Complete Senior project document template

Contents

Chapter One

1. Introduction

1.1 Background of the Organization/Background of the Project

- Write it in paragraphs
- Write about the background of the organization/project
    - Go from general background to specific

1.2 Statement of the Problem

- Write it in paragraphs
- Write about the current problems that exist and indicate how your project overcomes this problems
- One paragraph should only describe about one problem
- It should have a logical flow from general to specific

1.3 Objective of the Project

1.3.1 General Objective

- Write it in **not** more than two sentences.
- Describe the general objective you aim to achieve by the end of this project

1.3.2 Specific Objectives

- Write it using bullet points
- Write the specific steps you need to follow/take to achieve the general objective
- This should be measurable, achievable and tangible/can be seen in the document and in the implementation
- Each objective should be written in not more than one sentence
- Write it like **: To** identify the requirements for ……
- Should have b/n 5 – 12 objectives

1.4 Methodology

1.4.1 Data collection

- List and describe in short the types of data collection tools you are going to use like: survey/questionnaire, interview, observation, document analysis…etc.
- This tools will be included on the appendix section of the project

1.4.2 System development Process model

- Write it in paragraph
- Write about:
    - what system development process models are and how they help software development.
    - Which model you selected and why you chose it
    - How it suits your project
    - If there is a picture of the model , then include it

1.4.3 Design pattern

- Write in paragraph
- Write what design pattern you chose to follow like MVC, MVVM, singleton pattern or something else.
- Write why you chose this and how it works in general

1.4.4 Programming language

- You can use a list or a table or a paragraph properly
- Write which **languages** you chose from front to back
- In the case of the backend, give reason on why you chose that language

1.5 Tools

1.5.1 Hardware Tools

- You can use a list or a table
- Write any hardware you used(will be using) to develop this system PCs, mobile devices…etc.

1.5.2 Software Tools

- You can use a list or a table and describe each in short
- Include the frameworks, libraries, servers, IDEs, DBMS, runtime environments, browsers…etc.
- Also include the software you used to create the different diagrams like class, ER, …etc. and for UI design

1.6 Scope and Limitation of the Project

1.6.1 Scope of the Project

- Write it in bullet points
- Write about:
    - where it works?
    - For whom?
    - when?
    - on what platform?(web or desktop or mobile, or other)
    - included expected MAIN features?
        - Like what major components it will include

1.6.2 Limitations of the Project

- Write it in bullet points
- The limitations are the opposite of the scope.
- Write about:
    - what features that are normally expected aren't covered by your project?
    - where does it not work?
    - when doesn't it work?
    - who cannot use it?
    - on what platforms can a user not use it?

1.7 Significance of the Project

- Write it in bullet points
- This section explains the importance or potential impact of the project. It could be in terms of business value, social impact, research contribution, etc.
- List its uses for each user and how it benefits the user

1.8 Feasibility Study

- Write in one or two sentence, on the use of Feasibility studies and the type of Feasibility studies considered for this project.

1.8.1 Technical Feasibility

- Write in paragraphs

- Write about whether there exists correct required hardware and software resources and technologies which will be used for project development.
- And whether the selected tools are suitable and can do the job
- Also analyzes technical skills and capabilities of the team

1.8.2 Economic Feasibility

- Write in paragraphs
- Write whether the project is economically viable to develop given the development cost of the project and its operational cost.

1.8.3 Operational Feasibility (Maybe)

- Write in paragraphs
- Write whether it is possible to maintain and operate and keep running after the deployment of the system.
- Whether the proposed system solves the current problem when it's deployed

1.8.4 Legal Feasibility (Maybe- if there are any legal related issues only)

- Write in paragraphs
- Are there any legal restrictions on the development and deployment of the system

1.9 Risk Assessment

1.9.1 Risk

- Write in paragraphs and bullet points
- Mention and discuss the risks that could potentially stop you from developing this system fully
- Also write how you intend to mitigate or handle this obstacles.

1.9.2 Assumptions

- Write in bullet points
- Write the assumptions you are making related to the development, deployment and use of the system.

1.9.3 Constraints

- Write in bullet points
- Write the limitations you have to work with(that you have no control over)

1.10 Work Break Down

- Mainly use Gantt chart

This is a visual representation of the project tasks, their dependencies, and the timeline for completion

CHAPTER TWO

2 Business area analysis and requirement definition

2.1 Introduction

2.2 Business area analysis

2.2.1 Detailed analysis

2.2.2 Current system

2.2.3 Players of the existing system

2.2.4 Proposed system

2.2.5 Forms and reports used

2.3 Requirement Gathering

2.3.1 Requirement Gathering Techniques

2.4 Method of Communication

2.4.1 Communication techniques

2.5 Requirement definition

2.5.1 Functional requirement

2.5.1.1 Essential Use case Modeling

2.5.1.2 Actor description

2.5.1.3 Essential Use Case Description

2.5.1.4 Essential Use Interface Prototyping (Low Fidelity Prototype)

2.5.2 Collaboration modeling

2.5.3 Nonfunctional Requirements

2.6 System modeling

2.6.1 Introduction

2.6.2 System Use Case

2.6.3 UI Identification

2.6.4 Business Rules Identification

2.6.5 Actor Identification

Chapter 3

System Design

3.1 Introduction

- This section provides a brief introduction to the concepts of system design.

3.2 Purpose of the system

- This section explains the main purpose or goal of the proposed system.
- The purpose of the system could be "To provide a user-friendly, efficient, and secure method for hospital staff to access and update patient records."

3.3 Design goals

- These are the goals that the design of the system aims to achieve. They guide the design process and decisions.
- The design objectives might include "Maximize system usability", "Ensure data security", and "Optimize system performance."
- Aspects like scalability, performance, reliability, maintainability, and security.

3.4 Current software architecture (if applicable)

- This section describes the existing software architecture that the proposed system will be based on or replace.
- Discuss the components, modules, and their interactions.
- Discuss the limitations of the current architecture.
- Briefly explain why a new architecture is needed.
- Use diagrams if needed

3.5 Proposed software architecture

- Write a short paragraph about the high level description of the architecture of the system.

- Discuss how the proposed architecture addresses the design goals and requirements.
- Compare with the current architecture (if applicable)

3.5.1 Subsystem decomposition

- This is a decomposition of the system into smaller, manageable subsystems.
- These subsystems will be the components in the component diagram.
- Additionally,
    - Briefly describe each component/subsystem and its purpose.
    - Explain the dependencies between components/subsystem.

3.5.2 Component diagram

- This is a UML diagram that shows the components of the system and their interactions.

3.5.3 Deployment diagram

- This is a UML diagram that shows the physical deployment of the system components on hardware nodes.
- Show how the system will be deployed on servers and devices.
- Explain the deployment strategy and its impact on system performance and reliability.

3.5.4 Persistent data management

- This section describes how the system will manage and store data.
- Identify the types of data(documents, audio, video, user info.,…etc.) that will be stored persistently.
- Discuss the chosen database technology, data storage mechanisms. and data access methods.
- Maybe discuss data backup, recovery, and integrity.

3.5.5 Detailed Database Design

3.5.5.1 Relational tables

- List all the tables in your document in detail with their keys and types

3.5.5.2 Normalization

- Normalize any table that requires it (up to 3rd normal form if necessary)

3.5.5.3 EER

- Draw the extended entity relation diagram

3.5.5.4 OO-Relational mapping

- Show a mapping of the tables onto the objects(classes).

3.5.6 Access control and security

- This section outlines the measures that will be taken to ensure the system's security and control access to its resources.
- Discuss how data privacy and security are maintained.
- Explain user authentication, authorization, and data encryption methods.

3.5.7 Global software control

- This section describes how the overall control flow of the software will be managed.
- describe how requests are initiated and how subsystems synchronize.

3.5.8 Boundary conditions

- This section describes how the system will behave at its boundaries, like startup, shutdown, error conditions, extreme conditions like high user traffic/load …etc.
- specific extreme or unusual scenarios

And how the system will handle these conditions?

CHAPTER FOUR

Implementation

4.1 Development Tools
- 4.1.1 Server
  - 4.1.1.1 Server setup and configuration
- 4.1.2 Database
  - 4.1.2.1 Data storage and retrieval
  - 4.1.2.2 MVC or other design pattern followed
  - 4.1.2.3 Back End
- 4.1.3 API development (if used)
- 4.1.4 Database interactions (CRUD operations, data handling)
- 4.1.5 ORM (if used)
- 4.1.6 Authentication and authorization mechanisms
- 4.1.7 Session, tokens, cookies …etc.
- 4.1.8 Security
- 4.1.9 Any special algorithm used
- 4.1.10 Third party API integration
- 4.1.11 Payment service
- 4.1.12 Email…etc.
- 4.1.13 Front End
- 4.1.14 One Example for few components or one page…
  - 4.1.14.1 Some frontend tasks like filtering, front end security, routing, responsiveness
  - 4.1.14.2 API integration

4.2 Prototype of the System