Community skill exchange platform
Seek&Share

Feasibility Study

The objective for the feasibility study of the Seek&Share skill exchange platform is to perform a comprehensive evaluation to assess the practicality, viability, and potential success of the proposed software system before its development. Based on the analysis of the obtained information, we plan to make informed choices about how to proceed with the development of the software project.

For the feasibility study, we wanted to evaluate the project from 3 different perspectives, **technical feasibility studies** focusing on technology assessment, **operation feasibility studies** evaluating the operational needs and **economic feasibility studies** assessing financial viability.

I. Technical Feasibility

Technical feasibility studies focus on the technology assessment of the proposed software solution. In our study, we're evaluating the proposed solution's technologies capabilities against the desired functionality, scalability, and performance requirements of the application.

* **Maturity of technology:** Mature technologies offer greater stability, reliability, and support, making them more conducive to successful development and implementation. The chosen technology stack is mature as well, reducing the likelihood of technical issues, bugs, or failures during development and operation.
* **Importance of technical expertise of the team:** Since the chosen technology stack is mature, the project's development team already has past experiences with the it and this will allow them to leverage their expertise and insights to develop a quality product. The experienced team members can contribute valuable knowledge and best practices to streamline development processes. If any skill gaps are found within the team, appropriate training and resources can be made available to enhance their expertise in areas critical to project success.

Technologies chosen for the proposed software solution -

- Backend Development - Java, SpringBoot
- Frontend Development - Angular, HTML5, CSS
- Infrastructure & Database Management - MongoDB, AWS
- Version Control - Git

* Backend Development

- Backend development is responsible for handling user requests, managing data, and executing business logic through server-side components. Inefficient backend implementation can give rise to s**low response times, high latency, and sluggish performance** of the application that can frustrate users and deter them from using the platform.
- However, the chosen backend technology, **Spring Boot** is a very lightweight Java framework that facilitates the creation of high-performance backend services by **minimizing overhead and unnecessary dependencies**. Java's robust concurrency model and memory management capabilities contribute to the efficient execution of backend logic, reducing latency and improving throughput
- Spring Boot's comprehensive ecosystem offers built-in support for various components such as RESTful APIs, security mechanisms, data access layers and Java's object-oriented programming paradigm that promotes the **creation of modular and reusable code** allows developers to **rapidly develop and deploy backend services**, ensuring that the platform meets the functional requirements outlined in the project scope.
- Spring Boot applications can be deployed in a distributed architecture, allowing for horizontal scaling to **handle growing demand of user traffic and data volume**. By leveraging AWS services, it is possible to automate deployment and scaling processes, ensuring that the platform can s**cale seamlessly based on workload fluctuations**. The framework’s support for asynchronous processing and reactive programming enables the development of scalable and responsive backend services that can handle concurrent requests efficiently.

⁕ Frontend Development

- The chosen frontend technology, **Angular** is a robust frontend framework capable of developing **dynamic and responsive user interfaces**. Its component-based architecture facilitates the creation of reusable UI components, ensuring **consistency and modularity** across the application and its two-way data binding facilitates the creation of rich, interactive web applications that deliver a consistent and seamless user experience across devices and browsers
- Angular's modular architecture and lazy loading capabilities support scalability by enabling the **efficient loading of resources** and application based on user interactions. Developers can also employ code-splitting techniques, minimizing bundle sizes, and leveraging browser caching mechanisms to ensure that the platform scales seamlessly to meet growing user requirements.
- Angular's support for server-side rendering and progressive web app features enhances performance by enabling **faster initial page loads and offline functionality**. Angular's ahead-of-time (AOT) compilation and tree-shaking capabilities optimize code execution and reduce bundle sizes, improving application load times and rendering performance. It also supports a reactive programming paradigm that ensures that the UI updates are performed asynchronously and seamlessly.

⁕ Infrastructure & Database Management

- **MongoDB** is a notable NoSQL database that provides high levels of flexibility in data modelling and storage, allowing developers to store and manage diverse data types efficiently. **AWS** on the other hand offers developers the ability to provision virtual servers, store and retrieve data, and manage relational databases that help deploy scalable and resilient infrastructure components and ensure a seamless access to platform resources.
- MongoDB's horizontal scaling capabilities enable developers to distribute data across multiple nodes, ensuring that the database **can handle growing data volumes and concurrent user requests** effectively. AWS offers scalable infrastructure services such as auto-scaling, load balancing, and content delivery networks (CDNs), enabling developers to scale resources dynamically based on workload fluctuations and ensure that the **platform remains responsive and available** even during peak usage.

- MongoDB's in-memory caching and indexing capabilities optimize data access and retrieval, reducing query latency and improving database performance. AWS's global network infrastructure and data centres ensure **low-latency access to platform resources from anywhere in the world**, improving application responsiveness and user satisfaction. Its database services are also optimized for performance and reliability, ensuring that database operations are executed efficiently.

⁕ Git

- Git facilitates robust version control for the platform, allowing developers to track changes, manage branches, and merge code seamlessly. It ensures version control integrity and enables collaborative coding through features like branching and merging.
- Git's distributed architecture ensures scalability by enabling developers to work independently on local copies and synchronize changes seamlessly. This approach allows teams to scale their development efforts effectively while maintaining code consistency.
- Git is optimized for performance, offering fast operations and efficient data storage. Its compression techniques minimize repository size, while caching mechanisms accelerate common operations, ensuring smooth performance even in large-scale development environments.

Based on these capabilities and advantages provided by the technology stack, we can conclude that the proposed solution is technically feasible

II. Operational Feasibility

Analysis of the operational impact of the proposed solution on existing processes -

The market analysis reveals that the needs of our target audience is currently not being met by the existing competitors in the space of community skill exchange platforms. It shows us that there is still a large portion of the audience that is unhappy with the current offerings and is looking for alternatives.

I. The report revealed that our biggest competitor, **Meetup** showed 2 main weakness -

   a. **The first being, its heavy reliance on physical gatherings that inherently limits scalability and accessibility, particularly in situations where in-person events are impractical or restricted.**

      ↳ Our proposed solution incorporates virtual events and hybrid models that blend online and offline interactions overcoming the limitations of physical gatherings. This approach enables users to participate in skill exchange activities from anywhere, at any time, thereby enhancing accessibility and scalability.

      ↳ It also allows for greater flexibility and inclusivity, accommodating users with diverse schedules and geographical locations.

   b. **The second, its lack of structured learning resources compared to online platforms events.**

      ↳ Our platform offers a wide range of structured learning resources, including tutorials, courses, workshops, and webinars, covering various skill domains.

      ↳ The proposed solution leverages data analytics and machine learning algorithms to offer personalized learning paths tailored to each user's skills, interests, and learning goals. By analyzing user behaviour and preferences, the platform can recommend relevant learning resources, events, and connections, helping users navigate their learning journey effectively.

II. Analysis on **Skillshare** revealed the following weaknesses -

a. Skillshare's limited focus on community interaction and peer-to-peer learning hinders opportunities for collaborative learning and knowledge exchange among users.

↳ The foundational ethos of our platform revolves around fostering a collaborative spirit among our users. At the core of Seek&Share lies the belief that knowledge is most effectively exchanged and cultivated through collective efforts and shared experiences. Our platform is built upon principles of mutual support, cooperation, and the collective pursuit of learning and skill development.

↳ Our platform encourages users to host one-on-one and even one-to-many knowledge exchange sessions where users are not only motivated to continuously learn and develop new skills but also to give back to the community by teaching a learnt skill to the other members. Users are also encouraged to host virtual meetups, study groups, networking events, work on group projects, peer review assignments, utilize discussion forums, chat rooms, and interest groups to share their insights, experiences, and resources

b. Subscription model may deter price-sensitive users from accessing the platform's content and services, potentially limiting user acquisition and retention.

↳ Our platform utilizes a virtual token system called energy, an in-app purchasable item will allow the user to register themselves for either an in-person or virtual skill exchange events

↳ Newly registered users can begin their learning journey by purchasing energy (bundle of 5) through the app thus paving the way for a pay-as-you-go payment model allowing for flexibility and removing the barrier to entry for price-sensitive users. When a user desires to learn a skill, energy has to be spent to register themselves for skill exchange sessions. Once the user depletes his five energies, only upon teaching a skill of his expertise will the ability to purchase energy be unlocked

Identification of Challenges and Benefits -

Challenges:

* **User Adoption**

- User adoption is crucial for the success of the platform, as it determines the **extent to which users embrace and utilize its features**. Operational feasibility involves ensuring that the platform is user-friendly, intuitive, and accessible to its target audience.

- Designing a user-friendly and intuitive interface is crucial for encouraging adoption. However, **achieving simplicity while offering comprehensive functionality can be challenging**. Balancing these aspects requires extensive user research, usability testing, and iterative design improvements to meet diverse user needs and preferences.

- Sustaining user engagement and satisfaction over time requires ongoing effort and investment. **Users' needs and preferences evolve, necessitating continuous updates and improvements to the platform**. However, maintaining momentum and enthusiasm among users can be challenging, particularly if they encounter issues or perceive the platform's value to diminish over time. Therefore, fostering a culture of continuous improvement and innovation is essential for ensuring long-term user adoption and satisfaction.

* **Regulatory Compliance**

- Regulatory compliance is essential to ensure that the **platform operates within legal and ethical boundaries** and protects user rights and privacy. Ensuring adherence to relevant laws, regulations, and industry standards governing data protection, security, and user privacy requires great to detail and ongoing monitoring

- Regulatory landscapes can be **complex and constantly evolving**, with multiple laws and standards at the local, national, and international levels. Navigating these regulations requires in-depth knowledge and expertise to interpret their requirements accurately and ensure compliance.

- **Maintaining ongoing compliance with regulatory requirements and adapting to evolving regulations presents a complicated challenge** for the operational feasibility of our skill exchange platform. Regular audits and reviews are essential to assess the platform's adherence to relevant laws

and standards, evaluating data handling practices, security protocols, and privacy measures. These audits help detect and address compliance issues proactively, mitigating legal risks and ensuring continued adherence to regulatory requirements.

- **Staying informed about regulatory developments and adapting platform policies, procedures**, and technologies accordingly is essential for maintaining compliance over time. This proactive approach requires monitoring regulatory changes, assessing their impact on platform operations, and implementing necessary adjustments to ensure continued compliance.

Benefits:

* **Effective skill exchange**

- The proposed solution ensures effective skill exchange between the users and fosters an environment where users are motivated to continuously learn and develop new skills. It also emphasizes that users should not solely consume knowledge without also giving back to the community by contributing a skill. Similarly, it encourages users to engage in teaching others, provided they are actively learning and acquiring new skills themselves.
- Incorporating virtual events and hybrid models that blend online and offline interactions help overcome the limitations of physical gatherings. This allows for greater flexibility and inclusivity, accommodating users with diverse schedules and geographical locations and enables users to participate in skill exchange activities from anywhere, at any time, thereby enhancing accessibility and scalability.

* **High user engagement with the platform**

- With the implementation of a virtual token system called energy, an in-app purchasable item that allows the user to register themselves for either an in-person or virtual skill exchange events constantly engages the users to use the platform to learn new skills
- Through the development of the Progress passport system boasting gamification elements like badges and achievements for mastering skills, completing courses, contributing valuable insights to discussions, and helping others in the community etc. and participating in leaderboard chasing, challenges and special events foster a sense of achievement to everyone who makes progress and keep them actively engaged in the platform

Based on the above analysis of the operational impact of the proposed solution on existing processes and the huge extent of possible benefits, we can conclude that the proposed solution is operationally feasible

III. Economic Feasibility

Economic viability is a key criterion for assessing the feasibility of a skill exchange platform, with a focus on measuring its economic viability and potential return on investment (ROI) These assessments include cost-benefit analysis through analysis of resource availability and the calculation of potential ROI

Resource Availability -

- Resources availability include the financial, human and technical resources required to develop, deploy and maintain the platform. The resource availability assessment includes an assessment of the organization's financial strength, existing talent, and technical infrastructure to ensure adequate resources are allocated to develop and operate the platform
- Financing resources include funding for software development, infrastructure installation, marketing, and ongoing operations
  - ***The initial investment required for developing the platform is estimated to be around $250,000.*** This includes expenses such as software development, UI/UX design, database management, and marketing. Additionally, ongoing operational costs, which include hosting fees, maintenance, and support costs are estimated to be approximately $50,000 per year.
  - **To ensure economic feasibility, the organization has allocated a budget of $300,000 for its first year**, covering both initial investment and operational costs. This indicates that there is sufficient financial resources available to support the platform's development and operation.

- HR refers to experienced professionals with expertise in software development, UI/UX design, database management, and community management.
  - ↳ Experienced employees are essential for the successful development and operation of the platform. The organization has a dedicated team of software developers, UI/UX designers, database administrators, and community managers who are very knowledgeable in their respective fields.
  - ↳ In addition, the organization has invested in training programs to enhance the skills of existing employees and recruit new talent as needed. With a capable and experienced team in place, the organization is well-equipped to handle the technical and operational aspects of the platform effectively.
- Technical resources include hardware, software, and cloud services required to support platform functionality and scalability.
  - ↳ Technological infrastructure, including hardware, software, and cloud services, is required to support the platform's functionality and scalability. The organization has invested in state-of-the-art server infrastructure and cloud services from providers such as Amazon Web Services (AWS) and Microsoft Azure. This ensures that the platform can be easily scaled to meet users' growing demands while maintaining optimal performance and reliability.

Considering the available financial, human, and technological resources, it is evident that the Community Skill Exchange Platform is economically feasible. The organization has allocated extensive budget resources, assembled an experienced team, and invested in a robust technical infrastructure to support platform development and operation.

With careful planning and resource management, the platform is poised for success in providing valuable skill exchange opportunities within the community while ensuring long-term sustainability and growth.

Return on Investment (ROI) -

Estimating the potential ROI involves analyzing the expected profitability and revenue generated by the platform compared to the initial investment and ongoing operational costs. Potential revenue streams include the energy token system, advertising, and transaction fees. Additionally, the platform may generate indirect benefits such as increased brand visibility, customer loyalty, or community engagement, which contribute to its overall ROI.

1. **Energy Token Revenue:**
   i. Each Energy token bundle, consisting of 5 Energy tokens, is priced at $15.
   ii. Assuming 1,000 newly registered users purchase an Energy token bundle each month, the monthly revenue from Energy token sales would be $15,000.
   iii. Over the course of a year, the projected annual revenue from Energy token sales would be $180,000.
2. **Advertising Revenue:**
   i. Assuming an average click-through rate (CTR) of 1% and a cost-per-click (CPC) rate of $0.50, with 100,000 monthly ad impressions, the monthly revenue from advertising would be $500.
   ii. Over the course of a year, the projected annual revenue from advertising would be $6,000.
3. **Convenience Fees:**
   i. Assuming an average transaction fee of 5% on transactions worth $100,000 annually, the projected annual revenue from transaction fees would be $5,000.

Total Projected Annual Revenue = Energy Token Revenue + Advertising Revenue + Transaction Fees

Total Projected Annual Revenue = $180,000 + $6,000 + $5,000

Total Projected Annual Revenue = $191,000

ROI = (Total Projected Annual Revenue - Initial Investment) / Initial Investment

ROI = ($191,000 - $250,000) / $250,000

ROI ≈ -0.236 or -23.6%

4. **Indirect Benefits:**

   i. **Increased Brand Visibility**: The platform's presence in the community and on the internet enhances brand visibility, attracting more users and potential partners. This increased visibility can lead to partnerships, sponsorships, and collaborations that contribute to revenue generation and growth.

   ii. **Community Engagement**: A thriving community ecosystem drives active participation, user retention, and word-of-mouth referrals. Engaged users contribute valuable content, insights, and feedback, enhancing the platform's offerings and driving user acquisition and retention.

   iii. Calculating the exact value of indirect benefits may be challenging, but their contribution to the platform's overall success and ROI is significant. As such, it's essential to recognize and leverage these indirect benefits to maximize the platform's economic feasibility and long-term sustainability.

Based on these estimates, the projected ROI for the first year remains negative, indicating that the platform may struggle to cover its initial investment and operational costs.

Considering only the operational costs are expended the coming years and the possible increase in the number of platform users, sponsors, assumed cost from indirect benefits coupled with some adjustments to revenue streams, cost structures, and monetization strategies, the proposed solution is economically feasibility and on track to achieve positive ROI in subsequent years.

Project Plan

The "Software Project Plan" for Seek&Share aims to establish a comprehensive roadmap for the development and deployment of the software solution. Seek&Share is designed and constructed to be in a Top Down planning fashion and will follow an agile model. The Work breakdown structure entails the creation of a Gantt chart highlighting key phases and milestones, accompanied by an allocation of time to each project phase to ensure timely execution.

Milestones and deliverables are meticulously identified and described, spanning from requirements gathering to post-deployment activities, ensuring clarity and

accountability throughout the project lifecycle. Resource allocation delineates the distribution of human and technological resources across various phases, while critical dependencies are identified to mitigate risks and streamline project progression effectively.

Service Level Agreements:

1. Platform Availability:

- Uptime Guarantee: The platform will be available and accessible to users 99.9% of the time, excluding scheduled maintenance windows for every month on weekends. Any downtime exceeding this threshold will be promptly addressed, and users will be notified in advance of scheduled maintenance.

2. Response Time:

- Support Ticket Response: The platform's support team will acknowledge and respond to user inquiries or issues submitted via the support ticket system within 24 hours of receipt during regular business hours.

- Technical Assistance: Seek&Share will provide technical support to users experiencing issues with platform functionality or usability. The support team will strive to resolve technical issues promptly, with a target resolution time of 48 hours for non-critical issues and 24 hours for critical issues.

3. Data Security and Privacy:

- Data Protection: Seek&Share will implement robust security measures to protect user data from unauthorized access, breaches, or loss. All user data will be stored securely and encrypted, complying with relevant data protection regulations.

- Privacy Policy Compliance: Seek&Share will adhere to a comprehensive privacy policy that outlines how user data is collected, used, and managed. Users will have the option to review and agree to the privacy policy upon registration.

4. Skill Matching Algorithm:

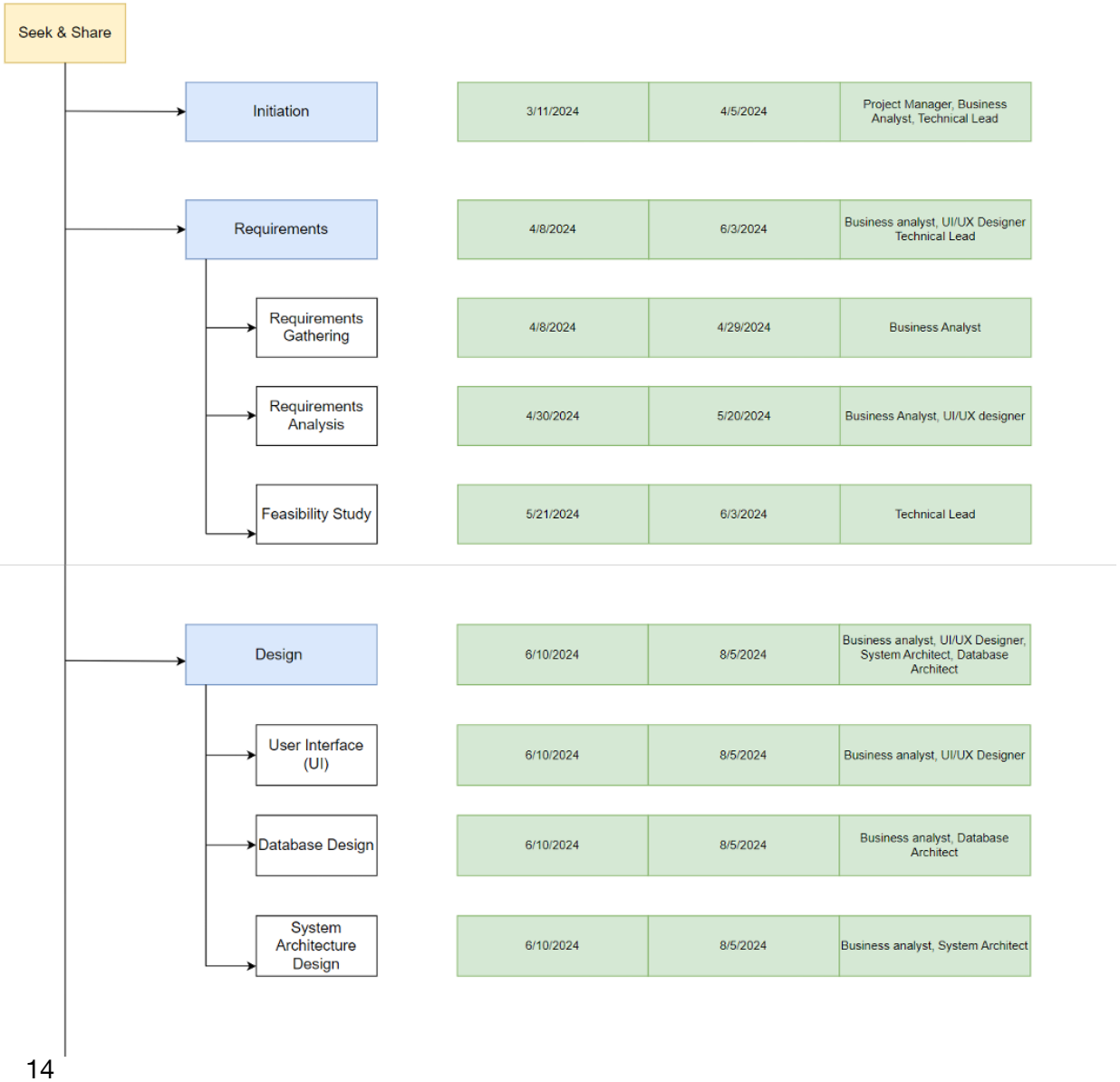
- Matching Accuracy: Seek&Share's skill matching algorithm will strive to provide accurate and relevant recommendations to users based on their skill interests and preferences. The platform will aim for a matching accuracy rate of 90% or higher, continually refining the algorithm based on user feedback.

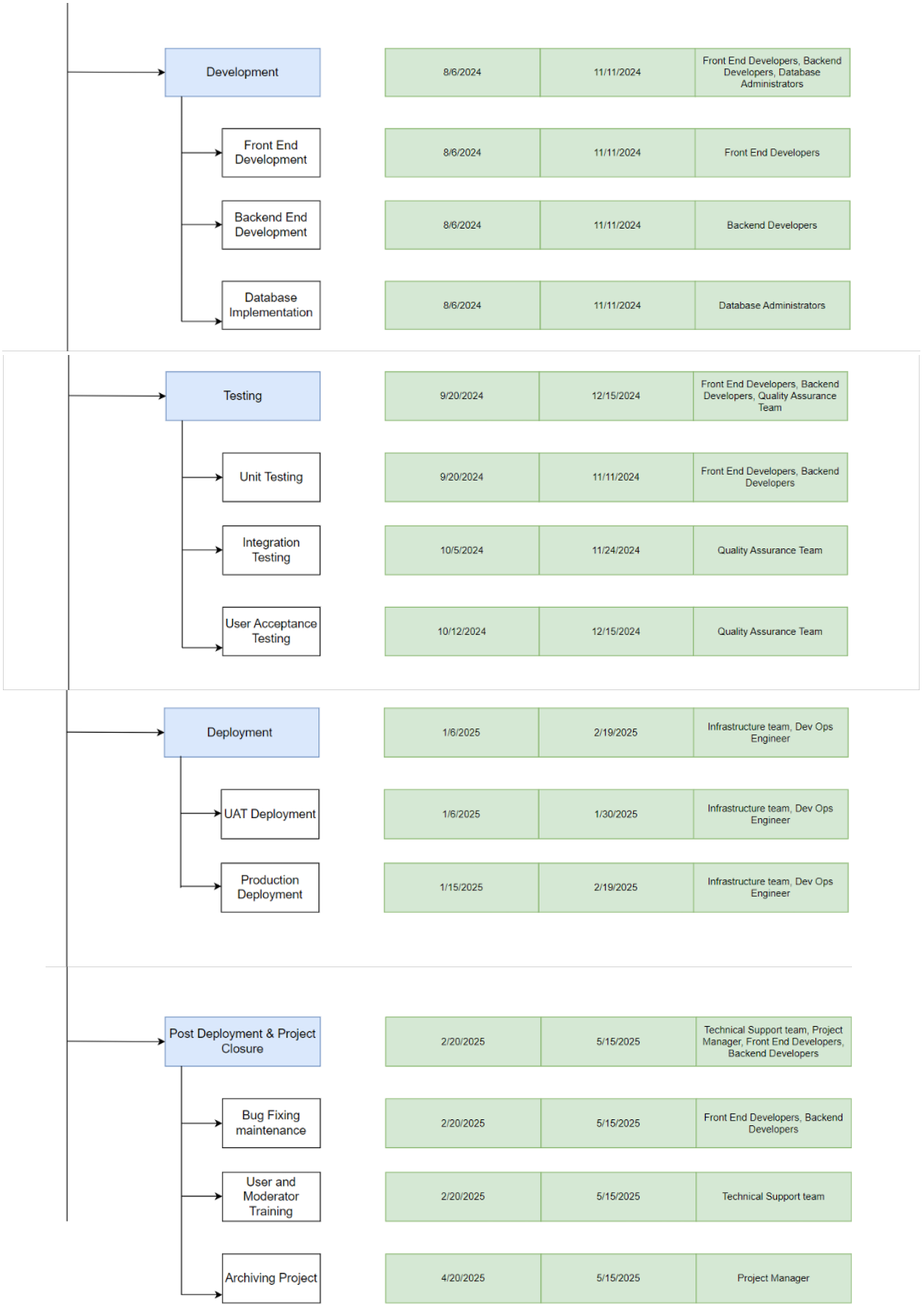
5. Technical Support:

- Technical Assistance: Seek&Share will provide technical support to users experiencing issues with platform functionality or usability. The support team will strive to resolve technical issues promptly, with a target resolution time of 48 hours for non-critical issues and 24 hours for critical issues.

Work Breakdown Structure:

The Work Breakdown Structure (WBS) for the "Software Solution Project Plan" divides tasks into manageable phases and it's durations and resources involved. Initiation involves project kickoff, requirements involves requirements gathering, analysis and feasibility study. Design includes user interface, database, and system architecture. Development covers front-end, back-end, and database implementation. Testing entails unit, integration, and user acceptance testing. Deployment includes user acceptance and production deployment. Post-deployment tasks involve bug fixing, user & moderator training, and project archiving.




Development
8/6/2024
11/11/2024
Front End Developers, Backend Developers, Database Administrators
Front End Development
8/6/2024
11/11/2024
Front End Developers
Backend End Development
8/6/2024
11/11/2024
Backend Developers
Database Implementation
8/6/2024
11/11/2024
Database Administrators
Testing
9/20/2024
12/15/2024
Front End Developers, Backend Developers, Quality Assurance Team
Unit Testing
9/20/2024
11/11/2024
Front End Developers, Backend Developers
Integration Testing
10/5/2024
11/24/2024
Quality Assurance Team
User Acceptance Testing
10/12/2024
12/15/2024
Quality Assurance Team
Deployment
1/6/2025
2/19/2025
Infrastructure team, Dev Ops Engineer
UAT Deployment
1/6/2025
1/30/2025
Infrastructure team, Dev Ops Engineer
Production Deployment
1/15/2025
2/19/2025
Infrastructure team, Dev Ops Engineer
Post Deployment & Project Closure
2/20/2025
5/15/2025
Technical Support team, Project Manager, Front End Developers, Backend Developers
Bug Fixing maintenance
2/20/2025
5/15/2025
Front End Developers, Backend Developers
User and Moderator Training
2/20/2025
5/15/2025
Technical Support team
Archiving Project
4/20/2025
5/15/2025
Project Manager

Project Timeline:

The project timeline for Seek & Share spans over a year and 2 months to comfortably develop the project. Buffers are kept in place such that in the event of any extraordinary circumstances like employer attrition, unavailability these buffers are kept in place.

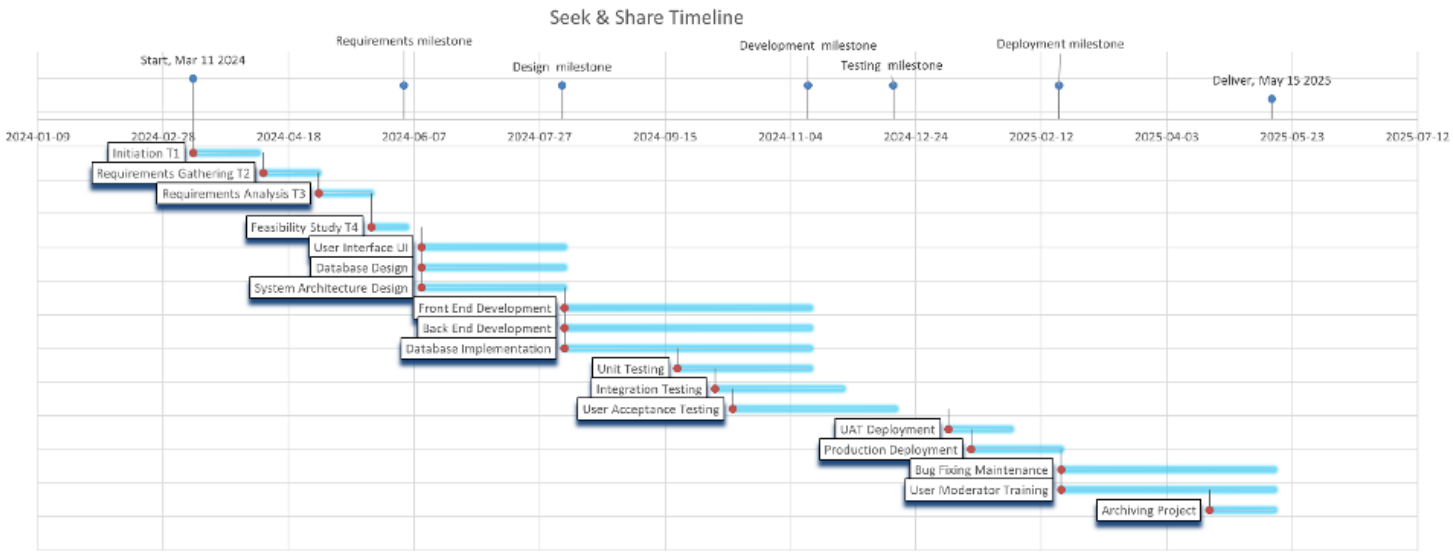
Start date – 03/11/2024 End Date – 05/15/2025

Phase	Task Name	Duration(Weeks)	Dependencies
Initiation	Initiation - T1	3.5	
Requirements	Requirements gathering – T2	3	
	Requirements Analysis – T3	3	T2
	Feasibility Study – T4	2	T2,T3
Design	User Interface Design – T5	7.5	
	Database Design – T6	7.5	
	System Architecture – T7	7.5	
Development	Front End Development – T8	13	T5
	Back End Development – T9	13	T7
	Database Implementation – T10	13	T6
Testing	Unit Testing – T11	7	T8,T9,T10
	Integration Testing – T12	7	T8,T9,T10
	User Acceptance Testing – T13	5	T8,T9,T10
Deployment	UAT Deployment – T14	3	T11, T12
	Production Deployment – T15	4	T13
Post Deployment & Project Closure	Bug Fixing Maintenance – T16	11	
	User and Moderator Training – T17	11	

Phase	Task Name	Duration(Weeks)	Dependencies
	Archiving Project – T18	3.8	

Gantt Chart for Seek & Share:

The Gantt Chart describes the various phases and it's subtasks spanning over the year and gives us a visualization of which tasks are overlapping and which are dependent on other tasks



Major Milestones & it's Deliverables:

I. **Requirements Milestone:** In this milestone, it is aimed to deliver the requirements analysis report and the feasibility study done for the project. A detailed document is submitted, pointing out each and every detail provided by the stakeholders and the scope for changes in each phase. This milestone is crucial since, this provides the foundation for the entire project.

- **Requirements Document**: This comprehensive document outlines all the functional and non-functional requirements of the Seek&Share platform. It includes features, user roles, system capabilities, and constraints. It serves as a reference for all stakeholders involved in the project.

- **User Stories and Acceptance Criteria**: User stories describe specific user interactions with the platform, focusing on what the user wants to

accomplish. Acceptance criteria define the conditions that must be met for a user story to be considered complete. These aid in prioritizing features and guiding development.

II. **Design Milestone :** In this milestone, it is aimed to deliver the designs of each major system in the project which include, UI/UX Design, System Design and Database Schema Design.

- **System Architecture Diagram**: This diagram illustrates the high-level structure of the Seek&Share platform, including components, modules, and their interactions. It defines how different parts of the system communicate with each other and ensures scalability, reliability, and performance.

- **Wireframes and Mockups:** Wireframes of user interface, outlining the layout and structure of screens or pages. Mockups are high-fidelity visualizations that provide a more detailed look and feel of the UI design. They help in visualizing the user experience and gathering feedback from stakeholders.

- **Database Schema:** The database schema defines the structure of the database, including tables, relationships, and constraints. It serves as a blueprint for organizing and storing data efficiently. The schema design ensures data integrity, consistency, and optimal performance.

III. **Development Milestone :** In this milestone, it is aimed to deliver the functional application of the project and this is the most crucial aspect of the project and the most volatile due to various factors affecting resources.

- **Functional Application:** During this phase a prototype is built and showed to the stakeholders for feedback and on an iterative basis, development will take place. Towards the end of this phase, a functional application will exist.

- **Version Control Systems:** The code repository, often managed using version control systems like Git, stores the source code of the platform. It enables collaboration among developers, tracks changes, and facilitates code management, review, and deployment.

IV. **Testing Milestone:** In this milestone, it is aimed to test the various modules of the project, like unit testing, stress testing, functional testing, integration testing and User Acceptance Testing. This is also a crucial phase since the quality of the project will depend on this phase.

- **Test Plans and Test Cases:** Test plans outline the testing approach, scope, objectives, and resources for validating the Seek&Share platform. Test cases are detailed instructions for executing tests and verifying the system's behaviour against expected outcomes.

- **Test Reports:** Test reports document the results of various testing activities, including test execution, defects found, and their status. They provide insights into the quality and readiness of the platform for deployment and inform decision-making.

- **Unit Tests:** Unit tests are automated tests designed to validate individual units or components of the system, such as functions, methods, or classes. They ensure that each unit behaves as expected and help detect and fix defects early in the development process.

- **Bug Tracking System:** A bug tracking system, such as Jira, facilitates the management and tracking of reported defects or issues throughout the testing and development phases. It helps prioritize and resolve bugs efficiently to ensure the platform's stability and reliability.

V. **Deployment Milestone:** In this milestone, the deployment of the project is made to UAT servers and Production servers and this will start when testing is done halfway, since it would be a better time for a closed set of users to test in the UAT servers and real time monitoring will be done in production servers

- **Deployed Application:** The deployed Seek&Share platform is made accessible to users in the production environment. It involves configuring servers, installing software, deploying code, and ensuring the platform's availability, security, and performance.

- **User Documentation:** User documentation provides instructions, guides, and tutorials to help users navigate and utilize the Seek&Share platform effectively. It covers features, functionalities, best practices, and troubleshooting tips to enhance user experience and adoption.

- **Post-Deployment Support Plan:** The support plan outlines the procedures, responsibilities, and resources for addressing user inquiries, issues, or maintenance tasks after the platform's deployment. It ensures ongoing support and continuous improvement of the platform's performance and reliability.

Resource Allocation:

For each phase of the project, resources need to be properly allocated and make sure their responsibilities, intended tools for those tasks are clearly defined along with the timeline of their tasks. For this project there is an initial division of Resources into onshore and offshore resource. The resources and their roles are defined below.

Onshore Resources: Business Analyst, Project Manager, Technical Lead

Outsourced Resources: UI/UX Designer, System Architect, Database Architect, Database Administrator, Backend Developer, DevOps Engineer, Quality Assurance Team, Infrastructure Team and Technical Support Team

1. **Project Initiation:** We need to define project scope, objectives, and deliverables, identify stakeholders and establish communication channels and conduct initial project kickoff meeting

    i. **Human Resources:**

    - **Project Manager:** Responsible for overseeing the initiation phase, defining project scope, and assembling the project team. He is available throughout the end of the project.

    - **Business Analyst:** Conducts stakeholder interviews, gathers requirements, and documents project objectives. They also have to be there until the requirements phase so as to clear any vagueness with the system architects and Database architects.

    - **Technical Lead:** Coordinates with the business analyst in setting up the trajectory of the project. They set up technical goals and various methods to initiate the requirements study.

    ii. **Technological Resources:**

    - **Collaboration Tools:** Microsoft Teams

    - **Project Management Software:** Jira for task management and tracking

2. **Requirement Analysis:** We need to gather requirements from stakeholders, including users, moderators, developers, etc. Analyze user stories, use cases, and functional requirements and prioritize requirements based on business value and feasibility.

i. **Human Resources:**

- **Business Analyst:** Continues to gather detailed requirements from stakeholders. Discusses with UI Designer and technical lead to make the project objectives clear.
- **UX/UI Designer:** Works on wireframes and prototypes based on gathered requirements and analyze requirements. This is done to gain iterative user feedback on the designs.
- **Technical Lead:** Conducts Feasibility studies and evaluates potential frameworks for platform development. Co ordinates with Business Analyst to discuss platform frameworks.

ii. **Technological Resources:**

- **Prototyping Tools:** Figma for creating UI prototypes.
- **Requirements Management Tools:** Confluence for managing requirements.

3. **Design Phase:** To design the three main systems of the project which is Front end Interface, backend interface and Database, done by UI/UX Designers, Database Architects and System Architects respectively.

i. **Human Resources:**

- **UI/UX Designers:** Develops the user interface based on wireframes and prototypes. Gains concrete feedback compared to the previous phase.
- **Database Architect:** Reviews the schema requirements provided by business analyst. Once requirements are clear, they start to design the database schema and relationships.
- **System Architects:** Reviews the requirements and designs the whole system architecture required for the project. They are also responsible in designing the API endpoints.

ii. **Technological Resources:**

- **Design Software:** Adobe Creative Suite for UI design
- **Database Design Tools:** Mongo DB

4. **Development Phase:** Resources are required to involve in the development phase of the project in three main subsystems which is frontend, backend and database.

   i. **Human Resources:**

   - **Frontend Developers:** Implement the UI design using Angular,HTML,CSS,Node Js based on the requirements. They work in accordance with backend and database administrators if needed
   - **Backend Developers:** Build the server-side logic and RESTful APIs in Java, Spring Boot based on the requirements. They work in accordance with frontend and database administrators if needed
   - **Database Administrators:** Implement and manage the database schema and handling the databases and writing DB scripts. They work in accordance with backend and backend if needed.

   ii. **Technological Resources:**

   - **Integrated Development Environments (IDEs):** Eclipse
   - **Version Control Systems:** Git
   - **Backend Tech Stack:** Java, Postman, Spring Boot, Hibernate, Redis server and Spring cloud RESTful APIs
   - **Front End Tech Stack:** Angular, HTML, CSS, Node Js
   - **Database & Infra:** Mongo DB, AWS

5. **Testing Phase:** After development, resources are required to test individual components, perform integration testing and user acceptance testing.

   i. **Human Resources:**

   - **Quality Assurance Engineers:** Conduct unit testing, integration testing, and user acceptance testing. Ensures the project is functional throughout the testing and gives feedback to the team responsible for it.
   - **Frontend Developers:** Assists in Testing and work on feedback received by QA team. Responsible to minimize the issues provided by Testing team.

- **Backend Developers:** Assists in Testing and work on feedback received by QA team. Responsible to minimize the issues provided by Testing team.

ii. **Technological Resources :**

- **Testing Tools:** Mockito for automated testing of the backend of the project.
- **Bug Tracking Systems:** GitHub Issues for tracking and managing bugs.

6. **Deployment Phase:** Deploying project to UAT servers and Production servers. Scripts are maintained and used for each deployment. Careful monitoring of the server's health and potential downtime caused by issues in code.

i. **Human Resources:**

- **DevOps Engineers:** Manage deployment scripts, configurations, and deployment processes. They also maintain a CI/CD pipeline to automate the builds
- **Infrastructure Team:** Assists Dev Ops engineers in deployment and manages the whole infrastructure of the project. They provide provisions for cloud management like virtual machines, AWS

ii. **Technological Resources :**

- **Cloud Platforms:** AWS
- **Deployment Automation Tools:** Jenkins, Travis CI, or GitLab CI/CD for deployment processes

7. **Post-Deployment Support and Project Closure:** Addressing bugs based on priority and small change requests if it can be accommodated. Project closure is carefully documented so that the project documents are clean and relevant which can help for Knowledge Base.

i. **Human Resources:**

- **Technical Support Team:** Provides ongoing support to users and addresses any post-deployment issues. Helps in resolving issues given by the users

- **Frontend Developers:** Bug Fixing issues reported by technical team. Assist project manger in archival process
- **Backend Developers:** Bug Fixing issues reported by technical team. Assist project manger in archival process
- **Project Manager:** Oversees the project closure and project archival process. Obtaining Sign-Offs from stakeholders, closing contracts and agreements

ii. **Technological Resources :**

- **Monitoring and Logging Tools:** Splunk for monitoring application performance and logging.

Critical Dependencies:

1. **Technological Dependencies:** Technological Dependencies include support for frameworks, libraries and cloud hosting platforms because, if support is stopped by vendor then it can be a risk for the project.
2. **Human Resource Dependencies:** Resources are vital for the project from the start till the end. Attrition rate of resources need to be reduced or minimized, preferably zero, throughout the project. Because, adding new team members mid way can cause a delay in producing the deliverables of the project.

Solution Proposal

- ☑ Users need to register on the site with basic information such as First Name, Last Name, Date of Birth, Location, required username, and password. After successful registration, users can log in and commence learning or teaching.
- ☑ Upon the user's first login, they are required to specify their skills or domain interests. It is mandatory to select at least 3 skills that greatly interest them. In case the user is unsure about their interests, an option is available for them to take a simple 15-question MCQ test to identify their interests. The questions will inquire about their day-to-day routines, recent exciting activities, happiest moments, what motivates them daily, and more.
- ☑ Our AI model will then help them determine their interests. Users have the provision to accept or deny the AI results. Once the user has entered their interests, upon login, they can view various events based on their interests in and around their location.

- With the range of interesting events, users can register for events and contact hosts using the available chat option. They can access event descriptions, dates, and reviews before registration. Upon registration, they will receive a confirmation email from the host. Furthermore, users can cancel their event registration at any time without incurring any penalties. Given that the focus is on skill exchange, walk-in sessions are also encouraged in most events. For events that require any specific tool access, registration is mandatory.

- To prioritize learners, the user is first logged into the website as a learner. Within the host tab, learners have the option to host events, thus becoming organizers.

- If a user plans to host an event, they must provide all details clearly for careful review and approval. This includes event name, category, location, available seats, timing, event description, target audience, and more. Each day's teaching goal is shared in advance by the teacher and can be viewed by learners.

- If the learner is hosting for the first time, they must undergo a basic rules and regulations test. This ensures that the host understands the purpose and the company policy. Furthermore, a general assessment is conducted for every host to determine the level of the skill they are going to teach. It is restricted to having only one host for the event. While there can be many individuals contributing to making the event happen, only one can register for the event and get it approved.

Key Features and Functionalities

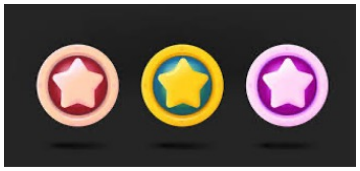
* To ensure that skills are exchanged mandatorily among the people, we have implemented a feature called the virtual currency system. In this currency technique, each user initially purchases 5 tokens. Whenever a user learns, they will sacrifice 1 token. However, users do not lose any tokens for hosting the token.

* The catch is, after 5 learnings, the user must teach at least once to earn the next 5 tokens. If all tokens are depleted, the user needs to purchase more. This system ensures that skills are exchanged within the community.

* The skill matching analysis divides the skill exchange provision into two categories: strongly matched and weakly matched people. In the strongly matched option, the learner and the teacher have something to teach and learn

from each other. Whereas the weakly matched option results in a single person either teaching or learning. Therefore, our skill-matching algorithm generally gives high priority to strongly matched people because we can indirectly achieve skill exchange through them.

* To keep the skill exchange motto alive in the long run, the system has a new feature named the progress passport system. Here, badges and ranks are given to boost the users' motivation. Certified badges/ranks from the hosts enable learners to become teachers soon. Additionally, winners of occasional challenges are awarded another high-level badge to represent them uniquely in the system.



Seasonal summits are arranged to facilitate meetings and introductions among nearby community members. Progress and entertaining elements from each community are demonstrated and discussed during these conventions. Workshops on trending topics, conducted by experts, are also part of these conventions to attract more participants. Seasonal events are organized, and skills popular in that season are offered by professionals to attract more people and foster a vibrant community.

Upon completion of a specific course, learners are encouraged to provide vigorous feedback about the course. It will roughly take 30 minutes to share every detail of the course. Since the entire system operates based on feedback, it is mandatory and the responsibility of each learner to provide it properly so that it will be useful for other learners.

Benefits and Impacts

* Users can spend their time meaningfully by learning new skills and sharing the skills they possess. They can interact with other community members, making the community more vibrant and productive. This allows them to become self-reliant in basic skills needed in everyday life.

* Instead of leading a mundane lifestyle of using phones and watching TV shows, this ecosystem encourages people to be more engaged and energetic. Regularly learning new skills equips users with talent and motivates them to learn more. Badges and honours received can be used to showcase mastery in various skills, attracting more people when exchanging skills.

* Local businesses and organizations indirectly benefit from the growth of the vibrant community, as it promotes local businesses over online shopping. Community moderators and the development team take pride in establishing a robust community for the well-being of the future. Investors receive a specific profit margin as per the agreement, while sponsors get the opportunity to advertise their products.

* Our target audience, ranging from 18 to 65 years old, will be pleased with the developing community. The younger participants quickly gain awareness and equip themselves with various essential skills. The system offers them valuable characteristics and exposes them to a diverse age group, fostering respect and teaching them how to interact with people. This fosters a love for the community, inspiring them to innovate new projects for its betterment.

Risk Assessment and Mitigation

The objective of the risk assessment and mitigation plan is to identify, assess, and address potential challenges and uncertainties associated with the project. Implementing of effective risk management strategies will help in minimizing the impact of identified risks such as project timelines, budget, resources, technology, and quality. Also, this helps in increasing the likelihood of project success and achieving our goals of creating a thriving community skill exchange platform.

1. Risk Identification

a. **Budget Risks**

i. **Unforeseen Expenses:** These expenses come up during the project which were not accounted for in the initial budget. These expenses may stem from various sources and can significantly impact the project's financial health if not properly managed. Expenses such as infrastructure costs, marketing expenses, staffing costs and technology upgrades come under unforeseen expenses.

ii. **Vendor Cost Escalation: I**dentifying the risk of vendor cost escalation involves analyzing historical procurement data, monitoring vendor performance closely, and reviewing contracts for unfavourable terms.

b. **Time/Scheduling Risks**

i. **Development Delays:** Promising of unrealistic deliverables, misunderstanding of project requirements and staffing shortages may result in slowing down the development of project and unnecessary rework of certain tasks , which will result in prolonging the overall project timeline.

c. **Resource Risks**

i. **Attrition:** Several key members of the development team, including the lead developer and a senior designer are very necessary and expensive to replace. If they decide to leave the project unexpectedly due to better job offers or personal reasons, it will affect the project's budget, deadlines, and many other important factors.

ii. **Resource Availability**: The availability of the staff may tamper the productivity due to the cultural and time-zone differences and remote work challenges. As this would lead to coordination difficulties and issues related to remote work setups impacting productivity.

iii. **Language Barrier:** The language problem presents a significant resource risk within the project, potentially hindering effective communication and collaboration among team members.

d. **Technology Risks**

i. **Compatibility Issues:** When developing an app across computers, tablets, and phones, as well as supporting different operating systems, could involve inconsistencies in user interface (UI) and user experience (UX) across various devices and platforms.

ii. **Security Breaches:** Insufficient measures taken in the platform's security may lead to compromising of user's data which will develop trust issues and damage control is nearly impossible for such risk.

iii. **Technological Advancements:** Enhancements in technology can adversely impact the project through increased complexity, leading to compatibility issues and integration challenges. As technology evolves rapidly, the project may face difficulties in keeping up with the latest advancements, potentially causing disruptions, delays, or unexpected expenses in development and maintenance processes.

e. **Quality Risks**

i. **User Interface Deficiencies:** User interface deficiencies could negatively affect the project by leading to user frustration, confusion, and dissatisfaction. This can result in decreased user engagement. Additionally poor UI may hinder users from efficiently navigating through the application, leading to decreased productivity, and causing them to seek alternatives. Overall, user interface deficiencies can undermine the application's usability, reputation, and success in the market.

ii. **Low Maintenance:** Inadequate customer support, neglecting regular maintenance of infrastructure and other artifacts of the project may lead to depletion of the quality of the application.

f. **Legal and Compliance Risks**

i. **Regulatory Adaptations:** Failing to comply with regulatory requirements can lead to legal consequences, reputational damage, and operational disruptions for the project. Non-compliance may result in fines, lawsuits, and loss of market access, impacting profitability and sustainability. Additionally, regulatory violations can erode trust among stakeholders and the public, further escalating the negative consequences.

2. Risk Impact Analysis

Managing risks in a project involves conducting risk analysis to assess the potential impact and likelihood of each risk occurring. This analysis helps prioritize risks, placing those with high probability and high impact at the top of the list, while risks with low impact and low probability are placed at the bottom. By organizing risks in this manner, the project manager can systematically address them, ensuring methods beforehand to deal with any challenges that may arise.

Different risks manifest at various stages of the project lifecycle. For example, issues with product quality may emerge during the design phase, necessitating rework and potentially affecting the project schedule. Similarly, unexpected defects discovered during testing may exceed the allocated time and budget for resolution. Additionally, unforeseen circumstances such as team member illness may lead to assignment delays.

Project risks are dynamic and can arise at any point in the project. Therefore, the project risk matrix, which outlines risks and their impacts and probabilities, should be regularly reviewed, and updated. This allows the project manager to

assess current risks and take appropriate remedial actions to mitigate their effects.

Project Risk Matrix:

Risk Category	Risk	Probability	Impact
Resource Risk	Attrition	High	High
Technology Risk	Security Breaches	High	High
Resource Risk	Language Barrier	High	High
Budget Risk	Unforeseen Expenses	Medium	High
Resource Risk	Resource Availability	Medium	High
Budget Risk	Vendor Cost Escalation	Medium	High
Technology Risk	Technological Advancements	Medium	High
Quality Risk	Low Maintenance	Low	High
Legal and Compliance Risk	Regulatory Adaptions	Low	High
Time/Scheduling Risk	Development Delays	Medium	Medium
Technology Risk	Compatibility Issues	Medium	Medium
Quality Risk	User Interface Deficiencies	Low	Medium

Risk Mitigation Strategies:

Risk mitigation involves addressing potential risks to minimize their impact on project objectives. It includes strategies such as contingency planning, implementing preventive measures, transferring risk to third parties, and developing response plans. Effective risk mitigation ensures proactive management of uncertainties, enhances project resilience, and increases the likelihood of project success

- **Unforeseen Expenses (Budget Risk):** Implement a contingency reserve in the budget to account for unexpected expenses. Conduct regular financial reviews to identify and address potential cost overruns. Prioritize expenses based on project priorities to allocate resources effectively.

- **Vendor Cost Escalation (Budget Risk):** Establishing clear terms with vendors upfront, including pricing structures and escalation clauses, and regularly monitor vendor performance and market conditions to identify and address potential cost increases promptly. One effective method to mitigate vendor cost escalation is to negotiate long-term contracts with fixed pricing.

- **Development Delays (Time/Scheduling Risk):** Adopt agile project management methodologies to enhance flexibility and responsiveness to changing requirements. Break down project tasks into smaller, manageable increments to facilitate incremental progress and mitigate the impact of delays. Implement robust project tracking and monitoring systems to identify potential delays early and take corrective actions promptly.

- **Attrition (Resource Risk):** Develop succession plans to mitigate the impact of key personnel departures. Cross-train team members to ensure redundancy and mitigate the risk of knowledge loss. Maintain open communication channels with team members to address concerns and foster a positive work environment.

- **Resource Availability (Resource Risk):** Introducing cultural sensitivity training to create an atmosphere of understanding and collaboration among diverse team members. Implementing collaboration tools that accommodate different time zones and providing flexible work arrangements can help overcome coordination difficulties and enhance productivity in remote work environments. Regular virtual meetings and team-building activities can also facilitate rapport-building and strengthen relationships among distributed teams.

- **Compatibility Issues (Technology Risk):** To address compatibility issues, perform comprehensive testing across various devices and operating systems during development, ensuring uniform functionality and user experience. Employ responsive design principles and prioritize development frameworks that are platform-friendly to reduce compatibility issues and improve interoperability.

- **Security Breaches (Technical Risk):** Implement a comprehensive cybersecurity strategy, including regular security audits, employee training, and incident response plans. Encrypt sensitive data and implement multi-factor authentication to enhance data security. Collaborate with cybersecurity experts and leverage threat intelligence to identify and mitigate potential security risks.

- **User Interface Deficiencies (Quality Risk):** Conduct user testing and feedback sessions to identify and address usability issues and improve the overall user experience. Invest in user-centric design principles and best practices to create intuitive and engaging interfaces. Regularly update and iterate on UI/UX designs based on user feedback and industry trends.

- **Low Maintenance (Quality Risk):** Investing on time on creating a robust maintenance schedule to address technical debt, software updates, and infrastructure maintenance. Allocate resources for ongoing support, bug fixes, and feature enhancements to ensure the long-term sustainability and quality of the project. Implement automated monitoring and alerting systems to proactively identify and address maintenance issues before they escalate.

- **Language Barrier (Resource Risk):** Establishing effective and clear communication guidelines and protocols to ensure that all team members understand expectations regarding language proficiency, communication channels, and etiquette. Utilize multilingual documentation, communication tools, and project management platforms to accommodate diverse language needs within the team. Invest in translation services or tools to facilitate effective communication and collaboration among team members with different language backgrounds.

- **Regulatory Adaptions (Legal and Compliance Risk):** Having a regulated research about relevant regulatory changes, assessing their impact on operations, and implementing of robust compliance controls and procedures will reduce the impact or the affect in high numbers. Training programs which will help in understanding the compliance obligations will be implemented, and simultaneously while regular reviews and audits will be conducted to identify areas for improvement. Also having a legal counsel and good adaptation plans is essential for effectively managing regulatory compliance in response to evolving regulatory requirements.

- **Technological Advancements (Technology Risk):** Technological advancements can be tackled by organizations with involving SME (Subject Matter Experts) who continuously monitor emerging technologies relevant to the project, assess their potential impact on operations, and implement strategies to adapt and leverage these advancements effectively. Necessary training programs should be provided to ensure employees, guided by SME, can navigate new technologies and cybersecurity risks.

Budget

All figures are estimates and subject to change based on project requirements and market conditions.Contingency funds are included to account for unexpected expenses or changes in project scope.Regular budget reviews and adjustments are recommended throughout the project lifecycle.

Following careful consideration, our projected capital requirement for development and maintenance over this period is estimated at **$250,000**. This budget allocation reflects our commitment to strategic planning and prudent financial management in ensuring the success and sustainability of our venture.

Dividing the total budget into key sections, we've outlined the following allocations: Development Costs, Marketing Expenses, Operations. Here's a detailed breakdown of the cost allocation across each section:

Breakdown

Development cost	$235,000
Marketing Expenses	$15,000
Operational Costs	$53,000
Total	$253,000

Development cost

Determining the overall development cost is crucial because it's likely to be the most expensive aspect, and any miscalculation could seriously impact our budget. Development cost includes the cost of outsourcing the development work and salaries of the onshore team. To estimate the cost of the development we performed **Function Point Analysis**, which is also a process third party development companies use to estimate cost. Our finding were:

Functions	Points
User Registration/Login	10
Profile Creation/Edit	15

Skill Matching Algorithm	90
Search for events with filters	5
Browse details of interested event/ Event Page	60
Registration for events (Real-time)	80
Host Tab for hosting and editing events	10
Energy Token System	50
Progress Passport System	80
Notifications	4
Admin Panel	20
Total Points	424

Taking into account the previous projects completed by a specific third-party company, we've concluded that allocating $180 per point is equitable. To be on the side of caution, we've added an extra 11%, bringing the total cost of developing each point to $200.

So, to finally calculate the Software Development Team Salaries:

= 200 x (425)

= 84800

Rounding off to the nearest thousandth = **$85,000**

Software Development Team Salaries will cost **$150,000**

Software Development	$85,000
Team Salaries (incl. Offshore)	$150,000
Total	$235,000

Marketing Expenses

This allocation underscores the strategic importance of marketing initiatives in establishing and expanding the platform's presence, attracting users, and fostering engagement, thus contributing to its overall success and sustainability.

Digital Marketing Campaigns	$10,000
Content Creation	$5,000
Total	$15,000

Operational Costs

This allocation reflects the project's commitment to ensuring a supportive work environment and adequate resources to facilitate seamless operations throughout its lifecycle.

Software Maintenance	$20,000
Cloud Services	$10,000
Software Licenses	$3,000
Contingency	$20,000
Total	$53,000