

Software Design Document

Time Table Generator For a College

Prepared by:

Govindu Leela Krishna(PES1UG21CS209)

Girish YP(PES1UG21CS207)

G Bharadwaja Pranav(PES1UG21CS190)

Harikrishna(PES1UG21CS217)

Version 1.0

Team: World Of Software

Revisions

Version	Primary Author	Description of Version	Date Completed

Table of Contents

1.Introduction

1.1 Purpose

1.2 Scope

1.3 Definitions, Acronyms and Abbreviations

2.System Overview

2.1 Backend

2.2 Frontend

3.System Components

3.1 Decomposition Description

3.2 Dependency Description

3.3 Class Diagrams

1. Introduction

1.1 Purpose

The purpose here is like setting the project's main goal – to make a clear plan for our college timetable system. This document is intended to guide developers and stakeholders through the system's design. It serves the purpose of clearly defining the project's confines, articulating precisely what elements fall within its purview, and distinguishing those that lie outside of it.

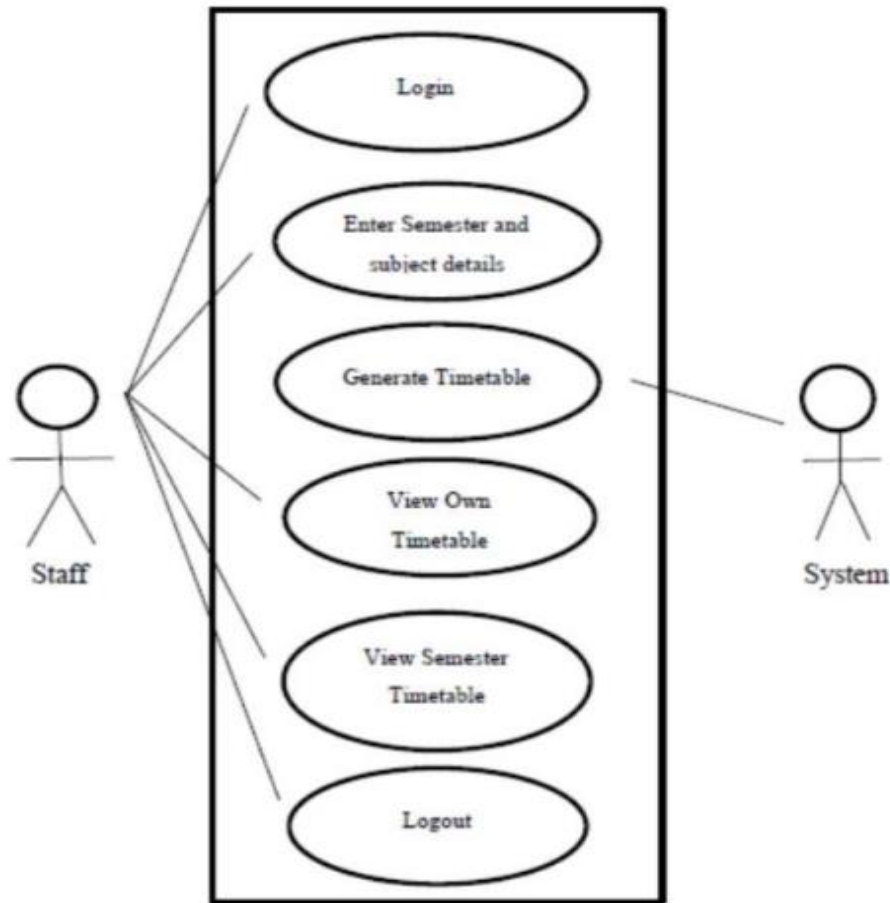
1.2 Scope

The establishment of a well-defined scope is instrumental in ensuring that the project adheres to its intended objectives without deviating into extraneous domains. It encompasses the design aspects, dependencies, and interactions within the software system. The SDD provides a clear picture of how this software addresses the college's time table generation requirements and ensures that we stay on track and only work on what's necessary.

1.3 Definitions, Acronyms and Abbreviations

Acronym	Meaning
TTG	Time Table Generator
SDD	Software Design Document
GUI	Graphic user Interface
API	Application Programming Interface
CTMS	College Time Table Management System

2. System Overview



Backend:

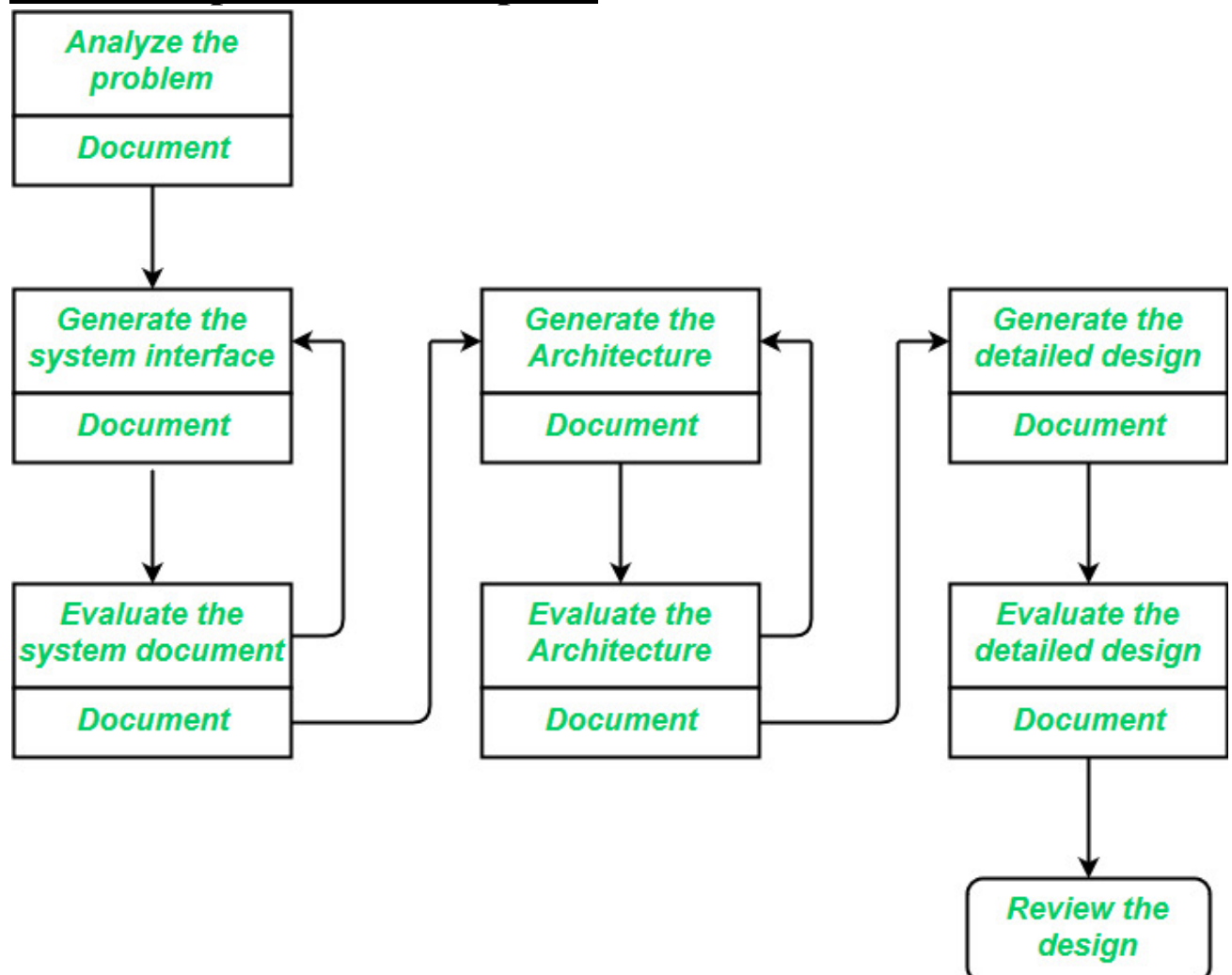
The backend component of our project, the College Timetable Management System (CTMS), primarily assumes the role of data management and facilitates communication with the front-end. We have chosen Python as the primary programming language to power the backend, primarily due to its extensive ecosystem of libraries and its well-suited capabilities for web development. Within this system, Python functions as the engine that processes data and manages the logic behind our timetable management. To ensure the persistence and accessibility of our data, we have adopted JSON (JavaScript Object Notation) files as the storage method. This choice guarantees that project-related information is stored securely and allows for swift retrieval when necessary.

Front End:

For the front-end component of our CTMS, we have harnessed the capabilities of React, a renowned web application framework. React empowers our system with an intuitive and user-friendly interface, rendering interactions with the platform smooth and uncomplicated. Its design philosophy is rooted in ease of use, enabling both college administrators and students to effortlessly engage with the system without requiring an extensive technical background. The use of React ensures that our timetable management system is approachable and user-centric.

3. System Components

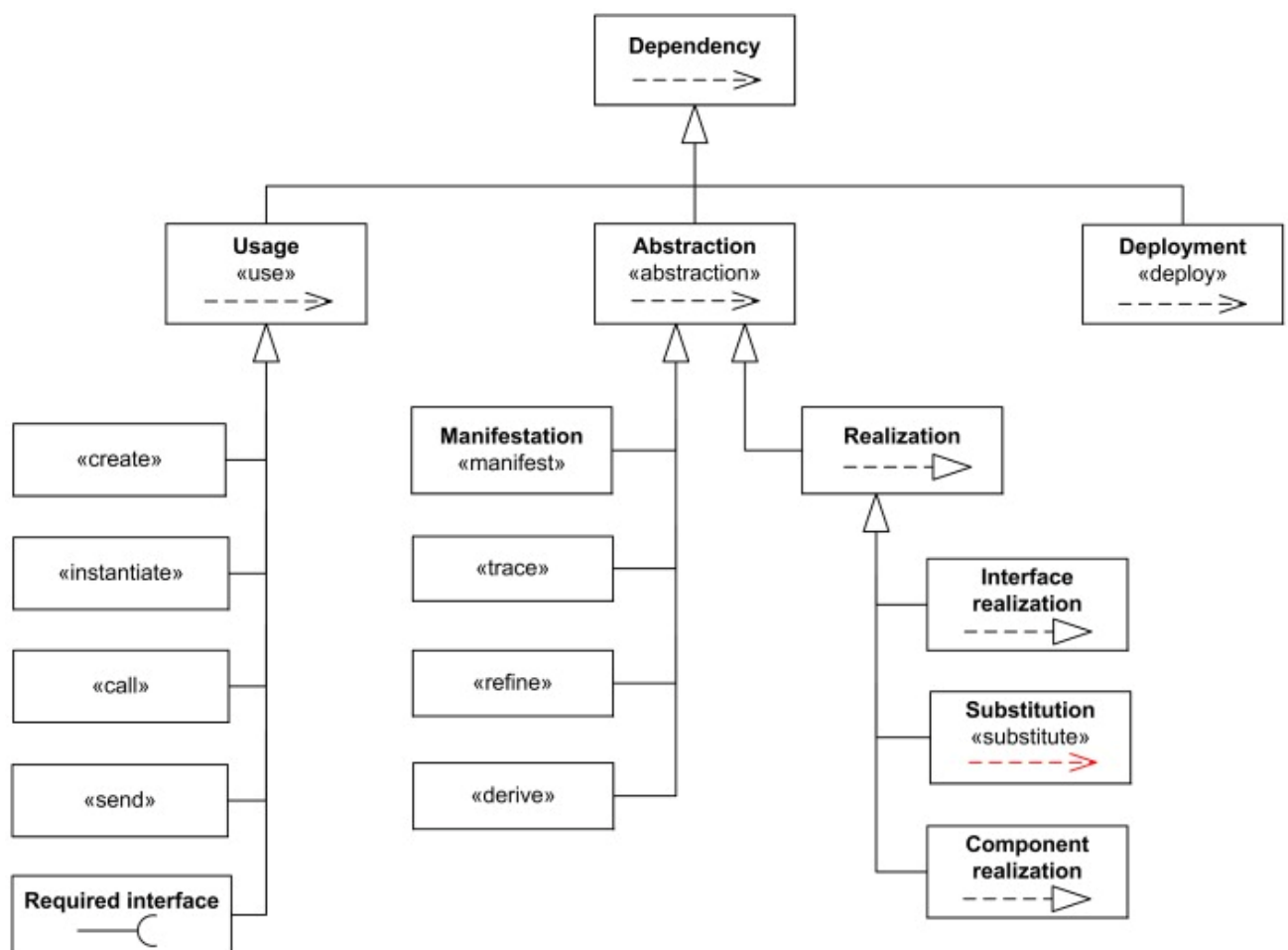
3.1 Decomposition Description



This document describes the problem that the design is intended to solve, as well as the requirements for the solution. Once the problem document has been analyzed, the

next step is to generate a system interface document. This document should describe the interfaces between the different components of the system, as well as the interactions between the system and its users. After the system interface document has been generated, the next step is to generate an architecture document. This document should describe the overall architecture of the system, including its components and how they interact with each other. The final step in the design process is to generate a detailed design document. This document should provide a detailed description of each component of the system, including its design and implementation. Once the detailed design document has been generated, the design should be reviewed to ensure that it meets all of the requirements and that it is feasible to implement.

3.2 Dependency Description



Components:

- User Login: This module allows users to log in to the CTMS.
- Project Display: This module allows users to view a list of all available projects.
- Tabular View: This module allows users to view a tabular representation of a selected project's timetable.
- Gantt-Chart View: This module allows users to view a Gantt chart representation of a selected project's timetable.
- Data Manager: This module records and manages all changes to the timetable.

Interdependencies:

- The User Login module is required by all other modules, as users must first log in to access the CTMS.
- The Project Display module is required by the Tabular View and Gantt-Chart View modules, as users must first select a project to view its timetable.
- The Tabular View and Gantt-Chart View modules are interdependent, as changes made to the timetable in one view are reflected in the other view.
- The Data Manager module is required by all other modules, as it is responsible for recording and managing all changes to the timetable.

3.3 Class Diagram

