WEB-BASED TIME AND PRODUCTIVITY ANALYSIS

A PROJECT REPORT

Submitted by,

ASHA S-20201CSE0787

VAISHNAVI R-20201CSE0812

ROSELINE C-20201CSE0815

Under the guidance of,

Ms. Sreelatha P K

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

At

PRESIDENCY UNIVERSITY

BENGALURU

JANUARY 2024

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Project report **"WEB-BASED TIME AND PRODUCTIVITY ANALYSIS"** being submitted by "ASHA S, VAISHNAVI R and ROSELINE C" bearing roll numbers "20201CSE0787,20201CSE0812 and 20201CSE0815" in partial fulfillment of the requirement for the award of degree of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out under my supervision.

Ms. SREELATHA P K
Assistant Professor
School of CSE
Presidency University

Dr. PALLAVI R
Professor& HOD
School of CSE
Presidency University

Dr. C. KALAIARASAN
Associate Dean
School of CSE
Presidency University

Dr. L. SHAKKEERA
Associate Dean
School of CSE
Presidency University

Dr. Md. SAMEERUDDIN KHAN
Dean
School of CSE
Presidency University

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We as a result of this declare that the work, which is being presented in the project report entitled **WEB-BASED TIME AND PRODUCTIVITY ANALYSIS** in partial fulfillment for the award of Degree of **Bachelor of Technology** in Computer Science and Engineering, is a record of our investigations carried out under the guidance of **Ms. Sreelatha P K, Assistant Professor, School of Computer Science and Engineering, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

Name	Roll no	Signature
ASHA S	20201CSE0787	Asha. S.
VAISHNAVI R	20201CSE0812	vaishnavi
ROSELINE C	20201CSE0815	Roseline.c

ABSTRACT

Effective time management is crucial for the success of software development projects. This project proposes a comprehensive method designed to optimize the allocation and tracking of time resources throughout the software development life cycle. The proposed method aims to calculate and manage time spent on various project activities, ensuring efficient programming of tasks and activities.

The method presented in these abstracts addresses the unique challenges faced in software development projects by providing a structured framework for time allocation. By systematically tracking and managing time, development teams can gain insights into resource utilization, identify potential bottlenecks, and make informed decisions to enhance overall project efficiency**.** For the successful project management and its success, many studies have been done but still, many of the software projects cannot end up well.

One of the main reasons behind it is time management and also analyzing productivity. So, the time tracking software is being developed, this will collect and analyze the insights to improve overall efficiencies, like Identifying inefficiencies, Resource Allocation, enhancing team efficiency, task management, performance measurement, and ensuring successful project outcomes. By applying these insights, organizations can unlock their full potential and achieve their goals effectively.

ACKNOWLEDGEMENT

First of all, we are indebted to the **GOD ALMIGHTY** for allowing me to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected **Dr. Md. Sameeruddin Khan**, Dean, School of Computer Science and Engineering, Presidency University for getting us permission to undergo the project.

We record our heartfelt gratitude to our beloved Associate Deans **Dr. Kalaiarasan C and Dr. Shakkeera L,** School of Computer Science and Engineering, Presidency University, and **Dr. Pallavi R.**, Head of the Department, School of Computer Science and Engineering, Presidency University for rendering timely help for the successful completion of this project.

We are greatly indebted to our guide **Ms. Sreelatha P K,** Assistant Professor, School of Computer Science and Engineering, Presidency University for her inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the University Project-II Coordinators **Dr. Sanjeev P Kaulgud, Dr. Mrutyunjaya MS,** and also the department Project Coordinators **Mr. Mohammed Zia Ur Rahman, Mr. Peniel John Whistely**.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

ASHA S

VAISHNAVI R

ROSELINE C

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER-1

INTRODUCTION

1.1THE OPENING PERSPECTIVE

The modern business landscape is dynamic and fast-paced, requiring organizations to efficiently manage their tasks and projects to stay competitive. A robust Project Management System (PMS) plays a crucial role in enhancing productivity by providing a structured approach to task allocation, monitoring, and completion. This project centers around the creation of a dynamic web application that will serve as a comprehensive tool for organizations to measure, analyze, and improve their time management and productivity. It will not only provide real-time insights into how time is allocated across various tasks but also offer valuable performance metrics to help teams and individuals achieve their goals more effectively. This report aims to analyze the significance of implementing a Task Management System to improve productivity within an organization. This project centers around the creation of a dynamic web application that will serve as a comprehensive tool for organizations to measure, analyze, and improve their time management and productivity. It will not only provide real-time insights into how time is allocated across various tasks but also offer valuable performance metrics to help teams and individuals achieve their goals more effectively.

The Essence of Time: In the grand symphony of existence, time stands as an invaluable asset, both finite and irreplaceable. Its scarcity prompts us to contemplate its optimal utilization, instigating a compelling need for effective time management. As we navigate the intricate tapestry woven by our commitments, aspirations, and obligations, the ability to harness time emerges as a linchpin for success and fulfillment. Each moment becomes a precious resource, waiting to be channeled with purpose and intentionality.

1.2AN OVERVIEW

Time and productivity analysis can be defined as the process of systematically examining how time is utilized and how it affects the overall productivity of individuals, teams, or organizations. Time Management refers to the practice of allocating and using time wisely to accomplish tasks and achieve objectives efficiently. Effective time management helps reduce

stress, increase work-life balance, and boost individual and team performance. Productivity is a measure of how efficiently resources, including time, are used to produce desired results. Improved productivity leads to higher outputs with the same or fewer resources.

It is essential for competitiveness and profitability in business and overall personal effectiveness. The main objectives of time and productivity analysis are to optimize resource allocation, improve efficiency, and enhance overall productivity Analyzing time and productivity is crucial for organizations as it allows them to identify bottlenecks, streamline processes, and make informed decisions to achieve better outcomes. The goal of this web-based software project is to create a comprehensive tool to track and analyze resource activity to improve time management and productivity in businesses. Users of the system can track time spent on a variety of tasks, such as coding, writing documentation, using SQL, and using the internet.

As we embark on this exploration of time management, the journey calls us to delve into the principles, strategies, and real-world applications that empower us to master the flow of time. This expedition invites us to cultivate a harmonious relationship with time, providing the tools to navigate the sands of our days with purpose, resilience, and the unwavering assurance that each moment is a canvas awaiting the strokes of intention and accomplishment. Together, let us unravel the mysteries of time, transforming it from an abstract concept into a tangible force that propels us toward a future defined by our deliberate choices and meaningful achievements.

CHAPTER-2

LITERATURE SURVEY

2.1 TIME MANAGEMENT PERCEPTIONS IN SOFTWARE TEAMS

In software development environments, the effective management of time in the context of perceptions within teams has emerged as a critical challenge. Software teams, driven by the dynamic nature of the industry and evolving user expectations, often struggle to balance efficient time utilization with the need to maintain a comprehensive and accurate perception of the product under development. The confluence of rapid technological advancements, shifting market demands, and the inherent complexity of software projects present a multifaceted problem that demands immediate attention.

- Ensure a more accurate alignment of internal team perceptions with external stakeholder expectations by implementing transparent communication channels and regular feedback mechanisms.
- Develop and implement a prioritization matrix that aids teams in setting clear goals and prioritizing tasks based on their impact on product perception and overall project success.
- Improve the accuracy of time estimates for project tasks by implementing reliable time estimation techniques, fostering collaboration among team members, and continuously refining estimation processes.
- Implement transparent communication protocols, including regular stand-up meetings, sprint reviews, and retrospectives, to ensure that all team members have a shared understanding of project goals, timelines, and challenges

2.2 OPTIMIZATION AND TIME MANAGEMENT OF WEEKLY CLASS SCHEDULE

Many students face challenges in optimizing and effectively managing their weekly class schedules, leading to issues such as missed deadlines, increased stress, and reduced overall productivity. The lack of a structured approach to time management often results in a suboptimal balance between academic commitments, extracurricular activities, and personal well-being.

2.3 LINEAR PROGRAMMING AND ITS APPLICATION TO SCHEDULING AND TIME MANAGEMENT:

Linear programming (LP) optimization has been known to be applied in many different fields, such as in manufacturing, food distribution, and renewable energy among other systems. The case successfully compared conventional approaches to the proposed novel approach of LP. Moving forward, it can be seen how LP can benefit many operations concerned with scheduling optimization. Therefore, it may be initially inferred that LP can also benefit personal scheduling for academic time management and productivity.

- **Prioritization and Task Management:**
  - Students to identify and prioritize Enable tasks effectively.
- **Effective Schedule Planning:**
  - Assist students in creating well-balanced weekly schedules that include dedicated study sessions, breaks, and extracurricular activities.
- **Digital Tools and Technology Integration:**
  - Encourage the use of digital planners, calendar apps, and productivity tools to streamline workflow and enhance organization.
- **Optimized Productivity:**
  - Educate students on the detrimental effects of multitasking and promote strategies for focused work.

2.4 EVOLVING TIME-MANAGEMENT-BASED PREDICTION FOR QUALITY CRITERIA IN A MULTI-STAGE PRODUCTION PROCESS

In a complex multi-stage production process, maintaining high-quality output is crucial or meeting customer expectations and industry standards. One of the key challenges in this context is predicting and managing the time required for each stage of production to ensure the final product's quality.

- Enhance Production Efficiency:
- Optimize Resource Utilization:
- Improve Quality Criteria Prediction:
- Mitigate Production Delays:

- Adapt to Dynamic Environmental Factors:
- Facilitate Cross-Stage Collaboration:
- Enable Real-Time Monitoring and Feedback

2.5 TIME MANAGEMENT METHOD FOR SOFTWARE DEVELOPMENT PROJECTS

The article aims to propose a method that calculates the time spent on the specific software development project and provides effective programming of activities.

- The objectives of the article are to propose a method for project time estimation and effective management in software development projects.
- The article aims to define inputs, activities, tools, roles, and output artifacts for each phase, allowing for the calculation of time in each phase.
- The article emphasizes the importance of effective project management in achieving project objectives within budget and estimated time.

2.6 IMPROVED MANAGEMENT OF ISSUE DEPENDENCIES IN ISSUE TRACKERS OF LARGE COLLABORATIVE PROJECTS

The problem statement of this article is to provide solutions for a better management-dependent issue in issue trackers, specifically in the context of a large collaborative of projects. The study aims to address the challenges faced by software project stakeholders in managing issue dependencies over the development life cycle and to provide concrete examples and theoretical models for the practical application of features that support the issue management tasks. The article presents three objectives for the proposed solution to improve dependency management in issue trackers:

- Users gain a better understanding of the existing issue dependency network of the issues.
- Users can search for missing dependencies and unidentified duplicate issues.
- Users can check the correct release assignments and priorities of the issue dependency network of issues and receive suggestions for resolving inconsistencies.

CHAPTER-3

RESEARCH GAPS OF EXISTING METHODS

- "**Productivity and Quality Management in the Software Development Process**" **(2006)**
    - Advantages:
        - Provides insights into the relationship between productivity and quality in software development.
        - Offers a holistic view of management practices that impact both productivity and quality.
    - Limitations:
        - Focuses on software development without specific attention to time tracking.
- **"Time-Tracking Tools for Increasing Productivity in Software Development" (2013)**
    - Advantages:
        - Directly addresses the use of time-tracking tools to enhance productivity in software development.
        - Presents real-world cases where time tracking positively influenced outcomes.
    - Limitations:
        - May lack a broader perspective on time management beyond software development.
- **"Time Management and Productivity Tools for Knowledge Workers" (2007)**
    - Advantages:
        - Explores tools applicable to knowledge workers, potentially relevant to diverse tasks.
        - Highlights the importance of time management for knowledge-intensive roles.
    - Limitations:
        - May not delve deeply into industry-specific challenges, such as software development.
- **"A Review of Time Management Literature" (2013)**

  - Advantages:
    - Provides a comprehensive overview of time management literature across various domains.
    - Offers a theoretical foundation for understanding time management practices.
  - Limitations:
    - Generalized approach; lacks specificity to software development or productivity in a corporate setting.

- **"A Study of Web Usage for Improving Web Productivity" (2009)**
  - Advantages:
    - Addresses web usage and productivity, potentially relevant for understanding internet-related activities.
    - Offers insights into optimizing web-based tasks.
  - Limitations:
    - May not provide a comprehensive analysis of overall time allocation.
- **"Analysis of Time Management and Productivity in Agile Software Development" (2009)**
  - Advantages:
    - Focuses on time management and productivity within the context of Agile software development.
    - Offers insights into the unique challenges and opportunities in Agile environments.
  - Limitations:
    - Specific to Agile; may not generalize well to other development methodologies.
- **"Web-Based Tools and Applications to Enhance Research Productivity" (2011)**
  - Advantages:
    - Discusses web-based tools applicable to enhancing productivity in a research context.
    - Offers insights into leveraging technology for improved outcomes.
  - Limitations:
    - Primarily tailored to research, potentially lacking relevance for broader organizational productivity.

CHAPTER-4
PROPOSED METHODOLOGY

The proposed system is a web-based application designed to capture and analyze the time spent by resources on various activities, such as coding, documentation, SQL etc. It offers a user-friendly interface for resource input and centralizes this data in a secure database. The system allows for customization of activity categories, providing flexibility for different organizational needs. One of the key features is its robust analytics and reporting tools that generate insights into resource productivity and time allocation patterns. Integration with organizational calendars and project management tools ensures seamless data input. Role-based access control safeguards sensitive data, while data visualization aids in interpreting productivity trends. Overall, the system aims to enhance resource productivity, optimize resource allocation, and facilitate data-driven decision-making. It offers transparency, accountability, and improved collaboration within the organization, making it an essential tool for productivity analysis.

Effective web-based time productivity analysis involves the use of various methods and tools to ensure that tasks are completed on time and within the allocated resources. Here are some proposed methods used in web-based time productivity analysis:

4.1 WORK BREAKDOWN STRUCTURE (WBS):

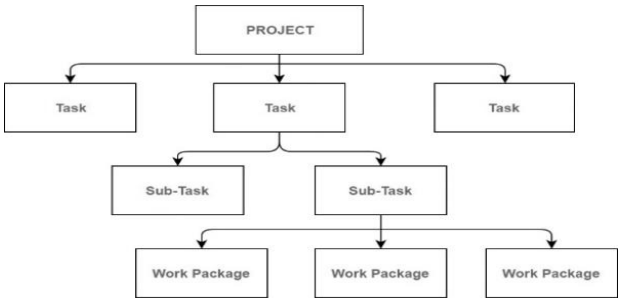


Fig 4.1 Levels of a Work Breakdown Structure

From the above figure (Fig 4.1),

The Work Breakdown Structure (WBS) is a hierarchical decomposition of the total scope of work to be carried out by the project team. It breaks down the project into smaller, more manageable components, making it easier to plan, execute, and control.

- **Task Identification:** Break down the entire project into distinct tasks and sub-tasks. This process involves identifying all the work that needs to be accomplished to complete the project.
- **Hierarchy Creation:** Organize tasks in a hierarchical structure. The highest level represents the overall project, and subsequent levels break down tasks into more detailed components.
- **Resource Allocation:** Assign resources (human, financial, and material) to each task. This step helps in understanding the resources required for each component of the project.
- **Time Estimation:** Estimate the time required to complete each task. This involves forecasting the duration for each component based on historical data, expert judgment, or other estimation techniques.
- **Dependency Identification:** Determine the dependencies between tasks. Understand which tasks are dependent on others and establish the sequence in which they need to be completed.
- **Milestone Definition:** Identify key milestones within the WBS. Milestones mark significant points in the project timeline and help in tracking progress.
- **Task Ownership and Responsibility:** Assign ownership and responsibility for each task. Clearly define who is responsible for the completion of specific components.
- **Scope Verification:** Use the WBS to verify that all project requirements are addressed. This ensures that nothing is overlooked during the planning phase.
- **Project Visualization: P**rovide a visual representation of the project structure. This visualization aids in communication and understanding among team members, stakeholders, and project managers.
- **Change Management:** Use the WBS as a reference point for change management. When changes occur, assess their impact on the WBS and adjust accordingly.
- **Monitoring and Reporting:** Use the WBS as a basis for monitoring and reporting project progress. Regularly review completed tasks and assess their impact on the overall project timeline.
- **Continuous Improvement:** Learn from the project experience and continuously improve the WBS for future projects. Lessons learned during the execution phase can inform adjustments to the WBS structure for enhanced efficiency in subsequent projects.

4.2 AGILE PROJECT MANAGEMENT:

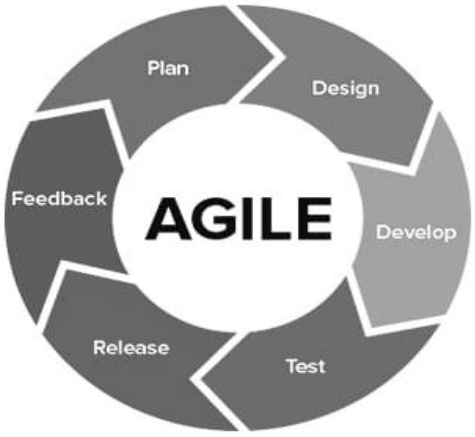


Fig 4.2 Phases of the Agile Methodology

From the above figure (Fig4.2) Agile Project Management is a widely adopted methodology for managing projects, particularly in the realm of software development. Here are some key aspects and methods associated with Agile Project Management:

- **Scrum:** Scrum is one of the most popular frameworks within Agile. It divides the project into small, iterative cycles called sprints, usually lasting 2-4 weeks.
- **Sprint Planning:** Plan tasks for the upcoming sprint.
- **Daily Stand-ups:** Brief team meetings to discuss progress and plan for the day.
- **Sprint Review:** Evaluate the completed work at the end of each sprint.
- **Retrospective:** Reflect on the sprint and identify areas for improvement.
- **Visualizing Work:** Using a Kanban board to display tasks and their statuses.
- **Work in Progress (WIP) Limits:** Setting limits on the number of tasks in progress to optimize workflow.
- **Continuous Delivery:** Encouraging a steady flow of work from start to finish.
- **Collaborative Work:** Encouraging constant communication and collaboration among team members.
- **Planning Poker:** A consensus-based technique for estimating work items.
- **Relative Sizing:** Comparing the sizes of tasks rather than providing absolute estimates.
- **Adaptive Planning Meetings:** Regularly reviewing and adapting project plans based on feedback and changes.

CHAPTER-5

OBJECTIVES

A Project Management System (PMS) is designed to help organizations plan, execute, and control projects effectively. The objectives of implementing a Project Management System include:

1. Project Planning:

- **Define Objectives**: Clearly outline project goals, scope, and objectives.
- **Resource Planning**: Allocate resources efficiently, including personnel, time, and budget.
- **Task Scheduling**: Develop a realistic and achievable timeline for project tasks.

2. Project Execution:

- **Task Assignment**: Assign responsibilities and tasks to team members based on their skills and expertise.
- **Communication**: Facilitate effective communication within the project team and stakeholders.
- **Progress Tracking**: Monitor project progress to ensure it aligns with the established schedule and milestones.

3. Resource Management:

- **Resource Allocation**: Allocate resources optimally to prevent overloading or underutilization of team members.
- **Risk Management**: Identify and mitigate potential risks that may impact the project's success.

4. Quality Management:

- **Quality Assurance**: Implement processes and standards to ensure the quality of project deliverables.
- **Testing and Validation**: Conduct thorough testing and validation of project components to meet quality standards.

5. Documentation and Reporting:

- **Document Management:** Maintain accurate and up-to-date project documentation.
- **Reporting:** Generate and distribute regular reports on project status, progress.

6. Change Management:

- **Change Control:** Implement a process for managing changes to project scope, schedule, or requirements.
- **Adaptability:** Respond effectively to changes in project requirements/external factors.

7. Closure and Evaluation:

- **Project Closure:** Ensure a smooth and orderly closure of the project, including documentation and handover.
- **Post-Project Evaluation:** Conduct a post-project review to identify lessons learned and areas for improvement in future projects.

8. Continuous Improvement:

- **Process Improvement:** Identify opportunities for process optimization and implement improvements.
- **Learning from Experience:** Apply lessons learned from past projects to enhance future project management practices.

By aligning with these objectives, a Project Management System can contribute to the successful completion of projects within scope, time, and budget constraints while meeting or exceeding quality expectations.

CHAPTER-6

SYSTEM DESIGN & IMPLEMENTATION

The design procedure for the time tracking and productivity analysis web application involves a systematic and iterative process, encompassing various stages from conceptualization to implementation.

6.1 ARCHITECTURE DIAGRAM: -

An architectural diagram is a visual representation that maps out the physical implementation for components of a software system. It shows the general structure of the software system and the associations, limitations, and boundaries between each element.

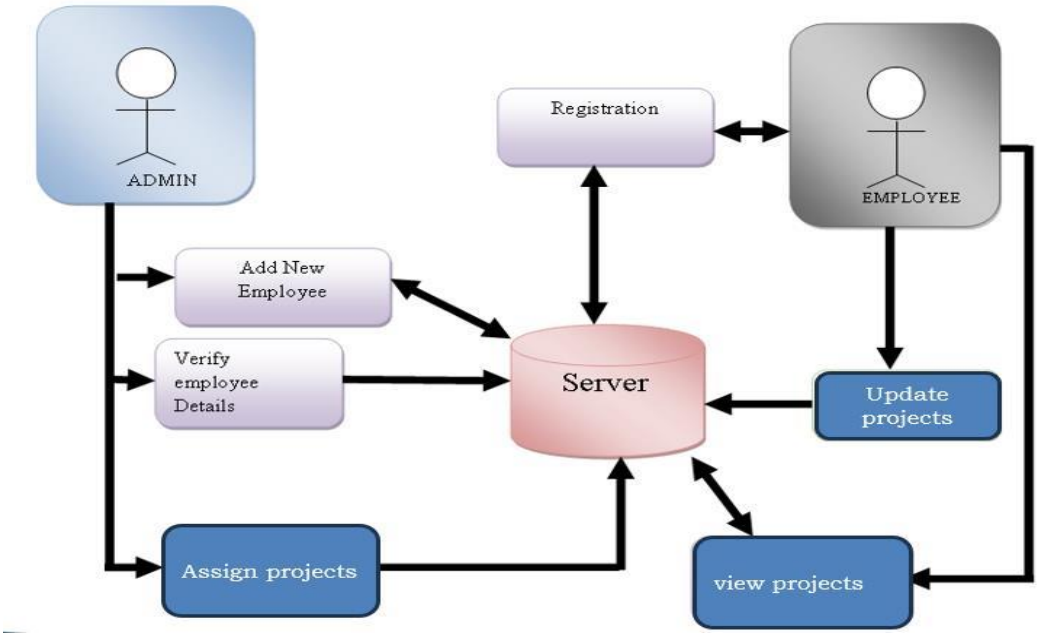


Fig 6.1 Architecture diagram

The above figure (Fig 6.1) is an example of an architecture diagram for such a system, along with explanation of each component:

- **User Interface (UI):** This is the front-end part of the system that users interact with. It includes web pages or mobile app screens where users can log in, view project details, and manage project timelines.
- **User Authentication:** Responsible for user authentication and authorization. Validates user credentials and ensures that only authorized users have access to the system.

- **Project Management Module:** This module handles the core functionalities related to project management. Includes features such as creating and editing projects, defining project timelines, and assigning tasks to team members.
- **Task Tracking:** Manages the individual tasks within a project. Allows users to create, update, and mark tasks as complete, providing real-time progress tracking.
- **Collaboration Tools:** Includes features like document sharing, comments, and discussion forums. Facilitates communication and collaboration among team members working on the same project.
- **Reporting and Analytics:** Gathers data from the system to generate reports and analytics on project progress.
  Provides insights into project timelines, resource allocation, and team performance.
- **Database:** Stores project data, user information, and other relevant information.
  Enables data retrieval and manipulation to support the various functionalities of the system.
- **Application Server:** Hosts the business logic of the application. Processes user requests, communicates with the database, and orchestrates the flow of data between different components.
- **Security Layer:** Ensures the security of the system by implementing measures such as encryption, access controls, and secure communication protocols.
- **API Gateway:** Manages and routes requests between different microservices and external APIs. Acts as a single entry point for client applications to interact with the backend services.
- **External APIs:** Integrates with third-party services or tools that complement the project time management system, such as project management tools, cloud services, or external data sources.
- **Monitoring and Logging:** Monitors the health and performance of the system. Logs events and errors for troubleshooting and auditing purposes.

This architecture diagram provides a high-level overview of the components and their interactions in a time management system. Keep in mind that the specific implementation details may vary based on the technology stack and requirements of the project.

6.2 ER DIAGRAM:

In database management systems, ER diagrams provide a graphical interface for designing databases. This simplifies the process of creating tables, defining relationships, and establishing constraints, offering an intuitive approach to dbs.

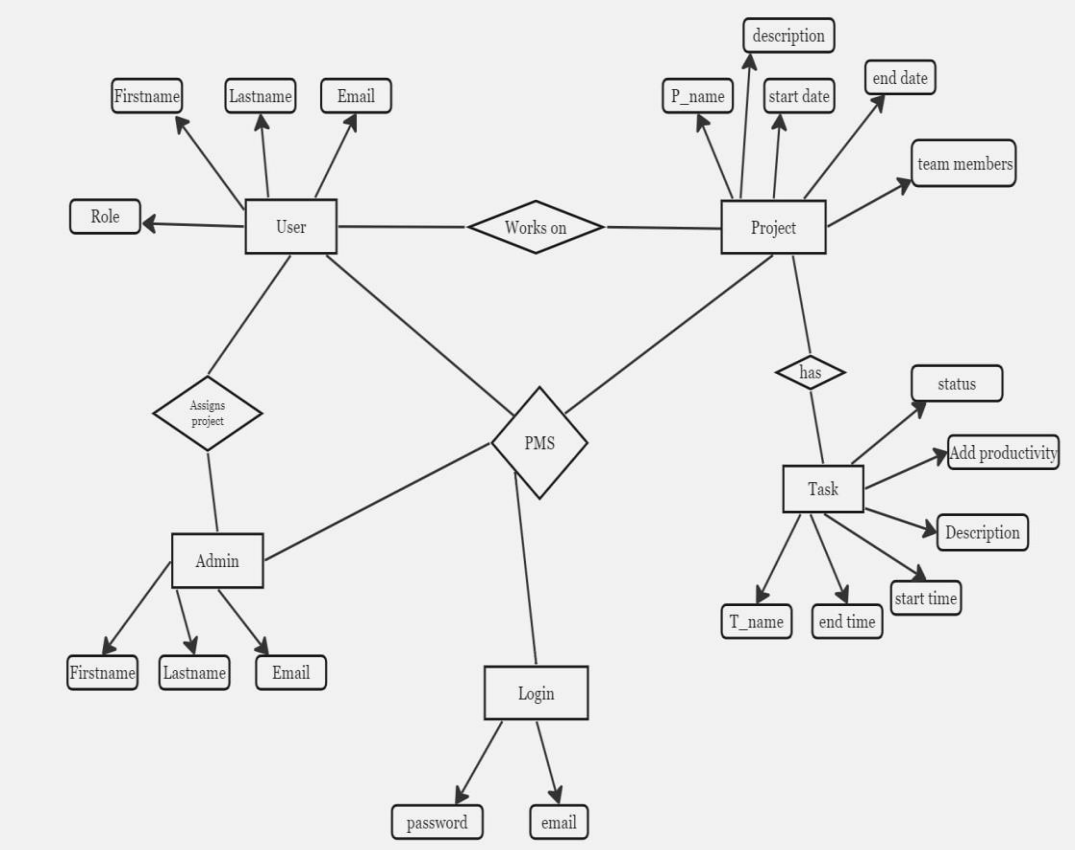


Fig 6.2 Entity Relationship Diagram

From the above figure (Fig 6.2), Creating an Entity-Relationship (ER) diagram for a time management system involves identifying and representing key entities, their attributes, and the relationships between them. Here's a basic explanation of an ER diagram for an online project time management system:

- **Entities:**
  - **User:**
    - **Attributes:** User ID (Primary Key), Username, Password, Email, FirstName, LastName, etc.
  - **Project:**
    - **Attributes:** Project ID (Primary Key), Project Name, Description, StartDate, EndDate, Status, etc.

- **Task:**
  - **Attributes:** TaskID (Primary Key), TaskName, Description, StartDate, EndDate, Status, Priority, etc.
- **Time Entry:**
  - **Attributes:** EntryID (Primary Key), UserID (Foreign Key referencing User), TaskID (Foreign Key referencing Task), StartTime, EndTime, Date, Duration, Comments, etc.

- **Relationships:**
  - **Manages:** Relates User to Project. One User can manage multiple projects, but each project is managed by one user.
  - **Participates:** Relates User to Project. One User can participate in multiple projects, and each project has multiple participants.
  - **Belongs To** Relates Task to Project. Each task belongs to one project, but a project can have multiple tasks.
  - **Logs Time:** Relates User to Task through Time Entry. Each time entry logs the time spent by a user on a specific task.

This ER diagram captures the basic structure of an online project time management system. Users can manage projects, participate in multiple projects, and log time entries for various tasks within those projects. The relationships between entities help maintain data integrity and ensure that information is organized and accessible. The attributes provide additional details for each entity, allowing for a comprehensive representation of the system. Keep in mind that the actual requirements of your system may dictate additional entities and relationships.

6.3 USE CASE DIAGRAM:

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses.

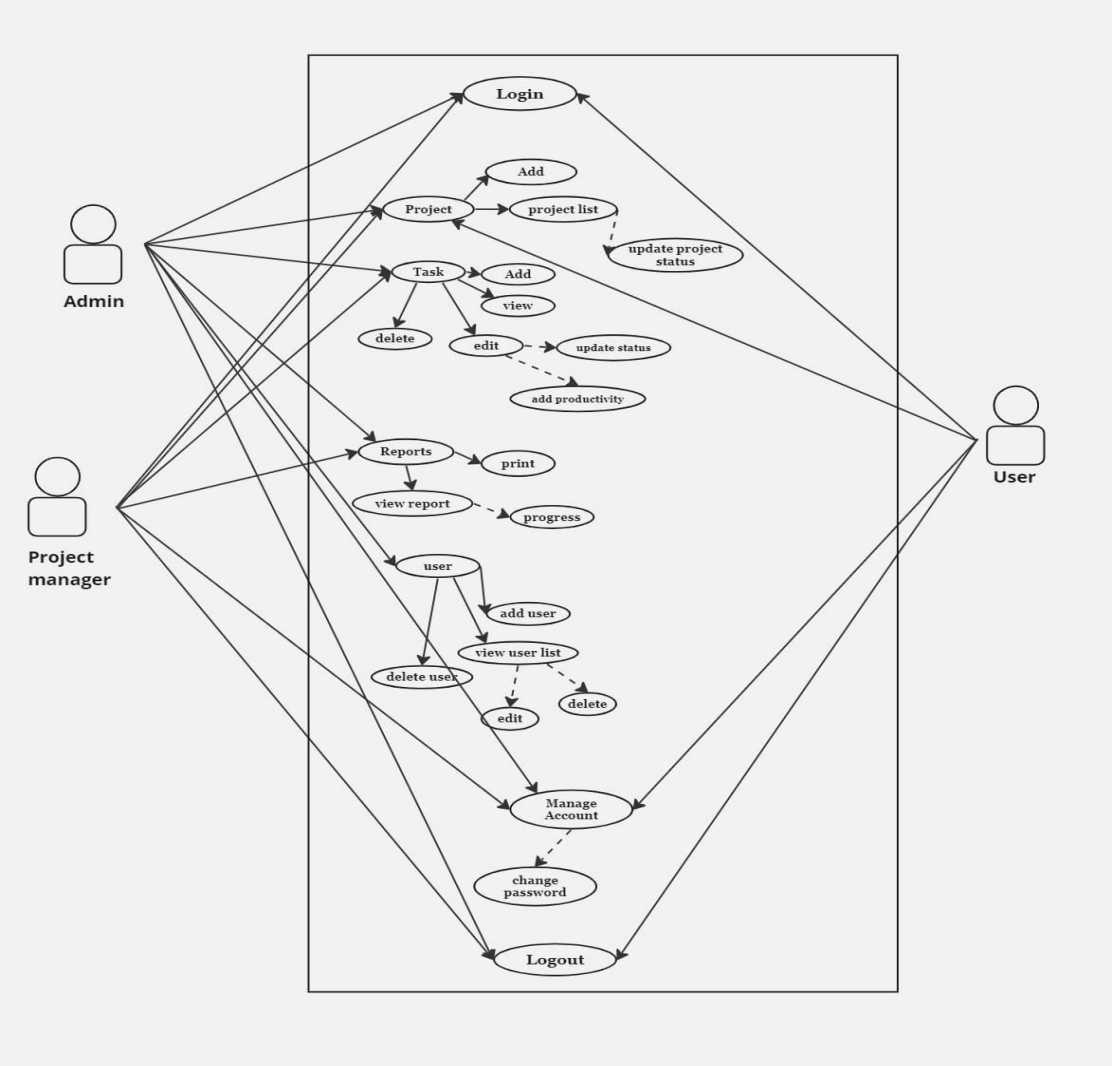


Fig 6.3 Use Case Diagram

From the above figure (Fig 6.3) use case diagram is a visual representation of the functional requirements of a system from the perspective of its users. It illustrates how users interact with a system and the various ways the system responds to those interactions. In the context of online project time management, a use case diagram could include actors, use cases, and their relationships. Here's an explanation:

- **Actors:**
    - **Project Manager:** The individual responsible for overseeing and managing the overall project.
    - **Team Member**: Any person involved in the project tasks and responsible for updating their task progress.
    - **Administrator:** Someone who has the authority to configure and manage system settings.
    - **System:** Represents the online project time management system itself.

- **Use Cases:**
    - **Login:**
        - **Actors:** Project Manager, Team Member, Administrator
        - **Description:** Users need to log in to access the online project time management system.
    - **View Project Dashboard:**
        - **Actors:** Project Manager, Team Member
        - **Description:** Users can view an overview of the project status, timelines, and key metrics on the dashboard.
    - **Create Project:**
        - **Actors:** Project Manager
        - **Description:** Project Managers can create a new project, define its parameters, and set up initial configurations.
    - **Assign Tasks:**
        - **Actors:** Project Manager
        - **Description:** Project Managers can assign tasks to team members and set deadlines for completion.
    - **Update Task Progress:**
        - **Actors:** Team Member
        - **Description:** Team Members can update the progress of their assigned tasks, mark them as complete, or indicate any delays.
    - **Generate Reports:**
        - **Actors:** Project Manager
        - **Description:** Project Managers can generate reports on project progress, time spent on tasks, and other relevant metrics.

- **Configure System Settings:**
  - **Actors:** Administrator
  - **Description:** Administrators can configure and manage system settings, user roles, and other parameters.
- **Receive Notifications:**
  - **Actors:** Project Manager, Team Member
  - **Description:** Users receive notifications about upcoming deadlines, task assignments, or other important updates.
- **Logout:**
  - **Actors:** Project Manager, Team Member, Administrator
  - **Description:** Users can log out of the system when their session is complete.

- **Relationships:**
  - **Association:** Connects actors with the relevant use cases to show who interacts with each use case.
  - **Include:** Demonstrates that one use case includes the functionality of another use case. For example, updating task progress is included in the overall task management process.
  - **Extend:** Represents optional or alternative behavior that can extend a base use case. For instance, generating advanced project reports could extend the basic report generation use case.

In summary, the use case diagram for time management helps to visualize the interactions between different actors and the system, showcasing the essential functionalities required for effective project management.

6.4 DATA FLOW DIAGRAM:

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination.

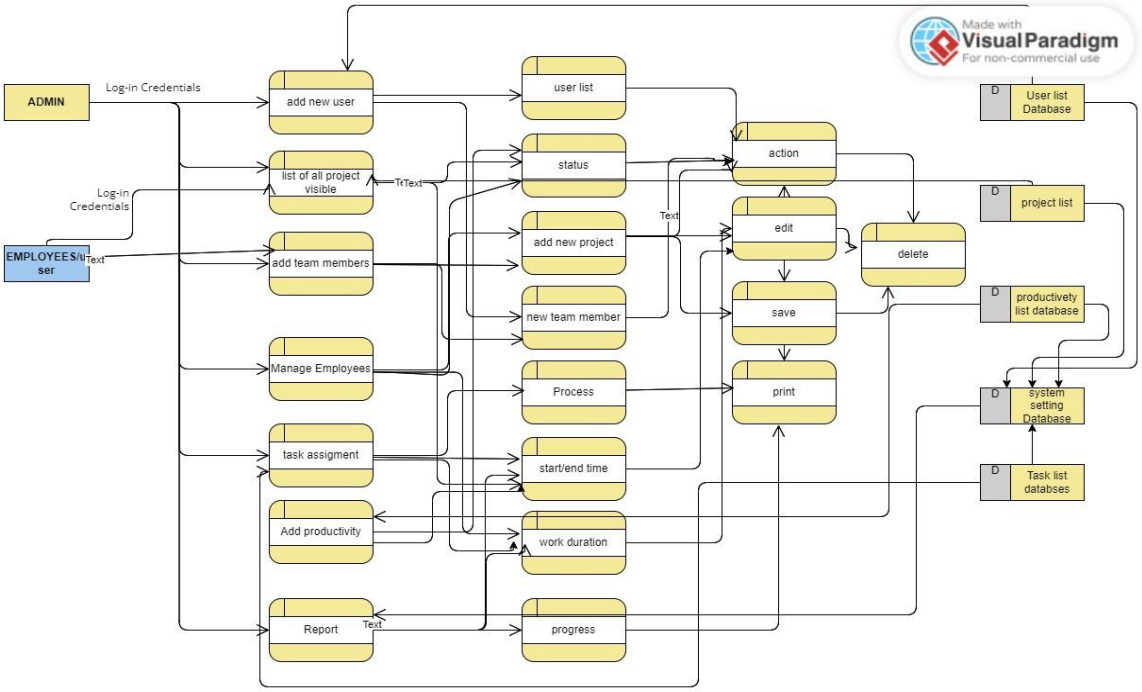


Fig 6.4 Data Flow diagram

From the above figure (Fig 6.4), the Data Flow Diagram (DFD) is a graphical representation of how data flows through a system, showcasing processes, data stores, and data movement. For an online project time management system, a DFD can help illustrate how information is processed and exchanged within the system. Here's a simplified explanation:

- **External Entities:**
  - **Project Managers/Users:** These are external entities who interact with the system. They provide input, such as project details and updates, and receive output, such as reports and notifications.
- **Processes:**
  - **Login/Authentication**: This process ensures that only authorized users can access the system. It involves validating usernames and passwords.
  - **Create Project:** Users initiate the creation of a new project by providing essential details such as project name, description, and start/end dates.
  - **Task Management:** This process involves creating, updating, and deleting tasks within a project. It manages the assignment of tasks to team members.

  - **Time Tracking:** Users log the time spent on tasks or update the system with the progress made on a task. This process may involve validation to ensure the entered data is accurate.
  - **Reporting:** This process generates various reports such as project timelines, task progress, and resource utilization. It may involve aggregating and analyzing data.
- **Data Stores:**
  - **Project Database:** This stores information about projects, tasks, users, and their relationships. It serves as a central repository for project-related data.
  - **User Profiles:** Information about users, including their roles, permissions, and login credentials, is stored in this data store.
  - **Time Logs:** Records of time spent on tasks are stored here, facilitating accurate tracking and reporting.
- **Data Flows:**
  - **Project Details:** Data flow from the 'Create Project' process to the 'Project Database,' updating it with the new project information.
  - **Task Updates:** Data flow from 'Task Management' processes to the 'Project Database,' updating task details.
  - **Time Logs:** Data flow from 'Time Tracking' processes to the 'Time Logs' data store, recording the time spent on tasks.
  - **Reports:** Processed data flows from the 'Reporting' process to external entities (Project Managers/Users), providing them with relevant project insights.
  - **Data Flow Diagram Levels:**
    - Level 0 DFD: Provides a high-level overview of the entire system, showing the main processes and external entities.
    - Level 1 DFDs: Break down each process into sub-processes, offering a more detailed view.
    - Level 2 DFDs: Further details specific sub-processes, making it even more granular.

By creating a Data Flow Diagram for a project time management system, you can visualize the flow of information, making it easier to understand, analyze, and improve the efficiency of the system.

6.5 CLASS DIAGRAM:

A class diagram is one type of UML diagram that represents the static view of a software system. A class diagram is used to visualize, describe, and document various aspects of a system, and also to create executable software code. It illustrates the types of objects present in a system and the different types of relationships that exist between them.

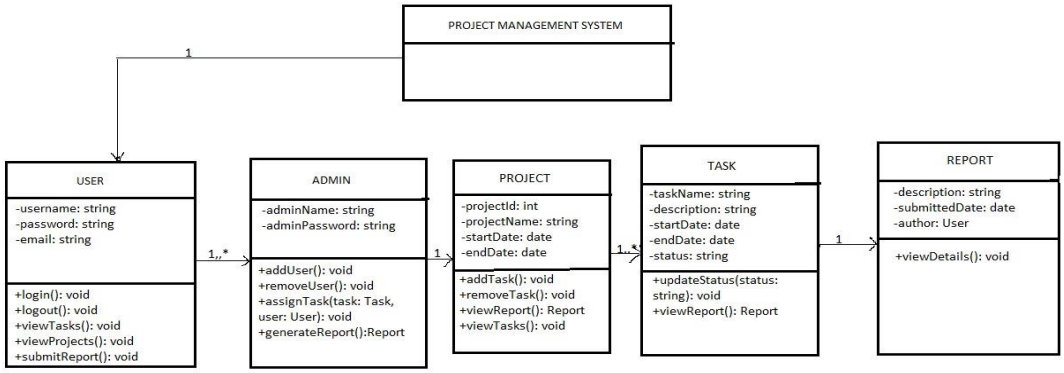


Fig 6.5 UML Class diagram

From the above figure (Fig 6.5) class diagram is a type of structural diagram that illustrates the structure of a system by showing the classes, their attributes, methods, and relationships. For an online project time management system, the class diagram can help depict the key entities and their interactions. Here's an explanation for a simplified class diagram:

- **User Class:**
  - **Attributes:** userID, username, password, email, role
  - **Methods:** login(), logout(), view projects(), manage tasks()
- **Project Class:**
  - **Attributes:** projectID, projectName, start Date, end Date, projectManager
  - **Methods:** createProject(), updateProjectDetails(), viewProjectDetails()
- **Task Class:**
  - **Attributes:** taskID, taskName, description, startDate, endDate, assignedTo, status
  - **Methods:**createTask(), updateTaskDetails(), assignTask(), completeTask()
- **Team Member Class:**
  - **Attributes:** memberID, memberName, email, role
  - **Methods:** viewAssignedTasks(), updateProfile()
- **TimeEntry Class:**

- **Attributes:** entryID, taskID, memberID, date, hoursWorked
- **Methods:** logTime(), viewTimeSheet()

- **Report Class:**
  - **Attributes:** reportID, projectID, startDate, endDate, generatedBy
  - **Methods:** generateProjectReport(), viewReports()
- **Role Class:**
  - **Attributes:** roleID, roleName
  - **Methods:** definePermissions(), assignRole()
- **Relationships**
  - Users can be associated with Projects (project manager, team members).
  - Projects have Tasks (tasks are assigned to team members).
  - Team Members log Time Entries for Tasks.
  - Notifications are sent to Users.
  - Reports are associated with Projects and generated by Users.
  - Roles have Permissions.
  - Scheduler is associated with Tasks.

This class diagram provides an overview of the main entities in an online project time management system and their relationships. It emphasizes user management, project and task tracking, time logging, notifications, and reporting features. Depending on the complexity of your system, you might need to expand or modify this diagram to meet specific requirements.

CHAPTER-7

TIMELINE FOR EXECUTION OF PROJECT

(GANTT CHART)

TASK	Duration	START	END
Requirement Analysis	2 weeks	10/9/23	24/9/2023
System Architecture Design	3 weeks	24/9/2023	14/10/2023
database design	2 weeks	14/10/2023	29/10/2023
backend development	4 weeks	29/10/2023	11/11/23
Frontend development	3 weeks	11/11/2023	3/12/2023
Security information	0.5 weeks	3/12/2023	9/12/2023
Testing	2 weeks	9/12/2023	19/12/2023
Documentation	1 weeks	19/12/2023	26/12/2023
Deployment	2 weeks	27/12/2023	7/1/2024

Table 7.1

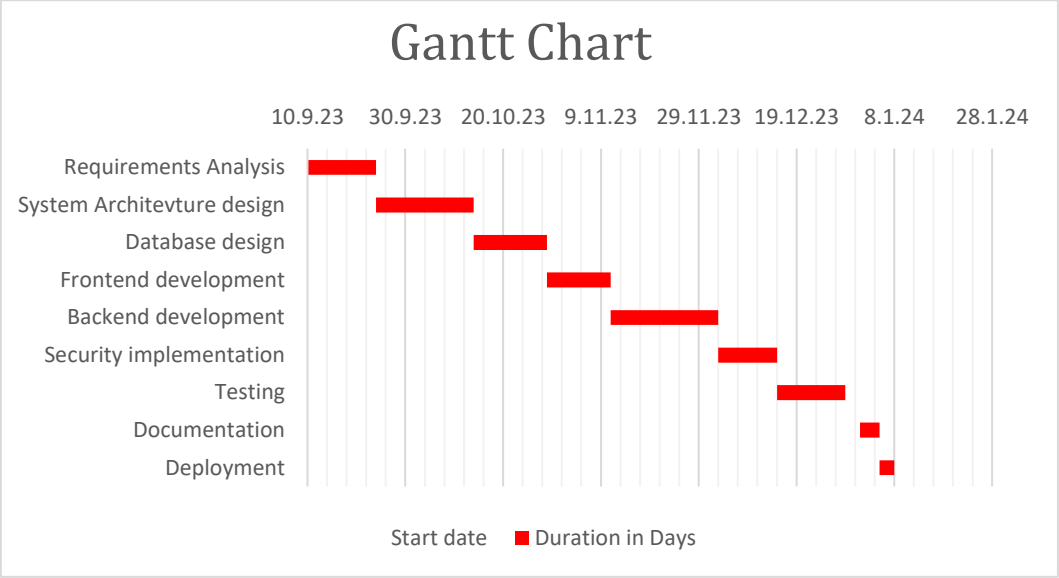
As shown in the table (Table 7.1),

Creating a timeline for the execution of a project depends on various factors, such as the complexity of the project, the resources available, and any specific milestones or deadlines. Here's a general outline for creating a project timeline:

- **Project Initiation (Week 1-2):**
  - Define project objectives, scope, and deliverables.
  - Identify key stakeholders and project team members.
  - Develop a project charter or initiation document.
- **Planning (Week 3-4):**
  - Create a detailed project plan outlining tasks, dependencies, and timelines.
  - Allocate resources and define roles and responsibilities.
  - Develop risk management and mitigation strategies.
  - Obtain necessary approvals for the project plan.
- **Execution (Week 5-10):**
  - Begin the actual work outlined in the project plan.
  - Hold regular team meetings and monitor progress.
  - Implement the necessary project management tools for tracking tasks and milestones.
  - Address and resolve any issues or roadblocks promptly.
  - Keep stakeholders informed of progress.

- **Monitoring and Controlling (Throughout the Project):**
    - Regularly review project status against the plan.
    - Track and manage changes to the project scope.
    - Monitor and control risks.
    - Ensure quality control and address any deviations from the plan.
- **Testing and Quality Assurance (Week 11-12):**
    - Conduct necessary testing procedures.
    - Verify that project deliverables meet the specified quality standards.
    - Make any required adjustments based on testing outcomes.
- **Deployment (Week 13-14):**
    - Implement the project deliverables.
    - Train end-users if necessary.
    - Monitor the system for any issues during the initial deployment.
- **Closure (Week 15):**
    - Conduct a project review to evaluate its success and identify areas for improvement.
    - Document lessons learned.
    - Obtain formal acceptance from stakeholders.
    - Close out the project, including any administrative tasks and finalizing documentation.

Keep in mind that this is a general guideline, and the timeline may need to be adjusted based on the specific requirements and constraints of your project. Additionally, it's crucial to regularly communicate with the project team and stakeholders, making adjustments to the timeline as needed.



Gantt Chart

A Gantt chart is a popular project management tool that provides a visual representation of a project schedule. It is particularly effective for illustrating the timeline of tasks, dependencies between tasks, and the overall progress of a project. Here's an explanation of key elements and how to read a Gantt chart:

- **Task List:** The left side of the chart lists all the tasks or activities that need to be completed for the project. Each task is represented by a separate row.
- **Timeline:** The top of the chart represents the project timeline, usually in days, weeks, or months, depending on the project's scale. It's the horizontal axis.
- **Task Bars:** Each task is represented by a horizontal bar that spans across the timeline. The length of the bar corresponds to the duration of the task.
- **Dependencies:** Arrows between tasks indicate dependencies, showing the order in which tasks must be completed. For example, Task B can only start when Task A is finished.
- **Milestones:** Key project milestones, which are significant points in the project timeline, are often marked on the Gantt chart. They could be the completion of a phase, client reviews, or important deadlines.
- **Progress Tracking:** As the project progresses, the Gantt chart can be updated to reflect the actual start and end dates of tasks. This helps in tracking whether the project is on schedule.
- **Critical Path:** The critical path is the sequence of tasks that determines the project's earliest completion date. It's often highlighted on the Gantt chart to emphasize tasks that, if delayed, could impact the project timeline.

CHAPTER-8

OUTCOMES

Project Management Systems (PMS) play a crucial role in ensuring successful planning, execution, and completion of projects. The objectives of a Project Management System are typically aimed at improving efficiency, collaboration, and overall project success. Here are some common objectives and the corresponding outcomes:

- **Efficient Resource Allocation:**
  - Outcome: Proper allocation of resources ensures that tasks are completed on time and within budget. It minimizes resource wastage and enhance overall project efficiency
- **Document and Information Management:**
  - Outcome: Efficient document and information management streamline access to project-related data. This reduces the chances of data loss, improves accountability, and facilitates knowledge sharing among team members.
- **Improved Decision Making:**
  - Outcome: Access to real-time project data and analytics enables better decision-making. Project managers can make informed choices, leading to more effective problem-solving and strategic planning.
- **Continuous Improvement:**
  - Outcome: Regular evaluation of project performance and feedback leads to continuous improvement. Lessons learned from one project can be applied to future projects, increasing overall organizational efficiency.
- **Accurate Planning and Scheduling**:
  - Outcome: Detailed planning and scheduling result in a realistic timeline for project completion. It helps identify potential bottlenecks and allows for timely adjustments to keep the project on track.
- **Risk Management:**
  - Outcome: A proactive approach to identifying, assessing, and mitigating risks helps in minimizing the impact of unforeseen events. This leads to a more resilient project that can adapt to changes without significant disruptions.

- **Foster Effective Communication**
    - Outcome: Improved communication leads to better collaboration among team members, stakeholders, and project managers, reducing the likelihood of misunderstandings and enhancing overall project efficiency.
- **Monitor and Evaluate Progress**
    - Outcome: Continuous monitoring allows for real-time assessment of project progress, making it easier to identify and address any deviations from the plan promptly.

In summary, the implementation of a Project Management System with these objectives results in streamlined processes, improved collaboration, reduced risks, and ultimately, successful project delivery.

CHAPTER-9

RESULTS AND DISCUSSIONS

9.1 PROBLEM DISCUSSION:

Offline time and productivity analysis methods come with their own set of challenges:

- **Delayed Reporting:** Offline methods often involve manual data collection, leading to delays in generating productivity reports. This lag can hinder timely decision-making and responsiveness to emerging issues.
- **Limited Accessibility**: Accessing productivity data offline can be restrictive, especially for remote teams or employees working from different locations. This limitation may hinder collaboration and real-time adjustments.
- **Error-Prone Manual Processes:** Reliance on manual data entry increases the likelihood of errors, impacting the accuracy of productivity analysis. Inconsistencies in data can lead to flawed insights and decision-making.
- **Inefficiency in Data Aggregation:** Compiling and aggregating data from various sources manually can be time-consuming and prone to oversights. This inefficiency may hinder a comprehensive understanding of overall productivity.
- **Lack of Real-Time Monitoring:** Offline methods lack the ability to provide real-time monitoring of tasks and projects. This absence of immediacy can result in missed opportunities to address issues promptly.
- **Difficulty in Scaling**: As organizations grow, the manual effort required for offline analysis becomes increasingly challenging to scale. This can lead to a lack of adaptability to changing workloads and complexities.
- **Limited Collaboration**: Offline analysis may impede effective collaboration, as team members may not have simultaneous access to productivity data. This can hinder collective problem-solving and hinder teamwork.
- **Data Security Risks:** Storing productivity data offline may pose security risks, especially if physical records are not adequately protected. Loss, theft, or damage to records could compromise sensitive information.

9.2 RESULT/SOLUTION:

An online platform for productivity analysis and time management offers several advantages:

- **Real-time Data Access:** Online platforms provide instant access to real-time data, enabling users to monitor productivity metrics and time usage promptly.
- **Remote Accessibility:** With the rise of remote work, online platforms facilitate tracking and management of productivity regardless of geographical location, ensuring consistent monitoring.
- **Automation and Integration**: Integrating with other tools and automating certain processes enhances efficiency, reducing manual effort in data collection and analysis
- **Customization and Scalability**: Online platforms often offer customizable features to adapt to the unique needs of different organizations, ensuring scalability as businesses grow.
- **Data Security and Backup**: Reliable online platforms prioritize data security, implementing measures such as encryption and regular backups to safeguard sensitive information.
- **Reporting and Insights:** These platforms often provide detailed reports and insights, helping organizations identify trends, assess performance, and make informed decisions to improve productivity.

In summary, an online platform for productivity analysis and time management offers flexibility, collaboration, and automation, making it a valuable tool for modern organizations striving to optimize their workflows.

CHAPTER-10

CONCLUSION

In conclusion, web-based time and productivity analysis tools have become invaluable assets in today's dynamic work environments. These tools offer a range of features and functionalities designed to empower individuals and teams to better understand, manage, and optimize their use of time. The ability to monitor digital activities, track tasks, and analyze productivity trends provides a wealth of insights that contribute to enhanced efficiency and effectiveness. Web-based time and productivity analysis tools not only assist in identifying time sinks and bottlenecks but also foster a culture of accountability and continuous improvement. The real-time data and visualizations offered by these tools enable informed decision-making, allowing individuals and organizations to adapt swiftly to changing priorities and demands.

By cultivating a harmonious relationship with time through these tools, users can transform the abstract concept of time into a tangible and strategic resource. Whether through the use of sophisticated analytics or intuitive interfaces, these tools aid in the cultivation of intentional choices, purposeful work, and meaningful achievements. They act as digital companions on the journey of productivity, offering actionable insights that propel individuals toward a future defined by their deliberate decisions and accomplishments.

As the workplace continues to evolve, the importance of web-based time and productivity analysis tools is likely to grow, providing individuals and teams with the means to navigate the complexities of modern work life while fostering a proactive and empowered approach to time management.

REFERENCES

[1]. Advance Recommendation System for the Formation of More Prolific and Dynamic Software Project Teams, Mahreen Ahmad,2018. R. Latone and J. Suarez, "Measuring social networks when forming information system project," The Journal of Systems and Software, pp. 304-323, 2017."A framework for freelancer assessment in online marketplace," in ICSE-SEIP '17 Proceedings of the 39th International Conference on Software Engineering: Software Engineering in Practice Track, Buenos Aires, Argentina, 2017.

[2]. L. Niamba, Geographical and gender disparities in the registration of births, marriages, and deaths in the Nouna Health and Demographic Surveillance System, Burkina Faso, CRVS Working Paper Series, Issue 1, (2020).

[3]. F. Azizi, N. Salari, A novel condition-based maintenance framework for parallel manufacturing systems based on bivariate birth/birth–death processes, Reliability Engineering & System Safety 229 (2023) 108798.

[4]. S. Dharwad Ker, S. Mills, Options for digital birth certificates, World Bank2019.

[5]. B. Manby, The Sustainable Development Goals and 'legal identity for all': 'First, not harm', World Development 139 (2021) 105343.

[6]. A.M. Kante, A. Mulungo, M. Ibraimo, A. Akum, N. Titus, A. Adriano, F. Van Dyk, I. Macicame, R.E. Black, A. Amouzou, Completeness and Factors Affecting Community Workers' Reporting of Births and Deaths in the Countrywide Mortality Surveillance for Action in Mozambique, The American journal of tropical medicine and hygiene 108(5 Suppl) (2023) 29.

[7]. C. Park, D. Moon, N. Do, and S. Bae, "A predictive maintenance

[8]. approach based on real-time internal parameter monitoring," The International Journal of Advanced Manufacturing Technology, vol. 85, no pp. 623–632, 2016.

[9]. S. Zhang, R. Dubay, and M. Charest, "A principal component analysis

[10]. model-based predictive controller for controlling part warpage in plastic

[11]. injection molding," Expert Systems with Applications, vol. 42, no. 6, pp. 2919–2927, 2015.

[12]. S. Ekwaro-Osire, A. Gonalves, and F. Alemayehu, Probabilistic Prognostics and Health Management of Energy Systems. New York: Springer, 2017.

[13]. P. Puschncr and C. Koza, “Calculating the Maximum Execution Trinco of Real-Time Programs. Real2. ICB. Kenny and IC-J. Lin, “Building Flexible Real-Time Systems Using the Flex Language,” Compacted;

[14]. Advance Recommendation System for the Formation of More Prolific and Dynamic Software Project Teams, Mahreen Ahmad,2018.

[15]. R. Latone and J. Suarez, "Measuring social networks when forming information system project," The Journal of Systems and Software, pp. 304-323, 2017.

[16]. "A framework for freelancer assessment in online marketplace," in ICSE-SEIP '17 Proceedings of the 39th International Conference on Software Engineering: Software Engineering in Practice Track, Buenos Aires, Argentina, 2017.

[17]. B. M. M. Q. R. U. Q. M. A. Fateh urn Rehman, "Scrum Software Maintenance Model: Efficient Software Maintenance in Agile Methodology," in 2018 21st Saudi Computer Society National Computer Conference (NCC), Riyadh, Saudi Arabia, 2018.

[18]. S. B. S. S. Apoorva Srivastava, "SCRUM Model for Agile Methodology," in International Conference on Computing, 2017.

[19]. M. J. (MJ), "Scrum Reference Card," [Online]. [Accessed December 2018].

APPENDIX-A

PSUEDOCODE

```
<!DOCTYPE html>
<html lang="en">
<?php session_start() ?>
<?php
  if(!isset($_SESSION['login_id']))
      header('location:login.php');
    include 'db_connect.php';
    ob_start();
  if(!isset($_SESSION['system'])){

    $system = $conn->query("SELECT * FROM system_settings")->fetch_array();
    foreach($system as $k => $v){
      $_SESSION['system'][$k] = $v;
    }
  }
  ob_end_flush();

  include 'header.php'
?>
<body class="hold-transition sidebar-mini layout-fixed layout-navbar-fixed
layout-footer-fixed">
<div class="wrapper">
  <?php include 'topbar.php' ?>
  <?php include 'sidebar.php' ?>

  <!-- Content Wrapper. Contains page content -->
  <div class="content-wrapper">
     <div class="toast" id="alert_toast" role="alert" aria-live="assertive"
aria-atomic="true">
      <div class="toast-body text-white">
      </div>
    </div>
    <div id="toastsContainerTopRight" class="toasts-top-right fixed"></div>
    <!-- Content Header (Page header) -->
    <div class="content-header">
      <div class="container-fluid">
        <div class="row mb-2">
          <div class="col-sm-6">
            <h1 class="m-0"><?php echo $title ?></h1>
          </div><!-- /.col -->

        </div><!-- /.row -->
           <hr class="border-primary">
```

```
      </div><!-- /.container-fluid -->
    </div>
    <!-- /.content-header -->

    <!-- Main content -->
    <section class="content">
      <div class="container-fluid">
         <?php
            $page = isset($_GET['page']) ? $_GET['page'] : 'home';
            if(!file_exists($page.".php")){
                include '404.html';
            }else{
            include $page.'.php';

            }
          ?>
      </div><!--/. container-fluid -->
    </section>
    <!-- /.content -->
    <div class="modal fade" id="confirm_modal" role='dialog'>
    <div class="modal-dialog modal-md" role="document">
      <div class="modal-content">
        <div class="modal-header">
        <h5 class="modal-title">Confirmation</h5>
      </div>
      <div class="modal-body">
        <div id="delete_content"></div>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-primary" id='confirm'
onclick="">Continue</button>
        <button type="button" class="btn btn-secondary" data-
dismiss="modal">Close</button>
      </div>
      </div>
    </div>
  </div>
  <div class="modal fade" id="uni_modal" role='dialog'>
    <div class="modal-dialog modal-md" role="document">
      <div class="modal-content">
        <div class="modal-header">
        <h5 class="modal-title"></h5>
      </div>
      <div class="modal-body">
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-primary" id='submit'
onclick="$('#uni_modal form').submit()">Save</button>
```

```
        <button type="button" class="btn btn-secondary" data-
dismiss="modal">Cancel</button>
      </div>
      </div>
    </div>
  </div>
  <div class="modal fade" id="uni_modal_right" role='dialog'>
    <div class="modal-dialog modal-full-height  modal-md" role="document">
      <div class="modal-content">
        <div class="modal-header">
        <h5 class="modal-title"></h5>
        <button type="button" class="close" data-dismiss="modal" aria-
label="Close">
          <span class="fa fa-arrow-right"></span>
        </button>
      </div>
      <div class="modal-body">
      </div>
      </div>
    </div>
  </div>
  <div class="modal fade" id="viewer_modal" role='dialog'>
    <div class="modal-dialog modal-md" role="document">
      <div class="modal-content">
              <button type="button" class="btn-close" data-
dismiss="modal"><span class="fa fa-times"></span></button>
              <img src="" alt="">
      </div>
    </div>
  </div>
  </div>
  <!-- /.content-wrapper -->

  <!-- Control Sidebar -->
  <aside class="control-sidebar control-sidebar-dark">
    <!-- Control sidebar content goes here -->
  </aside>
  <!-- /.control-sidebar -->

  <!-- Main Footer -->
  <footer class="main-footer">
    <strong>Copyright &copy; 2021 <a
href="https://www.campcodes.com/">CampCodes</a>.</strong>
    All rights reserved.
    <div class="float-right d-none d-sm-inline-block">
      <b><?php echo $_SESSION['system']['name'] ?></b>
    </div>
  </footer>
```

```
</div>
<!-- ./wrapper -->

<!-- REQUIRED SCRIPTS -->
<!-- jQuery -->
<!-- Bootstrap -->
<?php include 'footer.php' ?>
</body>
```

Loginpage.php

```
<?php include('db_connect.php') ?>
<?php
$twhere ="";
if($_SESSION['login_type'] != 1)
  $twhere = "  ";
?>
<!-- Info boxes -->
 <div class="col-12">
        <div class="card">
          <div class="card-body">
            Welcome <?php echo $_SESSION['login_name'] ?>!
          </div>
        </div>
  </div>
  <hr>
  <?php

    $where = "";
    if($_SESSION['login_type'] == 2){
      $where = " where manager_id = '{$_SESSION['login_id']}' ";
    }elseif($_SESSION['login_type'] == 3){
      $where = " where concat('[',REPLACE(user_ids,',','],['),']') LIKE
'%[{$_SESSION['login_id']}]%' ";
    }
```

```
    $where2 = "";
   if($_SESSION['login_type'] == 2){
     $where2 = " where p.manager_id = '{$_SESSION['login_id']}' ";
   }elseif($_SESSION['login_type'] == 3){
     $where2 = " where concat('[',REPLACE(p.user_ids,',','],['),']') LIKE
'%[{$_SESSION['login_id']}]%' ";
   }
?>
     <div class="row">
       <div class="col-md-8">
       <div class="card card-outline card-success">
         <div class="card-header">
           <b>Project Progress</b>
         </div>
         <div class="card-body p-0">
           <div class="table-responsive">
             <table class="table m-0 table-hover">
               <colgroup>
                 <col width="5%">
                 <col width="30%">
                 <col width="35%">
                 <col width="15%">
                 <col width="15%">
               </colgroup>
<thead>
                 <th>#</th>
                 <th>Project</th>
                 <th>Progress</th>
                 <th>Status</th>
                 <th></th>
               </thead>
               <tbody>
               <?php
               $i = 1;
```

```
$stat = array("Pending","Started","On-Progress","On-Hold","Over Due","Done");
$where = "";
if($_SESSION['login_type'] == 2){
 $where = " where manager_id = '{$_SESSION['login_id']}' ";
}elseif($_SESSION['login_type'] == 3){
 $where = " where concat('[',REPLACE(user_ids,',','],['),']') LIKE '%[{$_SESSION['login_id']}]%' ";
}
$qry = $conn->query("SELECT * FROM project_list $where order by name asc");
while($row= $qry->fetch_assoc()):
 $prog= 0;
$tprog = $conn->query("SELECT * FROM task_list $where order by name asc");
while($row= $qry->fetch_assoc()):
 $prog= 0;
$tprog = $conn->query("SELECT * FROM task_list where project_id = {$row['id']}")->num_rows;
$cprog = $conn->query("SELECT * FROM task_list where project_id = {$row['id']} and status = 3")->num_rows;
$prog = $tprog > 0 ? ($cprog/$tprog) * 100 : 0;
$prog = $prog > 0 ?  number_format($prog,2) : $prog;
$prod = $conn->query("SELECT * FROM user_productivity where project_id = {$row['id']}")->num_rows;
if($row['status'] == 0 && strtotime(date('Y-m-d')) >= strtotime($row['start_date'])):
if($prod  > 0  || $cprog > 0)
 $row['status'] = 2;
else
 $row['status'] = 1;
elseif($row['status'] == 0 && strtotime(date('Y-m-d')) > strtotime($row['end_date'])): $row['status'] = 4;
endif;
```

```
            ?>
            <tr>
              <td>
                <?php echo $i++ ?>
              </td>
              <td>
                <a>
                  <?php echo ucwords($row['name']) ?>
                </a>
                <br>
                <small>
                  Due: <?php echo date("Y-m-d",strtotime($row['end_date'])) ?>
                </small>
              </td>
              <td class="project_progress">
                <div class="progress progress-sm">
                  <div class="progress-bar bg-green" role="progressbar" aria-valuenow="57" aria-valuemin="0" aria-valuemax="100" style="width: <?php echo $prog ?>%">
                  </div>
                </div>
                <small>
                  <?php echo $prog ?>% Complete
                </small> </td>
              <td class="project-state">
                <?php
                  if($stat[$row['status']] =='Pending'){
                    echo "<span class='badge badge-secondary'>{$stat[$row['status']]}</span>";
                  }elseif($stat[$row['status']] =='Started'){
                    echo "<span class='badge badge-primary'>{$stat[$row['status']]}</span>";
                  }elseif($stat[$row['status']] =='On-Progress'){
                    echo "<span class='badge badge-
```

```
info'>{$stat[$row['status']]}</span>";
                    }elseif($stat[$row['status']] =='On-Hold'){
                      echo "<span class='badge badge-
warning'>{$stat[$row['status']]}</span>";
                    }elseif($stat[$row['status']] =='Over Due'){
                      echo "<span class='badge badge-
danger'>{$stat[$row['status']]}</span>";
                    }elseif($stat[$row['status']] =='Done'){
                      echo "<span class='badge badge-
success'>{$stat[$row['status']]}</span>";
                    }
                   ?>
                  </td> <td>
                   <a class="btn btn-primary btn-sm"
href="./index.php?page=view_project&id=<?php echo $row['id'] ?>">
                      <i class="fas fa-folder">
                      </i>
                      View
                   </a>
                  </td>
                </tr>
               <?php endwhile; ?>
               </tbody>
              </table>
             </div>
            </div>
           </div>
           </div>
```

APPENDIX-B

SCREENSHOTS

LOGIN PAGE

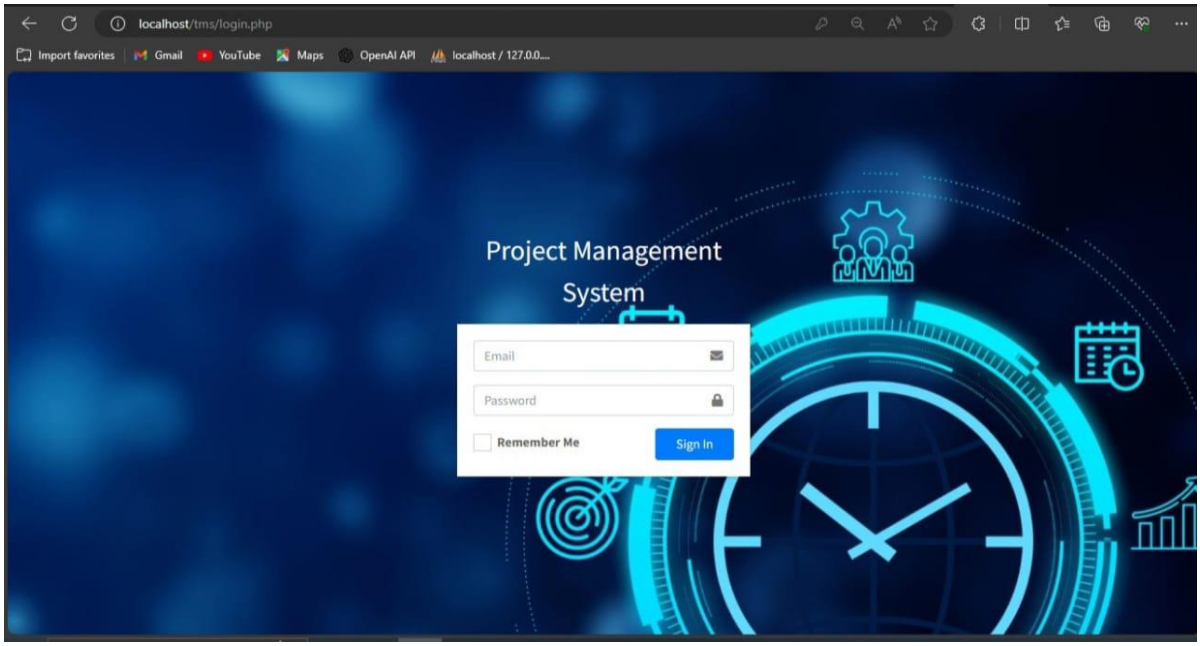


Fig.1 Login page

DASHBOARD(ADMIN-SIDE)

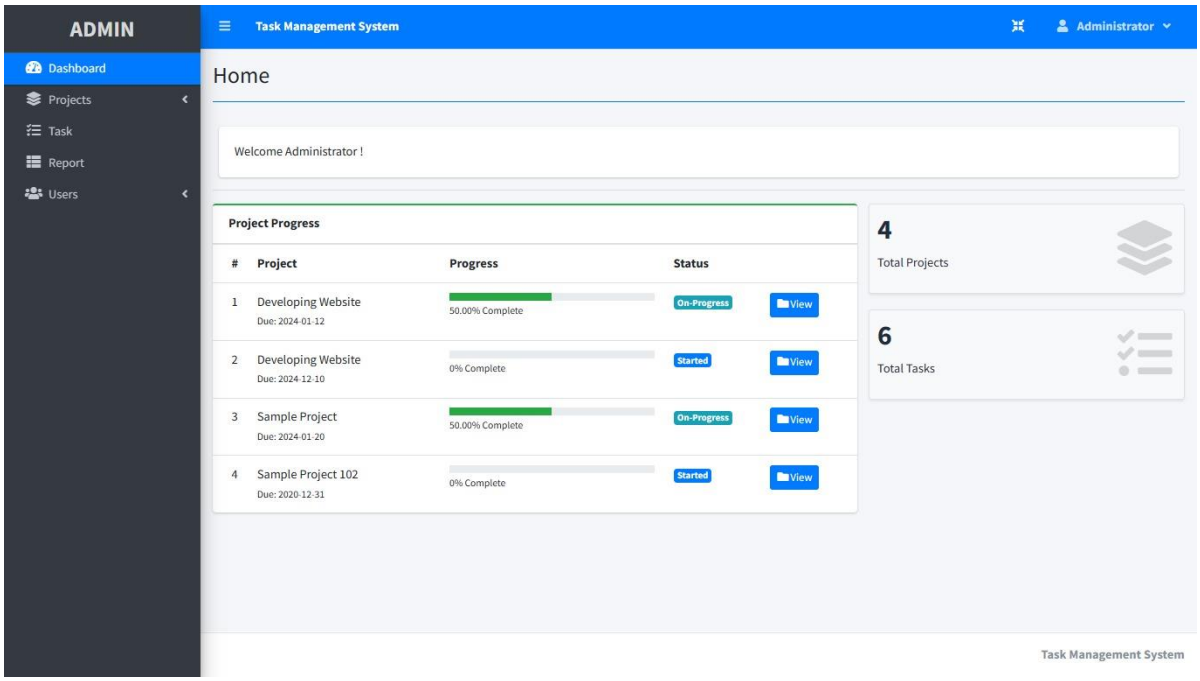


Fig.2 Admin dash board

ADD NEW USER

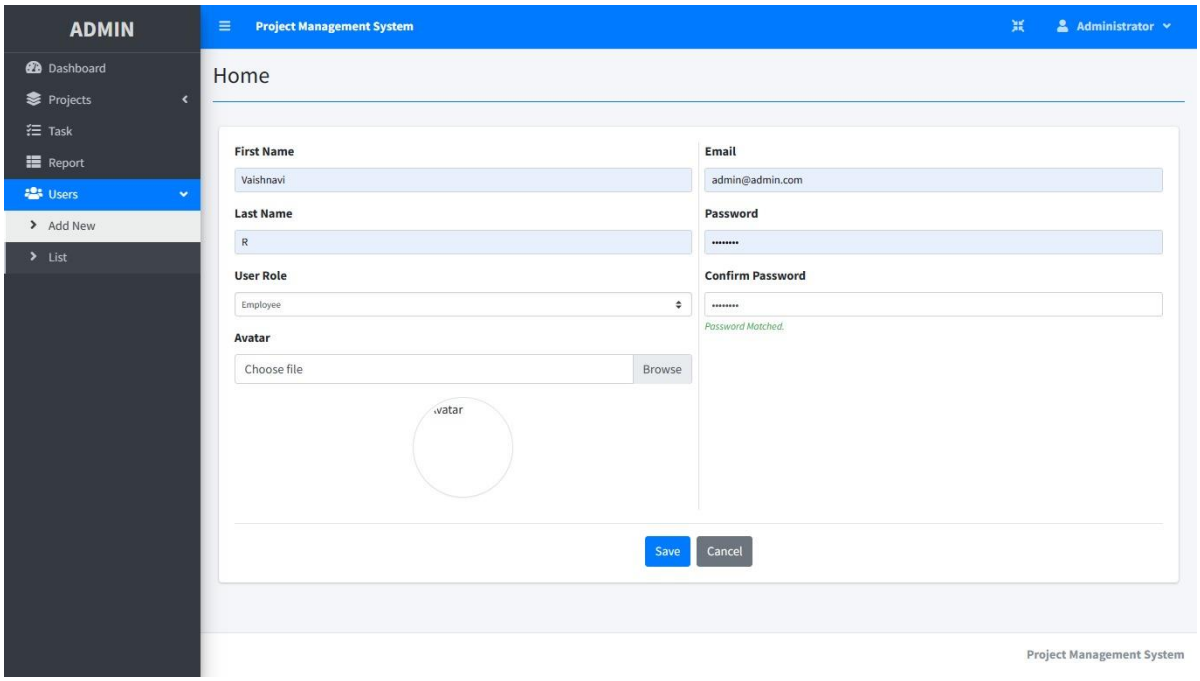


Fig.3 Admin adding new employee

LIST OF USERS

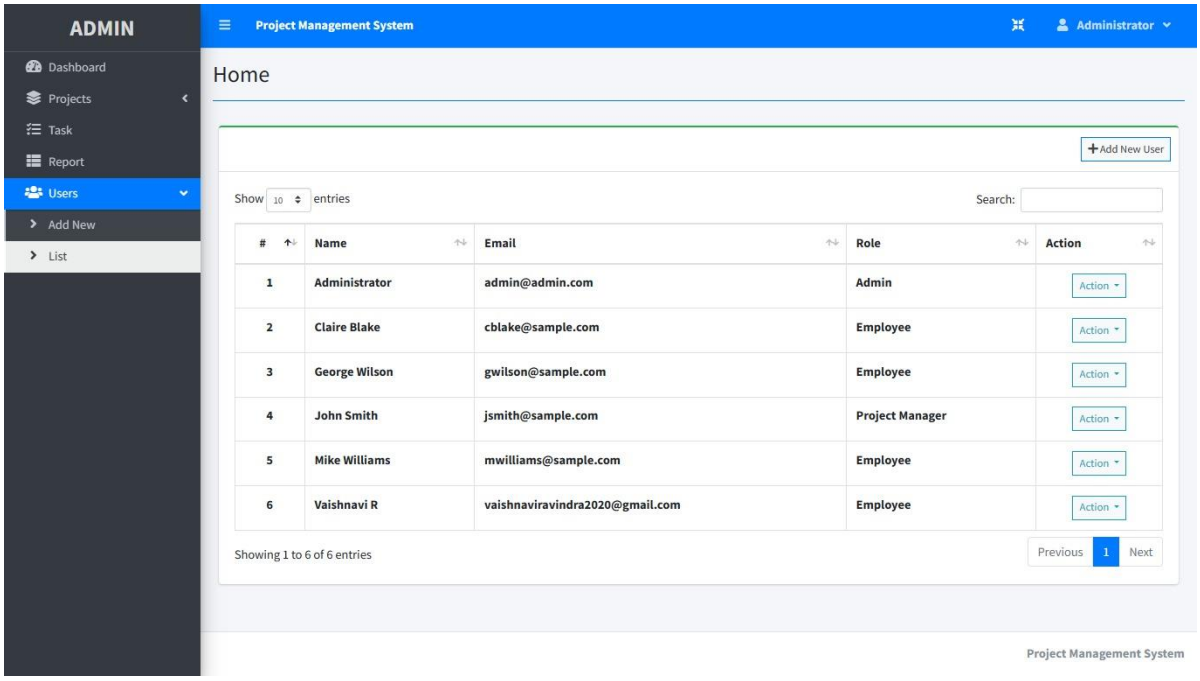


Fig.4 List of employees in company (stored in admin dashboard)

ADD NEW PROJECT

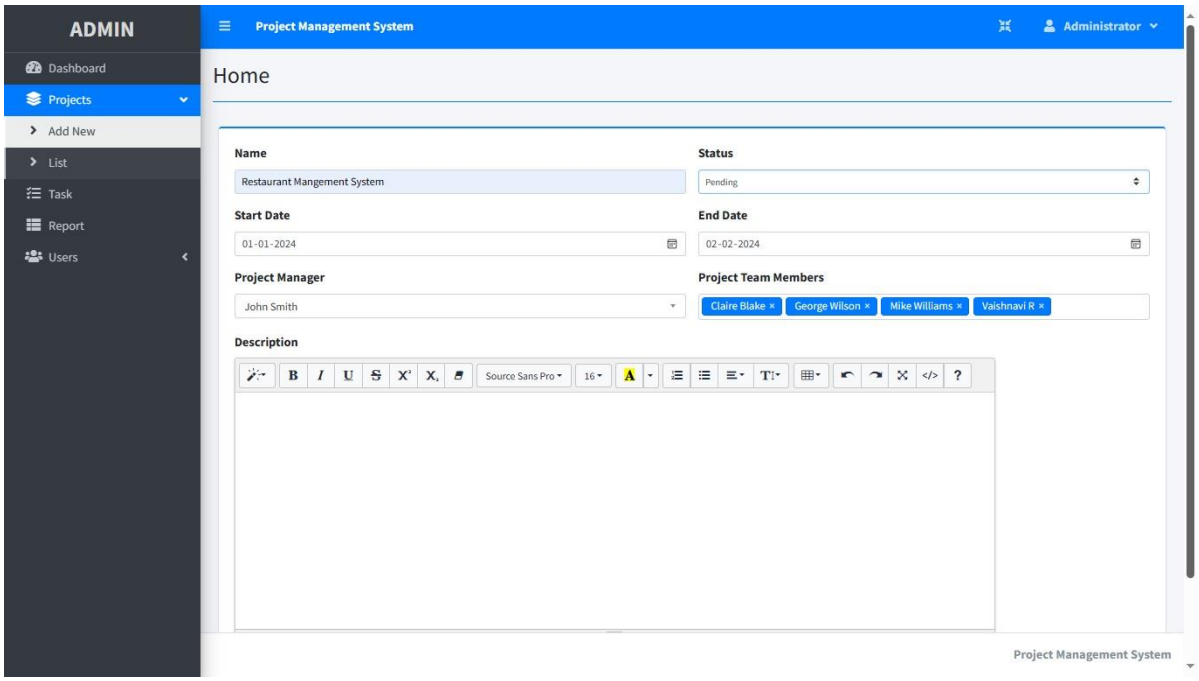


Fig.5 Admin assigning a manager to specific team members

ADD NEW TASK

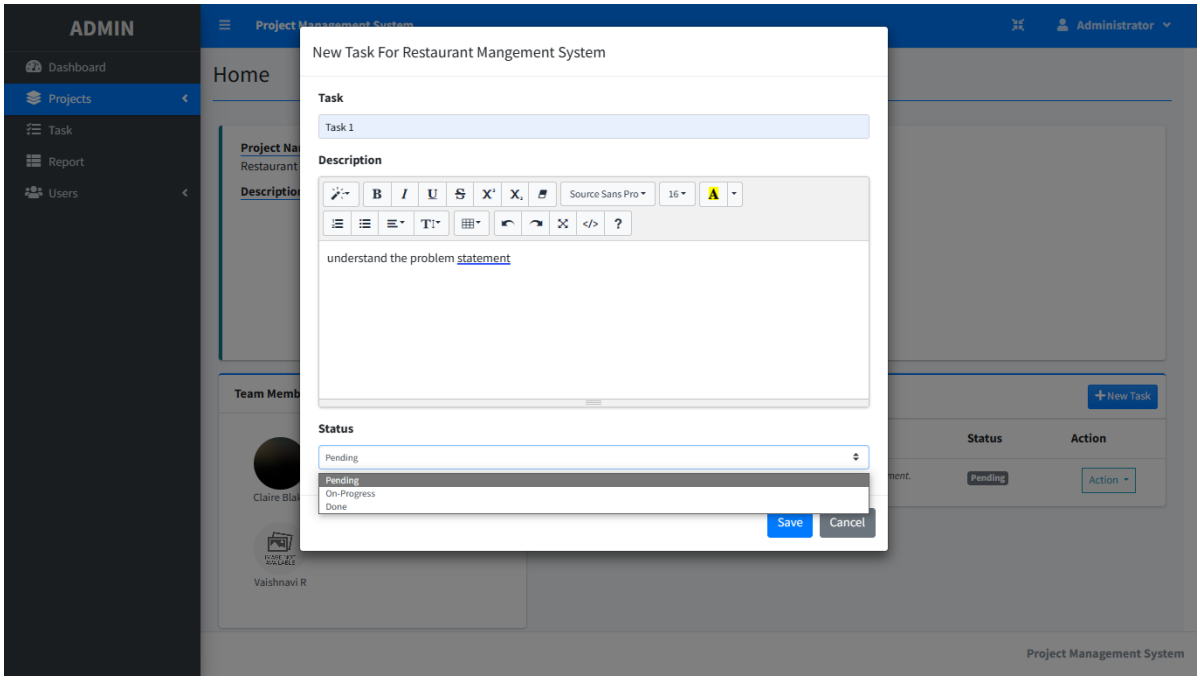


Fig.6 Admin assigning task to employee

(EMPLOYEE SIDE)

EMPLOYEE WILL BE ASSIGNED TASK

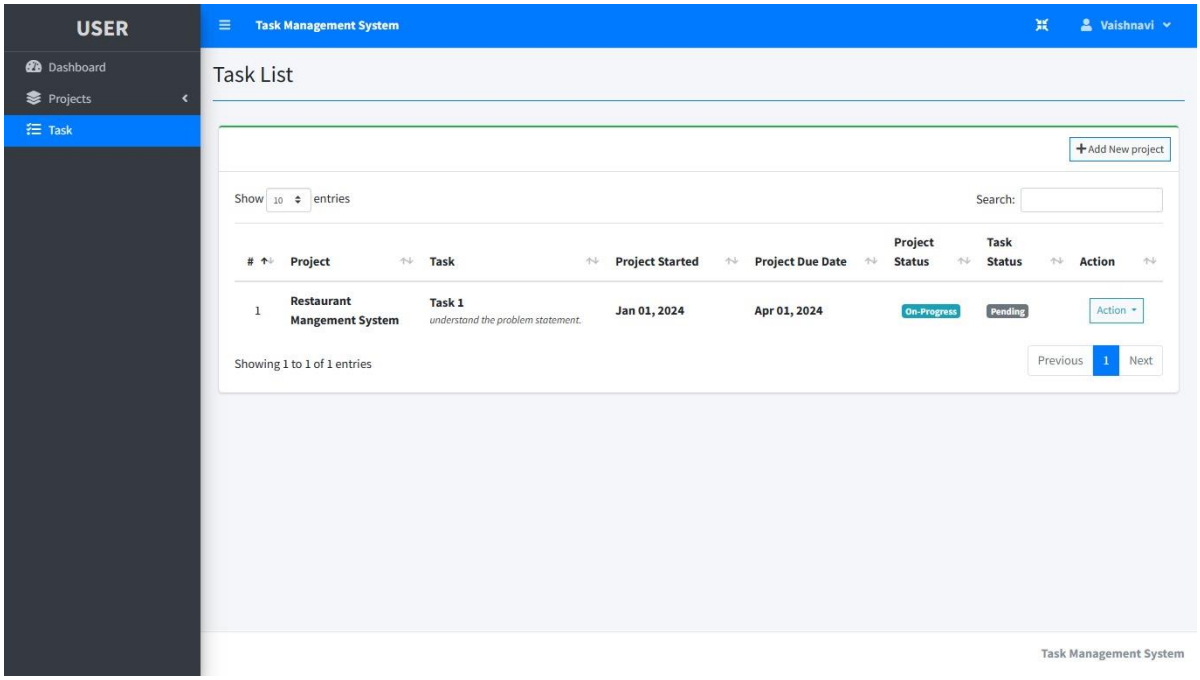


Fig. 7 User dash board

USER ADD PRODUCTIVITY

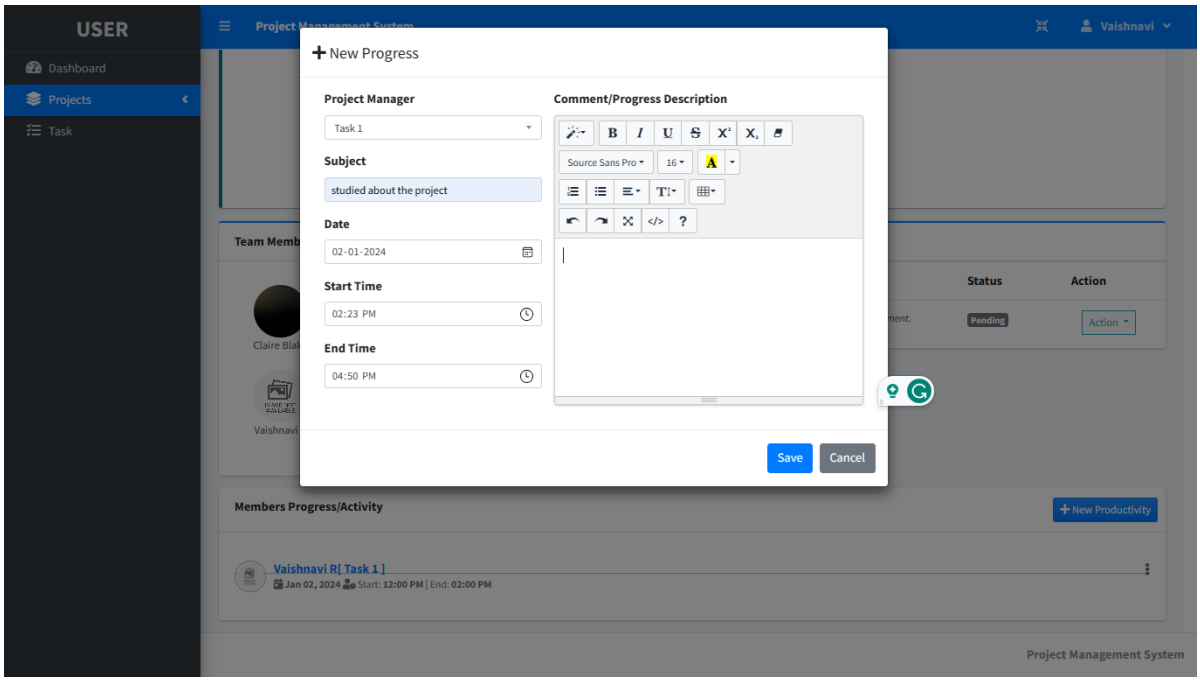


Fig.8 Manager assigning specific task to employee

ADMIN VIEW PRODUCTIVITY OF USER AND UPDATE STATUS

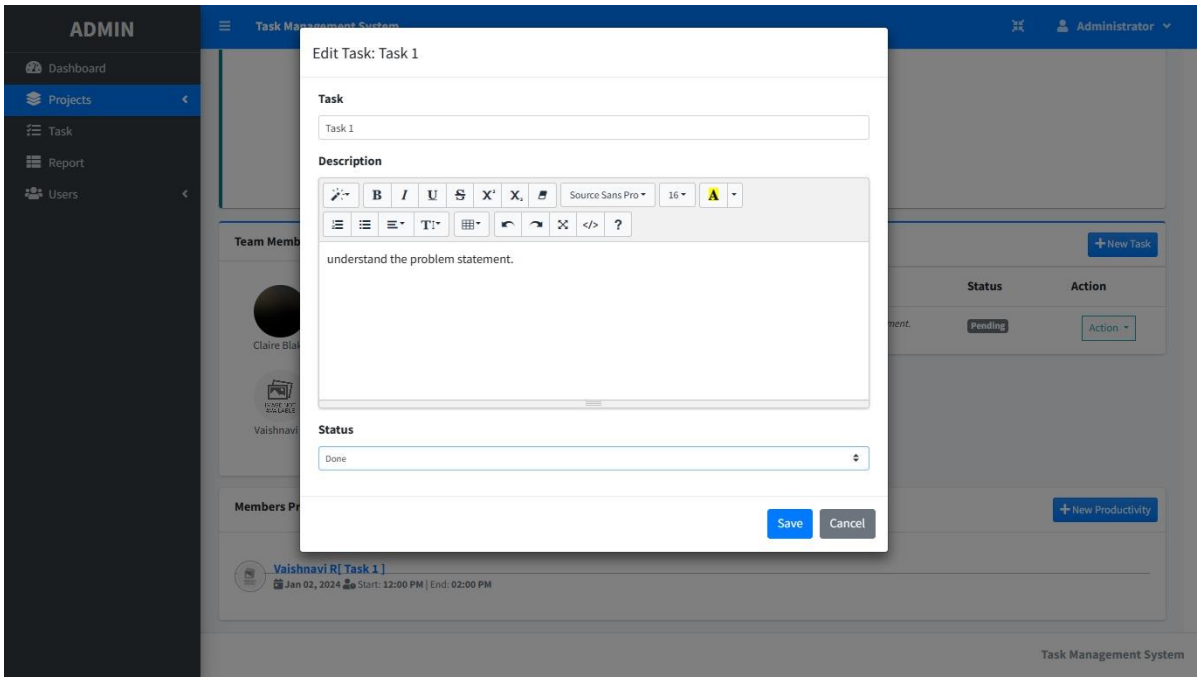


Fig.9 Status of the project

REPORT

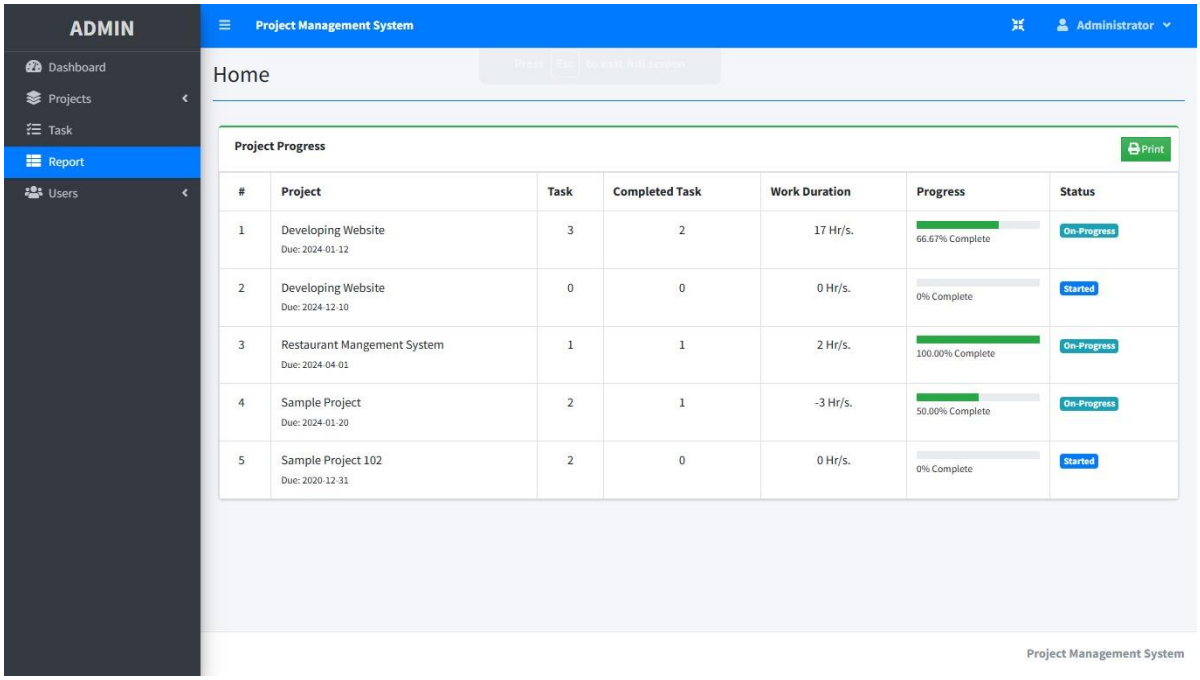


Fig.10 Reports status

APPENDIX-C

ENCLOSURES

7990172303 editor@ijcrt.org ijcrt.org

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS - IJCRT (IJCRT.ORG)

International Peer Reviewed & Refereed Journals, Open Access Journal

ISSN: 2320-2882 | Impact factor: 7.97 | ESTD Year: 2013

Scholarly open access journals, Peer-reviewed, and Refereed Journals, Impact factor 7.97 (Calculate by google scholar and Semantic Scholar | AI-Powered Research Tool) , Multidisciplinary, Monthly, Indexing in all major database & Metadata, Citation Generator, Digital Object Identifier(DOI)

Dear Author, Congratulation!!!

Your manuscript with Registration/Paper ID: IJCRT_249454 has been Accepted for publication in the International Journal of Creative Research Thoughts (IJCRT) | www.ijcrt.org | ISSN: 2320-2882 | International Peer Reviewed & Refereed Journals, Open Access Online and Print Journal.

IJCRT Impact Factor: 7.97 | UGC Approved Journal No: 49023 (18)

Check Your Paper Status: https://ijcrt.org/track.php

Your Paper Review Report :

Registration/Paper ID:	249454				
Title of the Paper:	Web-based Time And Productivity Analysis				
Criteria:	Continuity	Text structure	Understanding and Illustrations	Explanatory Power	Detailing
Points out of 100%:	87%	90%	88%	86%	88%
Unique Contents: 91%			Paper Accepted: Yes		
Overall Assessment (Comments):			Reviewer Comment Store in Online RMS System		
Publication of Paper:	Paper Accepted. Please complete payment and documents process. Paper will be published Within 01-02 Days after submission of payment proof and documents to editor@ijcrt.org. Complete below Step 1 and 2				

Check/Track Your Paper Status: https://ijcrt.org/track.php

Publication/Article Processing Fees

Indian Author	Foreign/International Author
₹ 1500 INR	$ 55 USD

STEP 1: Pay Publication Fees for Indian Author

Preferred Payment Link 1:	http://ijcrt.org/pay_form_2.php
(Use Preferred Payment Link 1 if not working then use below options.Using this paper will be published within 1 to 2 days.)	
Other Payment Link 2:	https://ijcrt.org/paymentInfoindian.php (Take More time in Payment Verification.)
UPI Payment Link 3:	https://ijcrt.org/upiijcrt.php (Take More time in Payment Verification.)

Only For Foreign/International Author Payment: Kindly Visit this Link

STEP 2: Send Payment Proof and Documents to editor@ijcrt.org

After Publication fees paid complete below process after that publication will take 1 to 2 days. Send payment screenshot and Documents to editor@ijcrt.org

Visit your AUTHORHOME using your Registration/Paper ID (IJCRT - 249454) and Your Registered Email ID -(as1579119@gmail.com) to complete below process. Login to Author Home https://ijcrt.org/track.php

Fill up Forms and Prepare Following Documents:

1. Final Formatted Manuscript/Research paper AS per IJCRT Paper Format (must be in

G149

report-Web-based Time and Productivity Analysis using Work Breakdown Structure and Agile Methodology

ORIGINALITY REPORT

15%	8%	6%	11%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Presidency University Student Paper	4%
2	Lathaniel Xavier Joseph Ablan, Nilo T. Bugtai, Robert Kerwin C. Billones. "Optimization and Time Management of Weekly Class Schedule with Activity Constraints for College Students using Linear Programming", 2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), 2022 Publication	1%
3	www.coursehero.com Internet Source	1%
4	helda.helsinki.fi Internet Source	1%
5	www.kluniversity.in Internet Source	1%
6	artoftesting.com Internet Source	1%
7	Submitted to Higher Education Commission Pakistan Student Paper	1%
8	Submitted to Purdue University Student Paper	1%
9	Submitted to University of Wales Institute, Cardiff Student Paper	<1%
10	Submitted to University of Bolton Student Paper	<1%
11	theses.liacs.nl Internet Source	<1%
12	Submitted to Fiji National University Student Paper	<1%
13	Submitted to Manipal University Student Paper	<1%
14	Submitted to Southern New Hampshire University - Continuing Education Student Paper	<1%
15	moviecultists.com Internet Source	<1%
16	Submitted to Colorado Technical University Student Paper	<1%

17	Submitted to University of Leicester Student Paper	<1 %
18	Submitted to King's Own Institute Student Paper	<1 %
19	apessay.elementfx.com Internet Source	<1 %
20	Submitted to CTI Education Group Student Paper	<1 %
21	1library.net Internet Source	<1 %
22	catalonica.bnc.cat Internet Source	<1 %
23	Mikko Raatikainen, Quim Motger, Clara Marie Luders, Xavier Franch et al. "Improved Management of Issue Dependencies in Issue Trackers of Large Collaborative Projects", IEEE Transactions on Software Engineering, 2022 Publication	<1 %
24	archive.org Internet Source	<1 %
25	ijeks.com Internet Source	<1 %
26	thequickadvisor.com Internet Source	<1 %
27	Nikolaos Spanoudakis, Pavlos Moraitis. "Chapter 4 The Agent Modeling Language (AMOLA)", Springer Science and Business Media LLC, 2008 Publication	<1 %
28	ebin.pub Internet Source	<1 %
29	essay365.x10.mx Internet Source	<1 %
30	terms.ameagle.com Internet Source	<1 %

Exclude quotes Off
Exclude bibliography Off
Exclude matches Off

SUSTAINABLE DEVELOPMENT GOALS

SDG 9 focuses on building resilient infrastructure, promoting inclusive and sustainable industrialization, and fostering innovation. In the context of a project management system, here's how it aligns:

Infrastructure Development: The project management system contributes to the development of digital infrastructure, enabling efficient collaboration and communication among team members. This supports the goal of building resilient and sustainable infrastructure.

Inclusive Industrialization: The system can facilitate inclusive project development by providing a platform for diverse team members, stakeholders, and communities to participate and collaborate in project activities.

Innovation: The project management system itself represents a form of innovation, streamlining project processes and improving efficiency. It supports the SDG 9 target of fostering innovation in various sectors.