TASKMATE
All-in-One Productivity Platform

Submitted by:
ADIL UR REHMAN (2024-CS-760)
ATIKA ASIF (2024-CS-498)
SHAWAL UBAID (2024-CS-484)
MEERAB IHTISHAM (2024-CS-459)

Supervised by:
Dr. Natasha Nigar

Department of Computer Science
Rachna College of Engineering and Technology
(A Constituent College of UET Lahore)

Contents

1 Problem Statement

In today's digital world, people use multiple websites and applications to create documents such as CVs, invoices, and business cards. This is time-consuming and inefficient. There is a lack of a unified platform where all these tools are available for free in one place. **TaskMate** solves this problem by offering an all-in-one solution that combines multiple productivity tools on a single, free, and user-friendly web platform.

TaskMate not only provides essential productivity tools in one platform but also allows users to contribute by adding their own tools. This makes it a continuously evolving community-driven platform where contributors receive proper credits and recognition for their work.

2 Introduction

TaskMate is a web-based platform (WBP) that provides tools like CV Generator, Resume Generator, Invoice Generator, and Card Generator. It uses **Node.js** as the backend to handle server-side logic and **MySQL** as the main database to manage user and tool data. For quick or temporary storage (e.g., preview data), **JSON files** are used to ensure fast and lightweight data handling. TaskMate also allows users to **create and add their own tools** to the platform. Contributors who develop new tools are **featured and credited** within the system, making TaskMate a **community-driven and continuously expanding platform**.

3 Key Feature

The most distinctive feature of **TaskMate** is its **user contribution system**. Unlike traditional platforms that only provide fixed tools, TaskMate allows users to **create and add their own tools** to the platform.

Once a tool is submitted and verified, it becomes part of the main website, and the **creator is credited as a featured contributor**. Their name or profile will appear alongside their tool, acknowledging their contribution to the community.

This feature transforms TaskMate into a **collaborative and evolving ecosystem** rather than a static collection of tools. It promotes **creativity, innovation, and community engagement**, allowing users to not only use the platform but also **expand and improve it** over time.

4 Project Charter

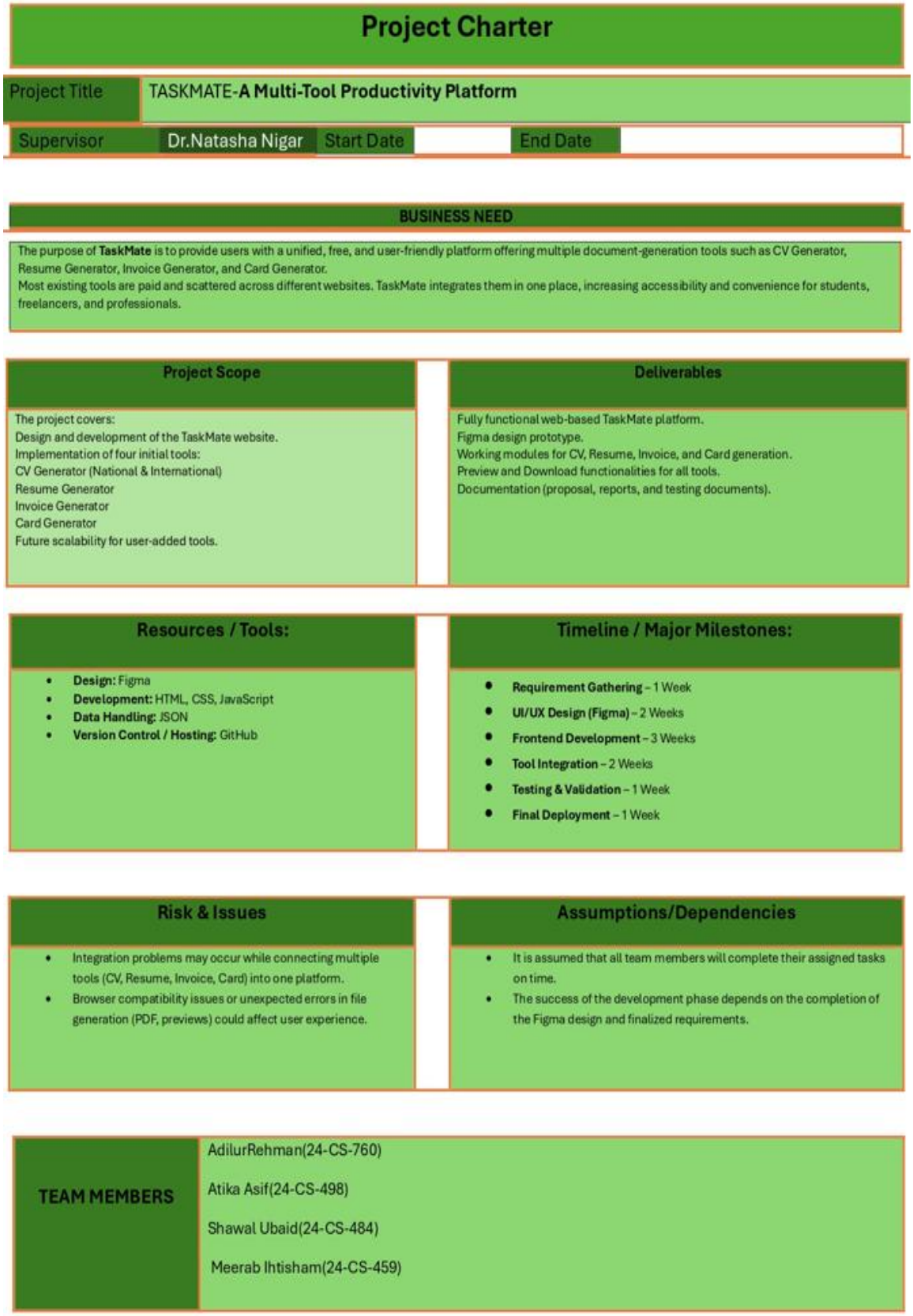

Fig 4.1:Project Charter for TaskMate

5 Literature Review

- **TaskMate** differentiates itself by combining multiple utilities on one site.
- Similar platforms are either paid or limited in scope, while TaskMate emphasizes **free usage** and **multi-tool integration**.

6 Gantt Chart

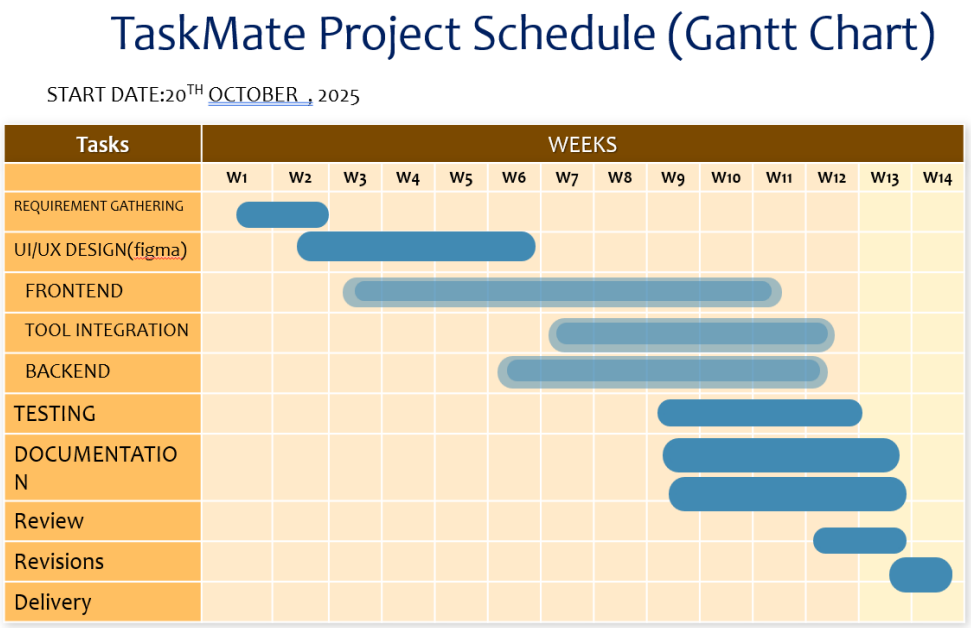

Fig 6.1: Gantt Chart for TaskMate Project Development Phases.

- **Requirement Analysis:**

Identify user needs and select tools (CV, Resume, Invoice, Card Generators).

- **Design:**

Create user interface prototypes in **Figma** and define data flow structure.

- **Development:**

Build the **frontend** using HTML, CSS, and JavaScript.
Use **Node.js** for backend logic and **MySQL** for data storage.
JSON files are used temporarily for quick data handling.

- **Testing:**

Perform functionality and UI testing to ensure smooth operation.

- **Deployment:**

Deploy the web application on **GitHub Pages** or **Netlify** for public access.

Work Break Down Structure

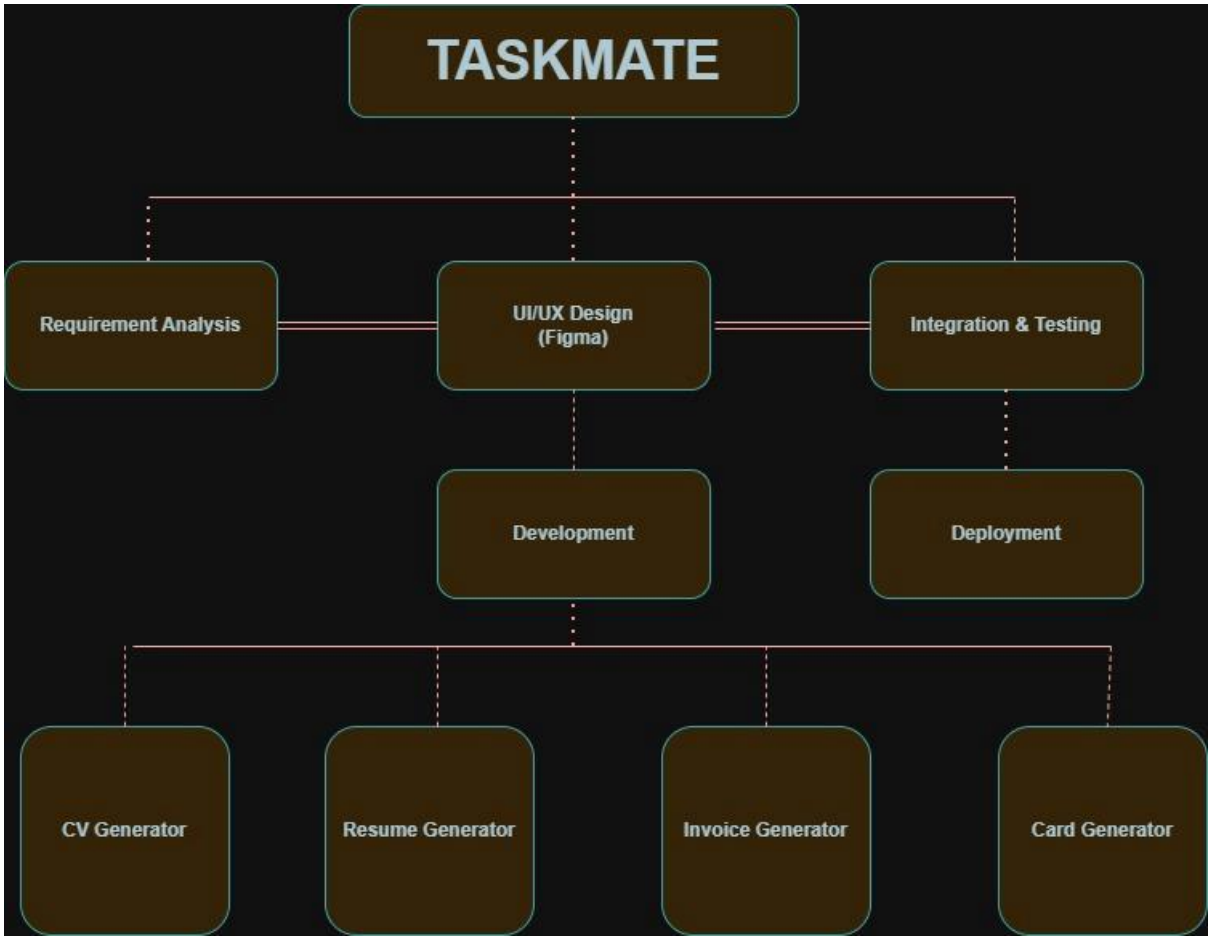

Fig 6.2: Work Breakdown Structure (WBS)

7 Data Set

TaskMate does not rely on any external dataset.
All data is **generated and managed internally** by the application.

- **MySQL Database** stores user details, tool data, and generated document information.
- **JSON files** are used temporarily to store user inputs and previews before final submission or saving.
- This approach ensures **fast access**, **lightweight processing**, and **easy scalability** for future tool additions.

8 Tech Stack

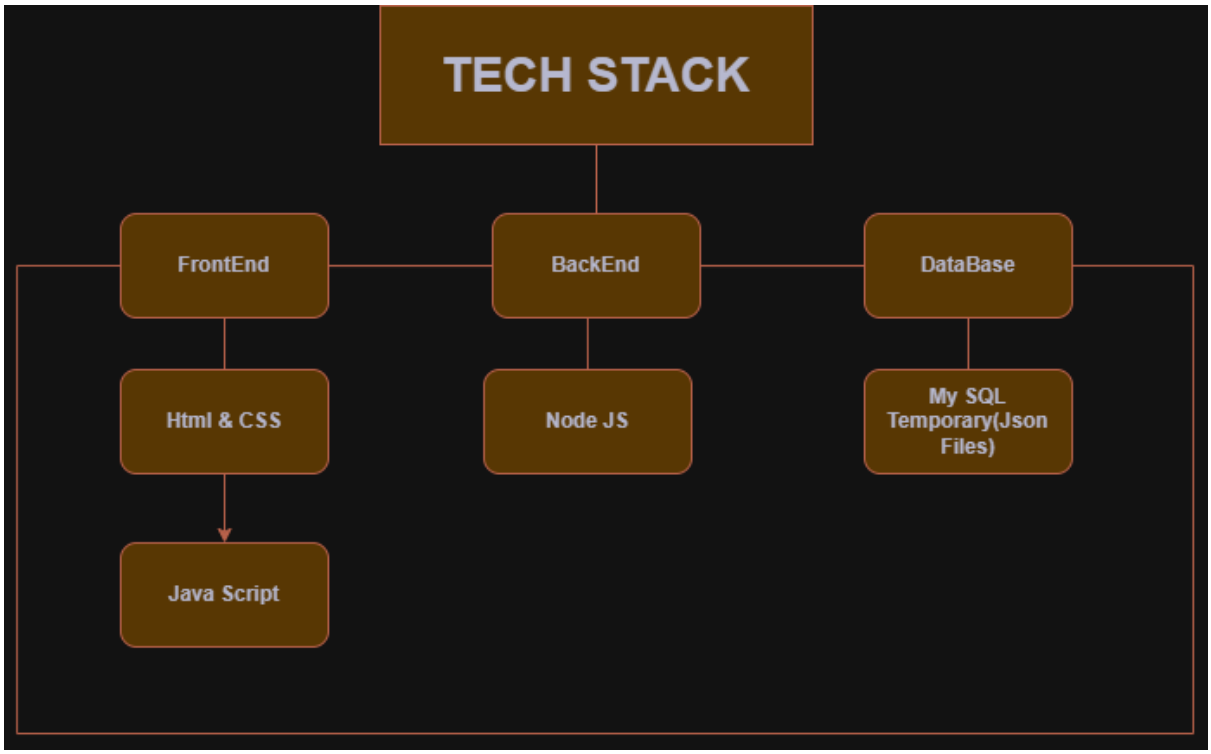

Fig 8.1: Technologies used for development purpose

8.1 Frontend

The frontend of **TaskMate** is designed to be **simple, responsive, and user-friendly**. It is developed using standard web technologies:

- **HTML5** – for structure and layout
- **CSS3** – for styling, responsiveness, and visual design
- **JavaScript (ES6)** – for interactivity and tool functionality

The interface will be designed in **Figma** before implementation to ensure a clean user experience.

8.2 Programming Languages

- **HTML, CSS, JavaScript** — for frontend design and interactivity
- **Node.js** — for backend logic and server-side processing
- **SQL** — for database queries and management in MySQL

8.3 Database

TaskMate uses **MySQL** as the primary database to manage user information, tool data, and generated document records.
It ensures data consistency, scalability, and secure storage for all platform operations.

Additionally, **JSON files** are used as **temporary storage** for handling quick operations such as live previews and unsaved inputs.

8.4 Backend

The backend of TaskMate is built using **Node.js**, which handles:

- Communication between the frontend and database
- User data and tool management
- JSON file operations for temporary data handling

9 Team Members Contributions

NAME	ROLL NO.	CONTRIBUTIONS
ADIL UR REHMAN	24-CS-760	Responsible for **coding and implementation** of the project using HTML, CSS, JavaScript, and Node.js.
ATIKA ASIF	24-CS-498	Created the **UI/UX designs in Figma**, ensuring a clean and user-friendly interface.
SHAWAL UBAID	24-CS-484	Prepared the **project documentation and proposal**, maintaining overall project coordination.
MEERAB IHTISHAM	24-CS-459	Performed **testing, debugging, and validation** of all modules before deployment.

10 Conclusion

TaskMate is a user-friendly, all-in-one productivity platform designed to simplify document creation by providing multiple tools on a single free platform.
It eliminates the need to switch between different apps, offering convenience and scalability for future enhancements.

TaskMate not only simplifies access to productivity tools but also empowers its user community by allowing them to contribute new tools. This credit-based system promotes innovation and ensures the platform remains dynamic and user-driven.