOUBLE
	date	DATE (50)
	status	VARCHAR (255)
	heir	VARCHAR (255)
	docs	BLOB
	created_at	TIMESTAMP
	updated_at	TIMESTAMP
Methods	Method Name	Description
	indexInc ()	To save incentive application data for INCENTIVE table in database.
	view ()	To view the status of the incentive application report
	insert ()	To update incentive application data in database.
	delete ()	To delete specific incentive application data based on 'id' in database.

	view2 ()	To view the incentive application
Algorithm	<pre> BEGIN IF indexInc() is called THEN create object Incentive to connect to Incentive Model RETURN to manageIncentive.apply page ELSE IF view () is called THEN create object update approved status to connect to Incentive Model RETURN to manageIncentive.view page ELSE IF insert () is called THEN create object insert to connect to Incentive Model RETURN to manageIncentive.apply page ELSE IF delete () is called THEN create object delete to connect to Incentive Model RETURN to manageIncentive.view2 page ELSE IF view2() is called THEN create object Incentive to connect to Incentive Model RETURN to manageIncentive.update page. END IF END </pre>	

4.3 Manage Incentive Model [SDD-REQ-300]



4.3.1 incentiveModel [SDD-REQ-301]

Class Type	Entity Class	
Responsibility	This class retrieve and store incentive details	
Attributes	Attributes Name	Attributes Type
	id	INT
	name	STRING
	job	VARCHAR (255)
	job_type	VARCHAR (255)
	salary	DOUBLE
	date	DATE (50)
	status	VARCHAR (255)
	heir	VARCHAR (255)
	docs	BLOB
	created_at	TIMESTAMP
	updated_at	TIMESTAMP
Methods	Method Name	Description
	None	None
Algorithm	None	

5. REQUIREMENT TRACEABILITY

5.1 Manage Incentive [SRS-REQ-900]

SHAMMENE MUNEESSVARAN [CB21018]

<i>SRS-UC-500-REQ-01</i>	<p>Apply the Incentive and Download the Document.</p> <p>The system provides the capability to the user to apply for incentives for marriage and download document incentives.</p>	<p><i>SDD-REQ-101</i></p> <p><i>SDD-REQ-201</i></p> <p><i>SDD-REQ-301</i></p>
<i>SRS-UC-500-REQ-02</i>	<p>Upload Document.</p> <p>The system provides the capability to Upload documents</p>	<p><i>SDD-REQ-101</i></p> <p><i>SDD-REQ-201</i></p> <p><i>SDD-REQ-301</i></p>
<i>SRS-UC-500-REQ-03</i>	<p>View Status</p> <p>The system provides the capability for the user to view incentive application status</p>	<p><i>SDD-REQ-102</i></p> <p><i>SDD-REQ-201</i></p> <p><i>SDD-REQ-301</i></p>
<i>SRS-UC-500-REQ-04</i>	<p>Invalid user information.</p> <p>The system can verify that the user details that are entered by users are invalid, and the system provides the capability to re-enter the information.</p>	<p><i>SDD-REQ-101</i></p>
<i>SRS-UC-500-REQ-05</i>	<p>Missing Documents.</p> <p>The system can verify that the user did not upload any documents and the system provides the capability to re-upload the incentive documents.</p>	<p><i>SDD-REQ-101</i></p>
<i>SRS-UC-500-REQ-06</i>	<p>Valid IC Number length.</p> <p>The system can verify if the IC length is shorter than 12 Numbers.</p>	<p><i>SDD-REQ-101</i></p>
<i>SRS-UC-500-REQ-07</i>	<p>One state citizen (Pahang) can apply for an incentive.</p>	<p><i>SDD-REQ-101</i></p>

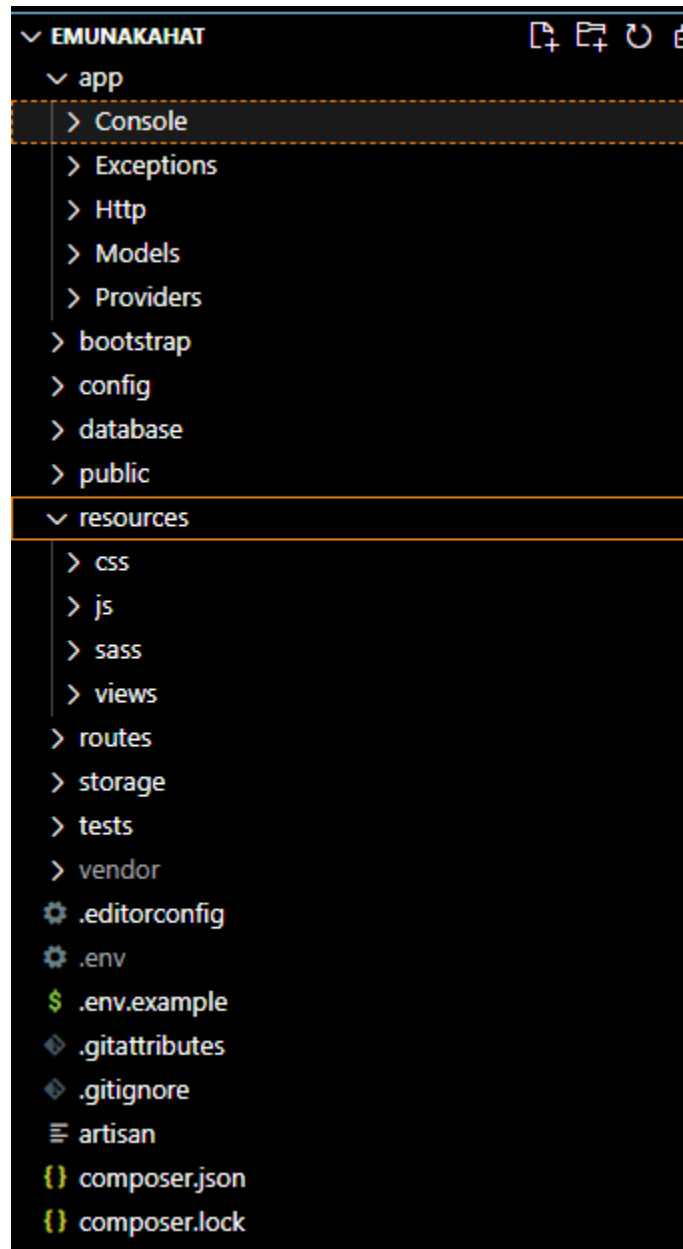
	The system verifies the user is from Pahang state before they can apply for an incentive and proceed to the next step.	
--	--	--

APPENDIX A

Requirements	Description
SRS-UC-500-REQ-01	<p>Apply the Incentive and Download the Document.</p> <p>The system provides the capability to the user to apply for incentives for marriage and download document incentives.</p>
SRS-UC-500-REQ-02	<p>Upload Document.</p> <p>The system provides the capability to Upload documents and download reports.</p>
SRS-UC-500-REQ-03	<p>View Status and Report.</p> <p>The system provides the capability for the user to view incentive application status and download reports.</p>
SRS-UC-500-REQ-04	<p>Invalid user information.</p> <p>The system can verify that the user details that are entered by users are invalid, and the system provides the capability to re-enter the information.</p>
SRS-UC-500-REQ-05	<p>Missing Documents.</p> <p>The system can verify that the user did not upload any documents and the system provides the capability to re-upload the incentive documents.</p>
SRS-UC-500-REQ-06	<p>Valid IC Number length.</p> <p>The system can verify if the IC length is shorter than 12 Numbers.</p>
SRS-UC-500-REQ-07	<p>One state citizen (Pahang) can apply for an incentive.</p> <p>The system verifies the user is from Pahang state before they can apply for an incentive and proceed to the next step.</p>

APPENDIX B

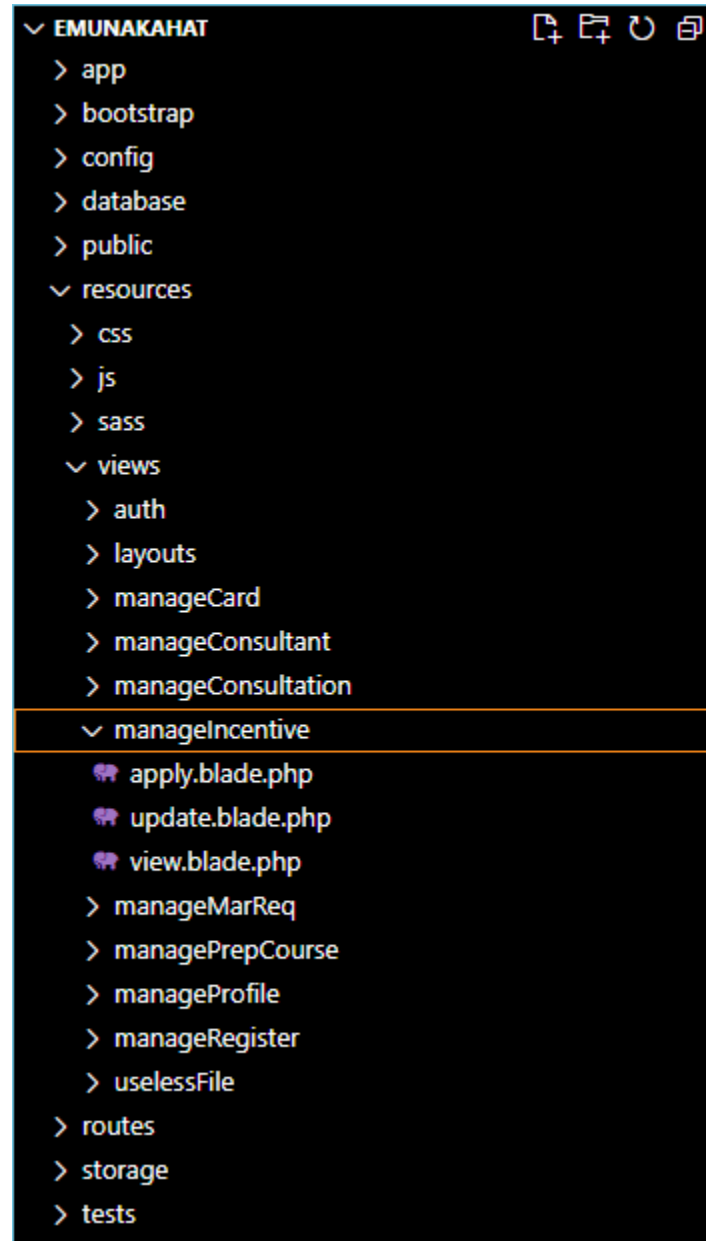
General Architecture (Chapter 3)



System Name	Layers		
emunakahat	resources	Views	
	app	Http	Controllers
		Model	
	route		

APPENDIX C1

manageIncentive (Chapter 4) – views layer



Layer	Package	Class
views	manageIncentive	Apply Update view

APPENDIX C2

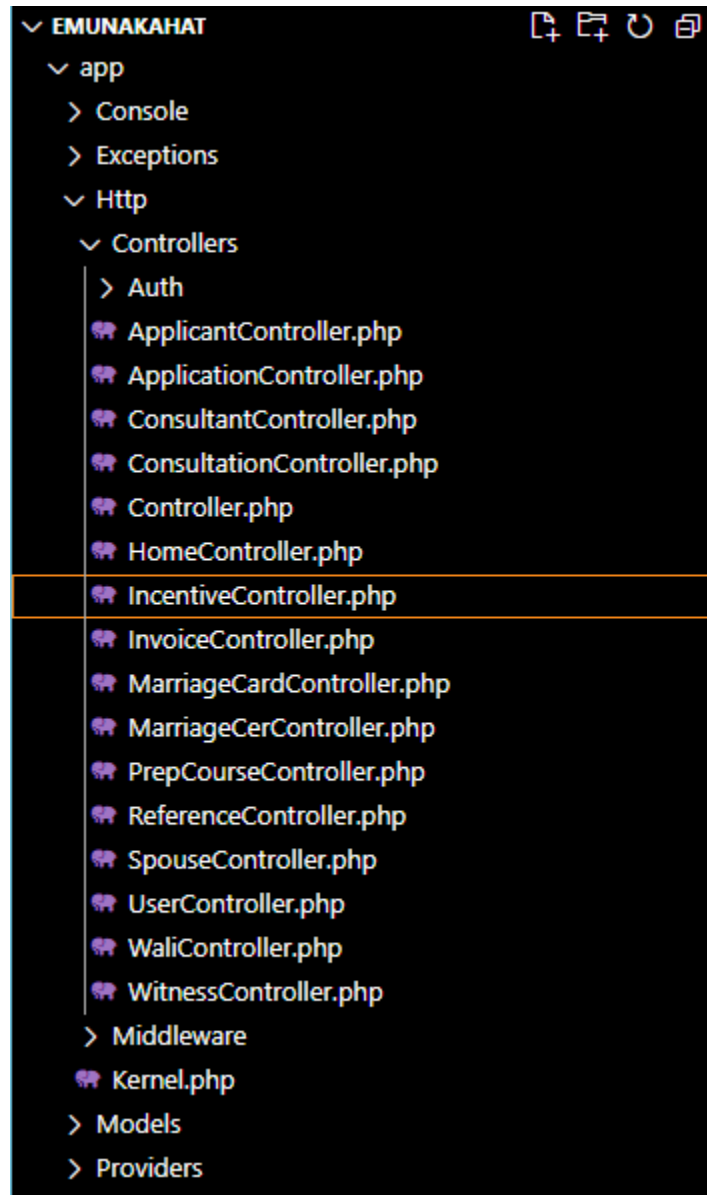
manageIncentive (Chapter 4) – design patern layer

```
▼ routes
  🐘 api.php
  🐘 channels.php
  🐘 console.php
  🐘 web.php
```

Layer	Class
routes	web

APPENDIX C3

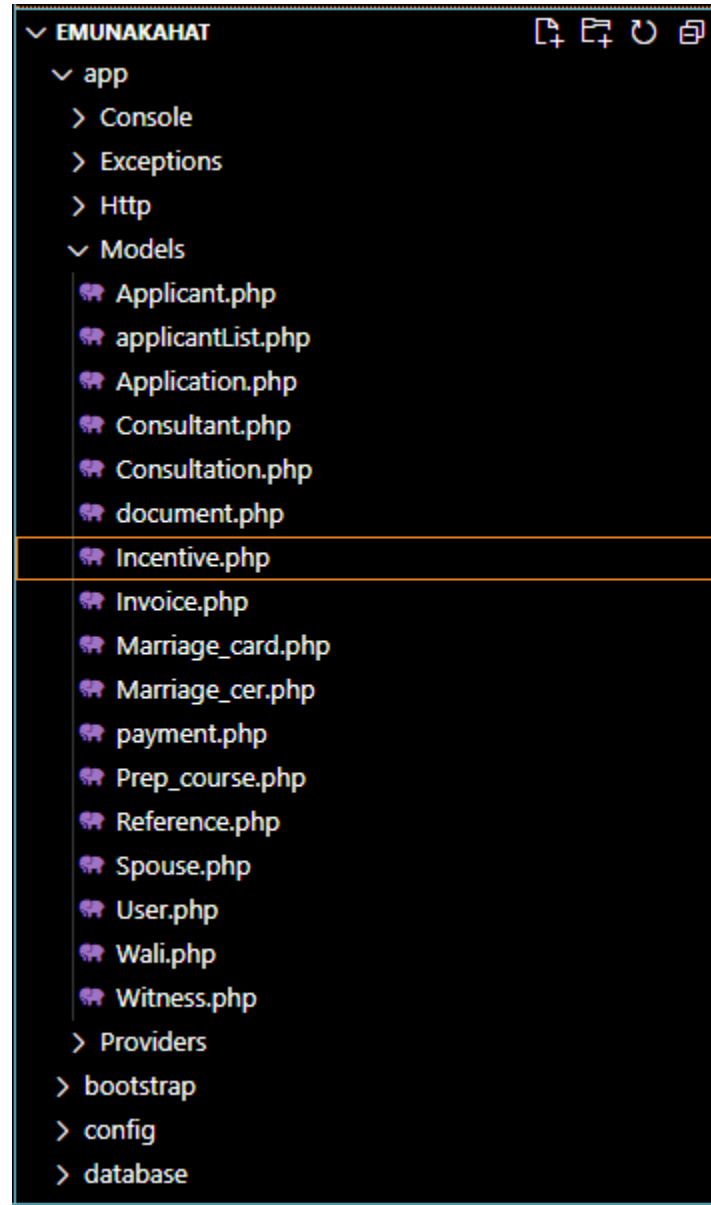
manageIncentive (Chapter 4) – controller layer



Layer	Class
Controllers	IncentiveController

APPENDIX C4

manageIncentive (Chapter 4) – model layer



Layer	Class
Models	Incentive