**Sell-Ease**

|  |
| --- |
|  |

**By:**

**Muzamil Khan– Team Member 1**

**27658**

**M. Bilal – Team Member 2**

**27950**

**Suffian – Team Member 3**

**27966**

**Supervised by:**

**Prof. Dr Muhammad Mansoor Alam**

**Faculty of Computing**

**Riphah International University, Islamabad**

**Spring/Fall 2024**

**A Dissertation Submitted To**

**Faculty of Computing,**

**Riphah International University, Islamabad**

**As a Partial Fulfillment of the Requirement for the Award of the Degree of**

**Bachelors of Science in Computer Science**

**Faculty of Computing**

**Riphah International University, Islamabad**

Date: [date of final presentation]

**Final Approval**

This is to certify that we have read the report submitted by **Muzamil Khan, M.Bilal , Suffian *(CMS #27658,27950,27966)***, for the partial fulfillment of the requirements for the degree of the Bachelors of Science in Computer Science (BSCS). It is our judgment that this report is of sufficient standard to warrant its acceptance by Riphah International University, Islamabad for the degree of Bachelors of Science in Computer Science (BSCS).

**Committee:**

|  |  |
| --- | --- |
| **1** | Prof Dr. Muhammad Mansoor Alam  (Supervisor) |
|  |  |
| **2** | Dr. Musharraf Ahmed  (Head of Department/chairman) |

**Declaration**

We hereby declare that this document “**Sell-Ease**” neither as a whole nor as a part has been copied out from any source. It is further declared that we have done this project with the accompanied report entirely on the basis of our personal efforts, under the proficient guidance of our teachers, especially our supervisor **Prof. Dr Muhammad Mansoor Alam**. If any part of the system is proved to be copied out from any source or found to be reproduction of any project from anywhere else, we shall stand by the consequences.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Muzamil Khan**

**27658**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**M. Bilal**

**27950**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Suffian**

**27966**

**Dedication**

Our project is dedicated to our parents, seniors, friends, and our supervisor "Prof Dr. Mansoor Alam" who has been our continual source of inspiration and whose support has helped this project succeed. This project would not have been possible without their trust and support.

**Acknowledgement**

First of all we are obliged to Allah Almighty the Merciful, the Beneficent and the source of all Knowledge, for granting us the courage and knowledge to complete this Project.

We owe a heartfelt thank you to our project supervisor, Dr. Mansoor Alam. His guidance has been a beacon of light throughout our project journey.His patience and knowledge were key in overcoming the challenges we faced. We are truly thankful for his dedication and the time he invested in us.

We also extend our deepest gratitude to our parents and family. Their unwavering belief in us and the values of hard work and integrity they have nurtured within us have been our guiding stars. It is with their blessings and constant encouragement that we have been able to achieve this milestone.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Muzamil Khan**

**27658**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**M. Bilal**

**27950**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Suffian**

**27966**

Table of Contents

[Table of Contents i](#_Toc113957278)

[List of Tables iii](#_Toc113957279)

[List of Figures iv](#_Toc113957280)

[Abstract 1](#_Toc113957281)

[Chapter 1: Introduction 2](#_Toc113957282)

[1.1 Goals and Objectives 2](#_Toc113957283)

[1.2 Scope of the Project 2](#_Toc113957284)

[Chapter 2: Literature Review 3](#_Toc113957285)

[2.1 Introduction 3](#_Toc113957286)

[2.2 Background and Problem Elaboration 3](#_Toc113957287)

[2.3 Detailed Literature Review 3](#_Toc113957288)

[2.3.1 Definitions 3](#_Toc113957289)

[2.3.2 Related Research Work 1 3](#_Toc113957290)

[2.3.3 Related Research Work 2 3](#_Toc113957291)

[2.4 Literature Review Summary Table 3](#_Toc113957292)

[2.5 Research Gap 3](#_Toc113957293)

[2.6 Problem Statement 3](#_Toc113957294)

[Chapter 3: Requirements and Design 4](#_Toc113957295)

[3.1 Requirements 4](#_Toc113957296)

[3.1.1 Functional Requirements 4](#_Toc113957297)

[3.1.2 Non-Functional Requirements 4](#_Toc113957298)

[3.1.3 Hardware and Software Requirements 4](#_Toc113957299)

[3.2 Proposed Methodology 4](#_Toc113957300)

[3.3 System Architecture 4](#_Toc113957301)

[3.4 Use Cases 4](#_Toc113957302)

[3.4.1 Sample Use Case Name Here 4](#_Toc113957303)

[3.5 Database Design *(Optional)* 6](#_Toc113957304)

[3.6 Class diagram *(Optional)* 6](#_Toc113957305)

[3.7 Sequence diagram *(Optional)* 6](#_Toc113957306)

[3.8 Any Other Artifact… 6](#_Toc113957307)

[3.9 Graphical User Interfaces(GUI) (*Optional)* 6](#_Toc113957308)

[Chapter 4: Implementation and Test Cases 7](#_Toc113957309)

[4.1 Implementation 7](#_Toc113957310)

[4.1.1 Implementation of First Component/Algorithm 7](#_Toc113957311)

[4.2 Test case Design and description 7](#_Toc113957312)

[4.2.1 Sample Test case No.1 7](#_Toc113957313)

[4.2.2 Sample Test case No.2 7](#_Toc113957314)

[4.3 Test Metrics 8](#_Toc113957315)

[4.3.1 Sample Test case Matric.No.1 8](#_Toc113957316)

[4.3.2 Sample Test case Metric.No.2 8](#_Toc113957317)

[4.3.3 Sample Test case Metric.No.3 8](#_Toc113957318)

[Chapter 5: Experimental Results and Analysis 9](#_Toc113957319)

[Chapter 6: Conclusion and Future Directions 10](#_Toc113957320)

[References 11](#_Toc113957321)

[Appendix 12](#_Toc113957322)

[Appendix A: Guidelines 12](#_Toc113957323)

[Appendix B: Heading of Sample Appendix B 12](#_Toc113957324)

[Formatting Guidelines 13](#_Toc113957325)

[Chapter 1: Heading 1 13](#_Toc113957326)

[1.1 Heading 2 13](#_Toc113957327)

[1.1.1 Heading 3 13](#_Toc113957328)

[Tables and Figures 14](#_Toc113957329)

[Equations 15](#_Toc113957330)

[Header/Footer 15](#_Toc113957331)

[Other Formatting Guidelines 15](#_Toc113957332)

[References 15](#_Toc113957333)

List of Tables

[Table 1: This is Sample table caption 1](#_Toc55559466)

[Table 2: This is Sample table caption 1](#_Toc55559467)

List of Figures

Figure 1: List of Styles 1

Figure 2: IEEE Reference style 1

# Abstract

Sell-Ease aims to create an automated system for predicting property prices based on features such as location, furnishing status, number of bedrooms, and total rooms. The purpose of this project is to eliminate the reliance on subjective valuations by owners and real estate agents, which can lead to inconsistencies. We gathered an extensive dataset of properties and conducted thorough preprocessing, including cleaning the data, removing outliers, and focusing exclusively on buy/sell transactions. Various predictive models, including linear regression and gradient boosting, were trained and tested to identify the most effective method for accurate price prediction. Our findings indicate that these models can reliably estimate property values based on the specified features. The next steps involve integrating the best-performing model into a user-friendly web application, allowing users to input property details and receive instant valuations. This system promises to provide a consistent and objective tool for property valuation, empowering owners with accurate market insights. Sell-Ease aims to revolutionize property pricing by offering a reliable, data-driven solution that reduces dependency on subjective assessments and enhances decision-making in the real estate market.

# Introduction

Sell-Ease is a web-based platform designed to provide property owners with an automated, data-driven tool for predicting property prices based on various features such as location, furnishing status, number of bedrooms, and total rooms. Currently, property valuation in the real estate market is conducted manually by owners and real estate agents, relying heavily on subjective judgment and experience. This manual approach often leads to inconsistencies and inaccuracies, making it challenging for property owners to obtain reliable estimates of their property values.

The main problem addressed by Sell-Ease is the lack of an automated system for property valuation. Traditional methods are not only time-consuming but also prone to biases and errors, resulting in a lack of confidence among property owners regarding the accuracy of their property valuations. By developing a predictive model using machine learning techniques, Sell-Ease aims to provide a consistent and objective valuation method, reducing dependency on subjective assessments.

Sell-Ease also enhances the user experience by offering a transparent, secure, and efficient platform for property owners to obtain accurate market insights. This innovation in property valuation not only empowers owners with precise information but also promotes informed decision-making in the real estate market. The ultimate goal is to revolutionize property pricing, fostering greater transparency and trust in the market.

## Goals and Objectives

The primary goal of Sell-Ease is to develop a web-based platform that accurately predicts property prices based on detailed property features, ensuring ease of use and reliability for property owners.

**Goals:**

* To provide an automated system for property valuation that eliminates the need for subjective assessments.
* To leverage machine learning techniques to enhance the accuracy and reliability of property price predictions.
* To develop a user-friendly web application where property owners can easily input property details and obtain instant valuations.

**Objectives:**

* To gather and preprocess a comprehensive dataset of property features and prices, ensuring high-quality data for model training.
* To train and test various predictive models, including linear regression and gradient boosting, to identify the most effective approach for property valuation.
* To address the limitations of current property valuation methods by providing an objective, data-driven tool that improves accuracy and reduces biases.
* To integrate the predictive model into a web application, offering a seamless user experience for property owners seeking accurate valuations.
* To enhance the transparency and efficiency of property valuations, thereby fostering greater trust and confidence in the real estate market.

## Scope of the Project

* **Property Valuation Platform:** The website will offer a comprehensive platform where property owners can easily input various property features such as location, furnishing status, number of bedrooms, and total rooms to obtain accurate price predictions.
* **Data Handling and Processing:** The system will gather and preprocess extensive property datasets, including data cleaning, outlier removal, and feature selection, to ensure high-quality input for model training and predictions.
* **Machine Learning Models:** Sell-Ease will utilize advanced machine learning models to predict property prices. The models will be trained and tested to achieve high accuracy and reliability in price estimations.
* **User Interaction:** The platform will provide a seamless and user-friendly interface for property owners to interact with the system. Users can enter property details and receive instant valuation results, enhancing their understanding of property market values.
* **Increased Interaction:** By offering accurate and immediate property valuations, the platform will facilitate better interaction between property owners and the real estate market, helping users make informed decisions based on reliable data.
* **Accessibility and Usability:** The platform will be designed to be accessible and user-friendly, catering to users with varying levels of technical expertise. The simplified interface will ensure that both novice and experienced users can navigate and utilize the system effectively.

# Literature Review

# Introduction

Sell-Ease is a web-based platform designed to provide property owners with an automated, data-driven tool for predicting property prices based on various features such as location, furnishing status, number of bedrooms, and total rooms. This platform addresses the significant limitations of current property valuation methods, which are often manual and subjective, relying heavily on the judgment of owners and real estate agents. These traditional methods can lead to inconsistent and inaccurate valuations, making it challenging for property owners to obtain reliable estimates of their property values.

## Background and Problem Elaboration

The real estate industry has been slow to adopt technological advancements, particularly in the realm of property valuation. Despite the availability of extensive data on property features and market trends, manual valuation methods remain prevalent. These methods are not only time-consuming but also prone to biases and errors, leading to a lack of confidence in the accuracy of property valuations.

Machine learning offers a promising solution to these challenges by enabling the development of predictive models that can analyze large datasets and identify patterns that are not immediately apparent to human evaluators. By leveraging machine learning techniques, it is possible to create models that provide accurate and objective property price predictions based on various features. This approach has the potential to revolutionize property valuation, making it more consistent, reliable, and efficient.

Existing property valuation systems, particularly in markets like Pakistan, often lack the necessary mechanisms to provide accurate and unbiased valuations. These systems also fail to address common issues such as the impact of location, furnishing status, and other critical features on property prices. Sell-Ease aims to fill this gap by developing a comprehensive, feature-based property valuation system that leverages advanced machine learning models to deliver accurate and reliable price predictions.

## Detailed Literature Review

### Definitions

**Property Valuation:** Property valuation is the process of determining the economic value of a real estate asset. This involves analyzing various features of the property, such as location, size, condition, and market trends, to estimate its fair market value.

**Machine Learning (ML):** Machine learning is a subfield of artificial intelligence that focuses on developing algorithms and statistical models that enable computers to perform tasks without explicit instructions. These systems learn from data, identify patterns, and make decisions with minimal human intervention. In the context of property valuation, ML algorithms can be used to predict property prices based on historical data and property features.

**Supervised Learning:** Supervised learning is a type of machine learning where the model is trained on a labeled dataset. This means that each training example is paired with an output label. The goal is to learn a mapping from inputs to outputs, allowing the model to make accurate predictions on new, unseen data. Techniques like linear regression and gradient boosting used in Sell-Ease fall under supervised learning.

**Linear Regression:** Linear regression is a statistical method that models the relationship between a dependent variable and one or more independent variables. It is widely used for predictive analysis and is one of the simplest and most interpretable machine learning models. In property valuation, linear regression can predict property prices based on various features.

**Gradient Boosting:** Gradient boosting is an ensemble machine learning technique that builds a series of weak learners, typically decision trees, in a sequential manner. Each new tree attempts to correct the errors made by the previous ones. This method is powerful for regression and classification tasks, making it suitable for predicting property prices with high accuracy.

**Data Preprocessing:** Data preprocessing involves transforming raw data into a clean and usable format. This step includes handling missing values, removing outliers, normalizing data, and encoding categorical variables. Proper preprocessing is crucial for improving the performance and accuracy of machine learning models.

**Feature Engineering:** Feature engineering is the process of creating new features from raw data to improve the performance of machine learning models. It involves selecting relevant features, transforming existing ones, and creating new features that better capture the underlying patterns in the data. For property valuation, features such as location, number of bedrooms, and property size are critical.

**Algorithm:** An algorithm is a well-defined set of instructions for solving a specific problem. In the context of Sell-Ease, algorithms refer to the computational procedures used to analyze property features and predict prices. This includes preprocessing steps, feature extraction methods, and the training and inference processes of the machine learning models.

**Data Augmentation:** Data augmentation is a technique used to increase the size and diversity of a dataset by applying various transformations to the existing data. This helps improve the robustness and generalization of machine learning models. Common data augmentation techniques include rotation, scaling, and adding noise. In property valuation, augmentation can be used to simulate different market conditions and property features.

**Model Training:** Model training is the process of feeding a machine learning algorithm with data to learn the patterns and relationships within the data. The training process involves optimizing the model's parameters to minimize prediction errors. In Sell-Ease, different models are trained on historical property data to predict future property prices.

**Model Evaluation:** Model evaluation is the process of assessing the performance of a machine learning model using various metrics such as accuracy, precision, recall, and mean squared error. This step is essential to ensure that the model generalizes well to new, unseen data. Sell-Ease uses evaluation metrics to determine the effectiveness of different models in predicting property prices.

**Pandas:** The name Pandas comes from a combination of the terms Panel Data and data analysis. In this research, it plays an important role in the following aspects:

1. Provide a simple and efficient DataFrame object with a default or custom labels.

2. Load the dataset through a CSV file and convert it into the processable objects.

3. Data normalization operations and missing value handling can be easily implemented.

4. Convenient to add, modify, or delete data columns in DataFrame.

5. Provides a variety of ways to work with datasets, such as building subsets, slicing, filtering, grouping, and reordering.

6. Integration with other libraries is implemented like Scipy, Scikit-learn, and Matplotlib.

**Numpy:** The name NumPy stands for Numerical Python, an array object library with several dimensions (also known as Nd arrays) and a selection of functions for handling these arrays. Users may operate mathematically and logically related operations on arrays using theNumPy library. In this study, NumPy uses two models to measure the mean root square errors.

**Scikit-learn:** Scikit-learn plays a key role in machine learning. It is a machine learning library in Python, based on NumPy, SciPy, Matplotlib, and other data science packages, covering almost all aspects of sample data, data preprocessing, model verification, feature selection, classification, regression, and so on in machine learning, and the functions are highly potent. In this project, it gives various assistance in the following ways. Firstly, LabelEncoder () is applied to train and encode the dataset. Secondly, train\_test\_split divides the dataset into the training set and the testing set in a certain percentage. Thirdly, use LinearRegression () and RandomForestRegressor () to set the models predicting house prices. Fourthly, calculate the mean absolute error and root mean square error by employing mean\_absolute\_error () and mean\_squared\_error () functions.

**Matplotlib:** Matplotlib is one of the most popular data visualization software packages in Python, supports cross-platform operation, is a commonly used 2D drawing library in Python, and provides some 3D drawing interfaces. Matplotlib is often used with NumPy and Pandas and is one of the indispensable and important tools in data analysis. In this research, matplotlib is used to draw the figures that show the comparison of actual prices and predicted prices.

### Related Research Work 1

This paper named “[**Prediction of US House Prices Based on Machine Learning**](https://www.semanticscholar.org/paper/Research-on-the-Prediction-of-US-House-Prices-Based-He/e8599af6eb800df4353596ffebf7e076df2aa479)” it tell us about the following things like linear regression and random forest on property data.  
With the increase in the standard, houses have become an indispensable part of people’s lives. However, purchasing houses needs to consider the prices. Therefore, this paper uses machine learning to predict US house prices. Based on the random forest and linear regression method used in this research. It is found that the use of the former has a good effect on predicting multivariate nonlinear relationships in house prices. After comparing the results of actual prices and predicting prices, residents can select different room structures by contrasting them with experimental types as the numerical results are basically accurate. This research provides theoretical value to the literature on housing price prediction and brings different enlightenment to policymakers, regulators, and investors. As a result, this research plays a key role in maintaining the order of the real estate market and helping people make good choices.  
According to Zillow Research [1], the national unemployment rate fell below the natural unemployment rate, just 4.3 percent in 2017. And the 30-year fixed-rate mortgage's typical interest rate (the most common loan product type for U.S. home buyers) is 4%. A very strong job market, coupled with historically high housing affordability, stimulates the demand for people to buy a house. For most people, there are two key factors to stimulate them to get the need to purchase houses. One is landing a steady job that makes people can afford the monthly mortgage payments on a home. Another is low mortgage interest rates also make benefit to relieve the pressure of buying a house by installments. There are many theoretical papers that have already conducted some research on the house price prediction. Christopher and Todd (2010) used two periods of economic prosperity to examine the relative role of economic fundamentals and market psychology in explaining house price dynamics in the United States [2]. Changchun and Hui (2018) compared the linear regression model and random forest model to predict house prices [3]. Quang et al. (2019) found that house prices strongly correlate to location, area, and population and used them in the extreme gradient boosting model, hybrid regression model, stacked generalization model, etc. to estimate the house price [4]. In addition, some research introduces some machine learning methods in different areas. Xiaolan et al (2022) used machine learning to determine the energetic material's qualities in advance [5]. Procedia Computer Science (2021) used machine learning to predict the trend of the stock market [6]. Susmita et al (2022) found that machine learning can better fit when designing the structures of rubble-mound breakwaters [7]. Thus, this research aims to adopt the linear regression model and random forest model to predict US house prices in size and layout. This study has played a great role in both theory and practice. It enriches the literature on machine learning prediction of housing prices and gets revelation to policymakers, regulators, and investors.

### Related Research Work 2

This paper “[**Optimization of linear regression in house price prediction**](https://ace.ewapublishing.org/article/966d9c1fc2c043b19d2cadaf5ac5f325)” also tell us the price prediction of houses using linear regression and other algorithms.

House price prediction plays a very important role in housing transactions. Linear regression based algorithms show good effects in predicting house prices. They have strong interpretability and fast operation speed. However, people ignore the estimation of deviations in linear regression (LR) algorithms. In this paper, k-nearest neighbor (KNN) algorithm is supposed to estimate deviations that are added to the result of linear regression to predict house prices accurately. Furthermore, deviation regression (DR) algorithm is supposed to make the prediction result more accurate. By utilizing Boston House Price data from Kaggle, extensive experiments are conducted and demonstrate the superior performance and compatibility of DR.

### Related Research Work 3

This paper is “[**Housing Price Prediction via Improved Machine Learning Techniques**](https://www.sciencedirect.com/science/article/pii/S1877050920316318?via%3Dihub)” it tells the following.  
House Price Index (HPI) is commonly used to estimate the changes in housing price. Since housing price is strongly correlated to other factors such as location, area, population, it requires other information apart from HPI to predict individual housing price. There has been a considerably large number of papers adopting traditional machine learning approaches to predict housing prices accurately, but they rarely concern about the performance of individual models and neglect the less popular yet complex models. As a result, to explore various impacts of features on prediction methods, this paper will apply both traditional and advanced machine learning approaches to investigate the difference among several advanced models. This paper will also comprehensively validate multiple techniques in model implementation on regression and provide an optimistic result for housing price prediction.

### Related Research Work 4

This paper “ [House Price Prediction Using Data Mining With Linear Regression And Neural Network Algorithms](https://www.semanticscholar.org/paper/House-Price-Prediction-Using-Data-Mining-with-and-Palupi/67fe71c05322ce8183b91e3dd920031293f7b18b) ” is about Housing price prediction as above papers but its also focuses on neural network algorithms.

The need for housing in big cities is very high because most offices and economic centers are in big cities. Limited land and high demand cause house prices to rise. Many developers build housing on the outskirts of big cities with access to trains and toll roads to make transportation easier. Property developers compete by providing the best prices, various choices of house specifications, ease of the mortgage process, and attractive promotions such as no down payment. A house is a long-term investment whose price increases yearly, so proper analysis is needed to buy a place to live in. Several factors influence the price of a house, including location, land area, building area, building type, and so on. This research aims to create a house price prediction model using the Linear Regression Algorithm and Neural Network so that the results can be useful for property agents in predicting house sales or from the buyer's side in predicting house prices. The results of this research use the Linear Regression Algorithm RMSE 0.775, while the Neural Network Algorithm uses RMSE 0.645. From this research, modeling using the Linear Regression Algorithm has better results. Still, the Linear Regression Algorithm and Neural Network Algorithm have RMSE results that are close to accurate and have small errors.

## Literature Review Summary Table

The columns in the table depend upon your problem and should be specific to your project.

Table 1: History of Computing Devices

The summary of various computing devices invented in the past from 1833-1901 is presented here.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. | Name, reference | Inventor | Year | Input | Output | Description |
| 1. | Analytical Engine, [1] | Charles Babbage | 1833 | Punch cards | Printer, curve plotter, bell | First general purpose computer that had an arithmetical logic unit and could compute using conditional branching and loops. Also incorporated integrated memory. |

## Research Gap

## Problem Statement

# Requirements and Design

Describe all modules of requirements and design in clear English text along with the necessary diagram and figures. Anyone reading your report should be able to reproduce your system/results after reading it.

**For each chapter provide a paragraph of introduction and in the end a paragraph of conclusions.** Make sure no heading/subheading is blank. Write text to introduce each section as well.

Introduce sub-heading as:

## Requirements

### Functional Requirements

### Non-Functional Requirements

### Hardware and Software Requirements

## Proposed Methodology

## System Architecture

## Use Cases

### Sample Use Case Name Here

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | | Sample Use Case Name Here | | |
| Actors | | Admin, Business Owner, Store Manager | | |
| Summary | | The user shall provide their email and password on the login form and after successful verification, redirect the user to the home page. | | |
| Pre-Conditions | | The user must be in the database records either added by any of the authorized users or added manually by a developer.  The user must not already be logged in. | | |
| Post-Conditions | | The user’s session is successfully established and shall be redirected to the home page. | | |
| Special Requirements | | None | | |
| Basic Flow | | | | |
| Actor Action | | | **System Response** | |
| 1 | The user opens the login page. | | 2 | The login page is displayed asking for email and password. |
| 3 | The user enters valid email and password. | | 4 | The system verifies the email and password, establishes a session for the user and redirects the user to the home page. |
| **Alternative Flow** | | | | |
| 3 | The user enters invalid email or password. | | 4-A | The system responds with an error message: *Incorrect email or password entered.* |

## Database Design *(Optional)*

## Class Diagram (*Optional)*

## Sequence diagram *(Optional)*

## Any Other Artifact…

## GUI Graphical User Interfaces (*Optional)*

This section should give the GUI dumps of each screen, with reference to the user. The navigation flow of each user is also required, and each GUI should mark the functionality/use case that it covers.

# Implementation and Test Cases

**For each chapter provide a paragraph of introduction and in the end a paragraph of conclusions.**

## Implementation

Whatever implementation that you have done so far, please elaborate here.

Give clear details of the algorithms that were implemented along with the platform and the APIs which were used. **For FYP-1, this chapter can be changed to description of prototype developed.**

### Implementation of First Component/Algorithm

Write implementation of first component of your system here.

## **Test case Design and description**

**This section will be added in FYP-II.** Summarize the common attributes of test cases. This may include input constraints that must be true for every input in the set of associated test cases, any shared environmental needs, any shared special procedural requirements, and any shared case dependencies. The following scheme is recommended for describing test cases in detail.

### Sample Test case No.1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<Software component Name>** | | | | | |
| **<Reference>** | | | | | |
| Test Case ID: | | *Reference Number* | Test Date: | | *Date* |
| Test case Version: | | *Version number* | Use Case Reference(s): | | *Relation to use cases* |
| Revision History: | | *Refer to previous test case identity (if any)* | | | |
| Objective | | *Need and scope of the testing* | | | |
| Product/Ver/Module: | | *Refer to overall system being built and the place of this test case in it.* | | | |
| Environment: | | *Necessary and desired properties of the test environment. (hardware/software)* | | | |
| Assumptions: | | *Assumptions that might affect the testing process.* | | | |
| Pre-Requisite: | | *Necessary condition that needs to be fulfilled prior to the test case.* | | | |
| Step No. | Execution description | | | Procedure result | |
|  | *Events being tested.* | | | *Mention software response.* | |
| Comments: | | | | | |
| *Passed* *Failed* *Not Executed* | | | | | |

### Sample Test case No.2

.

.

.

## Test Metrics

Summarize here the common ground of attributes of test case metrics.

### Sample Test case Matric.No.1

|  |  |
| --- | --- |
| Metric: | Purpose |
| Number of Test Cases: | Total number of test cases that you have developed for your system. |
| Number of Test Cases Passed: | The number of test cases that successfully passed |
| Number of Test Cases Failed: | The number of test cases that failed |
| Test Case Defect Density: | (No of test cases failed \* 100)  No of test cases executed |
| Test Case Effectiveness: | No of defects detected using test cases \*100  Total number of defects detected |
| Traceability Matrix: | Traceability is the ability to determine that each feature has a source in requirements and each requirement has a corresponding implemented feature. |

### Sample Test case Metric.No.2

### Sample Test case Metric.No.3

.

.

.

# Experimental Results and Analysis

**This chapter will be added in FYP-II.** Give proper analysis and discussion of experimental results (in plain English text) along with tables of results. **For each chapter provide a paragraph of introduction and in the end a paragraph of conclusions.**

# Conclusion and Future Directions

**This chapter is mandatory.** Give conclusions and summary of the work done. What were your findings and what were the results? Discuss in detail whether the scope of your project was entirely covered or not and whether the objectives of the project were met or not. What challenges did you face and what has been left out and why?

Sum up all the conclusions of all the chapters here to make a conclusion chapter. Do not repeat any text, just summarize it in different words.

Give recommendations for future work also. How your project can be further enhanced or improved? Future recommendations if someone wants to work on it. **For FYP-1 it is mandatory to list down a plan of the work to be done for FYP-2.**

# References

1. He, Yunling. “Research on the Prediction of US House Prices Based on Machine Learning.” BCP Business & Management (2022): n. pag.
2. Palupi, Endang. “House Price Prediction Using Data Mining with Linear Regression and Neural Network Algorithms.” Jurnal Riset Informatika (2023): n. pag.
3. Liangji Zhu. “Optimization of linear regression in house price prediction”. ACE (2023) Vol. 6: 684-691. DOI: 10.54254/2755-2721/6/20230928.
4. Quang Truong, Minh Nguyen, Hy Dang, Bo Mei, “Housing Price Prediction via Improved Machine Learning Techniques” Procedia Computer Science, Volume 174, 2020, Pages 433-442,ISSN 1877-0509,

# Appendix

## Appendix A: Guidelines

This section should include all supporting information from the project that was not included in the body of the report.  You should include surveys, complex statistical calculations, certain detailed tables and other such information in an appendix.  The information presented in this section is important to support the work presented in the body of the report but would make it more difficult to read and understand if presented within the body of the report.

Cite the appendix items in the report narrative (write "see Appendix A") and organize appendices (e.g., Appendix A, Appendix B,

Any tables, figures, forms, or other materials that are not totally central to the analysis but that need to be included are placed in the Appendix.

## Appendix B: Heading of Sample Appendix B

Following is a sample code with “code” style format.

Void SampleFunction(){

Print “Hello World.”;

}

# Formatting Guidelines

This document also serves as style guide for final year project reports. In order to give a similar high-quality appearance to all final year software project reports this template uses a collection of predefined Microsoft Word formatting styles. **These styles should be used without modification or replacement.** Font in the document is ***“Time New Roman”.*** This template provides following styles:

* **Title** – the main title style
* **Title2** – the subtitle style
* **Body Text** – style for paragraphs
* **Caption** – the style for a figure or table caption
* **Table Description** – the style for description of table, it must be added after caption.
* **Figure Description** - the style for description of figure, it must be added after caption.
* **Code** – the style for program source code

**int x** = 10; // Writing important code

* **Table Header Row** – Style for the header row of table
* **Table Grid** – the style for the data rows in the tables
* **Reference** – The style for references
* **Bullets** – The style for the bullet lists
* **Numbered** **List**– Style for numbered lists

All Heading styles with different level numbers are listed below.

# Heading 1

## Heading 2

### Heading 3

#### Heading 4

##### Heading 5

###### Heading 6

Heading 7

Heading 8

Heading 9

## Tables and Figures

Tables and figures should be centered horizontally. The caption button should be used to insert caption for both the figures and tables. All figures and tables must be numbered properly. Always refer to tables and figures according to their numbers. A table or figure can be cited as follows: ‘see Table1’ or ‘as shown in Table1’. The caption of table should be centered above the table and figure caption should be centered below the figure. Place the tables/figures close to their reference. Use “Table Header Row” and ‘Table Grid’ style for table’s header and data rows respectively. It is compulsory to provide brief description of table/figure after its caption. Styles for table and figure descriptions are “Table Description” and “Figure Description” respectively.

Press Ctrl+Shift+S to see list of styles mentioned above. Figure 1 shows the Apply Style window displaying the list of styles. Select any text then press Ctrl+Shift+S, the Apply Style window will show you the current style applied on that text and if required, you can change the style by selecting any other style from the “Style Name” dropdown.

This is brief description of above figure.

Figure 1: List of Styles

Table 2: This is Sample table caption

This is brief description of following Table.

|  |  |  |  |
| --- | --- | --- | --- |
| Header row | Header row | Header row | Header row |
| Row1 col1 | Row1 col2 | Row1 col3 | Row1 col4 |
| Row2 col1 | Row2 col2 | Row2 col3 | Row2 col4 |

Table 3: This is Sample table caption

This is brief description of following Table.

|  |  |  |  |
| --- | --- | --- | --- |
| Header row | Header row | Header row | Header row |
| Row1 col1 | Row1 col2 | Row1 col3 | Row1 col4 |
| Row2 col1 | Row2 col2 | Row2 col3 | Row2 col4 |

## Equations

Use equation editor to write equations in this report. Use last button of the custom tool bar to invoke equation editor. Similar to tables and figures, equations should also be aligned centered horizontally. Number all equations and insert them in parenthesis. Below is a sample equation and its reference number. An equation can be referenced like this: ‘it is clear from (1)’.

(1)

## Header/Footer

Notice the headers in this document, before Introduction (i.e. the main content of this document) page numbers are in roman numerals. The page numbers of the actual content start with Arabic numerals i.e. 1, 2, 3 and so on. All of the **odd numbered pages** contain title of your project while the **even numbered pages** contain the section heading (i.e. chapter’s name) in the headers.

## Other Formatting Guidelines

* Keep 2-4 GUIs in one page. Consume as much space as possible. Do not leave most of page blank unnecessarily.
* Do not break tables (or use cases) in multiple pages unless the table is too large to fit in one page.
* Re-arrange the content i.e., text, images, and tables properly to meet above two guidelines.

## References

Always refer to the source of information by inserting the reference number in square brackets like this [5]. The reference numbers can either be added at the end of the sentence or within the sentence without changing the punctuation of sentence. A reference can also be cited as follows: ‘as Ruskey [2] mentioned’. List each source only once on your reference page.



Figure 2: IEEE Reference style

This figure represents the styling information for adding references in IEEE format

**Following is a list of sample reference for various typed of sources in IEEE format.**

1. P.M. Morse and H. Feshback, *Methods* of *Theoretical Physics*. New York: McGraw Hill, 1953. **//Format for Book**
2. S.K. Kenue and J.F. Greenleaf, “Limited angle multifrequency diffiaction tomography,” *IEEE Trans. Sonics Ultrason*., vol. SU-29, no. 6, pp. 213-2 17, July 1982. **//Format for Journal Article**
3. B. Tsikos, “Segmentation of 3-D scenes using multi-modal interaction between machine vision and programmable mechanical scene manipulation,” Ph.D. dissertation, Univ. of Pennsylvania, BCE Dept., Philadelphia, 1987. [Add if applicable: University Microfilms, Inc., University of Michigan, Ann Arbor, Michigan.] **//Format for Dissertation or thesis**
4. R. Finkel, R. Taylor, R. Bolles, R. Paul, and J. Feldman, “An overview of AL, programming system for automation,” in *Proc. Fourth Int. Joint Conf Artif. Intell*., pp. 758-765, Sept. 3-7, 1975. **//Format for Proceedings paper**
5. “Technology threatens to shatter the world of college textbooks, *The Wall Street Journal*, vol 91, pp. Al, A8, June 1, 1993. **//Format for Newspaper article**
6. R. Cox and J. S. Turner, “Project Zeus: design of a broadband network and its application on a university campus,” Washington Univ., Dept. of Comp. Sci., Technical Report WUCS-91-45, July 30, 1991. **//Format for Technical Report**
7. M. Janzen, *Instant Access Accounting*. Computer software. Nexus Software, Inc IBM-PC, 1993. **//Format for** **Software**
8. Fuminao Okumura and Hajime Takagi, “Maglev Guideway On the Yamanashi Test Line,” *http://www.rtri.or.jp/rd/maglev2/okumura.html*, October 24, 1998. **//Format for** **World Wide Web** (give author and title if named)
9. “AT&T Supplies First CDMA Cellular System in Indonesia,” http://www.att.com/press/1095/951011.nsa.html, Feb 5, 1996. **//Format for World Wide Web**