

**6COSC007C.1 Enterprise Application Development**

**Coursework: 02**

**Informatics Institute of Technology**

**Department of Computing**

**BSc Computer Science (IIT Sri Lanka)**

Name: Shenal Anthony

IIT ID: 2018383

UOW ID: w1742306

**Table of Contents**

[1**.Design Changes and Reasons** 4](#_Toc92336004)

[2. **Evaluation: Implementation Choices** 8](#_Toc92336005)

[2.1 Maintainability and Readability 8](#_Toc92336006)

[2.2 Design Pattern 9](#_Toc92336007)

[2.3 Testability 10](#_Toc92336008)

[2.4 Security 12](#_Toc92336009)

[3. **User Interfaces** 13](#_Toc92336010)

**Table of Figures**

[Figure 1: Navigation Bar 4](#_Toc92336019)

[Figure 2 : Project Assign fields 5](#_Toc92336020)

[Figure 3:Create form 6](#_Toc92336021)

[Figure 4 : Alert Box 6](#_Toc92336022)

[Figure 5 : Confirmation Box 7](#_Toc92336023)

[Figure 6 : Company Update 7](#_Toc92336024)

[Figure 7: Comments On the code 8](#_Toc92336025)

[Figure 8 : File Hierarchy 9](#_Toc92336026)

[Figure 9 : MVC Structure 10](#_Toc92336027)

[Figure 10 : Array used in the system 10](#_Toc92336028)

[Figure 11 : Console messages and alert messages 11](#_Toc92336029)

[Figure 12 : Authorization to access the navigation features 12](#_Toc92336030)

[Figure 13 : Home Page 13](#_Toc92336031)

[Figure 14 : Manage Company Page 13](#_Toc92336032)

[Figure 15 : Company Details Page 13](#_Toc92336033)

[Figure 16 : Create Company Form 14](#_Toc92336034)

[Figure 17 :Update Company Form 14](#_Toc92336035)

[Figure 18 : User Manage Page 15](#_Toc92336036)

[Figure 19 : Projects Manage Page 15](#_Toc92336037)

[Figure 20 : Manage Tickets Page 15](#_Toc92336038)

# Design Changes and Reasons

The coursework specification had to design and implement a project management system. The implementation for the project management system which is the coursework two which will be discussed in this document. The first section of this document will discuss the design changes and the reasons for these changes. The designs for this were completed during the coursework one and have been implemented according to the designs. But for some reasons during the implementation the design has been changed due to some reasons.

Graphical user interface, application

Description automatically generatedThe software requirements that had been identified has been implemented other than for some few requirements. Mainly the view of the features of the system had been created other than the report generations and prediction. The reason for that was not able to connect the tables with each other and was not success in creating the assigning parts in the system. The complexity of the coding played a huge down fall during the implementation. Even the CRUD operations took a huge amount of time to implement.

**Figure 1: Navigation Bar**

The login interface that was design was also not implemented incorrectly. The system has the blazors built in login component for the admin to login. In creating the login was unsuccessful due to not been able to connect the database tables. The user table was initial supposed to be the table that should have been used in the login face but since the system was using the built in login and not using a custom login form made the login for other users not been able to login to the system.

Background pattern

Description automatically generatedThe project assigning to company and user and ticket assigning to the user was designed only on the use cases but was not added to the wireframes. The assigning was created using a drop down where the user can select the assignee and assign the project to a company, user and ticket to user. Apart from the assigning same as that the project assigning for the ticket, selecting the project to which the ticket belongs to and assign user to company for set the company of which the user is working for. Those changes were new and added just as the three main assigning.

**Figure 2 : Project Assign fields**

Some of the classes that was identified in the CRC table were not implemented but created in a single class to have maintainability of the system and the project. Such as the CRUD operation classes were created inside one class. Although the report generation and the prediction class and the model views have not been implemented.

When it came to the user interface, the wireframes that was originally designed had to be changed since the designs were not very user friendly and to be modified in order to get the user friendliness and also to make the readability even so. Also, the uses of confirmation and alert boxes to alert the user of the changes made and to ask confirmation of deleting a record form the system.

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated**Figure 3:Create form**

**Figure 4 : Alert Box**

A screenshot of a computer

Description automatically generated

**Figure 5 : Confirmation Box**

A final design change that was created was the implementing the edit functionality for the manage company section. During the design stage the update company was not included in the use case diagram since it was not that important. But during the implementation it was identified as it should exist since some details of the company might change during the time of creation like the address or the phone number. So, the update company feature was implemented so during a situation like that the system data could be changed.

**Graphical user interface, text, application

Description automatically generated**

**Figure 6 : Company Update**

# 2. Evaluation: Implementation Choices

In the evaluation section the implementation would be discussed as what technics that has been used to keep the readability and the maintainability of the code and the UI along with the design patterns that was used to implement and how it was helpful.

## 2.1 Maintainability and Readability

In order to keep a good maintainability and readability of the code, comments have been used through out the project and formatting of the code also gives the code the great readability and the file hierarchy of having the controllers the views and the models in different folders to make it easier for check files even after some time.

A screenshot of a computer

Description automatically generated

**Figure 7: Comments On the code**

Text

Description automatically generated

**Figure 8 : File Hierarchy**

### 2.2 Design Pattern

The MVC structure was the design pattern that was used to implement the solution. Creating the models that was identified along with their attributes and then creating the controllers with the crud operations and finally the view pages to view the output. Since this implementation is tested along the way Test Driven Development has also been used. The structure of the project is easy to access and edit since it follows the MVC structure.

*Text

Description automatically generatedA screenshot of a computer

Description automatically generated with medium confidence****Figure 9 : MVC Structure***

### 2.3 Testability

Before connecting to the database, the controllers and methods were tested out using Lists. Lists containing the data made it easy to use and high performance since it takes time to do operations with database. That helped to ease the productivity and prevent the bugs and have a clean solution before creating database and connecting to it.

A screenshot of a computer

Description automatically generated with medium confidence

**Figure 10 : Array used in the system**

Also console messages to identify the operations that are performed are executing correctly and to debug any issues that would occur when going through the system. Also, the use of alert boxes were help in displaying the success messages as a sign of completing the task that was given

Text

Description automatically generated

**Figure 11 : Console messages and alert messages**

### 2.4 Security

For security purpose the system cannot be used without getting logged in only the login link would be displayed in the home since if the features would be available freely anyone would be able to perfume any CRUD operation in the system and it would be a disaster.

Text

Description automatically generated

**Figure 12 : Authorization to access the navigation features**

# 3. User Interfaces

Graphical user interface, text, application

Description automatically generated

**Figure 13 : Home Page**

A screenshot of a computer

Description automatically generated

**Figure 14 : Manage Company Page**

**Figure 15 : Company Details PageGraphical user interface, text

Description automatically generated**

Graphical user interface, application

Description automatically generated

**Graphical user interface

Description automatically generatedFigure 16 : Create Company Form**

**Figure 17 :Update Company Form**

A screenshot of a computer

Description automatically generated

**Figure 18 : User Manage Page**

A screenshot of a computer

Description automatically generated

**Figure 19 : Projects Manage Page**

A screenshot of a computer

Description automatically generated

**Figure 20 : Manage Tickets Page**