Team Contract

SYST 17796 TEAM PROJECT

Team Name: Team Starter Immanuel

Please negotiate, sign, scan and include as the first page in your Deliverable 1.

Please note that if cheating is discovered in a group assignment each member will be charged with a cheating offense regardless of their involvement in the offense. Each member will receive the appropriate sanction based on their individual academic integrity history.

Please ensure that you understand the importance of academic honesty. Each member of the group is responsible to ensure the academic integrity of all of the submitted work, not just their own part. Placing your name on a submission indicates that you take responsibility for its content.

For further information, read Academic Integrity Policy here :
https://caps.sheridancollege.ca/student-guide/academic-policies-and-procedures.aspx

Team Member Names (Please Print)	Signatures	Student ID
Project Leader: Christian Malan		991720322

By signing this contract, we acknowledge having read the Sheridan Academic Integrity Policy

Responsibilities of the Project Leader include:

- Assigning tasks to other team members, including self, in a fair and equitable manner.
- Ensuring work is completed with accuracy, completeness and timeliness.
- Planning for task completion to ensure timelines are met.
- Notifying the professor of any issues in a timely manner so that corrective measures can be taken.
- Any other duties as deemed necessary for project completion.

What we will do if . . .

Scenario	**Accepted initials**	**We agree to do the following (Put an X corresponding to your choice in each box)**
Team member does not regularly attend team meetings and/or does not respond to communications in a timely manner.	CM	Project leader emails the student citing the concerns and cc's the professor so they are aware of the situation at the very onset _X_ (**Mandatory**). a) __ In addition to above, the leader/team will (add your own content here):
Team member does not deliver component on time due to severe illness or extreme personal problem.	CM	a) Team absorbs workload temporarily __ b) Team seeks advice from professor _X_ c) Team shifts target date if possible _X_ d) __ Other (specify):

Scenario	Accepted initials	We agree to do the following (Put an X corresponding to your choice in each box)
Team member has difficulty delivering component on time due to lack of understanding or ability.	CM	a) Team reassigns component __ b) Team helps member __ c) Team member must ask professor for help X d) __ Other (specify):
Team member does not deliver component on time due to lack of effort.	CM	a) Team absorbs workload __ b) Team member(s) ask professor to request a Participation Form from all team members. This *may* result in individualized grades being awarded for a deliverable __ c) Both a. and b. above __ d) ___ Other (specify):
Team cannot achieve consensus leaving one or more member(s) feeling that their voice(s) is/are not being heard in a decision which affects everyone.	CM	a) Team agrees to abide by majority vote __ b) Team seeks advice from the professor __ c) __ Other (specify):

Scenario	Accepted initials	We agree to do the following (Put an X corresponding to your choice in each box)
Team members do not share expectations for the quality of work on a particular deliverable.	CM	a) Team members will draw on each other's strengths to help bring the quality of the deliverable to a minimal acceptable level __ b) Team votes on each submission's quality __ c) Team member(s) ask professor to request a Participation Form from all team members, which may result in individualized grades being awarded for a deliverable __ d) __ Other (specify):
Team member behaves in an unprofessional manner, e.g. being rude, uncooperative and/or making one or more member(s) feel uncomfortable.	CM	a) Team agrees to avoid use of all vocabulary inappropriate to a business/college setting __ b) Team attempts to resolve the issue by airing the problem at a team meeting __ c) Team requests a meeting with the professor to discuss further __ d) __ Other (specify):
There is a dominant team member who insists on making all decisions on the team's behalf leaving some team members feeling like subordinates rather than equal members	CM	a) Team will actively solicit consensus on all decisions which affect project direction by asking for each member's decision and vote __ b) Team will express subordination feelings and attempt to resolve issue __ c) Team seeks advice from the professor __ d) __ Other (specify):

Scenario	Accepted initials	We agree to do the following (Put an X corresponding to your choice in each box)
Team has a member who refuses to participate in decision making but complains to others that s/he wasn't consulted	CM	a) Team forces decision sharing by routinely voting on all issues __ b) Team routinely checks with each other about perceived roles __ c) Team discusses the matter at team meeting __
Any other items/issues – use as many rows as needed		

SYST 17796 Deliverable 1 - **Design Document**

OVERVIEW

1. Project Background and Description

The trend in finding home and property is always moving and constant. It is crucial for both businesses owners and customers to decide when it comes to looking for the "right property". The system of the project will provide a solution by providing a computer-based system platform for the owner to showcase the properties while giving the customer an access to the listing with filters for their preferences.

The goal of this project is to develop a computer-based system and create a database for commercial use in the real estate business in accordance to the project deliverables on the Fundamentals of Software Design and Development in the duration of the course. The system will give the business owners an access to manage property listings in the interface. Owners can add and remove property in their listing or modify the information of the properties. On the customer side, users will be guided through questions to help them find their ideal home, or they can browse all available property listings. The system will also allow the customers to search for their ideal homes.

The project will have a starting code that relates to the database project used in the textbook of the course. The code for the system will use the Java language and apply the principles of object-oriented programming. The system will have an inventory pattern used in the real-world setting. Each property will be represented as an object in the class called "Property". The class called "Inventory" will manage the listed properties and store them in a list. The "Inventory" class will also provide methods to handle property searches. The system will aim to follow the MVC (Model-View-Controller) pattern, where the Model manages the data and algorithms in the database, the View is the interface where customers interact with the system, and the Controller processes customer preferences, updates the Model, and returns the results to the View for display.

2. Project Scope

Member: Christian Malan

- Lead Developer and Backend Developer
  - Create a flow of program that meets the project goals
  - Create the algorithm that will follow the program flow.
  - Create methods that will manage the property listing such as adding, revising and removing properties in the list.

        - Write the logic of the code that will be used in the system to validate the inputted data.

- Database Manager
    - Create and design the schema of the database.
    - Create a method that will be used to manage the property on the list.

- Frontend Developer
    - Create and develop code that will be used in the user interface
    - Create a method that will filter the property listing based on the answer of the customer in their preferences.

- QA Tester
    - Test the system functionality and make sure it aligns with the project goals.
    - Ensure that the system can authenticate if the user is the owner or the customer
    - Ensure that the system is responsive
    - Ensure that the system is user-friendly
    - Provide feedback every testing

The technical scope of the project was divided into two parts for the interface. The system will give the business owner an access to add, edit or remove the properties from the list. The system will also allow the customers to search for properties based on criteria like location, house type, number of bedrooms and other more helpful filters. The system will prompt questions to narrow down the ideal property of the customer. The project will be complete when the system allows owners to manage property listings and allows customers to search using criteria then displaying the list of their preferences as the output.

3. High-Level Requirements

The new system must include the following:

- Ability for the user to sing-in
- Ability for the owner to log-in with proper credentials
- Ability for the owner to add a new listing of property
- Ability for the owner to revise the information of the property on the listing
- Ability for the owner to remove the property from the listing
- Ability for the customer to search for properties based on multiple criteria that will be prompt to them.
- Ability for the customer to view all the information of the properties in the listing.

4. Implementation Plan

Git Repository URL: Github Link (https://github.com/chris-malan/SYST17796project.git)

The Git repository will be used to manage and track the progress of the project. Since the development team is composed of one member, the expected use of Git is to check code progress once a week during the laboratory time of the class on Tuesday or whenever there is an important update to the code at any given time. The repository will be organized into the following directories to ensure that the project is well-structured.

The repository that contains the project and will be push to the Git remote repository are src, uml and docs. Directory called src, which contains all the source code files related to the project. Directory called uml, which contains diagrams that visualize the structure of the system. Lastly, directory called docs, which contains all text files, documents, and other project requirements.

In this project, the developer will the follow coding standards that was commonly used in creating a program. These standards are using camelCase for naming conventions, adding comments to ensure maintainability and easy understanding of the code, and ensuring modularity by making sure that methods and classes are short and handle only one task at a time. The project will use the IDE used in the class which is the NetBeans IDE and use Visual paradigm for UML Diagram.

5. Design Considerations

The code use in the project system relates to the code use in the textbook use in the class and uses objected-oriented principles. In the textbook entitled "Head First Object-Oriented Analysis and Design", it was stated that one of the basic principles of object-oriented programming is the encapsulation, and it helps the flow of program because it makes the variables in the class private and make the system to be delegated into parts using logics (Bradley et al, 2006, Chapter 1).

For example, the starting code in this project that will be for the real-estate business, uses the principles of encapsulation. In the Property class, the developer ensures that each field for variable such as price, location, and numberOfBedrooms have a private field, this means that the variables cannot be directly accessed or modified by any person or code outside its own class.

Another example on our project codes is the use of getter and setter methods. These methods are used in all our variables. The setter methods in the Property class ensure that any input will be validated to avoid error on the system.

The second principle discussed in the textbook is delegation. According to the textbook, delegation is when an object delegates or give the specific task to other objects instead of performing it itself. This improves code reusability and functionality, while also avoiding repetitive code (Bradley et al., 2006, Chapter 1).

For example, the starting code used in our project uses the principles of delegation. The program has two classes. The Property class and the Inventory class. The Inventory class delegates the task of storing property details to the Property class, while it only maintains a list of Property objects.

Another example on our code is in the Inventory class. The methods in this class are responsible for filtering properties when a customer searches for a listing delegates this task to another method or class, such as a SearchProperty class.

The third principle of object-oriented programming that will be applied in this project is the flexibility and maintainability principle. In the textbook, it is stated that flexible program is codes that is easy to modify, maintain, and reuse (Bradley et al., 2006, Chapter 1). This means that flexible code may be given to other programmer and can be easily understood, which means that the code can be also used in other projects.

For example, the starting code in this project also upholds the principles of flexibility and maintainability. The starter code is modular, with class like Property class and Inventory class doing their own specific task, this allows the design of program to add new features without affecting the other part of the system.

Another example is how the methods in the Inventory class, such as addProperty method and searchProperties method are designed to handles one specific task like how the classes handle their own specific task. The build on how the class and methods are structured makes our system easier to maintain. Changes done on the code or error output that will be thrown will be localized. This means this changes and error will not disrupt other part of the code and can rectified easily.

These object-oriented programming principles are fundamental to the design and structure of the program in this project. By applying encapsulation, delegation, and flexibility, the project can create a more efficient and maintainable system.

REFERENCES

McLaughlin, B., Pollice, G., & West, D. (2006). *Head First Object-Oriented Analysis and Design.* O'Reilly Media, Inc. https://learning.oreilly.com/library/view/head-first-object-oriented/0596008678/?ar