[Project Overview 1](#_Toc367165166)

[Requirements Gathering 2](#_Toc367165167)

[Requirements 2](#_Toc367165168)

[Non-Functional Requirements 2](#_Toc367165169)

[Functional Features 2](#_Toc367165170)

[USE Case Diagram 2](#_Toc367165171)

[CRUD Analysis 2](#_Toc367165172)

[Web Page Prototype 2](#_Toc367165173)

[Site Map Diagram 3](#_Toc367165174)

[ERD or UML (Database) Class Diagram 3](#_Toc367165175)

[Presentation October 15, 1:00PM 3](#_Toc367165176)

|  |  |
| --- | --- |
|  |  |
| Due Date: | October 16, 12PM |
| Percentage of module mark: | 40% |
| Submission Details: | Submit one zipped folder named YourApplicationName. In the folder include these two items.   1. Your HTML web page prototypes in a zip file that is ready-to-run once it is extracted. 2. One word document that combines all screenshots of your case diagram, UML class (database) diagram, site map, and written documentation requested. Be sure to include the names of the people in your group at the top of this document.   Marks will be deducted for additional documents submitted. |
| Late Penalty: | 20% deducted each day this assignment is late. |

## Project Overview

With a group of three or four persons, your job is to design a system that will do something to earn money. As examples, you might design a system to offer a service such as event ticket sales, hotel bookings, or something else. Your design does not need to go to production but make the effort to keep it feasible and realistic.

## Requirements

### Non-Functional Requirements

5 **marks**

Identify the non-functional requirements of your application that are needed to make it successful. Most of the time, this section may only include platform needs and colour / logo requirements. For larger systems and organizations, this section might include business policies, system workload restrictions, and security restrictions.

### Functional Features

**10 marks**

On one page describe the key items that make your application fully functional. Use one full page to show enough detail about the necessary features but do not exceed this limit. For your application to be successful, list the functional features under essential, important, and extra categories.

## USE Case Diagram

Provide a use case diagram to show the system goals, the sub-systems, and actors involved for functional features only. The screenshot of this diagram must fit on one page and it must be readable.

**15 marks**

## CRUD Analysis

In tabular form, summarize the different use cases that lead to entities being created, read, updated, or deleted. Indicate which entities are affected for each case. **10 marks**

## Web Page Prototype

Design the web pages in static HTML. (Use an editor – don’t hand code your HTML in this phase.) The hyperlinks are the only controls that need to be functional so users can move easily from one page to the next. There should be no live data and no form of session management. You are encouraged to create your web pages using a web page editor with drag and drop capability.

Do not waste time making graphics, fancy CSS, or javascript in this phase. Stay focused on page layout for link navigation, control alignment and position, and static realistic looking data content. (Do not use ‘lorem ipsum’ or data that has no meaning such as ‘asdf’ for the address. Do not create overly silly data such as user accounts for ‘Barrack Obama’ or ‘Mickey Mouse’ – this does not look professional.) HTML quality, colors, images, and font style are unimportant. Controls do not have to work but project stakeholders should be able to navigate through the page links to understand the site flow.

Create different versions of the site prototype for each type of user to show what each user type sees while navigating through the options that are available to them.

**25 marks**

## Site Map Diagram

Summarize the page flow for the web page prototypes in one image that fits on one page. Keep it readable, organized, and professional. The diagram must indicate how pages flow from one to the other.

**5 marks**

## ERD or UML (Database) Class Diagram

Develop a relationally sound ERD or UML database diagram to model your tables, columns, and their associations with proper multiplicity or cardinality. Do not show aggregation, composition, direction, or inheritance – remember that data is flat. Show data types in your diagram that are suitable in size and format for each column. Use a uniform naming standard. Arrange your tables so they all fit on one page and so the diagram is readable on one page once the diagram is complete. Take one or more screenshot of this diagram and include it in the one word document that you submit. However you present it, make it look professional. Do not submit your visio or Visual Paradigm file.

**20 marks**

## Presentation October 16, 1:00PM

Details about the presentation grading scheme will be issued one week prior to the presentation. Your presentation should highlight your design with a focus on the web user interface prototype and any other models that you feel can help. Your team can plan to be responsible for presenting your design in a 15 minute to 20 minute demonstration during the afternoon of October 25.

**10 marks**

**Deductions**

Marks will be deducted for the following:

* Inconsistent documentation. For example, web page mock ups do not match the pages shown in the site map. Or, the data shown in the UML class diagram is not the same as the data in the online forms or reports.
* Hard-to-read documentation.
* Un-cropped and improperly sized images which are hard to read.
* Unprofessional appearance of documentation.