|  |
| --- |
| **Project Title:**  **AI Study Buddy** |
| **Project Summary:**  Problem Statement: Most study applications today provide only partial support for learners, focusing on narrow functions such as scheduling, flashcards, or digital note storage. This fragmented approach disrupts learning continuity and forces students to switch across multiple platforms, which increases cognitive load and reduces efficiency. Furthermore, few tools adapt dynamically to learner needs, such as detecting knowledge gaps or offering motivation to maintain consistent study habits. Research in self-regulated learning emphasizes that effective learning requires planning, monitoring, and reflection (Zimmerman, 2002). Current tools often neglect this holistic cycle, highlighting the need for an integrated system that addresses both cognitive and behavioral dimensions of study.  Solution Vision: The **AI Study Buddy** is a web-based platform designed to unify key learning supports into a single, adaptive system. Features such as note uploading, quiz generation, and knowledge gap detection align with constructivist principles, enabling learners to build and refine knowledge through active engagement. The Pomodoro timer supports self-regulated learning strategies by structuring focus and rest intervals, while gamification draws on motivational theory to sustain engagement. Multimodal input (text, voice, images) caters to diverse learning styles, consistent with Gardner’s multiple intelligences framework. Peer matching promotes collaborative learning, allowing students to co-construct knowledge and provide mutual support. For instance, a student could upload lecture notes, receive AI-generated summaries and quizzes, use the Pomodoro system for practice, and collaborate with peers on difficult topics. By integrating these features, AI Study Buddy aims to deliver not just functionality but an evidence-based, personalized, and engaging learning experience. |
| **Project Development:**  The first **Minimal Viable Product (MVP)** for the AI Study Buddy will focus on three essential features: note uploading, automatic quiz generation, and a Pomodoro timer. This limited scope allows the team to test the system’s core functionality while validating whether students find the platform engaging and useful for structuring their study routines. From deploying this MVP, we aim to learn how effectively users interact with the interface, whether the AI-generated outputs (summaries and quizzes) meet user expectations, and to what extent the Pomodoro feature supports focus and productivity. Key measurements will include system usability (task completion rates, time on task), performance (latency in generating quizzes or summaries), and user satisfaction (collected through structured feedback).  Development will follow the **SCRUM methodology**, with two-week sprints, daily stand-ups, and a rotating Scrum Master to ensure equitable responsibility. Each sprint will deliver incremental features, reviewed in sprint retrospectives to capture lessons learned and adjust the backlog accordingly.  The system will adopt modular cloud-native architecture. The **front-end** will be built as a React single-page application, hosted on CDN for scalability. The **back end** will consist of Node.js/Express microservices for user authentication, file handling, gamification, and business logic, complemented by Python/Flask services for AI-driven tasks such as summarization, OCR, and quiz generation. **Data sources** will include PostgreSQL (for structured data and embeddings), Redis (for caching and sessions), and AWS S3 (for note and media storage). Observability will be achieved through Prometheus, Grafana, and the ELK stack, while orchestration will be managed with Kubernetes.  This MVP establishes the technical foundation, validates user engagement, and provides actionable insights to guide subsequent development of advanced features such as peer matching, knowledge gap detection, and multimodal input. |
| **Evaluation:**  The evaluation of the system will begin with the establishment of a **beta testing pool** consisting of a reasonably sized and demographically diverse group of users. Diversity in age, technical expertise, and user background will allow the team to gather feedback that reflects a broad range of perspectives and potential use cases. Testers will be asked to interact with the system under realistic conditions, and their experiences will be documented through surveys, usability tasks, and structured feedback forms.  Issues identified by beta testers, whether technical bugs or usability concerns, will be prioritized and systematically addressed. Particular attention will be given to recurring problems or barriers that hinder effective system use. This stage will also provide insights into the strengths of the system, such as ease of use, efficiency, or innovative features, which can then be reinforced in future versions.  The evaluation process will directly inform further development by highlighting areas requiring improvement, guiding design refinements, and shaping future iterations of the system. Incorporating user-centered feedback ensures that the product evolves beyond theoretical design and aligns closely with end-user expectations, ultimately enhancing functionality, reliability, and overall user satisfaction. |
| **Project Management:**  The project will be coordinated using **Notion** as the primary project management tool. This platform allows the team to maintain a shared timeline, allocate tasks, and clearly track individual responsibilities. By using collaborative boards and progress trackers, all members can remain aligned on deliverables and deadlines, reducing the likelihood of missed tasks or miscommunication.  To ensure timely completion, the group has agreed to set internal submission deadlines one day prior to official deadlines. This practice provides a buffer to review work collectively, resolve outstanding issues, and ensure quality before final submission. The project will run in structured phases, beginning with requirements gathering and design, followed by development, testing, and integration, with weekly reviews to assess progress.  Success is defined in multiple layers. At a satisfactory level, the project will deliver a functional product that operates as intended with minimal system errors. A higher measure of success is achieved if the product demonstrates originality in concept and robustness in design while providing seamless user experience. Ultimately, success will be judged not only on technical functionality but also on the degree of innovation and the level of satisfaction it generates for end users. |

|  |
| --- |
| **Team Name:** AI Study Buddy |
| **Team Members:**   |  |  |  | | --- | --- | --- | | **Name** | **Student Number** | **Contact Number** | | Yurii |  |  | | Rumaysa |  |  | | Lorenzo |  |  | | Anika Siddiqui Mayesha | D24125187 | 353-830759696 | |  |  |  | |
| **Team Meetings:**  Team meetings are designed to balance consistency, flexibility, and respect for individual boundaries. The schedule includes two online meetings each week, one in-person meeting every Friday, and daily check-ins via text to ensure ongoing communication. Attendance is encouraged but not mandatory; if one member is absent, the meeting proceeds, but if two members are unavailable, the session is rescheduled where possible. If rescheduling is not feasible, the meeting continues with the remaining two members as a last resort.  Decision-making follows a majority rule to ensure fairness, and the team uses a rotating Scrum Master system to manage discussions and maintain structure. The rotation order is Yurii, Rumaysa, Lorenzo, Anika, and then back to Yurii, with each member serving for one week. This approach promotes shared responsibility and prevents overreliance on a single individual.  Meetings focus strictly on project-related issues, scheduling, and conflict resolution. Personal matters, such as absences or individual beliefs, are not subject to questioning, and boundaries expressed by team members are respected without exception. Turn-taking during discussions is managed by the Scrum Master, ensuring each member has the opportunity to contribute without interruption. This structure supports inclusivity, productivity, and mutual respect in all team interactions. |
| **Team Conflict:**  Team members naturally bring different habits and communication styles, which can become challenges if not addressed carefully. To manage this, weekly in-person meetings were introduced to help us to understand how each member prefers to communicate. Tasks were then allocated according to individual strengths, experience, and working styles, which encouraged fairness and efficiency. Any unresolved issues are brought to these meetings, where members are encouraged to share their perspectives, supported by reasoning or evidence. This structured process helps keep discussion’s objective and reduces the risk of misunderstandings. Conflict is managed through open dialogue and collective problem-solving, with an emphasis on building agreement rather than competing interests. Importantly, no single person holds veto power; instead, decisions are made through majority agreement. By promoting open communication, shared responsibility, and respect among members, the team actively reduces the likelihood of conflict and creates an environment that supports cooperation and productivity. |