

Paragon Task Management System: **Task-Tracker**
Project Charter
Version: 1.0

VERSION CONTROL

Version	Author	Role	Changes	Date
Draft	Mahir Rahman		Initial version	3/5/2024

1 PROJECT BACKGROUND

The system is a comprehensive software ticket and task management application designed to streamline the process of managing projects and support tasks within an organization. The application serves as a central platform for users to report issues, assign tasks, track progress, and analyze performance metrics across various projects.

2 PROJECT STRUCTURE

The goal is to deliver the service by May 30th, 2024 with the following modules:

- Unique ID user generation
- Personal Customizable Dashboard
- Global Task Viewing Dashboard
 - Ticket Management System for task generation
 - Task assignment capabilities
- Collaborative Tools
- Reporting and Analytics
- Calendar Module for event viewing and editing (personal and global)
- Notifications
- Access Control and Security
- FAQ Panel -> Used for Posting Potential System Updates (requests made by users) or App FAQ for system Help.

Target Users:

- **Project Managers:** Who need to oversee project progress and resource allocation.
- **IT Support Teams:** For managing and responding to support tickets efficiently.
- **Team Members:** Execute tasks and require coordination and communication tools to collaborate effectively.

3 BUSINESS NEEDS

This system is designed to meet critical business needs in the realm of project management and IT support across diverse industries. Enterprises require a robust tool to efficiently manage tasks, track issues, and facilitate seamless communication among team members, regardless of their physical location. This application addresses the need for improved task visibility and accountability, ensuring that every task and ticket is tracked from inception to resolution. It also enables better resource management by allowing project managers to allocate tasks based on real-time availability and expertise, thus optimizing workforce productivity. Additionally, TaskTracker offers extensive reporting and analytics features, which are essential for businesses aiming to make informed decisions based on performance metrics and trends. These capabilities ensure that the organization can adapt to changes quickly, maintain high service levels, and achieve operational excellence, all of which are crucial in maintaining competitive advantage in a dynamic market environment.

4 SCOPE

The scope of the TaskTracker project encompasses the development and implementation of a comprehensive task and ticket management system designed to streamline project management, issue tracking, and team collaboration within an organization. TaskTracker will offer robust features including

task assignment and scheduling, real-time notifications, ticket handling from creation to resolution, customizable user dashboards, and detailed reporting and analytics. The system will cater to various user roles such as administrators, project managers, and team members, each with tailored access and functionality to suit their specific needs. Additionally, the application will integrate with existing workflows and third-party tools to enhance its utility without disrupting current operations. This project will also ensure high levels of data security and system reliability to support continuous business operations. The end goal is to provide a scalable solution that improves operational efficiency, enhances communication and collaboration among team members, and supports data-driven decision-making through comprehensive analytics.

4.1 IN SCOPE

- Planning Phase
 - Determine the number of databases required + allocate storage space for each DB
 - Identify hosting requirements
 - Gather User based Requirements
 - Gather Legal Requirements.
 - Create UI/UX outlines
- Execution Phase
 - Develop an Online Service based upon architecture research in the planning phase
 - Iterate the service based on user feedback
 - Create training tutorials built on the app for ease of use.

4.2 OUT OF SCOPE

- Web public service hosting (only in-house for now)
- API development + Third Party Service Integration

5 SCOPE STATEMENT INFORMATION

5.1 PROJECT COMPLETION STATEMENT (MAJOR DELIVERABLES)

- Producers can list products, have orders placed, and be notified of incoming orders.
- Advertisers can buy ad space.
- Customers can browse products, place, pay for, and receive orders.
- Service is hosted and public.

5.2 MAJOR MILESTONES

- Milestone 1: Planning Phase Completion
- Milestone 2: Prototype Development
- Milestone 3: Testing and Feedback on Prototype Completed
- Milestone 4: Final Development
- Milestone 5: Launch with Comprehensive Documentation

5.3 PROJECT ASSUMPTIONS

We are assuming:

- Project Managers will actively use and provide feedback on the system.
- IT Support Staff will engage with the system to manage tickets.
- Users have the necessary devices and internet access to use the service.
- Hosting services provided will ensure the desired uptime.
- Database and backend services will handle the expected traffic efficiently.

5.4 PROJECT CONSTRAINTS

The projected constraints:

- The development team's size may affect the speed and scope of development.
- The service performance is limited by the hosting architecture.
- User experience is dependent on the reliability and scalability of the system infrastructure.

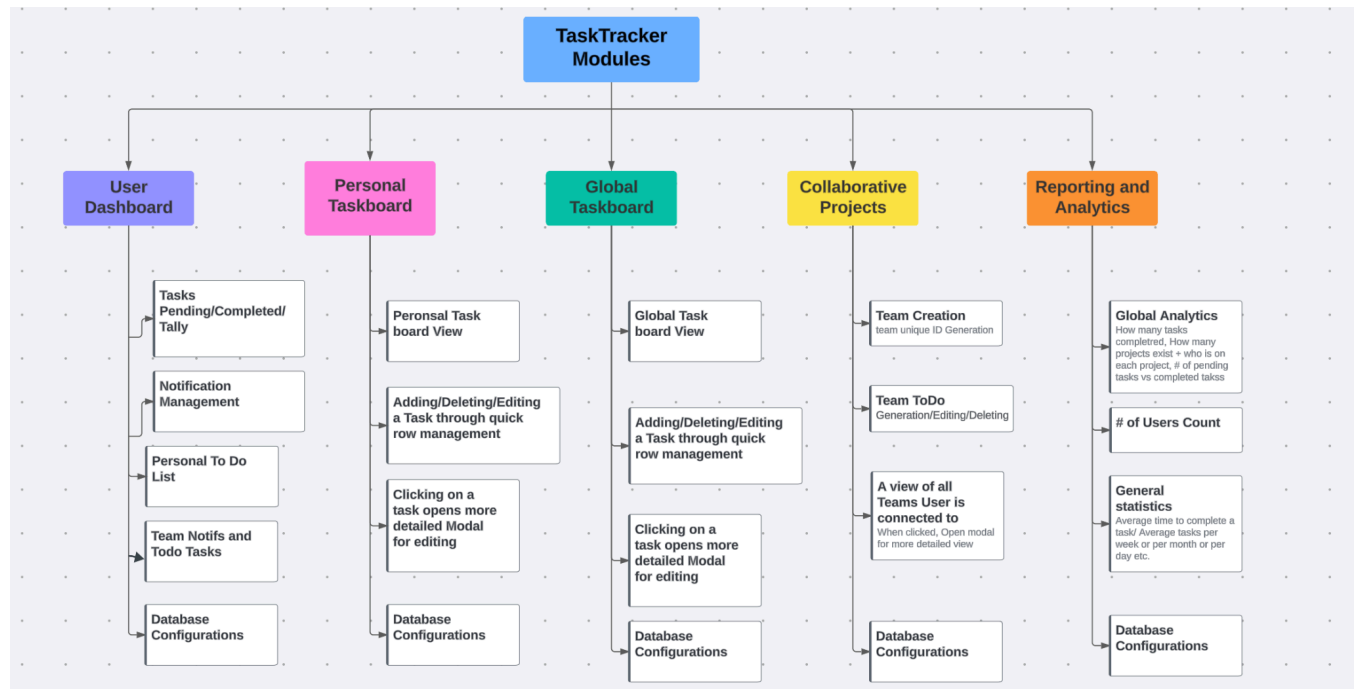
5.5 PROJECT DEPENDENCIES

The Dependencies for the project:

- Active participation from all user roles for real-world feedback and testing.
- Reliable and scalable infrastructure from service providers.
- Ongoing development and maintenance to adapt to evolving user needs and technological advancements.

6 WORK BREAKDOWN STRUCTURE

https://lucid.app/lucidspark/0c8d138c-f662-4b01-b6a5-7f01c6fcd821/edit?viewport_loc=1702%2C721%2C3200%2C1571%2C0_0&INVITATIONID=INV_03019287-84b8-4086-bbe0-bda6fcef6f8a



6.1 WBS LEVEL BREAKDOWN

6.1.1 TECHNICAL SPECIFICATIONS

6.1.2 FUNCTIONAL REQUIREMENTS

- Project Managers can create and manage projects, assign tasks, and monitor progress.
- Team Members can view assigned tasks, update statuses, and log hours worked.
- IT Support Staff can create, manage, and resolve support tickets.
- Administrators can configure system settings, and manage user roles, and permissions.

6.1.3 NON-FUNCTIONAL REQUIREMENTS

- The application must be responsive with an uptime of 99.99%.
- Hosted on a reliable cloud service with a robust infrastructure.
- Secure storage and processing of user data.
- Scalability to handle growing numbers of users and tasks.

6.1.4 REGULATORY REQUIREMENTS

- Compliance with data protection laws for storing user information.
- Adherence to software development and cybersecurity regulations.

6.1.5 SOFTWARE REQUIREMENTS

- Task and project management functionalities.
- Ticketing system for IT support.
- User account management system for settings and role configurations.
-

6.2 PROJECT WORKING STRUCTURE

6.2.1 PLATFORM DESIGN

- UI/UX design for User and Admin interfaces.
- System architecture design.
- Security and compliance design.

6.2.2 PLATFORM DEVELOPMENT

- Front-end Development (user interfaces for administrators, project managers, team members, and IT support).
- Back-end Development (database management, user role management, task and ticket processing).
- Integration of Third-party Services (cloud hosting, email services, API integrations).

6.2.3 TESTING AND Q/A

- Functional Testing (user role functionalities, task management, ticket resolution).
- Non-functional Testing (system load, performance, security audits).
- Feedback Collection and Iteration based on internal and beta tester feedback.

6.2.4 TRAINING AND DOCUMENTATION

- Develop Training Materials for all user roles within the system.
- Create User Guides and FAQs for system functionalities and common issues.

6.2.5 LAUNCH/GO-LIVE

- Final System Checks and Validation to ensure all features function as expected.
- Marketing and Communication Plan Execution to promote the system internally and externally.
- Go live and Operational Monitoring to handle any immediate issues post-launch.