

Princess Sumaya University for Technology

Department of Data Science

Semester YYYY/YYYY

**Project Title  
Subtitle (if any)**

**Prepared By:**

Student 1 Name  
Student 2 Name

**Supervised By:**

Supervisor’s Name

Project Submitted in partial fulfillment for the degree of Bachelor of Science in Data Science

Semester-Year

**Declaration of Originality**

This document has been written entirely by the undersigned team members of the project. The source of everyquoted text is clearly cited and there is no ambiguity in where the quoted text begins and ends. The source of any illustration, image or table that is not the work of the team members is also clearly cited. We are aware that using non-original text or material or paraphrasing or modifying it without proper citation is a violation of the university’s regulations and is subject to legal actions.

Names and Signatures of team members:

**Acknowledgments**

Express your appreciation to whoever helped you during your work or even before!

**Summary**

Give a complete but concise description of your work. The summary is a brief overview of your motivation, statement of purpose, general methodological approach, major results, discussion and conclusion. The abstract should not exceed one page.

**List of Abbreviations**

List the abbreviations you have used in your project if there are any and what they stand for.

**UML**: Unified Modeling Language.

**JSP**: Java Server Pages.

**DLL**: Dynamic Link Library.

**CMS**: Content Management System.

**Table of Contents**

The table of contents should be automatically generated by going to: *Insert >> Index And Tables>> Table of Contents.* Choose *Classic* as the format of the table and set the number of levels to be 3.

In order for the table of contents to be generated correctly:

* Each chapter title should be formatted using the style “Chapter”.
* Each section title should be formatted using the style “Section”.
* Each subsection title should be formatted using the style “Subsection”.

The table of contents should replace all of the text that is in this page.

**Table of Figures**

The table of Figures should be automatically generated by going to: *Insert >> Index And Tables>> Table of Figures.* Choose *Classic* as the format of the table and set the tab leader to dots.

In order for the table of figures to be generated correctly, the label of each figure should be formatted using the style “Figure”.

The table of figures should replace all of the text that is in this page.

**Table of Tables**

The table of tables should be automatically generated by going to: *Insert >> Index And Tables>> Table of Figures.* Choose *Table* as the caption, *Classic* as the format of the table and set the tab leader to dots.

In order for the table of contents to be generated correctly, the label of each table should be formatted using the style “Table”.

The table of tables should replace all of the text that is in this page.

**Chapter 1  
Introduction**

**1.1 Overview**

Give a general overview of the project, its importance and why you have chosen to work on it. Provide also a discussion of any scientific/technical background that is required to understand what the project is and the motivation behind it.

You can choose to arrange the information into subsections numbered as 1.1.x. The title of each subsection should be the style named “subsection”.

**1.2 Problem Statement**

In this section provide (at least) the following: 1) A precise description of the problem this project tries to solve, 2) A description of the outcomes of this project (example: a mobile app, a desktop application, etc) and 3) A description of the target audience/customers, how they will use the system and what impact it will have on them.

**1.3 Related Work**

Discuss in detail systems and related research papers that are similar to your proposed project. Provide a critical evaluation of these systems and explain how your system compares to them. Clearly mention if your system uses ideas/features from these systems.

**1.4 Contribution:**

Describe the contribution (s) of the project in terms of

* Novelty in the idea
* The audience that it serves and how
* Novelty in the choice of the model
* Novelty in the structure of the pipeline

**1.5 Document Outline**

Describe how this documentation is structured and what will be discussed in each of the following chapters.

**Chapter 2  
Project Plan**

**2.1 Project Deliverables**

List and describe the deliverables of the system. Examples of deliverables include: source code, documentation files, executables, datasets, databases, etc.

**2.1 Project Deliverables**

List and describe the deliverables of the system. Examples of deliverables include: source code, documentation files, executables, datasets, databases, etc.

**2.2 Project Tasks**

Subdivide the project into high level tasks and provide a timeline for the completion of each task. Tasks can be:

* Analysis
  + requirement identification (e.g: fact finding methods, data and dataset requirements, inputs, outputs, processes, performance, security levels, scalability, etc. )
  + requirement modeling,
  + data size statistical analysis
  + development strategy (e.g. hardware, software tools, programming language use, etc.)
  + Market need assessment - competitors assessment and comparison with project
  + Ethical consideration of data and system
  + any other relevant analysis task (e.g. distribute a survey, interview, etc.)
* Design
  + Architecture Design
  + User interface or system interface including input, output, reports
  + Database Design
* Implementation
  + Coding (e.g. APIs, pipeline, model, interface)
  + Testing
  + Documentation (e.g. prepare a requirements document),

Format the information you provide into a table that includes the following for each task: a task number, a name and description, a time duration, and what dependencies need to be completed before the task begins.

Provide charts like Gantt Charts and PERT Charts to illustrate the timeline of the project.

**2.3 Roles and Responsibilities**

Discuss the roles and responsibilities of each team-member in relation to each of the tasks. Be specific and provide enough details to allow an outsider to judge the workload for each team-member.

**2.4 Risk Assessment**

For each task, describe any associated risks that may prevent completing them. Indicate how probable the risk is, its impact on the system and how you plan to cope with it if it appears.

**2.5 Cost Estimation**

Discuss any costs that are required for implementing the project. Costs may include buying hardware, software licenses, or even costs of paperwork (for example).

**2.6 Project Management Tools**

Describe here any tools you intend to use to manage the development of the system, such as version control software and project management software.

**Chapter 3  
Requirements Specification**

**3.1 Stakeholders**

A stakeholder is any person or entity that is affected by the system or affects in in any sense. In other words, the stakeholders of the system are any users or entities that have an effect on the system requirements.

Describe each stakeholder, his interaction with the system and the importance of his role. Use tables to simplify the provided information.

**3.2 Platform Requirements**

Specify the *software* and *hardware* requirements for running the system. Clearly mention which requirements are a must and which are only recommended.

For systems that are made of sub-systems (e.g. a client side and a server side), make sure to list the requirements for each sub-system separately. For example, the client may need a browser to use the system, whereas the server may require different (more demanding) software and hardware requirements in order to respond to the client requests.

**3.3 Functional Requirements**

Provide a detailed list of all the functional requirements of the system. For each requirement, specify exactly what the input, output, processes and main constraints are. Mark also each requirement as either recommended or essential.

Use a table that contains a numbered list of the requirements and their related information. This will facilitate understanding the requirements and referencing them in the proceeding sections and chapters.

**3.4 Non-Functional Requirements**

Examples of non-functional requirements include: requirements related to performance, storage limits, code quality, documentation, accessibility, security, reliability, scalability, portability, user interface ease-of-use, etc.

Provide an example for each non-functional requirement in your system

**3.5 Other Requirements**

Include here any requirements that may not directly fall under any of the sections before. Examples include restrictions on which APIs can be used, data transmission protocols, data storage formats, etc.

**Chapter 4  
System Design**

In this section, provide the appropriate diagrams with Propper’s justification. Also, it should include the physical model design of your system.

**4.1 Architectural Design:**

Large scale diagram that describes the different system components like interface, database, pipeline, model and the way they are connected.

**4.2 Logical Model Design**

In this section, provide both a high level and low level design of the system you will develop. Here is a list of the diagrams that you may need to provide:

Object Oriented Approach:

* Use Case Diagrams.
* Object Diagrams.
* Package and Class Diagrams.
* Component Diagram.
* Deployment Diagram.
* Activity Diagram.
* Sequence Diagram.
* State Transition Diagram.

Make sure to organize this chapter into sections and subsections in a manner that is appropriate to the provided information. Make sure also to stick the formatting used in the other chapters.

**4.3 Physical Model Design**

System physical design may include the following as needed:

- User Interface/System interface Design

- Database design

- Pipeline design

- Initial model design

**Chapter 5  
Data Preprocessing**

This chapter is needed for data driven projects.

This chapter should describe issues related to the data ETL pipeline, including data extraction, data engineering, feature engineering, and the loading processes. The code functions related to this chapter should be provided in the Appendix. You need to refer to the line of codes while you describe your data pipeline in the following sections.

**5.1 Data Collection and Description**

In this section, you need to provide a detailed description of the data source and data collection process. You need to provide the metadata of your data and show a sample of the data.

**5.2 Data Profiling and Engineering**

This section should show a summary of the data quality issues and data cleaning. You should provide a clear justification of the techniques used. You need to provide examples of sample data before and after the data-cleaning process.

**5.3 Feature Engineering**

This section should show the feature preparation process, including feature extraction, feature rescaling, feature selection, and dimensionality reduction. You should provide a clear justification of the techniques used. You need to summarize the extracted features and provide examples of the prepared features.

**5.4 Data Loading**

This section should summarize the loading process of the prepared data in the data storage.

**Chapter 6  
Implementation**

This chapter should describe issues related to the implementation of the system. Make sure to include the following:

**6.1 General Implementation Description**

* A description of all of the programming languages, tools and APIs used and the reason behind choosing them.
* A general overview of the number of implemented classes, packages, scripts, lines of code, SQL queries, pages, etc.
* Description of any coding conventions used.

**6.2 Pipeline Implementation Description**

* Description of data pipeline structure, integration including any technology and any APIs written to complete the pipeline

**6.3 Model Implementation**

* Description of machine learning model choice and rational for choice
* Description of model development and serving including hyperparameter tuning where applicable

**6.4 Additional Implementation Details**

* The code functions related to this chapter should be provided in the Appendix. You need to refer to the line of codes while you describe your data pipeline in the following sections.
* A discussion of the main/critical algorithms/techniques/architecture developed or used. Describe the reason behind choosing/developing them, the main idea of the algorithm, and provide a pseudo-code for the algorithm and clarifying diagrams for architecture (if needed).
* A listing of which features (mentioned in chapter 2) have been implemented and which have been deferred for future project iterations. Use the same table provided in chapter 2 with an extra column to indicate if the feature has been implemented or not.

Subdivide your discussion into sections as appropriate and do not hesitate to discuss aspects of the implementation that are not listed above.

**Chapter 7  
Testing**

**7.1 Testing Approach**

Discuss the experiment design, including:

* Evaluation measures
* Model testing approach
* Data splitting technique used.
* Pipeline testing approach
* Complete system testing approach

**7.2 Testing Results**

Show and discuss the gathered results from your experiments.

**7.3 Discussion (optional)**

Discuss any insights and test results that have impacted the design or implementation of the system.

**Chapter 8  
Conclusions and Future Work**

Summarize any results achieved in this project and discuss how you intend to extend the project in the future.

**Appendix A  
Users’ Manual**

Provide a description (textual and pictorial) of how the system can be installed and used and how the errors messages should be interpreted.

**Appendix B  
Document Changes**

Discuss here any changes you have made to the document you have submitted in Project 1 and the reasons behind the changes.

**Appendix C  
Code Documentation**

List the code documentation. Such a documentation can usually be automatically generated using tools like JavaDoc and Doxygen.

**References**

Use any referencing style/standard and be consistent. Consider the IEEE citation style. IEEE citation style includes in-text citations, numbered in square brackets, which refer to the full citation listed in the reference list at the end of the paper. The reference list is organized numerically, not alphabetically.