****

**HAWASSA UNIVERSITY**

**FACULTY OF INFORMATICS**

**TITLE: ETHIOPIAN DRIVING LICENSE LEARNING WEBSITE**

**PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF REQUIREMENT FOR BACHELER OF SCIENCE (B.Sc.) IN INFORMATION SYSTEM**

**Group Members**

|  |  |  |
| --- | --- | --- |
| **No** | **Name** | **ID** |
| 1 | Redait Demissie | IS/0104/08 |
| 2 | Birhanu Achamyelew | IS/0027/08 |
| 3 | Sena Mulugeta | IS/0112/08 |
| 4 | Ayelech Assefa | IS/0019/08 |
| 5 | Mohammed Adem | IS/0088/08 |

**ADVISOR DATE SIGNATURE**

Mr. **ABEBAYEHU SAMUEL**  \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_

# ACKNOWLEDGMENT

First of all, we would like to thank **God** for guiding us in different ways we know or we didn’t know to be able to complete this task. Secondly, we express our great thanks and gratitude to our advisor **Mr. Abebayehu Samuel**, instructor in Hawassa University for his day to day encouraging support and guidance in carrying out the project.

We would also like to express gratitude and appreciation to our **department members** and **class mates** for their valuable advice and guidance. And we would like to thank our **parents, Friends** and those who helped us directly and indirectly for the successful completion of this project work. However, it would not have been possible without the Kind support and help of many individuals and organizations. We would like to extend our honest thanks to all of them.

# ABSTRACT

The main purpose of this project is to develop web based driving license learning system that can cover all the theoretical part of the driving learning system. Now a days we are living in an information age, so everything is changed from the manual system to automated system, which makes everything simple, interactive, time saving and requires less storage space for allocating resources. To do this proposal the project team used different data collection methods such as Document analysis and observation. In order to analyze and design the system we use Object oriented approach for both analyzing and designing the new system. To develop the application the team has passed through numerous phases that constitute to the development of an application starting from system analysis up to system implementation and testing. Since the current system is manual, to change this system to web based, we used different software and hardware tools like PHP, sublime text editor, XAMPP different flashes and some others.

**Contents**

[ACKNOWLEDGMENT 2](#_Toc11080183)

[ABSTRACT 3](#_Toc11080184)

[CHAPTER ONE 6](#_Toc11080185)

[1. Introduction 6](#_Toc11080186)

[1.1 Objective of the implementation 7](#_Toc11080187)

[CHAPTER TWO 8](#_Toc11080188)

[Algorithm design 8](#_Toc11080189)

[2. Introduction 8](#_Toc11080190)

[2.1 Algorithm Design 8](#_Toc11080191)

[CHAPTER THREE 12](#_Toc11080192)

[Coding procedures 12](#_Toc11080193)

[3.1 Coding standard 12](#_Toc11080194)

[Coding conventions serve the following purposes: 12](#_Toc11080195)

[CHAPTER FOUR 18](#_Toc11080196)

[Testing and debugging procedures 18](#_Toc11080197)

[4.1 Introduction 18](#_Toc11080198)

[4.2 Testing plan 18](#_Toc11080199)

[4.3 Testing approaches 18](#_Toc11080200)

[4.3.1 Black box testing 18](#_Toc11080201)

[CHAPTER FIVE 21](#_Toc11080202)

[Security of the system 21](#_Toc11080203)

[5.1. Database security (Discretionary Access Control and Mandatory Access Control) 21](#_Toc11080204)

[Mandatory Access Control 21](#_Toc11080205)

[System Security 22](#_Toc11080206)

[Availability 23](#_Toc11080207)

[Confidentiality 23](#_Toc11080208)

[CHAPTER SIX 24](#_Toc11080209)

[Conclusion 24](#_Toc11080210)

[6.1 Recommendation 25](#_Toc11080211)

[REFERENCES 25](#_Toc11080212)

# List of Acronyms

MySQL…………………………………………… My Structural Query Language

GUI ………………………………………………Graphical User Interface

HTML………………………………………………Hypertext Markup Language

DBMS…………………………………………….Database Management System

DAC……………………………………………………… Discretionary access control

MAC……………………………………………………… Mandatory access control

OOP………………………………………………….Object Oriented Programming

OOP………………………………………………….Object Oriented Programming

UI ……………………………………………………..User Interface

XML……………………………………………………eXtensible Markup Language

PHP……………………………………………………..Personal Home Page

# CHAPTER ONE

## Introduction

During this phase physical design specification must be turned into working computer code, and provide help for current and future users and take care of the system. The implementation is to change what we did in the design Phased to machine readable form by writing code using php to all modules. We also describe the help and support contents, user manual, Naming and coding standards. The system contains many forms that are connected to the data base in each individual form also combined in one module in order to work the system as whole. And also Implementation is the phase where objectives of physical operations of the system turned into reality i.e. real working model. The crucial phase in the system development life cycle is the successful implementation of the new system design. The process of converting as new system into an operational one is known as system implementation. This includes all those activities that take place to convert from an old system to a new system.

And then the code is tested until most of the errors have been detected and corrected. The purpose of this activity is to convert the final physical system specification into working model with reliable software and hardware. The system is implemented using PHP, HTML and uses MySQL for database. End users can access the system data only through the application layer which implements appropriate business rules to allow only authorized operations based on the actual user permissions; no direct access to the physical database is needed.

In this document we mainly focuses on the implementation part, implementation concerned techniques to develop the system, algorithm for the system, code samples of the system, data preparation, how to install the system, some testing techniques, startup strategy for the new installed system are briefly described in this part of documentation.

The project will have a convenient user interface for users to easily interact with the system. All group members believe that after the implementation of the project the working system of the organization will be satisfactory and effective.

In general the objective of this project is to develop an efficient and effective Driving license learning website that can every one use it easily.

## 1.1 Objective of the implementation

The objective of systems implementation phase is to convert the final physical system specification in to working and reliable software and hard ware, document the word that has been done, and provide help for current and future users.

# CHAPTER TWO

# Algorithm design

## Introduction

In this part we describe the algorithm of the operations or methods which found in class diagram using Pseudo code. Pseudo code is compact and informal high-level description of a computer programming algorithm that uses the structural conventions of a programming language but is intended for human reading rather than machine reading. Pseudo code typically omits details that are not essential for human understanding of the algorithm, such as variable declaration, system-specific code and subroutine. The programming language is augmented with natural language descriptions of the details, where convenient, or with compact mathematical notation. The purpose of using pseudo code is that it is easier for humans to understand than conventional programming language code, and that it is a compact and environment-independent description of the key principles of an algorithm.

## Algorithm Design

**1 Pseudo code insert content**

**Steps/procedure**

**Begin**

Variables –content id

- Title

- Content

**If** (\*variables are valid\*)

***Then***

Add to table content (content id, Title, Content)

***Otherwise***

Display “the inputs are invalid!”

**End if**

Display “the success message”

**End**

**2 Pseudo code insert exam**

**Steps/procedure**

**Begin**

Variables – question number

- Question

- Choice #1

- Choice #2

- Choice #3

- Choice #4

- Answer

**If** (\*variables are valid\*)

***Then***

Add to table question (question\_number, question)

Add to table choices (question\_number, choice, Answer)

***Otherwise***

Display “the inputs are invalid!”

**End if**

Display “the success message”

**End**

**3 Pseudo code insert user**

**Steps/procedure**

**Begin**

Variables – email

- username

- password

**If** (\*variables are valid\*)

***Then***

Add to table user (email, username, password)

***Otherwise***

Display “the inputs are invalid!”

**End if**

Display “the success message”

**End**

**4 Pseudo code update content**

**Steps/procedure**

**Begin**

Variables –content id

- Title

- Content

**If** (\*variables are valid\*)

***Then***

Update to table content (content id, Title, Content)

***Otherwise***

Display “the inputs are invalid!”

**End if**

Display “the success message”

**End**

**5 Pseudo code Delete content**

**Steps/procedure**

**Begin**

Variables –content id

- Title

- Content

**If** (\*variables are valid\*)

***Then***

Delete to table content (content id, Title, Content)

***Otherwise***

Display “the inputs are invalid!”

**End if**

Display “the success message”

**End**

**6 Pseudo code for login**

**Steps/procedure**

**Begin**

Variables:-username

-password

**If** (\*inputs are valid\*)

(\*select the username, password from database and compare with entered\*)

**If** username = entered username and

Password = entered password

(\*go to login page\*)

Otherwise

Display “login error!”

**End if**

Otherwise

Display “inputs are invalid try again!”

**End if**

**End**

**7 Pseudo Code for Logout**

**Steps/procedure**

**Begin**

**Variables: - session ID**

**Destroy session**

**Redirect to the login Page**

**End**

# CHAPTER THREE

# Coding procedures

## 3.1 Coding standard

While implementing or coding, the team chooses different Naming and coding standards. As we described on introduction part for implementation we use php programing language and html. In php there are a lot of naming and coding standards for variables, methods. While deploying the system we need to consider some rules and conventions.

Coding conventions serve the following purposes:

* They create a consistent look to the code, so that readers can focus on content, not layout.
* They enable readers to understand the code more quickly by making assumptions based on previous experience.
* They facilitate copying, changing, and maintaining the code.
* There should be documentation content with multiline comments between method name and statements and variable name.
* All comments should be written in the same language, be grammatically correct, and contain appropriate

1. Use // or /// but never /\* … \*/

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Comment block

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. Do not use inline-comments to explain obvious code. Well written code is self-documenting.
2. Only use comments for bad code to say “fix this code” – otherwise remove, or rewrite the code!

Do not omit access modifiers. Explicitly declare all identifiers with the appropriate access modifier instead of allowing the default.

* Write only one declaration per line.

**$question\_number = $\_POST['question\_number'];**

**$question\_text = $\_POST['question\_text'];**

**$correct\_choice = $\_POST['correct\_choice'];**

**$choices = array();**

**$choices[1] = $\_POST['Choice1'];**

**$choices[2] = $\_POST['Choice2'];**

**$choices[3] = $\_POST['Choice3'];**

**$choices[4] = $\_POST['Choice4'];**

* While naming variables, methods and use descriptive words or understandable by users. Capitalize the first letter of standard acronyms. Example getPosts() function as follows:

**function getPosts(){**

**$posts = array();**

**$posts[0] = $\_POST['id'];**

**$posts[1] = $\_POST['email'];**

**$posts[2] = $\_POST['username'];**

**return $posts;**

**}**

And in the html File we used to naming controls by the following way:-

<form method="post" action="register.php" name="register" onsubmit="return Validate();" name="vform">

<div class="form-field">

<h3>ለመመዝገብ</h3>

</div>

<div class="form-field form-group">

<label>ኢሜይል አድራሻ</label>

<input type="email" name="email" id = "email" placeholder="you@example.com" class="form-control">

<div id="email\_error"></div>

</div>

<div class="form-field form-group">

<label>ስም</label>

<input type="text" name="username" id = "uName" placeholder="user Name" class="form-control" id="myrequired"/>

<div id="uname\_error"></div>

</div>

<div class="form-field form-group">

<label>ማለፊያ ቃል</label>

<input type="password" name="Password" id="sPass" class="form-control">

</div>

<div class="form-field form-group">

<label>ማለፊያ ቃል ማረጋገጫ</label>

<input type="password" name="Password\_2" id="scPass" class="form-control">

<div id="pas\_error"></div>

</div>

<input type="submit" value="መዝግብ" class="btn" name="register">

</form>

* Put the opening brace ({) at the end of the if condition, Loops with one line. And put the closing brace (}) at new line after the if statements executed.

public function addUser(){

$con = mysqli\_connect("localhost","root","","projectdb");

if(!$con){

echo 'Connection Lost';

die();

}

if(isset($\_POST['register'])){

$con = mysqli\_connect("localhost","root","","projectdb");

$username = $\_POST["username"];

$uemail = $\_POST["email"];

$upassword = $\_POST["Password"];

$uconfirm = $\_POST["Password\_2"];

$stmt = "insert into user (username , email, password) values ('$username', '$uemail','$upassword')";

if ($upassword == $uconfirm){

$result = mysqli\_query($con, $stmt);

if ($result){

echo "<script> alert(' $username በመመዝገቦ እናመሰግናለን ');</script>";

mysqli\_close($con);

}

}

else {

echo "<script> alert('Error Occured');</script>";

}

}

}

}

We follow the naming standard of object oriented programming naming convention to give name for the variables, methods and for each event and control tools.

* We give name for variables that the name of the variable contains alphabets only and most variable is start with small letter. If the name of the variable is the combination of two words we make the second letter of the word as capital letter. Example age, firstName, getPosts etc.
* Naming standard for the database and the tables that found in the database and the fields in the tables are looks like the following. The name of the database for our project is that of Projectdb which consists the name of the project, And the name of the tables is named as the convention for table that directly related with the tables function for example the name for table content that contains the of contents information of our project. The name for the fields in the table is directly the name of the variables example that is small letter for each word but if the name is contain more than one word for the variable we used the capital letter for the second words of the first character.
* According to this Coding Standard, our system consists of the following modules are:

1. **Administrator**

* User Registration and Login
* Update User account
* Delete User account
* View users
* View Report
* Content Adding
* Update content
* Delete content
* Add Exam
* Update Exam
* Delete Exam

1. **User**

* Register
* Update Profile
* View Content
* Take Exam
* View Report

# CHAPTER FOUR

# Testing and debugging procedures

## 4.1 Introduction

This chapter contains a test plan of Ethiopian Driving License Learing website description of the initial version of first web based application prototype, which will be developed as a course project. To test our system we have prepared a test plan, test cases, and testing procedures.

## 4.2 Testing plan

The test plan describes how Ethiopian Driving License learning website project developed and tested to ensure, or at least improve, correctness. Test plan should be developed priority and concurrently with the coding phase. The objective of the testing phase is to validate the application package, for functional and nonfunctional requirements of the project. Our plan contains the methods or approaches to test, unit testing, integration testing and system testing.

## 4.3 Testing approaches

We have used the following testing types in order to come up with the final system.

### 4.3.1 Black box testing

It is some time called a behavioral or partition testing. This kind of testing focuses on the functional requirements of the software. It enables one to drive sets of input conditions that will exercise all functional requirements for a program. Black box testing is the technique of testing without having any knowledge of the application. Typically, while performing a black box test, a tester will interact with the systems user interface by providing inputs and examining out puts without knowing how and where the inputs are worked upon.

##### System testing

System testing tests the system as a whole. Once all the components are integrated, the application as a whole is tested rigorously to see that it meets the specified quality standards. In this level of testing process we have examined how necessarily the sub system work together to achieve the desired goal.

We have tested our system using different test cases to all the components in all pages of the system. We enter correct and incorrect data, and select all optional selections and finally we observe the outcomes, the responses for incorrect inputs and the validations.

* AlphaTesting**:** Testing using correct input and sees for any failures. If failure occurs find and correct the bug.
* BetaTesting**:** In this testing method, team will force the system to be tested for incorrect data input. The System will be tested by the team members. If any failures occurred while testing the system in all the above testing methods, the team will take immediate correction beginning where this fault occurred before jumping to next work so that it will meet the goal. If all the above testing methods are carried out and find to be valid the System will directly deploy.

We have provided a sample test for our system below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name | | | Ethiopian Driving License Learning Website | | |
| Test Case id | | | ST\_id#1 | | |
| Test Priority | | | High | | |
| What to Test | | | Administrator Screen/Page | | |
| Test Title | | | Verify all components found when the Administrator log in | | |
| Test Description | | | Test the page for Administrator | | |
| Pre condition | | | Administrator must log in with valid user name and pass word | | |
| Steps | Test steps | | | Test Data | Expected Data |
| 1 | Administrator log in to her/his screen/page | | |  |  |
| 2 | he/she navigates to the manage user button | | |  |  |
| 3 | he/she navigates to the manage content button | | |  |  |
| 4 | he/she navigates to the manage content button | | |  |  |
| 5 | finally click the reports | | |  |  |
| Post Condition | | input data are validates and contents are created and permanently stored to data base | | | |

##### White box testing

White-box testing is the detailed investigation of internal logic and structure of the code. White-box testing is also called glass testing or open-box testing. In order to perform white-box testing on an application, a tester needs to know the internal workings of the code. The tester needs to have a look inside the source code and find out which unit of the code is behaving inappropriately.

##### Unit testing

Unit testing is a confusion part of software development process. It involves individually testing unit of code separately to make sure that it works on its own, independent of the other units. Unit testing is essentially a set of path, test performed to examine the several different paths through the modules. It is usually an automated process and performed within the programmers IDE and is normally performed by software developers themselves or their peers. In rare cases it may also be performed by independent software testers.

Unit testing is an action used to validate that separate units of source code remains working properly. We have tested our system individually for loops, methods, statements, variables, mysql query and php scripts.

Our unit testing contains main forms or modules and has their own different operations that making the system effective and easy to manage. In this process over all features of the system were tests using test cases. We have conducted testing activities to check if there are errors and to correct them by the help of our test cases prepared and some of them have provided here.

##### Integration testing

We have used a systematic technique for constructing our program structure while conducting test to uncover errors associated with interacting. We have tested the group of components or units to gather to evaluate or ensure their quality performance and correctness. In addition to this we have also tested the integration of all components or modules of our system.

# CHAPTER FIVE

Security of the system

## 5.1. Database security (Discretionary Access Control and Mandatory Access Control)

In discretionary access control (DAC), we specify which subject can access the role to the users of the system. This model is called discretionary because the control of access is based on the discretion of the owner.

## Mandatory Access Control

In mandatory access control the system (and not the users) specify which subjects can access specific data’s. The MAC model is based on security labels. Subjects are given a security clearance or role when the system is making an access control decision, it tries to match the role of the user.

System access control and security is the main issue of the project. Therefore system identify someone who is, what he/she wants to access, check whether the somebody is authorized or not and granting permission or deny. Thus the system provides protection over system from unauthorized users and allows those who have an authority to access the system and even an authorized body forgot either of his/her user name or password the system prompts to correct again. We are concerned with information security and application controls.

Access control is intended to specify the tasks that a specified user can and cannot do or can and cannot have access. These can be achieved by giving certain privilege restrictions by using access limit/control mechanism through username and password.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Role** | **Tables On Database** | | |  |
| **Account** | **Content** | **Exam** | **Result** |
| **Administrator** | Add account  Update account  Delete account | Add Content  Update Content  Delete Content | Add Exam  Update Exam  Delete Exam | Delete Result  Update result |
| **Student** | Update, create Account | Check content | Check Exam | Check Result |

Table : Role and Responsibity Table

Our database resources are trusted and we make a secure data base using the security systems in MYSQL that is the better choice to build a secured database also, we work using MYSQL at the back end server which has a vital rule for the security of our system. The reliability that the system gives the right result on a search. The system will be available 100% of the time. The MTBF *(*Mean time between failures*)* is required to be one week. Software must run on any browser. The system can run on any internet explorer.

### System Security

Any information system must be secure to fulfill the three main components of system security which are confidentiality, integrity and availability, therefore to make the system that we are going to develop secure, we are going to employ authentication mechanism which is giving the user username and password so that no one apart from the authenticated user can get access to the system. End users can access the system data only through the application layer which implements appropriate business rules to allow only authorized operations based on the actual user permissions; no direct access to the physical database is needed.

* The system shall require the user to identify him/herself with username and password.
* The system will identify if the administrator tries to log in to the system with a non-existing account then the administrator should not be logged in. The administrator should be notified about log-in failure.
* The system will identify if a student or user tries to log in to the system with a non-existing account then the student/user should not be logged in. The student/user should be notified about log-in failure.
* If a student/user wants to create an account and the desired user name is occupied, the user should be asked to choose a different user name.

### Availability

When we see how dependable the system in normal operating times it is 24hrs function for the users of the system. And if the database becomes unavailable and it will used a method of recovering from a system failure. In the event of a failure, the last checkpoint serves as a recovery point. When the system has been fixed, and it will take several days and the restart program loads the last checkpoint and starts the system with the next in line.

### Confidentiality

The only people to see the data those authorized to see it. Private data is kept private; personal privacy is respected.

# CHAPTER SIX

# Conclusion

We have concluded that Ethiopian Driving License Learning web based system is suit for learning of students for different functions. It creates a exams especially based on the students education. For the development of the system we used HTML, PHP, JAVASCRIPT and CSS.

In developing this project we contributed our full capability with maximum interest and get ways toward developing a project. Finally the team has come up with an website that can be used to perform basic learning services as stated in this document.

## Recommendation

According to scope of this project the team develops web based system. Because of the time constraint we may have limitation which should be consider in, but in the feature the team believes that this system should be fully operationally by adding some functionality that are not included in the proposed system for example video lectures, students follow up, certified students, support several languages and etc. The future recommendation is students to make this project expanded and more automated with additional functionalities and by integrating with many new technologies.

# REFERENCES

|  |  |
| --- | --- |
| [1] | [Online]. <https://en.wikipedia.org/wiki/Web_information_system> |
| [2] | [Online]. <http://sebokwiki.org/wiki/System_Verification> |
| [24] | [Online]. <https://en.wikipedia.org/wiki/First_normal_form> |
| [25] | ( 2016, , November,) wikipedia Encyclopedia. [Online]. <https://en.wikipedia.org/wiki/Web_information_system> |
| [26] | R. B. Jackson and S. D J. W. Satzinger, *Systems Analysis and Design in a Changing World*, 6th ed., Thomson D, Ed. boston, 2012. |
| [27] | WIKIPEDIA ENCYCLOPEDIA. [Online]. <https://en.wikipedia.org/wiki/Software_requirements_specification> |