CHAMPLAIN COLLEGE LENNOXVILLE

FINAL PROJECT

FROM

*Munir Khaliqyar and Andrew Sherrer*

420-510-LE MOBILE APPLICATIONS

COMPUTER SCIENCE TECHNOLOGY

PRESENTED TO CAROLINE FORTIER

December 2023

Contents

[TIMELINE 3](#_Toc150851086)

[November 14th 3](#_Toc150851087)

[Project Scope Presentation 3](#_Toc150851088)

[November 14th – November 24th 3](#_Toc150851089)

[Project Design (5%) 3](#_Toc150851090)

[November 24th – December 8th 3](#_Toc150851091)

[Project Development (20%) 3](#_Toc150851092)

[December 8th – December 11th 4](#_Toc150851093)

[Project Presentation (5%) 4](#_Toc150851094)

[REQUIREMENTS 5](#_Toc150851095)

[User Stories (example) 5](#_Toc150851096)

[Project Design 6](#_Toc150851097)

[Activity Diagram (Process flow using BPMN) 6](#_Toc150851098)

[Mockups 7](#_Toc150851099)

[Database design 8](#_Toc150851100)

[User Acceptance Tests 9](#_Toc150851101)

# TIMELINE

## November 14th

### Project Scope Presentation

1. Requirements are defined in the Requirements section below.

## November 14th – November 24th

### Project Design (5%)

1. Alone or in teams of 2, provide the User Stories that will define the requirements of your app, and design the following elements in this document
   1. User Stories
   2. Activity Diagram (Process flow using BPMN)
   3. Mock-ups
   4. Database design
   5. List your user acceptance tests that will be used during your presentation
2. The sooner you send your Project Design in, the sooner it can be adjusted and be approved

## November 24th – December 8th

### Project Development (20%)

1. Alone, or in teams of 2, use the source code management tool of your choice
2. Breakdown your user stories into manageable tasks
3. Integrate yours and your teammates code often.

## December 8th – December 11th

### Project Presentation (5%)

1. On Teams or in person, teams will go through the UAT and Unit tests

# REQUIREMENTS

1. Use SQLite and the Room Library to store and manage your data (minimum : containing list of items, their description and their price)
2. Use the architecture best practices seen in class (UI Layer vs Data Layer)
3. Use fragments and the navigation graph to navigate between fragments, or composables
4. Navigate between at least 3 screens
5. There must be a purchase and a calculation of purchased total
6. Have a top bar with a menu
7. Have a Floating Action button to send an email
8. Use the Calendarview, or other datepicker to input your dates
9. Use Mapview and reference a URL inside your app
10. Create 3 unit tests for your application

# Project Design

## User Stories

We develop a mobile application for a client owning a **Bed and Breakfast**. Here are her requirements:

1. As a client, I want to be able to see information on the Bed and Breakfast (such as an information page in a menu)
2. As a user, I want to be able to search available booking dates so that I can continue to a reservation.
   1. Acceptance criteria
      1. Given I want to book a room, when I generate a search by entering the date in the search field, then I see a list of available rooms.
3. As a client, I want to choose the type of room I want to reserve.
4. As a client, I want to see the cost of each available room.
5. As a client, I want to pay for my room on reservation at the Bed and Breakfast
6. As a client, I want to be able to cancel my reservation without fee.
7. As a client, I want an email confirmation of my booking or cancellation.
8. As a client, I want to be able to find the location of the Bed and Breakfast easily.

## Activity Diagram (Process flow using BPMN)

(Double click to modify in Visio. You can also use any other tool you are more comfortable with)



## Mockups

(Double click to modify in Visio. You can also use any other tool you are more comfortable with)

For example:

 

Note: Don’t forget to indicate the navigation between screens

## Database design

(double click to modify in Visio. You can also use any other tool you are more comfortable with)

For example:



## User Acceptance Tests

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| User Story | Acceptance criteria | ID | Test Summary | Manual/  Auto | Status | QA | UAT |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |