Students Portal for Advanced Technological Institute Mattakkuliya

G.K.R Silva

Index number: 0801021

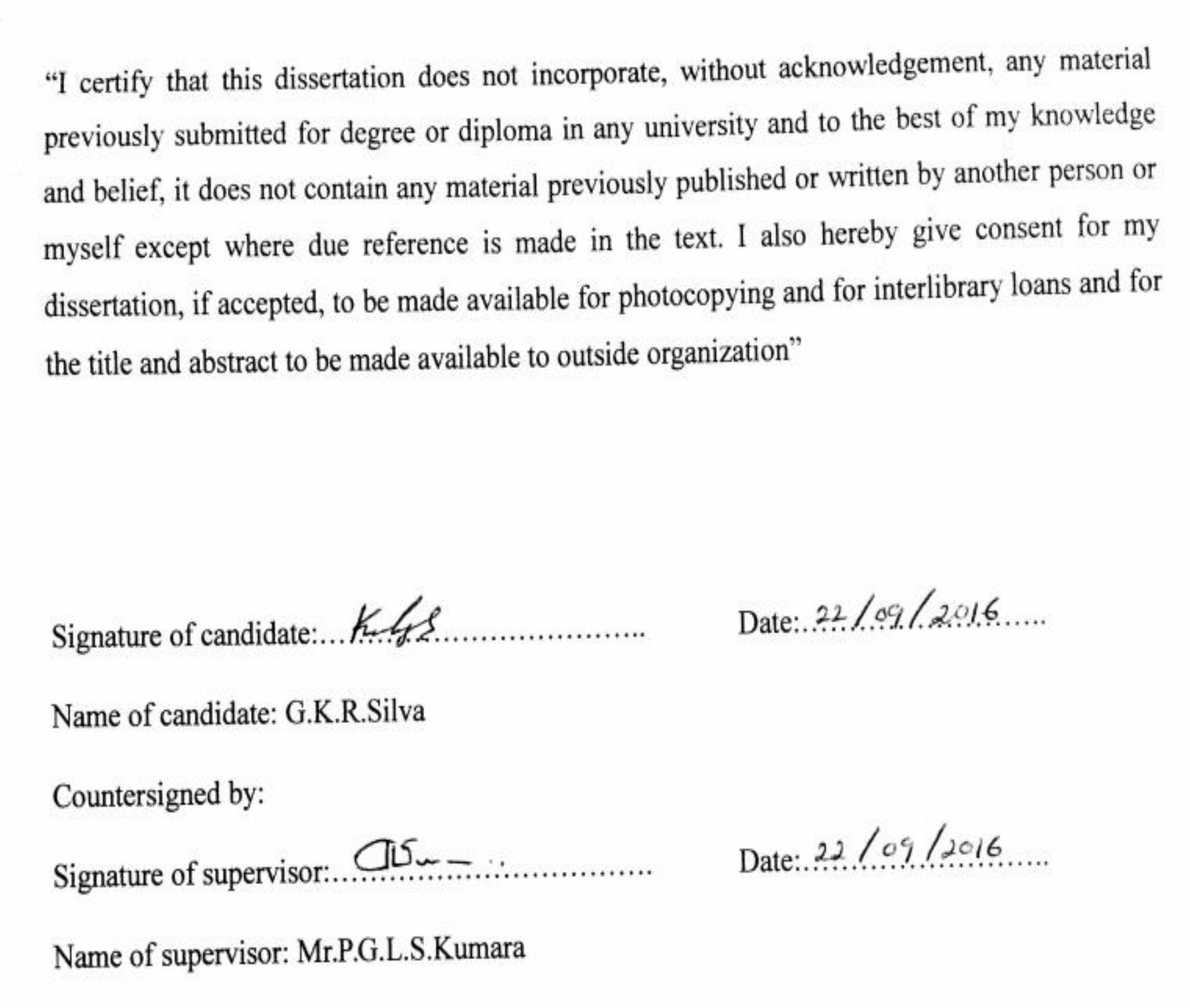
BIT Registration Number: R080102

Supervisor: Mr.P.G.L.S Kumara

November 2017

This dissertation is submitted in partial fulfillment of the requirement of the Degree of Bachelor of Information Technology (External) of the University of Colombo School of Computing

DECLARATION



"I certify that this dissertation does not incorporate, without acknowledgement, any material previously submitted for degree or diploma in any university and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, to be made available for photocopying and for interlibrary loans and for the title and abstract to be made available to outside organization"

Signature of candidate:…………………………… Date:22/09/2016…..

Name of candidate: G.K.R.Silva

Countersigned by:

Signature of supervisor:…………………………… Date:22/09/2016…..

Name of supervisor: Mr.P.G.L.S.Kumara

ABSTRACT

Advanced Technological Institute (ATI), Mattakkuliya has been the pioneer in producing incorporated engineers for the local and international industry. Five Higher national diploma in engineering (HNDE) courses are offered by the institute under three main Engineering Departments (Civil, Electrical, and Mechanical). The main Engineering fields consist of several sub disciplines as follows and the students are given the opportunity to specialize in their chosen option. All the HNDE programs are conducted in semester system in English medium included with 3 years (6 semesters) academic period and minimum 6 months (1 semester) industrial training according to the new syllabus. Regular industrial visits are also organized as possible.

But until now ATI has not used modern technology for sharing notes, delivering assignments, publish exam result. Advanced Technological Institute’s lectures are delivering assignment manually in the lecture hall. As well as students handover their answers for assignment manually. End semester exam results publish on notice board. Therefore students (especially who are in industrial training) have to visit ATI after release the exam result. As well as time table also publish on notice board

Students portal for Advanced Technological Institute Mattakkuliya is developed according to client requirements of prevent current manual systems’ draw backs. This system has provided many facilities such as register new students, register staff, create course, manage subjects, add time table, view time table, add results, view results, publish assignment, submit answer of assignment, print attendance sheet etc.

The student portal has been developed based on the Rational Unified Process (RUP) framework.

The system was developed using PHP (PHP Hypertext Processor) open source server side scripting language. MySQL was used as the database tool for designing the database. Apache web server was chosen to run the system. Additionally, Sublime Text 3 was chosen as the coding tool and Adobe Photoshop CS5 was chosen as the interface designing tool in the main stages of the developing.

ACKNOWLEDGEMENT

First of all I would like to express my gratitude and thankful to the BIT Coordinators and academic staff of University of Colombo School of Computing (UCSC) for giving us valuable degree program and providing us a good guidance continuously.

Special thank must go to my project supervisor Mr P.G.L.S.Kumara for his assistance and guidance.

I must thanks Mr Thushan Ranasinghe for giving me a great support and guidance to complete this project.

I would like to thanks staff of Advanced Technological Institute, Mattakkuliya for giving me great support and dedicated their valuable time to complete my project.

Table of Contents

List of Figures

List of Tables

List of Acronyms

HNDE	-	Higher National Diploma in Engineering
ATI	-	Advanced Technological Institute
QS	-	Quantity Surveying
BSE	-	Building Service Engineering
RUP	-	Rational Unified Process
UML	-	Unified Modeling Language
ER	-	Entity relationship
NF	-	Normal Form
HTML	-	Hypertext Mark- up Language
JSON	-	JavaScript Object Notation
CSS	-	Cascade Style Sheet
GB	-	Gigabyte
RAM	-	Random Access Memory
CPU	-	Central Processing Unit
GUI	-	Graphical User Interface

CHAPTER 01 – INTRODUCTION

1.1 INTRODUCTION

Advanced Technological Institute is a government technological center, which conducts Higher National Diploma, and National Diploma course in technological field. These similar to other government campuses around island.

Many campuses are using Learning Management System or student portal to collaborate with the academic staff and students. Advanced Technological Institute Mattakkuliya is using the manual system for their day today activities. This developing system will help to increase the competency and efficiency of their work, by reducing the heavy paper work. Further they can maintain and manage their day to day activities effectively and efficiently.

1.2 MOTIVATION OF THE PROJECT

Currently Client is carrying their operations in a manual way with lot of heavy paper works. This manual system will be very inefficient and time wasted. The following problems have been occurred in the existing manual system.

- Facing many difficulties when delivering and collecting assignment
- students (especially who are in industrial training) have to visit ATI when release exam results
- All calculations to generate reports are done manually so there is greater chance of errors

1.3 SCOPE OF THE PROPOSED PROJECT

When deciding to develop a system, the scope is a most important fact which we need to consider about. The scope of this project will be as follows.

- System has provide to store the student details, staff details, course details, resource details.
- System has facilitates to create the user accounts for students and staff.
- System has facilitates to show student results, assignment results.
- System has facilitates to manage the subject with the lecturer.
- System has provide the facility to publish notes and assignment
- System has facilitates to show the time table to students as well as to academic staff
- There are 3 user types such as admin, student, and staff.
- Users of each level can be accessed to the system with relevant user privileges.
- Student attendance sheet can print easily
- students able to know their results immediately
- Students able to see time table from anywhere
- Students able to receive assignments and notes easily
- Lectures able to collect students assignments easily

1.4 OBJECTIVES OF THE SYSTEM

Main objective of this system to collaborate staff and students for their academic activities easily and efficiency. Following are the objective of the system

- Increasing accuracy of the assignment submission and delivering notes
- Increasing the efficiency of the releasing exam results
- Increasing the accuracy and efficiency delivering the timetable
- Remote access facility to the system for end users through internet.

1.5 STRUCTURE OF THE DISSERTATION

This Dissertation provides the overall knowledge about the Transport Management System. Dissertation structure as follows

Chapter 02-Analysis

This chapter explains the requirement gathering techniques, details of the manual system, functional nonfunctional requirements and details of the existing systems.

Chapter 03-Design

This chapter explains the use case diagram of the proposed system, database design of the system and main interfaces of the system.

Chapter 04-Implementation

This chapter explains the hardware software requirements, development tools which is used for system, code features and reused existing codes of the system.

Chapter 05-Evaluation

This chapters explains the techniques of testing, details of software testing, high level test plan of the system and client evaluation of the system

Chapter 06-Conclusion

This chapter explains the future enhancements of the system and lesson learnt of the overall project work.

Appendixes

These are provided further details about the content of the dissertation chapters which were not included in the chapters.

CHAPTER 02 – ANALYSIS

2.1 INTRODUCTION

System Analysis is a very important phase of the software development lifecycle. System Analysis Definition: “The analysis of the role of a proposed system and the identification of the requirements that it should meet. This is the starting point for system design. The term is most commonly used in the context of commercial programming, where software developers are often classed as either system analysts or programmers. The systems analysts are responsible for identifying requirements (i.e. systems analysis) and producing a design. The programmers are then responsible for implementing it” **[1]**

2.2 FACT GATHERING TECHNIQUES

Gathering client’s requirements by using the fact finding techniques are the most critical part in the analysis phase. When gathering the requirements, there should be a proper way to handle these techniques. There are several fact finding techniques which can be used to collect the clear and accurate information. In this project facts were gathered by using following techniques.

- Interviews
- Questionnaires
- Sampling and documentation.

When we are gathering requirements using sampling and documentation, it would help to get a clear idea about the system.

- Interviews were conducted with the director and lecturers to gather requirements
- Questionnaires were distribute between the selected students to gather the student requirements

2.3 ANALYZING THE CURRENT MANUAL SYSTEM

2.3.1 STUDENT REGISTRATION AND STUDENTS ATTENDANT SHEET

When student register the ATI he/she want to open the personal file as well as registration panel should enter students' details to database manually and using registration form. Management assistant prepare the attendant sheet manually using that database and take the print out as the bulk. When who leave the course attendant sheet want to update, but in the manual system that very difficult

2.3.2 DELIVERING AND COLLECTING ASSIGNMENT

In the Advanced Technological Institute, lectures are delivering notes and assignment manually in the lecture hall .As well as students handover their answers for assignment manually.

2.3.3 RELEASE OF STUDENT RESULTS

Two or three months after the exam, the results publish on the notice board. In these period students want to ready to another examination and usually student stay vacation on their native homes, sometimes these houses are so far from institute. Therefore students (especially who are in industrial training) have to visit ATI after release the exam results.

2.3.4 TIME TABLE

Before start the semester Head of departments deliver the time tables to lecturers and demonstrators of their departments .Beginning of the semester students time table publish on notice board

2.3.5 DRAWBACKS OF THE CURRENT SYSTEM

- **Not User Friendly:** The existing system is not user friendly because when retrieve the data want to find lot of file.

- **Difficulty in Report Generating:** We require more calculations to generate the report so it is generated at the end of the session. And the student not get a single chance to improve their attendance

- **Manual Control**: All calculations to generate reports are done manually so there is greater chance of errors.

- **Lots of Paperwork**: Existing system requires lot of paper when register the student as well as submitting the assignment.

- **Time Consuming**: Every works are done manually so we cannot generate report in the middle of the session or as per the requirement because it is very time consuming

2.3.6 TOP-LEVEL USE CASE DIAGRAM OF THE EXISTING SYSTEM

Top level use case diagram of the existing system is given below figure 2.1.

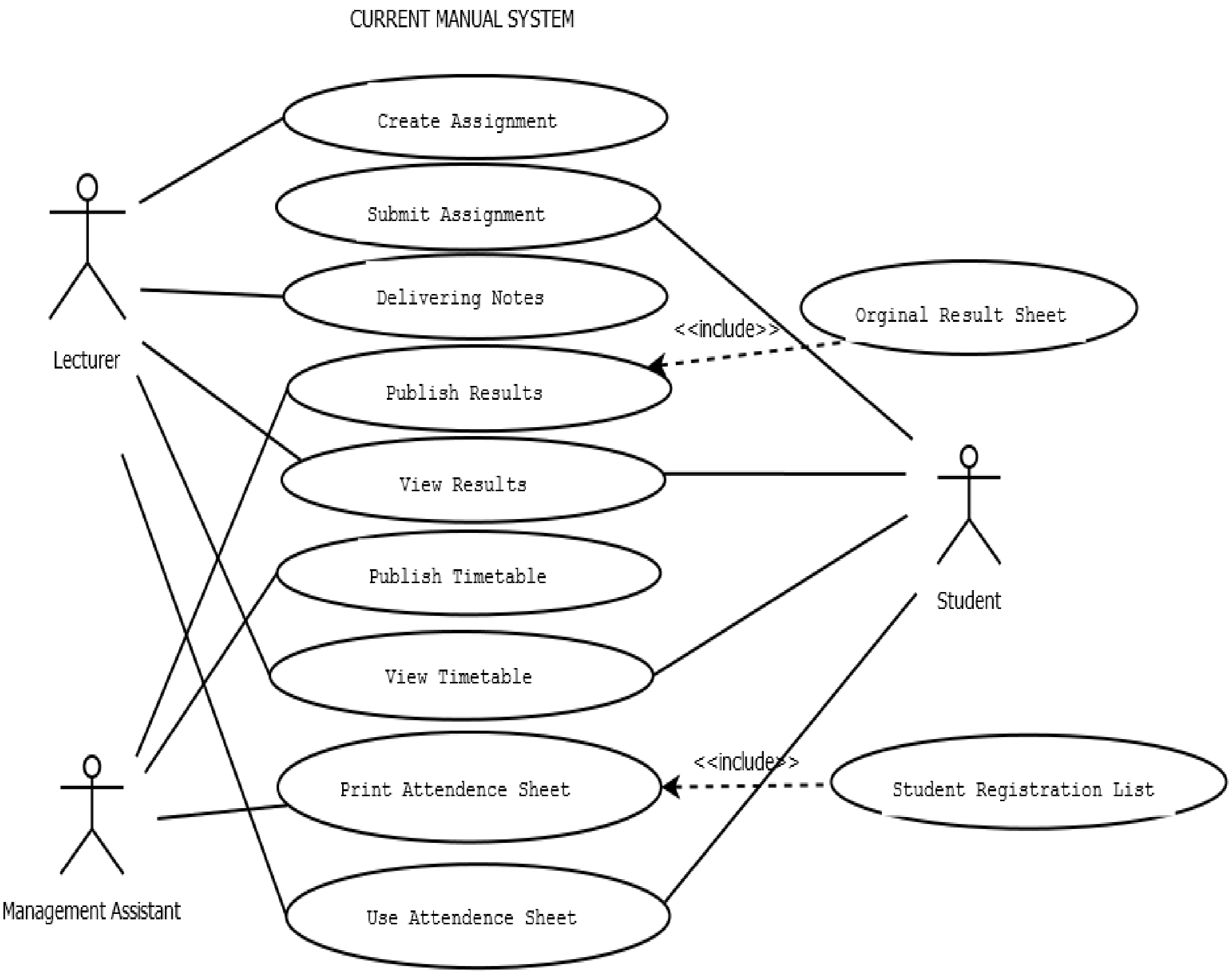


FIGURE 2.1 USE CASE DIAGRAM OF THE EXISTING SYSTEM

2.4 SIMILAR SYSTEMS AND LITERATURE REVIEW

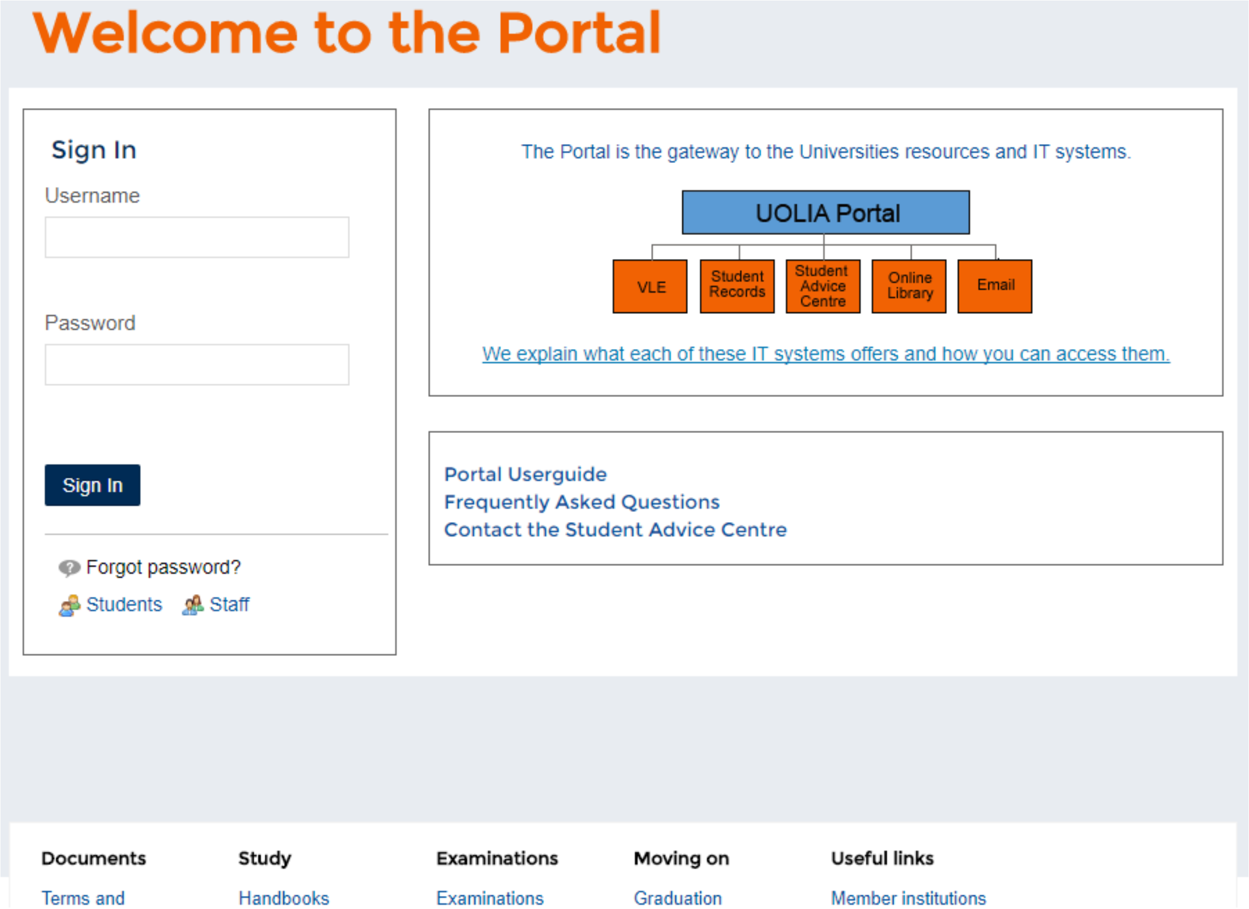
A student portal is an online gateway where students can log into institute or university website to access important program information.

By studying current famous student portal, more experiences could be obtained about how the developing system should be and how the required functionalities should be presented. Following are a few similar systems that were reviewed to build the system:

2.4.1 Student Portal University of London

This student portal is very good and very big system. This student portal provide important news and updates about key dates in the academic calendar, availability of our IT systems and University news. As well as Access to your Virtual Learning Environment (VLE). **[2]**

The home page of the student portal university of London is depicted by figure 2.2.



Welcome to the Portal

The Portal is the gateway to the Universities resources and IT systems.

We explain what each of these IT systems offers and how you can access them.

FIGURE 2.2 HOME PAGE OF THE STUDENT PORTAL

2.4.2 Student Portal University of gothenburg

In this student portal the "New Student" section can find information which is especially important as new student at the University of Gothenburg. Student will get information on how to get a student account, how to register, and how to get access to the learning platform, GUL, for example. The top menu contains seven different entrances to information and services. You can find all the links that are hidden under the menus by looking at the Student Portal start page. **[3]**

The home page of the student portal university of Gothenburg is depicted by figure 2.3.

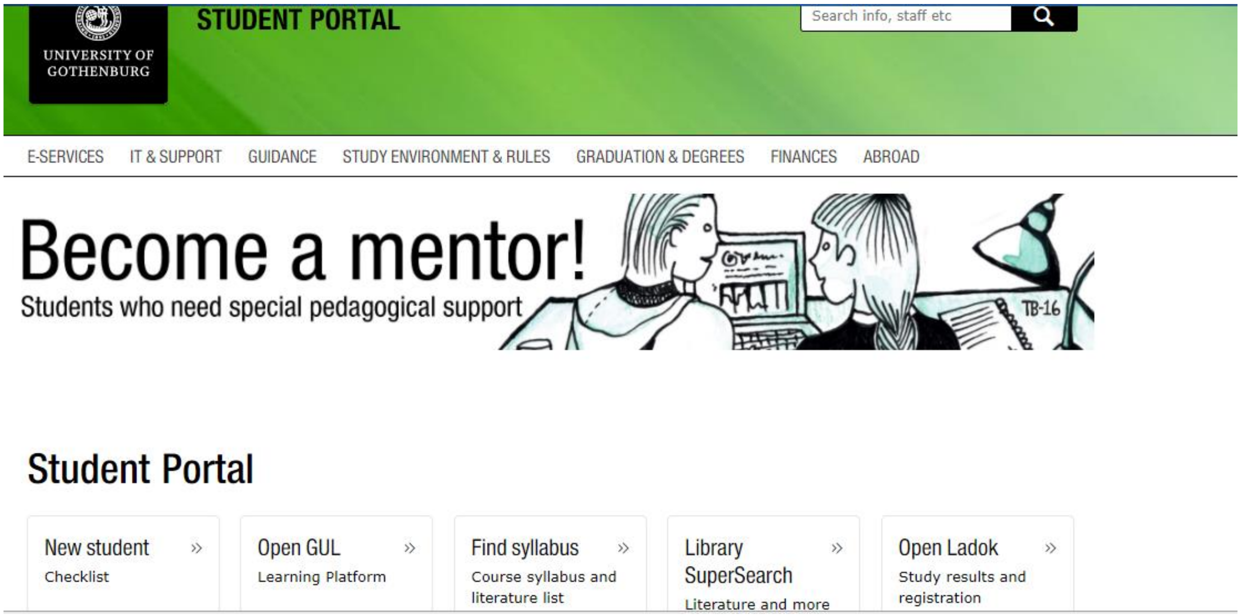


FIGURE 2.3 HOME PAGE OF THE STUDENT PORTAL UNIVERSITY OF GOTHENBURG

2.5 FUNCTIONAL REQUIREMENTS

Functional requirement of the new HNDE student portal are given below

- Lecturers should be able to
  - upload assignments, notes
  - download submitted assignments
  - update profile details, change password
  - take the print out of student attendance sheet

  - view the timetable

- Students should be able to
  - Edit profile
  - download assignments, notes
  - submit assignments
  - see exam results
  - see the timetable
- admin should be able to
  - Create / Delete staff accounts
  - List staff
  - List student
  - Approve students
  - Edit/delete students
  - View academic profile
  - Add /edit courses (e.g. Civil, Mechanical, Electrical, QS, BSE)
  - Add/edit subjects for each courses
  - Add/remove Semesters
  - Add timetable
  - Add/edit lecture hall/lab
  - List lecture hall/lab
  - Import student exam results
  - View student exam results
  - Print students attendance sheet
- System should generate report
  - Students list
  - Staff list
  - Subject list
  - Time table
  - Attendance sheet
  - Exam results

2.6 NON-FUNCTIONAL REQUIREMENTS

“In systems engineering and requirements engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors.” **[4]** .Success of this system is not only dependent on the functional requirements; it is also based on the non-functional requirements. It acts as a main role for the success of the system. Following are the major non-functional requirements which were identified at this stage:

- Data should be reliable; data backups should be done as necessary.
- Overall system should be simple as much as possible with more user friendly environment.
- Access privileges should be maintained accurately.
- Automated notification generating module should be generated at the accurate time levels.
- System security should be highly secure to prevent unauthorized modification and access

CHAPTER 03–DESIGN

3.1 INTRODUCTION

In the **design phase** the **architecture** is established. This phase starts with the requirement document delivered by the requirement phase and maps the requirements into an architecture. The **architecture** defines the components, their interfaces and behaviors. The deliverable design document is the architecture. The design document describes a plan to implement the requirements. This phase represents the ``how" phase. Details on computer programming languages and environments, machines, packages, application architecture, distributed architecture layering, memory size, platform, algorithms, data structures, global type definitions, interfaces, and many other engineering details are established. The design may include the usage of existing components **[5]**:

3.2 OVERVIEW OF PROCESS MODELS

The waterfall model this takes the fundamental process activities of specification, development, validation and evolution and represents them as separate Process phases such as requirements specification, software design, implementation, testing and soon. Evolutionary development this approach interleaves the activities of specification, development and validation. An initial system is rapidly developed from abstract specifications. This is then refined with customer input to produce a system that satisfies the customers’ needs. Component-based software engineering this approach is based on the existence of a significant number of reusable components. The system development process focuses on integrating these components into a system rather than developing them from scratch.

These three generic process models are widely used in current software engineering practice. They are not mutually exclusive and are often used together, especially for large systems development.[Ian Sommerville ,Software Engineering,2007]

3.3 PROCESS MODELS

The **Rational Unified Process** (**RUP**) is an iterative software development process framework created by the Rational Software Corporation, a division of IBM since 2003. RUP is not a single concrete prescriptive process, but rather an adaptable process framework, intended to be tailored by the development organizations and software project teams that will select the elements of the process that are appropriate for their needs. RUP is a specific implementation of the Unified Process.

The RUP has determined a project life-cycle consisting of four phases. These phases allow the process to be presented at a high level in a similar way to how a 'waterfall'-styled project might be presented, although in essence the key to the process lies in the iterations of development that lie within all of the phases. Also, each phase has one key objective and milestone at the end that denotes the objective being accomplished. The visualization of RUP phases and disciplines over time is referred to as the RUP hump chart.

Inception phase

The primary objective is to scope the system adequately as a basis for validating initial costing and budgets. In this phase the business case which includes business context, success factors (expected revenue, market recognition, etc.), and financial forecast is established. To complement the business case, a basic use case model, project plan, initial risk assessment and project description (the core project requirements, constraints and key features) are generated

If the project does not pass this milestone, called the life cycle objective milestone, it either can be cancelled or repeated after being redesigned to better meet the criteria**.[7]**

Elaboration phase

The primary objective is to mitigate the key risk items identified by analysis up to the end of this phase. The elaboration phase is where the project starts to take shape. In this phase the problem domain analysis is made and the architecture of the project gets its basic form.

This phase must pass the lifecycle architecture milestone criteria answering the following questions:

- Is the vision of the product stable?
- Is the architecture stable?
- Does the executable demonstration indicate that major risk elements are addressed and resolved?
- Is the construction phase plan sufficiently detailed and accurate?
- Do all stakeholders agree that the current vision can be achieved using current plan in the context of the current architecture?
- Is the actual vs. planned resource expenditure acceptable?

If the project cannot pass this milestone, there is still time for it to be canceled or redesigned. However, after leaving this phase, the project transitions into a high-risk operation where changes are much more difficult and detrimental when made.

The key domain analysis for the elaboration is the system architecture. **[8]**

Construction phase

The primary objective is to build the software system. In this phase, the main focus is on the development of components and other features of the system. This is the phase when the bulk of the coding takes place. In larger projects, several construction iterations may be developed in an effort to divide the use cases into manageable segments that produce demonstrable prototypes.

This phase produces the first external release of the software. Its conclusion is marked by the initial operational capability milestone. **[9]**

Transition phase

The primary objective is to 'transit' the system from development into production, making it available to and understood by the end user. The activities of this phase include training the end users and maintainers and beta testing the system to validate it against the end users' expectations. The system also goes through an evaluation phase, any developer which is not producing the required work is replaced or removed. The product is also checked against the quality level set in the Inception phase. **[10]**

If all objectives are met, the product release milestone is reached and the development cycle is finished.3.1 figure is describes the Rational unified Process Model

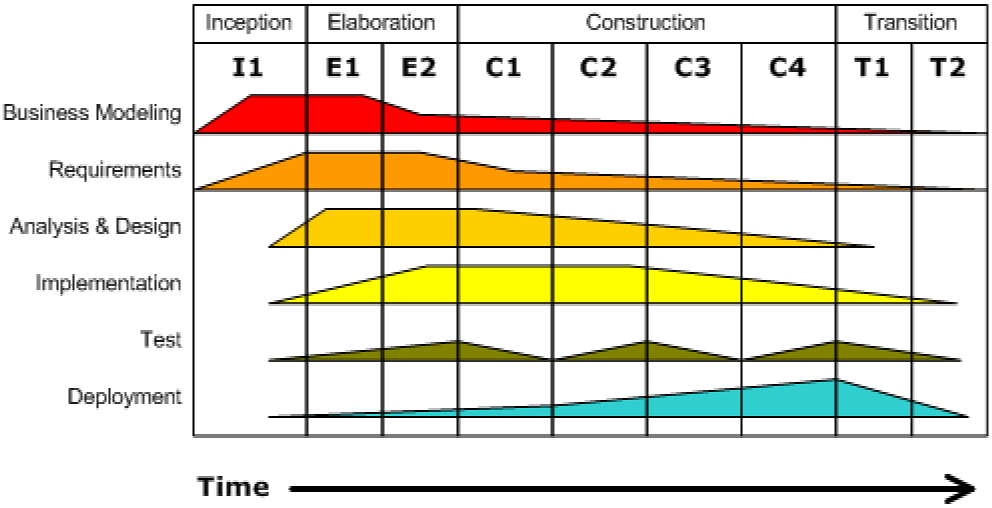


FIGURE 3.1 RATIONAL UNIFIED PROCESS MODEL 1

3.4 ALTERNATIVE SOLUTIONS TO THE SYSTEM

When discussing an alternative solution to this system it can categorize into few different sub topics.

Maintain system based with old file system

The entire institute working process based on the existing file based system can be used by the client. So it can maintain all the work by using paper and generate Institute related reports and valuable feedbacks by analyzing those paper works when necessary. By choosing this method the client has to waste time to get even a trace of previous data.

Purchase Commercial stand-alone Software

Purchases of commercial products to maintain each and every different kind of work in the institute such as user managing and providing attendance sheet can be made by the client. The main issues that can occur when using these kinds of software are as follows:

Platform dependence

Different kinds of operating systems are required by different kinds of software. Therefore, different computer operating systems maintaining different relevant software packages must be maintained by the user.

Software update dependencies

Some software packages depend on updates. The functionality as required cannot be performed by the software without updates. Additional software is required to run by some updates (Frame works). Sometimes more system performances to run are required by that additional software. Updating the computer system into modern system to meet that required performance has to be done ultimately by the user. It costs a lot.

Using free software Packages

Software demos to manage their work can be used by the client. The issue is the trustworthiness of those kinds of software. These kinds of demos can consist of different kinds of errors.

3.5 SYSTEM AS A WEB SOLUTION

As a web based solution, the system has been proposed for implementation without implementing it as a stand-alone system. The following reasons explain it:

- A web based system to automate day to day activities had been requested by the client.
- The cost of hardware as well as software can be reduced by the implementation of a web based solution.
- Web systems are platform independent and any different platform can be used.
- Access to the system can be obtained from anywhere without reaching to the office premises for that particular activity.
- When comparing with stand-alone systems, maintaining a web system is simple.

3.6 OBJECT ORIENTED DESIGNING

Object-oriented design is concerned with developing an object oriented model of a software system to implement the identified requirements. The main goal of this design approach is defining objects define objects and establishing relationship between classes by sending and receiving messages

Object oriented design is a method which is used to collaborate objects and attributes and methods of their objects. Unified Modeling Language is the widely used designing tool to perform objects oriented designing. Unified Modeling Language depicts the model of the objects. This design approach is very efficient way to implement designing phase. [Whitten and Bentley System Analysis and Design Methods ,2007,7th Edition]

3.6.1. HIGH LEVEL USE CASE DIAGRAM

Figure 3.2 show the high level use case diagram for the HNDE students portal

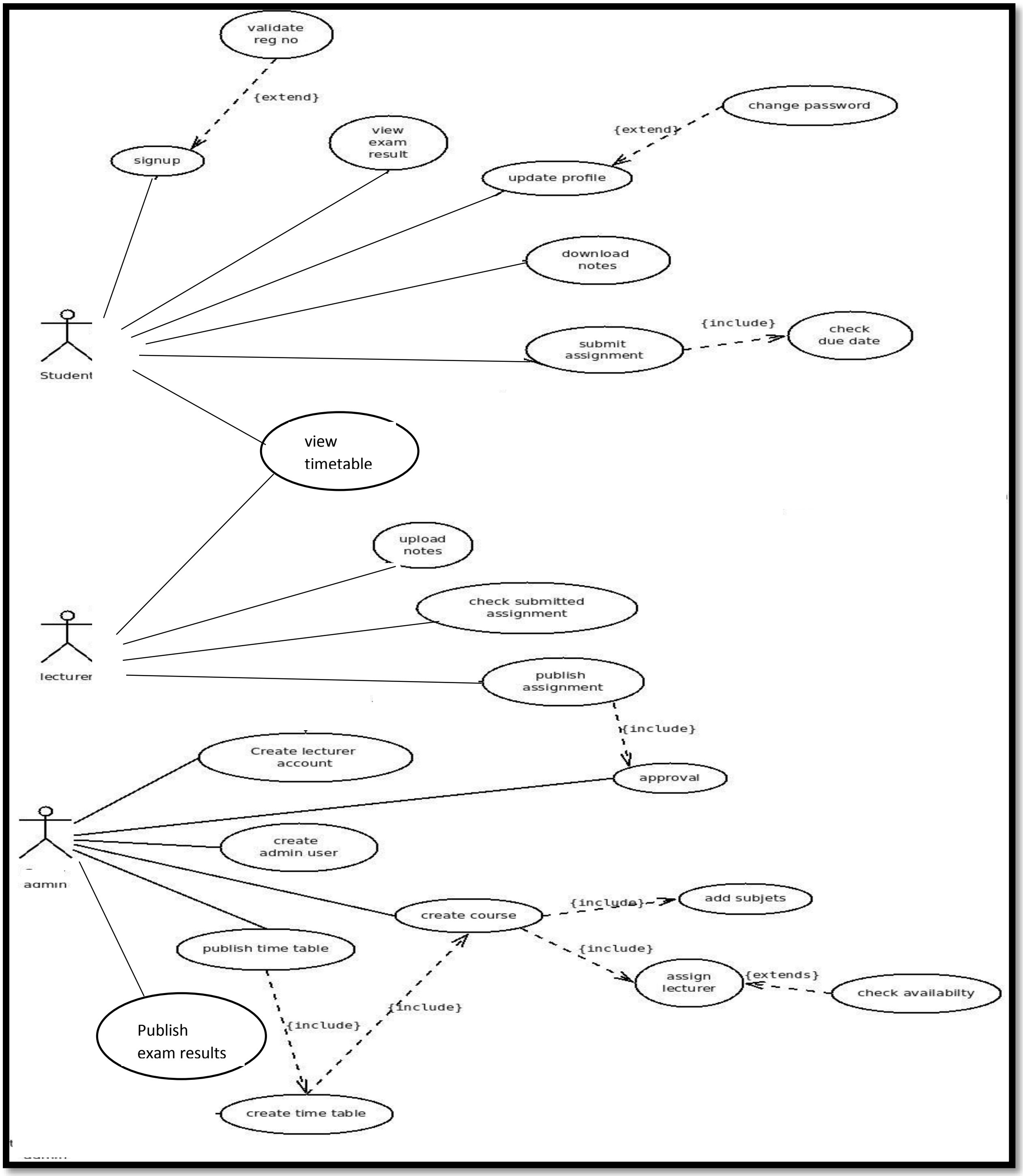


FIGURE 3.2 HIGH LEVEL USE CASE DIAGRAM FOR HNDE STUDENT PORTAL MATTAKKULIYA

3.6.2 CLASS DIAGRAM FOR THE PROPOSED SYSTEM

The class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application.

The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object oriented systems because they are the only UML diagrams which can be mapped directly with object oriented languages.

The class diagram shows a collection of classes, interfaces, associations, collaborations and constraints. It is also known as a structural diagram. [**11**].

Figure 3.3 show the Class Diagram for the HNDE student portal

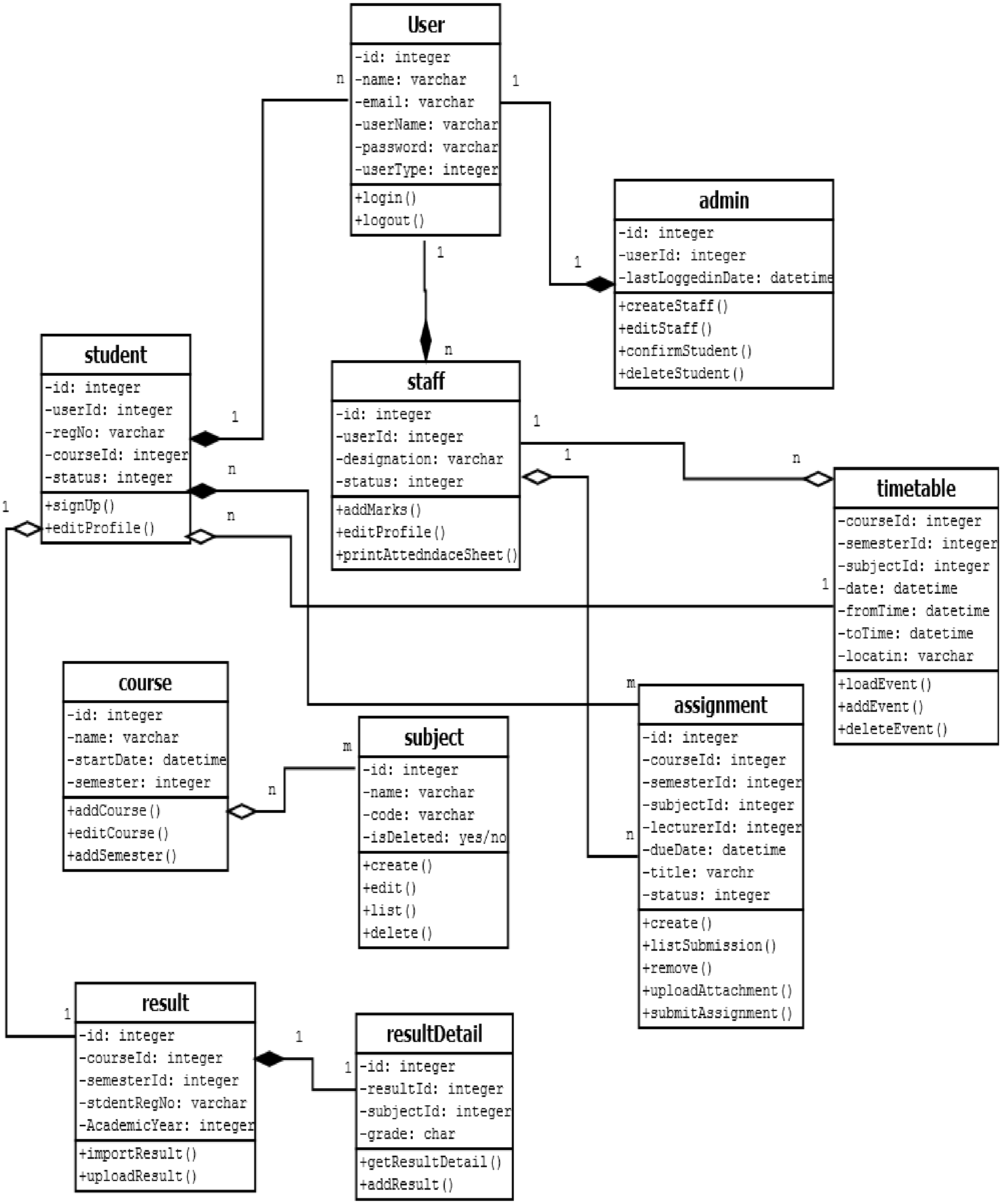


FIGURE 3.3 CLASS DIAGRAM FOR THE STUDENT PORTAL MATTKKULIYA

3.6.3Activity Diagram for the submit assignment

The activity diagram which relates to apply order module is displayed by figure 3.4.

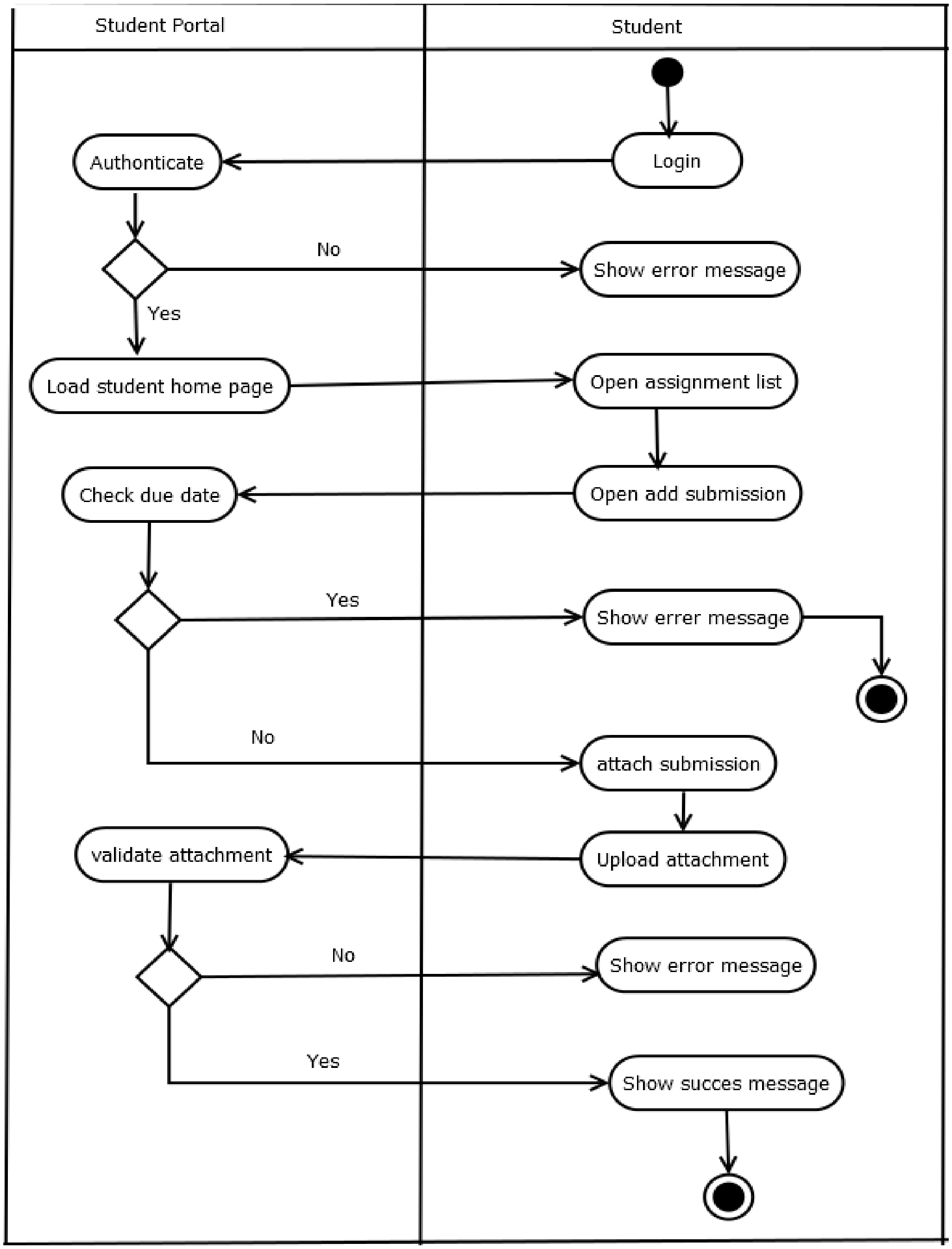


FIGURE 3.4 ACTIVITY DIAGRAM FOR THE SUBMIT ASSIGNMENT

3.6.4 Sequence Diagram for the Assignment module

The sequence diagram which relates to the assignment module is displayed by figure 3.5

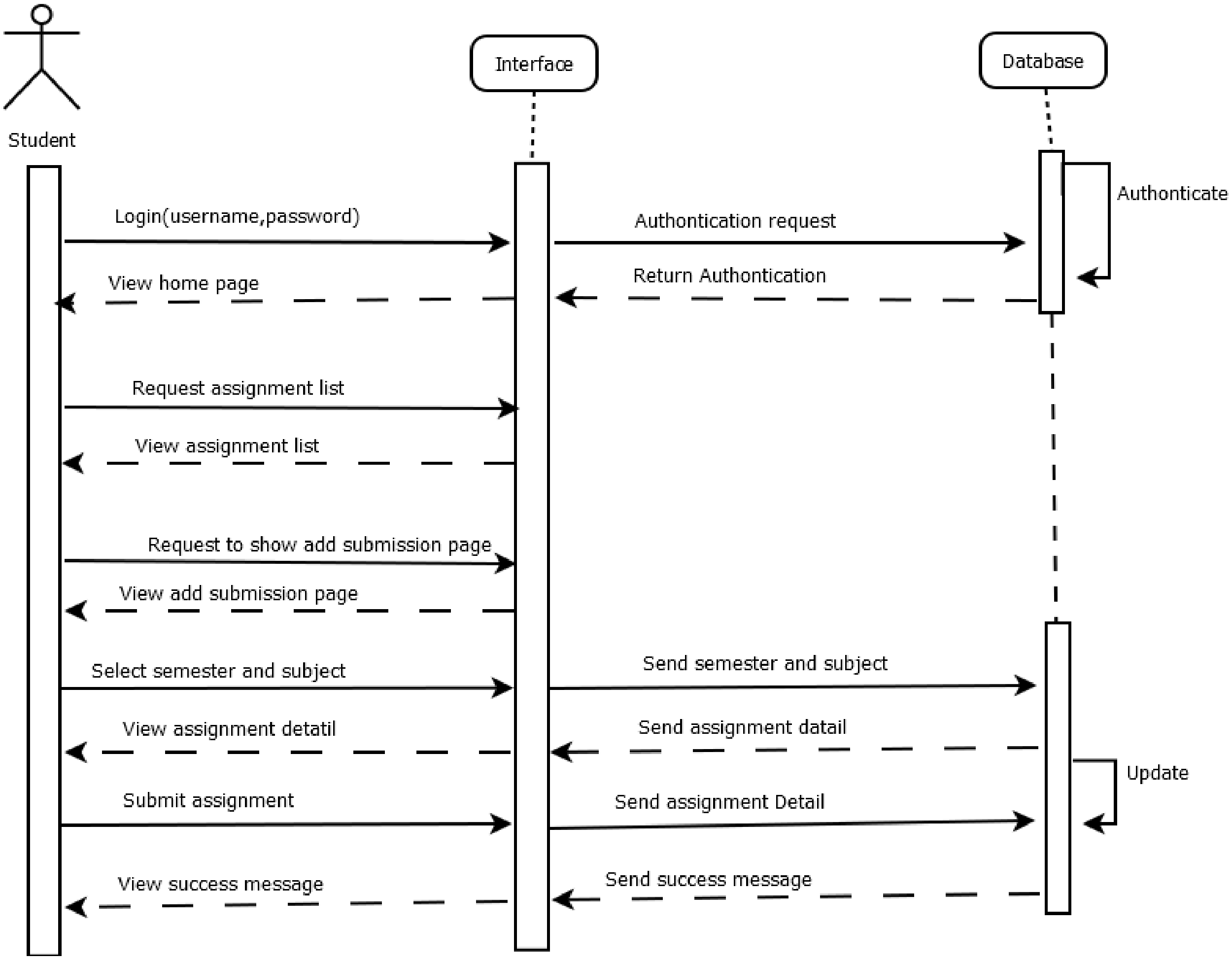


FIGURE 3.5 SEQUENCE DIAGRAM FOR ASSIGNMENT MODULE

3.7 DATABASE DESIGN

3.7.1 Basics of Good Database Design

Certain principles guide the database design process. The first principle is that duplicate information (also called redundant data) is bad, because it wastes space and increases the likelihood of errors and inconsistencies. The second principle is that the correctness and completeness of information is important. If your database contains incorrect information, any reports that pull information from the database will also contain incorrect information. As a result, any decisions you make that are based on those reports will then be misinformed. A good database design is, therefore, one that:

- Divides your information into subject-based tables to reduce redundant data.
- Provides Access with the information it requires to join the information in the tables together as needed.
- Helps support and ensure the accuracy and integrity of your information.
- Accommodates your data processing and reporting needs. **[12]**

3.7.2 Database Normalization

Normalization is the process of organizing data in a database. This includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency.

- First Normal Form (1NF)

Main objective of the 1NF is eliminating the repeating groups and multi valued columns and arranges them in a single table, and defines a primary key for identifying each related attribute.

- Second Normal Form (2NF)

Main objective of the 2NF is eliminating the partial dependencies and creating separate tables and relate tables with a foreign key.

- Third Normal Form (3NF)

Main objective of the 3NF is eliminating the transitive dependencies. **[13]**

3.8 ER DIAGRAM

An Entity Relationship Diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is a component of data. In other words, ER diagrams illustrate the logical structure of databases.

At first glance an entity relationship diagram looks very much like a flowchart. It is the specialized symbols, and the meanings of those symbols, that make it unique. **[14]**

Figure 3.6 is show the high level ER diagram for HNDE student portal mattakkuliya.

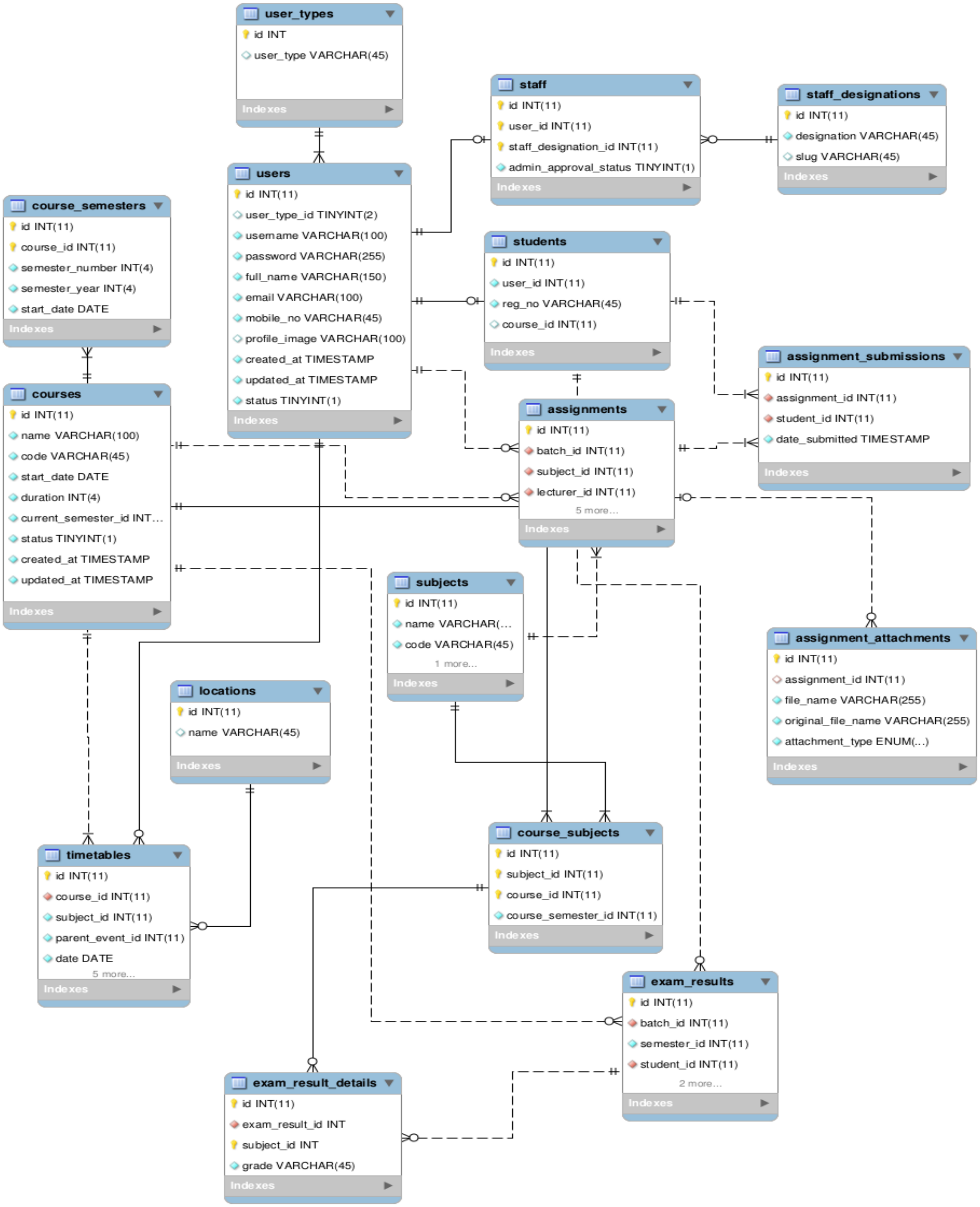


FIGURE 3.6 FIGURE HIGH LEVEL ER DIAGRAM FOR HNDE STUDENT PORTAL MATTAKKULIYA

3.9 USER INTERFACE DESIGNING FOR THE SYSTEM

Interface designing part is a critical part of the overall software designing process. Because many user errors can be occurred, if the interface designing is poor. Good interface designing should be match with the user expectation. That is the reason of criticalness of the user interface designing. When making user interface design decisions, you should take into asses the physical and mental capabilities of the people who use software. Following are the user interface design principles

- User Familiarity

  The interface should use terms and concepts drawn from the experience of the people who use the system

- Consistency

  The interfaces should be consistent in that wherever possible, comparable operations should be activated in the same way

- Minimal surprise

  Users should never be surprised by the behavior of the system

- Recoverability

  The interfaces should include mechanisms to allow users to recover from errors

- User guidance

  The interface should provide meaningful feedback when errors occur and provide help facility

- User Diversity

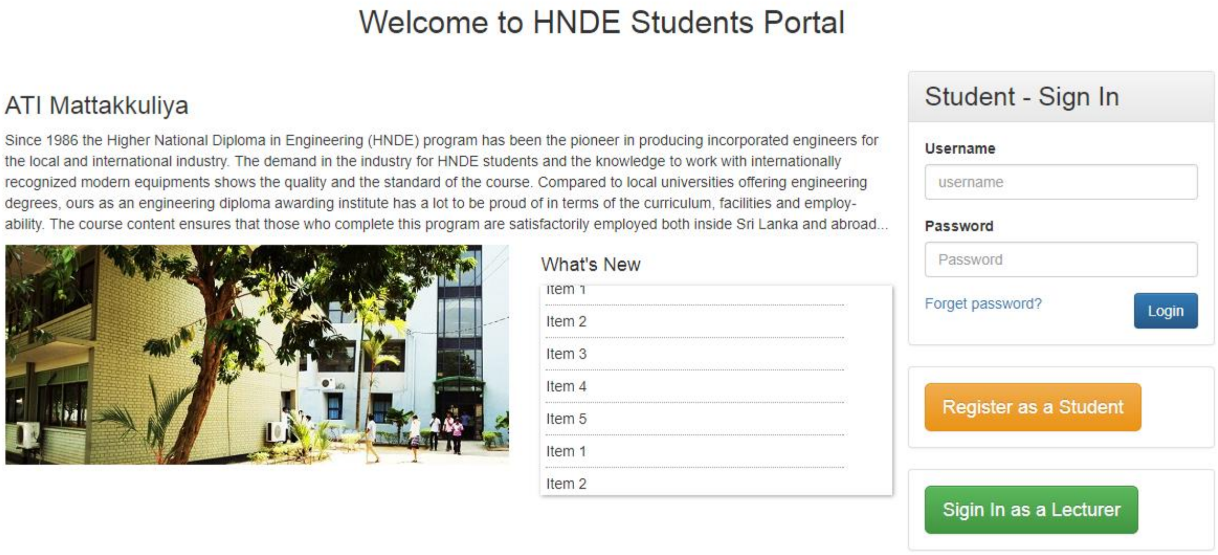
  The interface should provide appropriate interaction facilities for different type of users. [Ian Sommerville, Software Engineering, 8th edition, Addison-Wesley, 2007].

There are several actions were taken in order to ensure the good and user friendly interfaces throughout whole system.

- Eye friendly colors were used for this system.
- Tab indexes were used for easily navigation.
- Enter button of keyboard has used for the enter details instead of mouse click in order to increase the effectiveness of the system.
- Meaningful names and texts were used for button, forms and labels for increase the effectiveness of the system.
- After saving, updating and deleting record, every form should show the successful message
- Before saving, updating, deleting record, system should take permission from the user.
- System has provided meaningful error messages when fields are empty and when user enter invalid value
- System has used combo box, selection form filled in a consistent manner to increase the user friendliness of the system
- Before saving, updating, deleting record, system should take permission from the user. Following figure 3.6 and figure 3.7 Shows the confirmation message which is displaying before deleting record.

3.9. 1 HOME PAGE

Figure 3.4 shows the Home page of HNDE student portal



Welcome to HNDE Students Portal

ATI Mattakkuliya

Since 1986 the Higher National Diploma in Engineering (HNDE) program has been the pioneer in producing incorporated engineers for the local and international industry. The demand in the industry for HNDE students and the knowledge to work with internationally recognized modern equipments shows the quality and the standard of the course. Compared to local universities offering engineering degrees, ours as an engineering diploma awarding institute has a lot to be proud of in terms of the curriculum, facilities and employ-ability. The course content ensures that those who complete this program are satisfactorily employed both inside Sri Lanka and abroad...

FIGURE 3.4 HOME PAGE 1

3.9.2 USER LOGIN INTERFACE

This is a common interface for all users to log into the system. Only authorized users can access into the system. If the invalid user log into the system error messages are popup. Figure 3.5 shows the User login interface

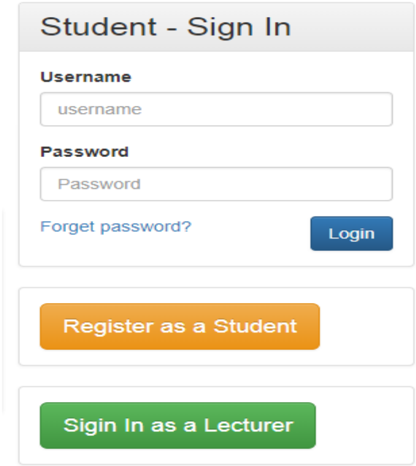
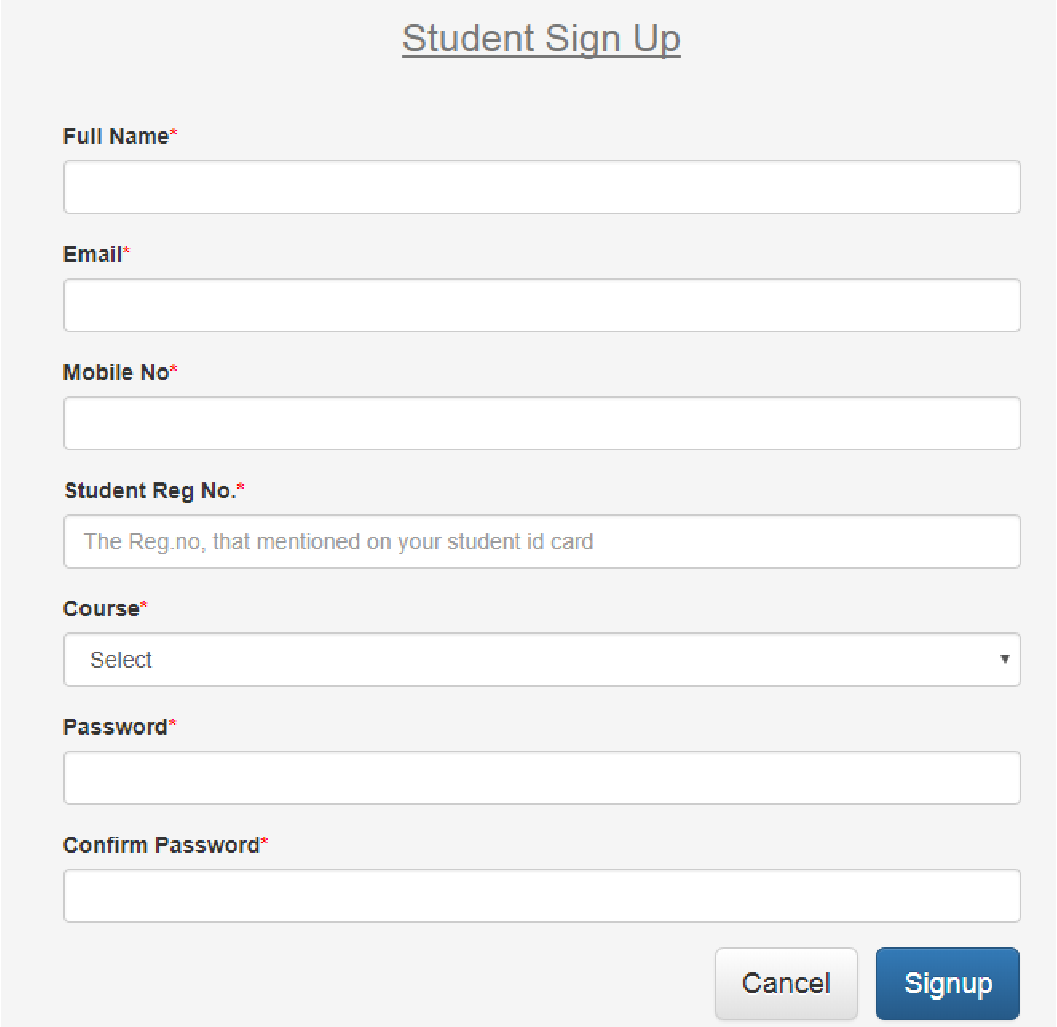


FIGURE 3.5 USER LOGIN INTERFACE

3.9.3 STUDENT SIGNUP

This is an interface for new student to create a user account .After student submit his following detail authorize person want to accept.



Student Sign Up

Full Name*

Email*

Mobile No*

Student Reg No.*

The Reg.no, that mentioned on your student id card

Course*

Select ▾

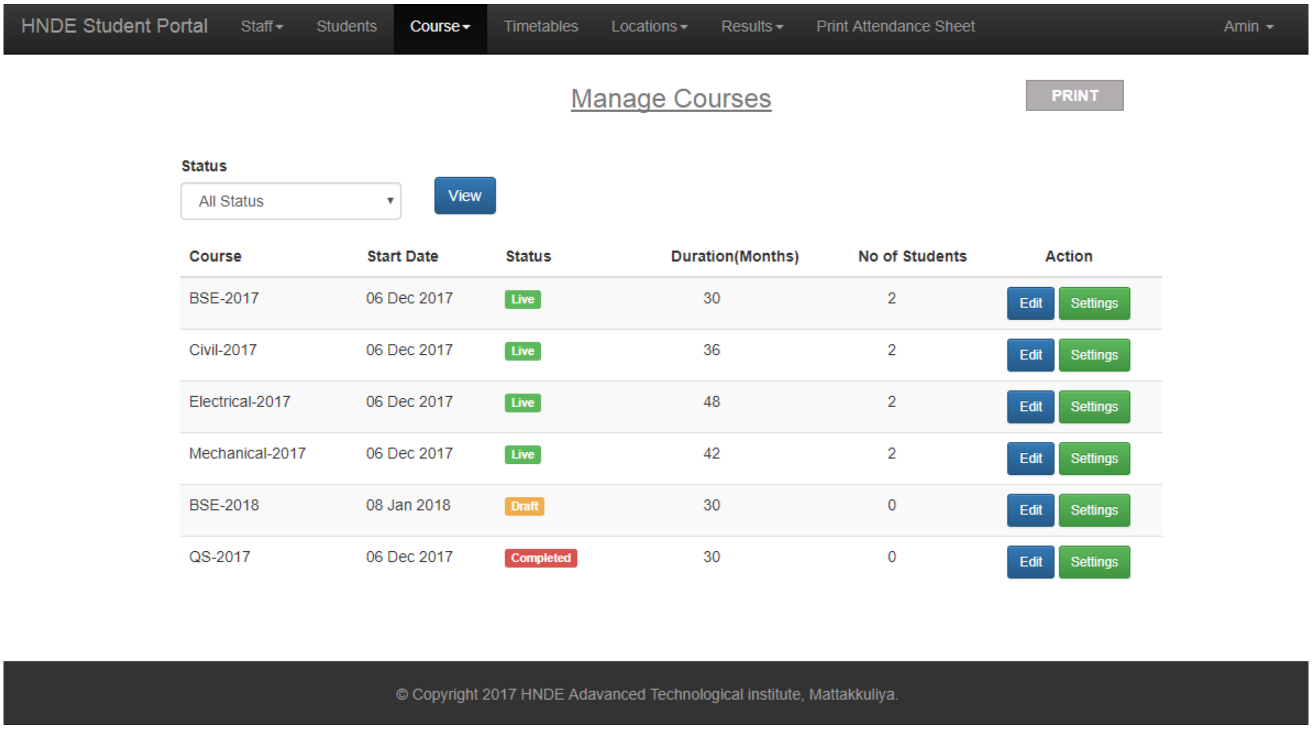
Password*

Confirm Password*

FIGURE 3.6 STUDENT SIGNUP

3.9.4 MANAGE COURSE

Administrator can manage the course. Using this interface can introduce the new course or remove available course from HNDE student portal



Manage Courses

Status: All Status

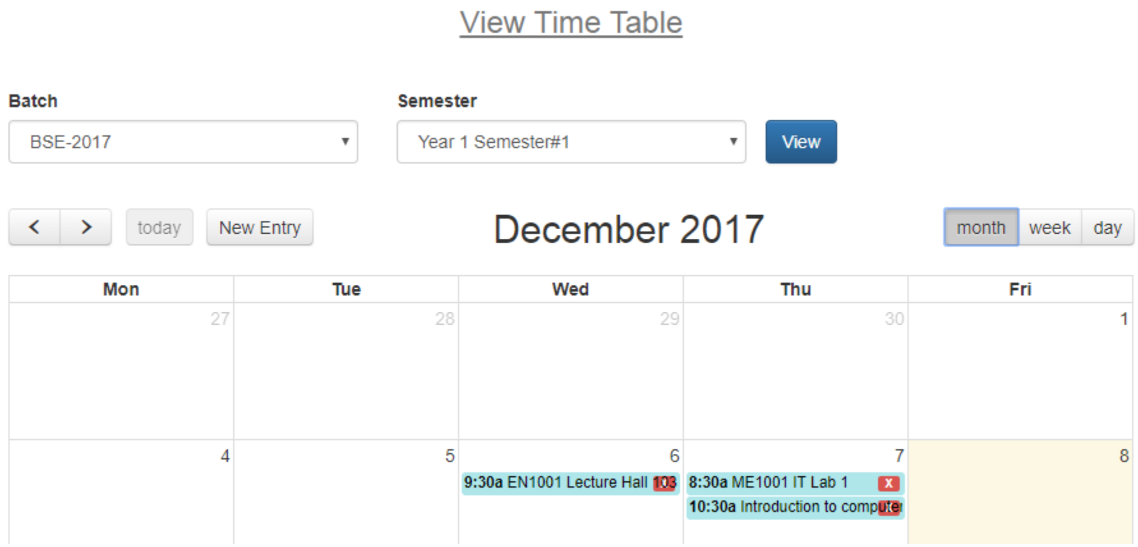
Course	Start Date	Status	Duration(Months)	No of Students	Action
BSE-2017	06 Dec 2017	Live	30	2	Edit Settings
Civil-2017	06 Dec 2017	Live	36	2	Edit Settings
Electrical-2017	06 Dec 2017	Live	48	2	Edit Settings
Mechanical-2017	06 Dec 2017	Live	42	2	Edit Settings
BSE-2018	08 Jan 2018	Draft	30	0	Edit Settings
QS-2017	06 Dec 2017	Completed	30	0	Edit Settings

© Copyright 2017 HNDE Adavanced Technological institute, Mattakkuliya.

FIGURE 3.7 MANAGE COURSE

3.9.5 TIME TABLE

Before start the semester want to upload the timetable from Administrator .Using this interface lectures and student can see their timetable monthly, weekly or daily. When lectures log the system, using this timetable lectures can know which lecture hall want to go relevant time.



View Time Table

Batch: BSE-2017

Semester: Year 1 Semester#1

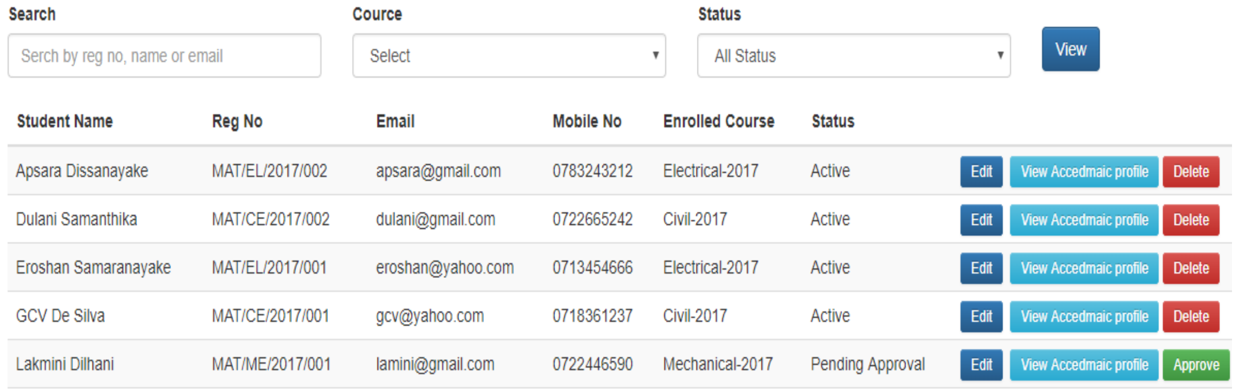
December 2017

Mon	Tue	Wed	Thu	Fri
27	28	29	30	1
4	5	6 9:30a EN1001 Lecture Hall 103	7 8:30a ME1001 IT Lab 1 x 10:30a Introduction to compute	8

FIGURE 3.8 ADD TIMETABLE

3.9.6 MANAGE STUDENT

After the registration of student admin can manage that student using this interface. Admin can filter the student list using this interface and take print out of student list.



Search: Serch by reg no, name or email

Cource: Select

Status: All Status

Student Name	Reg No	Email	Mobile No	Enrolled Course	Status	
Apsara Dissanayake	MAT/EL/2017/002	apsara@gmail.com	0783243212	Electrical-2017	Active	Edit View Accedmaic profile Delete
Dulani Samanthika	MAT/CE/2017/002	dulani@gmail.com	0722665242	Civil-2017	Active	Edit View Accedmaic profile Delete
Eroshan Samaranayake	MAT/EL/2017/001	eroshan@yahoo.com	0713454666	Electrical-2017	Active	Edit View Accedmaic profile Delete
GCV De Silva	MAT/CE/2017/001	gcv@yahoo.com	0718361237	Civil-2017	Active	Edit View Accedmaic profile Delete
Lakmini Dilhani	MAT/ME/2017/001	lamini@gmail.com	0722446590	Mechanical-2017	Pending Approval	Edit View Accedmaic profile Approve

FIGURE 3.9 MANAGE STUDENTS

CHAPTER 04–IMPLEMENTATION

4.1 INTRODUCTION

After completion of the design stage, implementation of the system begins according to what the design stage planed by using adequate tools and techniques. This is an important stage in the software development lifecycle, because this stage builds the system into a working system. This system has been developed as a modularized way and finally integrated all the modules as a complete system. This chapter mainly discusses about the implementation environment, the tools and techniques as well as the reusable components used to implement the system. The major code segments have been included to help the process functionalities of the system in this chapter. Code segmentations which were included in the system with a comment are meant to be used for future improvements.

4.2 SOFTWARE ENVIRONMENT

4.2.1 SUPPORTED LANGUAGES AND TECHNOLOGIES

The system has been developed using the following languages and technologies;

- Sublime Text 3

This software used for the coding of the system. It contains more supportive background when doing implementation like popup code hints.

- Adobe Photoshop CS5.

Photo retouching, interface designing had done by using this software.

- Software Ideas Modeler.

Diagram designing had done by using this software.

- PHP

This is a server side object oriented scripting language which used when developing the system.

- MySQL

This is the Database software which used to handle the database of the system.

- HTML

This is the basic web related language and it helps to keep the system structure clear and conscious.

- JavaScript

This is a Scripting language and this helps for client side validation and to dynamic the system components.

- Ajax

Ajax is a JavaScript based technology and it supports updating the system components without refreshing the whole system page.

- CSS

This is used for the styling of the system.

- JSON

JSON is syntax for storing and exchanging text information. Much like XML. JSON is smaller than XML, and faster and easier to parse.

- JQuery

It is a JavaScript library and used as reusable component when developing.

- Sublime Text Editor.

4.3 HARDWARE ENVIRONMENT

This system has been developed using a computer with the following configuration,

- Intel® Core™ 2 Duo CPU 2.00GHz
- 2 GB RAM
- 300 GB Hard disk

The system has an extended communication feature via mail. In order to implement this feature securely in real industry.

4.4 MAJOR CODE SEGMENTS

In the actual environment, the system has the three tier architecture. Therefore the database server is installed on one level whereas the web server, mail server is placed on a separate level. The database can be accessed only via the application server. These three tiers are implemented on the same computer and accessed via local-host only for the testing purposes. Following steps guide to setup the system in the actual environment.

Step 1: Database server setup XAMPP version should be installed on the computer dedicated for the database server. The reason behind installing XAMPP instead of separate MySQL server is to enable running phpMyAmin for more convenience. Please refer the section for database installation of Appendix :A-System documentation for configuring the database.

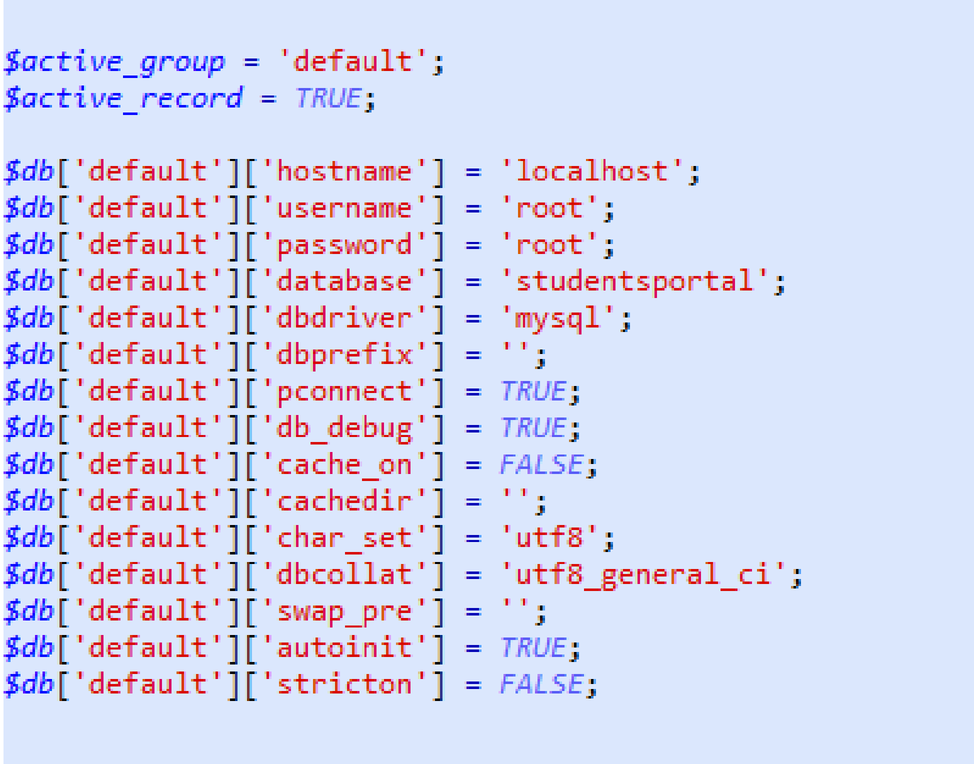
Step 2: Web server setup The computer dedicated for the web server also should consist of a XAMPP installation. The Mercury mail server which is included in the XAMPP pack should also be configured to the local-host. Make sure that the user names, passwords and ports are the same as in the application.

Step 3: Setting up the Interest updating schedule task The script for interest updates (interestUpdate.php) should be added in the web server as a scheduled task and make sure the task is set to run at midnight every day and on every system boot up.

- **Database Connection Page**

Database Connection Page

Every web system as well as stand-alone system always deals with a database. If something is added, deleted or updated in the system, the system always connects with the related database. Code segment given below how to set database configurations



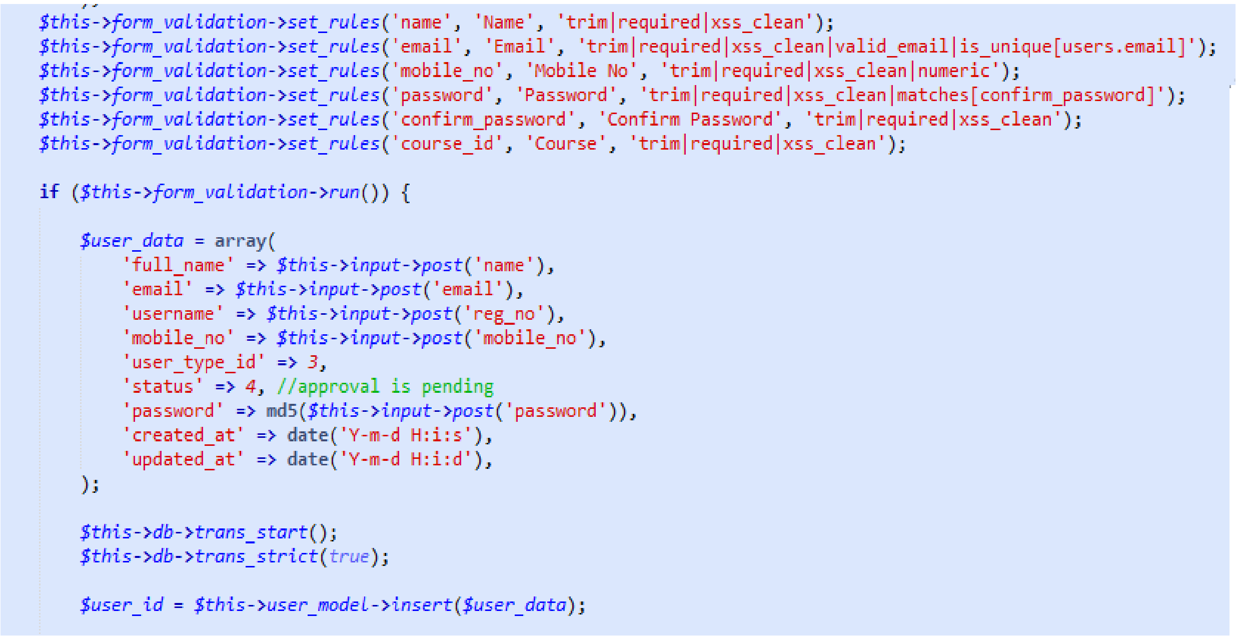
```
$active_group = 'default';
$active_record = TRUE;

$db['default']['hostname'] = 'localhost';
$db['default']['username'] = 'root';
$db['default']['password'] = 'root';
$db['default']['database'] = 'studentsportal';
$db['default']['dbdriver'] = 'mysql';
$db['default']['dbprefix'] = '';
$db['default']['pconnect'] = TRUE;
$db['default']['db_debug'] = TRUE;
$db['default']['cache_on'] = FALSE;
$db['default']['cachedir'] = '';
$db['default']['char_set'] = 'utf8';
$db['default']['dbcollat'] = 'utf8_general_ci';
$db['default']['swap_pre'] = '';
$db['default']['autoinit'] = TRUE;
$db['default']['stricton'] = FALSE;
```

FIGURE 4.1 DATABASE CONNECTION

- **Signing & create user session**

Controller code

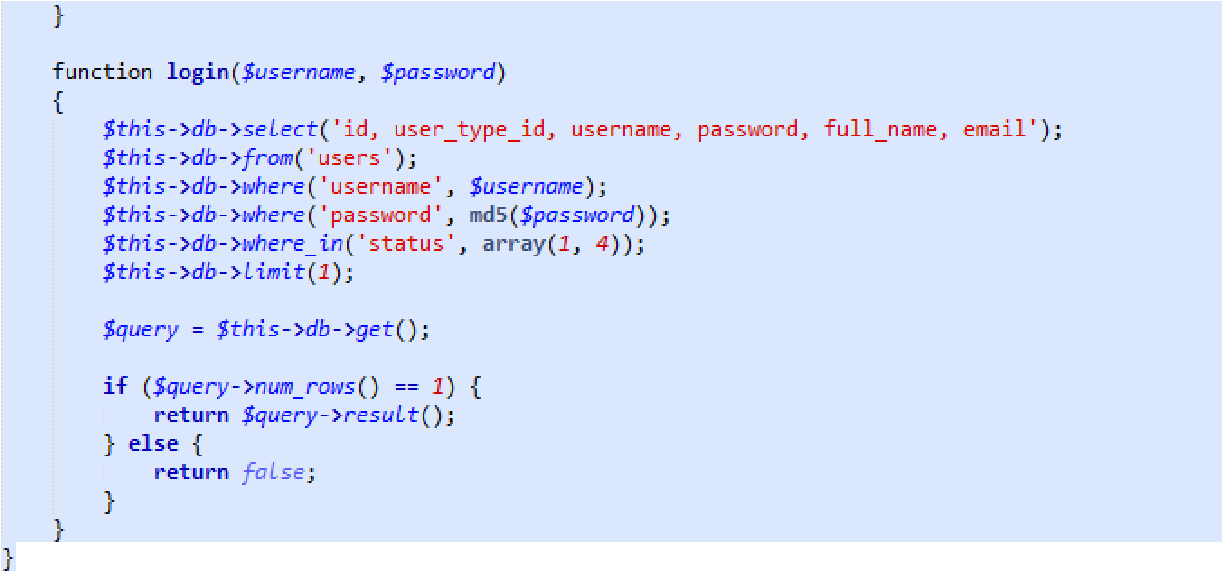
If user is entered user name & password correctly then above code is creating a session. user_id & user_type_id are the major attributes on the user session. user_id uses to check user is logged in and user_type_id uses to identify role of the logged in user. User type id is resides on the user table in the database for each and every user.



```
$this->form_validation->set_rules('name', 'Name', 'trim|required|xss_clean');
$this->form_validation->set_rules('email', 'Email', 'trim|required|xss_clean|valid_email|is_unique[users.email]');
$this->form_validation->set_rules('mobile_no', 'Mobile No', 'trim|required|xss_clean|numeric');
$this->form_validation->set_rules('password', 'Password', 'trim|required|xss_clean|matches[confirm_password]');
$this->form_validation->set_rules('confirm_password', 'Confirm Password', 'trim|required|xss_clean');
$this->form_validation->set_rules('course_id', 'Course', 'trim|required|xss_clean');

if ($this->form_validation->run()) {

    $user_data = array(
        'full_name' => $this->input->post('name'),
        'email' => $this->input->post('email'),
        'username' => $this->input->post('reg_no'),
        'mobile_no' => $this->input->post('mobile_no'),
        'user_type_id' => 3,
        'status' => 4, //approval is pending
        'password' => md5($this->input->post('password')),
        'created_at' => date('Y-m-d H:i:s'),
        'updated_at' => date('Y-m-d H:i:d'),
    );

    $this->db->trans_start();
    $this->db->trans_strict(true);

    $user_id = $this->user_model->insert($user_data);
```

FIGURE 4.2 SIGNING & CREATE USER SESSION

- Related model code of the login process (user_model.php)

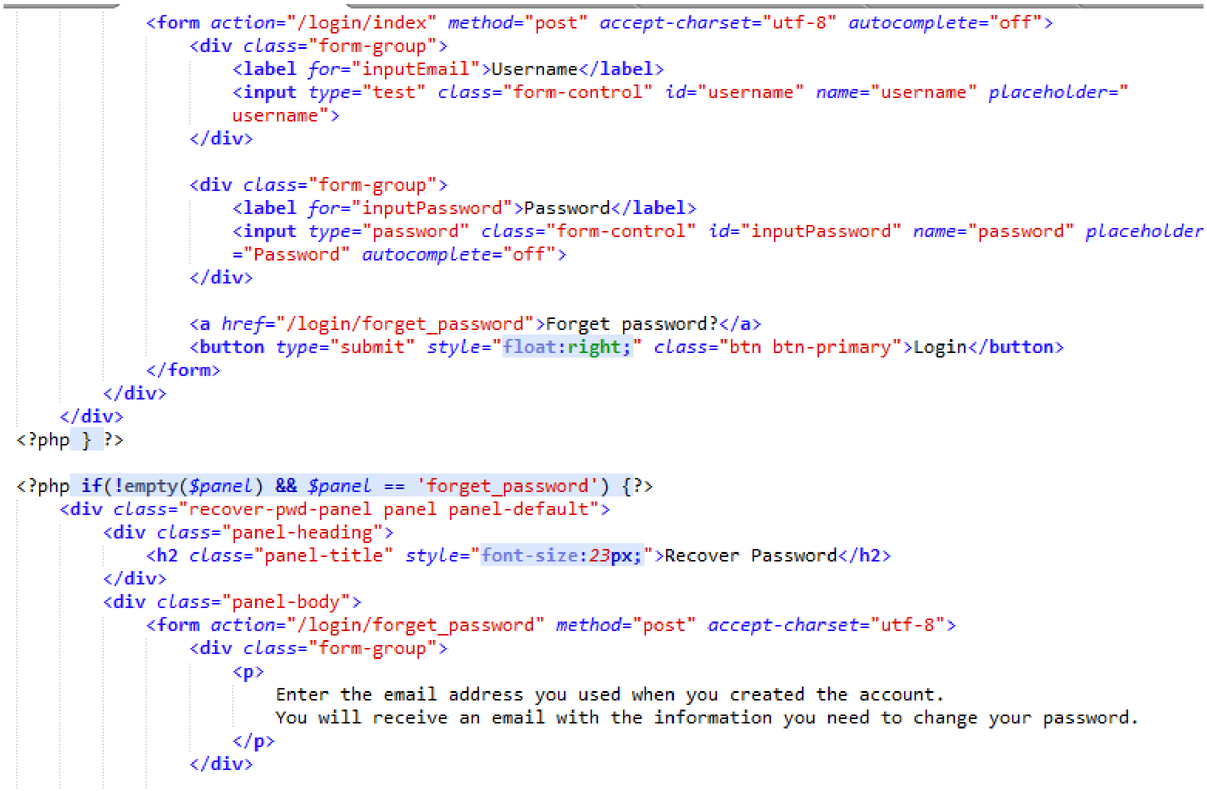


```
    }

    function login($username, $password)
    {
        $this->db->select('id, user_type_id, username, password, full_name, email');
        $this->db->from('users');
        $this->db->where('username', $username);
        $this->db->where('password', md5($password));
        $this->db->where_in('status', array(1, 4));
        $this->db->limit(1);

        $query = $this->db->get();

        if ($query->num_rows() == 1) {
            return $query->result();
        } else {
            return false;
        }
    }
}
```

FIGURE 4.3 RELATED MODEL CODE OF THE LOGIN PROCESS

Based on logged in user's user_type_id they are redirecting to their own home page e.g if a user has student login, that user will redirect to student home page once logged in successfully.

- View code related to login process (html code)

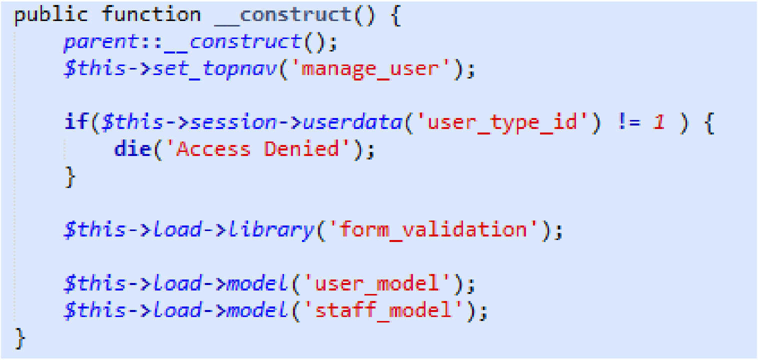


```
            <form action="/login/index" method="post" accept-charset="utf-8" autocomplete="off">
                <div class="form-group">
                    <label for="inputEmail">Username</label>
                    <input type="test" class="form-control" id="username" name="username" placeholder="
                    username">
                </div>

                <div class="form-group">
                    <label for="inputPassword">Password</label>
                    <input type="password" class="form-control" id="inputPassword" name="password" placeholder
                    ="Password" autocomplete="off">
                </div>

                <a href="/login/forget_password">Forget password?</a>
                <button type="submit" style="float:right;" class="btn btn-primary">Login</button>
            </form>
        </div>
    </div>
<?php } ?>

<?php if(!empty($panel) && $panel == 'forget_password') {?>
    <div class="recover-pwd-panel panel panel-default">
        <div class="panel-heading">
            <h2 class="panel-title" style="font-size:23px;">Recover Password</h2>
        </div>
        <div class="panel-body">
            <form action="/login/forget_password" method="post" accept-charset="utf-8">
                <div class="form-group">
                    <p>
                        Enter the email address you used when you created the account.
                        You will receive an email with the information you need to change your password.
                    </p>
                </div>
```

FIGURE 4.4 VIEW CODE RELATED TO LOGIN PROCESS

- User Authorization:

Authorization is check right user has right privileges. Following code segment depict how this system handles authorization. This part of code in the admin_mange_course controller.



```
public function __construct() {
    parent::__construct();
    $this->set_topnav('manage_user');

    if($this->session->userdata('user_type_id') != 1 ) {
        die('Access Denied');
    }

    $this->load->library('form_validation');

    $this->load->model('user_model');
    $this->load->model('staff_model');
}
```

FIGURE 4.5 USER AUTHORIZATION

4.5 REUSED MODULES

Please refer to the appendix-F for important codes

Following libraries/components have been used when implementing the system to take advantage of re-using existing libraries to make development more efficiently and make the system more useable.

BOOTSTRAP COMPONENTS

Bootstrap is a free and open-source front-end library for designing websites and web applications. It contains HTML- and CSS-based design templates for typography, forms, buttons, navigation...etc.

I have been used,

- bootstrap layout - to make the system UI responsive, attractive and better accessibility.
- date-picker - select a date easily in date-time filed.
- modal box - it provides attractive popups. I have used modal boxes where I've used popup windows
- navigation - I have used bootstrap navigation to implement the main menu of the system.
- form - all forms are developed using bootstrap-form so I could able to develop more attractive & responsiveness forms using bootstrap-form.

JQUERY

JQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML. It is free, open-source software.

I have been used following jQuery components to develop the system.

- JQuery Ajax

AJAX stands for Asynchronous JavaScript and XML. AJAX is a new technique for creating better, faster, and more interactive web applications with the help of XML, HTML, CSS, and Java Script.

I have been used to Ajax as much as possible to communicate with back-end of the system. So I could able to increase usability of the system by minimizing page reloads.

- JQuery events

I have used jQuery events (e.g. click, change, focus, document ready,...etc). I got the advantage of handle client side events more efficiently and develop

- PHP Mailer

I have used php mailer library to send email through the system. I have developed a wrapper class so I could able to re-use php mailer library throughout the system. I just need to call a single method when I need to add send email feature.

CHAPTER 05 – EVALUATION

5.1 INTRODUCTION

The system evaluation of performance, from the perspectives of both developers and users, of complex systems of hardware and software. Modern computer-based information systems have become increasingly complex because of networking, distributed computing, distributed and heterogeneous databases, and the need to store large quantities of data. People are relying increasingly on computer systems to support daily activities. When these systems fail, significant breakdowns may ensue therefore, in order to prevent these kinds of unnecessary failures, a good system evaluation and testing has to be introduced.

5.2. TESTING

Planning a test case is a very important aspect for developing system as well as for the completed system. Test plan should have the ability to test the functionality of the overall system. By properly testing a system, it can identify the errors which generate from the system and can correct them. The implemented system was tested using different test cases.

Since the development started, the test plan continued by testing the system units. After completing a system unit it was completely tested to identify whether it can function according to expectations. Therefore, early detection of the errors was helped by this testing stage.

After performing the system unit testing, next integration testing was done, and this can identify the errors and the required functionality of the units after integration. System testing was performed as the final stage for the completely developed web based application to check the functionality. This performed at the client's environment with client's original data. . **[13]**

5.3 TEST PLAN

The test Plan of the student portal describes the testing strategies and the approaches to testing. Quality assurance will be used to validate the quality of this system prior to its using in the real environment. It also contains various resources required for the successful completion of this project. When coding the student portal, a unit testing was carried out for the functions and procedures to ensure that they behave as intended. This is a code level testing which tests the units of codes. An integration testing has been done in order to test the modules (integration of units of code) of the system. This is essential when using different kinds of implementing technologies together. It ensures that all the integrated components are working cooperatively for the desired outcome.

After the completion of unit and integration testing the whole system has been tested properly. The system testing guarantees the standard of the system. A system testing was conducted using a similar environment where the system will be practically used. After completing the system a confirmation has been obtained from the clients through a user acceptance testing whether their requirements are fulfilled by the developed system.

Further the test plan includes two levels for implementing the "Black box" and "White box" testing methods. The Black box testing focuses on the functions and behaviors of the system. This testing does not need any knowledge on the internal structure of the system. The White box testing focuses the internal structure and logics of the system. This is a code level testing method and any malfunctioning units can be easily detected.

5.4 SYSTEM TEST CASES

A properly planned test case should have the ability to verify the relevant system component functionality. Therefore, to verify all the system functions there should be properly planned test cases for each and every function. Some test cases designed for major system modules are as follows:

5.4.1 Test Cases for User Authentication

The test cases and results of the user authentication are displayed by Table 5.1

Test Number	Test Description	Step to Test	Expected Result
1	Validate User Input Details.	Enter correct username Enter correct password	Successfully Login into the system
2	Validate User Input Details.	Enter correct username Enter incorrect password	Display error message “Incorrect username or password”
3	Validate User Input Details.	Enter incorrect username Enter correct password	Display error message “Incorrect username or password”
4	Validate User Input Details.	Enter incorrect username Enter incorrect password	Display error message “Incorrect username or password”

TABLE 5.1 USER AUTHENTICATION TEST CASES

5.4.2 Test Cases for Student Signup

The test cases and results of the student signup are displayed by Table 5.2

Test Number	Test Description	Step to Test	Expected Result
1	Validate Create Account	Enter existing Student Registration number	Display Message “Already inserted Registration Number”.
2	Validate Create Account	Try to create account without entering the Registration Number	Display error message “Please enter the Registration Number”
3	Validate Create Account	Try to create account without entering the Full name	Display error message “Please enter the Full name”
4	Validate Create Account	Try to create account without entering the Email	Display error message “Please enter the Email ”

5	Validate Create Account	Try to create account without entering mobile number	Display error message “Please enter the mobile number ”
6	Validate Create Account	Try to create account without selecting the Course	Display error message “Please select the course ”
7	Validate Create Account	Try to create account without enter the password	Display error message “Please enter the password”
8	Validate Create Account	Try to create account without enter the confirm password	Display error message “Please enter the confirm password ”
9	Validate Create Account	Confirm password not match with the password	Display error message “Confirm password not match with the password ”

TABLE 5.2 STUDENT SIGNUP TEST CASES

5.4.3 TEST CASES FOR COURSE MODULE

The test cases and results of the course module are displayed by Table 5.3

Test Number	Test Description	Step to Test	Expected Result
1	Validate course Information adding	Add Information with relevant details	Display Message “Successfully inserted course details”
2	Validate course Information adding	Try to add information by keeping all fields in empty	Popup the error message of the first field.
3	Validate course Information adding	Try to create course without selecting course category	Display error message “Please select course category”
4	Validate course Information adding	Try to create course without selecting course year	Display error message “Please select course year ”

5	Validate course Information adding	Try to create course without selecting course starting date	Display error message “Please course starting date ”
6	Validate course Information adding	Try to create course without entering course duration	Display error message “Please enter course duration ”
7	Validate course Information adding	Try to create course without adding semesters	Display error message “Please add semesters”
8	Validate course Information adding	Try to create course without adding subjects	Display error message “Please add subjects ”
9	Delete a semester	Choose semester from the list and Click Delete button	Popup the Confirm dialog box” Are you sure to Delete this semester?”
10	Delete a subject	Choose subject from the list and Click Delete button	Popup the Confirm dialog box” Are you sure to Delete this subject?”
11	Update a course	Choose course from the list, click edit button and Update details	Display Message “Product Successfully Updated”
12	Validate subject Information adding	Add Information with relevant details	Display Message “Successfully inserted subject details”
13	Validate subject Information adding	Try to add information by keeping all fields in empty	Popup the error message of the first field.
14	Validate subject Information adding	Try to create subject without selecting course category	Display error message “Please select course category”
15	Validate subject Information adding	Try to create subject without enter subject code	Display error message “Please enter subject code”
16	Validate subject Information adding	Try to create subject without selecting lecturer	Display error message “Please select lecturer”

TABLE 5.3 COURSE MODULE TEST CASES

5.4.4 Test Cases for timetable module

The test cases and results of the timetable module are displayed by Table 5.4

Test Number	Test Description	Step to Test	Expected Result
1	Validate timetable new entry	Add timetable with relevant details	Display Message “Successfully inserted timetable new entry”
2	Validate timetable new entry	Try to add timetable by keeping all fields in empty	Display the error message of the first field.
3	Validate timetable new entry	Add timetable without selecting subject	Display error message “Please select subject”
4	Validate timetable new entry	Add timetable without selecting lecturer	Display error message “Please select lecturer”
5	Validate timetable new entry	Add timetable without selecting location	Display error message “Please select location”
6	Validate timetable new entry	Add timetable without selecting date	Display error message “Please select date”
7	Validate timetable new entry	Add timetable without selecting start time and end time	Display error message “Please select start time and end time”
8	Validate timetable new entry	Select lecturer who is having lecture in same time	Display error message “Select lecturer is not available in this time period”
9	Validate timetable new entry	Select location where is having lecture/practical in same time	Display error message “Select lecture hall /lab is not available in this time period”
10	Validate timetable new entry	Select time period is already having lecture/practical	Display error message “Already having lecturer or practical in this time period”

TABLE 5.4 TIMETABLE MODULE TEST CASES

5.4.5 Test Cases for assignment module

The test cases and results of the assignment module are displayed by Table 5.5

Test Number	Test Description	Step to Test	Expected Result
1	Validate create assignment	Try to create assignment by keeping all fields in empty	Display the error message of the first field.
2	Validate create assignment	Try to create assignment without selecting course	Display error message "Please select course"
3	Validate create assignment	Try to create assignment without selecting subject	Display error message "Please select subject"
4	Validate create assignment	Try to create assignment without enter title	Display error message "Please enter title"
5	Validate create assignment	Try to create assignment without selecting due date	Display error message "Please select due date"
6	Validate create assignment	Try to create assignment without enter description	Display error message "Please enter description"
7	Validate create assignment	Try to create assignment without attach file	Display error message "No chosen file attach"
8	Validate create the repeat assignment	Tick on the is repeat assignment button	Display the subject list and want to select again which is repeat
9	Update a Product	Choose assignment from the list, click edit and Update details	Display Message "Product Successfully Updated"

TABLE 5.5 ASSIGNMENT MODULE TEST CASES

5.4.6 TEST CASES FOR RESULT MODULE

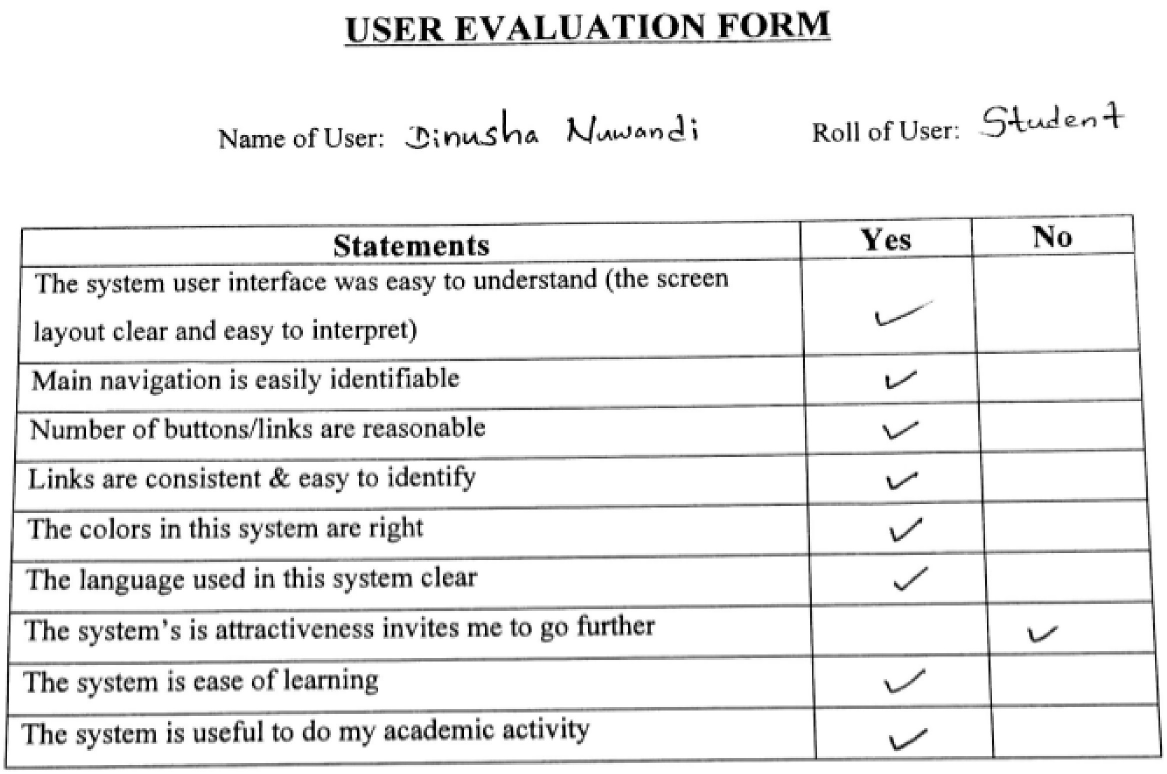
The test cases and results of the results module are displayed by Table 5.6

Test Number	Test Description	Step to Test	Expected Result
1	Validate import(add) results	Try to import(add) results by keeping all fields in empty	Display the error message of the first field.
2	Validate import results	Try to add results without selecting course	Display error message “Please select course”
3	Validate import results	Try to add results without selecting semester	Display error message “Please select semester”
4	Validate import results	Try to add results without attach file	Display error message “No chosen file attach”
5	Validate print results	Select course and semester ,click print button	Display pdf file with results”

TABLE 5.6 RESULTS MODULE TEST CASES

5.5 USER EVALUATION

Normally user evaluation is done by selecting different users of the system. In this student portal three users with different privileges. User evaluation questionnaire was given to target population and results has summarized. Following figure 5.1 shows user evaluation questionnaire.



USER EVALUATION FORM

Name of User: Dinusha Nuwandi Roll of User: Student

Statements	Yes	No
The system user interface was easy to understand (the screen layout clear and easy to interpret)	✓	
Main navigation is easily identifiable	✓	
Number of buttons/links are reasonable	✓	
Links are consistent & easy to identify	✓	
The colors in this system are right	✓	
The language used in this system clear	✓	
The system's is attractiveness invites me to go further		✓
The system is ease of learning	✓	
The system is useful to do my academic activity	✓	

FIGURE 5.1 USER EVALUATION FORM

CHAPTER 06–CONCLUSION

6.1 INTRODUCTION

The World is running on web communication. Therefore, every Government sectors develop their information technology flank as well as private sectors. ATI Mattakuliya has not used any student portal or learning management system. Therefore client decided to create a web site for acquire the details about institute, online submission system. Team can glad to predicate our perseverance became success within project time duration. The testing purpose is 100% present accurate. Team can provide much secure authentication system via this site. Additionally, Student can get soon their exam results anywhere in the world.

In this project, provides user authentication system. According to this system users have privileges to access the functions. Project team identified four main users in the current system, they are administrator, lecturer, Students and staff. They can simply log to the system via username and password.

Guest user can simply identify the programs, courses, messages and put any comment in his mind. These comment messages are simply viewed by administrator.

Lecturer can upload 10 MB files per month. Student can download them. It is the easy to sharing the resources efficiency.

Student and Other authenticated users consider the NOTICE and NEWS displays. It is indicated very useful and temporal event will be held future.

6.2 LESSONS LEARNT

This developed system does not merely fulfill the requirement of the final year of the Degree program; however it assists me to practically apply the knowledge learnt throughout the past three years. When assigning the project proposal, I did not have much of an idea on how to carry out the project. When progressing through step by step according to the guideline provided by the university, I gained a valuable knowledge on how to do a successful professional system development project.

By doing the development process according to a schedule, I learnt how to do my day-to-day activities by managing time efficiently.

The implementation phase was the toughest and most interesting phase of the project, as it allowed me to try out practically the academic knowledge that I have gained on programming languages such as PHP, Java Scripts, CSS and many more development tools and techniques. Further I learnt configuring E-mail servers according to the system.

Writing the dissertation was another interesting task of the project. It provided me with lessons on how to write a report in a professional manner. It helped me to develop my skills on writing and designing technical reports.

6.3 FUTURE ENHANCEMENT

This proposed system was developed within a time constraint and functionalities which were agreed by the client. Following are the some of the future enhancement of the system

- Introducing this student portal as part of HNDE web site
- Try to insert the upcoming events of ATI mattakkuliya
- Link with the library system
- Introducing online payments through the web

REFERENCES

[1]:http://encyclopedia2.thefreedictionary.com/Systems+Analysis+Definition

[2]: https://my.londoninternational.ac.uk/

[3]:https://studentportal.gu.se/english;jsessionid=xxd1j9tl68qa1505l4dxqw?skipSSOCheck=true&referer=https%3A%2F%2Fwww.google.lk%2F

[4]: http://www.requirements.com/nonfunctionalrequirements/tabid/91/default.aspx

[5]:http://infolab.stanford.edu/~burback/watersluice/node11.html

[6]: https://en.wikipedia.org/wiki/Rational_Unified_Process

[7]: https://en.wikipedia.org/wiki/Rational_Unified_Process#Inception_phase

[8]:https://en.wikipedia.org/ /Rational_Unified_Process#Elaboration_phase_.28Ortner.29

[9]: https://en.wikipedia.org/wiki/Rational_Unified_Process#Construction_phase

[10]: https://en.wikipedia.org/wiki/Rational_Unified_Process#Transition_phase

[11]: https://www.tutorialspoint.com/uml/uml_class_diagram.htm

[12]:http://office.microsoft.com/en-us/access-help/database-design-basics

[13]: https://www.smartdraw.com/entity-relationship-diagram/

[14]:https://support.microsoft.com/en-us/kb/283878

[15]: http://www.testingexcellence.com/types-of-software-testing-complete-list/

APPENDIX A-SYSTEM DOCUMENTATION

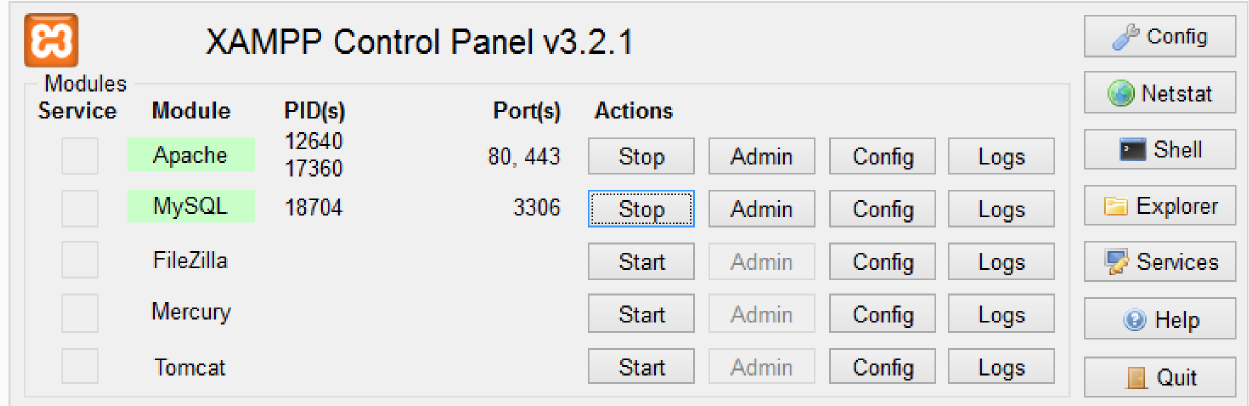
System Manual

Step1:-Install the XAMPP

Step2:-Run the XAMPP

Step3:-Start the apache & MySql

Figure A.1 shows how to start the apache and MySql



Service	Module	PID(s)	Port(s)	Actions			
□	Apache	12640 17360	80, 443	Stop	Admin	Config	Logs
□	MySQL	18704	3306	Stop	Admin	Config	Logs
□	FileZilla			Start	Admin	Config	Logs
□	Mercury			Start	Admin	Config	Logs
□	Tomcat			Start	Admin	Config	Logs

Figure A. 1 Start the apache & MySql

Step 4:- Open your web browser and type http://localhost/ or http://127.0.0.1/ in the URL and press Enter. If you have installed Xampp properly, you could now see the xampp home page on your browser window.

- Select phpMyAdmin tool from the left side panel of the Xampp home page In the phpMyAdmin
  window click in the text field named "Create new database" and type "studentsportal" and press „Create" button. A new database named "studentsportal" will be created.

- Now click on the “Import” tab from the tabs located in the top of the window and click on

  the “Chose file” button located next to the “Location of the text file” option in “File to import” section.   It will give you a browsing window.

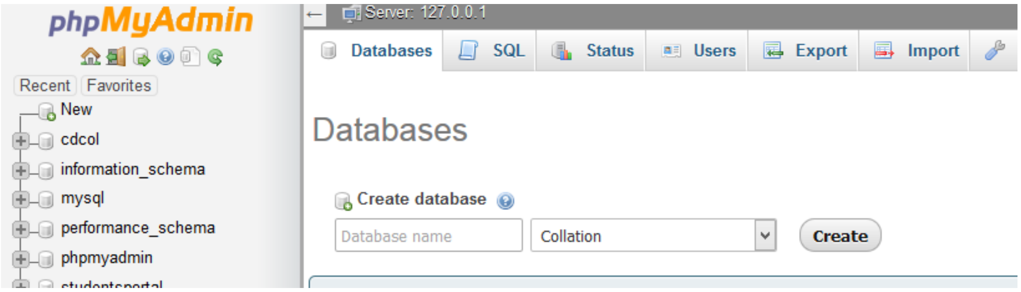


FIGURE A. 2: CREATE NEW DATABASE

SYSTEM INSTALLATION

- Browse the studentsportal CD, Locate the folder “studentsportal” from the path: X:\ studentsportal (Note: X is your CD-ROM drive letter).
- Copy the entire folder and paste it in the following location

C:\xampp\htdocs\student-portal

APPENDIX B –DESIGN DOCUMENTATION

USE CASE DIAGRAMS AND DESCRIPTION

Figure B.1 shows the high level use case diagram for HNDE student portal mattakkuliya

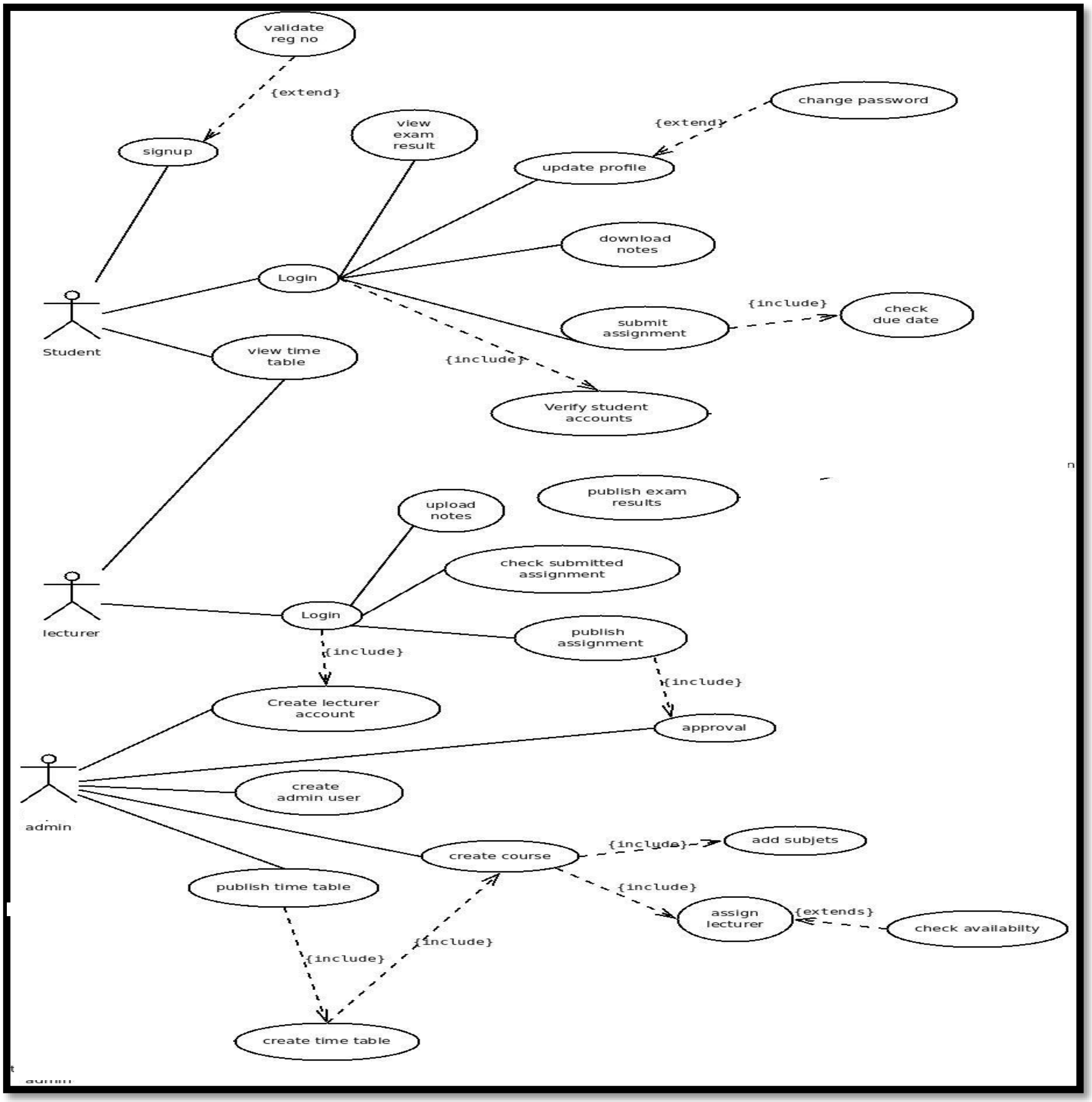


FIGURE B.1 HIGH LEVEL USE CASE DIAGRAM

USE CASE DESCRIPTIONS

It provides a detailed description about the Use Cases defined in the diagram. Some of the main Use Case descriptions related to the high level Use Case diagrams are mentioned below.

Table B.1 shows the description of student login use case .

Use case name	Login
Actor	Student
Description	Student can login to the system using his/her username and password
Pre-Conditions	Having privileges to enter to the system (Approve the signup request from Admin)
Flow of Events	1. Provide user name and password 2. User authentication
Post-Conditions	Student can use the system

TABLE B.1:USE CASE DESCRIPTION-LOGIN

Signup use case description given below in the table B.2

Use case name	Signup
Actor	Student
Description	New student can sign up to enter the system
Pre-Conditions	Who are the not register student in system
Flow of Events	1. Student Reg No. 2. Full Name 3. Email 4. Mobile 5. Select the course 6. Enter the password
Post-Conditions	Send request to create the account

TABLE B. 2 : USE CASE DESCRIPTION - SIGNUP

Table B.3 shows the description of lecturer login use case .

Use case name	Login
Actor	Lecturer
Description	Lecturer can login to the system using his/her username and password
Pre-Conditions	Having privileges to enter to the system (Create the Lecturer account from Admin)
Flow of Events	1. Provide user name and password 2. User authentication
Post-Conditions	Lecturer can use the system

TABLE B. 3 : USE CASE DESCRIPTION – LOGIN

Table B.4 shows the description of upload notes and assignments use case .

Use case name	Upload notes/Assignments
Actor	Lecturer
Description	Lecturer can upload notes & assignments
Pre-Conditions	Log in to the system
Flow of Events	Upload the assignments, Set the end date
Post-Conditions	Student can download assignment

TABLE B. 4 : USE CASE DESCRIPTION – UPLOAD NOTES

Table B.5 shows the description of publish time table use case .

Use case name	Publish Time Table
Actor	Admin
Description	Publish the time table
Pre-Conditions	Batch & lecturers want to create
Flow of Events	1. Select batch 2. Select lecturer 3. Select location 4. Set date & time 5. Select If it is repeat 6. If it is repeat select repeat frequency

TABLE B. 5 : USE CASE DESCRIPTION – PUBLISH TIME TABLE

Table B.6 shows the description of create lecture account use case .

Use case name	Create lecturer account
Actor	Admin
Description	Admin can create the lecturer account
Pre-Conditions	Who are request to create the account
Flow of Events	1. Enter the full name 2. Enter the Email 3.Enter the mobile no 4.Select the designation
Post-Conditions	Create the lecturer account

TABLE B. 6 : USE CASE DESCRIPTION –CREATE LECTURE ACCOUNT

CLASS DIAGRAM FOR THE HNDE STUDENT PORTAL

Figure B.2 show the Class Diagram for the HNDE student portal

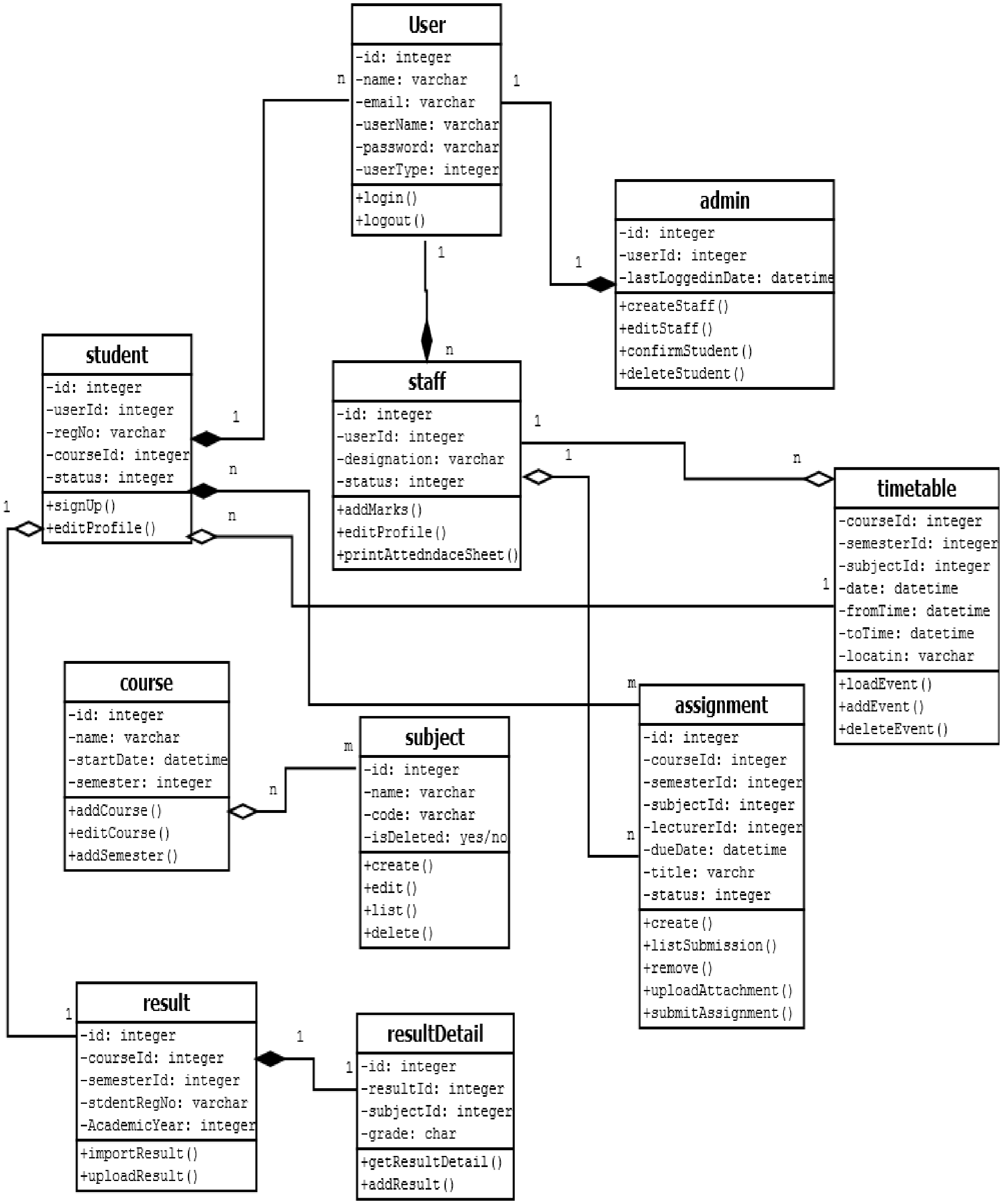


FIGURE B.2 CLASS DIAGRAM FOR THE STUDENT PORTAL MATTKKULIYA

ACTIVITY DIAGRAM FOR THE SUBMIT ASSIGNMENT

The activity diagram which relates to apply order module is displayed by figure B.3.

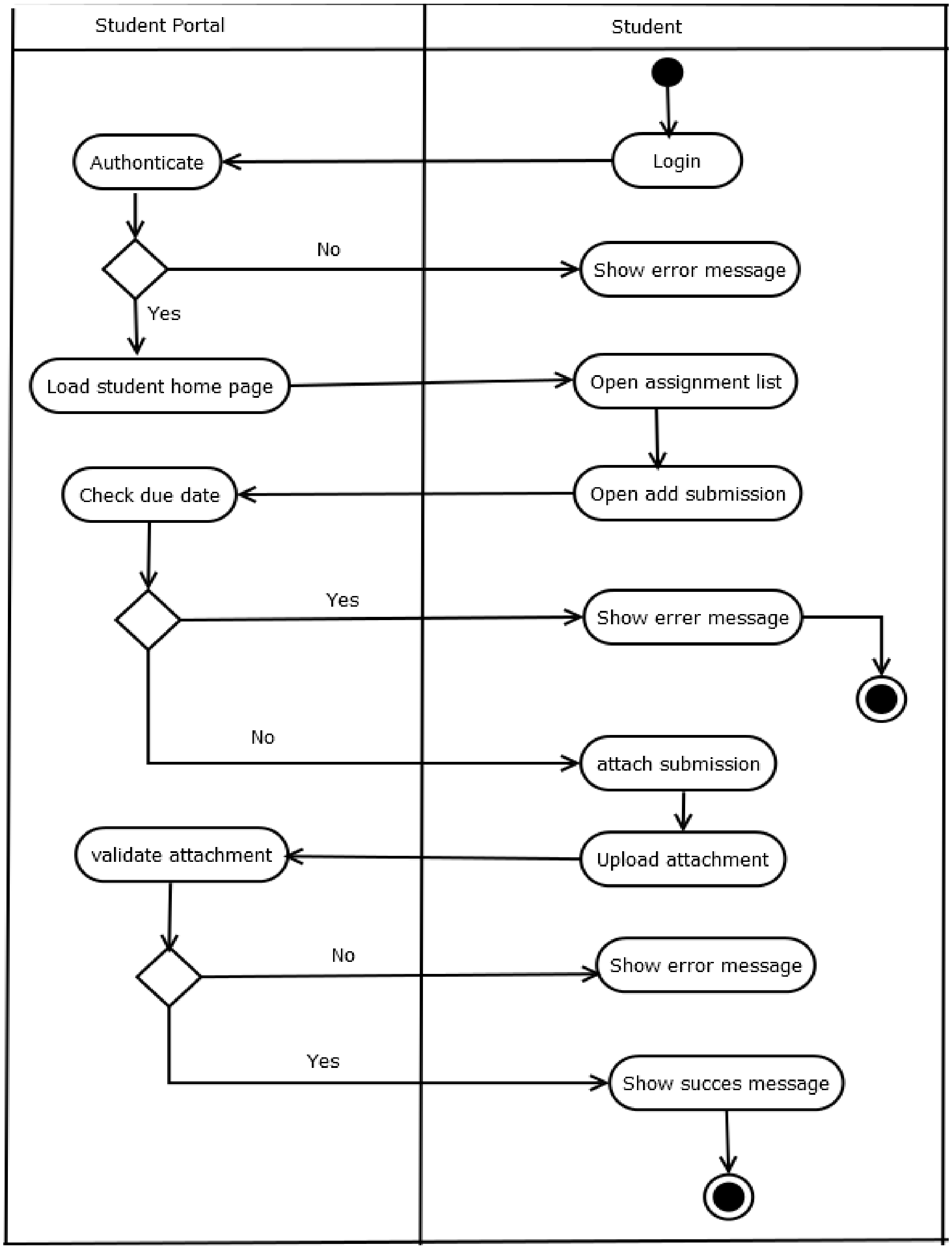


FIGURE B.3 ACTIVITY DIAGRAM FOR THE SUBMIT ASSIGNMENT

SEQUENCE DIAGRAM FOR THE ASSIGNMENT MODULE

The sequence diagram which relates to the assignment module is displayed by figure B.4

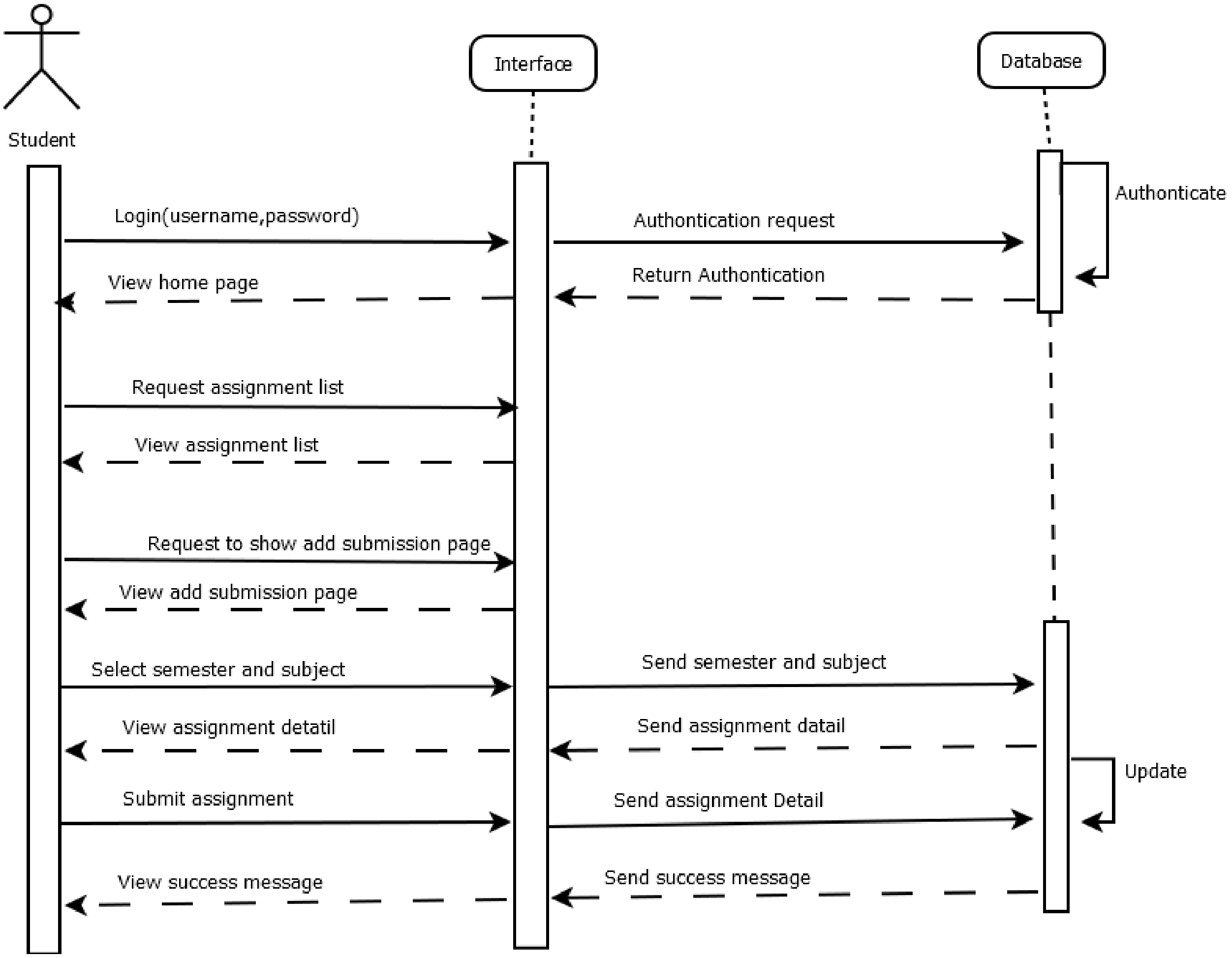
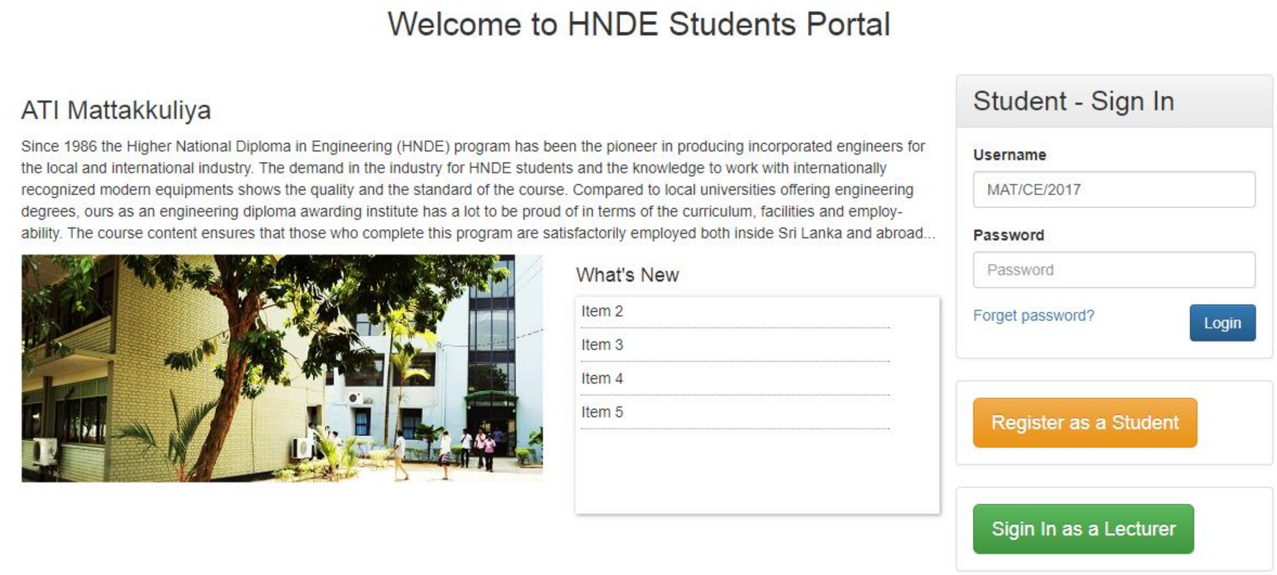


FIGURE B.4 SEQUENCE DIAGRAM FOR ASSIGNMENT MODULE

APPENDIX C -USER DOCUMENTATION

HOME PAGE

All the users can be access the system through this home page. Following figure C.2 shows it looks like.



Welcome to HNDE Students Portal

ATI Mattakkuliya

Since 1986 the Higher National Diploma in Engineering (HNDE) program has been the pioneer in producing incorporated engineers for the local and international industry. The demand in the industry for HNDE students and the knowledge to work with internationally recognized modern equipments shows the quality and the standard of the course. Compared to local universities offering engineering degrees, ours as an engineering diploma awarding institute has a lot to be proud of in terms of the curriculum, facilities and employ-ability. The course content ensures that those who complete this program are satisfactorily employed both inside Sri Lanka and abroad...

Student - Sign In

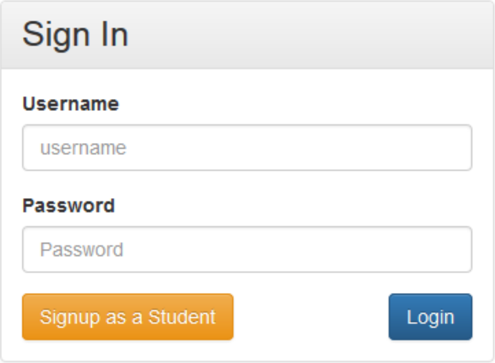
Username: MAT/CE/2017

Password: Password

FIGURE C.1 HOME PAGE

LOGIN FORM

Following Figure C.1 shows user login form which allows users to log into the system. All levels of users can log into the system in one form. When user try to log into the system check whether this user is valid user or not otherwise system display error message.


Sign In

Username

username

Password

Password

FIGURE C.1 LOGIN PAGE 1

GENERAL GUIDANCE TO USE THE ADMIN FUNCTIONS

HOW TO MANAGE THE STAFF

Click on the staff menu

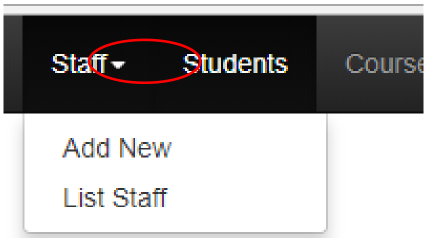


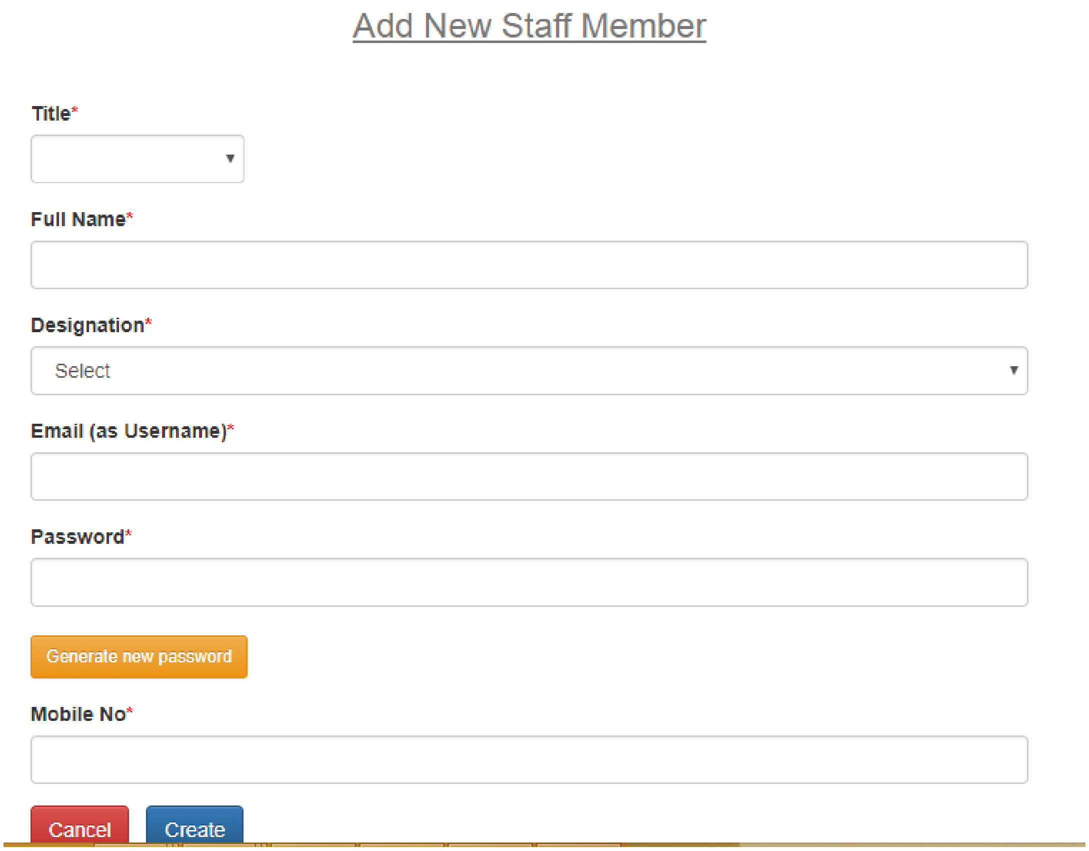
FIGURE C.3 STAFF MENU

HOW TO ADD NEW STAFF

Staff member create by admin. When create the staff member, that staff member will receive the email

Step 1:-Click on Add new in Staff menu
Step 2:-Fill required details & click create button (Then email will send to given email address)
Step 3:-Staff member want change the password (1st time system generate password want use)

Figure C.4 shows how to add new staff member



Add New Staff Member

Title*

Full Name*

Designation*

Select

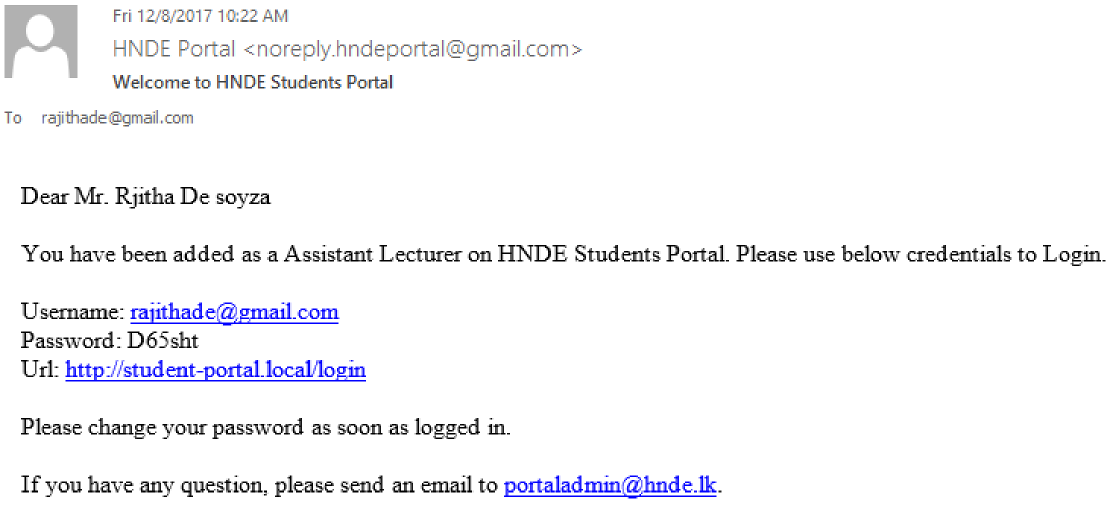
Email (as Username)*

Password*

Mobile No*

FIGURE C.4 ADD NEW STAFF

When create new staff by admin,that staff member receive the email like figure C.5.After that first of all staff member want to log the system using that password and then chnge the password.



Fri 12/8/2017 10:22 AM

HNDE Portal <noreply.hndeportal@gmail.com>

Welcome to HNDE Students Portal

To rajithade@gmail.com

Dear Mr. Rjitha De soyza

You have been added as a Assistant Lecturer on HNDE Students Portal. Please use below credentials to Login.

Username: rajithade@gmail.com
Password: D65sht
Url: http://student-portal.local/login

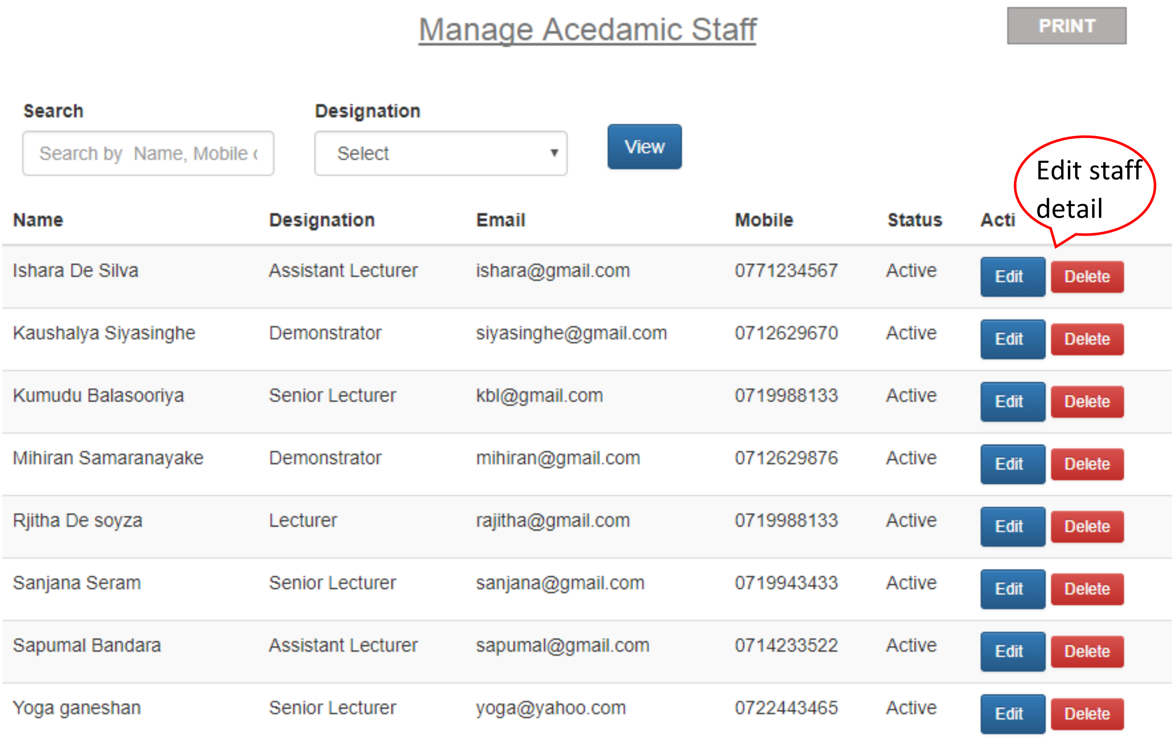
Please change your password as soon as logged in.

If you have any question, please send an email to portaladmin@hnde.lk.

FIGURE C.5 NEW STAFF MEMBER RECEIVED MAIL

View Staff

After the registration of staff member, admin can manage that staff using this interface. Admin can filter the staff list using this interface and take print out of staff list.



Manage Acedamic Staff

Search: Search by Name, Mobile (

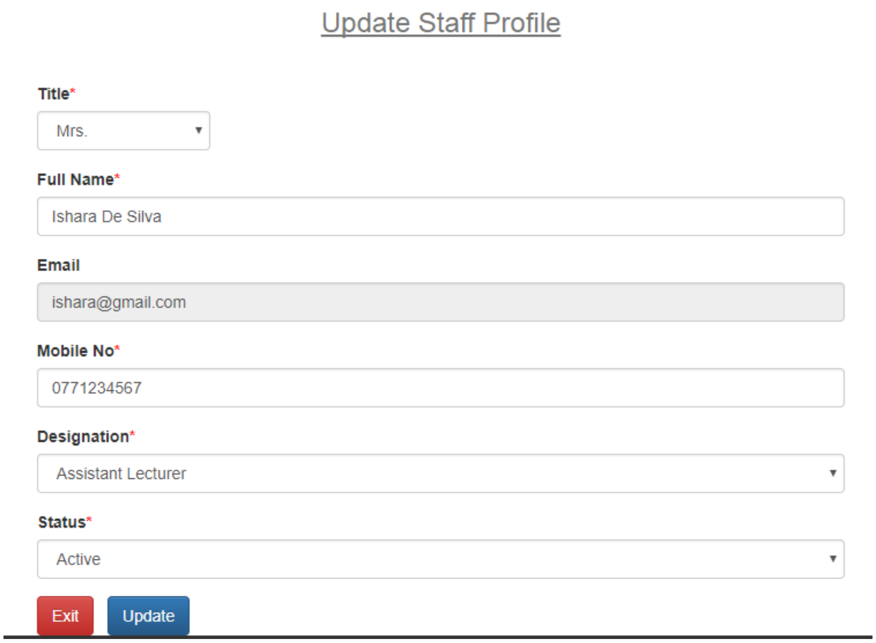
Designation: Select ▾

Name	Designation	Email	Mobile	Status	Acti
Ishara De Silva	Assistant Lecturer	ishara@gmail.com	0771234567	Active	Edit Delete
Kaushalya Siyasinghe	Demonstrator	siyasinghe@gmail.com	0712629670	Active	Edit Delete
Kumudu Balasooriya	Senior Lecturer	kbl@gmail.com	0719988133	Active	Edit Delete
Mihiran Samaranayake	Demonstrator	mihiran@gmail.com	0712629876	Active	Edit Delete
Rjitha De soyza	Lecturer	rajitha@gmail.com	0719988133	Active	Edit Delete
Sanjana Seram	Senior Lecturer	sanjana@gmail.com	0719943433	Active	Edit Delete
Sapumal Bandara	Assistant Lecturer	sapumal@gmail.com	0714233522	Active	Edit Delete
Yoga ganeshan	Senior Lecturer	yoga@yahoo.com	0722443465	Active	Edit Delete

Figure C.5: View staff

Edit Staff

Admin can edit selected staff member in the above interface shown in figure C.5.
After edit staff detail click the update button



Update Staff Profile

Title* Mrs.

Full Name* Ishara De Silva

Email ishara@gmail.com

Mobile No* 0771234567

Designation* Assistant Lecturer

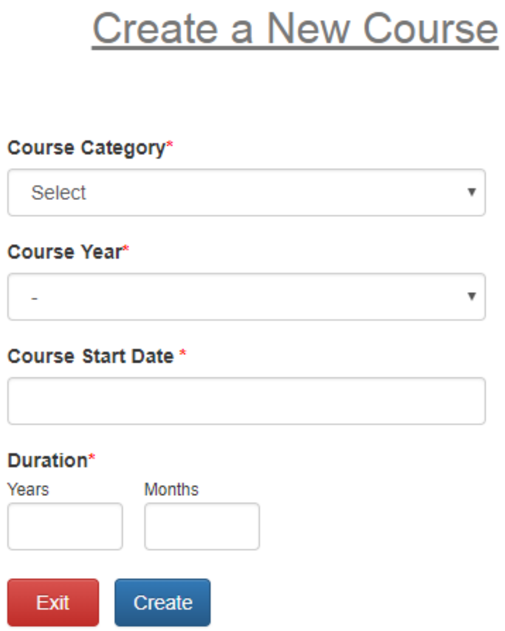
Status* Active

FIGURE C.6 EDIT STAFF

HOW TO MANAGE COURSE

CREATE COURSE

Every year course want to update. As the example 2015 QS and 2016 QS are different courses. Because course may be update.



Create a New Course

Course Category*

Select ▾

Course Year*

- ▾

Course Start Date *

Duration*

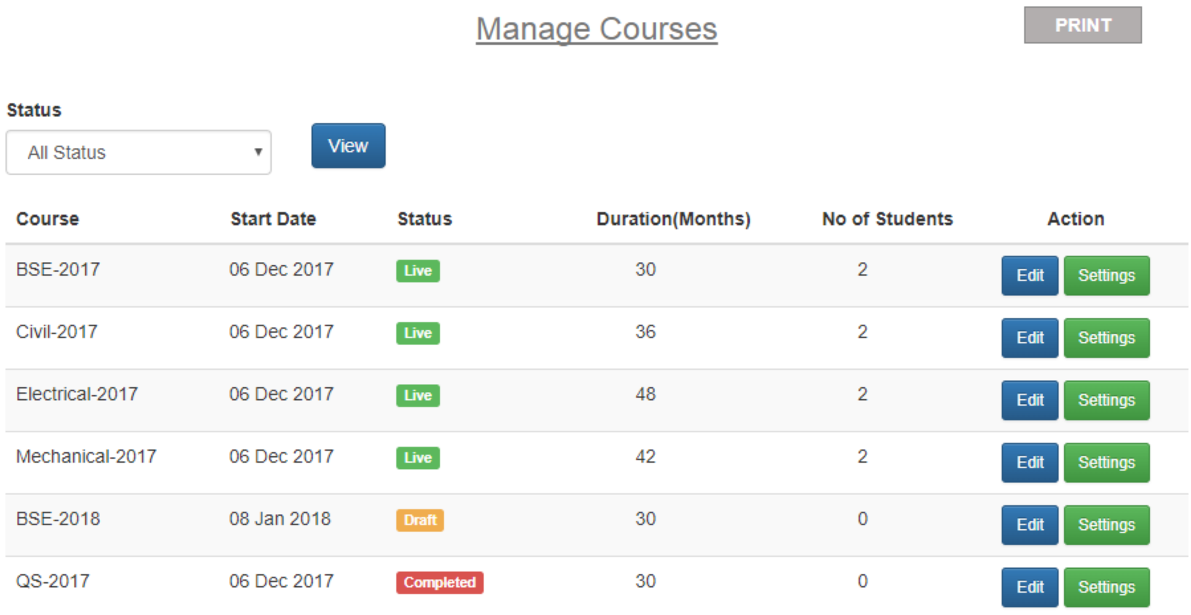
Years

Months

FIGURE C.7 CREATE COURSE

VIEW COURSE

Using this figure C.8 interface we can see what the available course are in this system. As well as can take the print out of course list



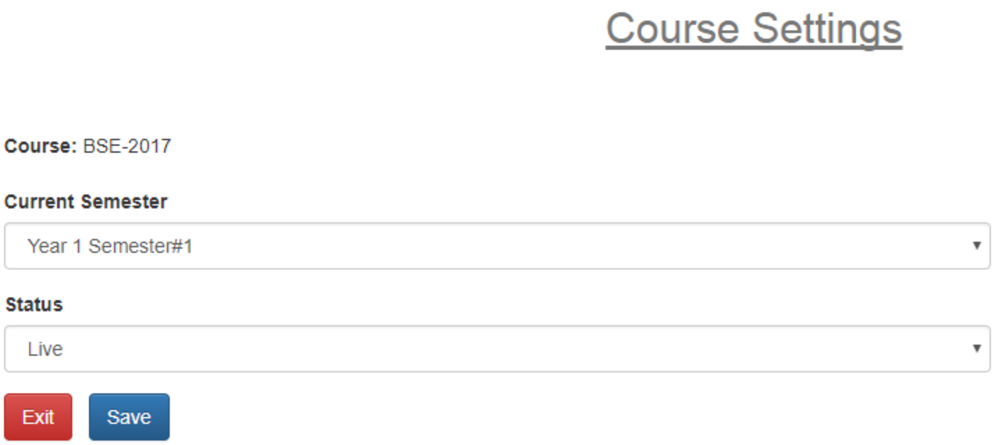
Status: All Status

Course	Start Date	Status	Duration(Months)	No of Students	Action
BSE-2017	06 Dec 2017	Live	30	2	Edit Settings
Civil-2017	06 Dec 2017	Live	36	2	Edit Settings
Electrical-2017	06 Dec 2017	Live	48	2	Edit Settings
Mechanical-2017	06 Dec 2017	Live	42	2	Edit Settings
BSE-2018	08 Jan 2018	Draft	30	0	Edit Settings
QS-2017	06 Dec 2017	Completed	30	0	Edit Settings

FIGURE C.7 VIEW COURSE

COURSE SETTINGS

Using this interface can set status (Live/Draft/Completed) of the selected course in current semester



Course: BSE-2017

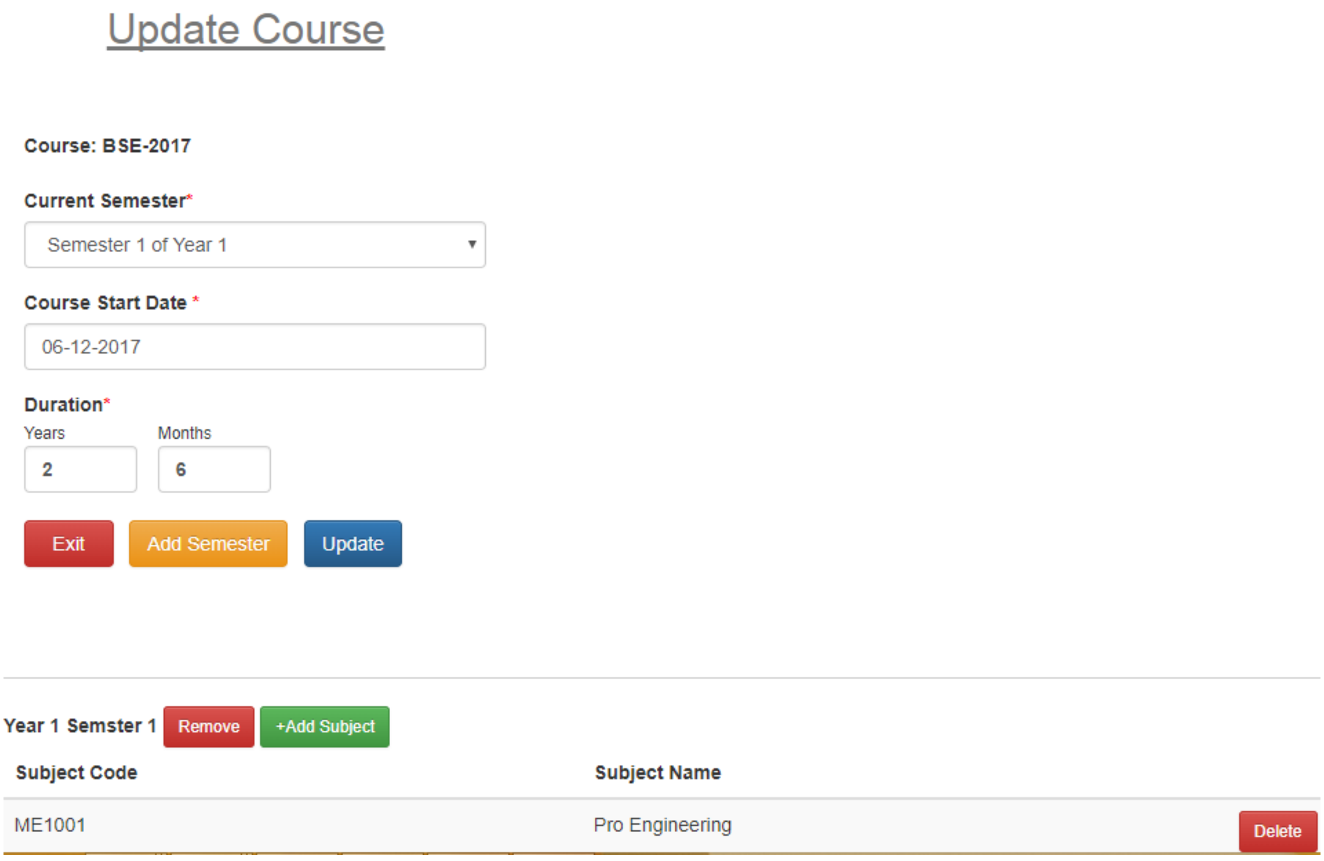
Current Semester: Year 1 Semester#1

Status: Live

FIGURE C.8: COURSE SETTINGS

UPDATE COURSE & ADD SEMESTER

Every course has the semester. (1year has 2 semester).first select the course and click the edit. Then can see one of the button “Add semester”



Update Course

Course: BSE-2017

Current Semester*

Semester 1 of Year 1

Course Start Date *

06-12-2017

Duration*

Years: 2

Months: 6

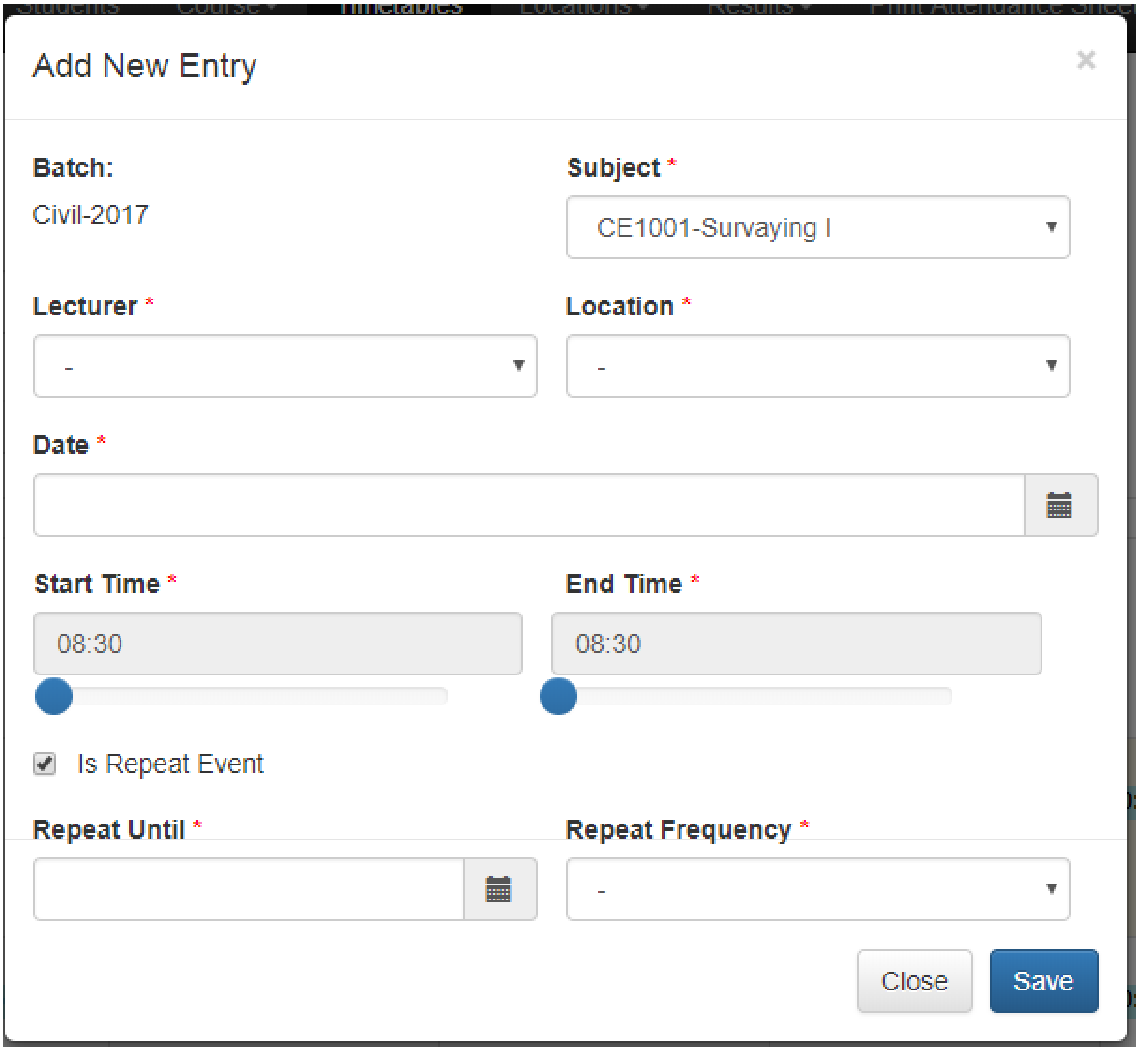
Year 1 Semster 1

Subject Code	Subject Name	
ME1001	Pro Engineering	Delete

FIGURE C.9 UPDATE COURSE AND ADD SEMESTER

ADD TIMETABLE

Student and staff members can see their time table, when enter the timetable by admin. Figure C.10 display the add new entry to timetable.This new entry is repeating as weekly or daily want to tick “is repeat event” check box.



Add New Entry ×

Batch:
Civil-2017

Subject *
CE1001-Survaying I ▾

Lecturer *
- ▾

Location *
- ▾

Date *

Start Time *
08:30

End Time *
08:30

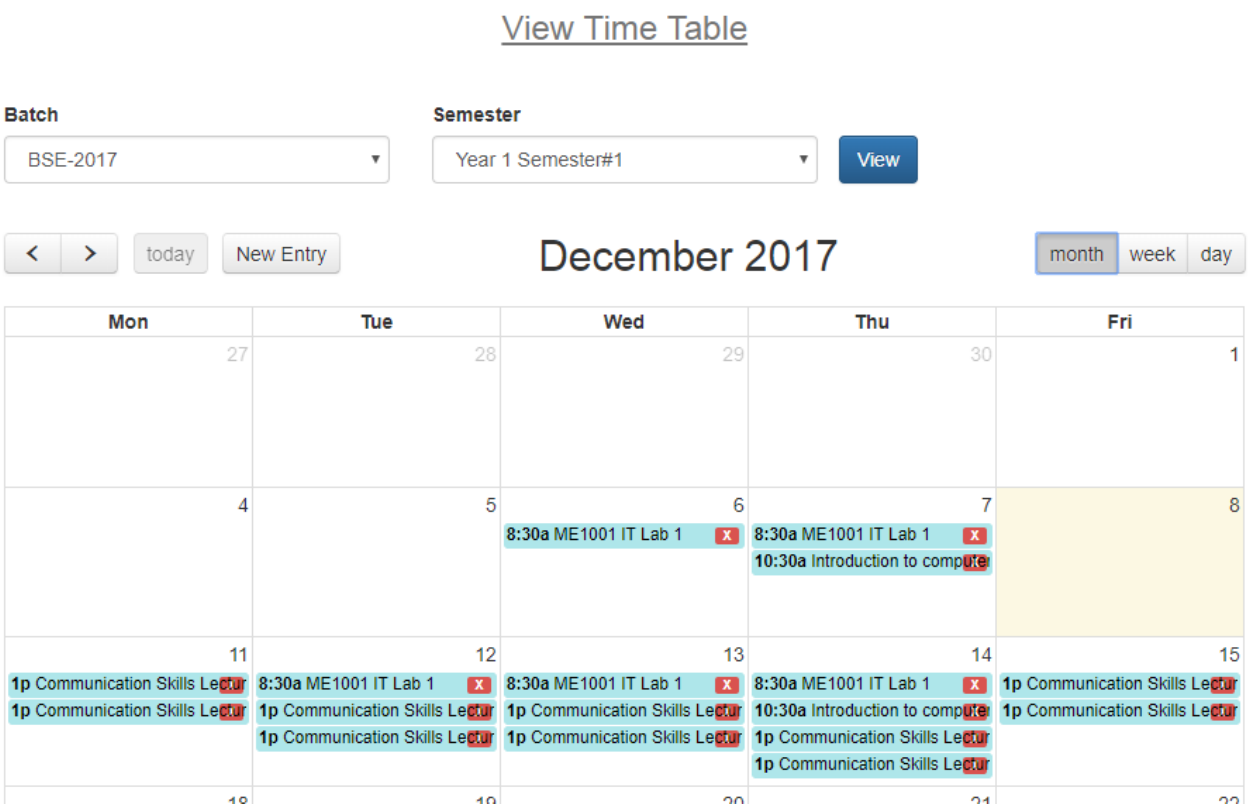
☑ Is Repeat Event

Repeat Until *

Repeat Frequency *
- ▾

FIGURE C.10 NEW ENTRY OF TIMETABLE

Then shows the view timetable .In this interface can show the timetable course wise weekly, daily and monthly.Figure C.11



View Time Table

Batch: BSE-2017

Semester: Year 1 Semester#1

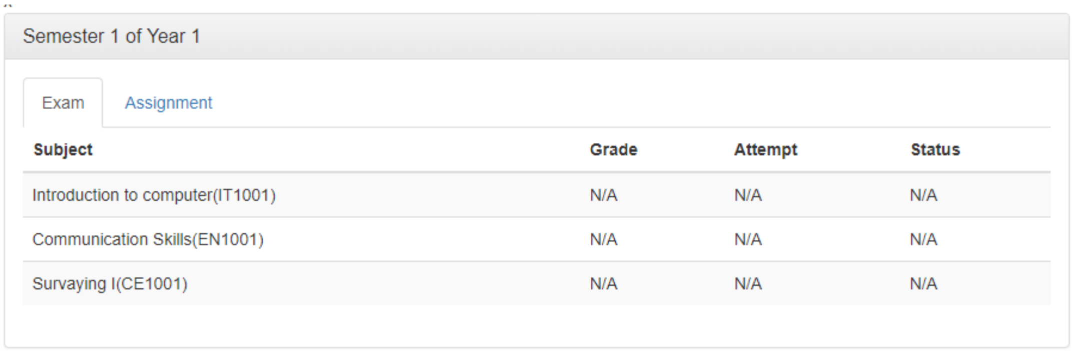
December 2017

Mon	Tue	Wed	Thu	Fri
27	28	29	30	1
4	5	6 8:30a ME1001 IT Lab 1	7 8:30a ME1001 IT Lab 1 10:30a Introduction to compute	8
11 1p Communication Skills Lectur 1p Communication Skills Lectur	12 8:30a ME1001 IT Lab 1 1p Communication Skills Lectur 1p Communication Skills Lectur	13 8:30a ME1001 IT Lab 1 1p Communication Skills Lectur 1p Communication Skills Lectur	14 8:30a ME1001 IT Lab 1 10:30a Introduction to compute 1p Communication Skills Lectur 1p Communication Skills Lectur	15 1p Communication Skills Lectur 1p Communication Skills Lectur
18	19	20	21	22

FIGURE C.11 VIEW TIMETABLE

VIEW ACADEMIC PROFILE

In the student list, when click the view academic profile button in front of selected student admin can see the interface like figure C.12



Semester 1 of Year 1

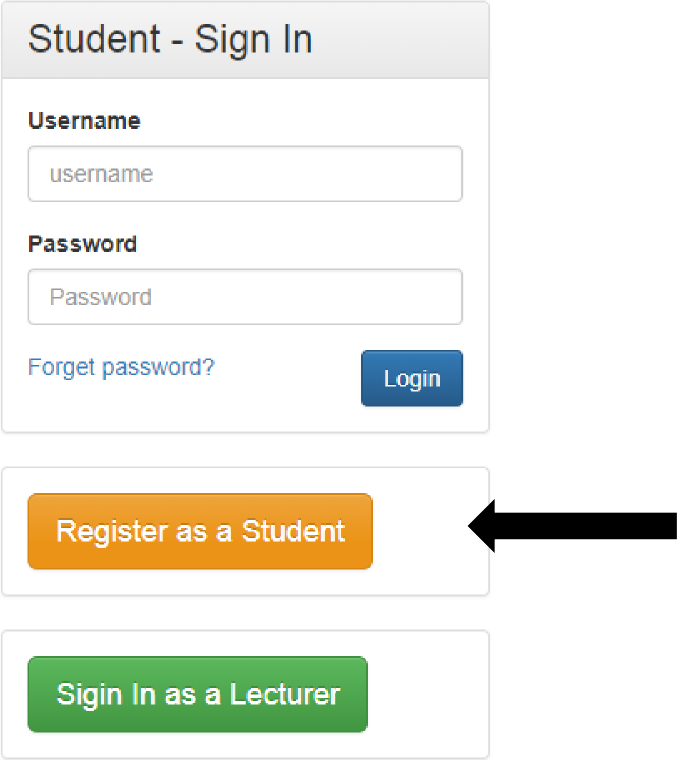
Subject	Grade	Attempt	Status
Introduction to computer(IT1001)	N/A	N/A	N/A
Communication Skills(EN1001)	N/A	N/A	N/A
Survaying I(CE1001)	N/A	N/A	N/A

FIGURE C.12 VIEW ACADEMIC PROFILE

STUDENT ROLL

STUDENT REGISTRATION

In the home page student registration button is appear like as Figure C.12.



Click here

FIGURE C.12 STUDENT REGISTRATION

STUDENT SIGNUP

When click the register as student button student signup page will appear. Student signup page shows in figure C.13

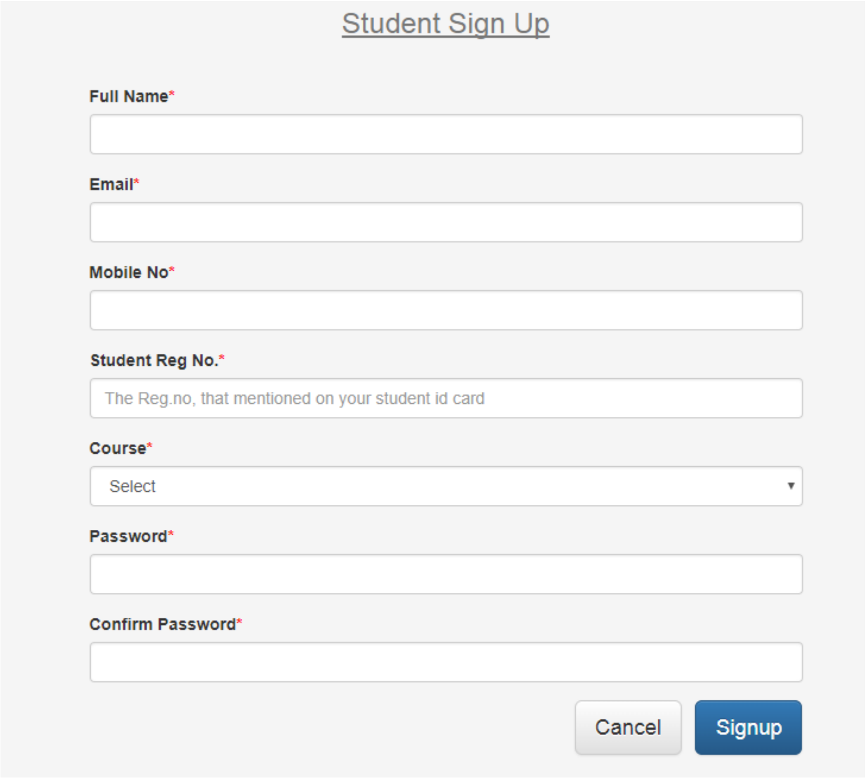
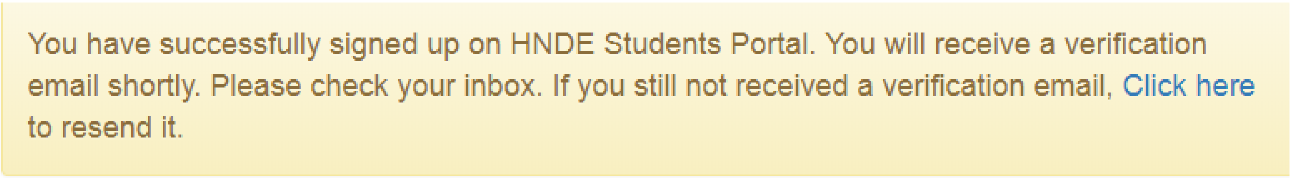


FIGURE C.13 STUDENT SIGNUP

After that show the message to like *Figure C.10*



You have successfully signed up on HNDE Students Portal. You will receive a verification email shortly. Please check your inbox. If you still not received a verification email, Click here to resend it.

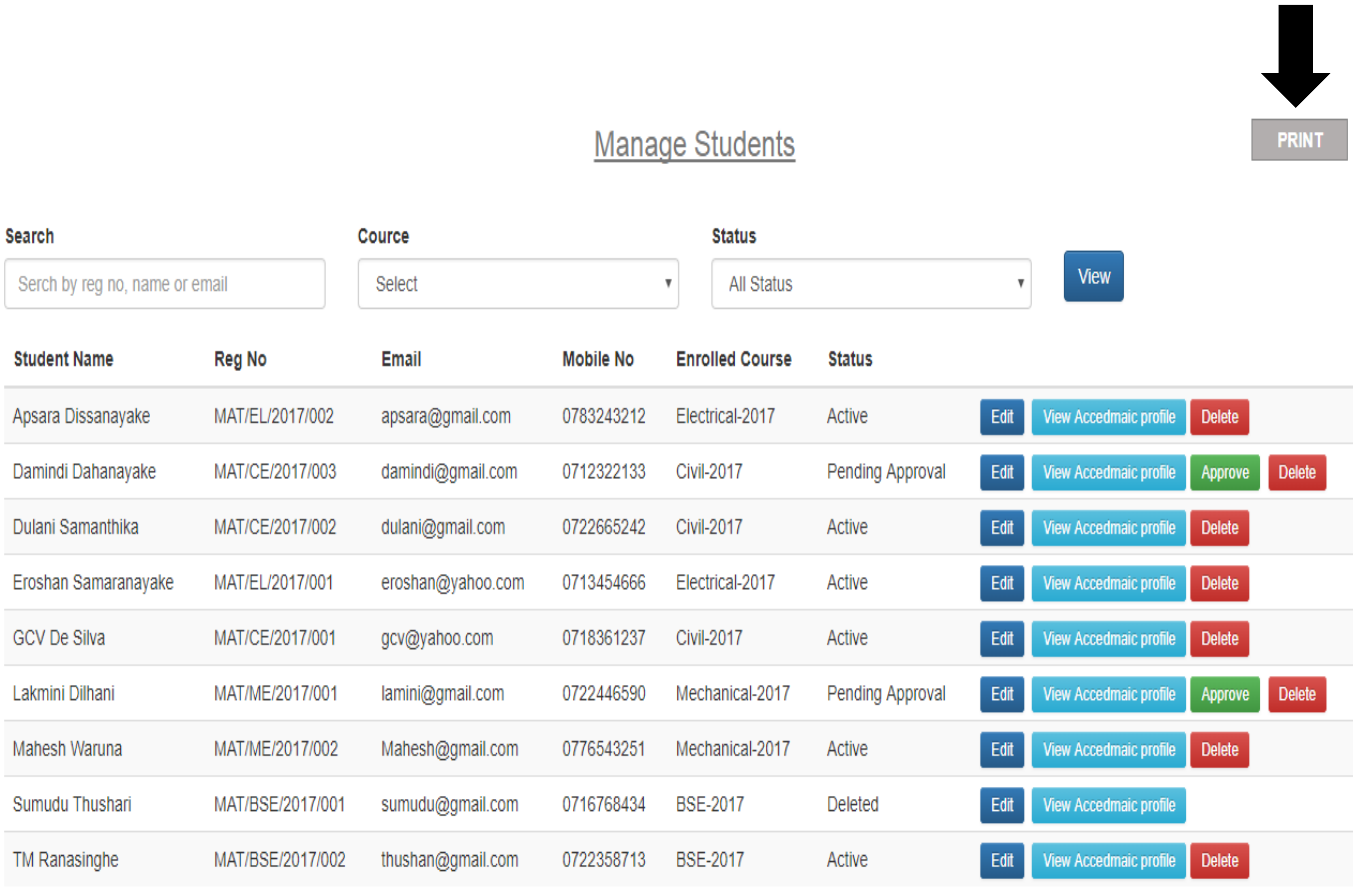
FIGURE C.14 VERIFICATION MAIL

APPENDIX D-MANAGEMENT REPORTS

STUDENT LIST

Admin can generate this report. Using this window admin can take the student list course wise, status wise as well as can search by registration no ,name or email

Click on here



Manage Students

Search: Serch by reg no, name or email

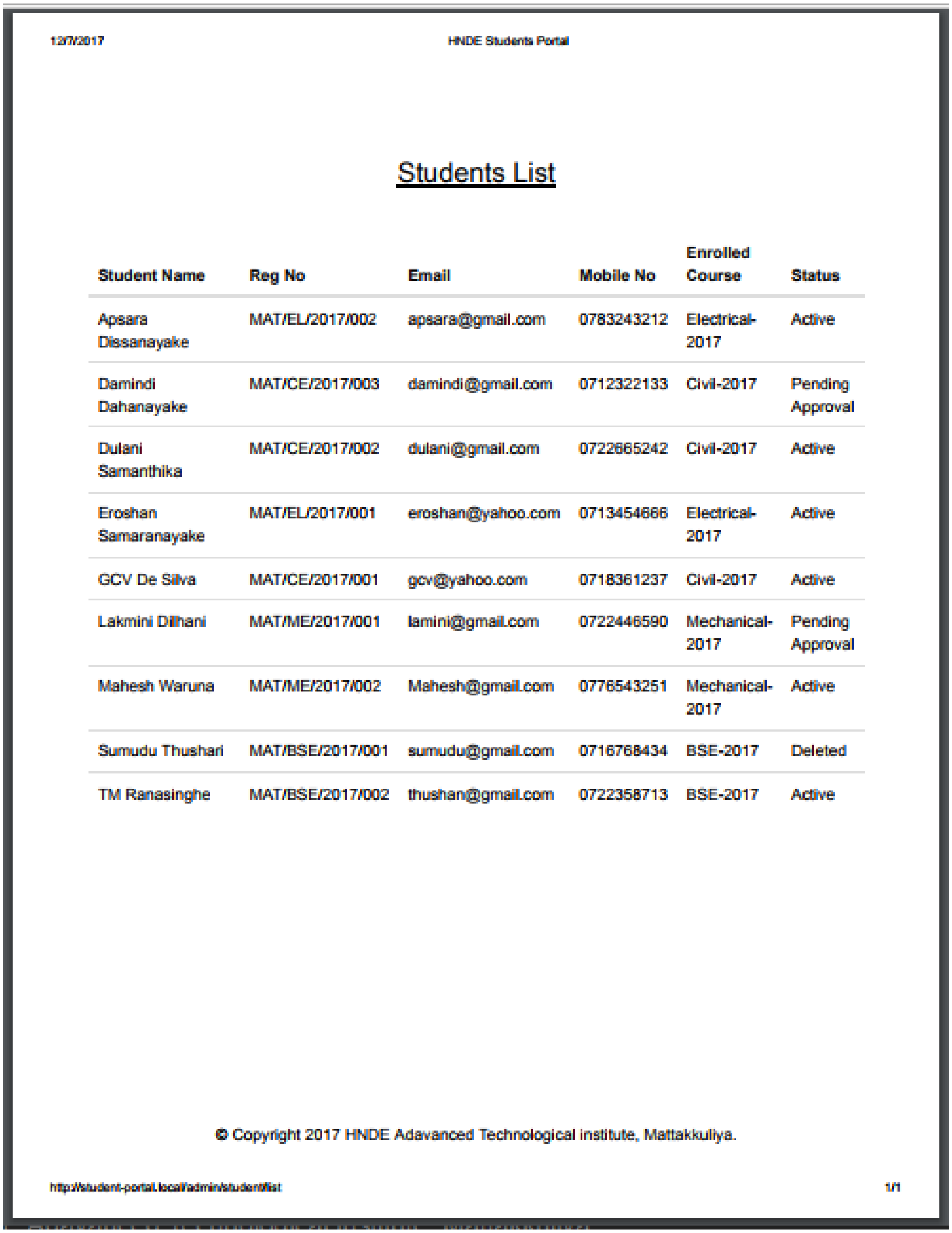
Cource: Select

Status: All Status

Student Name	Reg No	Email	Mobile No	Enrolled Course	Status	
Apsara Dissanayake	MAT/EL/2017/002	apsara@gmail.com	0783243212	Electrical-2017	Active	Edit View Accedmaic profile Delete
Damindi Dahanayake	MAT/CE/2017/003	damindi@gmail.com	0712322133	Civil-2017	Pending Approval	Edit View Accedmaic profile Approve Delete
Dulani Samanthika	MAT/CE/2017/002	dulani@gmail.com	0722665242	Civil-2017	Active	Edit View Accedmaic profile Delete
Eroshan Samaranayake	MAT/EL/2017/001	eroshan@yahoo.com	0713454666	Electrical-2017	Active	Edit View Accedmaic profile Delete
GCV De Silva	MAT/CE/2017/001	gcv@yahoo.com	0718361237	Civil-2017	Active	Edit View Accedmaic profile Delete
Lakmini Dilhani	MAT/ME/2017/001	lamini@gmail.com	0722446590	Mechanical-2017	Pending Approval	Edit View Accedmaic profile Approve Delete
Mahesh Waruna	MAT/ME/2017/002	Mahesh@gmail.com	0776543251	Mechanical-2017	Active	Edit View Accedmaic profile Delete
Sumudu Thushari	MAT/BSE/2017/001	sumudu@gmail.com	0716768434	BSE-2017	Deleted	Edit View Accedmaic profile
TM Ranasinghe	MAT/BSE/2017/002	thushan@gmail.com	0722358713	BSE-2017	Active	Edit View Accedmaic profile Delete

FIGURE D.1 STUDENT LIST

After click on view button and then click the print button .Then can take the report (print out) of student list.That is shows in figure D.2



12/7/2017 HNDE Students Portal

Students List

Student Name	Reg No	Email	Mobile No	Enrolled Course	Status
Apsara Dissanayake	MAT/EL/2017/002	apsara@gmail.com	0783243212	Electrical-2017	Active
Damindi Dahanayake	MAT/CE/2017/003	damindi@gmail.com	0712322133	Civil-2017	Pending Approval
Dulani Samanthika	MAT/CE/2017/002	dulani@gmail.com	0722665242	Civil-2017	Active
Eroshan Samaranayake	MAT/EL/2017/001	eroshan@yahoo.com	0713454666	Electrical-2017	Active
GCV De Silva	MAT/CE/2017/001	gcv@yahoo.com	0718361237	Civil-2017	Active
Lakmini Dilhani	MAT/ME/2017/001	lamini@gmail.com	0722446590	Mechanical-2017	Pending Approval
Mahesh Waruna	MAT/ME/2017/002	Mahesh@gmail.com	0776543251	Mechanical-2017	Active
Sumudu Thushari	MAT/BSE/2017/001	sumudu@gmail.com	0716768434	BSE-2017	Deleted
TM Ranasinghe	MAT/BSE/2017/002	thushan@gmail.com	0722358713	BSE-2017	Active

© Copyright 2017 HNDE Adavanced Technological institute, Mattakkuliya.

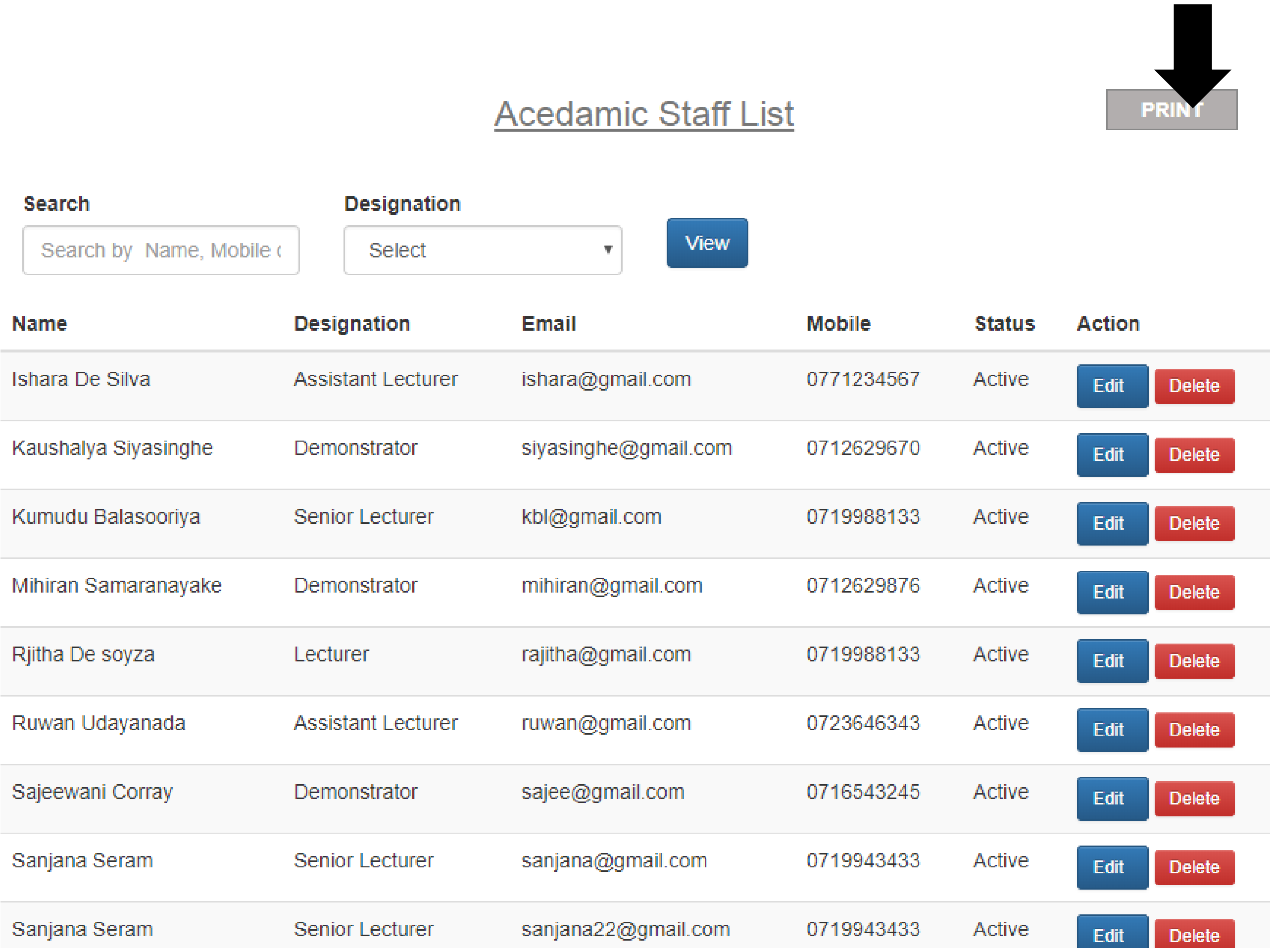
http://student-portal.local/admin/studentlist 1/1

FIGURE D.2 STUDENT LIST REPORT

STAFF LIST

Admin can generate this report. Using this window admin can take the list of registered staff of the system. Using this window admin can filter the staff list using designation, as well as can search by name or email or mobile no. Staff list shows in figure D.3

Click on here



Acedamic Staff List

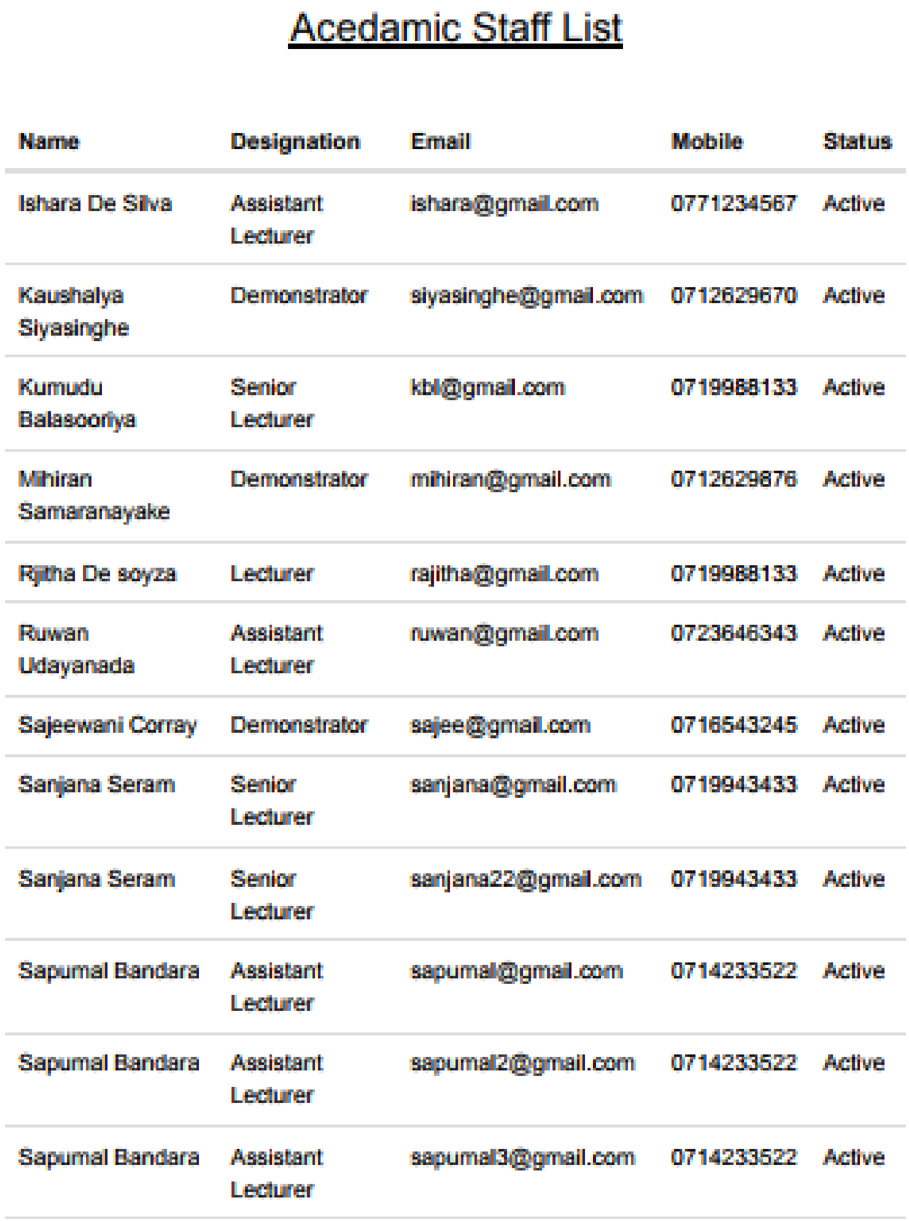
Search: Search by Name, Mobile (

Designation: Select

Name	Designation	Email	Mobile	Status	Action
Ishara De Silva	Assistant Lecturer	ishara@gmail.com	0771234567	Active	Edit Delete
Kaushalya Siyasinghe	Demonstrator	siyasinghe@gmail.com	0712629670	Active	Edit Delete
Kumudu Balasooriya	Senior Lecturer	kbl@gmail.com	0719988133	Active	Edit Delete
Mihiran Samaranayake	Demonstrator	mihiran@gmail.com	0712629876	Active	Edit Delete
Rjitha De soyza	Lecturer	rajitha@gmail.com	0719988133	Active	Edit Delete
Ruwan Udayanada	Assistant Lecturer	ruwan@gmail.com	0723646343	Active	Edit Delete
Sajeewani Corray	Demonstrator	sajee@gmail.com	0716543245	Active	Edit Delete
Sanjana Seram	Senior Lecturer	sanjana@gmail.com	0719943433	Active	Edit Delete
Sanjana Seram	Senior Lecturer	sanjana22@gmail.com	0719943433	Active	Edit Delete

FIGURE D.3 STAFF LIST

After click the print button .Then can take the report (print out) of staff list.That is shows in figure D.4



Acedamic Staff List

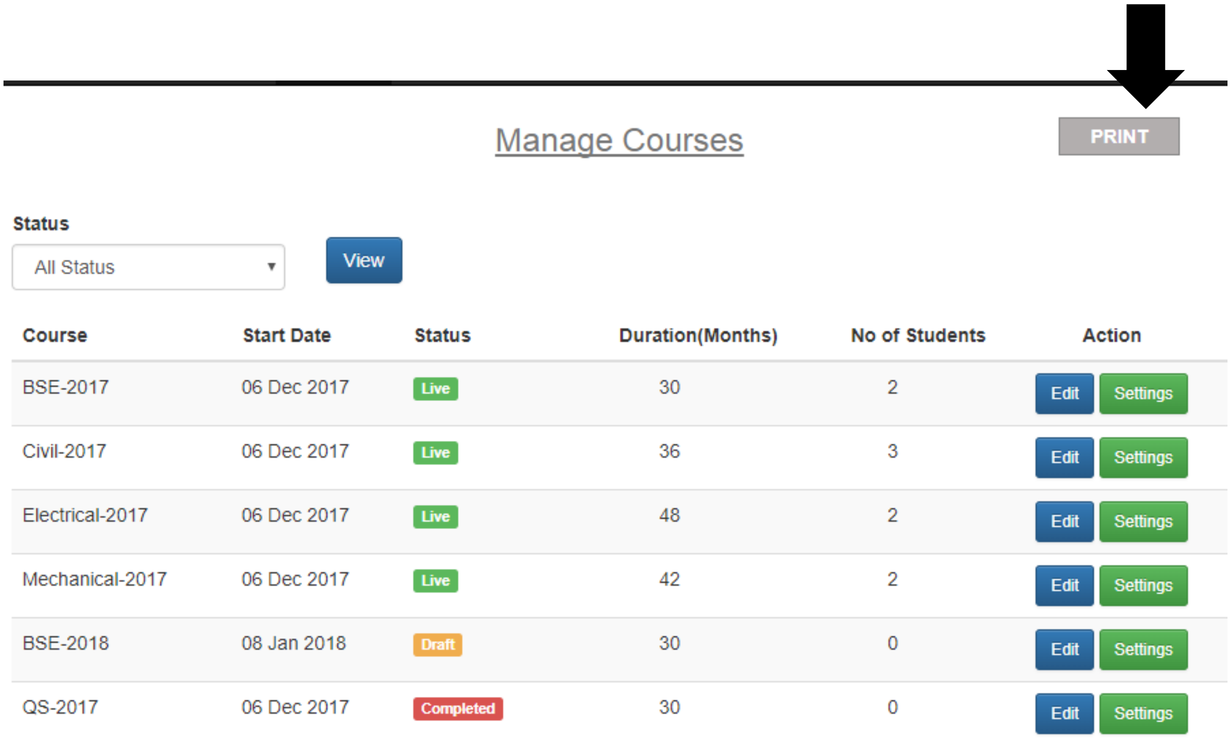
Name	Designation	Email	Mobile	Status
Ishara De Silva	Assistant Lecturer	ishara@gmail.com	0771234567	Active
Kaushalya Siyasinghe	Demonstrator	siyasinghe@gmail.com	0712629670	Active
Kumudu Balasooriya	Senior Lecturer	kbl@gmail.com	0719988133	Active
Mihiran Samaranayake	Demonstrator	mihiran@gmail.com	0712629876	Active
Rjitha De soyza	Lecturer	rajitha@gmail.com	0719988133	Active
Ruwan Udayanada	Assistant Lecturer	ruwan@gmail.com	0723646343	Active
Sajeewani Corray	Demonstrator	sajee@gmail.com	0716543245	Active
Sanjana Seram	Senior Lecturer	sanjana@gmail.com	0719943433	Active
Sanjana Seram	Senior Lecturer	sanjana22@gmail.com	0719943433	Active
Sapumal Bandara	Assistant Lecturer	sapumal@gmail.com	0714233522	Active
Sapumal Bandara	Assistant Lecturer	sapumal2@gmail.com	0714233522	Active
Sapumal Bandara	Assistant Lecturer	sapumal3@gmail.com	0714233522	Active

FIGURE D.4 STAFF LIST REPORT

COURSE LIST

Admin can generate this report. Using this window admin can take the list of all course available in the system and can filter that using status as live, draft and completed. That display in figure D.5

Click on here



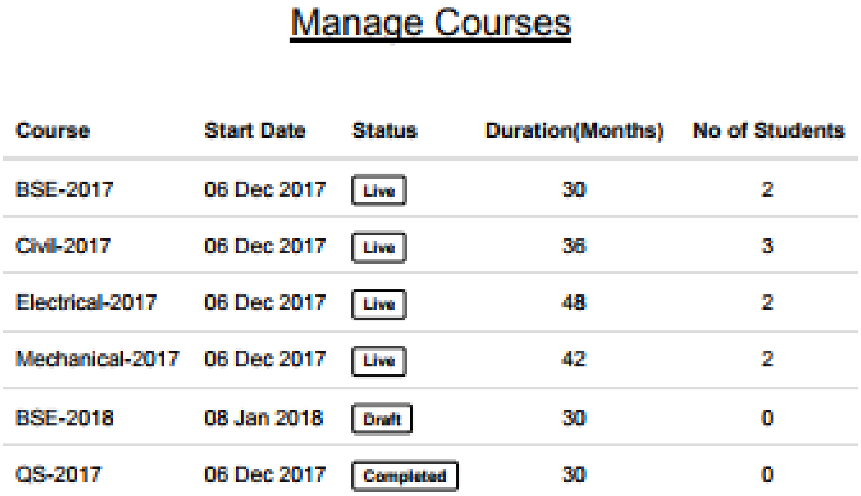
Manage Courses

Status: All Status

Course	Start Date	Status	Duration(Months)	No of Students	Action
BSE-2017	06 Dec 2017	Live	30	2	Edit Settings
Civil-2017	06 Dec 2017	Live	36	3	Edit Settings
Electrical-2017	06 Dec 2017	Live	48	2	Edit Settings
Mechanical-2017	06 Dec 2017	Live	42	2	Edit Settings
BSE-2018	08 Jan 2018	Draft	30	0	Edit Settings
QS-2017	06 Dec 2017	Completed	30	0	Edit Settings

Figure D.5: course list

After click the print button .Then can take the report (print out) of course list. That is shows in figure D.6



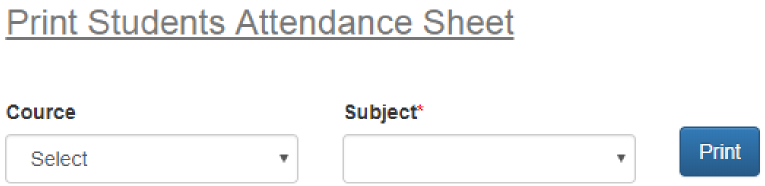
Manage Courses

Course	Start Date	Status	Duration(Months)	No of Students
BSE-2017	06 Dec 2017	Live	30	2
Civil-2017	06 Dec 2017	Live	36	3
Electrical-2017	06 Dec 2017	Live	48	2
Mechanical-2017	06 Dec 2017	Live	42	2
BSE-2018	08 Jan 2018	Draft	30	0
QS-2017	06 Dec 2017	Completed	30	0

FIGURE D.4 REPORT OF COURSE LIST

ATENDANCE SHEET

Before take the attendance sheet want to select course and subject. Figure D.5 shows that interface



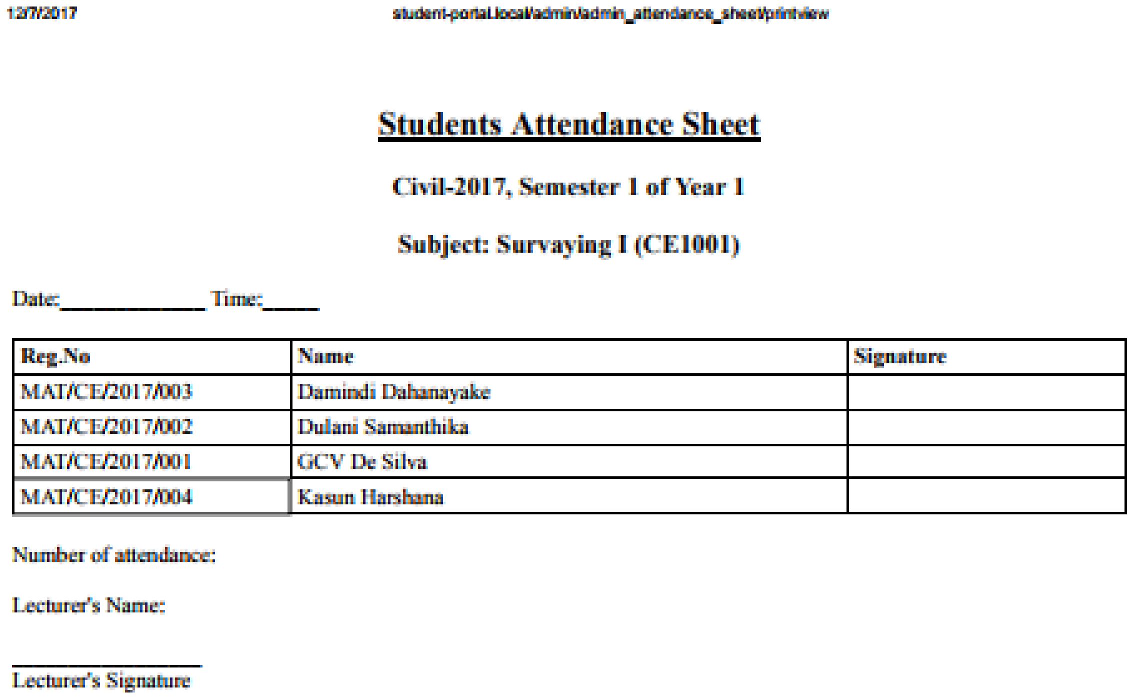
Print Students Attendance Sheet

Cource: Select ▾

Subject*: ▾

FIGURE D.5 STUDENT ATTENDANCE SHEET

After click the print button .Then can take student attendance sheet with course name and subject. That is shows in figure D.6



12/7/2017 student-portal.local/admin/admin_attendance_sheet/printview

Students Attendance Sheet

Civil-2017, Semester 1 of Year 1

Subject: Survaying I (CE1001)

Date:____________ Time:_____

Reg.No	Name	Signature
MAT/CE/2017/003	Damindi Dahanayake	
MAT/CE/2017/002	Dulani Samanthika	
MAT/CE/2017/001	GCV De Silva	
MAT/CE/2017/004	Kasun Harshana	

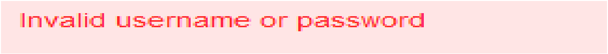
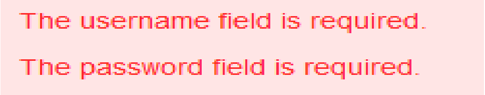
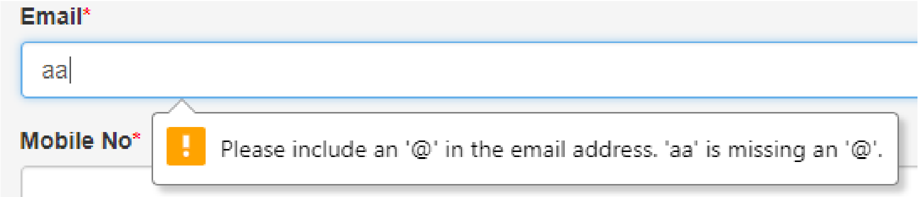
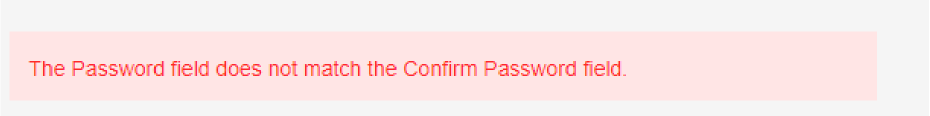
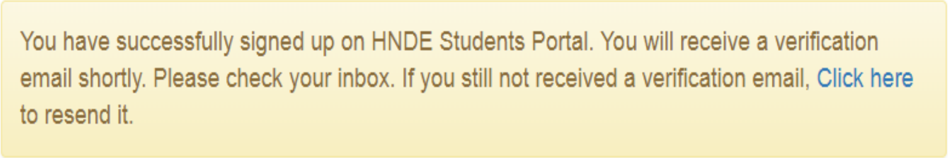
Number of attendance:

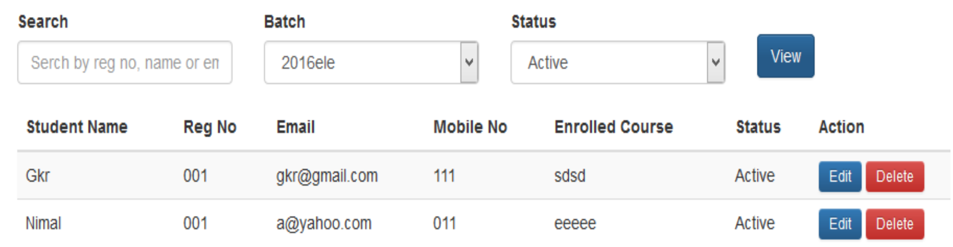
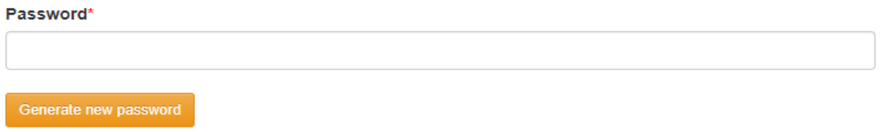
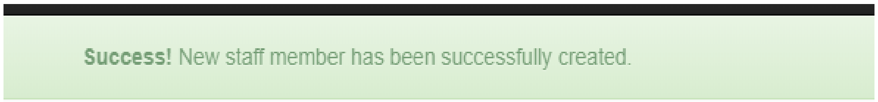
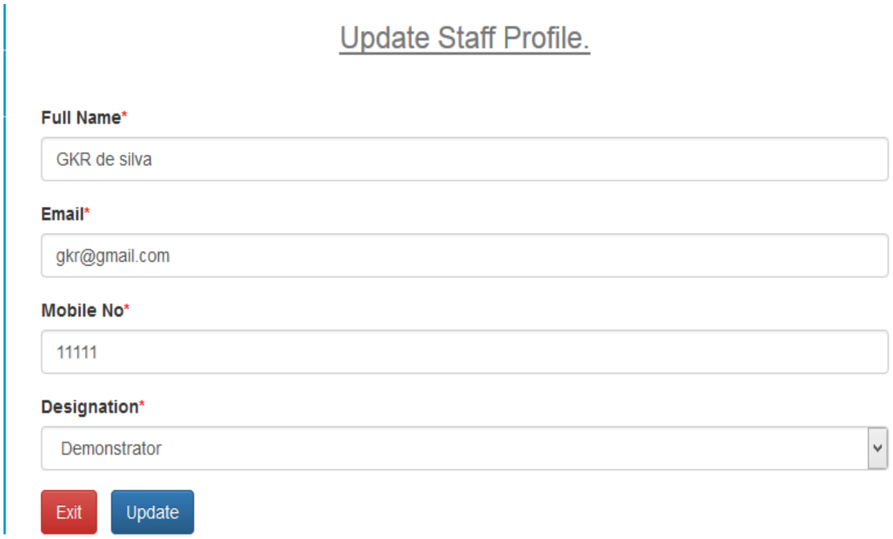
Lecturer's Name:

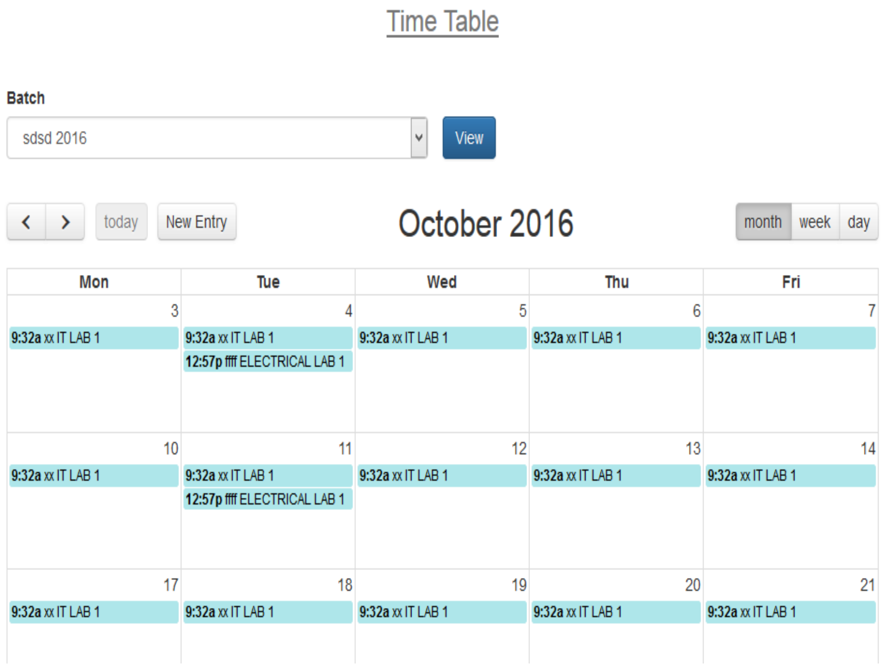
____
Lecturer's Signature

FIGURE D.6 PRINT STUDENT ATTENDANCE SHEET

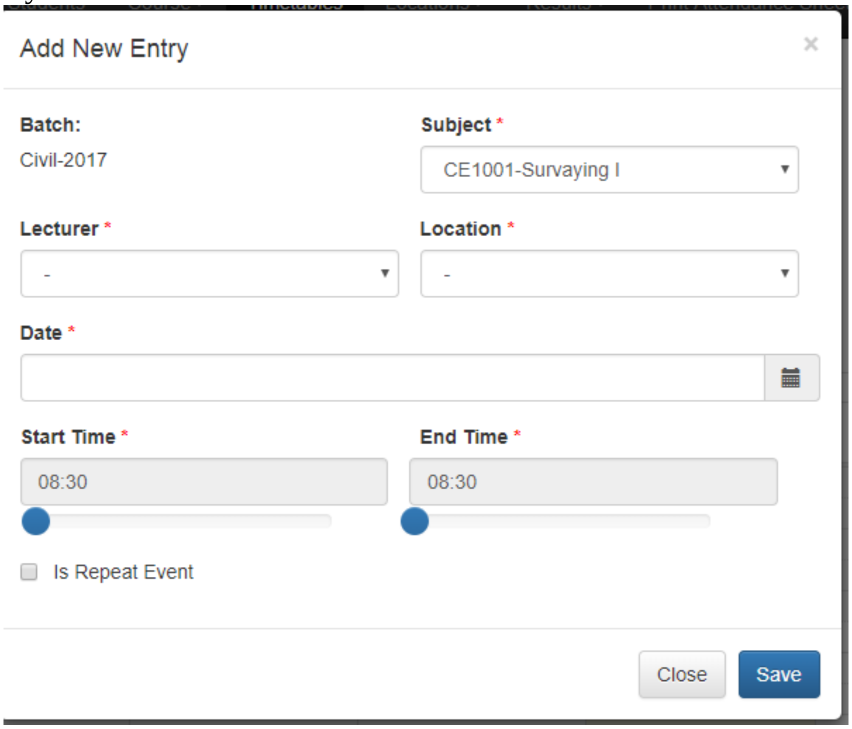
APPENDIX E - TEST RESULTS

Test results for User module in the HNDE student portal		
ID	**Test data inputs and received results with Screen shots**	**Status**
1	Enter “Admin” for user name and “test” for password which are invalid  Invalid username or password	✓
2	Without enter any username or password  The username field is required. The password field is required.	✓
3	Enter wrong format for Email  Email* aa Mobile No* Please include an '@' in the email address. 'aa' is missing an '@'.	✓
4	Enter wrong confirm password in student signup  The Password field does not match the Confirm Password field.	✓
4	Enter all information of student signup  You have successfully signed up on HNDE Students Portal. You will receive a verification email shortly. Please check your inbox. If you still not received a verification email, Click here to resend it.	✓
5	Select status” Active” and click view student	✓

<table>
<tr><td></td><td>
Search: Serch by reg no, name or en
Batch: 2016ele
Status: Active
View
<table>
<tr><th>Student Name</th><th>Reg No</th><th>Email</th><th>Mobile No</th><th>Enrolled Course</th><th>Status</th><th>Action</th></tr>
<tr><td>Gkr</td><td>001</td><td>gkr@gmail.com</td><td>111</td><td>sdsd</td><td>Active</td><td>Edit Delete</td></tr>
<tr><td>Nimal</td><td>001</td><td>a@yahoo.com</td><td>011</td><td>eeeee</td><td>Active</td><td>Edit Delete</td></tr>
</table></td><td></td></tr>
<tr><td>6</td><td>Click the generate new password button to enter the pass word in add staff page

Password*
Generate new password</td><td>✓</td></tr>
<tr><td>7</td><td>Enter all detail correctly

Success! New staff member has been successfully created.</td><td>✓</td></tr>
<tr><td>8</td><td>Select staff from staff list and click “Edit”

Update Staff Profile.
Full Name*: GKR de silva
Email*: gkr@gmail.com
Mobile No*: 11111
Designation*: Demonstrator
Exit Update</td><td>✓</td></tr>
</table>

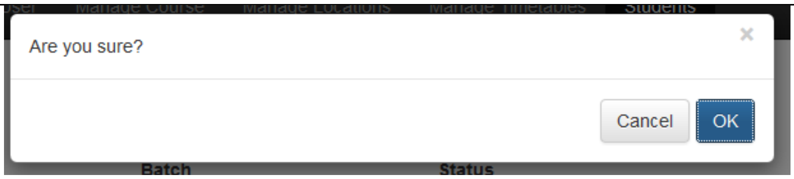
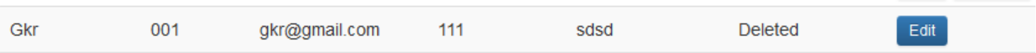
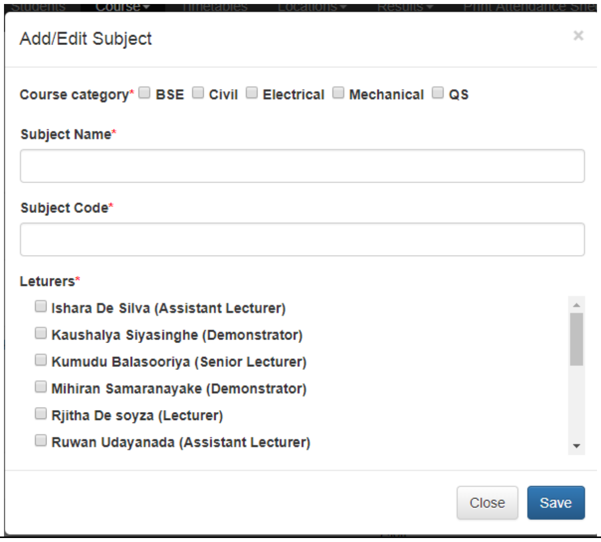
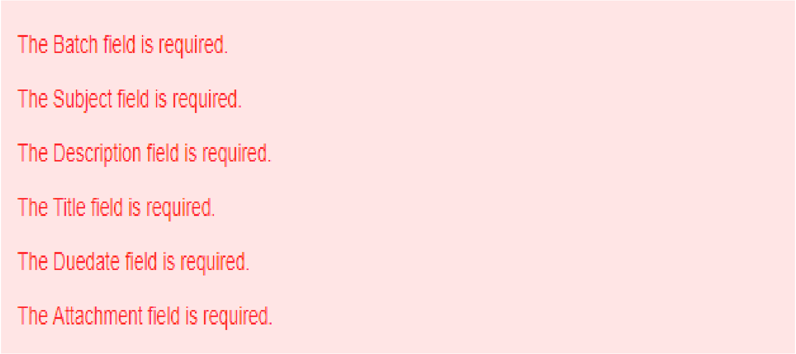
<table>
<tr><th colspan="3">Test results for timetable module in the HNDE student portal</th></tr>
<tr><td>9</td><td>Select the batch “sdsd 2016” and view on the time table window

Time Table
Batch
sdsd 2016

October 2016

<table>
<tr><th>Mon</th><th>Tue</th><th>Wed</th><th>Thu</th><th>Fri</th></tr>
<tr><td>3
9:32a xx IT LAB 1</td><td>4
9:32a xx IT LAB 1
12:57p fff ELECTRICAL LAB 1</td><td>5
9:32a xx IT LAB 1</td><td>6
9:32a xx IT LAB 1</td><td>7
9:32a xx IT LAB 1</td></tr>
<tr><td>10
9:32a xx IT LAB 1</td><td>11
9:32a xx IT LAB 1
12:57p fff ELECTRICAL LAB 1</td><td>12
9:32a xx IT LAB 1</td><td>13
9:32a xx IT LAB 1</td><td>14
9:32a xx IT LAB 1</td></tr>
<tr><td>17
9:32a xx IT LAB 1</td><td>18
9:32a xx IT LAB 1</td><td>19
9:32a xx IT LAB 1</td><td>20
9:32a xx IT LAB 1</td><td>21
9:32a xx IT LAB 1</td></tr>
</table>
</td><td>✓</td></tr>
<tr><td>10</td><td>Subject,Lecture,Location,Date,From time,To time,tick is repeat event,Select frequency and click save

Add New Entry

Batch:
Civil-2017
Subject *
CE1001-Survaying I
Lecturer *
-
Location *
-
Date *
Start Time *
08:30
End Time *
08:30
☐ Is Repeat Event

</td><td>✓</td></tr>
</table>

	 Are you sure? Click OK  Gkr 001 gkr@gmail.com 111 sdsd Deleted 	
Test results for course module in the HNDE student portal		
11	Click on “Add New” button in the manage subject page. Select the course category and enter the subject name IT and subject code IT001.Then select the assign lecturer and click save.  Add/Edit Subject Course category* □ BSE □ Civil □ Electrical □ Mechanical □ QS Subject Name* Subject Code* Leturers* □ Ishara De Silva (Assistant Lecturer) □ Kaushalya Siyasinghe (Demonstrator) □ Kumudu Balasooriya (Senior Lecturer) □ Mihiran Samaranayake (Demonstrator) □ Rjitha De soyza (Lecturer) □ Ruwan Udayanada (Assistant Lecturer) 	✓
12	With out enter assignment detail  The Batch field is required. The Subject field is required. The Description field is required. The Title field is required. The Duedate field is required. The Attachment field is required.	✓

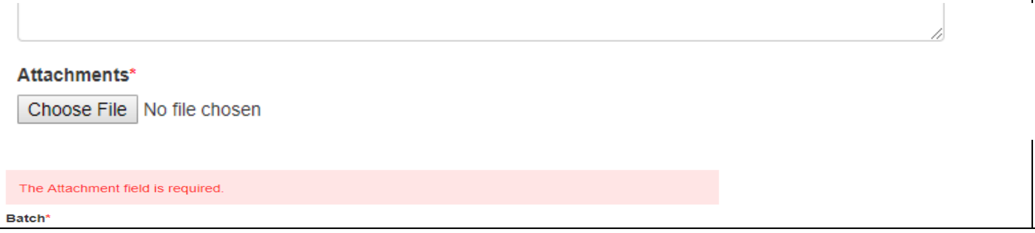
13	Without upload assignment file  Attachments* Choose File No file chosen The Attachment field is required. Batch*	✓

Table E.1Test results

APPENDIX F - CODE LISTING

Code listing appendix consists of an overview of the codes used for the development of the system.

LOGIN VIEW

```
<div class="container">
    <div class="row">
        <div class="col-md-12"></div>
    </div>
    <div class="row row-login-form">
        <div class="col-md-4 col-md-offset-4">
            <div class="login-panel panel panel-default">
                <div class="panel-heading">
                    <h2 class="panel-title" style="font-size:25px;">Sign In</h2>
                </div>
                <div class="panel-body">

                    <?php if(validation_errors()) {?>
                        <div class="validation-errors">
                            <?php echo validation_errors();?>
                        </div>
                    <?php } ?>

                    <?php echo form_open('/login/index'); ?>
                        <div class="form-group">
                            <label for="inputEmail">Username</label>
                            <input type="test" class="form-control" id="username"
name="username" placeholder="username">
                        </div>

                        <div class="form-group">
                            <label for="inputPassword">Password</label>
                            <input type="password" class="form-control" id="inputPassword"
name="password" placeholder="Password">
                        </div>

                        <a href="/student/signup" class="btn btn-warning">Signup as a
Student</a>
                        <button type="submit" style="float:right;" class="btn btn-
primary">Login</button>
                    </form>
                </div>
            </div>
        </div>
    </div>
</div>
```

CREATE STAFF

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');?>

<div class="container-fluid">
    <div class="row">
        <div class="col-md-8">

            <h3 class="text-muted">
                Create Staff Member.
            </h3>

            <?php if(validation_errors()) {?>
                <div class="validation-errors">
                    <?php echo validation_errors(); ?>
                </div>
            <?php } ?>

            <form role="form" name="manage_ac_user_form" method="post"
action="/admin/staff/new">

                <div class="form-group">
                    <label for="full_name">Full Name<span
class="required">*</span></label>
                    <input type="text" class="form-control" id="full_name" name="full_name"
value="<?=set_value('full_name')?>"/>
                </div>

                <div class="form-group">
                    <label for="email">Email<span class="required">*</span></label>
                    <input type="email" class="form-control" id="email" name="email"
value="<?=set_value('email')?>"/>
                </div>

                <div class="form-group">
                    <label for="password">Password<span class="required">*</span></label>
                    <input type="password" class="form-control" id="password"
name="password"/>
                </div>

                <div class="form-group">
                    <label for="confirm_password">Confirm Password<span
class="required">*</span></label>
                    <input type="password" class="form-control" id="confirm_password"
name="confirm_password"/>
                </div>

                <div class="form-group">
                    <label for="mobile_no">Mobile No<span
class="required">*</span></label>
                    <input type="telephone" class="form-control" id="mobile_no"
name="mobile_no" value="<?=set_value('mobile_no')?>"/>
                </div>

                <div class="form-group">
                    <label for="staff_type">Designation<span
class="required">*</span></label>
                    <select class="form-control" name="designation" id="designation">
                        <option value="0">Select</option>
                        <?php foreach($designations as $desg) {?>
                            <option value="<?=$desg->id?>" <?= $desg->id ==
set_value('designation') ? 'selected="selected"' : '';?>><?=$desg->designation?></option>
                        <?php } ?>
                    </select>
                </div>

                <button type="reset" class="btn btn-danger">Cancel</button>&nbsp;&nbsp;
                <button type="submit" class="btn btn-primary">Create</button>
            </form>
        </div>
    </div>
</div>
```

MANAGE COURSE

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');?>
```

```
<script type="text/javascript" src="/assets/js/admin_course.js"></script>

<div class="col-md-10">
    <h3 class="text-muted">Manage Courses</h3>
    <div class="dataTable_wrapper">
        <table class="table table-striped table-hover" id="subjects-table">
            <thead>
                <tr>
                    <th>Course Name</th>
                    <th>Course Code</th>
                    <th>Start Date</th>
                    <th>Duration</th>
                    <th>Number of Students</th>
                    <th>Action</th>
                </tr>
            </thead>

            <tbody>

                <?php foreach($list as $course) {?>
                    <tr>
                        <td><?=$course->name?></td>
                        <td><?=$course->code?></td>
                        <td><?=date('d M Y', strtotime($course->start_date))?></td>
                        <td><?=$course->duration?></td>
                        <td><?=$course->student_count?></td>
                        <td>&nbsp;</td>
                        <td>
                            <a href="/admin/course/edit/<?=$course->id?>" class="btn btn-sm btn-
primary">Edit</a>
                            <a href="/admin/course/settings  /<?=$course->id?>" class="btn btn-sm
btn-success">Set Current Semster</a>
                            <?php /*<a href="/admin/course/status/complete/<?=$course->id?>"
class="btn btn-sm btn-danger">Complete</a> */?>
                        </td>
                    </tr>
                <?php } ?>
            </tbody>
        </table>
    </div>
</div>
```

USER MODLE

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class User_model extends CI_Model
{
        //allowed login types.
        private $login_types = array('student', 'lecturer', 'admin');
        private $table = 'users';

        public function __construct()
        {
                parent::__construct();
        }

        public function get($id = 0) {
    $this->db->where('id', $id);
    $query = $this->db->get($this->table);
    return $query->first_row();
  }

        public function insert($data = array()) {
                if (!$this->db->insert($this->table, $data)) {
      return false;
    }
    return $this->db->insert_id();
        }
    public function update($id, $data = array()) {
  $this->db->where('id', $id);
  return $this->db->update($this->table, $data);
    }



    function login($username, $password)
    {
        $this->db->select('id, user_type_id, username, password, full_name, email');
        $this->db->from('users');
        $this->db->where('username', $username);
        $this->db->where('password', md5($password));
        $this->db->limit(1);

        $query = $this->db->get();

        if ($query->num_rows() == 1) {
            return $query->result();
        } else {
            return false;
        }
    }
}
```

USER MODLE

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class User_model extends CI_Model
{
        //allowed login types.
        private $login_types = array('student', 'lecturer', 'admin');
        private $table = 'users';

        public function __construct()
        {
                parent::__construct();
        }

        public function get($id = 0) {
      $this->db->where('id', $id);
      $query = $this->db->get($this->table);
      return $query->first_row();
   }

        public function insert($data = array()) {
                if (!$this->db->insert($this->table, $data)) {
        return false;
      }
      return $this->db->insert_id();
        }

        public function update($id, $data = array()) {
```

```
        $this->db->where('id', $id);
        return $this->db->update($this->table, $data);
           }



           function login($username, $password)
           {
                    $this->db->select('id, user_type_id, username, password, full_name, email');
                    $this->db->from('users');
                    $this->db->where('username', $username);
                    $this->db->where('password', md5($password));
                    $this->db->limit(1);

                    $query = $this->db->get();

                    if ($query->num_rows() == 1) {
                             return $query->result();
                    } else {
                             return false;
                    }
           }
}
```

STAFF MODEL

```
           <?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
           class Staff_model extends CI_Model
```

```
{
            //allowed login types.

            private $table = 'staff';

            public function __construct()
            {
                        parent::__construct();
            }

            public function insert($data = array()) {
                        if (!$this->db->insert($this->table, $data)) {
            return false;
        }
        return $this->db->insert_id();
            }

            public function update($id, $data = array()) {
        $this->db->where('id', $id);
        return $this->db->update($this->table, $data);
            }

    public function update_by_userid($user_id, $data = array()) {
        $this->db->where('user_id', $user_id);
        return $this->db->update($this->table, $data);
    }

    public function get_staff_profile($user_id) {
        $query = $this->db->select('staff.*, user.full_name, user.email, user.mobile_no,
sd.designation')
            ->join('users user', 'staff.user_id =  user.id', 'inner')
            ->join('staff_designations sd', 'sd.id = staff.staff_designation_id', 'left')
            ->where('user.user_type_id', 2)
            ->where('user.status', 1)
            ->where('staff.user_id', $user_id)
            ->get('staff');
        return $query->row();
    }

    public function get_stffs() {
        $query = $this->db->select('staff.*, user.full_name, user.email, user.mobile_no,
sd.designation')
            ->join('users user', 'staff.user_id =  user.id', 'inner')
```

```
                ->join('staff_designations sd', 'sd.id = staff.staff_designation_id', 'left')
                ->where('user.user_type_id', 2)
                ->where('user.status', 1)
                ->order_by('user.full_name')
                ->get('staff');

            return $query->result();
        }
    }
```

ASSIGNMENT MODEL

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class Assignment_model extends CI_Model {

     private $table = 'assignments';

     public function __construct() {
             parent::__construct();
     }

     public function get($id = 0) {
    $this->db->where('id', $id);
    $query = $this->db->get($this->table);
    return $query->first_row();
  }

     public function insert($data = array()) {
             if (!$this->db->insert($this->table, $data)) {
      return false;
    }

    return $this->db->insert_id();
     }

     public function update($id, $data = array()) {
    $this->db->where('id', $id);
    return $this->db->update($this->table, $data);
     }

  public function get_assigment_details($lecturer_id) {
    $sql  = "SELECT a.*, aa.file_name, s.name subject_name, c.name course_name,
c.start_date course_start FROM assignments a ";
    $sql .= "LEFT JOIN assignment_attachments aa ON aa.assignment_id = a.id ";
    $sql .= "LEFT JOIN subjects s ON s.id = a.subject_id ";
    $sql .= "LEFT JOIN courses c ON c.id =  a.batch_id ";
    $sql .= "WHERE a.lecturer_id = ? ";

    $query = $this->db->query($sql, array($lecturer_id));
    return $query->result();
  }

  public function get_by_student($student_user_id) {

    $sql = "SELECT course_id FROM students WHERE user_id = ? LIMIT 1";
    $query = $this->db->query($sql, array($student_user_id));
    $stu = $query->row();

    if(is_object($stu)) {
      $sql  = "SELECT ass.*, sub.name AS subject_name, sub.code AS subject_code FROM
assignments AS ass ";
      $sql .= "LEFT JOIN subjects sub ON sub.id = ass.subject_id ";
      $sql .= "WHERE ass.status IN (1,2) AND ass.batch_id = ? ";
      $sql .= "ORDER BY ass.due_date DESC ";

      $query = $this->db->query($sql, array($stu->course_id));
      return $query->result();
    }
    return array();
  }

}
```

MANAGE TIME TABLE

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Admin_manage_timetable extends MY_Controller {

         public function __construct() {
                    parent::__construct();
        $this->set_topnav('manage_timetable');
```

```
         }

    public function index() {
        $data = array();

        $course_id = (int)$this->input->get('filter_course_id');
        $semester_id = (int)$this->input->get('semester_id');

        $this->load->model('course_model');
        $this->load->model('course_semester_model');
        $this->load->model('course_subject_model');
        $this->load->model('staff_model');
        $this->load->model('location_model');

        $data['locations'] = $this->location_model->get_locations();
        $data['courses'] = $this->course_model->get_course_list();

        if((int)$course_id <= 0 && count($data['courses']) > 0 ) {
            $first_course = reset($data['courses']);
            $course_id = $first_course->id;
        }

        $data['course_id'] = $course_id;

        if($course_id > 0 ) {
            $data['course'] = $this->course_model->get($course_id);
            $data['lecturers'] = $this->staff_model->get_stffs();

            if(is_object($data['course']) && $data['course']->current_semester_id > 0) {
                $data['semester'] = $this->course_semester_model->get($data['course']-
>current_semester_id);

                if(is_object($data['semester'])) {
                    $data['subjects'] = $this->course_subject_model-
>get_by_semester($data['semester']->id);
                }
            }
        }

        $this->layout->view('/admin/timetable/index', $data);

    }
```

```
public function load_events() {

    $this->load->model('timetable_model');

    $response =  [];
    $course_id = $this->input->post('course_id');

    $event_data_all = $this->timetable_model->get_events($course_id);

    $events = [];
    foreach($event_data_all as $event) {

        $title  = $event->subject_name ;
        $title .= ' ' . $event->location_name;
        $events[] = array(
            'id' => $event->id,
            'title' => $title,
            'start' => $event->date . 'T' . $event->time_from,
            'end' => $event->date . 'T' . $event->time_to,
            'allDay' => false
        );
    }

    $response['events'] = $events;

    echo json_encode($events);
    exit;
}

public function save_event() {

    $this->load->library('form_validation');
    $this->load->model('timetable_model');

    $data = [];

    $this->form_validation->set_rules('batch_id', 'Batch', 'trim|required|xss_clean');
    $this->form_validation->set_rules('subject_id', 'Subject', 'trim|required|xss_clean');
    $this->form_validation->set_rules('lecturer_id', 'Lecturer', 'trim|required|xss_clean');
    $this->form_validation->set_rules('location_id', 'Location', 'trim|required|xss_clean');
    $this->form_validation->set_rules('lecturer_id', 'Lecturer', 'trim|required|xss_clean');
    $this->form_validation->set_rules('event_date', 'Date', 'trim|required|xss_clean');
```

```
        $this->form_validation->set_rules('event_start_time', 'Start Time', 
'trim|required|xss_clean');
        $this->form_validation->set_rules('event_end_time', 'End Time', 
'trim|required|xss_clean');


        $is_repeatable = $this->input->post('is_repeatable');
        $is_repeatable_event  = false;
        if(!empty($is_repeatable) && $is_repeatable == '1') {
          $this->form_validation->set_rules('repeat_end_date', 'Repeat Until', 
'trim|required|xss_clean');
          $this->form_validation->set_rules('event_repeat_frequency', 'Frequency', 
'trim|required|xss_clean');
          $is_repeatable_event = true;

        }

        if($this->form_validation->run() === true) {

          $event_date = convert_db_date_format($this->input->post('event_date'));
          $event_start_time = $this->input->post('event_start_time');
          $event_end_time = $this->input->post('event_end_time');

          $save_events =  [];
          $save_event = array(
            'course_id' => $this->input->post('batch_id'),
            'subject_id' => $this->input->post('subject_id'),
            'date' => $event_date,
            'time_from' => $this->input->post('event_start_time'),
            'time_to' => $this->input->post('event_end_time'),
            'lecturer_id' => $this->input->post('lecturer_id'),
            'location_id' => $this->input->post('location_id'),
            'parent_event_id' => 0
          );

          $save_events[0] = $save_event;

          if($is_repeatable_event) {
            $repeat_end_date = convert_db_date_format($this->input->post('repeat_end_date'));
            $event_repeat_frequency = $this->input->post('event_repeat_frequency');

            $next_date = $event_date;
            do {
                    if($event_repeat_frequency == 'daily') {
                        $next_date = date('Y-m-d', strtotime("$next_date +1 Day"));

                    } elseif($event_repeat_frequency == 'weekly') {
                        $next_date = date('Y-m-d', strtotime("$next_date +7 Day"));

                    } else {
                        $next_date = $repeat_end_date;
                    }

                    //ignore if next day is a weekend.
                    if($event_repeat_frequency == 'daily' && in_array(date('w',
strtotime($next_date)), array(0, 6))) {
                        continue;
                    }

                    $save_event['date'] = $next_date;
                    $save_event['parent_event_id'] = -1;
                    $save_events[] = $save_event;

                } while ($next_date < $repeat_end_date);
            }

            $parent_event_id = 0;
            foreach($save_events as $key => $event) {

                $event['parent_event_id'] = $parent_event_id;
                $eid = $this->timetable_model->insert($event);

                //parent event
                if($key == 0) {
                    $parent_event_id = $eid;
                }
            }

            if($parent_event_id > 0 ) {
                $data['success'] = 1;
            }
        } else {
            $data['errors'] = validation_errors();
        }
        echo json_encode($data);
        exit;
      }
  }
```

JAVA SCRIPT

```
<script type="text/javascript">
  $(document).ready(function() {
    $('.btn-approve').on('click', function() {
      var id = $(this).data('id');
      admin_student_update_status(id, 1, 'approve', $(this));
    });

    $('.btn-delete').on('click', function() {
      var id = $(this).data('id');
      var btn = $(this);
      bootbox.confirm("Are you sure?", function(confirm){
        if(confirm) {
          admin_student_update_status(id, 3, 'delete', btn);
        }
      })

    });

    function admin_student_update_status(id, status, newstatus, btn) {
      $.ajax({
        url: '/admin/admin_manage_user/update_user_status',
        type: 'post',
        data:   {'id': id, 'status': status},
        dataType:'json',
        success: function(data) {
          if(data.status == '1') {
            var newstatus_text = '';

            if(newstatus == 'approve') {
              newstatus_text = 'Approved';

            } else if(newstatus == 'delete') {
              newstatus_text = 'Deleted';
            }
            $('#stu-row-' + id).find('.stu-status').html(newstatus_text);
            btn.hide();
          } else {
            alert('An error occured. Please refresh the page and try.');
          }
        }
      })
    }
  });
</script>
```

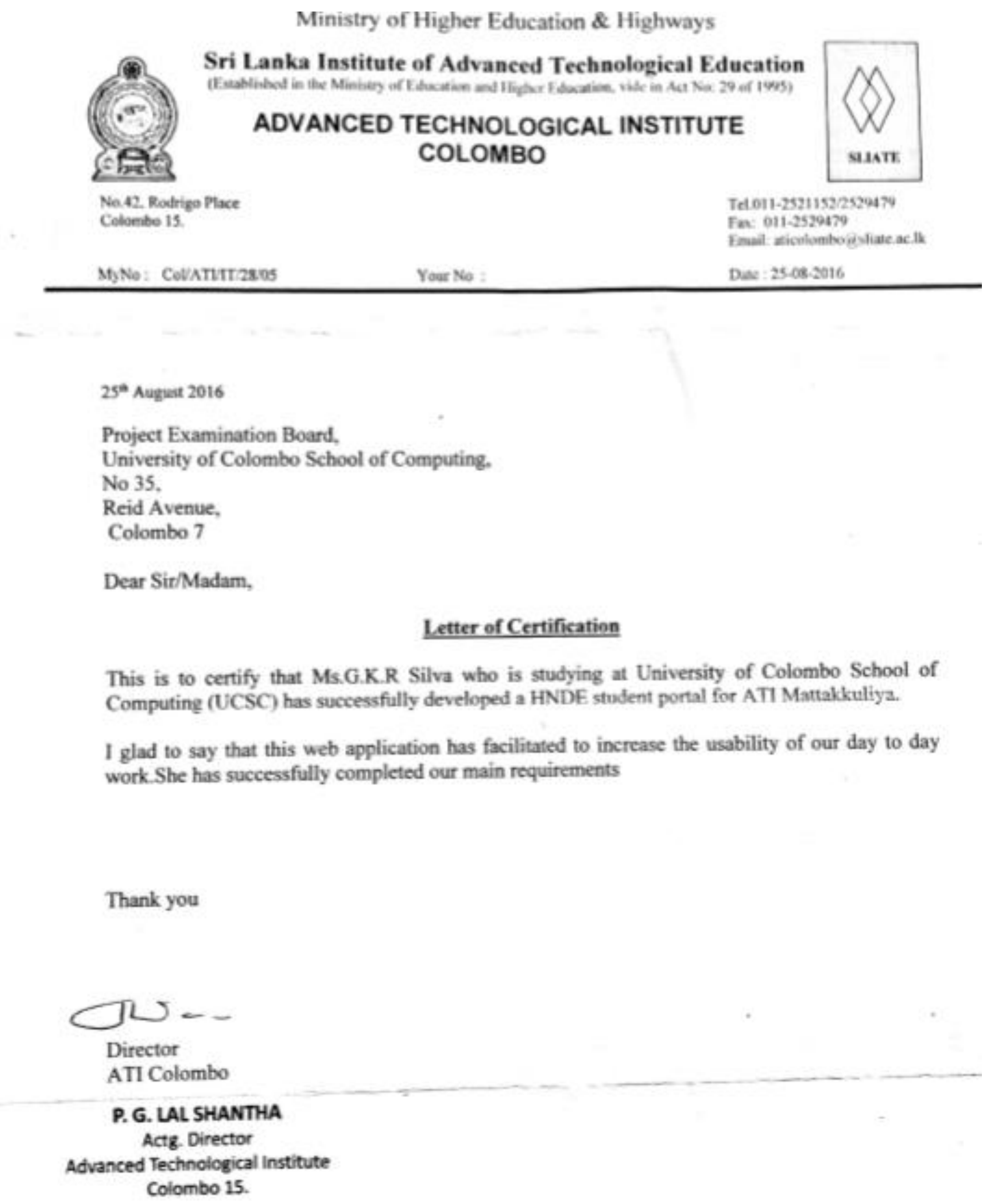
CASCADE STYLE SHEET

```
/* Sticky footer styles
-------------------------------------------------- */
html, body{
  height:100%
}
```

```
body {
   font-family: "Helvetica Neue", Arial, sans-serif;
}

footer {
   position: relative;
   margin-top: -60px;; /* negative value of footer height */
   height: 60px;
   clear:both;
   text-align: center;
   background-color: #333;
}


/* Custom page CSS
-------------------------------------------------- */

body > .container {
   padding: 60px 15px 0;
}

#main-content-wrap{
   overflow:auto;
   overflow-x: hidden;
   min-height: 100%;
   padding-bottom:60px;
   padding-top: 50px;
}
```

```
.main-content-wrap > .row {
   background-color: #0099cc;
}

#main-content-wrap > .row > #right-container{
   background-color: #fff;
}

#left-nav {
   padding-right: 0;
}
#left-nav ul.navbar-nav{
   width: 100%;
}

#left-nav li {
   width: 100%;
   border-bottom: 1px solid #ccc;
}
#left-nav li a {
   text-decoration: none;
   color: #fff;
}

#left-nav li > a:hover, #left-nav li > a:active {
   background-color: #ccc;
}

.text-muted {
   /*background-color: #ccc;*/
   margin-bottom: 30px;
   padding: 8px 0;
   text-align: center;
   text-decoration: underline;
}

.bold{
   font-weight: bold;
}
}
.validation-errors{
   background-color: #ffe5e5;
   color: #ff2828;
   margin-bottom: 8px;
   padding: 14px 10px 1px 13px;
}
.red {
   color: #ff2828;
}
.row-login-form{margin-top:20px;}

#course_semsters_container {
   border-top: 1px solid #ccc;
   padding-top: 20px;
}
```

APPENDIX G-CLIENT CERTIFICATE



Ministry of Higher Education & Highways

Sri Lanka Institute of Advanced Technological Education
(Established in the Ministry of Education and Higher Education, vide in Act No: 29 of 1995)

ADVANCED TECHNOLOGICAL INSTITUTE
COLOMBO

SLIATE

No.42, Rodrigo Place
Colombo 15.

Tel.011-2521152/2529479
Fax: 011-2529479
Email: aticolombo@sliate.ac.lk

MyNo : Col/ATI/IT/28/05 Your No : Date : 25-08-2016

25th August 2016

Project Examination Board,
University of Colombo School of Computing,
No 35,
Reid Avenue,
Colombo 7

Dear Sir/Madam,

Letter of Certification

This is to certify that Ms.G.K.R Silva who is studying at University of Colombo School of Computing (UCSC) has successfully developed a HNDE student portal for ATI Mattakkuliya.

I glad to say that this web application has facilitated to increase the usability of our day to day work.She has successfully completed our main requirements

Thank you

Director
ATI Colombo

P. G. LAL SHANTHA
Actg. Director
Advanced Technological Institute
Colombo 15.