ADNU Employee Research System: An online Research Showcase for ADNU Employees

Master of in

Master of in

Master of in

Thesis submitted to the faculty of the
Department of Ateneo de Naga University
College of Computer Studies, Ateneo de Naga University
in partial fulfillment of the requirements for their respective
Bachelor of Science degrees

Allan A. Sioson, Ph.D., Chairman
First Panel Member
Second Panel Member
Third Panel Member

June 12, 2019
Naga City, Philippines

Keywords: keywordOne, keywordTwo, keywordThree

ADNU Employee Research System: An online Research Showcase for ADNU Employees

by

, , and

Committee Chairman: Allan A. Sioson, Ph.D.

Department of Computer Science

(ABSTRACT)

Write your abstract here. Preferably around 300 words.

We dedicate this project to our family for their support, love, and encouragement. Also, we would like to thank our adviser, Ms. Marianne P. Ang for her guidance and support throughout the development of our senior project.
We express our heartfelt gratitude to our God above all.

ACKNOWLEDGEMENTS

We would like to thank God for giving us determination, strength, knowledge, and guidance to finish this project.

Also, thanks to our family for the undying support throughout our college years; We would like to express our gratitude to our friends for motivating us during the development of our project;

For the trust, accommodation, and for the given the opportunity to work with them, gratefulness is extended to Sir Marlon M. Decillo, Compensation and Benefits Officer, Ma'am Leanne Gemelly B. Briones, Systems Administration Specialist, Jon-Jon S. Elicay, Checker;

Special thanks to Ma'am Myllan B. Toledana, the HRMO Director, for approving our request regarding the data we need for our senior project.

To our adviser, Ms. Marianne P. Ang, to our panelist, Mr. Frederick D. Olan˜o, Ms. Jelly P. Aureus, and Mr. Rey Herman R. Vidallo, thank you for giving us guidance, pieces of advice, constructive comments, and suggestions for our project. Without them, this project would not have been possible. Also, to our proofreader Ms. Vanessa M. Abay for looking into our study to allow our research information flow smoothly.

We thank everyone who helped us finish this research.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

1.1 Project Context

The Ateneo de Naga University has employees that are partaking in research but without an online repository website to store and showcase their research.The rise in digital services where the information can be stored through digital devices with the use of internet is efficient.The the traditional way of searching information is too time consuming.The need of digital libraries are needed so that there is an easy access to research.[1] .

1.2 Purpose and Description

The purpose of the proposed project is to create a web application that showcases the research paper made by the researchers of all the different departments and colleges of Ateneo de Naga University. The employees of Adnu will be able to upload their research papers through the system to make a digital copy of their work for other users to see.

The AdNU Employee Research System will collect and store research paper of employees of the Ateneo de Naga University.The search function is not only limited to the title of projects but can also be used to search other projects by an author while also be able to sort by category such as research type, date published, title, etc.

1.3 Objectives

1.3.1 General Objectives

The goal of the project is to develop a web-based repository of research papers for the employees of all department and colleges of the Ateneo de Naga University and will be managed by the URC as the admin.

1.3.2 Specific Objectives

- Admin will be able to create accounts for verified employees.
- Employees of Ateneo de Naga University to be able to upload their research paper using the system
- All users to be able to view a dashboard that contains all research paper made by the employees of AdNU
- Create Notification System
- To export backup data and report summaries.
- To create an advance search function.
- Create database to store the data
- To conduct a user testing and provide an evaluation
- To fully develop the proposed system and deploy to URC

1.4 Scope and Limitations

The System will be develop as a web-based online repository of research papers of the employees of all departments and colleges in Ateneo de Naga University.

- Guest users to be restricted to viewing only abstract paper
- Faculty can input details regarding their completed research, presented research and published research.

- The inputted details can be summarized into citations(based on APA 6th edition format)
- The URC can sort the inputted information based on:
    1. year of completion, year of presentation, year of publication
    2. type of publication (e.g. book, journal, etc.)
    3. faculty name
    4. type of research work (e.g. completed, presented, published)
    5. department / unit
- Employees and Admin to manage their research papers (edit,delete)
- The URC can unsubmit and email faculty if details provided are not appropriate

This System is limited that only employees of Ateneo de Naga University can upload research projects and users that are not employees can only view the projects. The research projects will be following the APA format as the standard and PDF as the file standard.

Chapter 2

Methodology

2.1 Software Development Model

The waterfall process model will be used as a development life cycle of the project.
In the first phase, Requirement Analysis, The requirements of the project were gathered and analyzed as the foundation of the proposal. An interview will be conducted to the URC and All Departments of the Ateneo de Naga University in order to gain information about the current existing processes of the Information System of the Research Papers and analyze the scope of the project that are going to be developed.
In the Second phase, Design, The initial start of the software architecture in which to create the essential diagrams and models that will be used in the project. The design functions and features are described with the inital user interface.

In the Third phase, Development. In the Chapter 3 in which the technical background listed are going to be used in this phase. The coding for each module that are described in the objectives will begin.
In the Fourth phase, Testing. After the development of the System, A testing will be conducted to the users to be able to test the application to look for faults and receive feedback. The debugging will commence based on the results from testing.
Lastly, Deployment. The system are expected to be fully functional and met the requirements of

the project then it will be deployed to the URC.

2.1.1 Data Dictionary

2.2 System Testing

The data collected by the researchers were analyzed in this part of the project, in order to develop the proposed system and achieve its goal. The above diagram serves as a guide to the implementation of the proposed system and to communicate with our client ADNU HRMO to give us ideas about how we will evaluate our system. The modules created in the proposed system would act as a substitute in the organization for their current system processes. These are the things that underwent unit testing: the checking of attendance using mobile application, confirmation of absences using the web application, changing of venue/classroom, and approving of the absences of the faculty by the chairperson or the dean of the university.

2.2.1 Methods

This section details test roles and major activities for the testing.

Personnel

The experiment team will be composed of three people: The test monitor and 2 observers. The test monitor will be responsible for interaction with test participant. Each observer will be assigned two users that they have to observe in the duration of the test. They have to keep record of feedback and all issues encountered. During the test itself, they have to work independently, keenly observing their corresponding users.

Setup of Testing Environment

The test will take place in HR Office and in the classroom. The user will be provided with a seat, laptop and a mobile phone, if the user has none. The test monitor will be seated next to the participant. The observers will be standing or sitting at far enough from the users to not make the users uncomfortable but close enough to be able to see the users' screens and hear their comments.

Selection of Participants

The participant should be a faculty, an admin, and checker. They should be able read and use laptop and mobile phone.

Note: that it is necessary to filter participants even before the test itself. You should give your candidate participants a questionnaire that determines whether they fit your criteria for a test participant at all (see Sample Screening Questionnaire). If they do not fit your criteria, you should not use them for the test. In this case, for example, anyone who is under 18 or over 40 cannot be part of the test. Anyone who has not taken any English classes could not be part of the test.

Scheduling of the Test

The task monitor will set an appointment to the HR and some faculty member to conduct the testing at their convenient time.

Welcoming the Participants

When the test participants enter the test venue, the test monitor will ask them to sit in the designated area. The test monitor introduces the project team and explains the purpose of the test.

Demographic Questionnaire

The test monitor administers a demographics questionnaire (see Demographics Questionnaire) to gather some relevant facts about the users.

Task start and observation

After the demographics questionnaire, the test monitor can ask the user to begin using the service. The observers begin their observations when the timer starts (See Timing 60).

During the test, the test monitor can answer questions or provide assistance, if requested.

SUS Questionnaire

After the timer stops, the test monitor should ask the user to stop using the service. The test monitor should then administer the SUS Questionnaire (see SUS Questionnaire).

Debriefing

After the SUS Questionnaire, the test monitor interviews the users based on the questions on the Debriefing questionnaire (see Debriefing Questionnaire).

Thanking the user The test monitor thanks the users.

Summary of results

The group should compute for the SUS Score (See Computing SUS) and summarize the results based on the report template (see Summary of Results).

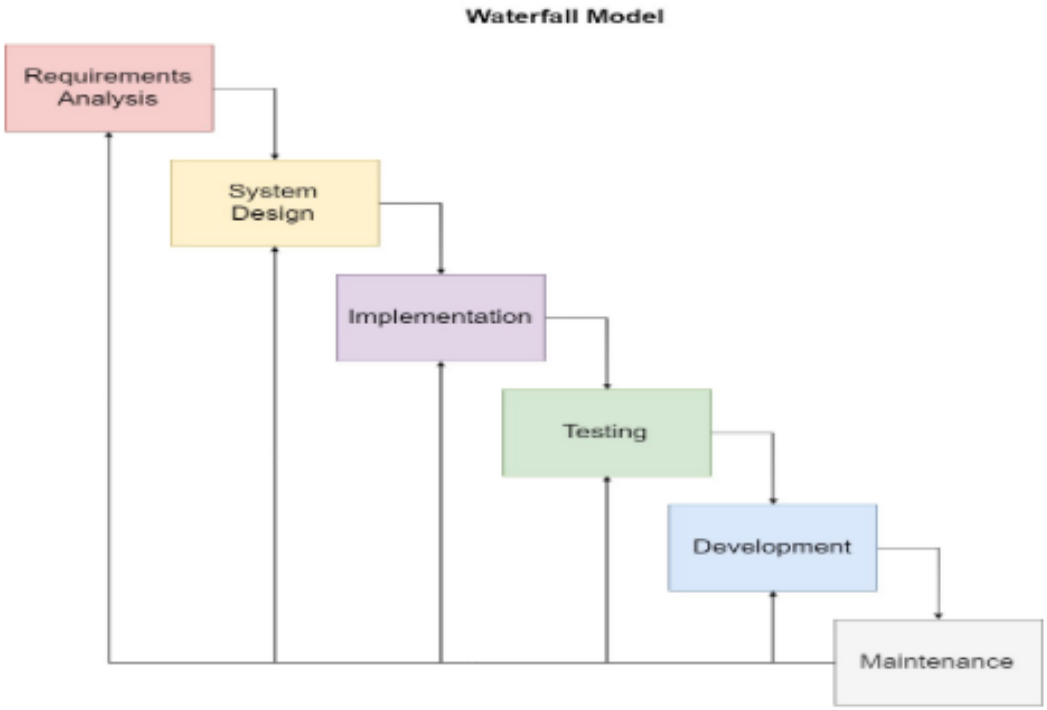

Figure 2.1: Waterfall Model for AERS

AUTHOR					
Attribute Name	Data Type	Constraint	Size	Nullable	Description
author_id	VARCHAR	PRIMARY	9	No	Faculty ID, Primary key of the table
user_id	VARCHAR		64	No	First Name of the Faculty
publication_id	VARCHAR		64	No	Middle Name of the Faculty
first_Name	VARCHAR		64	No	Last Name of the Faculty
middle_initial	VARCHAR		24	No	Position of the Faculty
last_name	VARCHAR	UNIQUE	64	No	Email of the Faculty
is_employee	VARCHAR	UNIQUE	11	No	Contact Number of the Faculty

Table 2.1: Author t Table

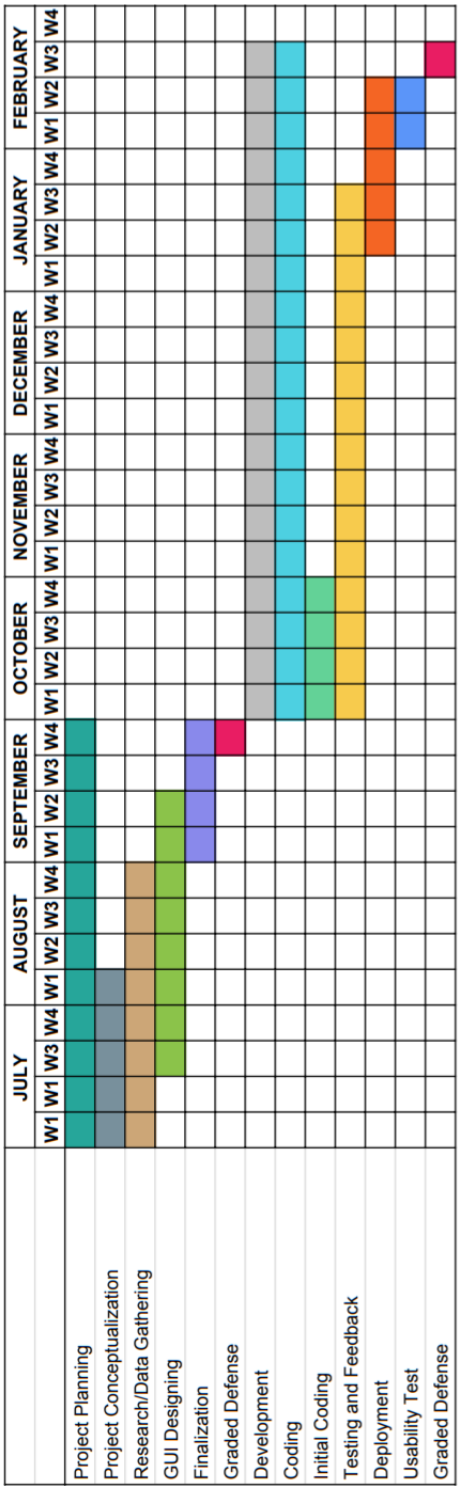

Figure 2.2: Gannt Chart

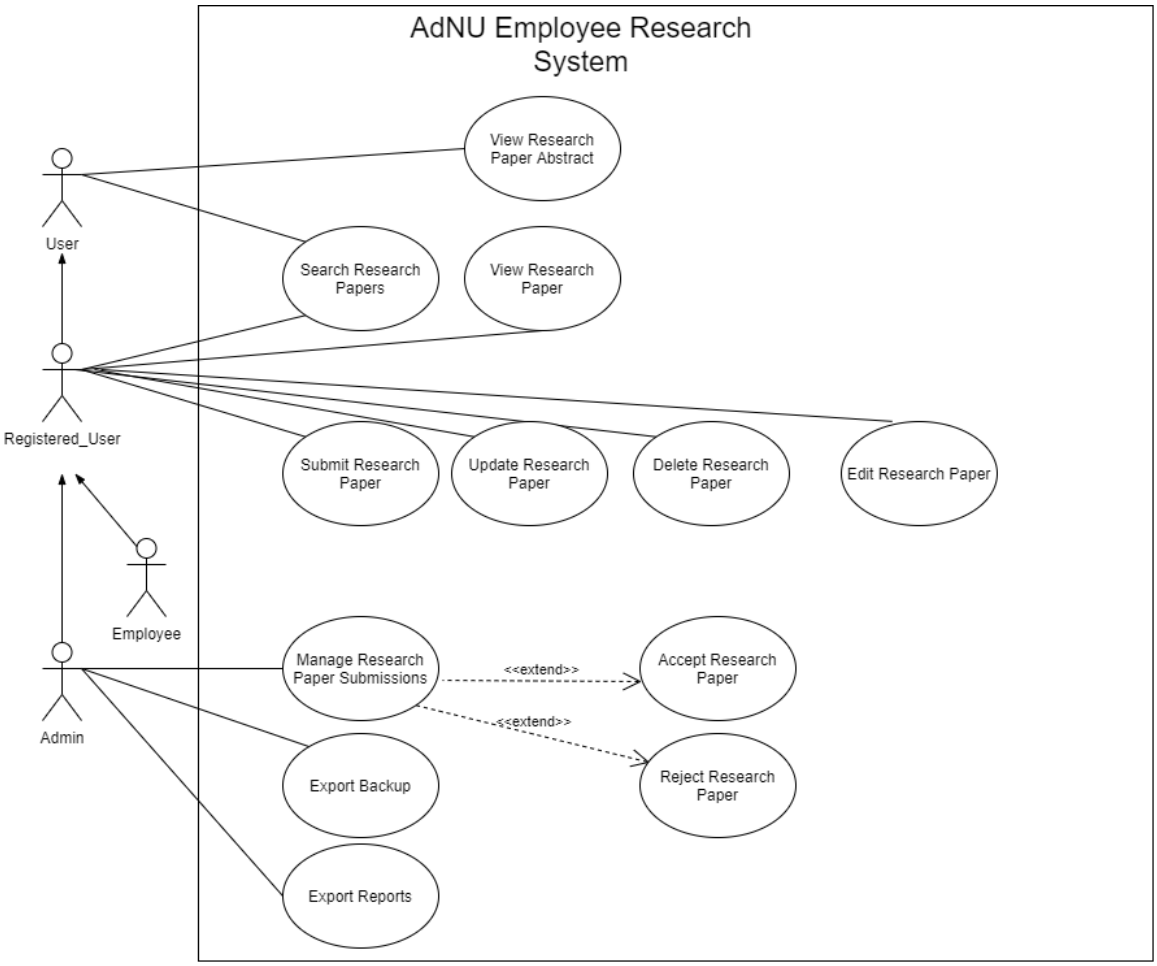

Figure 2.3: Use-Case Diagram of ADNU Employee Research System

USER					
Attribute Name	Data Type	Constraints	Size	Nullable	Description
user_id	VARCHAR	PRIMARY	9	No	Primary key of the table
username	VARCHAR	FOREIGN	9	Yes	Checker ID of a staff
First_Name	VARCHAR		64	No	First name of the staff
Middle_Name	VARCHAR		64	No	Middle Name of the staff
email	VARCHAR		64	No	Surname of the staff
Password	VARCHAR		256	No	Password of the account
department	VARCHAR		1,0	No	Account Privilege
$\text{contact}_n umber$	Timestamp		2	Yes	Timestamp of the last login
user_type	VARCHAR		64	No	type of user

Table 2.2: User table

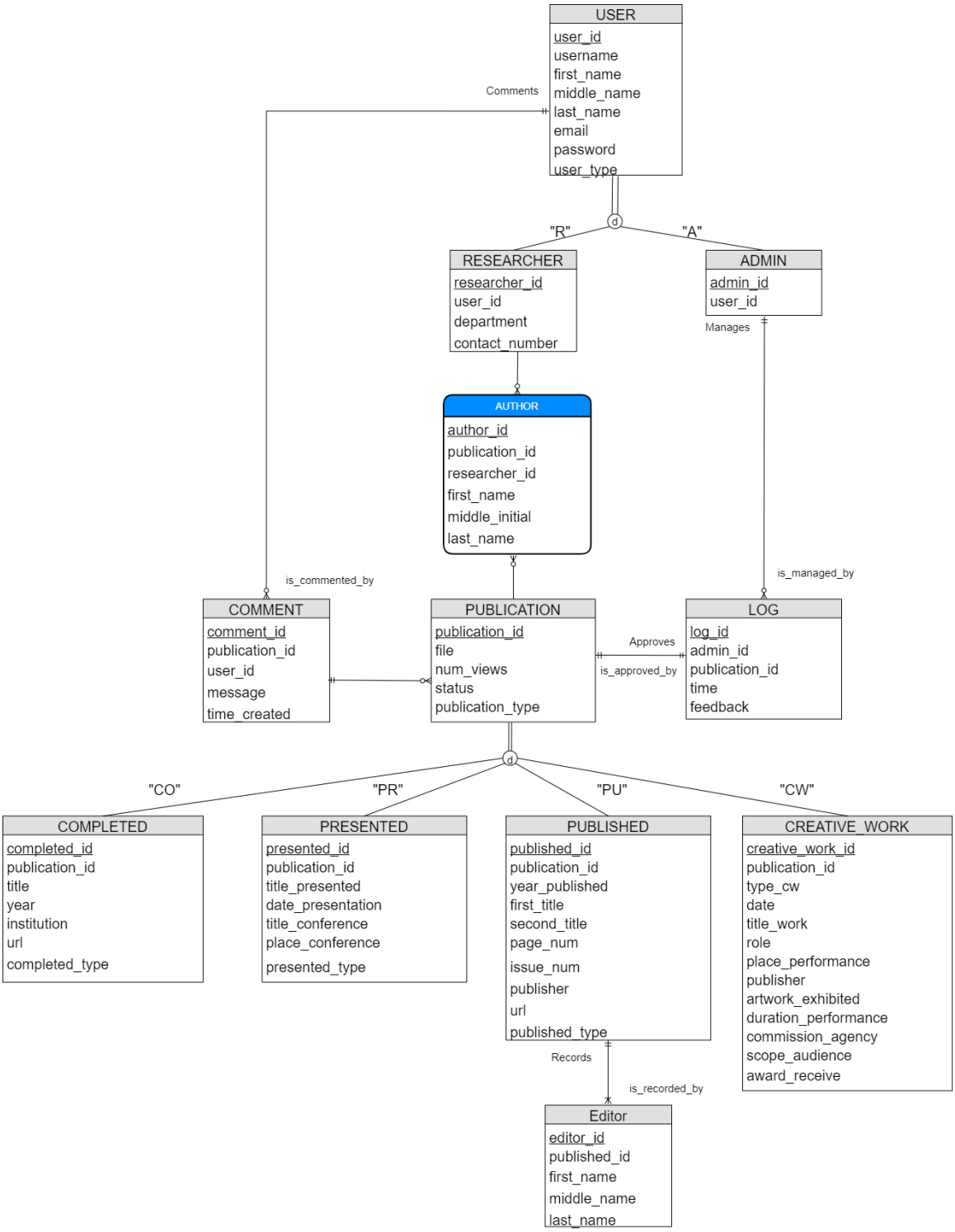

Figure 2.4: Entity Relational Diagram of Adnu Employee Research System

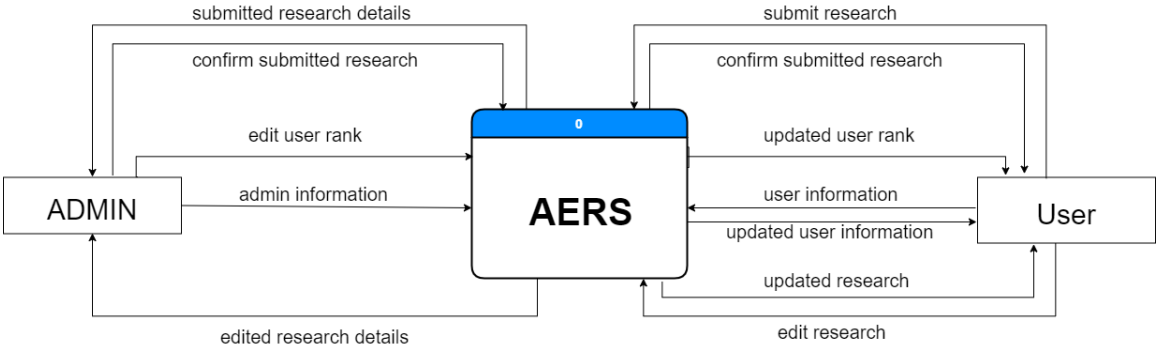

Figure 2.5: Context Level Data Flow Diagram

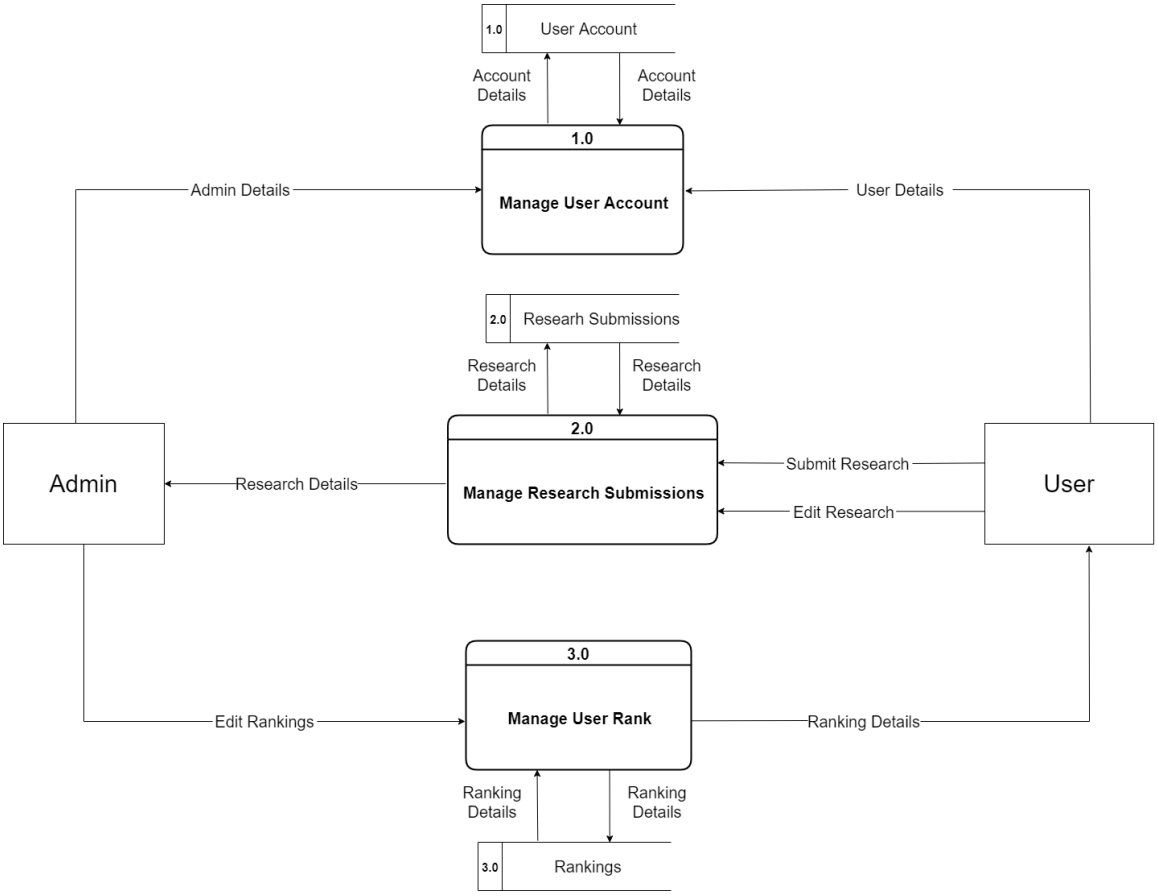

Figure 2.6: Level 1 Data Flow Diagram

COMMENT					
Attribute Name	Data Type	Constraints	Size	Nullable	Description
comment_id	VARCHAR	PRIMARY	9	No	Primary key of the checker
publication_id		1,0	Yes	Assign Route for the checker	
user_id	VARCHAR		32	YES	Access Token of Checker
message	VARCHAR		32	YES	message of the comment
time_created	Timestamp		32	YES	time when comment was pos

Table 2.3: Web's Checker Table

COMPLETED					
Attribute Name	Data Type	Constraints	Size	Nullable	Description
completed_id	VARCHAR	PRIMARY	15	No	Room ID from MIS
publication_id	NUMBER		3	Yes	Building ID of the Room
title	VARCHAR		64	Yes	System Assigned Name
year	VARCHAR		64	Yes	year completed
institution	VARCHAR		64	Yes	institution completed
location	VARCHAR		64	Yes	location completed
url	VARCHAR		64	Yes	url of completed work
completed_type	VARCHAR		64	Yes	type completed

Table 2.4: Web's Room Table

CREATIVE_WORKS					
Attribute Name	Data Type	Constraints	Size	Nullable	Description
cw_id	VARCHAR	PRIMARY	9	No	Primary Key of the table
publication_id	VARCHAR	FOREIGN	9	Yes	Foreign key of the table
type_cw	VARCHAR	FOREIGN	9	Yes	Foreign key of the table
month_year	VARCHAR	FOREIGN	9	Yes	Foreign key of the table
title_work	VARCHAR	FOREIGN	9	Yes	Foreign key of the table
role	VARCHAR	FOREIGN	9	Yes	Foreign key of the table
place_performance	DATE			No	Date from the attendance of faculty
publisher	TIMESTAMP		0	Yes	Checking of attendance 15 minutes aft
artwork_exhibited	TIMESTAMP		0	Yes	Checking of attendance 20 minutes bef
duration_performance	VARCHAR		64	Yes	This will act as the evidence when the
commission_agency_File	VARCHAR		64	Yes	
scope_audience	CHAR		3	Yes	Salary deduction from their absences
award_received	VARCHAR		12	Yes	Will determine the final remarks wheth
Notified	NUMBER			Yes	Notifier flag
Validated	NUMBER			Yes	Will be a flag if the attendance was va

Table 2.5: Web's Faculty Attendance Table

EDITOR					
Attribute Name	Data Type	Constraints	Size	Nullable	Description
editor_id	VARCHAR	PRIMARY	15	No	Room ID from MIS
publication_id	NUMBER		3	Yes	Building ID of the Room
first_name	VARCHAR		64	Yes	System Assigned Name
middle_name	VARCHAR		64	Yes	year completed
last_name	VARCHAR		64	Yes	institution completed

Table 2.6: Web's Room Table

NOTIFICATION					
Attribute Name	Data Type	Constraints	Size	Nullable	Description
notification_id	VARCHAR	PRIMARY	15	No	Room ID from MIS
user_id	NUMBER		3	Yes	Building ID of the Room
publication_id	VARCHAR		64	Yes	System Assigned Name
type	VARCHAR		64	Yes	year completed
time	VARCHAR		64	Yes	institution completed
status	VARCHAR		64	Yes	location completed
url	VARCHAR		64	Yes	url of completed work

Table 2.7: Web's Room Table

PRESENTED					
Attribute Name	Data Type	Constraints	Size	Nullable	Description
presented_id	VARCHAR	PRIMARY	15	No	Room ID from MIS
publication_id	NUMBER		3	Yes	Building ID of the Room
title_presented	VARCHAR		64	Yes	System Assigned Name
date_presentation	VARCHAR		64	Yes	year completed
institution	VARCHAR		64	Yes	institution completed
title_conference	VARCHAR		64	Yes	location completed
place_conference	VARCHAR		64	Yes	url of completed work
present_type	VARCHAR		64	Yes	type completed

Table 2.8: Web's Room Table

PUBLICATION					
Attribute Name	Data Type	Constraints	Size	Nullable	Description
publication_id	VARCHAR	PRIMARY	15	No	Room ID from MIS
submittor	NUMBER		3	Yes	Building ID of the Room
file	VARCHAR		64	Yes	System Assigned Name
date_submission	VARCHAR		64	Yes	year completed
abstract	VARCHAR		64	Yes	institution completed
num_views	VARCHAR		64	Yes	location completed
status	VARCHAR		64	Yes	url of completed work
feedback	VARCHAR		64	Yes	type completed
publication_type	VARCHAR		64	Yes	type completed

Table 2.9: Web's Room Table

PUBLISHED					
Attribute Name	Data Type	Constraints	Size	Nullable	Description
published_id	VARCHAR	PRIMARY	15	No	Room ID from MIS
publication_id	NUMBER		3	Yes	Building ID of the Room
year_published	VARCHAR		64	Yes	System Assigned Name
title1	VARCHAR		64	Yes	year completed
title2	VARCHAR		64	Yes	institution completed
num_views	VARCHAR		64	Yes	location completed
page_num	VARCHAR		64	Yes	url of completed work
publisher	VARCHAR		64	Yes	type completed
url	VARCHAR		64	Yes	type completed
published_type	VARCHAR		64	Yes	type completed

Table 2.10: Web's Room Table

basicstyle=, numbers=left, stepnumber=1, showstringspaces=false, tabsize=1, breaklines=true, breakatwhitespace=false,

Appendix A

Code Listing

A.1 Mobile Application

A.1.1 main.dart

import 'package:ahcfams$_m$obile/bloc/theme−bloc.bloc.dart'; $import'package : ahcfams_mobile/ui/login.page.dart'; i$
$flutter/material.dart'; import'package : flutter/services.dart'; import'package : provider/provider.dart'; import'p$
$shared_preferences/shared_preferences.dart'$;

Future main() async WidgetsFlutterBinding.ensureInitialized(); SharedPreferences prefs = await SharedPreferences.getInstance(); await SystemChrome.setPreferredOrientations([DeviceOrientation.portraitUp]); runApp(MyApp(prefs: prefs));

class MyApp extends StatelessWidget final SharedPreferences prefs; MyApp(this.prefs);

@override Widget build(BuildContext context) return MultiProvider(providers: [ChangeNotifierProvider¡ThemeBloc¿.value(value: ThemeBloc('ateneoTheme'),)], child: MaterialApp(title: 'AHCFAMS Mobile', debugShowCheckedModeBanner: false, theme: ThemeData(brightness: Brightness.light, primarySwatch: Colors.blue,), home: LoginPage(prefs),),);

A.1.2 home.page.dart

import 'package:ahcfams$_m$obile/model/user.model.dart'; $import'package : ahcfams_mobile/ui/dashboard.page.dart'$
$ahcfams_mobile/ui/pending−task.page.dart'; import'package : ahcfams_mobile/ui/settings.page.dart'; import'pack$

$ahcfams_mobile/ui/sync.page.dart'; import'package : ahcfams_mobile/widget/icon-badge.widget.dart'; import'pac$
$flutter/cupertino.dart'; import'package : flutter/material.dart';$

class HomePage extends StatefulWidget final User currentUser; HomePage(this.currentUser);

@override $_HomePageStatecreateState() =>_H omePageState();$

class $_HomePageStateextendsState < HomePage > withSingleTickerProviderStateMixinTabController_contr$

A.1.3 attendance-list.page.dart

import 'dart:io';

import 'package:ahcfams$_mobile/bloc/theme-bloc.bloc.dart'; import'package : ahcfams_mobile/common/shared-$
$functions.common.dart'; import'package : ahcfams_mobile/db/database.helper.dart'; import'package :$
$ahcfams_mobile/model/faculty-attendance-list.model.dart'; import'package : ahcfams_mobile/model/note.model$
$ahcfams_mobile/ui/faculty-details.page.dart'; import'package : ahcfams_mobile/ui/image-view.page.dart'; impor$
$flutter/cupertino.dart'; import'package : flutter/material.dart'; import'package : image_picker/image_picker.dart';$
$provider/provider.dart';$

class AttendanceListPage extends StatefulWidget final String buildingName; AttendanceList-Page(this.buildingName);

@override $_AttendanceListPageStatecreateState() =>_A ttendanceListPageState();$

class $_AttendanceListPageStateextendsState < AttendanceListPage > List < FacultyAttendanceList > facult$

List¡Note¿ notes = List¡Note¿(); int notesLength;

final GlobalKey¡ScaffoldState¿ $_scaffoldKey = GlobalKey < ScaffoldState > ();$

@override void initState() super.initState(); $_init();$

$_init()asyncsetState(() =>_l oading = true); facultyAttendance = awaitDatabaseHelper.db.getAttendanceList($

@override Widget build(BuildContext context) return Scaffold(key: $_scaffoldKey, appBar :_a$
$ppBar(), body :_l oading?Center(child : CircularProgressIndicator(),) :_s how(),);$

$_appBar()returnAppBar(title : Text('AttendanceList'), actions :< Widget > [IconButton(icon : Icon(Icons.sor$
$_show()returnListView.builder(itemCount : facultyAttendanceLength, itemBuilder : (context, index)print(fa$
);

$_attendance(FacultyAttendanceListf)returnContainer(child : Card(elevation : 1, child : InkWell(child : Conta$
$_showNoteDesc(StringnoteId)Stringdesc; for(varfinnotes)if(f.noteId == noteId)desc = f.description;$

return desc;

$_changeNote(StringfacultyAttendanceId, StringnoteId)asyncprint(awaitDatabaseHelper.db.changeNote(facu$

$_{a}bsent(String_{a}ttendanceId)asyncawait_{o}penCamera(); await_{s}aveImage(_{a}ttendanceId); awaitDatabaseHelper.d$

$_{i}nit(); print(DatabaseHelper.db.getFacultyAttendance(_{a}ttendanceId));$

$_{p}resent(String_{a}ttendanceId)asyncawaitDatabaseHelper.db.markPresence(_{a}ttendanceId, SharedFunctions().t$

$_{i}nit(); print(DatabaseHelper.db.getFacultyAttendance(_{a}ttendanceId));$

$_{o}penCamera()asyncvar_{p}icture = awaitImagePicker.pickImage(source : ImageSource.camera, imageQuality$

$_{s}aveImage(String_{a}ttendanceId)asyncfinalStringpath = awaitSharedFunctions().localPath; finalStringfile$

await imageFile.copy('$path$/fileName.jpg');

await DatabaseHelper.db.saveImageFileName($_{a}ttendanceId, fileName +' .jpg'$);

A.1.4 building-list.page.dart

import 'package:ahcfams$_{m}obile/db/database.helper.dart'; import'package : ahcfams_{m}obile/ui/attendance-$
$list.page.dart'; import'package : flutter/cupertino.dart'; import'package : flutter/material.dart';$

class BuildingListPage extends StatefulWidget BuildingListPage(Key key) : super(key: key);

@override $_{B}uildingListPageStatecreateState() =>_{B}uildingListPageState();$

class $_{B}uildingListPageStateextendsState < BuildingListPage >$

final GlobalKey¡ScaffoldState¿ $_{s}caffoldKey = GlobalKey < ScaffoldState > (); bool_{l}oading =$
$true;$

List $_{b}uildingList; int_{b}uildingLength;$

@override void initState() super.initState(); $_{i}nit();$

$_{i}nit()async_{b}uildingList = awaitDatabaseHelper.db.getBuildings(); setState(() =>_{l} oading = false);_{b}uildingL$

@override Widget build(BuildContext context) return Scaffold(key: $_{s}caffoldKey, appBar :_{a}$
$ppBar(), body :_{l} oading?Center(child : CircularProgressIndicator()) :_{s} how());$

$_{a}ppBar()returnAppBar(title : Text('BuildingList'), backgroundColor : Colors.blue$[800],);

$_{s}how()return_{b}uildingLength == 0?Center(child : Text('NoDataAvailable'),) : ListView.builder(itemCount :_{b}$

$_{b}uilding(String_{b}uildingName)returnContainer(child : Card(elevation : 2, child : ListTile(leading : Icon(Icon$

A.1.5 confirmation-notice.page.dart

import 'dart:io'; import 'dart:typed$_{d}ata'$;

import 'package:ahcfams$_{m}obile/common/shared - functions.common.dart'; import'package :$
$ahcfams_{m}obile/db/database.helper.dart'; import'package : ahcfams_{m}obile/model/confirmation-$

$notice-list.model.dart'; import'package : ahcfams_mobile/model/confirmation-notice.model.dart'; import'packag$
$flutter/material.dart'; import'package : flutter_signature_pad/flutter_signature_pad.dart';$

import 'package:image/image.dart' as Im;

import 'dart:ui' as ui;

class ConfirmationNoticePage extends StatefulWidget final ConfirmationNoticeList notice; Confirmation NoticePage(this.notice);

@override $_C onfirmationNoticePageStatecreateState() =>_C onfirmationNoticePageState();$

class $_C onfirmationNoticePageStateextendsState < ConfirmationNoticePage >$

final $_s ign = GlobalKey < SignatureState > (); String_s ignaturePath = null;$

String $_d ropdownV alue =' Absent';$

String $_r eason = null;$

@override void initState() super.initState();

@override Widget build(BuildContext context) return Scaffold(appBar: $_a ppBar(), body :_s how(),);$

$_a ppBar()returnAppBar(title : Text('ConfirmationNotice'), backgroundColor : Colors.blue[800],);$

$_s how()returnContainer(padding : EdgeInsets.all(20), child : Column(crossAxisAlignment : CrossAxisAlignr$

$_s aveNotice()asyncfinalsign =_s ign.currentState; finalfileName = awaitSharedFunctions().generateFileNa$

Im.Image image1 = Im.decodeImage(new File(path).readAsBytesSync()); new File(path)..writeAsBytesSync(Im.e

setState(() =¿ $_s ignaturePath = file.path);$

print((await File($_s ignaturePath).length()/1000).toString() + "Kb");

if($_d ropdownV alue ==' Absent')ConfirmationNoticenotice = newConfirmationNotice(confirmationNoticeI$

await DatabaseHelper.db.updateNotice(notice); else ConfirmationNotice notice = new Confirmation Notice(confirmationNoticeId: widget.notice.confirmationNoticeId, confirmationNoticeDate: widget.notice.confirmationNoticeDate, reason: $_r eason, electronicSignature : fileName, remarks :$ "", $confirmed :$ "1", $noticeSynchronized :'$ 0',);

print(await DatabaseHelper.db.updateNotice(notice)); print(await DatabaseHelper.db.confirmPresent(widget.not widget.notice.confirmationNoticeId)); sign.clear(); Navigator.of(context).pop();

$_b ottomButton()returnExpanded(child : Align(alignment : Alignment.bottomCenter, child : Row(mainAxisAli$

$_s ignaturePad()returnContainer(height : 200, color : Colors.grey[300], padding : EdgeInsets.all(8), child : Sign$ points in the signature'); ,),);

$_d ropdown()returnContainer(child : DropdownButton(value :_d ropdownV alue, style : TextStyle(color : Colors.$

$_{r}easonTextField()returnExpanded(child : TextField(keyboardType : TextInputType.multiline, maxLines : 4,$

A.1.6 dashboard.page.dart

import 'package:ahcfams$_{m}obile/bloc/theme-bloc.bloc.dart'; import'package : ahcfams_{m}obile/common/shared-functions.common.dart'; import'package : ahcfams_{m}obile/db/database.helper.dart'; import'package : ahcfams_{m}obile/service/push.service.dart'; import'package : ahcfams_{m}obile/ui/building-list.page.dart'; import'p ahcfams_{m}obile/ui/faculty-list.page.dart'; import'package : flutter/cupertino.dart'; import'package : flutter/material.dart'; import'package : provider/provider.dart';

class DashboardPage extends StatefulWidget DashboardPage(Key key) : super(key: key);

@override $_{D}ashboardPageStatecreateState() =>_{D}$ ashboardPageState();

class $_{D}ashboardPageStateextendsState <$ DashboardPage >

final GlobalKey¡ScaffoldState¿ $_{s}caffoldKey = GlobalKey <$ ScaffoldState > ();

@override Widget build(BuildContext context) return Scaffold(key: $_{s}caffoldKey, appBar :_{a}$ ppBar(), body :$_{d}$ ashBoard(),);

$_{a}ppBar()returnAppBar(title : Text('Dashboard'), elevation : 0.0, actions :< Widget > [IconButton(icon : Icon$

$_{d}ashBoard()returnContainer(child : GridView.count(padding : constEdgeInsets.all(10), crossAxisSpacing : 5$

$_{b}uildings()returnContainer(child : Card(child : InkWell(child : Column(children :< Widget > [Icon(Icons.li$

A.1.7 faculty-details.page.dart

import 'package:ahcfams$_{m}obile/db/database.helper.dart'; import'package : ahcfams_{m}obile/model/confirmation-notice-list.model.dart'; import'package : ahcfams_{m}obile/model/faculty.model.dart'; import'package : ahcfams_{m}obile/ui/confirmation-notice.page.dart'; import'package : ahcfams_{m}obile/widget/simple-card.widget.dart'; import'package : flutter/material.dart';

class FacultyDetails extends StatefulWidget final String facultyId; FacultyDetails(this.facultyId);

@override $_{F}acultyDetailsStatecreateState() =>_{F}$ acultyDetailsState();

class $_{F}acultyDetailsStateextendsState < FacultyDetails > Facultyfaculty; List < ConfirmationNoticeList >$

@override void initState() super.initState(); $_{i}nit()$;

$_{i}nit()asyncfaculty = awaitDatabaseHelper.db.getSingleFaculty(widget.facultyId); confirmationNoticeList =$

@override Widget build(BuildContext context) return Scaffold(appBar: $_{a}ppBar(), body :_{l}$ oading?Center(child : CircularProgressIndicator(),) :$_{f}$ acultyDetails(),);

$_{a}ppBar()returnAppBar(title : Text('FacultyProfile'), backgroundColor : Colors.blue$[800],);

$_{f}acultyDetails()returnContainer(child : Column(children :< Widget > [SimpleCard("Name : " + faculty.na$

$_{n}otices()returnContainer(child : noticeLength == 0?Center(child : Text('NoData'),) : Expanded(child : List$

$//\ _{n}otice(ConfirmationNoticeListnotice)//returnInkWell(//child : Container(//padding : EdgeInsets.all(10$

$_{n}otice(ConfirmationNoticeListnotice)returnContainer(child : Card(child : ListTile(title : Text(notice.subje$

A.1.8 faculty-list.page.dart

import 'package:ahcfams$_{m}obile/bloc/theme-bloc.bloc.dart'; import'package : ahcfams_{m}obile/db/database.helper.da$

$ahcfams_{m}obile/model/faculty.model.dart'; import'package : ahcfams_{m}obile/ui/faculty-details.page.dart'; impo$

$ahcfams_{m}obile/widget/faculty-search.widget.dart'; import'package : flutter/cupertino.dart'; import'package :$

$flutter/material.dart'; import'package : provider/provider.dart';$

class FacultyListPage extends StatefulWidget FacultyListPage(Key key) : super(key: key);

@override $_{F}acultyListPageStatecreateState() =>_{F}\ acultyListPageState();$

class $_{F}acultyListPageStateextendsState < FacultyListPage > finalGlobalKey < ScaffoldState >_{s}\ caffoldK$

bool $_{l}oading = true;$

$_{i}nitFacultyList()asyncfaculty = awaitDatabaseHelper.db.getFaculty(); setState(() =>_{l}\ oading = false); prin$

@override void initState() super.initState(); $_{i}nitFacultyList();$

@override Widget build(BuildContext context) return Scaffold(key: $_{s}caffoldKey, appBar :_{a}$

$ppBar(), body :_{l}\ oading?Center(child : CircularProgressIndicator(),) :_{s}\ how(),);$

$_{a}ppBar()returnAppBar(title : Text('Faculty'), actions :< Widget > [IconButton(icon : Icon(Icons.search), on$

$_{s}how()returnfacultyLength == 0?Center(child : Text('NoDataAvailable'),) : ListView.builder(itemCount :$

$_{f}aculty(Faculty faculty)returnContainer(child : Card(elevation : 2, child : ListTile(leading : Icon(Icons.accou$

A.1.9 home.page.dart

import 'package:ahcfams$_{m}obile/bloc/theme-bloc.bloc.dart'; import'package : ahcfams_{m}obile/model/user.model.da$

$ahcfams_{m}obile/ui/dashboard.page.dart'; import'package : ahcfams_{m}obile/ui/pending-task.page.dart'; import'pa$

$ahcfams_{m}obile/ui/settings.page.dart'; import'package : ahcfams_{m}obile/ui/sync.page.dart'; import'package :$

$ahcfams_{m}obile/widget/icon-badge.widget.dart'; import'package : flutter/cupertino.dart'; import'package :$

$flutter/material.dart'; import'package : provider/provider.dart';$

class HomePage extends StatefulWidget final User currentUser; HomePage(this.currentUser);

@override $_{H}omePageStatecreateState() =>_{H}\ omePageState();$

class $_{H}omePageStateextendsState < HomePage > withSingleTickerProviderStateMixinTabController_{c}ontr$

A.1.10 image-view.page.dart

import 'dart:io';

import 'package:ahcfams$_m$obile/common/shared − functions.common.dart'; import'package : flutter/material.dart';

class ImageView extends StatefulWidget final String firstImageFile; final String secondImage-File; ImageView(this.firstImageFile, this.secondImageFile);

@override $_I$mageViewStatecreateState() =>$_I$ mageViewState();

class $_I$mageViewStateextendsState < ImageView > String$_p$ath;

@override void initState() super.initState(); $_i$nit();

$_i$nit()async$_p$ath = awaitSharedFunctions().localPath; setState(() =>$_p$ ath);

@override Widget build(BuildContext context) return Scaffold(appBar: $_a$ppBar(), body :$_b$ ody(),);

$_a$ppBar()returnAppBar(backgroundColor : Colors.blue[800], title : Text('Images'),);

$_b$ody()returnContainer(padding : EdgeInsets.all(10), child : ListView(children :< Widget > [$_s$howImage(),],)

$_s$howImage()returnColumn(children :< Widget > [Text('FirstImage', style : TextStyle(fontSize : 20),), wid

A.1.11 login.page.dart

import 'package:ahcfams$_m$obile/bloc/theme−bloc.bloc.dart'; import'package : ahcfams$_m$obile/model/user.model.da ahcfams$_m$obile/service/auth.service.dart'; import'package : flutter/cupertino.dart'; import'package : flutter/material.dart'; import'package : permission$_h$andler/permission$_h$andler.dart'; import'package : provider/provider.dart'; import'package : shared$_p$references/shared$_p$references.dart';

class LoginPage extends StatefulWidget final SharedPreferences prefs; LoginPage(this.prefs);

@override $_L$oginPageStatecreateState() =>$_L$ oginPageState();

class $_L$oginPageStateextendsState < LoginPage > String$_u$ser; bool$_u$serError = false; String$_p$assword; bool$_p$a

@override void initState() super.initState(); $_r$equestPermission();

@override Widget build(BuildContext context) final theme = Provider.of¡ThemeBloc¿(context); return MaterialApp(theme: theme.getTheme(), debugShowCheckedModeBanner: false, home: $_l$oginPage(),);

$_l$oginPage()returnScaffold(key :$_s$ caffoldKey, body :$_l$ oading?Center(child : CircularProgressIndicator(),) :

$_l$ogo()returnImage.asset('assets/adnuseal.png');

$_u$serNameInput()returnTextField(keyboardType : TextInputType.text, autofocus : false, decoration : InputI

$_{p}asswordInput()returnTextField(autofocus : false, obscureText : true, decoration : InputDecoration(hintTex$

$_{l}oginButton()returnRaisedButton(padding : EdgeInsets.all(12), color : Colors.blue[900], shape : RoundedRect$

$_{l}ogin(Stringuser, Stringpassword)asyncsetState(() =>_{l} oading = true); tryUsercurrentUser = awaitAuthSer$

$_{r}equestPermission()asyncMap < PermissionGroup, PermissionStatus > statuses = awaitPermissionHandl$

PermissionStatus status = statuses[PermissionGroup.camera];

switch (status) case PermissionStatus.denied: await PermissionHandler().openAppSettings();

break;

A.1.12 setting.page.dart

import 'package:ahcfams$_{m}$obile/model/user.model.dart'; import'package : ahcfams$_{m}$obile/service/auth.service.da$f lutter/cupertino.dart'; import'package : flutter/material.dart';$

class SettingsPage extends StatefulWidget final User currentUser; SettingsPage(this.currentUser);

@override $_{S}ettingsPageStatecreateState() =>_{S}$ ettingsPageState();

class $_{S}ettingsPageStateextendsState < SettingsPage > @overrideWidgetbuild(BuildContextcontext)returnS$

$_{a}ppBar()returnAppBar(title : Text('Settings'), elevation : 0.0, backgroundColor : Colors.blue[800], actions :<$

$_{c}ontent()returnContainer(child : ListView(children :< Widget > [Card(elevation : 1, child : ListTile(leading$

A.1.13 sync.page.dart

import 'package:ahcfams$_{m}$obile/bloc/theme−bloc.bloc.dart'; import'package : ahcfams$_{m}$obile/service/fetch.service$ahcfams_{m}obile/service/push.service.dart'; import'package : flutter/cupertino.dart'; import'package :$
$flutter/material.dart'; import'package : provider/provider.dart';$

class SyncPage extends StatefulWidget SyncPage(Key key) : super(key: key);

@override $_{S}yncPageStatecreateState() =>_{S}$ yncPageState();

class $_{S}yncPageStateextendsState < SyncPage >$

bool $_{l}oading = false; @overrideWidgetbuild(BuildContextcontext)returnScaffold(appBar :_{a}$ ppBar(), body :$_{l}$

$_{a}ppBar()returnAppBar(title : Text('Sync'), elevation : 0.0, actions :< Widget > [IconButton(icon : Icon(Icon.$

$_{s}ync()returnContainer(child : GridView.count(padding : constEdgeInsets.all(10), crossAxisSpacing : 5, main$

$_{u}pdateData()returnContainer(child : Card(child : InkWell(child : Column(children :< Widget > [Icon(Icons$

$_{s}endRecords()returnContainer(child : Card(child : InkWell(child : Column(children :< Widget > [Icon(Icon$

$_{u}pdateDailyData()returnContainer(child : Card(child : InkWell(child : Column(children :< Widget > [Icon($

$_{u}ploadData()asyncsetState(() =>_{l} oading = true); awaitPush().pushFacultyAttendance(); awaitPush().pushC$

$_fetchData()asyncsetState(() =>_l\ oading = true); awaitFetch().fetchClassSchedule(); awaitFetch().fetchFac$

$_fetchDailyData()asyncsetState(() =>_l\ oading = true); awaitFetch().fetchFacultyAttendance(); awaitFetch()$

A.1.14 auth.service.dart

import 'dart:convert';

import 'package:ahcfams$_m$*obile/model/user.model.dart*'; *import'package* : *flutter/material.dart*'; *import'packa*

http/http.dart'ashttp; *import'package* : *ahcfams*$_m$*obile/common/constant.dart'asconstant*; *import'package* :

shared$_p$*references/shared*$_p$*references.dart*';

import '../ui/home.page.dart'; import '../ui/login.page.dart';

class AuthService

signIn(String userId, String password, BuildContext context) async String loginURL = 'http://'

+ constant.domain + '/ADNU$_H$*RMO*−*College*−*Faculty*−*Attendance*−*Monitoring*−*System/index.php/api/login*

'username' : *userId*,*' password'* : *password*,; *varresponse* = *awaithttp.post*(*loginURL*, *body* : *data*); *if*(*response.sta*

200)*UsercurrentUser* = *User.fromJson*(*json.decode*(*response.body*)); *if*(*currentUser.status* == 200)*Navigator.p*

init(User user) async SharedPreferences prefs = await SharedPreferences.getInstance(); prefs.clear();

if(user != null) var userId = user.userId; var token = user.token; var routeId = user.routeId;

if(token.isNotEmpty user.status == 200) prefs.setString('userId', userId); prefs.setString('token',

token); prefs.setString('routeId', routeId);

signOut(BuildContext context) async SharedPreferences prefs = await SharedPreferences.getInstance();

prefs.clear(); Navigator.pushReplacement(context, MaterialPageRoute(builder: $(_)$ => *LoginPage*(*prefs*)));

A.1.15 fetch.service.dart

import 'dart:convert';

import 'package:ahcfams$_m$*obile/common/constant.dart'asconstant*; *import'package* : *ahcfams*$_m$*obile/db/databa*

ahcfams$_m$*obile/model/class*−*schedule.model.dart*'; *import'package* : *ahcfams*$_m$*obile/model/confirmation*−

notice.model.dart'; *import'package* : *ahcfams*$_m$*obile/model/faculty*−*attendance.model.dart*'; *import'package* :

ahcfams$_m$*obile/model/faculty.model.dart*'; *import'package* : *ahcfams*$_m$*obile/model/note.model.dart*'; *import'pack*

ahcfams$_m$*obile/model/room.model.dart*'; *import'package* : *shared*$_p$*references/shared*$_p$*references.dart*'; *import'p*

http/http.dart'ashttp;

class Fetch

fetchFaculty() async SharedPreferences prefs = await SharedPreferences.getInstance(); String url = "http://" + constant.domain + "/ADNU$_H RMO-College-Faculty-Attendance-Monitoring-System/index.php/api/faculty"; Map data = 'id' : prefs.getString('userId'), 'token' : prefs.getString('token'), ; var response = await http.post(url, body: data); if(response.statusCode == 200) var jsonData = await json.decode(response.body); print('Faculty data has been fetched'); for (var f in jsonData) Faculty currFaculty = Faculty.fromJson(f); await DatabaseHelper.db.insertFaculty(currFaculty); print('Faculty data has been inserted');

fetchClassSchedule() async SharedPreferences prefs = await SharedPreferences.getInstance(); String url = "http://" + constant.domain + "/ADNU$_H RMO - College - Faculty - Attendance - Monitoring-System/index.php/api/class_schedule"; $Mapdata = 'id' : prefs.getString('userId'), 'token' : prefs.ge$ var response = await http.post(url, body: data); if(response.statusCode == 200) var jsonData = await json.decode(response.body); print('Class Schedule Data has been fetched'); for (var f in jsonData) ClassSchedule classSchedule = ClassSchedule.fromJson(f); await DatabaseHelper.db.insertClassSchedule(classSchedule); print('Class Schedule data has been inserted');

fetchRoom() async SharedPreferences prefs = await SharedPreferences.getInstance(); String url = "http://" + constant.domain + "/ADNU$_H RMO-College-Faculty-Attendance-Monitoring-System/index.php/api/room"; $Mapdata = 'id' : prefs.getString('userId'), 'token' : prefs.getString('token'),;$ var response = await http.post(url, body: data); if(response.statusCode == 200) var jsonData = await json.decode(response.body); print('Room Data has been fetched'); for (var r in jsonData) Room room = Room.fromJson(r); await DatabaseHelper.db.insertRoom(room); print('Room data has been inserted');

fetchFacultyAttendance() async SharedPreferences prefs = await SharedPreferences.getInstance(); String url = "http://" + constant.domain + "/ADNU$_H RMO - College - Faculty - Attendance - Monitoring-System/index.php/api/get_faculty_attendance"; $Mapdata = 'id' : prefs.getString('userId'), 'token' : p$ var response = await http.post(url, body: data); if(response.statusCode == 200) var jsonData = await json.decode(response.body); print("Faculty Attendance has been fetched"); for (var f in jsonData) FacultyAttendance facultyAttendance = FacultyAttendance.fromJson(f); await DatabaseHelper.db.insertFacultyAttendance(facultyAttendance); print("Faculty Attendance has been inserted"); print("Faculty Attendance Length: " + jsonData.length.toString());

fetchConfirmationNotice() async SharedPreferences prefs = await SharedPreferences.getInstance(); String url = "http://" + constant.domain + "/ADNU$_H RMO - College - Faculty - Attendance - Monitoring - System/index.php/api/get_confirmation_notice"; $Mapdata = 'id' : prefs.getString('userId'), 'token'$

var response = await http.post(url, body: data);

if(response.statusCode == 200) var jsonData = await json.decode(response.body); print("Confirmation Notice has been fetched"); for (var c in jsonData) ConfirmationNotice confirmation = Confirmation-Notice.fromJson(c); await DatabaseHelper.db.insertConfirmationNotice(confirmation); print("Confirmation Notices has been inserted"); print("Confirmation Notice Length: " + jsonData.length.toString());

fetchNote() async String url = "http://" + constant.domain + "/ADNU$_H RMO - College - Faculty - Attendance - Monitoring - System/index.php/api/note";

var response = await http.get(url);

var jsonData = await json.decode(response.body); for (var n in jsonData) Note note = Note.fromJson(n); await DatabaseHelper.db.insertNote(note); print("Notes has been inserted"); print("Notes has Length: " + jsonData.length.toString());

A.1.16 push.service.dart

import 'dart:convert';

import 'package:ahcfams$_m$obile/common/shared - $f$unctions.common.dart'; $import 'package$: $ahcfams_m obile/db/database.helper.dart'; import 'package : dio/dio.dart'; import 'package : shared_preferences/sha$ $ahcfams_m obile/common/constant.dart' as constant;

class Push pushFacultyAttendance() async SharedPreferences prefs = await SharedPreferences.getInstance(); final String path = await SharedFunctions().localPath;

Dio dio = new Dio(); print(path);

List¡Map¡String, dynamic¿¿ $_f acultyAttendance = await DatabaseHelper.db.getCompletedFacultyAttendance()$;

var url = "http://" + constant.domain + "/ADNU$_H RMO - College - Faculty - Attendance - Monitoring - System/api/post_faculty_attendance";

print($_f acultyAttendance.length$);

for (var f in $_f acultyAttendance$)

FormData data = new FormData.fromMap('id' : prefs.getString('userId'), 'token' : prefs.getString('token'), 'faid' : f['facultyAttendanceId'], 'sid' : f['staffId'], 'csid' : f['classScheduleId'], 'adate' : f['attendanceDate'], 'fcheck' : f['firstCheck'], 'scheck' : f['secondCheck'], 'fipath' : f['firstImageFile'] != null ? await

MultipartFile.fromFile(path + '/' + f['firstImageFile'], filename: f['firstImageFile']) : '', 'sipath' : f['secondImageFile'] != null ? await MultipartFile.fromFile(path + '/' + f['secondImageFile'], filename: f['secondImageFile']) : '', 'status' : f['status'], 'validated' : f['validated'], 'cnid' : f['confirmationNoticeId'], 'nid' : '');

// print(await io.File(path + '/' + f['firstImageFile']).exists()); // print(path + '/' + f['firstImageFile']); // print(data.fields);

var response = await dio.post(url, data: data); var jsonResponse = json.decode(response.data);

print("Faculty Attendance: " + jsonResponse.toString()); if(jsonResponse['status'] == 201) await DatabaseHelper.db.synchronize(f['facultyAttendanceId'], 'faculty$_{a}$ttendance');

pushConfirmationNotice() async SharedPreferences prefs = await SharedPreferences.getInstance();

final String path = await SharedFunctions().localPath;

Dio dio = new Dio();

List¡Map¡String, dynamic¿¿ $_{c}onfirmationNotice = awaitDatabaseHelper.db.getConfirmationNotice(); varurl$ "$http://"+constant.domain+"/ADNU_HRMO-College-Faculty-Attendance-Monitoring-System/index.php/api/post_confirmation_notice$";

print($_{c}onfirmationNotice.length$);

for (var c in $_{c}onfirmationNotice)FormDatadata = newFormData.fromMap('id' : prefs.getString('userId'),'$

var response = await dio.post(url, data: data); var jsonResponse = json.decode(response.data);

print("Confirmation Notice: " + jsonResponse.toString());

if(jsonResponse['status'] == 201) await DatabaseHelper.db.synchronize(c['confirmationNoticeId'], "confirmation$_{n}otice$");

A.1.17 class-schedule.model.dart

import 'package:ahcfams$_{m}obile/common/shared - functions.common.dart'$;

class ClassSchedule final String classScheduleId; final String facultyId; final String semester; final String schoolYear; final String startTime; final String endTime; final String classSection; final String classDay; final String subjectCode; final String hours;

ClassSchedule(this.classScheduleId, this.facultyId, this.semester, this.schoolYear, this.startTime, this.endTime, this.classSection, this.classDay, this.subjectCode, this.hours);

factory ClassSchedule.fromJson(Map¡String, dynamic¿ json) return ClassSchedule(classScheduleId: json['CLASS$_{S}CHEDULE_{I}D'], facultyId : json['FACULTY_{I}D'], semester : json['SEMESTER'], schoolY$_{e}$

$json['SCHOOL_YEAR'], startTime : SharedFunctions().time12HoursTo24Hours(json['START_TIME'].toString$
$SharedFunctions().time12HoursTo24Hours(json['END_TIME'].toString()), classSection : json['CLASS_SECTI$
$json['CLASS_DAY'], subjectCode : json['SUBJECT_CODE'], hours : json['HOURS']);$

Map¡String, dynamic¿ toMap() =¿ "classScheduleId" : classScheduleId, "facultyId" : facultyId, "semester" : semester, "schoolYear" : schoolYear, "startTime" : startTime, "endTime" : endTime, "classSection" : classSection, "classDay" : classDay, "subjectCode" : subjectCode, "hours" : hours ;

A.1.18 confirmation-notice-list.model.dart

class ConfirmationNoticeList final String name; final String confirmationNoticeId; final String facultyAttendanceId; final String attendanceDate; final String classTime; final String subjectCode; final String classSection; final String room; final String confirmationNoticeDate;

ConfirmationNoticeList(this.name, this.confirmationNoticeId, this.facultyAttendanceId, this.attendanceDate,this. this.subjectCode, this.classSection, this.room, this.confirmationNoticeDate);

A.1.19 confirmation-notice.model.dart

class ConfirmationNotice final String confirmationNoticeId; final String confirmationNoticeDate; final String reason; final String electronicSignature; final String remarks; final String confirmed; final String noticeSynchronized;

ConfirmationNotice(this.confirmationNoticeId, this.confirmationNoticeDate, this.reason, this.electronicSignature, this.remarks, this.confirmed, this.noticeSynchronized);

factory ConfirmationNotice.fromJson(Map¡String, dynamic¿ json) return ConfirmationNotice(confirmationNoticeId: json['CONFIRMATION_NOTICE_ID'], $confirmationNoticeDate : json['CONFIRMATIO$
$json['REASON'], electronicSignature : json['ELECTRONIC_SIGNATURE'], remarks : json['REMARKS'], co$
$json['CONFIRMED'],);$

Map¡String, dynamic¿ toMap() =¿ "confirmationNoticeId" : confirmationNoticeId, "confirmationNoticeDate" : confirmationNoticeDate, "reason" : reason, "electronicSignature" : electronicSignature, "remarks" : remarks, "confirmed" : confirmed, "noticeSynchronized" : 0, ;

A.1.20 faculty-attendance-list.model.dart

class FacultyAttendanceList final String facultyAttendanceId; final String facultyId; final String name; final String room; final String subjectCode; final String startTime; final String endTime; final String firstCheck; final String secondCheck; final String classDay; final String attendanceDate; final String firstImageFile; final String secondImageFile; final String noteId;

FacultyAttendanceList(this.facultyAttendanceId, this.facultyId, this.name, this.room, this.subjectCode, this.startTime, this.endTime, this.firstCheck, this.secondCheck, this.classDay, this.attendanceDate, this.firstImageFile, this.secondImageFile, this.noteId);

A.1.21 faculty-attendance.model.dart

class FacultyAttendance final String facultyAttendanceId; final String classScheduleId; final String confirmationNoticeId; final String roomId; final String attendanceDate; final String firstCheck; final String secondCheck; final String firstImageFile; final String secondImageFile; final String status; final String validated; final String noteId;

FacultyAttendance(this.facultyAttendanceId, this.classScheduleId, this.confirmationNoticeId, this.roomId, this.attendanceDate, this.firstCheck, this.secondCheck, this.firstImageFile, this.secondImageFile, this.status, this.validated, this.noteId,);

factory FacultyAttendance.fromJson(Map¡String, dynamic¿ json) return FacultyAttendance(facultyAttendanceId: json['FACULTY$_{A}TTENDANCE_{I}D'], $classScheduleId : json['CLASS_{S}CHEDULE_{I}D'], confi$ $json['CONFIRMATION_{N}OTICE_{I}D'], roomId : json['ROOM_{I}D'], attendanceDate : json['ATTENDANCE_{D}A$ $json['FIRST_{C}HECK'], secondCheck : json['SECOND_{C}HECK'], firstImageFile : json['FIRST_{I}MAGE_{F}ILE'$ $json['SECOND_{I}MAGE_{F}ILE'], status : json['STATUS'], validated : json['VALIDATED'], noteId :$ $json['NOTE_{I}D'],);$

Map¡String, dynamic¿ toMap() =¿ // this is used to insert data to db "facultyAttendanceId" : facultyAttendanceId, "classScheduleId" : classScheduleId, "confirmationNoticeId" : confirmationNoticeId, "roomId" : roomId, "attendanceDate" : attendanceDate, "firstCheck" : firstCheck, "secondCheck" : secondCheck, "firstImageFile" : firstImageFile, "secondImageFile" : secondImageFile, "status" : status, "validated" : 0, "attendanceSynchronized" : 1, "noteId" : noteId, ;

A.1.22 faculty.model.dart

class Faculty final String facultyId; final String name; final String department; final String college;

Faculty(this.facultyId, this.name, this.department, this.college);

factory Faculty.fromJson(Map¡String, dynamic¿ json) return Faculty(facultyId: json['FACULTY$_ID'], $name$: $json['NAME']$, $department$: $json['DEPARTMENT']$, $college$: $json['COLLEGE']$,);

Map¡String, dynamic¿ toMap() =¿ "facultyId" : facultyId, "name" : name, "department" : department, "college" : college, ;

A.1.23 note.model.dart

class Note final String noteId; final String description;

Note(this.noteId, this.description);

factory Note.fromJson(Map¡String, dynamic¿ json) return Note(noteId: json['NOTE$_ID'], $description$: $json['DESCRIPTION']$,);

Map¡String, dynamic¿ toMap()=¿ "noteId" : noteId, "description" : description, ;

A.1.24 room.model.dart

class Room final String roomId; final String roomName; final String buildingName; final String route;

Room(this.roomId, this.roomName, this.buildingName, this.route);

factory Room.fromJson(Map¡String, dynamic¿ json) return Room(roomId: json['ROOM$_ID'], $roomName$: $json['ROOM_NAME']$, $buildingName$: $json['BUILDING_NAME']$, $route$: $json['ROUTE']$,);

Map¡String, dynamic¿ toMap() =¿ "roomId" : roomId, "roomName" : roomName, "buildingName" : buildingName, "route" : route, ;

A.1.25 user.model.dart

class Room final String roomId; final String roomName; final String buildingName; final String route;

Room(this.roomId, this.roomName, this.buildingName, this.route);

factory Room.fromJson(Map¡String, dynamic¿ json) return Room(roomId: json['ROOM$_ID'], $roomName$: $json['ROOM_NAME']$, $buildingName$: $json['BUILDING_NAME']$, $route$: $json['ROUTE']$,);

Map¡String, dynamic¿ toMap() =¿ "roomId" : roomId, "roomName" : roomName, "building-Name" : buildingName, "route" : route, ;

A.1.26 database.helper.dart

import 'dart:io'; import 'package:ahcfams$_{m}obile/common/shared-functions.common.dart'; import'package : ahcfams$_{m}obile/model/class-schedule.model.dart'; import'package : ahcfams$_{m}obile/model/confirmation-notice-list.model.dart'; import'package : ahcfams$_{m}obile/model/confirmation-notice.model.dart'; import'packag ahcfams$_{m}obile/model/faculty-attendance-list.model.dart'; import'package : ahcfams$_{m}obile/model/faculty-attendance.model.dart'; import'package : ahcfams$_{m}obile/model/faculty.model.dart'; import'package : ahcfams$_{m}obile/model/note.model.dart'; import'package : ahcfams$_{m}obile/model/room.model.dart'; import'packa path/path.dart'; import'package : path$_{p}rovider/path$_{p}rovider.dart'; import'package : shared$_{p}references/shared$_{p}r sqflite/sqflite.dart';

class DatabaseHelper DatabaseHelper.$_{(}$();

static final DatabaseHelper db = DatabaseHelper.$_{(}$(); $staticDatabase_{d}atabase$;

Future¡Database¿ get database async if($_{d}atabase != null)return_{d}atabase;$_{d}atabase = awaitinitDB()$;

return $_{d}atabase$;

initDB() async Directory documentsDirectory = await getApplicationDocumentsDirectory(); String path = join(documentsDirectory.path, "ahcfams.db"); return await openDatabase(path, version: 1, onOpen: (db) , onCreate: (Database db, int version) async await db.execute("CREATE TABLE FACULTY(" "facultyId VARCHAR PRIMARY KEY NOT NULL," "name VARCHAR NOT NULL," "department VARCHAR NOT NULL," "college VARCHAR)");

await db.execute("CREATE TABLE CLASS$_{S}CHEDULE$(""$classScheduleIdVARCHARPRIMARYKEYNO$

await db.execute("CREATE TABLE ROOM(" "roomId VARCHAR PRIMARY KEY NOT NULL," "roomName VARCHAR NOT NULL," "buildingName VARCHAR NOT NULL," "route INTEGER NOT NULL)");

await db.execute("CREATE TABLE FACULTY$_{A}TTENDANCE$(""$facultyAttendanceIdVARCHARPRIMA$

await db.execute("CREATE TABLE CONFIRMATION$_{N}OTICE$(""$confirmationNoticeIdVARCHARPRIM$

await db.execute("CREATE TABLE NOTE(" "noteId VARCHAR PRIMARY KEY NOT NULL," "description VARCHAR NOT NULL)"););

insertFaculty(Faculty faculty)async final db = await database; Batch batch = db.batch();

if (! await isRecorded(faculty.facultyId, 'facultyId', 'FACULTY')) batch.insert('FACULTY', fac-
ulty.toMap(), conflictAlgorithm: ConflictAlgorithm.replace); else var $_facultyId = faculty.facultyId; batch.update('1$
"$facultyId =' _facultyId'"$, $conflictAlgorithm : ConflictAlgorithm.replace$); $varres = awaitbatch.commit(noResul$
$true$);

return res;

insertClassSchedule(ClassSchedule classSchedule) async final db = await database; Batch batch
= db.batch();

if(! await isRecorded(classSchedule.classScheduleId, 'classScheduleId', 'class$_schedule'$))$batch.insert('CLASS_SCH$
$awaitbatch.commit(noResult : true)$;

return res;

insertRoom(Room room) async final db = await database; Batch batch = db.batch();

if(! await isRecorded(room.roomId, 'roomId', 'ROOM')) batch.insert("ROOM", room.toMap(),
conflictAlgorithm: ConflictAlgorithm.replace); else var $_roomId = room.roomId; batch.update('ROOM', room.toMap$
"$roomId =' _roomId'"$, $conflictAlgorithm : ConflictAlgorithm.replace$); $varres = awaitbatch.commit(noResult :$
$true$);

return res;

insertFacultyAttendance(FacultyAttendance facultyAttendance) async final db = await database;
Batch batch = db.batch();

if(!await isRecorded(facultyAttendance.facultyAttendanceId, 'facultyAttendanceId', 'faculty$_attendance'$))$batch.in$
$awaitbatch.commit(noResult : true)$;

return res;

insertConfirmationNotice(ConfirmationNotice confirmationNotice) async final db = await database;
Batch batch = db.batch();

if(! await isRecorded(confirmationNotice.confirmationNoticeId, 'confirmationNoticeId', 'confirmation$_notice'$))$batc$
$awaitbatch.commit(noResult : true); returnres$;

insertNote(Note note) async final db = await database; Batch batch = db.batch();

if(! await isRecorded(note.noteId, "noteId", "note")) batch.insert('NOTE', note.toMap(), con-
flictAlgorithm: ConflictAlgorithm.replace); else var $_noteId = note.noteId; batch.update("NOTE", note.toMap(), wh$
"$noteId =' _noteId'"$); $varres = awaitbatch.commit(noResult : true)$;

return res;

Future¡List¡Faculty¿¿ getFaculty()async final db = await database; final List¡Map¡String, dy-

namic¿¿ maps = await db.query('FACULTY',orderBy: 'name'); return List.generate(maps.length, (i) return Faculty(facultyId: maps[i]['facultyId'], name: maps[i]['name'], department: maps[i]['department'], college: maps[i]['college']););

Future¡Faculty¿ getSingleFaculty(String facultyId) async

final db = await database; final List¡Map¡String, dynamic¿¿ res = await db.rawQuery("select * from FACULTY where facultyId = '$facultyId'");

return Faculty(facultyId: res.elementAt(0)['facultyId'], name: res.elementAt(0)['name'], department: res.elementAt(0)['department'], college: res.elementAt(0)['college'],);

getBuildings() async final db = await database; SharedPreferences prefs = await SharedPreferences.getInstance();

final List¡Map¡String, dynamic¿¿ maps = await db.rawQuery("SELECT distinct room.buildingName FROM faculty$_a$ttendanceINNERJOINclass$_s$cheduleONfaculty$_a$ttendance.classScheduleId = class$_s$chedule.clas faculty.facultyIdINNERJOINroomonroom.roomId = faculty$_a$ttendance.roomIdwhereclass$_s$chedule.startTime "+SharedFunctions().timeNow24Hours()+"'andclass$_s$chedule.endTime >=' "+SharedFunctions().timeNow24 "'andfaculty$_a$ttendance.attendanceDate =' "+SharedFunctions().dateNow()+"'androom.route =' " + prefs.getString("routeId") + "'orderbyfaculty$_a$ttendance.roomIdasc");

return maps;

getAttendanceList(String $_b$uildingName)asyncfinaldb = awaitdatabase;

final List¡Map¡String, dynamic¿¿ maps = await db.rawQuery("SELECT faculty$_a$ttendance.facultyAttendanceId, class$_s$chedule.classScheduleIdINNERJOINfacultyONclass$_s$chedule.facultyId = faculty.facultyIdINNERJO faculty$_a$ttendance.roomIdwhereroom.buildingName =' "+$_b$uildingName+"'andclass$_s$chedule.startTime <=' "+SharedFunctions().timeNow24Hours()+"'andclass$_s$chedule.endTime >=' "+SharedFunctions().timeNow24 "'andfaculty$_a$ttendance.attendanceDate =' "+SharedFunctions().dateNow()+"'orderbyfaculty$_a$ttendance.roomI

return List.generate(maps.length, (i) return FacultyAttendanceList(facultyAttendanceId: maps[i]['facultyAttenda facultyId: maps[i]['facultyId'], name: maps[i]['name'], room: maps[i]['roomId'], subjectCode: maps[i]['subjectCode'], startTime: SharedFunctions().time24HoursTo12Hours(maps[i]['startTime'].toString()) , endTime: SharedFunctions().time24HoursTo12Hours(maps[i]['endTime'].toString()) , firstCheck: maps[i]['firstCheck'], secondCheck: maps[i]['secondCheck'], classDay: maps[i]['classDay'], attendanceDate: maps[i]['attendanceDate'], firstImageFile: maps[i]['firstImageFile'], secondImageFile: maps[i]['secondImageFile'], noteId: maps[i]['noteId'],););

getNotes() async final db = await database; final List¡Map¡String, dynamic¿¿ res = await

db.query('note');

return List.generate(res.length, (i) return Note(noteId: res[i]['noteId'], description: res[i]['description'],
););

getNote(String noteId) async final db = await database; final List¡Map¡String, dynamic¿¿ res =
await db.rawQuery("select note.noteId, note.description from note where noteId = '$noteId'"); $returnNote(noteId :$
$res.elementAt(0)['noteId'], description : res.elementAt(0)['description'],);$

changeNote(String facultyAttendanceId, String noteId) async final db = await database; Map¡String,
dynamic¿ value = "noteId" : noteId, ;

var res = db.update("faculty$_a$$ttendance", $value, where :$ "$facultyAttendanceId =$'facultyAttendanceId'",
);

return res;

markPresence(String $_f$$acultyAttendanceId, String_t$$ime)asyncfinaldb = awaitdatabase; Map < String, dynamic$

if(query.isNotEmpty) var res = db.update("FACULTY$_A$$TTENDANCE", $firstCheck, where :$
"$facultyAttendanceId =_f$$acultyAttendanceId'"); $returnres; elsevarres = db.update("FACULTY_ATTENDANC$

Future¡List¡ConfirmationNoticeList¿¿ getConfirmationNoticeListOfAFaculty(String $_f$$acultyId)asyncfinaldb = a$

List¡Map¡String, dynamic¿¿ maps = await db.rawQuery("select faculty$_a$$ttendance.facultyAttendanceId, class_sch$
$||class_schedule.endTimeASclassTime, class_schedule.classSection, faculty_attendance.attendanceDate, confirmati$
$faculty_attendance.confirmationNoticeIdinnerjoinclass_scheduleonfaculty_attendance.classScheduleId =$
$class_schedule.classScheduleIdinnerjoinfacultyonclass_schedule.facultyId = faculty.facultyIdinnerjoinroomonf$
$room.roomIdwhereclass_schedule.facultyId =_f$$acultyId'andconfirmation_notice.confirmed = 0andconfirmation_$
0"); $returnList.generate(maps.length, (i)returnConfirmationNoticeList(name : maps[i]['name'], confirmationN$

Future getConfirmationNotice() async final db = await database; final List¡Map¡String, dynamic¿¿ maps = await db.rawQuery("select confirmation$_n$$otice.confirmationNoticeId, faculty_attendance.facultyA$
$faculty_attendance.confirmationNoticeIdwhereconfirmation_notice.confirmed =$ 1'$andconfirmation_notice.notic$
0");

return maps;

Future getCompletedFacultyAttendance() async final db = await database; List¡Map¡String, dynamic¿¿ maps = await db.rawQuery("select facultyAttendanceId, staffId, classScheduleId, attendanceDate, firstCheck, secondCheck, firstImageFile, secondImageFile, status, validated, confirmationNoticeId from faculty$_a$$ttendancewherefirstCheckISNOTNULLandsecondCheckISNOTNULLandattendance$
0");

return maps;

changeStatus(String facultyAttendanceId) async final db = await database; var res1 = await db.rawQuery("select * from faculty$_a$ttendancewhere$facultyAttendanceId$ ='facultyAttendanceId' and firstImageFile IS NULL and secondImageFile IS NULL");

if(res1.isNotEmpty) Map¡String, dynamic¿ map = 'status' : "Present", ; await db.update('faculty$_a$ttendance', ma "$facultyAttendanceId$ ='facultyAttendanceId'"); else Map¡String, dynamic¿ map1 = 'status' : 'Absent', ; Map¡String, dynamic¿ map2 = 'validated' : 'null', ;

await db.update('faculty$_a$ttendance', $map1, where$: "$facultyAttendanceId$ ='facultyAttendanceId'"); await db.update('faculty$_a$ttendance', $map2, where$: "$facultyAttendanceId$ ='facultyAttendanceId'");

saveImageFileName(String facultyAttendanceId, String fileName) async final db = await database;

var $_res1 = awaitdb.rawQuery("select*fromfaculty_attendancewherefacultyAttendanceId =$'facultyAttendance and firstCheck IS NULL");

if($_res1.isNotEmpty)Map < String, dynamic >$_data = $'firstImageFile' : fileName;

await db.update('faculty$_a$ttendance',$_data, where$: "$facultyAttendanceId$ ='facultyAttendanceId'"); else Map¡String, dynamic¿ $_data = 'secondImageFile' : fileName;

await db.update('faculty$_a$ttendance',$_data, where$: "$facultyAttendanceId$ ='facultyAttendanceId'"); print($_res1);

getFacultyAttendance(String $_facultyAttendanceId)asyncfinaldb = awaitdatabase;

var $_res = awaitdb.rawQuery("select*fromfaculty_attendancewherefacultyAttendanceId =$'_facultyAttendance

print($_res);

synchronize(String id, String table) async final db = await database;

if(table == "faculty$_a$ttendance")$Map < String, dynamic > data = 'attendanceSynchronized' : 1,; db.update('fa$ else if(table == "confirmation$_n$otice")$Map < String, dynamic > data = 'noticeSynchronized' : 1,; db.update('confi$

isRecorded(String value, String colName, String table) async final db = await database;

var res = await db.rawQuery("select $colNamefrom$table where $colName$ ='value'");

if(res.length == 0) return false; else return true;

updateNotice(ConfirmationNotice notice) async final db = await database;

String noticeId = notice.confirmationNoticeId.toString();

var res = db.update('CONFIRMATION$_N OTICE'$, notice.toMap(), where : "$confirmationNoticeId$ ='noticeId"

return res;

emptyTable(String table) async final db = await database;

await db.rawQuery("DELETE from '*table*'");

confirmPresent(String facultyAttendanceId, String confirmationNoticeId) async final db = await database;

Map¡String, dynamic¿ map = "confirmationNoticeId" : confirmationNoticeId, "status" : "Present", "validated" : 0, "attendanceSynchronized" : 0, ;

var res = db.update("FACULTY$_{A}TTENDANCE$", *map*, *where* : "$facultyAttendanceId$ ='facultyAttendanceId return res;

A.1.27 shared-functions.common.dart

import 'package:intl/intl.dart'; import 'package:path$_{p}rovider/path_{p}rovider.dart'$; $import'package$: $shared_{p}references/shared_{p}references.dart'$;

class SharedFunctions

dateNow() var now = new DateTime.now(); return new DateFormat('dd-MMM-yy').format(now.toLocal()).toUpp

timeNow12Hours() // hh:mm:ss a 01:30 PM var now = new DateTime.now(); return new DateFormat('dd-MMM-yyyy h.mm.ss a').format(now.toLocal());

timeNow24Hours() var now = new DateTime.now(); return new DateFormat('HH:mm:ss').format(now.toLocal());

time12HoursTo24Hours(String time) time = time.replaceAll("NN", "PM"); var now = new DateFormat('hh:mma').parse(time);

return DateFormat("HH:mm").format(now);

time24HoursTo12Hours(String time) var now = new DateFormat('HH:mm').parse(time);

return DateFormat("hh:mma").format(now);

time12HoursToDateTime(String time) var date = DateFormat('dd-MMM-yyyy').format(new DateTime.now()); var time1 = DateFormat("hh.mm.ss a").format(new DateFormat('HH:mm:ss a').parse(time));

var dateTime = date + ' ' + time;

return dateTime;

Future¡String¿ get localPath async final directory = await getApplicationDocumentsDirectory();

return directory.path;

Future generateFileName() async SharedPreferences prefs = await SharedPreferences.getInstance();
String fileName = ""; var now = new DateTime.now(); fileName = fileName + now.millisecondsSinceEpoch.toString()
+ prefs.getString('userId') + prefs.getString('routeId');

return fileName;

A.1.28 simple-card.widget.dart

import 'package:flutter/material.dart';

class SimpleCard extends StatelessWidget final $_{t}itle; constSimpleCard(this._{t}itle)$;

@override Widget build(BuildContext context) return Container(child: Card(margin: EdgeIn-
sets.fromLTRB(0, 1, 0, 1), elevation: 1, child: ListTile(title: Text($_{t}itle$),), $shape : RoundedRectangleBorder(border$
$BorderRadius.circular(0)),),);$

A.1.29 faculty-search.widget.dart

import 'package:ahcfams$_{m}obile/model/faculty.model.dart'; import'package : ahcfams_{m}obile/ui/faculty-$
$details.page.dart'; import'package : flutter/material.dart'$;

class DataSearch extends SearchDelegate final List¡Faculty¿ facultyList; List¡Faculty¿ $_{s}uggestionList$;

DataSearch(this.facultyList); @override List¡Widget¿ buildActions(BuildContext context) re-
turn [IconButton(icon: Icon(Icons.clear), onPressed: () query = "";),];

@override Widget buildLeading(BuildContext context) return IconButton(icon: AnimatedIcon(
icon: AnimatedIcons.menu$_{a}rrow, progress : transitionAnimation,), onPressed : ()this.close(context, null););$

@override Widget buildResults(BuildContext context) return ListView.builder(itemCount: $_{s}uggestionList.lengt$
$(context, index)return_{f}aculty(context,_{s} uggestionList[index]););$

@override Widget buildSuggestions(BuildContext context) $_{s}uggestionList = query.isEmpty?facultyList :$
$facultyList.where((q) => q.name.toLowerCase().contains(query)).toList(); returnListView.builder(itemCount :$
$uggestionList.length, itemBuilder : (context, index)return_{f}aculty(context,_{s} uggestionList[index]););$

$_{f}aculty(BuildContextcontext, Facultyfaculty)returnContainer(child : Card(elevation : 2, child : ListTile(isT$

A.1.30 pubspec.yaml

name: ahcfams$_{m}obiledescription : AnewFlutterproject.$

version: 1.0.0+1

environment: sdk: "¿=2.1.0 ¡3.0.0"

dependencies: flutter: sdk: flutter sqflite: path: shared$_preferences$:^{0} .5.6 + 1*http* :^{0} .12.0 + 4$path_provider$:^{1} .6.0*fluttertoast* :^{3} .1.3*camera* : $image_picker$:^{0} .6.3+1*intl* :^{0} .16.1$permission_handler$:$'^4$ .2.0 + $hotfix$.3$'dio$:^{3} .0.8$flutter_signature_pad$:^{2} .0.1*provider* :^{4} .0.4$cupertino_icons$:^{0} .1.2

dev$_dependencies$: $flutter_test$: *sdk* : *flutter*

flutter:

uses-material-design: true

assets: - assets/

fonts: - family: Monstserrat fonts: - asset: fonts/Montserrat-Regular.ttf - asset: fonts/Montserrat-Italic.ttf style: italic

A.2 Web Application - Controllers

A.2.1 AdminController.php

[language=PHP] ¡?php

ini$_set('max_execution_time'$, 0); ini$_set('memory_limit'$,' 2048$M'$);

class AdminController extends CI$_Controllervar$months;

function $_construct()parent::_construct();$this-¿months = array('1' =¿ 'January', '2' =¿ 'February', '3' =¿ 'March', '4' =¿ 'April', '5' =¿ 'May', '6' =¿ 'June', '7' =¿ 'July', '8' =¿ 'August', '9' =¿ 'September', '10' =¿ 'October', '11' =¿ 'November', '12' =¿ 'December');

function facultyAttendance() $staff$ =this-¿session-¿userdata('staff'); if($staff! = NULL$)data = array('title' =¿ 'Faculty Attendance', 'time' =¿ $this- > facultyattendancemodel- > getTime()$,$'classday'$ =>this-¿facultyattendancemodel-¿getClassDay(), 'sched' =¿ $this- > facultyattendancemodel- > getDailySchedule()$,$'sch$ ¿facultyreportmodel-¿getSchoolYear(), 'semesters' =¿ $this- > facultyreportmodel- > getSemester()$,$'staff'$ =>st 'designation' =¿ NULL, 'pendingsCount' =¿ $this- > facultyattendancemodel- > getAllPendingConfirmationCo$ ¿load-¿view('templates/header-main'); $this- > load- > view('templates/header - admin'$,data); $this- > load- > view('pages/hr/faculty/facultyattendance'$,data); $this- > load- > view('templates/footer'$,da else redirect(base$_url('login'$));

function fetchFacultyAttendance() $getData$ =this-¿facultyattendancemodel-¿fetchFacultyAttendanceData(); $output = array($"$data$"$ =>getData-¿result$_array()$//data); echo json$_encode$(output);

function fetchFacultyList() $getData$ =this-¿facultyattendancemodel-¿fetchFacultyListData(); $output$ = $array$("$data$" =>getData-¿result$_a rray$()); $echojson_e ncode$(output);

function fetchClassSchedule() $getData$ =this-¿facultyattendancemodel-¿fetchClassScheduleData(); $output = array$("$data$" =>getData-¿result$_a rray$()); $echojson_e ncode$(output);

function fetchSummaryReport() $getData$ =this-¿facultyattendancemodel-¿fetchSummaryReportData(); $output = array$("$data$" =>getData-¿result$_a rray$()); $echojson_e ncode$(output);

function fetchRegisteredAcc() $getData$ =this-¿facultyattendancemodel-¿fetchRegisteredAccData(); $output = array$("$data$" =>getData-¿result$_a rray$()); $echojson_e ncode$(output);

function fetchNote() $getData$ =this-¿facultyattendancemodel-¿fetchNoteData(); $output = array$("$data$" =>getDa ¿result$_a rray$()); $echojson_e ncode$(output);

function classSchedules() $staff$ =this-¿session-¿userdata('staff'); if($staff! = NULL$)data = array('title' =¿ 'Class Schedule', 'schedule' =¿ $this- > facultyattendancemodel- > getClassSchedule()$,$' staff'$ =>s 'designation' =¿ NULL , 'pendingsCount' =¿ $this- > facultyattendancemodel- > getAllPendingConfirmationCo$ ¿load-¿view('templates/header-main'); $this- > load- > view('templates/header - admin'$,data); $this- > load- > view('pages/hr/general/classschedule'$,data); $this- > load- > view('templates/footer'$,data); else redirect(base$_u rl('login'$));

function sendNotif(id)$IDofFacultyAttendance$config = array('protocol' =¿ 'smtp', 'smtp$_h ost'$ =>$'$ $ssl : //smtp.googlemail.com',' smtp_p ort'$ => 465,$' smtp_u ser'$ =>$'$ @gmail.com', $sender'semail'smtp_p ass'$ =>$'$ $', sender'spassword'mailtype'$ =>$'$ $html'$);this-¿email-¿initialize($config$);this-¿email-¿set$_n ewline$("");data $= this- > facultyattendancemodel- > getEmailDetails$(id); if($data- > num_r ows() > 0)foreach$(data-¿result$_a rray()as$details) $date$ =details['ATTENDANCE$_D ATE'$];this-¿email-¿from('testemailtest1230@gmail.com', 'HRMO'); $this- > email- > to$(emailaddress); $this- > email- > subject('AbsenceNotification'$);this-¿email-¿message("You were recorded as absent on this date $date.confirmyourabsenceusingHRMOWebsiteorchecke$ ¿email-¿send()) echo 'Email sent.'; $this- > facultyattendancemodel- > updateNotif$(id);

function note() $data = array('title'$ =>$'$ $AttendanceNotes',' staff'$ =>staff); $this- > load- > view('templates/header-main'$);this-¿load-¿view('templates/header-admin', $data$);this-¿load-¿view('pages/hr/facul $data$);this-¿load-¿view('templates/footer', $data$);

function classRooms() $staff$ =this-¿session-¿userdata('staff'); $search = strtolower$(this-¿input-¿post('search')); $hidden$ =this-¿input-¿post('hidden'); if($search! = null$)this-¿session-¿set$_u serdata(array('search'$ = $search$ =this-¿session-¿userdata('search'); else if($hidden$)this-¿session-¿unset$_u serdata('search'$);search $= this- > session- > userdata('search'$);config['per$_p age'$] = 5;config['total$_r ows'$] =this-¿facultyattendancemodel-

¿getRoomsCount(*search*);config['base$_u$*rl*'] = *base$_u$rl*('*class*−*rooms*');config['uri$_s$*egment*'] = 2; //config['num$_l$*inks*'] =
floor(config['total$_r$*ows*']/config['per$_p$*age*']);config['num$_l$*inks*'] = 5; // *Pagination* config['full$_t$*ag$_o$pen*'] =′< *ulclass* = "*pagination*" >′;config['full$_t$*ag$_c$lose*'] =′< /*ul* >′;config['attributes'] = ['class' =¿ 'page-link']; *config*[′*first$_l$ink*′] = *false*;config['last$_l$*ink*'] = *false*;config['first$_t$*ag$_o$pen*'] =′< *liclass* = "*page*−*item*" >′;config['first$_t$*ag$_c$lose*'] =′< /*li* >′;config['prev$_l$*ink*'] =′ *laquo*′;config['prev$_t$*ag$_o$pen*'] =′< *liclass* = "*page*−*item*" >′;config['prev$_t$*ag$_c$lose*'] =′< /*li* >′;config['next$_l$*ink*'] =′ *raquo*′;config['next$_t$*ag$_o$pen*'] =′< *liclass* = "*page* − *item*" >′;config['next$_t$*ag$_c$lose*'] =′< /*li* >′;config['last$_t$*ag$_o$pen*'] =′< *liclass* = "*page* − *item*" >′;config['last$_t$*ag$_c$lose*'] =′< /*li* >′;config['cur$_t$*ag$_o$pen*'] =′< *liclass* = "*page* − *itemactive*" >< *ahref* = ""*class* = "*page* − *link*" >′;config['cur$_t$*ag$_c$lose*'] =′< *spanclass* = "*sr* − *only*" > (*current*) < /*span* >< /*a* >< /*li* >′;config['num$_t$*ag$_o$pen*'] =′< *liclass* = "*page* − *item*" >′ ;config['num$_t$*ag$_c$lose*'] =′< /*li* >′; // *Pagination* this-¿pagination-¿initialize(*config*);page = *this*− > *uri*− > *segment*(2); *if*(staff != NULL) *data* = *array*(′*title*′ => "*ClassRooms*",′ *venues*′ =>this-¿facultyattendancemodel-¿getRoomsPagination(*search*,config['per$_p$*age*'],page), 'roomBuildings' =¿ *this*− > *facultyattendancemodel*− > *getBuildings*(),′ *staff*′ =>staff, 'designation' =¿ NULL, 'searched' =¿ *search*,′ *pendingsCount*′ =>this-¿facultyattendancemodel-¿getAllPendingConfirmationCount()); *this*− > *load*− > *view*(′*templates*/*header* − *main*′);this-¿load-¿view('templates/header-admin', *data*);this-¿load-¿view('pages/hr/general/classroom', *data*);this-¿load-¿view('templates/footer', *data*); *elseredirect*(*b*

function buildingList() *staff* =this-¿session-¿userdata('staff'); *search* = *strtolower*(this-¿input-¿post('search')); *hidden* =this-¿input-¿post('hidden'); if(*search*! = *null*)this-¿session-¿set$_u$*serdata*(*array*(′*search*′ = *search* =this-¿session-¿userdata('search'); else if(*hidden*)this-¿session-¿unset$_u$*serdata*(′*search*′);search = *this*− > *session*− > *userdata*(′*search*′);config['per$_p$*age*'] = 5;config['total$_r$*ows*'] =this-¿facultyattendancemodel-¿getBuildingsCount(*search*);config['base$_u$*rl*'] = *base$_u$rl*('*building*−*list*');config['uri$_s$*egment*'] = 2;config['num$_l$*inks*'] = *floor*(config['total$_r$*ows*']/config['per$_p$*age*']); // *Pagination* config['full$_t$*ag$_o$pen*'] =′< *ulclass* = "*pagination*" >′;config['full$_t$*ag$_c$lose*'] =′< /*ul* >′;config['attributes'] = ['class' =¿ 'page-link']; *config*[′*first$_l$ink*′] = *false*;config['last$_l$*ink*'] = *false*;config['first$_t$*ag$_o$pen*'] =′< *liclass* = "*page*−*item*" >′;config['first$_t$*ag$_c$lose*'] =′< /*li* >′;config['prev$_l$*ink*'] =′ *laquo*′;config['prev$_t$*ag$_o$pen*'] =′< *liclass* = "*page*−*item*" >′;config['prev$_t$*ag$_c$lose*'] =′< /*li* >′;config['next$_l$*ink*'] =′ *raquo*′;config['next$_t$*ag$_o$pen*'] =′< *liclass* = "*page* − *item*" >′;config['next$_t$*ag$_c$lose*'] =′< /*li* >′;config['last$_t$*ag$_o$pen*'] =′< *liclass* = "*page* − *item*" >′;config['last$_t$*ag$_c$lose*'] =′< /*li* >′;config['cur$_t$*ag$_o$pen*'] =′< *liclass* = "*page* − *itemactive*" >< *ahref* = ""*class* = "*page* − *link*" >′;config['cur$_t$*ag$_c$lose*'] =′< *spanclass* = "*sr* − *only*" > (*current*) < /*span* >< /*a* >< /*li* >′;config['num$_t$*ag$_o$pen*'] =′< *liclass* = "*page* − *item*" >′

;config['num$_t ag_c lose'] =' < /li >'; // *Pagination* this-¿pagination-¿initialize(*config*);page
$= this- > uri- > segment(2); if$(staff != NULL) $data = array('title' =>$ "*Building*",' *buildings*' =>this-
¿facultyattendancemodel-¿getBuildingsPagination(*search*,config['per$_p age'$],page), 'staff' =¿ *staff*,' *designation*' =>
NULL,' *searched*' =>search, 'pendingsCount' =¿ *this- > facultyattendancemodel- > getAllPendingConfirmati*
¿load-¿view('templates/header-main'); *this- > load- > view('templates/header − admin'*,data);
this- > load- > view('pages/hr/general/building',data); *this- > load- > view('templates/footer'*,data);
else redirect(base$_u rl('login'$));

function generateAttendanceList() *staff* =this-¿session-¿userdata('staff'); *genAttendanceList* =this-
¿api$_m odel- > generate_f aculty_a ttendance(); if$(staff != NULL)$ redirect(base$_u rl('update?message =$
true')); *elseredirect*(*base$_u rl('login'$*));

function syncData() *message* =this-¿input-¿get('message'); *staff* =this-¿session-¿userdata('staff');
if(*staff*! = *NULL*)data = array('title' =¿ 'Sync', 'syncData' =¿ 'Synchronize Data', 'genAttendanceList' =¿ 'Generate Attendance List', 'bandr' =¿ 'Building/Room', 'staff' =¿ *staff*,' *designation*' =>
NULL,' *message*' =>message, 'pendingsCount' =¿ *this- > facultyattendancemodel- > getAllPendingConfirma*
¿load-¿view('templates/header-main'); *this- > load- > view('templates/header − admin'*,data);
this- > load- > view('pages/hr/general/update',data); *this- > load- > view('templates/footer'*,data);
else redirect(base$_u rl('login'$));

function syncFacultyData() *staff* =this-¿session-¿userdata('staff'); if(*staff*! = *NULL*)syncFacultyData
$= this- > api_m odel- > insertFacultyData(); redirect(base_u rl('update?message = true')); elseredirect(base_u rl('lo$

function syncClassSchedule() *staff* =this-¿session-¿userdata('staff'); if(*staff*! = *NULL*)syncClassSchedule
$= this- > api_m odel- > syncClassSchedule(); if$(syncClassSchedule) redirect(base$_u rl('update?message =$
true')); *elseredirect*(*base$_u rl('login'$*));*elseredirect*(*base$_u rl('login'$*));

function syncChangeVenueToAttendance() *this- > facultyattendancemodel- > AddChangeVenueToFacultyA*
venue − admin'));

function registration() *staff* =this-¿session-¿userdata('staff'); *routes* =this-¿facultyattendancemodel-
¿getRoutes(); *config*['*per$_p age'$] = 5;config['total$_r ows'$] =this-¿facultyattendancemodel-¿getAllHRAccountsCount();
config['*base$_u rl'$] = *base$_u rl('register'$*);config['uri$_s egment'$] = 2;config['num$_l inks'$] = *floor*(config['total$_r ows'$]/config
ulclass = "*pagination*" >';config['full$_t ag_c lose'$] =' < /ul >';config['attributes'] = ['class' =¿ 'page-
link']; *config*['*first$_l ink'$] = *false*;config['last$_l ink'$] = *false*;config['first$_t ag_o pen'$] =' < *liclass* =
"*page−item*" >';config['first$_t ag_c lose'$] =' < /li >';config['prev$_l ink'$] =' *laquo*';config['prev$_t ag_o pen'$] =' <
liclass = "*page−item*" >';config['prev$_t ag_c lose'$] =' < /li >';config['next$_l ink'$] =' *raquo*';config['next$_t ag_o pen'$] =' <

$liclass$ = "$page-item$" >';config['next$_tag_close$'] ='< /li >';config['last$_tag_open$'] ='< $liclass$ = "$page-item$" >';config['last$_tag_close$'] ='< /li >';config['cur$_tag_open$'] ='< $liclass$ = "$page-itemactive$" >< $ahref$ = ""$class$ = "$page-link$" >';config['cur$_tag_close$'] ='< $spanclass$ = "$sr-only$" > ($current$) < /span >< /a >< /li >';config['num$_tag_open$'] ='< $liclass$ = "$page-item$" >';config['num$_tag_close$'] ='< /li >'; // Pagination page = $this->uri->segment$(2);this-¿pagination-¿initialize($config$);accounts = $this->facultyattendancemodel->hrmoAccounts$(config['per$_page$'],page); $data = array('title' =>' HRMOAccounts',' staff' =>$staff, 'designation' =¿ NULL, 'accounts' =¿ $accounts$,'$routes$' =>routes, 'pendingsCount' =¿ $this->facultyattendancemodel->getAllPendingConfirmationCount()$; if(staff != null) $requiredPrivilege = array(0, 1)$; //listtheacceptedprivilegeif(this-¿isAuthorized($this->session->userdata('privilege')$,requiredPriv $this->load->view('templates/header-main')$;this-¿load-¿view('templates/header-admin', $data$);this-¿load-¿view('pages/hr/general/registration', $data$);this-¿load-¿view('templates/footer', $data$); $else$this-¿load-¿view('templates/header-main'); $this->load->view('templates/header-admin'$,data); $this->load->view('errors/unauthorized'$);this-¿load-¿view('templates/footer', $data$);

function inputConfirmChangeVenue($accept$,id) if($this->session->userdata('idNumber'))if$(accept == 0$ —— $accept == 1)if$(this-¿session-¿userdata('designation') == 'Dean' —— $this->session->userdata('designation') ==' Chairperson'$)this-¿facultyattendancemodel-¿confirmChangeVenue($accept$,id); elseif($this->session->userdata('privelege') == 0$)this-¿facultyattendancemodel-¿confirmChangeVenue($accept$,i if($this->session->userdata('staff') ==' staff'$)$redirect(base_url('change-venue-admin'))$; $elseredirect(base$ $venue'))$; $elseredirect(base_url('weblogin/logout'))$;$elseredirect(base_url('weblogin/logout'))$;

function isAuthorized($userPrivilege$,requiredPrivilege) foreach ($requiredPrivilegeas$priv) if($userPrivilege$ ==p return true; return false;

function changeVenue() $config['per_page'] = 5$;config['total$_rows$'] =this-¿facultyattendancemodel-¿getAllChangeVenueCount(); $config['base_url'] = base_url('change-venue-admin')$;config['uri$_segment$'] = 2; //config['num$_links$'] = $floor$(config['total$_rows$']/config['per$_page$']);config['num$_links$'] = 5;config['full$_tag_open$'] ='< $ulclass$ = "$pagination$" >';config['full$_tag_close$'] ='< /ul >';config['attributes'] = ['class' =¿ 'page-link']; $config['first_link'] = false$;config['last$_link$'] = $false$;config['first$_tag_open$'] ='< $liclass$ = "$page-item$" >';config['first$_tag_close$'] ='< /li >';config['prev$_link$'] =' laquo';config['prev$_tag_open$'] ='< $liclass$ = "$page-item$" >';config['prev$_tag_close$'] ='< /li >';config['next$_link$'] =' raquo';config['next$_tag_open$'] ='< $liclass$ = "$page-item$" >';config['next$_tag_close$'] ='< /li >';config['last$_tag_open$'] ='< $liclass$ = "$page-item$" >';config['last$_tag_close$'] ='< /li >';config['cur$_tag_open$'] ='< $liclass$ = "$page-$

itemactive" >< *ahref* = ""*class* = "*page* − *link*" >';config['cur$_t$*ag*$_c$*lose*'] ='< *spanclass* = "*sr* − *only*" > (*current*) < /*span* >< /*a* >< /*li* >';config['num$_t$*ag*$_o$*pen*'] ='< *liclass* = "*page* − *item*" >' ;config['num$_t$*ag*$_c$*lose*'] ='< /*li* >';this-¿pagination-¿initialize(*config*);page = *this*− > *uri*− > *segment*(config['uri$_s$*eg* = array('title' =¿ 'Change Venue', 'staff' =¿ 'staff', 'changeVenueDatas' =¿ *this*− > *facultyattendancemodel*− > *getAllChangeVenue*(config['per$_p$*age*'],page), 'class$_s$*cheds*' =>this-¿facultyattendancemodel-¿getAllClassSchedule(), 'rooms' =¿ *this*− > *facultyattendancemodel*− > *getRooms*(),'*pendingsCount*' =>this-¿facultyattendancemodel-¿getAllPendingConfirmationCount()); if(*this*− > *session*− > *userdata*('*staff*'))this-¿load-¿view('templates/heade main'); *this*− > *load*− > *view*('*templates/header*−*admin*',data); *this*− > *load*− > *view*('*pages/hr/general/chang this*− > *load*− > *view*('*templates/footer*',data); else *this*− > *load*− > *view*('*templates/header* − *main*');this-¿load-¿view('templates/header-admin', *data*);this-¿load-¿view('errors/unauthorized', *data*);this-¿load-¿view('templates/footer', *data*);

function pendingConfirmation() *config*['*per*$_p$*age*'] = 5;config['total$_r$*ows*'] =this-¿facultyattendancemodel-¿getAllPendingConfirmationCount(); *config*['*base*$_u$*rl*'] = *base*$_u$*rl*('*pending*−*confirmation*');config['uri$_s$*egment*'] = 2; //config['num$_l$*inks*'] = *floor*(config['total$_r$*ows*']/config['per$_p$*age*']);config['num$_l$*inks*'] = 5;config['full$_t$*ag*$_o$*pen*'] ='< *ulclass* = "*pagination*" >';config['full$_t$*ag*$_c$*lose*'] ='< /*ul* >';config['attributes'] = ['class' =¿ 'page-link']; *config*['*first*$_l$*ink*'] = *false*;config['last$_l$*ink*'] = *false*;config['first$_t$*ag*$_o$*pen*'] ='< *liclass* = "*page*−*item*" >';config['first$_t$*ag*$_c$*lose*'] ='< /*li* >';config['prev$_l$*ink*'] =' *laquo*';config['prev$_t$*ag*$_o$*pen*'] ='< *liclass* = "*page*−*item*" >';config['prev$_t$*ag*$_c$*lose*'] ='< /*li* >';config['next$_l$*ink*'] =' *raquo*';config['next$_t$*ag*$_o$*pen*'] ='< *liclass* = "*page* − *item*" >';config['next$_t$*ag*$_c$*lose*'] ='< /*li* >';config['last$_t$*ag*$_o$*pen*'] ='< *liclass* = "*page* − *item*" >';config['last$_t$*ag*$_c$*lose*'] ='< /*li* >';config['cur$_t$*ag*$_o$*pen*'] ='< *liclass* = "*page* − *itemactive*" >< *ahref* = ""*class* = "*page* − *link*" >';config['cur$_t$*ag*$_c$*lose*'] ='< *spanclass* = "*sr* − *only*" > (*current*) < /*span* >< /*a* >< /*li* >';config['num$_t$*ag*$_o$*pen*'] ='< *liclass* = "*page* − *item*" >' ;config['num$_t$*ag*$_c$*lose*'] ='< /*li* >';this-¿pagination-¿initialize(*config*);page = *this*− > *uri*− > *segment*(config['uri$_s$*eg* = array('title' =¿ 'Pending Confirmation', 'staff' =¿ 'staff', 'pendings' =¿ *this*− > *facultyattendancemodel*− > *getAllPendingConfirmation*(config['per$_p$*age*'],page), 'pendingsCount' =¿ *this*− > *facultyattendancemodel*− > *getAllPendingConfirmationCount*()); *if*(this-¿session-¿userdata('staff')) *this*− > *load*− > *view*('*templates/heade main*');this-¿load-¿view('templates/header-admin', *data*);this-¿load-¿view('pages/hr/faculty/pendingconfirmation', *data*);this-¿load-¿view('templates/footer', *data*); *else*this-¿load-¿view('templates/header-main'); *this*− > *load*− > *view*('*templates/header*−*admin*',data); *this*− > *load*− > *view*('*errors/unauthorized*',data); *this*− > *load*− > *view*('*templates/footer*',data);

function addRoomInfo() *data* = *array*('*ROOM*$_I$*D*' =>this-¿input-¿post('room'), 'BUILDING$_I$*D*' =>this-

¿input-¿post('buildingName'), 'ROOM$_NAME'$ =>this-¿input-¿post('roomName')); $this->facultyattendancemo$
$addRoom$(data); redirect(base$_url('class-rooms')$);

function addBuildingInfo() $data = array('BUILDING_NAME'$ =>this-¿input-¿post('buildingName'),
'ROUTE' =¿ $this->input->post('buildingRoute')$);this-¿facultyattendancemodel-¿addBuilding($data$); $redirect(b$
$list')$);

function addAccountInfo() $salt = file_get_contents('C:.txt')$;password = md5($this->input->$
$post('password')$.salt); if($this->input->post('privilege') == 3$||this-¿input-¿post('privilege')
$== 1)$ $checker_id = NULL$; dochecker$_id$ =this-¿api$_model->generate_unique_id(9)$; $while$(!this-
¿api$_model->is_unique_id_available('CHECKER_ID'$,checker$_id,'CHECKER'$));this-¿form$_validation->$
$set_rules('accountid','AccountID','required')$;this-¿form$_validation->set_rules('firstname','FirstName','require$
¿form$_validation->set_rules('middlename','MiddleName','required')$;this-¿form$_validation->$
$set_rules('lastname','LastName','required')$;this-¿form$_validation->set_rules('password','Password','trim|requi$
¿form$_validation->set_rules('route','Route','required')$;data1 = array('STAFF$_ID'$ =>this-¿input-
¿post('accountid'), 'CHECKER$_ID'$ =>checker$_id,'FIRST_NAME'$ =>this-¿input-¿post('firstname'),
'MIDDLE$_NAME'$ =>this-¿input-¿post('middlename'), 'LAST$_NAME'$ =>this-¿input-¿post('lastname'),
'PRIVILEGE' =¿ $this->input->post('privilege'),'PASSWORD'$ =>password); $data2 =$
$array('CHECKER_ID'$ =>checker$_id,'ROUTE'$ =>this-¿input-¿post('route'),); if($this->form_validation->$
$run() == TRUE)$this-¿facultyattendancemodel-¿registerCheckerAccount($data$1,data2); redirect(base$_url('register')$)
¿form$_validation->set_rules('accountid','AccountID','required')$;this-¿form$_validation->set_rules('firstname','$
¿form$_validation->set_rules('middlename','MiddleName','required')$;this-¿form$_validation->$
$set_rules('lastname','LastName','required')$;this-¿form$_validation->set_rules('password','Password','trim|requi$
= array('STAFF$_ID'$ =>this-¿input-¿post('accountid'), 'FIRST$_NAME'$ =>this-¿input-¿post('firstname'),
'MIDDLE$_NAME'$ =>this-¿input-¿post('middlename'), 'LAST$_NAME'$ =>this-¿input-¿post('lastname'),
'PRIVILEGE' =¿ $this->input->post('privilege'),'PASSWORD'$ =>password);

if($this->form_validation->run() == TRUE)$this-¿facultyattendancemodel-¿registerAccount($data$); $redirect$

function addChangeVenueInfo() $venue_id = NULL$; dovenue$_id$ =this-¿api$_model->generate_unique_id(9)$; $while$(!t
¿api$_model->is_unique_id_available('CHANGE_VENUE_ID'$,venue$_id,'CHANGE_VENUE_FORM'$));date$_of_change =$
¿input-¿post('date'); $data = array('CHANGE_VENUE_ID'$ =>venue$_id,'CLASS_SCHEDULE_ID'$ =>this-
¿input-¿post('class$_sched'),'ROOM_ID'$ =>this-¿input-¿post('room'), 'DATE$_OF_CHANGE'$ => $date$("$d-$
$M-y$", $strtotime$(date$_of_change$)),$'DATE_FILED'$ => $date('d-M-yh.i.sA'), //timestamp'DATE_APPROVED'$ =
$NULL,'ATTACHMENT'$ =>'','$CONFIRMED'$ => $NULL$);this-¿pagination-¿initialize(); $this->$

$facultyattendancemodel->fileChangeVenue$(data); if($this->session->userdata('staff') ==$ 'staff')$redirect(base_url('change-venue-admin'))$;$elseredirect(base_url('change-venue'))$;

function insertAllRooms() $staff$ =this-¿session-¿userdata('staff'); if($staff != NULL$)insertRoom $= this->api_model->insertAllRoom(); if($(insertRoom) redirect(base$_url('update?message = true'))$; $elseredirect(base_url('login'))$;$echojson_encode$(insertRoom); else redirect(base$_url('login'))$;

function editRoomInfo() $roomid$ =this-¿input-¿post('roomid'); $roomname$ =this-¿input-¿post('roomname'); $buildingid$ =this-¿input-¿post('buildingName'); if($this->session->userdata('privelege') == 0)if$(this-¿facultyattendancemodel-¿editRoom($roomid$,roomname, $buildingid$))$redirect(base_url('class-rooms'))$; $else$this-¿load-¿view('errors/failed'); else $this->load->view('errors/unauthorized')$;

function editBuilding() $buildingid$ =this-¿input-¿post('buildingId'); $buildingname$ =this-¿input-¿post('buildingName'); $buildingroute$ =this-¿input-¿post('buildingRoute'); if($this->session->userdata('privelege') == 0)this-¿facultyattendancemodel-¿editBuilding($buildingid$,buildingname, $buildingroute$); $re ist'))$;

function deleteBuilding() $buildingid$ =this-¿input-¿post('buildingid'); if($this->session->userdata('privelege') == 0)if$(!this-¿facultyattendancemodel-¿deleteBuilding($buildingid$))this-¿load-¿view('errors/failed'); else redirect(base$_url('building-list'))$; $else$this-¿load-¿view('errors/unauthorized');

function facultyList() $staff$ =this-¿session-¿userdata('staff'); if($staff != null$)data = array('staff' =¿ $staff,'designation' => NULL,'title' =>' FacultyList','pendingsCount' =>$this-¿facultyattendancemod ¿getAllPendingConfirmationCount()); $this->load->view('templates/header-main')$;this-¿load-¿view('templates/header-admin', $data$);this-¿load-¿view('pages/hr/faculty/facultylist', $data$);this-¿load-¿view('templates/footer', $data$); $elseredirect(base_url('login'))$;

function searchFaculty() $search$ =this-¿input-¿post('search'); $facultyData$ =this-¿facultyattendancemodel-¿searchFacultyName($search$);data['result'] = $facultyData$; $echojson_encode$(data);

function searchClass() $search$ =this-¿input-¿post('search'); $data['result']$ =this-¿facultyattendancemodel-¿searchClassSchedule($search$); $echojson_encode$(data);

function reports() $staff$ =this-¿session-¿userdata('staff'); if($staff != NULL$)data = array('title' =¿ 'Monthly Absence Report', 'schoolyears' =¿ $this->facultyreportmodel->getSchoolYear(),'colls' =>$this ¿facultyreportmodel-¿getCollege(), 'staff' =¿ $staff,'designation' => NULL,'pendingsCount' =>$this-¿facultyattendancemodel-¿getAllPendingConfirmationCount()); $this->load->view('templates/header-main')$;this-¿load-¿view('templates/header-admin', $data$);this-¿load-¿view('pages/hr/reports/reportPage',

data);this-¿load-¿view('templates/footer', *data*); *elseredirect*($base_url('login')$);

Faculty Summary Report Page function summaryreport() $staff$ =this-¿session-¿userdata('staff'); if($staff != NULL$)data = array('title' =¿ 'Faculty Attendance Summary Report', 'schoolyears' =¿ $this->facultyreportmodel->getSchoolYear()$,'colls' =>this-¿facultyreportmodel-¿getCollege(), 'staff' =¿ $staff$,'designation' => $NULL$,'pendingsCount' =>this-¿facultyattendancemodel-¿getAllPendingConfirm); $this->load->view('templates/header-main')$;this-¿load-¿view('templates/header-admin', *data*);this-¿load-¿view('pages/hr/reports/summaryreport', *data*);this-¿load-¿view('templates/footer', *data*); *elseredirect*($base_url('login')$);

MONTHLY REPORT FUNCTIONS function searchAbsenceData() *dept* =this-¿input-¿post('dept'); *coll* =this-¿input-¿post('coll'); *sem* =this-¿input-¿post('sem'); *month* =this-¿input-¿post('month'); *syear* =this-¿input-¿post('syear'); *data* =this-¿facultyreportmodel-¿getAbsenceData(*dept*,coll, *sem*,month, *syear*); *if*(data-¿$num_rows()$ > 0)*foreach*(data-¿$result_array()$*as*row)?¿ ¡tr¿ ¡td¿¡?=$row['LAST_NAME']$;? >, <? =row['FIRST$_N AME'$];? ><? =row['MIDDLE$_I NITIAL'$]? > . < /td >< td ><? =row['COLLEGE']; ?¿¡/td¿ ¡td¿¡?=$row['DEPARTMENT']$;? >< /td >< td ><? =row['ATTENDANCE$_D ATE'$];? >< /td >< td ><? =row['SUBJECT$_C ODE'$];? >< /td >< td ><? =row['STATUS'];?¿¡/td¿ ¡td¿¡/td¿ ¡td¿¡?=$row['SALARY_DEDUCTION']$;? >< /td >< /tr ><?*phpelse*? >< *tr* >< *tdcolspan* = "12" ><?*phpecho*′*N*

function getDepartment() *coll* =this-¿input-¿post('col'); *depts* =this-¿facultyreportmodel-¿getDepartment(*coll*); *f* as *dept*)? >< *optionvalue* = " <? =dept['DEPARTMENT'];?¿"¿¡?=$dept['DEPARTMENT']$;? >< /*option* ><?*php*

FACULTY SUMMARY REPORT FUNCTIONS get all the faculty of that department and college function findFaculty() *dept* =this-¿input-¿post('dept'); *col* =this-¿input-¿post('col'); *data* =this-¿facultyreportmodel-¿getFaculty(*dept*,col); if(*data*-> $num_rows()$ > 0)*foreach*(data-¿$result_array()$*as*row)?¿ ¡option value="¡?=$row['FACULTY_ID']$;? > " ><? =row['LAST$_N AME'$]; *echo*", "; *echo*row['FIRST$_N AME'$]; *echo*" . < /*option* ><?*php*

function searchFacultyData() *col* =this-¿input-¿post('col'); *dept* =this-¿input-¿post('dept'); *id* =this-¿input-¿post('ins'); *sem* =this-¿input-¿post('sem'); *syear* =this-¿input-¿post('syear'); *data* =this-¿facultyreportmodel-¿getFacultyData(*col*,dept, *id*,sem, *syear*); *if*(data-¿$num_rows()$ > 0)*foreach*(data-¿$result_array()$*as*row) ?¿ ¡tr¿ ¡td¿¡?=$row['LAST_NAME']$;? >,<? =row['FIRST$_N AME'$];? ><? =row['MIDDLE$_I NI$. < /td >< td ><? =row['COLLEGE']?¿¡/td¿ ¡td¿¡?=$row['DEPARTMENT']$;? >< /td >< td ><? =row['ATTENDANCE$_D ATE'$];? >< /td >< td ><? =row['STATUS'];?¿¡/td¿ ¡td¿¡/td¿ ¡td¿¡/td¿ ¡td¿¡?=$row['SALARY_DEDUCTION']$;? >< /td >< /tr ><?*phpelse*? >< *tr* >< *tdcolspan* = "12" ><?*p*

PRINT Monthly Absence Reportks function printMonthlyAbsenceReport() $this->load->library('PDF')$;coll = $this->input->post('coll')$;dept = $this->input->post('dept')$;sem = $this->input->post('sem')$;mon = $this->input->post('mon')$;syear = $this->input->post('syear')$;data = $this->facultyreportmodel->getAbsenceReportData$(coll, *dept*,sem, *mon*,syear); $info$ =this-¿facultyreportmodel-¿getAbsenceReportinfo(*mon*,year); *month* =this-¿months; foreach(*monthas*key =¿ $m)if$(mon == *key*)mon = m;info = $data->row()$;sem = $info->SEMESTER; if$(sem == 1) $sem =' 1st'; elseif$(sem == 2) $sem =' 2nd'; elseif$(sem == 3) $sem =' 2nd(Sum)'; else$sem = 'n/a'; STORE DATA INTO A TWO-DIMENSIONAL ARRAY $arr = array(); foreach$(data-¿result() as *row*)arr[] = array('id' =¿ $row->FACULTY_ID,' lname'$ =>row-¿LAST$_N AME,' fname'$ =>row-¿FIRST$_N AME,' mini'$ =>row-¿MIDDLE$_I NITIAL,' dept'$ =>row-¿DEPARTMENT, 'coll' =¿ $row->COLLEGE); GETUNIQUEARRAY, NODUPLICATEARRAY$finfo = array(); foreach($arras$row) if(!in$_a rray$(row, $finfo$))finfo[] = *row*;output = array(); foreach($data->result()as$row) $output[] = array('id'$ =>row-¿FACULTY$_I D,' date'$ =>row-¿ATTENDANCE$_D ATE,' scode'$ =>row-¿SUBJECT$_C ODE,' csec'$ = ¿CLASS$_S ECTION,' stime'$ =>row-¿START$_T IME,' etime'$ =>row-¿END$_T IME,' status'$ =>row-¿STATUS); $header = array('Date',' Subject',' Hours',' Remarks',' Minutes',' FiledLeave',' ForSalaryDeduction'$ = FALSE; *policy* = "";title = "MONTHLY FACULTY ABSENCE SUMMARY REPORTthe month of ".*mon*."".sem." Semester S/Y ".$info->SCHOOL_Y EAR$;note = "Note- This is a system generated so this does not require signature of the office representative."; PDF is located @ applications ¿¿ libraries $this->pdf = newPDF('L')$;this-¿pdf-¿SetMargins(15,10); $this->pdf->setTitleReport$(title); $this->pdf->setFooterNote$(note, *sumnote*,policy); $this->pdf->AddPage()$;this-¿pdf-¿monthlyReporTable(*output*,header,$finfo$); //this-¿pdf-¿monthlyReporTable(*data*,header); $this->pdf->Output()$;

PRINT Faculty Absence Summary Report function printFacultyReport() $this->load->library('PDF')$;col = $this->input->post('col')$;dept = $this->input->post('dept')$;id = $this->input->post('ins')$;sem = $this->input->post('sem')$;syear = $this->input->post('syear')$;data = $this->facultyreportmodel->printFacultyDataReport$(col, *dept*,id, *sem*,syear); *roow* =data-¿row(); *sem* =roow-¿SEMESTER; if(sem == 1)sem = '1st'; elseif(sem == 2)sem = '2nd'; elseif(sem == 3)sem = 'Sum'; else sem = "n/a";sumnote = TRUE; $policy = array('OB - OfficialBusiness',' MU-Make-UpClass',' AD-AdvanceClass',' SU-Subtitution',' AWOL-AbsentwithoutOfficialLeave',' TCw/oN-TransferofClassroomWithoutNotice',' SL-SickLeave',' EL-EmergencyLeave',' PL-PersonalLeave',' PR-Proctoring',' ED-EarltDismissal',' LWOW -$

$LeaveWithoutPay', 'PTL-PaternityLeave', 'STL-StudeyLeave', 'ML-MaternityLeave'$);months = array(); foreach($data->result()as$row) $months[] = array('currmon' => date('n', strtotime(row-¿ATTENDANCE$_{D}ATE$))); $GETUNIQUEARRAY, NODUPLICATEARRAY$umon = array(); foreach($monthsas if(!in$_{a}rray$(row, $umon$))umon[] = row;dat = array(); foreach($data->result()as$row) $dat[]$ = $array('date' =>$row-¿ATTENDANCE$_{D}ATE, 'scode' =>$row-¿SUBJECT$_{C}ODE, 'csec' =>$row-¿CLASS$_{S}ECTION$, ¿START$_{T}IME, 'etime' =>$row-¿END$_{T}IME, 'status' =>$row-¿STATUS); $title = "FACULTY ATTENDANCES$ Semester S/Y ".$roow->SCHOOL_{Y}EAR$;note = "*Total class hours served on regular class schedule + (MU + AD):Policy statement 4. Absence made-up for within the cap for conducting make-up class although reflected in the attendance record shall not be counted against the teacher and therefore shall not affect computations for merit points, permanecy etc. (Policy and guideline on Make-up Class, Appendix B. College Faculty Manual):"; PDF is located @ applications ¿¿ libraries $this->pdf = newPDF('L')$;this-¿pdf-¿SetMargins(15,10); $this->pdf->setTitleReport$(title); $this->pdf->setFooterNote$(note, $sumnote$,policy); $this->pdf->AddPage$();this-¿pdf-¿facultySummaryReport($data$,dat, $umon$);this-¿pdf-¿Output();

A.2.2 Faculty.php

[language=PHP] ¡? class Faculty extends CI$_{C}ontrollervar$id; var $designation; var$department; var $college; var$fname; var $lname; var$staff;

constructor function $_{construct()parent::_{construct();}}$this-¿id = $this->session->userdata('idNumber')$;this-¿designation = $this->session->userdata('designation')$;this-¿college = $this->session->userdata('college')$;this-¿department = $this->session->userdata('department')$;this-¿fname = $this->session->userdata('fName')$;this-¿lname = $this->session->userdata('lName')$;this-¿staff = NULL;

function index() $countPendingConfirmation$ =this-¿facultymodel-¿countPendingConfirmation($this->id$);countConfirmedPresent = $this->facultymodel->countPresentStatus$(this-¿id); $countConfirmedAbsent$ =t ¿facultymodel-¿countAbsentStatus($this->id$);todaysAbsentRecord = $this->facultymodel->getTodaysAbsentRecord$(this-¿id); $todaysAbsentRecordDepartment$ =this-¿facultymodel-¿getTodaysAbsentRecordI $department$);data = array('title' =¿ 'Dashboard', 'designation' =¿ $this->designation, 'staff' =>$this-¿staff, 'countPendingConfirmation' =¿ $countPendingConfirmation->COUNT, 'countConfirmedPresent' =>$co ¿COUNT, 'countConfirmedAbsent' =¿ $countConfirmedAbsent->COUNT, 'todaysAbsentRecord' =>$todaysAbse); if($this->designation == 'CollegeFaculty'$)this-¿load-¿view('templates/header-main'); $this->$

load− > *view*('*templates*/*header*−*faculty*',data); *this*− > *load*− > *view*('*pages*/*faculty*/*my*/*mydashboard*',data); *this*− > *load*− > *view*('*templates*/*footer*',data); elseif(*this*− > *designation* ==' *Chairperson*')data = array('title' =¿ 'Dashboard', 'designation' =¿ *this*− > *designation*,' *staff*' =>this-¿staff, 'countPendingConfirmation' =¿ *countPendingConfirmation*− > *COUNT*,' *countConfirmedPresent*' =>countConfirme ¿COUNT, 'countConfirmedAbsent' =¿ *countConfirmedAbsent*− > *COUNT*,' *todaysAbsentRecord*' =>todaysAbse 'todaysAbsentRecordDept' =¿ *todaysAbsentRecordDepartment*,' *changeVenueRequestCount*' =>this-¿facultyattendancemodel-¿getDeptChangeVenueRequestCount(*this*− > *department*),' *appealCount*' =>this-¿facultymodel-¿getAttendanceAppealsCount(*this*− > *college*,this-¿department)); *this*− > *load*− > *view*('*templates*/*header*−*main*');this-¿load-¿view('templates/header-faculty', *data*);this-¿load-¿view('pages/faculty/ *data*);this-¿load-¿view('templates/footer', *data*); *elseif*(this-¿designation == 'Dean') *data* = *array*('*title*' =>' *Dashboard*',' *designation*' =>this-¿designation, 'staff' =¿ *this*− > *staff*,' *countPendingConfirmation*' =>countPe ¿COUNT, 'countConfirmedPresent' =¿ *countConfirmedPresent*− > *COUNT*,' *countConfirmedAbsent*' =>count ¿COUNT, 'todaysAbsentRecord' =¿ *todaysAbsentRecord*,' *todaysCollegeAbsentRecord*' =>this-¿facultymodel-¿getTodaysAbsentRecordCollege(*this*− > *college*));this-¿load-¿view('templates/header-main'); *this*− > *load*− > *view*('*templates*/*header*−*faculty*',data); *this*− > *load*− > *view*('*pages*/*faculty*/*dean*/*dea* *this*− > *load*− > *view*('*templates*/*footer*',data); else redirect(base$_u rl$('*login*'));

function profileIndex() *data* = *array*('*title*' =>' *Profile*',' *profile*' =>this-¿facultymodel-¿getMyProfile(*this*− > *id*),' *designation*' =>this-¿designation, 'staff' =¿ *this*− > *staff*,' *countPendingConfirmation*' =>this-¿facultymodel-¿countPendingConfirmation(*this*− > *id*)− > *COUNT*); *if*(this-¿designation == 'College Faculty') *this*− > *load*− > *view*('*templates*/*header*−*main*');this-¿load-¿view('templates/header-faculty', *data*);this-¿load-¿view('pages/faculty/my/myprofile', *data*);this-¿load-¿view('templates/footer', *data*); *elseif*(this-¿designation == 'Chairperson') *this*− > *load*− > *view*('*templates*/*header* − *main*');this-¿load-¿view('templates/header-faculty', *data*);this-¿load-¿view('pages/faculty/my/myprofile', *data*);this-¿load-¿view('templates/footer', *data*); *elseif*(this-¿designation == 'Dean') *this*− > *load*− > *view*('*templates*/*header*−*main*');this-¿load-¿view('templates/header-faculty', *data*);this-¿load-¿view('pages/faculty/ *data*);this-¿load-¿view('templates/footer', *data*);

function myDailyReport() *data* = *array*('*count*$_p$*ending*' =>this-¿facultymodel-¿countPendingConfirmation(*this*- *id*),' *count*$_a$*bsent*' =>this-¿facultymodel-¿countAbsentStatus(*this*− > *id*),' *count*$_p$*resent*' =>this-¿facultymodel-¿countPresentStatus(*this*− > *id*),' *designation*' =>this-¿designation, 'staff' =¿ *this*− > *staff*);this-¿load-¿view('templates/header-main'); *this*− > *load*− > *view*('*templates*/*header* − *faculty*',data); *this*− > *load*− > *view*('*pages*/*faculty*/*my*/*mydailyreport*',data); *this*− > *load*− >

view('*templates*/*footer*',data);

function myClassSchedule() *id* =this-¿session-¿userdata('idNumber'); *search* = *strtolower*(this-¿input-¿post('search')); *hidden* =this-¿input-¿post('hidden'); if(*search*! = *null*)this-¿session-¿set$_u$*serdata*(*array*('*sea*
search =this-¿session-¿userdata('search'); else if(*hidden*)this-¿session-¿unset$_u$*serdata*('*search*');search
= *this*− > *session*− > *userdata*('*search*');config['per$_p$*age*'] = 5;config['total$_r$*ows*'] =this-¿facultymodel-¿getMyScheduleCount(*search*,id); *config*['*base*$_u$*rl*'] = *base*$_u$*rl*('*my*−*class*−*schedule*');config['uri$_s$*egment*'] =
2;config['num$_l$*inks*'] = *floor*(config['total$_r$*ows*']/config['per$_p$*age*']); // *Pagination* config['full$_t$*ag*$_o$*pen*'] =′< *ulclass* = "*pagination*" >′;config['full$_t$*ag*$_c$*lose*'] =′< /*ul* >′;config['attributes'] = ['class' =¿ 'page-link']; *config*['*first*$_l$*ink*'] = *false*;config['last$_l$*ink*'] = *false*;config['first$_t$*ag*$_o$*pen*'] =′< *liclass* = "*page*−*item*" >′;config['first$_t$*ag*$_c$*lose*'] =′< /*li* >′;config['prev$_l$*ink*'] =′ *laquo*′;config['prev$_t$*ag*$_o$*pen*'] =′< *liclass* = "*page*−*item*" >′;config['prev$_t$*ag*$_c$*lose*'] =′< /*li* >′;config['next$_l$*ink*'] =′ *raquo*′;config['next$_t$*ag*$_o$*pen*'] =′< *liclass* = "*page* − *item*" >′;config['next$_t$*ag*$_c$*lose*'] =′< /*li* >′;config['last$_t$*ag*$_o$*pen*'] =′< *liclass* = "*page* − *item*" >′;config['last$_t$*ag*$_c$*lose*'] =′< /*li* >′;config['cur$_t$*ag*$_o$*pen*'] =′< *liclass* = "*page* − *itemactive*" >< *ahref* = ""*class* = "*page* − *link*" >′;config['cur$_t$*ag*$_c$*lose*'] =′< *spanclass* = "*sr* − *only*" > (*current*) < /*span* >< /*a* >< /*li* >′;config['num$_t$*ag*$_o$*pen*'] =′< *liclass* = "*page* − *item*" >′;config['num$_t$*ag*$_c$*lose*'] =′< /*li* >′; // *Pagination* this-¿pagination-¿initialize(*config*);page = *this*− > *uri*− > *segment*(2);data = array('title' =¿ 'My Class Schedule', 'my$_s$*chedule*′ =>this-¿facultymodel-¿getMySchedulePagination(*search*,this-¿id, *config*['*per*$_p$*age*'],page), 'designation' =¿ *this*− > *designation*,′ *staff*′ =>this-¿staff, 'searched' =¿ *search*,′ *countPendingConfirmation*′ =>this-¿facultymodel-¿countPendingConfirmation(*this*− > *id*)− > *COUNT*,′ *changeVenueRequestCount*′ =>this-¿facultyattendancemodel-¿getDeptChangeVenueRequestCount(*this*− > *department*),′ *appealCount*′ =>this-¿facultymodel-¿getAttendanceAppealsCount(*this*− > *college*,this-¿department)); *this*− > *load*− > *view*('*templates*/*header*−*main*');this-¿load-¿view('templates/header-faculty', *data*);this-¿load-¿view('pages/faculty/*data*);this-¿load-¿view('templates/footer', *data*);

function myDailyAttendance() *data* = *array*('*title*′ =>′ *MyDailyAttendance*′,′ *my*$_r$*ecord*′ =>this-¿facultymodel-¿getMyRecord(*this*− > *id*),′ *designation*′ =>this-¿designation, 'staff' =¿ *this*− > *staff*,′ *countPendingConfirmation*′ =>this-¿facultymodel-¿countPendingConfirmation(*this*− > *id*)− > *COUNT*,′ *changeVenueRequestCount*′ =>this-¿facultyattendancemodel-¿getDeptChangeVenueRequestCount(*this*− > *department*),′ *appealCount*′ =>this-¿facultymodel-¿getAttendanceAppealsCount(*this*− > *college*,this-¿department)); *this*− > *load*− > *view*('*templates*/*header*−*main*');this-¿load-¿view('templates/header-faculty', *data*);this-¿load-¿view('pages/faculty/my/mydailyattendance', *data*);this-¿load-¿view('templates/footer',

data);

function myPendingConfirmation() $search = strtolower$(this-¿input-¿post('search')); $hidden$ =this-¿input-¿post('hidden'); if($search! = null$)this-¿session-¿set$_u serdata(array('search' =>$search)); $search$ =this-¿session-¿userdata('search'); else if($hidden$)this-¿session-¿unset$_u serdata('search')$;search = $this- > session- > userdata('search')$;config['per$_p age'] = 5$;config['total$_r ows'] =$this-¿facultymodel-¿confirmNoticeCount(t id);config['base$_u rl'] = base_u rl('my-pending-confirmation')$;config['uri$_s egment'] = 2$;config['num$_l inks'] = floor$(config['total$_r ows'$]/config['per$_p age'$]); // $Pagination$ config['full$_t ag_o pen'] =' < ulclass = "pagination" >'$;config['full$_t ag_c lose'] =' < /ul >'$;config['attributes'] = ['class' =¿ 'page-link']; $config['first_l ink'] = false$;config['last$_l ink'] = false$;config['first$_t ag_o pen'] =' < liclass = "page-item" >'$;config['first$_t ag_c lose'] =' < /li >'$;config['prev$_l ink'] =' laquo'$;config['prev$_t ag_o pen'] =' < liclass = "page-item" >'$;config['prev$_t ag_c lose'] =' < /li >'$;config['next$_l ink'] =' raquo'$;config['next$_t ag_o pen'] =' < liclass = "page - item" >'$;config['next$_t ag_c lose'] =' < /li >'$;config['last$_t ag_o pen'] =' < liclass = "page - item" >'$;config['last$_t ag_c lose'] =' < /li >'$;config['cur$_t ag_o pen'] =' < liclass = "page - itemactive" >< ahref = ""class = "page - link" >'$;config['cur$_t ag_c lose'] =' < spanclass = "sr - only" > (current) < /span >< /a >< /li >'$;config['num$_t ag_o pen'] =' < liclass = "page - item" >'$;config['num$_t ag_c lose'] =' < /li >'$; // $Pagination$ this-¿pagination-¿initialize($config$);page = $this- > uri- > segment$(2);data = array('title' =¿ 'Pending Confirmation', 'my$_p ending' =>$this-¿facultymodel-¿confirmNoticePagination($this- > id$,config['per$_p age'$],page), 'designation' =¿ $this- > designation,' staff' =>$this-¿staff, 'searched' =¿ $search,' countPendingConfirmation' =>$this-¿facultymodel-¿countPendingConfirmation($this- > id) - > COUNT,' changeVenueRequestCount' =>$this-¿facultyattendancemodel-¿getDeptChangeVenueRequestCount($this- > department),' appealCount' =>$this-¿facultymodel-¿getAttendanceAppealsCount($this- > college$,this-¿department)); $this- > load- > view('templates/header-main')$;this-¿load-¿view('templates/header-faculty', $data$);this-¿load-¿view('pages/faculty/$data$);this-¿load-¿view('templates/footer', $data$);

function called when the faculty confirm their status from the pending confirmations function confirmStatus(aId,cId, $action$,reason) $reason = str_r eplace('if($!empty(reason)) if($action == 1)confisconfirm$this-¿facultymodel-¿updateStatus(aId,action); $this- > facultymodel- > confirmedStatus$(cId, $reason$); $else//validated$1this-¿facultymodel-¿updateValidated(aId);this-¿facultymodel-¿confirmedStatus(cId,reason redirect(base$_u rl('my-pending-confirmation'))$; $elseredirect(base_u rl('my - pending - confirmation'))$;

display all absence appeal to their respective department function absenceAppeal() $data = array('title' =>' AbsenceAppeals',' appeals' =>$this-¿facultymodel-¿attendanceAppeals($this- >$

college,this-¿department), 'designation' =¿ *this*− > *designation*,′ *staff*′ =>this-¿staff, 'changeV-enueRequestCount' =¿ *this*− > *facultyattendancemodel*− > *getDeptChangeVenueRequestCount*(this-¿department), 'appealCount' =¿ *this*− > *facultymodel*− > *getAttendanceAppealsCount*(this-¿college, *this*− > *department*));this-¿load-¿view('templates/header-main'); *this*− > *load*− > *view*(′*templates*/*head*
faculty′,data); *this*− > *load*− > *view*(′*pages*/*faculty*/*chairperson*/*absenceappeal*′,data); *this*− > *load*− > *view*(′*templates*/*footer*′,data);

function called when the chairperson validates an appeal of the faculty member function validate(*aid*,cid, *action*,remarks) *this*− > *facultymodel*− > *validateAppeal*(aid, *cid*,action, *remarks*); *redirect*(*base*$_u$*rl*(′*absence*− *appeal*′));

function facultyAttendanceRecord() *config*[′*per*$_p$*age*′] = 5;config['total$_r$*ows*′] =this-¿facultymodel-¿getFacultyAttendanceRecordChairCount(*this*− > *department*);config['base$_u$*rl*′] = *base*$_u$*rl*(′*faculty*− *attendance*−*record*′);config['uri$_s$*egment*′] = 2;config['num$_l$*inks*′] = *floor*(config['total$_r$*ows*′]/config['per$_p$*age*′]); // *ulclass* = "*pagination*" >′;config['full$_t$*ag*$_c$*lose*′] =′< /*ul* >′;config['attributes'] = ['class' =¿ 'page-link']; *config*[′*first*$_l$*ink*′] = *false*;config['last$_l$*ink*′] = *false*;config['first$_t$*ag*$_o$*pen*′] =′< *liclass* = "*page*−*item*" >′;config['first$_t$*ag*$_c$*lose*′] =′< /*li* >′;config['prev$_l$*ink*′] =′ *laquo*′;config['prev$_t$*ag*$_o$*pen*′] =′< *liclass* = "*page*−*item*" >′;config['prev$_t$*ag*$_c$*lose*′] =′< /*li* >′;config['next$_l$*ink*′] =′ *raquo*′;config['next$_t$*ag*$_o$*pen*′] =′< *liclass* = "*page* − *item*" >′;config['next$_t$*ag*$_c$*lose*′] =′< /*li* >′;config['last$_t$*ag*$_o$*pen*′] =′< *liclass* = "*page* − *item*" >′;config['last$_t$*ag*$_c$*lose*′] =′< /*li* >′;config['cur$_t$*ag*$_o$*pen*′] =′< *liclass* = "*page* − *itemactive*" >< *ahref* = ""*class* = "*page* − *link*" >′;config['cur$_t$*ag*$_c$*lose*′] =′< *spanclass* = "*sr* − *only*" > (*current*) < /*span* >< /*a* >< /*li* >′;config['num$_t$*ag*$_o$*pen*′] =′< *liclass* = "*page* − *item*" >′ ;config['num$_t$*ag*$_c$*lose*′] =′< /*li* >′; // *Pagination* this-¿pagination-¿initialize(*config*);page = *this*− > *uri*− > *segment*(2);coll = *this*− > *session*− > *userdata*(′*college*′);dept = *this*− > *session*− > *userdata*(′*department*′);data = array('title' =¿ 'Faculty Attendance Record', 'record' =¿ *this*− > *facultymodel*− > *getFacultyAttendanceRecordChair*(this-¿department, *config*[′*per*$_p$*age*′],page), 'designation' =¿ *this*− > *designation*,′ *staff*′ =>this-¿staff, 'changeVenueRequestCount' =¿ *this*− > *facultyattendancemodel*− > *getDeptChangeVenueRequestCount*(this-¿department), 'appealCount' =¿ *this*− > *facultymodel*− > *getAttendanceAppealsCount*(this-¿college, *this*− > *department*));this-¿load-¿view('templates/header-main'); *this*− > *load*− > *view*(′*templates*/*header* − *faculty*′,data); *this*− > *load*− > *view*(′*pages*/*faculty*/*chairperson*/*chairfacultyrecord*′,data); *this*− > *load*− > *view*(′*templates*/*footer*′,data);

function changeVenue() *data* = *array*(′*title*′ =>′ *ChangeVenue*′,′ *designation*′ =>this-¿designation,

'staff' =¿ $this->staff$,'$changeVenueDatas$' =>this-¿facultyattendancemodel-¿getChangeVenue($this->id$),'$class_scheds$' =>this-¿facultyattendancemodel-¿getClassScheduleFaculty($this->id$),'$rooms$' =>this-¿facultyattendancemodel-¿getRooms(), 'changeVenueRequestDatas' =¿ $this->facultyattendancemodel->getDeptChangeVenueRequest$(this-¿department), 'countPendingConfirmation' =¿ $this->facultymodel->countPendingConfirmation$(this-¿id)-¿COUNT, 'changeVenueRequestCount' =¿ $this->facultyattendancemodel->getDeptChangeVenueRequestCount$(this-¿department), 'appealCount' =¿ $this->facultymodel->getAttendanceAppealsCount$(this-¿college, $this->department$));this-¿load-¿view('templates/header-main'); $this->load->view('templates/header-faculty'$,data); $this->load->view('pages/faculty/changeve$ $this->load->view('templates/footer'$,data);

function homeCollege() $data = array('title' =>' HomeCollege','designation' =>$this-¿designation, 'staff' =¿ $this->staff$);this-¿load-¿view('templates/header-main'); $this->load->view('templates/header-faculty'$,data); $this->load->view('pages/faculty/dean/homeCollege'$,data); $this->load->view('templates/footer'$,data);

function facultyDailyReport() $data = array('title' =>' FacultyDailyAttendance','records' =>$this-¿facultymodel-¿getFacultyAttendanceRecordDean($this->college$),'$designation$' =>this-¿designation, 'staff' =¿ $this->staff$);this-¿load-¿view('templates/header-main'); $this->load->view('templates/header-faculty'$,data); $this->load->view('pages/faculty/dean/deanfacultyrecord'$,data); $this->load->view('templates/footer'$,data); Dean pages ending

SEARCH FUNCTIONS function searchAttendance() mon =this-¿input-¿post('mon'); day =this-¿input-¿post('day'); $data$ =this-¿facultymodel-¿getAttendance(mon,day, $this->id$); if(data-¿num$_rows$() > 0)$foreach$(data-¿result$_array$()asrow) ?¿ ¡tr¿ ¡td¿¡?=$row['ATTENDANCE_DATE'$]; ? >< /td >< td ><? =row['CLASS$_DAY'$]; ? >< /td >< td ><? =row['START$_TIME'$]; ? > - <? =row['END$_TIME'$]; ? >< /td >< td ><? =row['SUBJECT$_CODE'$]; ? >< /td >< td ><? =row['ROOM$_ID'$]; ? >< /td >< td ><? =row['STATUS']; ?¿¡/td¿ ¡/tr¿ ¡?php else ?¿ ¡tr¿ ¡td colspan="12"¿¡?php echo 'No Data Found!'; ?¿¡/td¿ ¡/tr¿ ¡?php

function fetchMyClassSchedule() $getData$ =this-¿facultymodel-¿fetchMyClassScheduleAjax($this->id$);output = array("data" =¿ $getData->result_array$());

echo json$_encode$(output);

function fetchMyAttendance() $getData$ =this-¿facultymodel-¿fetchMyAttendanceAjax($this->id$);output = array("data" =¿ $getData->result_array$());

echo json$_encode$(output); ?¿

A.2.3 API.php

[language=PHP] ¡?php defined('BASEPATH') OR exit('No direct script access allowed'); class API extends CI$_{C}ontrollerpublicfunction_{c}onstruct()parent::_{c}onstruct();$ this-¿load-¿model('api$_{m}odel'$);this-¿load-¿helper('url$_{h}elper'$);this-¿load-¿helper('file');

public function login() $username =_{P}OST['username'];$salt = 'r@1peuf@$ci0'$;saltedhashedPassword = md5($_{P}OST['password'].$salt); $data =$this-¿api$_{m}odel- > login$(username, $saltedhashedPassword); echojson_{e}ncod$

public function faculty($fid = null$)id = $_{P}OST['id'];$token = $_{P}OST['token'];$data = $this- > api_{m}odel- > get_{f}aculty$(fid, id,token); echo json$_{e}ncode$(data);

public function staff($sid = null$)id = $_{P}OST['id'];$token = $_{P}OST['token'];$data = $this- > api_{m}odel- > get_{s}taff$(sid, id,token); echo json$_{e}ncode$(data);

public function room($rid = null$)id = $_{P}OST['id'];$token = $_{P}OST['token'];$data = $this- > api_{m}odel- > get_{r}oom$(rid, id,token); echo json$_{e}ncode$(data);

public function route($rid = null$)id = $_{P}OST['id'];$token = $_{P}OST['token'];$data = $this- > api_{m}odel- > get_{r}oute$(rid, id,token); echo json$_{e}ncode$(data);

public function confirmation$_{n}otice$(cid = null) $id =_{P}OST['id'];$token = $_{P}OST['token'];$data = $this- > api_{m}odel- > get_{c}onfirmation_{n}otice$(cid, id,token); echo json$_{e}ncode$(data);

public function class$_{s}chedule$(cid = null) $id =_{P}OST['id'];$token = $_{P}OST['token'];$data = $this- > api_{m}odel- > get_{c}lass_{s}chedule$(id, $token$); $echojson_{e}ncode$(data);

public function post$_{f}aculty_{a}ttendance$(faid = null) // TEMPORARY $path =' assets/images/';$id = $_{P}OST['id'];$token = $_{P}OST['token'];$cnid = $_{P}OST['cnid'];if($!(this-¿api$_{m}odel- > check_{s}taff_{a}uthorization$(id, $token$)))$echojson_{e}ncode(array('status' =>$ 401,$' message' =>' UnauthorizedAccess'));elseif$(this-¿api$_{m}odel- > check_{t}oken_{a}vailability$(id)) $faid =_{P}OST['faid'];$staff$_{i}d =_{P}OST['sid'];$class$_{s}chedule_{i}d =_{P}OST['csid'] null; else$first$_{c}heck =_{P}OST['fcheck'];if(empty(_{P}OST['scheck']))$second$_{c}heck = null; else$second$_{c}heck =_{P}OST['sch = _{P}OST['status'];if(_{P}OST['validated'] ==$ "$null$")validated = null; else $validated =_{P}OST['validated'];if(_{P}OST['$ "$null$")nid = null; else $nid =_{P}OST['nid'];$first$_{i}mage['filename'] = null;$first$_{i}mage['upload_{p}ath'] =$path; $first_{i}mage['allowed_{t}ypes'] =' gif|jpg|png|bmp|jpeg';$first$_{i}mage['max_{s}ize'] =' 0';$first$_{i}mage['encrypt_{n}ame'] = false;$this-¿load-¿library('upload', $first_{i}mage$);second$_{i}mage['filename'] = null;$second$_{i}mage['upload_{p}ath'] =$path; $second_{i}mage['allowed_{t}ypes'] =' gif|jpg|png|bmp|jpeg';$second$_{i}mage['max_{s}ize'] =' 0';$second$_{i}mage['encrypt_{n}ame'] = false;$this-¿load-¿library('upload', $second_{i}mage$); // $UploadFirstImage$ //if(!this-¿upload-¿do$_{u}pload('fipath'))$uploadedDetails = $this- > upload- > display_{e}rrors(); else$first$_{i}mage['filename'] =$this-

¿upload-¿data('file$_name$'); // *UploadSecondImage* //*if*(!this-¿upload-¿do$_upload$('*sipath*'))uploadedDetails = *this*− > *upload*− > *display*$_errors$(); *else*second$_image$['*filename*'] =this-¿upload-¿data('file$_name$');data = *this*− > *api*$_model$− > *post*$_faculty_attendance$(faid, *id*,token, *staff*$_id$,class$_schedule_id$,attendance$_date$,nid, *first*$_check$,second$_check$,path.*first*$_image$['*filename*'],path.*second*$_image$['*filename*'],status, *validated*,cnid); echo json$_encode$(data); else echo json$_encode$(*array*('*status*' => 402,'*message*' =>' *TokenhasExpired*'));

public function get$_faculty_attendance$(faid = null) *id* =$_POST$['*id*'];token = $_POST$['*token*'];data = *this*− > *api*$_model$− > *get*$_faculty_attendance$(id, *token*); *echojson*$_encode$(data);

public function post$_confirmation_notice$()path = 'assets/signatures/'; *id* =$_POST$['*id*'];token = $_POST$['*token*']; *if*(!(this-¿api$_model$− > *check*$_staff_authorization$(id, *token*)))*echojson*$_encode$(*array*('*status*' => 401 ¿api$_model$− > *check*$_token_availability$(id)) *faid* =$_POST$['*faid*'];cnid = $_POST$['*cnid*'];date = $_POST$['*date*'];reason = $_POST$['*reason*'];remarks = $_POST$['*remarks*'];confirmed = $_POST$['*confirmed*'];signature$_image$['*filename*'] = *null*;signature$_image$['*upload*$_path$'] =path; *signature*$_image$['*allowed*$_types$'] =' *gif*|*jpg*|*png*|*bmp*|*jpeg*';signature$_image$ 0';signature$_image$['*encrypt*$_name$'] = *true*;this-¿load-¿library('upload', *signature*$_image$);

// Upload Signature // if(!*this*− > *upload*− > *do*$_upload$('*esignature*'))uploadedDetails = *this*− > *upload*− > *display*$_errors$(); *echojson*$_encode$(uploadedDetails); else *signature*$_image$['*filename*'] =this-¿upload-¿data('file$_name$');data = *this*− > *api*$_model$− > *post*$_confirmation_notice$(faid, *cnid*,date, *reason*,path.*signature*$_image$['*filename*'],remarks, *confirmed*); *echojson*$_encode$(data); else echo json$_encode$(*array*('*s* 402,'*message*' =>' *TokenhasExpired*','*image*' =>signature$_image$['*filename*']));

public function get$_confirmation_notice$()id = $_POST$['*id*'];token = $_POST$['*token*']; *if*(!(this-¿api$_model$− > *check*$_staff_authorization$(id, *token*)))*echojson*$_encode$(*array*('*status*' => 401,'*message*' =>' *UnauthorizedAccess*') = *this*− > *api*$_model$− > *get*$_confirmation_notice$(); *echojson*$_encode$(data);

public function get$_all_count$()id = $_POST$['*id*'];token = $_POST$['*token*'];data = *this*− > *api*$_model$− > *get*$_all_count$(id, *token*); *echojson*$_encode$(data);

public function test() *data* =this-¿api$_model$− > *test*(); *echojson*$_encode$(data);

public function syncFacultyData() *data* =this-¿api$_model$− > *getEmployeeID*(); *echojson*$_encode$(data);

public function note() *data* =this-¿api$_model$− > *getNote*(); *echojson*$_encode$(data);

A.3 Web Application - Models

A.3.1 APIModel.php

[language=PHP] ¡?php /* API$_{m}odel.php$

ADNU-HRMO College Faculty Attendance Monitoring System Ralph San Jose Eufracio 4th Year BSIT Student College of Computer Studies, Department of Computer Science */ defined('BASEPATH') OR exit('No direct script access allowed'); ini$_{s}et('max_{e}xecution_{t}ime', 0); ini_{s}et('memory_{l}imit', '2048M');$

class API$_{m}odelextendsCI_{M}odel$

public function $_{c}onstruct()$this-¿load-¿database();

public function generate$_{u}nique_{i}d$(size) $characters =$' 0123456789$abcdefghijklmnopqrstuvwxyzABCDEFGHIJ$ = strlen($characters$);randomString = '';

for ($i = 0$;i ¡ $size$;i++) $randomString.$ =characters[rand(0, $charactersLength - 1$)];

return $randomString$;

public function is$_{u}nique_{i}d_{a}vailable$(uniquecolname, uid,table) // returns true if the id has "NO" duplicate $this- > db- > select('*')$;this-¿db-¿from($table$);this-¿db-¿where($uniquecolname$,uid);

$query$ =this-¿db-¿get()-¿row();

if($query == null)returntrue; elsereturnfalse;$

function isScheduleUnique($subCode$,section, $faculty_{i}d)//returnstrueifclassScheduleisunique$this-¿db-¿select('CLASS$_{S}CHEDULE_{I}D'$);this-¿db-¿from('CLASS$_{S}CHEDULE'$);where = array('FACULTY$_{I}D'$ =>fac 'CLASS$_{S}ECTION'$ =>section); $this- > db- > where$(where); $query$ =this-¿db-¿get()-¿row();

if($query == null)returntrue;elsereturnfalse;$

public function check$_{s}taff_{a}uthorization$(id, $token)//returnstrueifthestaffisexistingandhaveavalidtoken//que = this- > db- > select('*')- > from('STAFF')- > where('STAFF_{I}D'$,id)-¿where('TOKEN', $token)- > get()- > row(); //select * fromstaffwherestaff_{i}d =$id and where token = $token$;this-¿db-¿select('STAFF$_{I}D'$);this-¿db-¿from('STAFF'); $this- > db- > join('CHECKER', 'CHECKER.CHECKER_{I}D = STAFF.CHECKER_{I}D', 'inner'$);this-¿db-¿where('STAFF.STAFF$_{I}D'$,id)-¿where('CHECKER.TOKEN', $token$);query = $this- > db- > get()- > row()$;

if($query == '')//ifthequerydoesnotreturnanydataornodatafoundreturnfalse;elsereturntrue;$

public function check$_{t}oken_{a}vailability$(id) //wil return false if the token is expired $query$ =this-¿db-¿select('STAFF$_{I}D, LAST_{L}OGIN'$)- > $from('STAFF')- > where('STAFF_{I}D'$,id)-¿get()-¿row(); if($query == '')returnarray('message' =>' ERROR'$);elsecurrent$_{d}ate = strtotime(date('Y-$

$m-dH:m:s'$)); //$assignthecurrentdateandtimetothisvariable$date = DateTime::createFromFormat('d-M-y h.i.s.u A', $query->LAST_LOGIN$);

$last_login$ = $strtotime$(date-¿format('Y-m-d H:m:s'));

$difference$ = abs(current$_date$−last$_login$)/(60 ∗ 60); //$inhours$

if($difference$ > 1.000000000000000)//$thetokenwillbeunusableafter1hourreturnfalse$;$elsereturntrue$;

public function get$_faculty$(fid = null, id,token) if(!($this->check_staff_authorization$(id, $token$)))$returnarray('st$ ¿check$_token_availability$(id)) if(empty(fid))this-¿db-¿select("FACULTY$_ID, CONCAT(CONCAT(CONCAT(CON$ ¿db-¿from('FACULTY');

$query$ =this-¿db-¿get(); return $query->result$();

else $query$ =this-¿db-¿get$_where('FACULTY', array('FACULTY_ID'$ =>fid)); return $query->result$(); $elsereturnarray('status'$ => 402,'$message'$ =>'$TokenhasExpired'$);

public function get$_staff$(sid = null, id,token) if(!($this->check_staff_authorization$(id, $token$)))$returnarray('sta$ ¿check$_token_availability$(id)) if(empty(sid))query = $this->db->get('STAFF')$; $return$query-¿result(); else $query$ =this-¿db-¿get$_where('STAFF', array('STAFF_ID'$ =>sid)); return $query->result$(); $elsereturnarray('status'$ => 402,'$message'$ =>'$TokenhasExpired'$);

public function get$_room$(rid = null, id,token) if(!($this->check_staff_authorization$(id, $token$)))$returnarray('stat$ ¿check$_token_availability$(id)) if(empty(rid))this-¿db-¿select('ROOM.ROOM$_ID, ROOM.ROOM_NAME, BUILDIN$ ¿db-¿from('ROOM'); $this->db->join('BUILDING','ROOM.BUILDING_ID = BUILDING.BUILDING_I$ = $this->db->get()$; $return$query-¿result();

else $query$ =this-¿db-¿get$_where('ROOM', array('ROOM_ID'$ =>rid)); return $query->result()$; $elsereturnarra$

public function get$_faculty_attendance$(id, $token$)if(!(this-¿check$_staff_authorization$(id, $token$)))$returnarray('stat$ = $this->get_checker_route$(id);

if($this->check_token_availability$(id)) $this->db->select("FACULTY_ATTENDANCE.FACULTY_ATTEN$ $mi:ssAM')ASFIRST_CHECK, TO_CHAR(CAST(FACULTY_ATTENDANCE.SECOND_CHECKASDATE),'$ $mi:ssAM')ASSECOND_CHECK, REPLACE(FACULTY_ATTENDANCE.FIRST_IMAGE_FILE,'assets/ima$ ¿db-¿from('FACULTY$_ATTENDANCE'$);this-¿db-¿join('CLASS$_SCHEDULE','FACULTY_ATTENDANCE.CL$ $CLASS_SCHEDULE.CLASS_SCHEDULE_ID','inner'$);this-¿db-¿join('CONFIRMATION$_NOTICEC','FACULT$ $C.CONFIRMATION_NOTICE_ID','left'$);this-¿db-¿join('CHANGE$_VENUE_FORM','FACULTY_ATTENDANC$ $CHANGE_VENUE_FORM.CHANGE_VENUE_ID','left'$);

$where = array('FACULTY_ATTENDANCE.ATTENDANCE_DATE' => strtoupper(date('d-M-y')),'FACULTY_ATTENDANCE.VALIDATEDISNULL' => null,//'FACULTY_ATTENDANCE.STAT$

null,);

$or_{w}here = array('FACULTY_{A}TTENDANCE.VALIDATED' => 0,'FACULTY_{A}TTENDANCE.STATUS$

null);

this− > *db*− > *where*(where); //*this*− > *db*− > *where*('$FACULTY_{A}TTENDANCE.STATUSISNULL'$, *null*

¿db-¿where('FACULTY$_{A}TTENDANCE.VALIDATEDISNULL'$, *null*); //this-¿db-¿or$_{w}here$(or$_{w}here$);

query =this-¿db-¿get();

return *query*− > *result*(); *elsereturnarray*('*status*' => 402,'*message*' =>' *TokenhasExpired*');

public function get$_{c}hecker_{r}oute$(staff$_{i}d$)this-¿db-¿select('CHECKER.ROUTE'); *this*− > *db*− > *from*('*STAFF*');this-¿db-¿join('CHECKER', 'STAFF.CHECKER$_{I}D = CHECKER.CHECKER_{I}D$','*inner*');thi

¿db-¿where('STAFF$_{I}D$',staff$_{i}d$);

query =this-¿db-¿get()-¿row();

return *query*− > *ROUTE*;

public function post$_{f}aculty_{a}ttendance$(faid, *id*,token, *staff*$_{i}d$,class$_{s}chedule_{i}d$,attendance$_{d}ate$,nid, *first*$_{c}heck$,second$_{c}heck$,fimage$_{f}ile$,simage$_{f}ile$,status, *validated*,cnid) //Error handling not implemented

if(!(*this*− > *check*$_{s}taff_{a}uthorization$(id, *token*)))*returnarray*('*status*' => 401,'*message*' =>' *UnauthorizedA*

array('$STAFF_{I}D$' => *null*,'$CLASS_{S}CHEDULE_{I}D$' => *null*,'$CONFIRMATION_{N}OTICE_{I}D$' =>cnid,

'ATTENDANCE$_{D}ATE$' =>attendance$_{d}ate$,'$NOTE_{I}D$' =>nid, 'FIRST$_{C}HECK$' =>first$_{c}heck$,'$SECOND_{C}HEC$

"*No*",'*STATUS*' =>status, 'NOTIFIED' =¿ 0, 'VALIDATED' =¿ *validated*);

this− > *db*− > *set*(first$_{d}ata$);this-¿db-¿where('FACULTY$_{A}TTENDANCE_{I}D$',faid); *first*$_{q}uery$ =this-¿db-¿update('FACULTY$_{A}TTENDANCE$');

this− > *db*− > *set*('$STAFF_{I}D$',staff$_{i}d$);this-¿db-¿set('CLASS$_{S}CHEDULE_{I}D$',class$_{s}chedule_{i}d$);this-¿db-¿where('FACULTY$_{A}TTENDANCE_{I}D$',faid);

second$_{q}uery$ =this-¿db-¿update('FACULTY$_{A}TTENDANCE$');

if(*status* == "*Absent*"validated == null *cnid* == *null*)third$_{q}uery$ =this-¿create$_{c}onfirmation_{n}otice$(faid);

if(*first*$_{q}uery$second$_{q}uery$)*returnarray*('*status*' => 201,'*message*' =>' *Created*','*Validated*' =>validated);

else return array('status' =¿ 304, 'message' =¿ 'Error: Database not update');

public function get$_{c}lass_{s}chedule$(id, *token*)*if*(!(this-¿check$_{s}taff_{a}uthorization$(id, *token*)))*returnarray*('*status*' =

¿check$_{t}oken_{a}vailability$(id)) *this*− > *db*− > *select*('$CLASS_{S}CHEDULE_{I}D, FACULTY_{I}D, SEMESTER, SCHC$

¿db-¿from('CLASS$_{S}CHEDULE$');

$array = array('SEMESTER' =>' 1','SCHOOL_YEAR' =>' 2019','START_TIME! ='=> null,'END_TIME! ='=> null);this-¿db-¿where(array);query = this- > db- > get(); return query-¿result(); else return array('status' =¿ 402, 'message' =¿ 'Token has Expired');

public function update_token(checker_id,token) this- > db- > set('TOKEN',token); this- > db- > where('CHECKER_ID',checker_id);this-¿db-¿update('CHECKER');

public function update_staff_last_login(id, date)fdate = new DateTime(date);date = fdate- > format('d − F − yh.i.s.uA');

this- > db- > set('LAST_LOGIN',date); this- > db- > where('STAFF_ID',id); this- > db- > update('STAFF');

public function login(id,password) /* This login system will be only used by the mobile application users The login system will be only limited to ADMINS, CHECKERS AND HRMO CHECKER OFFICER */ this- > db- > select('STAFF.STAFF_ID, STAFF.CHECKER_ID, STAFF.PRIVILEGE, CHEC ¿db-¿from('STAFF'); this- > db- > join('CHECKER','CHECKER.CHECKER_ID = STAFF.CHECKER_I ¿db-¿where('STAFF_ID',id)-¿where('PASSWORD', password)- > where('PRIVILEGE! =', 2)- > where('PRIVILEGE! =', 4)- > where('PRIVILEGE! =', 5)- > where('PRIVILEGE! =' , 6)- > where('PRIVILEGE! =', 7);query = this- > db- > get()- > row();

if(query ==='')returnarray('status' => 204,'message' =>' Usernotfound');else

date = date('Y − m − dH : i : s'); //assignthecurrentdateoftheserver, YYYY − MM − DDHH : II : SStoken = md5(rand(0,9).query- > STAFF_ID.password.date); //createauniquehashusingrand(), id, passwor

this- > update_token(query-¿CHECKER_ID,token); this- > update_staff_last_login(query-¿STAFF_ID,date);

return array('status' =¿ 200, 'message' =¿ 'Successfully login', 'staff_id' =>query-¿STAFF_ID,'token' =>token, 'route_id' =>query-¿ROUTE);

public function get_all_count(id, token)if(!(this-¿check_staff_authorization(id, token)))returnarray('status' => 40 ¿check_token_availability(id)) this- > db- > select('ROOM.ROOM_ID, ROOM.ROOM_NAME, BUILDING.BUI ¿db-¿from('ROOM'); this- > db- > join('BUILDING','ROOM.BUILDING_ID = BUILDING.BUILDING_I = this- > db- > get()- > num_rows();

class_schedule =this-¿db-¿get('CLASS_SCHEDULE')- > num_rows();

faculty =this-¿db-¿get('FACULTY')-¿num_rows();

faculty_attendance =this-¿db-¿get('FACULTY_ATTENDANCE')- > num_rows();

return array('CLASS_SCHEDULE' =>class_schedule,'ROOM' =>room, 'FACULTY' =¿ faculty,'FACULTY_A

public function post_confirmation_notice(faid, cnid,date, reason,esignature, remarks,confirmed)

```
//this->db->set('CONFIRMATION_NOTICE_ID',cnid); this->db->set('CONFIRMATION_NOTI
this->db->set('REASON',reason); this->db->set('ELECTRONIC_SIGNATURE',esignature);
this->db->set('CONFIRMED',confirmed); this->db->where('CONFIRMATION_NOTICE_ID',cnid);
query =this-¿db-¿update('CONFIRMATION_NOTICE');
if(query)returnarray('status' => 201,'message' =>' Updated','imageFile' =>esignature); else
return array('status' =¿ 304, 'message' =¿ 'Error: Database not update');
public function create_confirmation_notice(faid) cnid = ""; docnid = this->generate_unique_id('9'); while(!(this-
¿is_unique_id_available('CONFIRMATION_NOTICE_ID',cnid, 'CONFIRMATION_NOTICE')));
data = array('CONFIRMATION_NOTICE_ID' =>cnid, 'CONFIRMATION_NOTICE_DATE' =>
date('d-M-yh.i.s.uA'),'REASON' => "",'ELECTRONIC_SIGNATURE' => "",'REMARKS' =>
"",'CONFIRMED' => 0);
query =this-¿db-¿insert('CONFIRMATION_NOTICE',data);
if(query)this-¿db-¿set('CONFIRMATION_NOTICE_ID',cnid); this->db->where('FACULTY_ATTENDAN
this->db->update('FACULTY_ATTENDANCE');
public function get_confirmation_notice()this-¿db-¿select('*'); this->db->from('CONFIRMATION_NOTI
¿db-¿where('CONFIRMED', 0); query =this-¿db-¿get()-¿result();
return query;
Generate Faculty Attendance
// generate_faculty_attedancecanbemanuallytriggeredortimetriggered(ifpossible)publicfunctiongenerate_faculty
¿get_class_schedule_ids("1","2019"); //changeargumentsthisifnecessary
foreach(class_schedule_idas id) if(this->class_day(id['CLASS_DAY'])if(!this-¿is_faculty_attendance_created(id['CL
M-y')))dofaculty_id =this-¿api_model->generate_unique_id('9'); while(!this-¿is_unique_id_available('FACULTY_ATTE
if(this->is_venue_changed(id['CLASS_SCHEDULE_ID']))change_venue_id =this-¿get_change_venue_id(id['CLASS_S
= array( 'FACULTY_ATTENDANCE_ID' =>faculty_id,'STAFF_ID' =>'','CLASS_SCHEDULE_ID' =>id['CLASS
,'CHANGE_VENUE_ID' =>change_venue_id,'ATTENDANCE_DATE' => date('d-M-y'),'FIRST_CHECK' =>
,'SECOND_CHECK' =>'','FIRST_IMAGE_FILE' =>'','SECOND_IMAGE_FILE' =>'','SALARY_DEDUCTIO
,'STATUS' =>'','NOTIFIED' =>' 0','VALIDATED' => null);query1 = this->db->
insert('FACULTY_ATTENDANCE',data); else data = array('FACULTY_ATTENDANCE_ID' =>faculty_id,'ST
,'CLASS_SCHEDULE_ID' =>id['CLASS_SCHEDULE_ID'],'CONFIRMATION_NOTICE_ID' =>''
,'CHANGE_VENUE_ID' =>'','ATTENDANCE_DATE' => date('d-M-y'),'FIRST_CHECK' =>''
,'SECOND_CHECK' =>'','FIRST_IMAGE_FILE' =>'','SECOND_IMAGE_FILE' =>'','SALARY_DEDUCTIO
```

,'$STATUS'$ =>'',' $NOTIFIED'$ =>' 0',' $VALIDATED'$ => $null$);query2 = $this->db->$ $insert('FACULTY_ATTENDANCE'$,data); return true;

public function get$_c lass_s chedule_i ds$(semester, $school_y ear$)arrReturn = array(); $this->db->$ $select('CLASS_SCHEDULE_ID, CLASS_DAY'$);this-¿db-¿from('CLASS$_S CHEDULE'$);this-¿db-¿where('SEMESTE $semester$);this-¿db-¿where('SCHOOL$_Y EAR'$,school$_y ear$);this-¿db-¿where('START$_T IME$! =', $null$);this-¿db-¿where('END$_T IME$! =', $null$);

$query$ =this-¿db-¿get();

$result$ =query-¿result$_a rray()$; if(!$empty$(result)) $arrReturn$ =result; return $arrReturn$;

public function is$_v enue_c hanged$(class$_s chedule_i d$)this-¿db-¿select('CHANGE$_V ENUE_I D'$);this-¿db-¿from('CHANGE$_V ENUE_F ORM'$);

$array = array('CLASS_SCHEDULE_ID'$ =>class$_s chedule_i d$,' $CONFIRMED'$ => 1,' $DATE_O F_C HANGE'$ = $date('d - M - y'))$;

$this->db->where$(array); $query$ =this-¿db-¿get()-¿row();

if($query == null) return false; else return true;

public function get$_c hange_v enue_i d$(class$_s chedule_i d$)this-¿db-¿select('CHANGE$_V ENUE_I D'$);this-¿db-¿from('CHANGE$_V ENUE_F ORM'$);

$array = array('CLASS_SCHEDULE_ID'$ =>class$_s chedule_i d$);this-¿db-¿where($array$);

$query$ =this-¿db-¿get()-¿row();

return $query->CHANGE_V ENUE_I D$;

public function is$_f aculty_a ttendance_c reated$(class$_s chedule_i d$,attendance$_d ate$)this-¿db-¿select('FACULTY$_A TTEN$ ¿db-¿from('FACULTY$_A TTENDANCE'$);this-¿db-¿where('CLASS$_S CHEDULE_I D'$,class$_s chedule_i d$);this-¿db-¿where('ATTENDANCE$_D ATE'$,attendance$_d ate$);

$query$ =this-¿db-¿get()-¿row();

if($query == null) return false; else return true;

public function class$_d ay$(class$_d ay$)//$will return true/ * possible class_d ay values MWF = if the days are not straight$ = '-'; $day1 = array('SU','M','T','W','TH','F','SA'$);day2 = array('SU' =¿ 'Sunday', 'M' =¿ 'Monday', 'T' =¿ 'Tuesday', 'W' =¿ 'Wednesday', 'TH' =¿ 'Thursday', 'F' =¿ 'Friday', 'SA' =¿ 'Saturday'); $day3 = array('Sunday'$ =>' SU',' $Monday'$ =>' M',' $Tuesday'$ =>' T',' $Wednesday'$ =>' W',' $Thursday'$ =>' TH',' $Friday'$ =>' F',' $Saturday'$ =>' SA');

if(strpos($class_d ay$,dash) !== false) return $this->check_d ay_d ashed$(class$_d ay$); $else if$(this-¿check$_d ay_t hree_l etters$(c ¿check$_d ay_a bbr$(class$_d ay$)) $return true$; $else return false$;

public function check$_day_three$_letters$(class_day)$

if($class_day == strtoupper(date('D')))returntrue;elsereturnfalse;$

public function check$_day_abbr$(class$_day$)day1 = array('SU' =¿ 'SUN', 'M' =¿ 'MON', 'T' =¿ 'TUE', 'W' =¿ 'WED', 'TH' =¿ 'THU', 'F' =¿ 'FRI', 'SA' =¿ 'SAT'); $day2 = array('SU','M','T','W','TH','F','SA$ = count($day2) - 1;i ¿= 0; $i--)if(strpos($class$_day$,day2[i])! == $false)if$(day1[$day2$[i]] == strtoupper(date('D'))) return true; return false;

public function check$_day_ashed$(class$_day$)class$_days = array()$;range; $day1 = array('SU','M','T','W','TH','F',$ = array('SU' =¿ 'SUN', 'M' =¿ 'MON', 'T' =¿ 'TUE', 'W' =¿ 'WED', 'TH' =¿ 'THU', 'F' =¿ 'FRI', 'SA' =¿ 'SAT');

for($i = count$(day1) -1; $i >= 0$;i–) if(strpos($class_day$,day1[i])! == $false)array_push$(class$_days$,day1[i]);

for($i = array_search$(class$_days$[0],day1); $i >= array_search$(class$_days$[1],day1); $i--)if$(day2[$day1$[i]] == strtoupper(date('D'))) return true; return false;

public function test() $this- > db- > select('FACULTY_ID')$;this-¿db-¿from('FACULTY'); $this- > db- > or_like('DESIGNATION','Faculty','both')$;this-¿db-¿or$_like('DESIGNATION','Dean','both')$;this-¿db-¿or$_like('DESIGNATION','Chairperson','both')$;ids = $this- > db- > get()- > result_array()$;count = 0;

foreach ($idsas$id) $count+ = 1$;

return $count; ConsumingAPIfunctiongetAllFacultyAPI()$key = file$_get_contents('C :_key.txt')$;headers = array('Content-Type: application/json',);

$url =$ "$https://services.adnu.edu.ph/api/v2/employee/college_faculty_info/X-API-KEY/$".key; $ch = curl_init()$;

curl$_setopt$(ch, CURLOPT$_URL$,url); curl$_setopt$(ch, CURLOPT$_POST, false)$; $curl_setopt$(ch, CURLOPT$_HTTPHE$ curl$_setopt$(ch, CURLOPT$_RETURNTRANSFER, true)$;

$result = curl_exec$(ch); curl$_close$(ch);

$items = (array)json_decode$(result);

return $items$;

function getFacultyAPI($employee_id$)key = file$_get_contents('C :_key.txt')$;headers = array('Content-Type: application/json',);

$url =$ "$https://services.adnu.edu.ph/api/v2/employee/college_faculty_info/emp_id/$".employee$_id$."$/X-API-KEY/$".key; $ch = curl_init()$;

curl$_setopt$(ch, CURLOPT$_URL$,url); curl$_setopt$(ch, CURLOPT$_POST, false)$; $curl_setopt$(ch, CURLOPT$_HTTPHE$

curl$_s$*etopt*(ch, CURLOPT$_R$*ETURNTRANSFER*, *true*);

result = *curl$_e$xec*(ch); curl$_c$*lose*(ch);

items = (*array*)*json$_d$ecode*(result);

return *items*;

function insertFacultyData()

items =this-¿getAllFacultyAPI();

foreach(*items*[′*data*′]*as*item) if(preg$_m$*atch*("/*Faculty*/",item-¿designation) —— preg$_m$*atch*("/*Dean*/",item-¿designation) —— preg$_m$*atch*(′/*Chairperson*/′,item-¿designation))

designation =item-¿designation; //*college* =this-¿getCollege(*item*− > *department*);college = *this*− > *getCollege*(item-¿department);

if(!preg$_m$*atch*("/*CollegeofNursing*/",item-¿department)) if(preg$_m$*atch*("/*Collegeof*/",item-¿department)) *college* = *str$_r$eplace*("−*Dean*′*sOffice*","",item-¿department); else *college* = "*CollegeofNursing*";

if(preg$_m$*atch*(′/*Dean*/′,designation)) *designation* =′ *Dean*′;

/*if(preg$_m$*atch*("/*CollegeFaculty*/",designation)) *designation* = "*Instructor*";∗/

data = *array*(′*FACULTY$_I$D*′ =>item-¿employee$_i$*d*,′ *FIRST$_N$AME*′ =>item-¿first$_n$*ame*,′ *MIDDLE$_N$AME*′ = ¿middle$_n$*ame*,′ *LAST$_N$AME*′ =>item-¿surname, 'DESIGNATION' =¿ *designation*,′ *EMAIL*′ =>item-¿email, 'CONTACT$_N$*UMBER*′ =>item-¿contact$_n$*umber*,′ *DEPARTMENT*′ =>item-¿department, 'COLLEGE' =¿ *college*); *if*(this-¿is$_u$*nique$_i$d$_a$vailable*(′*FACULTY$_I$D*′,item-¿employee$_i$*d*,′ *FACULTY*′))this-¿db-¿insert('FACULTY', *data*); *else*this-¿db-¿set(*data*);this-¿db-¿where('FACULTY$_I$*D*′,item-¿employee$_i$*d*);this-¿db-¿update('FACULTY');

return true;

function getCollege(*department*)cba = array("Accountancy Department", "Business Management Courses Department", "Allied Business Courses Department"); *chss* = *array*("*SocialSciencesDepartment*","*T* = array("Secondary and Elementary Education Department", "Physical Education Department"); *cse* = *array*("*CivilEngineeringDepartment*","*ECEandCpEDepartment*","*NaturalSciencesDepartment*","*Mat* = array("Computer Science Department", "Digital Arts and Computer Animation Department");

foreach (*cbaas*dept) if(*dept* ==department) return "College of Business and Accountancy"; foreach (*chssas*dept) if(*dept* ==department) return "College of Humanities and Social Sciences"; foreach (*coeas*dept) if(*dept* ==department) return "College of Education"; foreach (*cseas*dept) if(*dept* ==department) return "College of Science and Engineering"; foreach (*ccsas*dept) if(*dept* ==department) return "College of Computer Studies"; if(*department* ==′ *CollegeofNursing*′)*return*"*CollegeofNursing*";

function getEmployeeID() $items$ =this-¿getFacultyAPI('200101962');

if($items['code'] == 0)return$items['data'][0]-¿first$_name$;

function getClassSchedule($faculty_id$)key = file$_get_contents('C :_k ey.txt')$;headers = array('Content-Type: application/json',);

$url = "https://services.adnu.edu.ph/api/v2/employee/college_faculty_schedule/syear/2019/term/1/emp_id/".f$
$API - KEY/$".key; $ch = curl_init()$;

curl$_setopt$(ch, CURLOPT$_URL$,url); curl$_setopt$(ch, CURLOPT$_POST, false)$; $curl_setopt$(ch, CURLOPT$_HTTPHE$
curl$_setopt$(ch, CURLOPT$_RETURNTRANSFER, true)$;

$result = curl_exec$(ch); curl$_close$(ch); $items = (array)json_decode$(result);

return $items$;

function syncClassSchedule() $this- > db- > select('FACULTY_ID')$;this-¿db-¿from('FACULTY');
$this- > db- > or_like('DESIGNATION',' Faculty',' both')$;this-¿db-¿or$_like('DESIGNATION',' Dean',' both')$;th
¿db-¿or$_like('DESIGNATION',' Chairperson',' both')$;ids = $this- > db- > get()- > result_array()$; $foreach$(ids as id)class = $this- > getClassSchedule$(id['FACULTY$_ID'])$; if(class['code'] == 0) foreach ($class['data']as$data) $class_schedule_id = 0; if$(this-¿isScheduleUnique($data- > SUBJ_CODE$,data-¿SECTION, $data- > EMP_ID)$)doclass$_schedule_id$ =this-¿api$_model- > generate_unique_id('9'); while$(!this-¿is$_unique_id_available('CLASS_S$
= str$_replace('NN',' PM'$,data-¿TIMEFROM); $timeto = str_replace('NN',' PM'$,data-¿TIMETO);
$hour$ =this-¿timeDifferenceByHour($timefrom$,timeto); $array = array('CLASS_SCHEDULE_ID' =>$class$_schedule_i$
¿ROOM, 'FACULTY$_ID' =>$data-¿EMP$_ID,' SEMESTER' =>$data-¿SEM, 'SCHOOL$_YEAR' =>$data-¿SYEAR, 'START$_TIME' =>$timefrom, 'END$_TIME' =>$timeto, 'CLASS$_SECTION' =>$data-¿SECTION, 'CLASS$_DAY' =>$data-¿DAY, 'SUBJECT$_CODE' =>$data-¿SUBJ$_CODE,' HOURS' =>$hour
); $query$ =this-¿db-¿insert('CLASS$_SCHEDULE'$,array); return true;

function insertAllRoom() $this- > db- > select('FACULTY_ID')$;this-¿db-¿from('FACULTY');
$faculty_ids$ =this-¿db-¿get()-¿result$_array()$; $foreach$(faculty$_ids$asid) $class_schedules$ =this-¿getClassSchedule($id['FA$
2)$foreach$(class$_schedules['data']as$class) if($class- > ROOM! = null$)data = array('ROOM$_ID' =>$class-¿ROOM, 'BUILDING$_ID' => null,' ROOM_NAME' =>$class-¿ROOM); if($this- > is_unique_id_available('ROOM_ID'$
¿ROOM, 'ROOM')) $this- > db- > insert('ROOM'$,data); return true;

function timeDifferenceByHour($stime$,etime) return number$_format(abs($strtotime(etime) - strtotime($stime$))/60/

function getNote() /*Authentication is not required on this API*/ $this- > db- > select('*')$;this-¿db-¿from('NOTE'); $query$ =this-¿db-¿get()-¿result$_array()$; $return$query; ?¿

A.3.2 FacultyAttendanceModel.php

[language=PHP] ¡?php

class FacultyAttendanceModel extends CI$_{M}odelfunctionfetchFacultyAttendanceData$()this-¿db-¿select('ATTENDANCE$_{D}ATE, FIRST_{C}HECK, SECOND_{C}HECK, LAST_{N}AME, FIRST_{N}AME, substr(MID$¿db-¿from('FACULTY$_{A}TTENDANCEA'$);this-¿db-¿join('CONFIRMATION$_{N}OTICEN', N.CONFIRMATION$A.CONFIRMATION$_{N}OTICE_{I}D'$);this-¿db-¿join('CLASS$_{S}CHEDULEC', C.CLASS_{S}CHEDULE_{I}D =$A.CLASS$_{S}CHEDULE_{I}D'$);this-¿db-¿join('FACULTY F', 'F.FACULTY$_{I}D = C.FACULTY_{I}D'$);

$this- > db- > order_{b}y("START_{T}IME","DESC");$

return $this- > db- > get();$

function fetchFacultyListData() $this- > db- > select('FACULTY_{I}D, LAST_{N}AME, FIRST_{N}AME, substr(M$¿db-¿get('FACULTY');

function fetchClassScheduleData() $this- > db- > select('SUBJECT_{C}ODE, ROOM_{I}D, START_{T}IME, END_{T}$¿db-¿from('CLASS$_{S}CHEDULE'$);this-¿db-¿join(' FACULTY','FACULTY.FACULTY$_{I}D = CLASS_{S}CHEDULE.I$¿db-¿get();

function fetchSummaryReportData() $this- > db- > select('FIRST_{N}AME, LAST_{N}AME, COLLEGE, DEPA$¿db-¿from('FACULTY$_{A}TTENDANCEA'$);this-¿db-¿join('CLASS$_{S}CHEDULEC', A.CLASS_{S}CHEDULE_{I}D =$C.CLASS$_{S}CHEDULE_{I}D'$);this-¿db-¿join('FACULTY F', 'C.FACULTY$_{I}D = F.FACULTY_{I}D'$);this-¿db-¿join('CONFIRMATION$_{N}OTICEN', A.CONFIRMATION_{N}OTICE_{I}D = N.CONFIRMATION_{N}OTIC$¿db-¿where('STATUS', 'Absent'); $this- > db- > where('N.CONFIRMED', 1)$;this-¿db-¿where('VALIDATED !=', 0);

return $this- > db- > get();$

function fetchRegisteredAccData() $this- > db- > select("S.STAFF_{I}D, CONCAT(CONCAT(CONCAT(CO$¿db-¿from('STAFF S'); $this- > db- > join('CHECKERC', S.CHECKER_{I}D = C.CHECKER_{I}D', left');$

return $this- > db- > get();$

function fetchNoteData() $this- > db- > select('ATTENDANCE_{D}ATE, FIRST_{N}AME, LAST_{N}AME, SUBS$¿db-¿from('FACULTY$_{A}TTENDANCEA'$);this-¿db-¿join('NOTE N', 'A.NOTE$_{I}D = N.NOTE_{I}D'$);

return $this- > db- > get();$

function getTime() $this- > db- > select('START_{T}IME, END_{T}IME, CLASS_{D}AY, SEMESTER, SCHOOL$¿db-¿get('CLASS$_{S}CHEDULE'$)- > result_{a}rray();$

function getClassDay() $this- > db- > select('CLASS_{D}AY'$);this-¿db-¿distinct(); $this- > db- > order_{b}y('CLASS_{D}AY', ASC'); returnthis$-¿db-¿get('CLASS$_{S}CHEDULE'$)- > result_{a}rray();$

function getClassTime() $this->db->select('START_TIME, END_TIME');$this-¿db-¿distinct(); $this->db->order_by('START_TIME','DESC'); return$this-¿db-¿get('CLASS_SCHEDULE')-> $result_array();$

function getSemester() $this->db->select('SEMESTER');$this-¿db-¿distinct(); $this->db->order_by('SEMESTER','ASC'); return$this-¿db-¿get('CLASS_SCHEDULE')-> $result_array();$

function getDailySchedule() $query =$this-¿db-¿get('CLASS_SCHEDULE'); return$query-¿result_array();

function getClassSchedule()

$this->db->select('SUBJECT_CODE, ROOM_ID, START_TIME, END_TIME, CLASS_DAY, CLASS_SECT$ ¿db-¿from('CLASS_SCHEDULE');this-¿db-¿join(' FACULTY','FACULTY.FACULTY_ID = CLASS_SCHEDULE.I $= this->db->get();$

return $query->result_array();$

function getClassScheduleFaculty($facultyid$)this-¿db-¿select('CLASS_SCHEDULE_ID, SUBJECT_CODE, CLAS ¿db-¿from('CLASS_SCHEDULE');this-¿db-¿join(' FACULTY','FACULTY.FACULTY_ID = CLASS_SCHEDULE.I ¿db-¿where('FACULTY.FACULTY_ID',facultyid);

$query =$this-¿db-¿get(); return $query->result_array();$

function getAllClassSchedule() $this->db->select('CLASS_SCHEDULE_ID, SUBJECT_CODE, CLASS_SEC$ ¿db-¿from('CLASS_SCHEDULE');this-¿db-¿join(' FACULTY','FACULTY.FACULTY_ID = CLASS_SCHEDULE.I ¿db-¿order_by('SUBJECT_CODE','ASC');this-¿db-¿order_by('CLASS_SECTION','ASC');

$query =$this-¿db-¿get(); if(!empty($query$))if(query-¿result_array() == 1)$return$query-¿result_array(); $else return$"

function getRoutes() $this->db->select('DISTINCT(ROUTE) AS ROUTE');$this-¿db-¿from('BUILDING');

$query =$this-¿db-¿get(); if(!empty($query$))if(query-¿result_array() == 1)$return$query-¿result_array(); $else return$"

function getRooms() $this->db->from('ROOM');$this-¿db-¿join('BUILDING', 'ROOM.BUILDING_ID = $BUILDING.BUILDING_ID');$this-¿db-¿order_by("$ROOM_NAME$","ASC");query = $this->db->get(); return$query-¿result_array();

function getBuildings() $this->db->order_by('BUILDING_NAME');$query = $this->db->get('BUILDING'); return$query-¿result_array();

//Use to get email for Absence Notification function getEmailDetails(id)

$this->db->select('EMAIL, ATTENDANCE_DATE');$this-¿db-¿from('FACULTY'); $this->db->join('CLASS_SCHEDULE','CLASS_SCHEDULE.FACULTY_ID = FACULTY.FACULTY_ID');$this-¿db-¿join('FACULTY_ATTENDANCE',' FACULTY_ATTENDANCE.CLASS_SCHEDULE_ID = $CLASS_SCHEDULE.CLASS_SCHEDULE_ID');$this-¿db-¿where('FACULTY_ATTENDANCE_ID',id);

$this->db->where('STATUS','Absent')$;this-¿db-¿where('NOTIFIED', 0); return $this->db->get()$;

function getChangeVenue($facultyid$)this-¿db-¿select('S.SUBJECT$_{C}ODE$, $S.CLASS_{S}ECTION$, $S.CLASS_{D}AY$,
¿db-¿from('CHANGE$_{V}ENUE_{F}ORMV'$);this-¿db-¿join('CLASS$_{S}CHEDULES'$,'$V.CLASS_{S}CHEDULE_{I}D = S.CLASS_{S}CHEDULE_{I}D'$,'$inner'$);this-¿db-¿join('FACULTY F', 'S.FACULTY$_{I}D = F.FACULTY_{I}D'$,'$inner'$);wh
= array('F.FACULTY$_{I}D'$ =>facultyid, 'DATE$_{O}F_{C}HANGE >='$=> $date('d-M-y'))$;this-¿db-
¿where($where$);this-¿db-¿order$_{b}y('V.DATE_{F}ILED'$,'$DESC'$);query = $this->db->get(); if($!$empty(query))
if($query->result_{a}rray() == 1)return$query-¿result$_{a}rray(); elsereturn$"0"; $elsereturn$"0";

function getAllChangeVenue($limit$,start) $this->db->select("CONCAT(CONCAT(CONCAT(CONCAT(I$
¿db-¿from('CHANGE$_{V}ENUE_{F}ORMV'$);this-¿db-¿join('CLASS$_{S}CHEDULES'$,'$V.CLASS_{S}CHEDULE_{I}D = S.CLASS_{S}CHEDULE_{I}D'$,'$inner'$);this-¿db-¿join('FACULTY F', 'S.FACULTY$_{I}D = F.FACULTY_{I}D'$,'$inner'$);th
¿db-¿order$_{b}y('V.CONFIRMED'$,'$DESC'$);this-¿db-¿order$_{b}y('V.DATE_{F}ILED'$,'$DESC'$);this-¿db-
¿where('DATE$_{O}F_{C}HANGE >='$, $date('d-M-y'))$;this-¿db-¿limit($limit$,start); $query$ =this-¿db-
¿get(); if(!empty($query$))if(query-¿result$_{a}rray()$ == 1)$return$query-¿result$_{a}rray(); elsereturn$"0"; $elsereturn$"0";

function getDeptChangeVenueRequest($department$)this-¿db-¿select("V.CHANGE$_{V}ENUE_{I}D, CONCAT(CON$
¿db-¿from('CHANGE$_{V}ENUE_{F}ORMV'$);this-¿db-¿join('CLASS$_{S}CHEDULES'$,'$V.CLASS_{S}CHEDULE_{I}D = S.CLASS_{S}CHEDULE_{I}D'$,'$inner'$);this-¿db-¿join('FACULTY F', 'S.FACULTY$_{I}D = F.FACULTY_{I}D'$,'$inner'$);wh
= array('F.DEPARTMENT' =¿ $department$,'$V.CONFIRMED'$ => $null$);this-¿db-¿where($where$);

$query$ =this-¿db-¿get(); if(!empty($query$))if(query-¿result$_{a}rray()$ == 1)$return$query-¿result$_{a}rray(); elsereturn$"

function getDeptChangeVenueRequestCount($department$)this-¿db-¿select("V.CHANGE$_{V}ENUE_{I}D, CONCAT($
¿db-¿from('CHANGE$_{V}ENUE_{F}ORMV'$);this-¿db-¿join('CLASS$_{S}CHEDULES'$,'$V.CLASS_{S}CHEDULE_{I}D = S.CLASS_{S}CHEDULE_{I}D'$,'$inner'$);this-¿db-¿join('FACULTY F', 'S.FACULTY$_{I}D = F.FACULTY_{I}D'$,'$inner'$);wh
= array('F.DEPARTMENT' =¿ $department$,'$V.CONFIRMED'$ => $null$);this-¿db-¿where($where$);query
= $this->db->get(); if($!$empty(query)) if($query->num_{r}ows == 1)return$query-¿num$_{r}ows$; $elsereturn$"0"; $else$

function getAllChangeVenueRequest() $this->db->select("V.CHANGE_{V}ENUE_{I}D, CONCAT(CONCAT($
¿db-¿from('CHANGE$_{V}ENUE_{F}ORMV'$);this-¿db-¿join('CLASS$_{S}CHEDULES'$,'$V.CLASS_{S}CHEDULE_{I}D = S.CLASS_{S}CHEDULE_{I}D'$,'$inner'$);this-¿db-¿join('FACULTY F', 'S.FACULTY$_{I}D = F.FACULTY_{I}D'$,'$inner'$);wh
= array('V.CONFIRMED' =¿ null, 'V.DATE$_{O}F_{C}HANGE >='$=> $date('d-M-y'))$;this-¿db-
¿where($where$);query = $this->db->get(); if($!$empty(query)) if($query->result_{a}rray() ==
1)$return$query-¿result$_{a}rray(); elsereturn$"0"; $elsereturn$"0";

function getAllPendingConfirmation($limit$,start) $this->db->select("CONCAT(CONCAT(CONCAT(CON$

¿db-¿from('CONFIRMATION$_NOTICEN'$);this-¿db-¿join('FACULTY$_ATTENDANCEA'$,'$ $N.CONFIRMATIO$
$A.CONFIRMATION_NOTICE_ID'$,'$ $inner'$);this-¿db-¿join('CLASS$_SCHEDULES'$,'$ $A.CLASS_SCHEDULE_ID$ =
$S.CLASS_SCHEDULE_ID'$,'$ $inner'$);this-¿db-¿join('FACULTY F', 'S.FACULTY$_ID = F.FACULTY_ID'$);this-
¿db-¿where('N.CONFIRMED', 0); $this- > db- > order_by('N.CONFIRMATION_NOTICE_DATE'$,'$ $DESC'$);this-
¿db-¿limit($limit$,start); $query$ =this-¿db-¿get(); if(!empty($query$))if(query-¿result$_array() > 1)return$query-
¿result$_array()$; $elsereturn$"0"; $elsereturn$"0";

function getAllPendingConfirmationCount() $this- > db- > select$("$CONCAT(CONCAT(CONCAT(CONCA$
¿db-¿from('CONFIRMATION$_NOTICEN'$);this-¿db-¿join('FACULTY$_ATTENDANCEA'$,'$ $N.CONFIRMATIO$
$A.CONFIRMATION_NOTICE_ID'$,'$ $inner'$);this-¿db-¿join('CLASS$_SCHEDULES'$,'$ $A.CLASS_SCHEDULE_ID$ =
$S.CLASS_SCHEDULE_ID'$,'$ $inner'$);this-¿db-¿join('FACULTY F', 'S.FACULTY$_ID = F.FACULTY_ID'$);this-
¿db-¿where('N.CONFIRMED', 0);

$this- > db- > order_by('N.CONFIRMATION_NOTICE_DATE'$,'$ $DESC'$);

$query$ =this-¿db-¿get(); if(!empty($query$))if(query-¿num$_rows > 1)return$query-¿num$_rows$; $elsereturn$"0"; $elsere$

function getRoomsPagination($search$,limit, $start$)this-¿db-¿from('ROOM'); $this- > db- > join('BUILDING'$,'
$BUILDING.BUILDING_ID'$);this-¿db-¿order$_by$("$ROOM_NAME$","ASC"); if(search != null) $this- >$
$db- > like('LOWER(ROOM.ROOM_ID)'$,search, 'both'); $this- > db- > or_like('LOWER(ROOM.ROOM_NAM$E
'both'); $this- > db- > or_like('LOWER(BUILDING.BUILDING_NAME)'$,search, 'both'); $this- >$
$db- > limit$(limit, $start$);query = $this- > db- > get()$; $return$query-¿result$_array()$;

function getBuildingsPagination($search$,limit, $start$)this-¿db-¿order$_by('BUILDING_NAME'$); if(search
!= null) $this- > db- > like('LOWER(BUILDING_NAME)'$,search, 'both'); $this- > db- >$
$or_like('LOWER(ROUTE)'$,search, 'both'); $this- > db- > limit$(limit, $start$);query = $this- >$
$db- > get('BUILDING'); return$query-¿result$_array()$;

function getAllHRAccountsCount() $this- > db- > select('*'$);this-¿db-¿from('STAFF'); $query$ =this-
¿db-¿get(); if(!empty($query$))if(query-¿num$_rows == 1)return$query-¿num$_rows$; $elsereturn$"0"; $elsereturn$"0";

function searchFacultyName($search$)search = strtolower($search$);this-¿db-¿select('*'); $this- >$
$db- > from('FACULTY'$);

if($search$! = $NULL$)this-¿db-¿like('LOWER(FACULTY$_ID)'$,search, 'both'); $this- > db- >$
$or_like('LOWER(FIRST_NAME)'$,search, 'both'); $this- > db- > or_like('LOWER(LAST_NAME)'$,search,
'both'); $this- > db- > or_like('LOWER(MIDDLE_NAME)'$,search, 'both'); $this- > db- >$
$order_by('LAST_NAME'$,'$ ASC'); $return$this-¿db-¿get()-¿result$_array()$;

function searchClassSchedule($facultyID$)this-¿db-¿select('*'); $this- > db- > from('CLASS_SCHEDULE'$);thi

¿db-¿where('CLASS$_S CHEDULE.FACULTY_I D'$,facultyID);

return $this->db->get()->result_array()$;

function addBuilding($data$)this-¿db-¿insert('BUILDING', $data$);

function addRoom($data$)this-¿db-¿insert('ROOM', $data$);

function AddChangeVenueToFacultyAttendance() $this->db->select('CHANGE_V ENUE_I D, CLASS_S CHE$
¿db-¿from('CHANGE$_V ENUE_F ORM'$);where = array('DATE$_O F_C HANGE'$ => $date('d - M -$
$y'),'CONFIRMED'$ => 1);this-¿db-¿where($where$);

$changeVenues$ =this-¿db-¿get()-¿result$_a rray()$;

foreach ($changeVenuesas$changeVenue) $this->db->set('CHANGE_V ENUE_I D'$,changeVenue['CHANGE$_V E$
= array('CLASS$_S CHEDULE_I D'$ =>changeVenue['CLASS$_S CHEDULE_I D'$],$'ATTENDANCE_D ATE'$ =>
$date('d - M - y'))$;this-¿db-¿where($where$);this-¿db-¿update('FACULTY$_A TTENDANCE'$);

function updateNotif(id)

$this->db->set('NOTIFIED', 1)$;this-¿db-¿where('FACULTY$_A TTENDANCE_I D'$,id); $this->$
$db->update('FACULTY_A TTENDANCE')$;

function editBuilding($buildingId$,buildingName, $buildingRoute$)array = array('BUILDING$_N AME'$ =>buildingN
'ROUTE' =¿ $buildingRoute$);this-¿db-¿set($array$);this-¿db-¿where('BUILDING$_I D'$,buildingId); $this->$
$db->update('BUILDING')$;

function editRoom($roomid$,roomname, $buildingid$)array = array('ROOM$_N AME'$ =>roomname,
'BUILDING$_I D'$ =>buildingid); $this->db->set($array); $this->db->where('ROOM_I D'$,roomid);
return $this->db->update('ROOM')$;

function deleteBuilding($buildingId$)

$this->db->where('BUILDING_I D'$,buildingId); $query$ =this-¿db-¿delete('BUILDING');
if($query)returntrue;elsereturnfalse$;

function getRoomsCount($search$)this-¿db-¿from('ROOM'); $this->db->join('BUILDING',' ROOM.BUIL$
$BUILDING.BUILDING_I D'$);this-¿db-¿order$_b y("ROOM_N AME","ASC"); if($search != null) $this->$
$db->like('LOWER(ROOM.ROOM_I D)'$,search, 'both'); $this->db->or_l ike('LOWER(ROOM.ROOM_N AME$
'both'); $this->db->or_l ike('LOWER(BUILDING.BUILDING_N AME)'$,search, 'both'); $query$ =this-
¿db-¿get(); return $query->num_r ows()$;

function getBuildingsCount($search$)this-¿db-¿order$_b y('BUILDING_N AME'); if($search != null)$
$this->db->like('LOWER(BUILDING_N AME)'$,search, 'both'); $this->db->or_l ike('LOWER(ROUTE)'$,s
'both'); $query$ =this-¿db-¿get('BUILDING'); return $query->num_r ows()$;

function getAllChangeVenueCount() $this->db->where('DATE_OF_CHANGE >=', date('d-M-y'))$;query = $this->db->get('CHANGE_VENUE_FORM')$; $if(!empty(query))$ if($query->num_rows() == 1)return$query-¿num$_rows()$; $elsereturn$"0";$elsereturn$"0";

function countPendingConfirmation() $this->db->select('COUNT(C.CONFIRMATION_NOTICE_ID)AS$ ¿db-¿from('CONFIRMATION$_NOTICEC'$);this-¿db-¿join('FACULTY$_ATTENDANCEA','C.CONFIRMATION$ $A.CONFIRMATION_NOTICE_ID','inner'$);where = array('C.CONFIRMED' =¿ 0); $this->db->where$(where); $query$ =this-¿db-¿get()-¿row(); return $query$;

function countPresentStatus() $this->db->select('COUNT(A.FACULTY_ATTENDANCE_ID)ASCOUNT'$ ¿db-¿from('FACULTY$_ATTENDANCEA'$);this-¿db-¿join('CLASS$_SCHEDULES','A.CLASS_SCHEDULE_ID =$ $S.CLASS_SCHEDULE_ID','inner'$);where = array('A.STATUS' =¿ 'Present', 'A.VALIDATED IS NULL' =¿ null); $this->db->where$(where); $query$ =this-¿db-¿get()-¿row(); return $query->$ $COUNT$;

function countAbsentStatus() $this->db->select('COUNT(A.FACULTY_ATTENDANCE_ID)ASCOUNT'$ ¿db-¿from('FACULTY$_ATTENDANCEA'$);this-¿db-¿join('CLASS$_SCHEDULES','A.CLASS_SCHEDULE_ID =$ $S.CLASS_SCHEDULE_ID','inner'$);where = array('A.STATUS' =¿ 'Absent', 'A.VALIDATED' =¿ 1); $this->db->where$(where); $query$ =this-¿db-¿get()-¿row(); return $query->COUNT$;

function latestAbsent() $this->db->select("CONCAT(CONCAT(F.FIRST_NAME,' '),F.LAST_NAME),S.$ ¿db-¿from('FACULTY F'); $this->db->join('CLASS_SCHEDULES','F.FACULTY_ID =$ $S.FACULTY_ID','left'$);this-¿db-¿join('FACULTY$_ATTENDANCEA','S.CLASS_SCHEDULE_ID =$ $A.CLASS_SCHEDULE_ID','left'$);where = array('A.STATUS' =¿ 'Absent', 'A.ATTENDANCE$_DATE' =>$ $date('d-M-y'))$;this-¿db-¿where('A.STATUS', 'Absent'); $this->db->order_by('A.SECOND_CHECK','DESC'$ ¿db-¿limit(5);

$query$ =this-¿db-¿get(); if(!empty($query$))if(query-¿result$_array() == 1)return$query-¿result$_array(); elsereturn$"

function hrmoAccounts($limit$,start) $this->db->select("S.STAFF_ID,CONCAT(CONCAT(CONCAT(CO$ ¿db-¿from('STAFF S'); $this->db->join('CHECKERC','S.CHECKER_ID = C.CHECKER_ID','left'$);this-¿db-¿order$_by('S.PRIVILEGE','ASC'$);this-¿db-¿limit($limit$,start); $query$ =this-¿db-¿get(); if(!empty($query$))if(qu ¿result$_array() == 1)return$query-¿result$_array(); elsereturn$"0";$elsereturn$"0";

function registerAccount($data$)this-¿db-¿insert('STAFF', $data$);

function registerCheckerAccount($data$1,data2) $this->db->insert('CHECKER'$,data2); $this->db->insert('STAFF'$,data1);

function fileChangeVenue($data$)this-¿db-¿insert('CHANGE$_VENUE_FORM'$,data);

function confirmChangeVenue($accept$,id) $this->db->set('CONFIRMED'$,accept); $this->db->where('CHANGE_VENUE_ID'$,id); $this->db->update('CHANGE_VENUE_FORM'); ?>

A.3.3 FacultyModel.php

[language=PHP] ¡?php

class FacultyModel extends CI_Model

Personal Faculty Data user info function user_info(idNum) $this->db->where('FACULTY_ID'$,idNum); $query =this-¿db-¿get('FACULTY'); return $query->row_array();

Profile function getMyProfile($idNum$)

$this->db->where('FACULTY_ID'$,idNum); $query =this-¿db-¿get('FACULTY'); return $query->row();

function countPendingConfirmation($idNum$)

$this->db->select('COUNT(C.CONFIRMATION_NOTICE_ID)ASCOUNT'$);this-¿db-¿from('CONFIRMATION_NOTICEC'$);this-¿db-¿join('FACULTY_ATTENDANCEA',' C.CONFIRMATION_NOTA.CONFIRMATION_NOTICE_ID',' inner'$);this-¿db-¿join('CLASS_SCHEDULES',' A.CLASS_SCHEDULE_ID = S.CLASS_SCHEDULE_ID',' inner'$);where = array('C.CONFIRMED' =¿ 0, 'S.FACULTY_ID' =>idNum); $this->db->where(where); $query =this-¿db-¿get()-¿row(); return $query;

function countAbsentStatus($idNum$)

$this->db->select('COUNT(A.FACULTY_ATTENDANCE_ID)ASCOUNT'$);this-¿db-¿from('FACULTY_A¿db-¿join('CLASS_SCHEDULES',' A.CLASS_SCHEDULE_ID = S.CLASS_SCHEDULE_ID',' inner'$);where = array('A.STATUS' =¿ 'Absent', 'S.FACULTY_ID' =>idNum, 'A.VALIDATED' =¿ 1); $this->db->where(where); $query =this-¿db-¿get()-¿row(); return $query;

function countPresentStatus($idNum$)

$this->db->select('COUNT(A.FACULTY_ATTENDANCE_ID)ASCOUNT'$);this-¿db-¿from('FACULTY_A¿db-¿join('CLASS_SCHEDULES',' A.CLASS_SCHEDULE_ID = S.CLASS_SCHEDULE_ID',' inner'$);where = array('A.STATUS' =¿ 'Present', 'S.FACULTY_ID' =>idNum); $this->db->where(where); $query =this-¿db-¿get()-¿row();

return $query$;

function getMyScheduleCount($search$,idNum)

$this->db->where('FACULTY_ID'$,idNum); if($search$! = $null$)this-¿db-¿like('LOWER(SUBJECT_CODE)',se'both'); $query =this-¿db-¿get('CLASS_SCHEDULE'$); $return$query-¿num_rows();

MY Class Schedule function getMySchedule($idNum$)

$this->db->where('FACULTY_ID',$idNum); query =$this->db->get('CLASS_SCHEDULE'); return$query->result_array();

function getMySchedulePagination($search$,idNum, $limit$,start)

$this->db->where('FACULTY_ID',$idNum); $this->db->limit($limit, $start$); if($search != null) $this->db->like('LOWER(SUBJECT_CODE)',$search, 'both'); $query =$this->db->get('CLASS_SCHEDULE'); return$query->result_array();

MY Attendance Records function getMyRecord($idNum$)

$this->db->select('A.ATTENDANCE_DATE, C.CLASS_DAY, C.START_TIME, C.END_TIME, C.SUBJE$
¿db->from('CLASS_SCHEDULEC');this->db->where('FACULTY_ID',idNum); $this->db->join('FACULTY_AT$
$C.CLASS_SCHEDULE_ID');this->db->join('CONFIRMATION_NOTICEN',' N.FACULTY_ATTENDANCE_ID =$
$A.FACULTY_ATTENDANCE_ID');this->db->join('CHANGE_VENUE_FORMV',' A.CHANGE_VENUE_ID =$
$V.CHANGE_VENUE_ID',' left');

$query =$this->db->get(); if(!empty($query$))if(query->result_array() == 1)return$query->result_array(); elsereturn"

function getTodaysAbsentRecord($idNum$)this->db->select('S.SUBJECT_CODE, $S.START_TIME, S.END_TIME$
¿db->from('FACULTY_ATTENDANCEA');this->db->join('CLASS_SCHEDULES',' $A.CLASS_SCHEDULE_ID =$
$S.CLASS_SCHEDULE_ID',' inner');where = array('S.FACULTY_ID' =>idNum, 'A.ATTENDANCE_DATE' =>$
$strtoupper(date('d-M-y')),' A.STATUS' =>' Absent');this->db->where($where$);this->db->limit(5);
$query =$this->db->get()->result_array();

return $query$;

function getTodaysAbsentRecordDepartment($dept$)this->db->select('S.SUBJECT_CODE, $S.START_TIME, S.EN$
¿db->from('FACULTY_ATTENDANCEA');this->db->join('CLASS_SCHEDULES',' $A.CLASS_SCHEDULE_ID =$
$S.CLASS_SCHEDULE_ID',' inner');this->db->join('FACULTY F', 'S.FACULTY_ID = F.FACULTY_ID',' inner');wh$
= array('F.DEPARTMENT' =¿ $dept$,' $A.STATUS' =>' Absent',' A.ATTENDANCE_DATE' =>$
$strtoupper(date('d-M-y')));this->db->where($where$);query = $this->db->get()->result_array(); return$query;

function getTodaysAbsentRecordCollege($coll$)this->db->select('S.SUBJECT_CODE, $S.START_TIME, S.END_TI$
¿db->from('FACULTY_ATTENDANCEA');this->db->join('CLASS_SCHEDULES',' $A.CLASS_SCHEDULE_ID =$
$S.CLASS_SCHEDULE_ID',' inner');this->db->join('FACULTY F', 'S.FACULTY_ID = F.FACULTY_ID',' inner');wh$
= array('F.COLLEGE' =¿ $coll$,' $A.STATUS' =>' Absent',' A.ATTENDANCE_DATE' => strtoupper(date('d-$
$M-y')));this->db->where($where$);query = $this->db->get()->result_array(); return$query;

display all confirmation notice which are marked absent for certain instructor

function confirmNotice($idNum$)

$this->db->select('FACULTY_ATTENDANCE_ID, N.CONFIRMATION_NOTICE_ID, ATTENDANCE$
¿db-¿from('CONFIRMATION$_N OTICEN'$);this-¿db-¿where('FACULTY$_I D'$,idNum); $this->db->$
$where('STATUS',' Absent'$);this-¿db-¿where('CONFIRMED', 0); $this->db->join('FACULTY_ATTENDANC$
$A.CONFIRMATION_NOTICE_ID'$);this-¿db-¿join('CLASS$_S CHEDULEC'$,$' A.CLASS_SCHEDULE_ID =$
$C.CLASS_SCHEDULE_ID'$);

$query$ =this-¿db-¿get(); return $query->result_array()$;

function confirmNoticePagination($idNum$,limit, $start$)

$this->db->select('FACULTY_ATTENDANCE_ID, N.CONFIRMATION_NOTICE_ID, ATTENDANCE$
¿db-¿from('CONFIRMATION$_N OTICEN'$);this-¿db-¿where('FACULTY$_I D'$,idNum); $this->db->$
$where('STATUS',' Absent'$);this-¿db-¿where('CONFIRMED', 0); $this->db->join('FACULTY_ATTENDANC$
$A.CONFIRMATION_NOTICE_ID'$);this-¿db-¿join('CLASS$_S CHEDULEC'$,$' A.CLASS_SCHEDULE_ID =$
$C.CLASS_SCHEDULE_ID'$);this-¿db-¿limit($limit$,start);

$query$ =this-¿db-¿get(); return $query->result_array()$;

function confirmNoticeCount($idNum$)

$this->db->select('FACULTY_ATTENDANCE_ID, N.CONFIRMATION_NOTICE_ID, ATTENDANCE$
¿db-¿from('CONFIRMATION$_N OTICEN'$);this-¿db-¿where('FACULTY$_I D'$,idNum); $this->db->$
$where('STATUS',' Absent'$);this-¿db-¿where('CONFIRMED', 0); $this->db->join('FACULTY_ATTENDANC$
$A.CONFIRMATION_NOTICE_ID'$);this-¿db-¿join('CLASS$_S CHEDULEC'$,$' A.CLASS_SCHEDULE_ID =$
$C.CLASS_SCHEDULE_ID'$);

$query$ =this-¿db-¿get(); return $query->num_rows()$;

function confirmedStatus(cId,reason) $date = date('d-M-Y'$);set = array('CONFIRMATION$_N OTICE_D ATE'$ =
'CONFIRMED' =¿ 1, 'REASON' =¿ $reason$);this-¿db-¿set(set);this-¿db-¿where('CONFIRMATION$_N OTICE_I D'$,cI
$this->db->update('CONFIRMATION_NOTICE'$);

set attendance to present after confirming w/ reason function updateStatus(aID,conf)

if($conf == 1$)this-¿db-¿set('STATUS', 'Present'); $this->db->set('VALIDATED'$, 0);this-
¿db-¿where('FACULTY$_A TTENDANCE_I D'$,aID); $this->db->update('FACULTY_ATTENDANCE'$);

function updateValidated(aID)this-¿db-¿set('VALIDATED', 1); $this->db->where('FACULTY_ATTENDA$
$this->db->update('FACULTY_ATTENDANCE'$);

set the confirmation notice true 1 = Confirmation Notice has been confirmed and 0 = for confir-

mation notice hasn't been confirmed function confirmAsPresent(*cnid*,reason) $date = date('d-M-Y')$;set = array('CONFIRMATION$_N OTICE_D ATE'$ =>date, 'CONFIRMED' =¿ 1, 'REASON' =¿ *reason*);this-¿db-¿set(*set*);this-¿db-¿where('CONFIRMATION$_N OTICE_I D'$,cnid); $this->db->update('CONFIRMATION_NOTICE')$;

function confirmAsAbsent(*cnid*,reason) $date = date('d-M-Y')$;rem .= 'Absent = "'.*reason*.'"';set = array('CONFIRMATION$_N OTICE_D ATE'$ =>date, 'CONFIRMED' =¿ 1, 'REMARKS' =¿ *rem*);this-¿db-¿set(*set*);this-¿db-¿where('CONFIRMATION$_N OTICE_I D'$,cnid); $this->db->update('CONFIR$

set attendance to present after confirming w/ reason //returns a Chairperson value even there's additional wording before and after the word function designationChairperson() $this->db->select('DESIGNATION')$;this-¿db-¿distinct() ; $this->db->like('DESIGNATION','Chairperson','both')$; re ¿db-¿get('FACULTY')-¿row(); //returns a Dean value even there's additional wording before and after the word function designationDean() $this->db->select('DESIGNATION')$;this-¿db-¿distinct(); $this->db->like('DESIGNATION','Dean','both')$; $return$this-¿db-¿get('FACULTY')-¿row(); //returns a Faculty value even there's additional wording before or after the word function designationFaculty() $this->db->select('DESIGNATION')$;this-¿db-¿distinct(); $this->db->like('DESIGNATION','Faculty','both')$; $return$this-¿db-¿get('FACULTY')-¿row();

CHAIRPERSON display all appeals function attendanceAppeals(*coll*,dept)

$this->db->select('FACULTY_ATTENDANCE_ID, N.CONFIRMATION_NOTICE_ID, LAST_NAME, FI$ ¿db-¿from('CONFIRMATION$_N OTICEN'$);this-¿db-¿join('FACULTY$_A TTENDANCEA'$,' $A.CONFIRMATION$ $N.CONFIRMATION_NOTICE_ID'$);this-¿db-¿join('CLASS$_S CHEDULEC'$,' $C.CLASS_SCHEDULE_ID =$ $A.CLASS_SCHEDULE_ID'$);this-¿db-¿join('FACULTY F','F.FACULTY$_I D = C.FACULTY_ID'$);this-¿db-¿where('CONFIRMED', 1); if($coll != NULL$)this-¿db-¿where('COLLEGE', *coll*); $else$this-¿db-¿where('DEPARTMENT', *dept*);

return $this->db->get()->result_array()$;

function getAttendanceAppealsCount(*coll*,dept)

$this->db->select('FACULTY_ATTENDANCE_ID, N.CONFIRMATION_NOTICE_ID, LAST_NAME, FI$ ¿db-¿from('CONFIRMATION$_N OTICEN'$);this-¿db-¿join('FACULTY$_A TTENDANCEA'$,' $A.CONFIRMATION$ $N.CONFIRMATION_NOTICE_ID'$);this-¿db-¿join('CLASS$_S CHEDULEC'$,' $C.CLASS_SCHEDULE_ID =$ $A.CLASS_SCHEDULE_ID'$);this-¿db-¿join('FACULTY F','F.FACULTY$_I D = C.FACULTY_ID'$);this-¿db-¿where('CONFIRMED', 1); $this->db->where('VALIDATED',0); if($coll != NULL) $this->db->where('COLLEGE'$,coll); else $this->db->where('DEPARTMENT'$,dept);

return $this->db->get()->num_rows()$;

function validateAppeal(aid,cid, $action$,remarks)

if($action == 1$)this-¿db-¿set('STATUS', "Present"); $this->db->set('VALIDATED', 1)$;this-¿db-¿where('FACULTY$_A$TTENDANCE$_I$D',aid); $this->db->update('FACULTY_ATTENDANCE')$;

$this->db->set('REMARKS'$,remarks); $this->db->where('CONFIRMATION_NOTICE_ID'$,cid); $this->db->update('CONFIRMATION_NOTICE')$; $else$this-¿db-¿set('STATUS', 'Absent'); $this->db->set('VALIDATED', 1)$;this-¿db-¿where('FACULTY$_A$TTENDANCE$_I$D',aid); $this->db->update('FACULTY_ATTENDANCE')$;

$this->db->set('REMARKS'$,remarks); $this->db->where('CONFIRMATION_NOTICE_ID'$,cid); $this->db->update('CONFIRMATION_NOTICE')$;

DEAN BY COLLEGE AND CHAIRPERSON BY DEPARTMENT function getFacultyAttendanceRecordDean(c

¿db-¿select('FIRST$_N$AME, LAST$_N$AME, ATTENDANCE$_D$ATE, STATUS, SUBJECT$_C$ODE, CLASS$_D$AY, C

¿db-¿from('FACULTY$_A$TTENDANCEA');this-¿db-¿join('CLASS$_S$CHEDULEC','A.CLASS$_S$CHEDULE$_I$D =

C.CLASS$_S$CHEDULE$_I$D');this-¿db-¿join('FACULTY F', 'C.FACULTY$_I$D = F.FACULTY$_I$D');this-¿db-¿where('A.STATUS !=', null); $this->db->where('COLLEGE'$,coll);

return $this->db->get()->result_array()$;

function getFacultyAttendanceRecordChair($dept$,limit, $start$)this-¿db-¿select('FIRST$_N$AME, LAST$_N$AME, AT

¿db-¿from('FACULTY$_A$TTENDANCEA');this-¿db-¿join('CLASS$_S$CHEDULEC','A.CLASS$_S$CHEDULE$_I$D =

C.CLASS$_S$CHEDULE$_I$D');this-¿db-¿join('FACULTY F', 'C.FACULTY$_I$D = F.FACULTY$_I$D');this-¿db-¿where('A.STATUS !=', null); $this->db->where('DEPARTMENT'$,dept); $this->db->limit$(limit, $start$); $return$this-¿db-¿get()-¿result$_a$rray();

function getFacultyAttendanceRecordChairCount($dept$)this-¿db-¿select('FIRST$_N$AME, LAST$_N$AME, ATTEN

¿db-¿from('FACULTY$_A$TTENDANCEA');this-¿db-¿join('CLASS$_S$CHEDULEC','A.CLASS$_S$CHEDULE$_I$D =

C.CLASS$_S$CHEDULE$_I$D');this-¿db-¿join('FACULTY F', 'C.FACULTY$_I$D = F.FACULTY$_I$D');this-¿db-¿where('A.STATUS !=', null);

$this->db->where('DEPARTMENT'$,dept);

return $this->db->get()->num_rows()$;

function getAttendance(mon,day, id)0this-¿db-¿select('A.ATTENDANCE$_D$ATE, C.CLASS$_D$AY, C.START$_T$IM

¿db-¿from('FACULTY$_A$TTENDANCEA');this-¿db-¿join('CLASS$_S$CHEDULEC','A.CLASS$_S$CHEDULE$_I$D =

C.CLASS$_S$CHEDULE$_I$D','inner');this-¿db-¿where('C.FACULTY$_I$D',id); $this->db->where('A.STATUSIS$M

if($mon != NULL$)this-¿db-¿like('EXTRACT(MONTH FROM ATTENDANCE$_D$ATE)',mon,

'none'); $this->db->like('EXTRACT(DAYFROMATTENDANCE_DATE)',day, 'none');

$this->db->order_by('A.ATTENDANCE_DATE','DESC'); return this-¿db-¿get(); MOBILE FUNCTIONS function createConfirmationNotice($faculty_attendance_id$)date = date('d-F-y h.i.s.u A');

do $confirmation_notice$ =this-¿api_model->generate_unique_id(9); while(!(this-¿api_model->is_unique_id_available('CONFIRMATION_NOTICE_ID',confirmation_notice,'CONFIRMATION_NOTICE'))); $data = array('CONFIRMATION_NOTICE_ID'$ =>confirmation_notice,'FACULTY_ATTENDANCE_ID' =>f 'REASON' =¿ "", 'ELECTRONIC_SIGNATURE' => "",'REMARKS' => "Absent",'CONFIRMED' => "0");

$query$ =this-¿db-¿insert('CONFIRMATION_NOTICE',data);

if($query$)return true; else return false;

function send_email_notification(faculty_attendance_id)config = array('protocol' =¿ 'smtp', 'smtp_host' =>' ssl://smtp.googlemail.com','smtp_port' => 465,'smtp_user' =>' @gmail.com', sender'semail'smtp_pass' =>' ', sender'spassword'mailtype' =>' html');

$this->email->initialize$(config); $this->email->set_newline$("");

$emailaddress$ =this-¿get_email(faculty_attendance_id);

$this->email->from('testemailtest1230@gmail.com','HRMO')$;this-¿email-¿to($emailaddress$);this-¿email-¿subject('Absence Notification'); $this->email->message('You were recorded as absent.')$;

if($this->email->send()$)return true; else return false;

function fetchMyClassScheduleAjax(id)this-¿db-¿where('FACULTY_ID',id); return $this->db->get('CLASS_SCHEDULE')$;

function fetchMyAttendanceAjax(id)

$this->db->select('ATTENDANCE_DATE, CLASS_DAY, START_TIME, END_TIME, SUBJECT_CODE,$ ¿db-¿from('CLASS_SCHEDULEC');this-¿db-¿join('FACULTY_ATTENDANCEA','A.CLASS_SCHEDULE_ID = C.CLASS_SCHEDULE_ID');this-¿db-¿join('CONFIRMATION_NOTICEN','N.FACULTY_ATTENDANCE_ID = A.FACULTY_ATTENDANCE_ID');this-¿db-¿join('CHANGE_VENUE_FORMV','A.CHANGE_VENUE_ID = V.CHANGE_VENUE_ID','left');this-¿db-¿where('FACULTY_ID',id);

return $this->db->get()$; ? >

A.3.4 FacultyReportModel.php

[language=PHP] ¡?php

class FacultyReportModel extends CI$_{M}odel$

returns unique school year function getSchoolYear() $this->db->select('SCHOOL_YEAR');$this-¿db-¿distinct(); $this->db->order_by('SCHOOL_YEAR','DESC'); return$this-¿db-¿get('CLASS$_{S}CHEDULE')-result_array();$

returns unique semester function getSemester() $this->db->select('SEMESTER');$this-¿db-¿distinct(); $this->db->order_by('SEMESTER','ASC'); return$this-¿db-¿get('CLASS$_{S}CHEDULE')->result_array();$

returns unique colleges with data having College function getCollege() $this->db->select('COLLEGE');$this-¿db-¿distinct(); $this->db->like('COLLEGE','College');$this-¿db-¿order$_{by}('COLLEGE','ASC'); return$this-¿db-¿get('FACULTY')-¿result$_{array}();$

returns unique departments with data having Department function getDepartment(*coll*)this-¿db-¿select('DEPARTMENT'); $this->db->distinct();$this-¿db-¿where('COLLEGE', *coll*);this-¿db-¿like('DEPARTMENT', 'Department'); $this->db->order_by('DEPARTMENT','ASC'); return$this-¿db-¿get('FACULTY')-¿result$_{array}();$

Monthly Absence Report

returns all data if parameters are null returns faculty, school year and attendance data base on search function getAbsenceData(*dept*,coll, *sem*,month, *syear*)this-¿db-¿select('LAST$_{NAME}, FIRST_{NAME}, SUBS$
¿db-¿from('FACULTY$_{ATTENDANCEA}$');this-¿db-¿join('CLASS$_{S}CHEDULEC','A.CLASS_{S}CHEDULE_{ID}=$
$C.CLASS_{S}CHEDULE_{ID}$');this-¿db-¿join('FACULTY F', 'C.FACULTY$_{ID}=F.FACULTY_{ID}$');this-¿db-¿join('CONFIRMATION$_{NOTICEN}$','$A.CONFIRMATION_{NOTICE_{ID}}=N.CONFIRMATION_{NOTIC}$
¿db-¿where('STATUS', 'Absent'); $this->db->where('N.CONFIRMED',1);$this-¿db-¿where('VALIDATED !=', '0'); $this->db->or_where('VALIDATEDISNULL',null);$

if($month!=null$)this-¿db-¿like('DEPARTMENT', *dept*,'*both*');this-¿db-¿like('COLLEGE', *coll*,'*both*');this-¿db-¿like('SEMESTER', *sem*,'*both*');this-¿db-¿like('EXTRACT(MONTH FROM ATTENDANCE$_{DATE}$)',month, 'both'); $this->db->like('SCHOOL_YEAR'$,syear, 'both'); else $this->db->like('EXTRACT(MONTHFR$
1);this-¿db-¿order$_{by}('LAST_{NAME}','ASC'); return$this-¿db-¿get();

Monthly Report returns faculty, school year and attendance data base for reporting function getAbsenceReportData(*coll*,dept, *sem*,mon, *syear*)this-¿db-¿select('F.FACULTY$_{ID}, LAST_{NAME}, FIRST_{NAME}$,
¿db-¿from('FACULTY$_{ATTENDANCEA}$');this-¿db-¿join('CLASS$_{S}CHEDULEC','A.CLASS_{S}CHEDULE_{ID}=$
$C.CLASS_{S}CHEDULE_{ID}$');this-¿db-¿join('FACULTY F', 'C.FACULTY$_{ID}=F.FACULTY_{ID}$');this-¿db-¿join('CONFIRMATION$_{NOTICEN}$','$A.CONFIRMATION_{NOTICE_{ID}}=N.CONFIRMATION_{NOTIC}$

¿db-¿where('STATUS', 'Absent'); $this->db->where('N.CONFIRMED', 1)$;this-¿db-¿where('VALIDATED !=', '0'); $this->db->or_whereh('VALIDATEDISNULL', null)$; if(mon != null) $this->db->like('DEPARTMENT'$,dept, 'both'); $this->db->like('COLLEGE'$,coll, 'both'); $this->db->like('SEMESTER'$,sem, 'both'); $this->db->like('EXTRACT(MONTHFROMATTENDANCE_DA$ 'both'); $this->db->like('SCHOOL_YEAR'$,syear, 'both'); $this->db->order_by('LAST_NAME', 'ASC'); retu ¿db-¿get();

returns faculty data, depending on college and department function getFaculty(*dept*,col)

$this->db->select('LAST_NAME, FIRST_NAME, MIDDLE_NAME, FACULTY_ID')$;this-¿db-¿where('COLLEGE', *col*);this-¿db-¿where('DEPARTMENT', *dept*);this-¿db-¿order$_by('LAST_NAME', 'ASC'); re$ ¿db-¿get('FACULTY');

FACULTY REPORT SUMMARY

returns all data if parameters are null returns faculty, school year and attendance data base on search function getFacultyData(*col*,dept, *id*,sem, *syear*)this-¿db-¿select('F.FACULTY$_ID, FIRST_NAME, LAST_NA$ ¿db-¿from('FACULTY$_ATTENDANCEA'$);this-¿db-¿join('CLASS$_SCHEDULEC'$,' $A.CLASS_SCHEDULE_ID =$ $C.CLASS_SCHEDULE_ID'$);this-¿db-¿join('FACULTY F', 'C.FACULTY$_ID = F.FACULTY_ID'$);this-¿db-¿join('CONFIRMATION$_NOTICEN'$,' $A.CONFIRMATION_NOTICE_ID = N.CONFIRMATION_NOTIC$ ¿db-¿where('STATUS', 'Absent'); $this->db->where('N.CONFIRMED', 1)$;this-¿db-¿where('VALIDATED !=', 0); $this->db->or_whereh('VALIDATEDISNULL', null)$;

if(!empty(*id*))this-¿db-¿where('F.FACULTY$_ID'$,id); $this->db->like('COLLEGE'$,col, 'both'); $this->db->like('DEPARTMENT'$,dept, 'both'); $this->db->like('SEMESTER'$,sem, 'both'); $this->db->like('SCHOOL_YEAR'$,syear, 'both');

return $this->db->get()$;

Faculty Summary Report returns faculty, school year and attendance data base for reporting function printFacultyDataReport(*col*,dept, *id*,sem, *syear*)this-¿db-¿select('F.FACULTY$_ID, FIRST_NAME, LAST_N$ ¿db-¿from('FACULTY$_ATTENDANCEA'$);this-¿db-¿join('CLASS$_SCHEDULEC'$,' $A.CLASS_SCHEDULE_ID =$ $C.CLASS_SCHEDULE_ID'$);this-¿db-¿join('FACULTY F', 'C.FACULTY$_ID = F.FACULTY_ID'$);this-¿db-¿join('CONFIRMATION$_NOTICEN'$,' $A.CONFIRMATION_NOTICE_ID = N.CONFIRMATION_NOTIC$ ¿db-¿where('STATUS', 'Absent'); $this->db->where('N.CONFIRMED', 1)$;this-¿db-¿where('VALIDATED !=', 0); $this->db->or_whereh('VALIDATEDISNULL', null)$;

if($id != NULL$)this-¿db-¿where('F.FACULTY$_ID'$,id, 'both'); $this->db->like('COLLEGE'$,col, 'both'); $this->db->like('DEPARTMENT'$,dept, 'both'); $this->db->like('SEMESTER'$,sem,

'both'); $this->db->like('SCHOOL_YEAR'$,syear, 'both'); $this->db->group_by('F.FACULTY_ID'); return

¿db-¿get(); ?¿

VITA

Rennel Jun P. Del Monte, Ralph S. Eufracio, Jaime C. Pequiras III are BS Information Technology students of the Department of Computer Science at the Ateneo de Naga University.