INFO8003

Final Major Project

Developing a Business System

**Project Title:**  
*SkyRocketing*

**Theme:**  
An application to support small hi-tech, start-up companies

**Team Members:**

1. *Vinyake Rattan(8937185)*
2. *Joel*
3. *Suraj(8950720)*
4. *Deep Pate (906430)*

**Instructor:**  
*Amandeep Kaur*

**Submission Date:**  
*10/12/2024*

**Institution:**  
*[Conestoga College]*

**Course Code:**  
INFO8003

**Index:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Keyword** | **Page Number** | | Introduction | 3 | | Project Theme | 4 | | Preliminary Research | 4 | | Brainstorming | 5 | | Research – Web Market Analysis | 6 | | Description | 6 | | Technical Requirements | 7 | | Personas | 8 | | Use Cases | 9 | | Activity Diagram | 13 | | Entities | 14 | | Content Plans | 19 | | Prototypes | 25 | | Class Diagram | 31 | | Agile Plan | 33 | | **Conclusion** | 38 | |

**Project Title:** **SkyRocketing: A Business System for High-Tech Startups**

**Overview:** This project focuses on developing an innovative and scalable business system tailored for small high-tech startup companies. The system, named **SkyRocketing**, aims to streamline business operations, enhance collaboration, and promote sustainability. By integrating multiple functionalities into one cohesive platform, SkyRocketing aspires to be an indispensable tool for entrepreneurs seeking efficient growth and sustainability.

**Key Features:**

1. **Team Collaboration Tools**: Chat, file sharing, and task tracking.
2. **Project Management Tools**: Task delegation, scheduling, and milestone tracking.
3. **Client Relationship Management (CRM)**: Tools for nurturing client relationships.
4. **Analytics and AI Recommendations**: Workflow optimization and market insights.
5. **Sustainability Tools**: Energy tracking and eco-friendly resource suggestions.
6. **Customizable Dashboards**: Personalized layouts tailored to business needs.

**Development Team and Roles:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Team Member** | **Role** | **Responsibilities** | **Scrum Role** |
| **Vinny** | Front-End Developer | - Develop and implement user interfaces (UI) for the dashboard, task management, and calendar. - Ensure responsive and mobile-friendly design. - Collaborate with designers to maintain consistent branding and usability. | Development Team Member |
| **Joel** | Back-End Developer | - Build and maintain APIs for task management, user authentication, and data integration. - Ensure secure and scalable server-side architecture. - Implement database schema and queries. | Development Team Member |
| **Suraj** | Scrum Master | - Facilitate daily stand-ups, sprint planning, and retrospectives. - Remove roadblocks for the team to ensure smooth progress. - Monitor sprint progress and adherence to timelines. | Scrum Master |
| **Deep** | Product Owner | - Define and prioritize the product backlog. - Work closely with stakeholders to gather and refine requirements. - Ensure the product vision aligns with business goals and user needs. | Product Owner |

**Additional Roles:**

* **Collaborative Responsibilities:**
  + All team members contribute to testing, bug fixing, and peer reviews during the sprint.
  + Participate in Agile ceremonies such as sprint planning, reviews, and retrospectives.

This structure ensures effective collaboration and clear accountability, enabling the team to deliver the **SkyRocketing Project** efficiently.

**Introduction:**

The SkyRocketing project aims to develop an innovative business system tailored to support small high-tech startup companies. The focus is on addressing the challenges faced by startups in managing their operations, enhancing collaboration, and scaling their businesses efficiently. By integrating various functionalities into a single platform, SkyRocketing strives to become an indispensable tool for entrepreneurs seeking growth and sustainability.

**1. Project Theme**

Theme: The application will support small high-tech startup companies, helping them with various aspects such as business operations, communication, resource management, or project collaboration.

Proposed Application Name: SkyRocketing

This name suggests a startup that is rapidly growing and scaling, fitting for small high-tech companies looking to launch and grow quickly.

**2. Preliminary Research**

Key Research Areas:

Entrepreneurship: Understanding how small startups operate, key challenges they face, and the tools they need to succeed.

Sustainability: Ensuring the app can support eco-friendly practices, such as resource management or carbon footprint tracking.

Facilities Planning: Helping small businesses manage their office spaces, logistics, and supply chains.

AI Tools Used:

AI tools like GPT-4 were used to explore articles and guides about startup operations, sustainability in tech, and facilities management to build foundational knowledge.

Prompts Used:

* What are the challenges small tech startups face?
* How can sustainability be integrated into a tech startup's daily operations?
* Best practices in facilities management for small businesses.

Extended Insights:

* + Startups often struggle with resource allocation, time management, and maintaining competitive edges. Skyrocketing’s integration can alleviate these pain points.
  + Sustainability features may include real-time energy consumption tracking and environmentally friendly supply chain suggestions.
  + Facilities planning tools could help startups forecast space requirements based on growth projections.

**3. Brainstorming**

Idea Summary:

SkyRocketing will offer a comprehensive tool to help small startups streamline business operations such as project management, task delegation, and communication.

Key Features:

* Team Collaboration Tools: Chat, file sharing, and task tracking.
* Scheduling and Resource Management: Integrated tools to optimize team productivity.
* Client Relationship Management (CRM):Track and nurture client relationships.
* Sales and Marketing Tools: Customer insights and email campaign management.

Unique Selling Proposition (USP):

SkyRocketing focuses on supporting small businesses by simplifying multiple workflows into one platform that scales with the business.

Further Brainstorming Outcomes:

* AI-driven analytics for market trends and business performance.
* Interactive onboarding for new team members.
* Mobile-first design to cater to on-the-go entrepreneurs.

**4. Research – Web Market Analysis**

Current Solutions:

* Trello (Project Management)
* Slack (Communication)
* HubSpot (CRM)

Market Gap:

While there are individual tools available for specific needs, no platform combines all essential workflows for startups. SkyRocketing aims to fill this gap by offering a unified solution.

Differentiation:

* Integration of multiple functionalities into a single platform.
* Customizable dashboards tailored to business needs.
* AI-based assistance for workflow optimization and sales predictions.

Extended Market Analysis:

* Competitors lack in providing robust cross-functional integrations.
* Many startups use fragmented tools, which SkyRocketing seeks to replace with a cohesive system.
* Survey insights indicate a rising demand for all-in-one platforms among early-stage entrepreneurs.

**5. Description**

Refined Description:

SkyRocketing is an all-in-one platform designed for high-tech startups to streamline business operations, foster collaboration, and promote scalability.

Preliminary Features:

* Dashboard Customization: Personalized layouts to fit specific business needs.
* Integrated Team Management and CRM Tools: Unifying collaboration and client management.
* Automated AI Suggestions: Providing insights on workflow efficiency and potential sales strategies.

Advanced Feature Ideas:

* Integration with wearable devices for health and productivity tracking.
* AR/VR capabilities for immersive project planning.
* Automated compliance checks for legal and financial requirements.

Possible Revenue Streams:

* Subscription-Based Pricing Model: Monthly or annual subscription tiers based on features.
* Premium Features Available as Add-Ons: Advanced AI tools, additional storage, and exclusive integrations.
* Enterprise Solutions: Custom packages for larger organizations or scaling startups.

Future Goals:

* Expansion into augmented reality for project visualization.
* Integration with advanced analytics tools for detailed business performance reports.
* Partnership opportunities with co-working spaces and incubators for bundled offerings.

Roadmap for Future Development:

* Phase 1: Core platform development and beta testing.
* Phase 2: Implementation of AI and sustainability features.
* Phase 3: Global launch with multilingual support
* Phase 4: Continuous improvements based on user feedback and market trends.

Long-Term Vision:

SkyRocketing aspires to be the go-to platform for startups globally, enabling them to achieve their growth potential while maintaining operational efficiency and sustainability.

**6. Technical Requirements**

* **Hardware Features Needed:** The app will primarily run on cloud-based servers (AWS or Azure), and no specific hardware will be required. It’s a SaaS model.
* **Input Controls:**
  + **Task creation and updates** via form fields.
  + **Calendar interface** for scheduling and resource management.
  + **Dashboard interactions** for monitoring project and team metrics.
* **Constraints & Advantages:**
  + **Constraints:** Limited by the features of third-party APIs (for email, calendar, etc.).
  + **Advantages:** Cloud-based system allows scalability without significant hardware investments.

**7. Personas:**

**Persona 1:**

A person in a suit and tie

Description automatically generated

**Persona 2:**

A screenshot of a business presentation

Description automatically generated

**Persona 3:**

A blue and white background with text

Description automatically generated

**8. Use Cases**

|  |  |  |
| --- | --- | --- |
| Use case name | Create Task (SELECTED ½) | |
| Description | A user can create a new task, assign it to any team member, set deadlines, and attach resources | |
| actors | Admin, Team Member | |
|  | Admin | Team Member |
| The admin can create tasks and assign them to team members. | A team member can create tasks to track their team's progress. |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Manage Calendar (SELECTED 2/2) | | |
| Description | A user can schedule meetings, track project milestones, and sync with external calendars. | | |
| actors | Admin, Team member, External System | | |
|  | Admin | Team member | External system |
| The admin can schedule meetings, set project deadlines, and track milestones for the entire team | A team member can view scheduled meetings, add personal events or milestones, and set reminders. | External calendar tools (e.g., Google Calendar or Microsoft Outlook) may be integrated to sync events and milestones. |

|  |  |  |
| --- | --- | --- |
| Use case name | Generate Reports | |
| Description | A user can generate reports for projects, team performance, and resource allocation. | |
| actors | Admin, Team member | |
|  | Admin | Team member |
| Generates reports on team performance, project progress, and resource allocation | May access reports relevant to their tasks or projects |
|  |  |

|  |  |  |
| --- | --- | --- |
| Use case name | Manage Teams | |
| Description | An admin can add or remove team members, assign roles, and monitor their contributions. | |
| actors | Admin | |
|  | Adds or removes team members, assigns roles, and monitors contributions. |  |
|  |  |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Integrate External tools | | |
| Description | Users can sync the application with third-party tools like Google Calendar, Slack, or Trello. | | |
| actors | Admin, Team member, External system | | |
|  | Admin | Team member | External System |
| Configures and manages integrations with external tools | Accesses integrated tools as configured by the admin. | Provides APIs or services for integration. |
|  | | |

|  |  |  |
| --- | --- | --- |
| Use case name | Manage Notifications | |
| Description | Users can configure notifications for task updates, deadlines, and meetings. | |
| actors | Admin, Team Member | |
|  | Admin | Team Member |
| Configures system-wide notification settings | Personalizes notification preferences for tasks and events. |
|  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| Use case name | Manage Marketing Campaign | |
| Description | Marketing leads can create and monitor email campaigns or track leads through the CRM. | |
| actors | Marketing Lead, Admin | |
|  | Marketing Lead | Admin |
| Creates and manages marketing campaigns, tracks leads and evaluates performance. | Oversees marketing efforts and analytics at a higher level. |
|  |  |

|  |  |  |
| --- | --- | --- |
| Use case name | Generate Invoices | |
| Description | Admins can generate invoices for clients based on project milestones or services delivered | |
| actors | Admin | |
|  | Creates and sends invoices to clients based on project milestones or services rendered. |
|  |
|  |

A diagram of a company's workflow

Description automatically generated

**9. Activity Diagram:**

A diagram of a task

Description automatically generated

**10. Entities:**

1. **Entities diagram:**

*The following figure shows the* ***Entity Relationship Diagram (ERD)*** *for the* ***SkyRocketing*** *platform. It depicts key entities like* ***User****,* ***Task****,* ***Project****,* ***Team****,* ***Client****, and* ***CRM****, along with their attributes and relationships. This diagram illustrates how the platform's data structure supports operations such as task management, project tracking, and client relationship management.*

*A diagram of a computer network

Description automatically generated with medium confidence*

**Entity Relationship Diagram (ERD) - Detailed Explanation**

The **Entity Relationship Diagram (ERD)** for **SkyRocketing** illustrates the database structure of the application and represents how different entities interact with each other. The SkyRocketing platform is an all-in-one tool designed to help small, fast-growing startups manage their business operations, communication, resources, and projects. The ERD serves as a blueprint for how the platform will store, retrieve, and manage data.

**1. Entities and Their Attributes**

**User Entity**

The **User** entity stores all the details related to the platform users. These users may include entrepreneurs, team members, or administrators who interact with the platform.

* **userID (PK)**: A unique identifier for each user. This is a primary key (PK), ensuring that every user in the system can be uniquely identified.
* **name**: The full name of the user (e.g., "Sarah" or "David").
* **email**: The email address of the user. This field can be used for communication and login purposes.
* **role**: The role of the user (e.g., Admin, Manager, Developer). This is important because it determines the user’s access rights and what actions they can perform in the system.
* **password**: A hashed version of the user’s password, used for authentication.

**Task Entity**

The **Task** entity represents the individual tasks that need to be completed as part of a project. These tasks are assigned to users, and each task has specific details such as its status, description, and deadline.

* **taskID (PK)**: A unique identifier for each task. It serves as the primary key.
* **title**: The title or name of the task, which gives a brief overview of what the task is about (e.g., "Complete Product Design").
* **description**: A more detailed description of the task, explaining the steps or objectives involved.
* **status**: The current status of the task (e.g., "In Progress", "Completed", "Pending"). This helps track task progress.
* **startDate**: The date when the task was initiated or started. This helps in tracking project timelines.
* **assigneeID (FK)**: Foreign Key (FK) referencing the **User** entity. It identifies which user the task is assigned to. A user can be assigned multiple tasks.
* **projectID (FK)**: Foreign Key (FK) referencing the **Project** entity. It indicates which project the task is a part of.

**Project Entity**

The **Project** entity captures the details of a project within the platform. A project can have multiple tasks and teams assigned to it, and it helps track the overall progress.

* **projectID (PK)**: A unique identifier for each project. This is a primary key.
* **name**: The name of the project (e.g., "Mobile App Development").
* **description**: A description of the project, explaining its scope, objectives, and key deliverables.
* **startDate**: The date the project was started. This helps track project timelines.
* **endDate**: The expected or actual end date of the project. This field can be nullable as some projects may not have an end date set immediately.

**Team Entity**

The **Team** entity stores information about the different teams that work on the various tasks within the project. Teams can include members with different roles, such as developers, designers, or marketing experts.

* **teamID (PK)**: A unique identifier for each team. This serves as the primary key.
* **teamName**: The name of the team (e.g., "Frontend Development Team").
* **members**: A count or a list of the number of members within the team. This can include the user IDs of the members associated with the team.

**Client Entity**

The **Client** entity stores information about the clients of the startup. These could be external businesses or individuals that the startup works with, either on projects or as a customer for the product or service being developed.

* **clientID (PK)**: A unique identifier for each client. This is the primary key.
* **name**: The name of the client organization or individual.
* **email**: The email address of the client, used for communication.
* **contactPerson**: The name of the contact person from the client’s organization (e.g., a project manager or business representative).

**CRM Entity**

The **CRM** (Customer Relationship Management) entity captures client-related interactions and notes. This entity helps the startup manage customer relationships and track the status of different client interactions.

* **crmID (PK)**: A unique identifier for each CRM record.
* **clientID (FK)**: Foreign Key referencing the **Client** entity. This helps link the CRM record to a specific client.
* **notes**: Notes related to the client, which could include customer feedback, project status updates, or communication logs.

**2. Relationships Between Entities**

The relationships between entities define how they are connected and how data flows between them.

**User to Task (One-to-Many Relationship)**

* **One user can create and be assigned many tasks**, but each task is assigned to only one user.
* The **User** entity is linked to the **Task** entity through a **one-to-many relationship**. This relationship indicates that a single user can have many tasks, but each task is assigned to one user.

**Explanation**: For instance, Sarah, as the entrepreneur, may create multiple tasks related to different aspects of the project, but each task can be assigned to a different team member.

**Task to Project (Many-to-One Relationship)**

* **Many tasks belong to one project**, but each task is part of only one project.
* The **Task** entity has a **many-to-one relationship** with the **Project** entity. This means a project can have many tasks, but each task belongs to only one project.

**Explanation**: For example, a project such as "Mobile App Development" will have many tasks (e.g., design, coding, testing), but each task belongs to this one project.

**Task to Team (Many-to-One Relationship)**

* **Many tasks can be managed by one team**, but each task is handled by only one team.
* The **Task** entity is linked to the **Team** entity with a **many-to-one relationship**. This relationship signifies that a team can work on multiple tasks, but each task is managed by only one team.

**Explanation**: A team (e.g., the frontend development team) will be responsible for managing multiple tasks, but each task can only be assigned to one team.

**Client to CRM (One-to-Many Relationship)**

* **One client can have multiple CRM records**.
* The **Client** entity has a **one-to-many relationship** with the **CRM** entity. This relationship shows that a single client can have several CRM records associated with them (e.g., different projects, feedback, or notes related to that client).

**Explanation**: For example, a client such as "Tech Innovators Inc." may have multiple CRM entries, tracking different aspects of their projects or their communications with the startup.

**Project to Team (One-to-Many Relationship)**

* **One project can involve multiple teams**, but each team can work on several projects.
* The **Project** entity is related to the **Team** entity through a **one-to-many relationship**. This shows that a project can have multiple teams working on different aspects of the project (e.g., design team, development team, testing team).

**Explanation**: For example, a project such as "Mobile App Development" will involve multiple teams working on various tasks related to the project.

**3. Nullability and Constraints**

* **Nullable Fields**: Some attributes like **endDate** in the **Project** entity or **notes** in the **CRM** entity can be nullable. Not every project will have an end date immediately, and not all CRM records will have notes at creation time.
* **Non-nullable Fields**: Attributes like **userID**, **taskID**, **projectID**, **teamID**, **clientID**, and **crmID** are non-nullable, as they are required to uniquely identify records within their respective entities.

**4. Summary of Cardinality**

* **One-to-Many (1:N)**: This relationship is used where one instance of an entity is associated with many instances of another entity. For example, a **User** can create and be assigned many **Tasks**, but each **Task** is only assigned to one **User**.
* **Many-to-One (N:1)**: This relationship signifies that many instances of one entity can be linked to one instance of another entity. For example, many **Tasks** can belong to one **Project**.

**11. Content Plans:**

**Content Plan for SkyRocketing**

**1. Dashboard Screen**

The **Dashboard** is the central hub of the application. It will be the first page the user sees after logging in. This page needs to provide an overview of all the important metrics and updates the user needs to stay informed about the business.

**Content and Features:**

* **Navigation Bar:**
  + Links to various parts of the application (e.g., Projects, Tasks, Calendar, Team, Settings).
* **Welcome Message/Overview:**
  + A personalized greeting such as "Welcome, [User Name]!"
  + A brief summary of tasks due today, upcoming meetings, or deadlines.
* **Widgets/Task Overview:**
  + A widget displaying the list of active tasks with progress bars, highlighting overdue or upcoming tasks.
* **Recent Activities:**
  + A section showing recent activity like new tasks, messages, or team updates.
* **Quick Access Buttons:**
  + A button to quickly add new tasks, schedule meetings, or check the project status.
* **Notifications Section:**
  + Alerts for tasks, deadlines, or project updates.

**Input Controls:**

* **Search Bar:**
  + A search bar at the top to quickly find specific tasks, projects, or team members.
* **Dropdown Menus:**
  + Options for filtering the tasks, such as sorting by due date, priority, or team member.
* **Task Creation Button:**
  + A floating action button to create new tasks directly from the Dashboard.

**2. Task Management Screen**

The **Task Management** screen is where users can create, view, and manage tasks associated with projects. It will allow for detailed task tracking, assignments, deadlines, and progress updates.

**Content and Features:**

* **Task List:**
  + A list view or Kanban-style board showing all tasks and their statuses (To Do, In Progress, Completed).
  + Tasks will have different colors to indicate their status.
* **Filters and Sorting Options:**
  + Filters to view tasks by priority, assigned person, or deadline.
  + Sorting options to reorder tasks by their due date, priority, etc.
* **Task Details Panel:**
  + When a task is clicked, a panel or modal will display more details like description, assignee, comments, and any associated files.
* **Task Creation Form:**
  + A form to add a new task with fields for the title, description, due date, priority, and assignee.
* **Progress Tracker:**
  + A visual progress bar showing the current completion status of the task.

**Input Controls:**

* **Text Fields:**
  + For entering task name, description, and deadlines.
* **Drop-down Menus:**
  + For selecting assignees and priorities.
* **Checkboxes or Radio Buttons:**
  + For marking tasks as completed or assigning categories (e.g., urgent, high-priority).

**3. Calendar Screen**

The **Calendar** screen is crucial for scheduling meetings, setting deadlines, and managing milestones. It integrates with other functionalities like tasks and projects to give users a visual representation of their work schedule.

**Content and Features:**

* **Monthly/Weekly View:**
  + Toggle between monthly or weekly views to see tasks and events.
* **Event Details:**
  + When clicking on an event or task, the user can see more information such as time, description, and attendees.
* **Drag-and-Drop Scheduling:**
  + Users can easily reschedule tasks and meetings by dragging them to different time slots.
* **Milestone Tracker:**
  + Track important project milestones and their deadlines.
* **Integration with Tasks:**
  + Tasks that have deadlines will appear on the calendar, showing their due dates.

**Input Controls:**

* **Date Picker:**
  + A date-picker widget to select specific days to view tasks or meetings.
* **Time Slot Selection:**
  + Users can select times to schedule meetings or set deadlines.
* **Create New Event Button:**
  + A button to schedule new tasks, meetings, or milestones directly from the calendar view.

**4. Project Management Screen**

The **Project Management** screen allows users to manage and track the progress of individual projects. It’s an essential feature for startup teams to ensure tasks are on track, and project goals are being met.

**Content and Features:**

* **Project List:**
  + Display active projects with their progress percentages, due dates, and team members involved.
* **Project Progress Tracker:**
  + A visual bar or pie chart showing the overall progress of the project.
* **Project Details Page:**
  + Clicking on a project shows a detailed view with all tasks, milestones, and team members associated with it.
* **Add New Project Button:**
  + A button to create a new project by entering project name, description, and assigning team members.
* **Milestone Tracker:**
  + A visual timeline that tracks key project milestones and their deadlines.

**Input Controls:**

* **Text Inputs:**
  + For entering the project title, description, and milestones.
* **Dropdowns or Multi-select:**
  + For assigning team members and defining project stages.

**5. Team Management Screen**

This screen focuses on team collaboration and resource management. It allows startup teams to manage their roles, assign responsibilities, and communicate within the app.

**Content and Features:**

* **Team Member List:**
  + Display all members, their roles, and status (active, on leave, etc.).
* **Roles and Permissions:**
  + Administrators can assign roles and permissions, such as Team Leader, Developer, or Designer.
* **Activity Feed:**
  + A live feed showing updates from team members, such as task completions or project comments.
* **Team Member Profile Pages:**
  + Profiles for each team member displaying their tasks, projects, and overall performance.

**Input Controls:**

* **Role Assignment Dropdown:**
  + Dropdown menu to assign roles to new members or update existing ones.
* **Search Box:**
  + A search bar to find team members based on name, role, or department.

**6. Settings and Profile Screen**

The **Settings** screen lets users manage their account settings, notifications, and integrations with other tools (e.g., email, calendar).

**Content and Features:**

* **Profile Information:**
  + Users can update their name, email, password, and profile picture.
* **Notification Preferences:**
  + Options to set notification preferences for tasks, events, and project updates.
* **App Integrations:**
  + Settings for connecting to external tools like Slack, Google Calendar, and email clients.
* **Account Management:**
  + Options to upgrade the plan, view billing history, or delete the account.

**Input Controls:**

* **Text Fields:**
  + For updating the user profile information.
* **Checkboxes:**
  + For toggling notification settings on/off.
* **Buttons:**
  + For saving changes, upgrading plans, or logging out.

**Additional Features to Consider**

1. **AI Recommendations:**
   * AI can suggest improvements for task allocation or project planning based on the company’s past performance and user behavior.
2. **Collaboration Tools:**
   * Include real-time document sharing, chat channels, or video conferencing within the app to help teams stay in sync.
3. **Reports and Analytics:**
   * Generate reports on productivity, resource usage, and team performance to provide valuable insights for decision-making.

By organizing the content for each screen in this way, we ensure that every part of the application is carefully considered in terms of functionality, ease of use, and value to the user. Each screen's layout is designed with a focus on minimizing clutter while maximizing usability. This will help you build a comprehensive and coherent app interface for SkyRocketing.

**12. Prototypes:**

**1. Dashboard Screen**

The **Dashboard Screen** serves as the central hub for the SkyRocketing app. It provides a quick overview of the most important information, including active tasks, project status, deadlines, and team updates. The navigation bar at the top gives users easy access to other areas like Tasks, Projects, CRM, Calendar, and Settings. Widgets such as Task Summaries, Project Overviews, and Notifications allow users to get an at-a-glance view of the critical aspects of their workload and upcoming events. Additionally, floating action buttons enable quick creation of tasks and projects.

Screens screenshot of a chat

Description automatically generated

A screenshot of a phone

Description automatically generated

**2) Task Management Screen.**

The **Task Management Screen** is designed to help users efficiently manage their tasks. It displays a list of all tasks with key details like title, assignee, due date, and status. Tasks can be filtered by various parameters, such as progress or assigned team members. When a task is clicked, the user is directed to the Task Detail View where they can update task information, assign team members, adjust deadlines, add attachments, and mark tasks as completed. Action buttons like "Save Changes" and "Delete Task" provide full control over the task’s lifecycle.

A screenshot of a phone

Description automatically generated

Screens screenshot of a task management app

Description automatically generated

**3) Project Management Screen.**

The **Project Management or Project Manager Screen** allows users to oversee and manage ongoing projects. It features a project list that includes key information such as the project name, start date, team lead, and progress. A detailed view for each project displays more in-depth information, such as the project description, associated tasks, and team members. Action buttons allow users to create new tasks for specific projects, edit project details, or delete projects. The progress bar visually represents the completion of each project, making it easy for users to track progress at a glance.

Screens screenshot of a phone

Description automatically generated

**4) Calendar Screen**

The **Calendar Screen** provides users with a clear view of upcoming meetings, task deadlines, and project milestones. The monthly calendar view displays tasks, meetings, and important events, while individual events show detailed information, such as event name, date, description, and assignee. A floating action button allows users to quickly add new events or tasks, which can be filled in with relevant details such as title, date, description, and assignee. This screen helps users stay organized and ensures that deadlines and appointments are met.

Screens screenshot of a phone

Description automatically generated

**5) CRM Screen**

The **CRM (Customer Relationship Management) Screen** is designed for users to track and manage client relationships. The client list displays essential information such as client name, email, contact person, and relationship status (Lead, In Progress, Converted). The detailed view for each client includes contact details, a notes section for client interactions, a sales pipeline view to track the client’s progress, and recent interaction history. Action buttons allow users to add new clients, update relationship status, and record additional notes future reference.

Screens screenshot of a phone

Description automatically generated

**6) User Profile Screen**

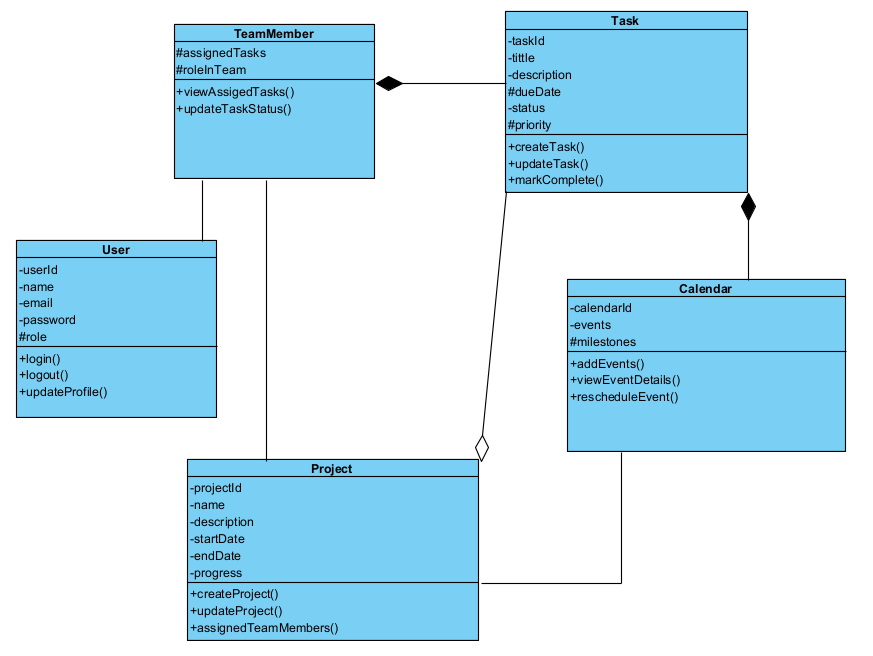
The **User Profile Screen** enables users to manage their personal information and account settings. The screen displays the user’s name, email, and role, with the option to edit profile details. Users can also update their password by entering the current password and setting a new one. Additionally, the screen provides options to toggle notification settings, allowing users to control which alerts they receive for tasks, projects, and deadlines. This screen helps users maintain control over their personal settings within the app, ensuring they are always up to date with their account preferences.

Screens screenshot of a phone

Description automatically generated

Each of these screens is crucial for the efficient operation of the SkyRocketing app, ensuring that users can easily manage tasks, projects, clients, and their own profiles. Together, these prototypes form the backbone of a comprehensive tool that enhances user productivity and supports business growth.

**13. Class Diagram:**

****

Relationships:

**User and TeamMember**: Inheritance (is-a)

TeamMember inherits all User properties and methods, but it also has unique properties like roleInTeam and assignedTasks.

**Project and Task**: aggregation

A project is made up of several task items. Although they can function separately, these duties are logically bundled under the project.

**Task and TeamMember**: Association

A TeamMember can work on more than one Task object, and a Task can be assigned to more than one TeamMember object.

**Calendar and Task:** Composition

Multiple task deadlines or timelines are listed on a calendar, and the scheduling specifics of these tasks are closely linked to the calendar.

**Calendar and Project:** Association

A Calendar is linked to multiple Project objects to track milestones and deadlines, but the lifecycle of the projects is not tied to the calendar.

**Project and TeamMember**: Association

A Project involves multiple TeamMember objects, and each TeamMember can contribute to multiple Project objects.

**14. Agile Plan:**

Here are **three sprints** for the SkyRocketing project development, considering an Agile methodology with a focus on delivering a Minimum Viable Product (MVP) while incorporating critical features progressively.

**Sprint 1: Core Functionality Development**

**Objective:** Build the foundation of the platform with essential features to allow basic project management and team collaboration.

**Duration:** 2 weeks

**Key Features:**

1. **User Authentication & Profile Management:**
   * User login/signup with role-based access (Admin, Team Member, Marketing Lead).
   * Profile management (update name, email, and password).
2. **Dashboard:**
   * Centralized dashboard displaying active tasks, recent activities, and quick access buttons.
   * Personalized greeting and task summary for each user.
3. **Task Management:**
   * Create, assign, and track tasks.
   * Display tasks in a list and Kanban-style board.
4. **Team Management:**
   * Add/remove team members and assign roles.
   * View a list of team members with statuses (active, inactive).
5. **Basic Notifications:**
   * Notification alerts for task assignments and deadlines.

**Deliverables:**

* Fully functional web-based platform with task creation, assignment, and tracking.
* User-friendly dashboard with real-time updates.
* Tested and deployed core features.

**Sprint 2: Advanced Scheduling & Integration**

**Objective:** Enhance productivity features by adding scheduling tools, calendar integration, and basic analytics.

**Duration:** 3 weeks

**Key Features:**

1. **Calendar Integration:**
   * Monthly and weekly views for task deadlines and meetings.
   * Drag-and-drop scheduling for tasks and events.
   * Sync with external tools like Google Calendar or Microsoft Outlook.
2. **Milestone Tracker:**
   * Track project milestones with visual timelines.
   * Automatic updates on milestone completion.
3. **Basic CRM Integration:**
   * Simple client tracking (name, contact, project details).
   * Integration with external email clients for communications.
4. **Reports:**
   * Generate basic reports on task completion, team performance, and project progress.
   * Export reports as PDFs.
5. **Enhanced Notifications:**
   * Configurable notifications for events, deadlines, and task updates.
   * Personalized settings for notification preferences.

**Deliverables:**

* Scheduling tools integrated into the platform.
* A functioning calendar with external tool synchronization.
* Basic reporting capabilities for tracking performance metrics.

**Sprint 3: AI-Driven Features and Scalability**

**Objective:** Introduce AI-driven tools for optimization and prepare the platform for scalability and broader usability.

**Duration:** 3 weeks

**Key Features:**

1. **AI Recommendations:**
   * Workflow optimization suggestions based on task and team data.
   * Predictive analytics for sales trends and resource allocation.
2. **Customizable Dashboards:**
   * Allow users to personalize dashboard layouts and widgets.
   * Include options for toggling between different visualizations (e.g., charts, progress bars).
3. **Marketing Campaign Management:**
   * Tools for creating and tracking email campaigns.
   * Basic analytics for campaign performance.
4. **Scalability Features:**
   * Multilingual support to cater to global audiences.
   * Improved cloud infrastructure for handling larger teams.
5. **Security Enhancements:**
   * Data encryption for user and client information.
   * Role-based permissions for secure access.

**Deliverables:**

* AI-powered insights integrated into the system.
* Enhanced marketing and CRM functionalities.
* Scalable and secure platform ready for global deployment.

**Summary of Sprints:**

* **Sprint 1** lays the foundation with core functionalities.
* **Sprint 2** focuses on enhanced productivity and collaboration through scheduling, milestones, and reports.
* **Sprint 3** delivers AI-driven insights, marketing tools, and scalable infrastructure to complete the MVP and prepare for growth.

**Appendices**

**Appendix A: AI Tools and Prompts**

**AI Tool Used:** ChatGPT

* **Prompt 1:**  
  *"Generate innovative application ideas that can support small hi-tech start-ups. Consider entrepreneurship, sustainability, and facilities planning."*
  + **AI Output (Excerpt):**
    - Idea 1: A resource management platform that helps start-ups track and optimize office space and equipment usage.
    - Idea 2: A virtual mentor matching system connecting start-ups with experienced advisors in their field.
    - Idea 3: A funding opportunity tracker tailored to niche industries.
* **Prompt 2:**  
  *"What are key features that should be included in an app to support sustainability in small start-ups?"*
  + **AI Output (Excerpt):**
    - Green supply chain tracking tools.
    - Energy consumption monitoring.
    - Carbon footprint calculators with actionable tips.
* **Adaptations:** These ideas were refined to align with the team's goals, selecting Idea 2 (mentor matching system) as the primary concept.

**Appendix B: Research Materials**

**Sources Consulted:**

1. "Best Practices for Entrepreneurship in Technology Start-Ups"
2. "Sustainable Facilities Planning for Modern Businesses"
3. "Existing Apps for Start-Up Support: Competitor Analysis"

**Key Insights:**

* **Competitor Analysis:**
  + Applications like LinkedIn and AngelList provide some networking features but lack personalized mentoring for niche tech start-ups.
  + No existing platform combines sustainability and mentor connections effectively.
* **Market Gaps Identified:**
  + Lack of tools for environmental impact tracking in start-up operations.
  + Limited resources tailored to early-stage founders in emerging industries.

**Appendix C: Agile Planning Spreadsheet**

**Link to Repository:** [Insert GitHub, Azure, or AWS repository link here]

**Spreadsheet Overview:**

* The Agile Plan outlines three 3-week sprints, focusing on incremental development of features and use cases.
* Key milestones include:
  + Sprint 1: Basic functionality and prototype design for mentor matching.
  + Sprint 2: Integration of sustainability features and user interface refinement.
  + Sprint 3: Testing, feedback incorporation, and final polishing.

**Appendix D: Additional Supporting Materials**

1. **Brainstorming Notes:**
   * Mentor matching system identified as the central concept.
   * Secondary features brainstormed: funding alerts, sustainability tracking.
2. **Rough Sketches:**
   * Initial wireframe for user onboarding.
   * Dashboard concept for tracking mentor-mentee interactions.
3. **Meeting Minutes:**
   * Weekly team discussions documented in shared files.

**Appendix E: Glossary of Terms**

* **Mentor Matching System:** A platform feature that connects start-ups with experienced professionals based on shared goals and expertise.
* **Sustainability Tracking:** Tools that monitor and provide insights on the environmental impact of business activities.
* **ERD (Entity-Relationship Diagram):** A diagram that visually represents the data structure and relationships within a database.

**Conclusion:**

The **SkyRocketing project** is a visionary initiative designed to address the multifaceted challenges faced by small high-tech startups. Through its unified platform, it streamlines business operations, fosters team collaboration, and enables effective resource management. With a strong focus on sustainability, innovative features, and AI-driven insights, the application promises to be an essential tool for startups aiming to scale efficiently and sustainably.

Key differentiators, such as the integration of multiple functionalities into a single cohesive system and advanced analytics for decision-making, position SkyRocketing uniquely in the competitive landscape. By prioritizing usability, adaptability, and scalability, the platform ensures relevance across diverse startup ecosystems globally.

Looking forward, the development roadmap outlines a clear progression, from establishing core functionalities to implementing advanced technologies such as AR/VR and multilingual support. These planned enhancements reflect a commitment to continuous improvement and alignment with user needs and market trends.

In summary, **SkyRocketing** aspires to revolutionize the way startups manage and grow their operations, ultimately empowering entrepreneurs to achieve their goals with efficiency, innovation, and sustainability at the forefront.