CEBU INSTITUTE OF TECHNOLOGY UNIVERSITY

COLLEGE OF COMPUTER STUDIES

Software Project Proposal

for

AudioScholar: Transforming Audio into Actionable Insights for Learners

AudioScholar: Transforming Audio into Actionable Insights for Learners

Proponent(s):

Biacolo, Math Lee L.
Terence, John Duterte
Orlanes, John Nathan
Barrientos, Claive Justin
Alpez, Christian Brent

Adviser:

Jasmine A. Tulin

_____

Consultation Schedule:

TULIWED11001200

Date of Submission:

February 22, 2025

1. Executive Summary

1.1 Problem statement and proposed solution:

Traditional note-taking during lectures is inherently inefficient. Students struggle to simultaneously listen, understand, and meticulously write notes, often missing crucial information and creating disorganized records. AudioScholar is designed to transform this inefficient process into an efficient one. It is a smart, multi-user application that records lecture audio and uses AI-powered summarization to generate structured, actionable insights after the lecture. This allows students to focus on active listening during lectures and receive well-organized, efficient study materials afterwards.

1.2 Expected benefits and impact:

AudioScholar is expected to significantly enhance the learning experience by making lecture note-taking efficient and effective. By automating the summarization process and providing personalized learning resources, AudioScholar empowers students to review lecture content more effectively, leading to improved comprehension and academic performance. The application's offline recording capabilities ensure accessibility, and its freemium model makes efficient note-taking accessible to a wider range of students. Ultimately, AudioScholar aims to revolutionize lecture note-taking, making it an efficient and supportive tool for learning.

2. Background and Problem Statement

2.1 Description of the problem or gap: Inefficiency of Traditional Note-Taking

Traditional note-taking during lectures is fundamentally inefficient. Students are forced to divide their cognitive resources, attempting to simultaneously actively listen, comprehend complex information, and laboriously write notes. This multi-tasking is inherently inefficient, leading to a trade-off: students who prioritize detailed note-taking often sacrifice active listening and comprehension, while those who focus on listening end up with sparse and inefficient notes for later review. The fast pace and information density of lectures amplify this inefficiency, as students struggle to capture key information quickly enough, resulting in incomplete and disorganized notes that are inefficient to use for studying. This inefficiency is further compounded in large lecture settings, where student engagement can be lower, making the already inefficient process even less effective (Pohl, 2015).

Research underscores the inefficiency of relying solely on traditional methods. While Gur, Dilci, Coskun, and Delican (2013) demonstrate the cognitive benefits of note-taking in general for listening comprehension, the very act of manual note-taking during a lecture can be an inefficient bottleneck to learning. Wang (2021) highlights the value of active note-taking strategies for factual recall, but traditional methods often fall into passive transcription, which is an inefficient use of cognitive effort during lectures. Artz, Johnson, Robson, and Taengnoi (2020) emphasize that digital tools alone are not a solution, pointing to the need for effective note-taking processes. Traditional manual note-taking, even with digital tools, can remain an inefficient process if it doesn't address the core challenges of real-time capture and organization. KRIOU and BENSADIA (2020)'s study directly reveals student-reported difficulties that contribute to the inefficiency of traditional note-taking, such as "lack of concentration," "fast speed rate of speech," and poor "organization of notes," all highlighting aspects that make manual note-taking a laborious and inefficient task.

Students are often advised on note-taking techniques and provided with general tools, but these resources rarely address the core inefficiency of the traditional process itself – the real-time struggle to actively listen and simultaneously create effective notes during fast-paced lectures. Lecture recordings, while helpful for review, do not solve the problem of inefficient information capture during the lecture. Therefore, the central problem is the inherent inefficiency of traditional note-taking, which hinders active learning and effective study. AudioScholar aims to directly address this inefficiency by

providing a more streamlined and effective approach.

2.2 Limitations of existing solutions: Failing to Address Inefficient Note-Taking

Existing solutions, while attempting to address aspects of lecture learning, largely fail to tackle the core problem of inefficient traditional note-taking. They do not provide a truly efficient alternative to the manual, real-time note-taking process:

- **Manual Transcription Services:** These services are not real-time and do not make the during-lecture note-taking process more efficient. They simply shift the inefficient task of transcription to a later time, and still require manual summarization and organization, remaining an inefficient solution for lecture note capture.
- **Generic Voice Recorders:** These capture audio but do not make note-taking more efficient. Students are still burdened with the inefficient tasks of manually transcribing, summarizing, and organizing recordings. This maintains the inefficiency of the post-lecture processing workflow.
- **Online-Only Lecture Capture Systems:** These systems may record lectures, but they do not make the student's personal note-taking process more efficient, especially for in-person lectures. They are primarily review tools and do not offer an efficient way to capture and process information during the lecture itself.
- **Limited AI Integration:** Some applications offer basic AI features, but often do not focus on making the entire note-taking workflow more efficient, from lecture capture to study material generation. They may offer features, but not necessarily features that streamline the inefficient core process.
- **Lack of Offline Focus:** Solutions requiring constant internet are inefficient for students in environments with limited connectivity, hindering their ability to engage in efficient note-taking in all learning contexts.

2.3 Justification for the proposed solution:

AudioScholar is justified as a necessary and innovative solution because it directly addresses the inefficiency of traditional note-taking by providing a demonstrably more efficient alternative:

- **Efficient Automated Summarization:** AudioScholar automates the summarization process after the lecture, transforming recorded audio into structured notes efficiently. This eliminates

the inefficient manual summarization process, saving students significant time and effort.

- **Offline Recording for Efficiency:** Offline recording ensures students can efficiently capture lectures in any environment, removing the barrier of internet dependency and enabling efficient note-taking regardless of location.

- **Intelligent Post-Lecture Processing for Efficiency:** By leveraging AI APIs, AudioScholar efficiently processes recordings after the lecture to generate summaries and recommendations. This streamlines the post-lecture study process, making review and comprehension more efficient.

- **Contextual Enrichment for Enhanced Efficiency:** Optional PowerPoint integration enhances the efficiency of AI summarization by providing contextual information, leading to more accurate and relevant summaries, further improving study efficiency.

- **User-Centric Design for Efficient Workflow:** Features like optional cloud sync and tiered access are designed to support an efficient and personalized note-taking workflow, allowing users to optimize their process for maximum efficiency.

- **Freemium Model for Efficient Accessibility:** The freemium model aims to make efficient note-taking tools accessible to a wider student population, promoting broader access to more efficient learning methods.

By directly targeting the inefficiency of traditional note-taking and offering a comprehensive, intelligent, and accessible approach designed for efficiency at every stage, AudioScholar is a justified and valuable project. It aims to transform lecture note-taking from an inefficient burden into an efficient and supportive tool for learning, empowering students to study more effectively and achieve better academic outcomes through a more efficient learning process.

3. Project Objectives

3.1 Main Objectives (SMART Goals)

- **Objective 1: Enable Efficient Lecture Recording and Upload:** Develop a system that allows users to easily and reliably record lectures via a mobile application (even offline) and upload pre-recorded audio files through both the mobile and web interfaces. To be completed by Implementation Increment 1 (end of Week 2).
    - **Specific:** Focuses on lecture recording and audio upload functionalities.
    - **Measurable:** Can be measured by the successful implementation and user adoption of recording and upload features, and the types of formats supported.
    - **Achievable:** Feasible within the project scope, as detailed functional requirements exist.
    - **Relevant:** Directly addresses the initial stage of efficient note-taking – capturing the audio.
    - **Time-bound:** To be completed by Implementation Increment 1 (end of Week 2).
- **Objective 2: Automate Audio Processing and Generate Structured Lecture Summaries:** Implement a server-side module that automatically processes recorded or uploaded audio using AI (Google Gemini AI API) to generate structured and concise summaries of lecture content. To be completed by Implementation Increment 2 (end of Week 4).
    - **Specific:** Focuses on automated audio processing and summarization using AI.
    - **Measurable:** Can be measured by the successful integration of the AI API, the quality of the generated summaries, and the time taken for processing.
    - **Achievable:** Relies on the capabilities of the Google Gemini AI API, which is referenced.
    - **Relevant:** Directly addresses the inefficiency of manual note-taking and post-lecture processing.
    - **Time-bound:** To be completed by Implementation Increment 2 (end of Week 4).
- **Objective 3: Provide Personalized Learning Material Recommendations:** Develop a server-side

module that analyzes lecture content (summaries) and generates personalized recommendations for supplementary learning materials, initially focusing on YouTube resources. To be completed by Implementation Increment 3 (end of Week 6).

  - **Specific:** Focuses on generating learning material recommendations from YouTube based on lecture content.
  - **Measurable:** Can be measured by the successful integration with the YouTube Data API, the relevance of the recommendations, and user engagement with the recommendations.
  - **Achievable:** Relies on the capabilities of the YouTube Data API, which is referenced.
  - **Relevant:** Enhances learning comprehension by providing additional resources.
  - **Time-bound:** To be completed by Implementation Increment 3 (end of Week 6).

- **Objective 4: Ensure Secure User Authentication and Account Management:** Implement a robust system for user authentication (via Google OAuth 2.0, GitHub OAuth 2.0, and email/password) and account management within the mobile application. To be completed by Implementation Increment 1 (end of Week 2).
  - **Specific:** Focuses on secure user authentication and basic account management features.
  - **Measurable:** Can be measured by the successful implementation of the different authentication methods, the security of user data, and user satisfaction with the login process.
  - **Achievable:** Relies on established authentication protocols and Firebase services.
  - **Relevant:** Essential for providing personalized services and securing user data.
  - **Time-bound:** To be completed by Implementation Increment 1 (end of Week 2).
- **Objective 5: Enable Optional Cloud Synchronization of User Data:** Provide users with the option to securely synchronize their recordings and summaries to the cloud (Firebase Storage) for data backup and accessibility across devices. To be completed by Implementation Increment 4 (end of Week 8).
  - **Specific:** Focuses on optional cloud synchronization of recordings and summaries using

Firebase Storage.

- **Measurable:** Can be measured by the successful implementation of the synchronization functionality, data consistency across devices, and user configuration of sync settings.
- **Achievable:** Relies on the capabilities of Firebase Storage.
- **Relevant:** Enhances data security and user convenience.
- **Time-bound:** To be completed by Implementation Increment 4 (end of Week 8).

- **Objective 6: Integrate PowerPoint Upload for Enhanced Summarization Context:** Allow users to optionally upload lecture PowerPoint presentations via the mobile application to provide contextual information that enhances the accuracy and relevance of AI-generated summaries. To be completed by Implementation Increment 3 (end of Week 6).
  - **Specific:** Focuses on PowerPoint integration to improve AI summarization.
  - **Measurable:** Can be measured by the successful implementation of the upload and association process and the perceived improvement in summary quality when PowerPoint is integrated.
  - **Achievable:** Feasible within the project scope, as functional requirements are outlined.
  - **Relevant:** Aims to further improve the efficiency and quality of the generated study materials.
  - **Time-bound:** To be completed by Implementation Increment 3 (end of Week 6).
- **Objective 7: Develop a Web Interface for Content Access and Management:** Create a web interface that allows users to access their recordings, review generated summaries, and explore learning material recommendations from a web browser, also including the ability to upload pre-recorded audio. To be completed progressively across Increments 2, 3, and 4 (Weeks 3-8).
  - **Specific:** Focuses on the functionality of the web interface for accessing and managing content.
  - **Measurable:** Can be measured by the successful development and deployment of the web interface and the usability of its features.

- **Achievable:** Involves web development technologies like ReactJS.
- **Relevant:** Provides broader accessibility and convenience for users.
- **Time-bound:** To be completed progressively across Increments 2, 3, and 4 (Weeks 3-8).

- **Objective 8: Implement a Freemium Model for Feature Access Control:** Implement a freemium service model that controls access to different features and functionalities (e.g., background recording, recommendation sources) based on the user's authentication status. To be completed by Implementation Increment 4 (end of Week 8).
  - **Specific:** Focuses on the implementation of the freemium model and feature access control.
  - **Measurable:** Can be measured by the successful implementation of tiered access and the correct restriction of features for unauthenticated and free users.
  - **Achievable:** Involves logic within the application to check user status.
  - **Relevant:** Supports the project's business model and accessibility goals.
  - **Time-bound:** To be completed by Implementation Increment 4 (end of Week 8).

3.2. Specific Objectives (Key Deliverables)

3.2.1. Functional Requirements

- **FR1: Lecture Recording (Mobile & Web Upload):** The system shall allow users to record lectures via the mobile application, even in offline mode, storing audio files locally. The web interface shall allow users to upload pre-recorded audio files.
- **FR2: Audio Processing and Summarization:** The system shall automatically process recorded or uploaded audio to generate structured summaries using AI APIs (Google Gemini AI API).
- **FR3: Learning Material Recommendation:** The system shall provide personalized learning material recommendations based on lecture content, leveraging AI APIs and potentially integrating with YouTube and other educational resources.
- **FR4: User Authentication and Account Management (Mobile):** The mobile application shall

handle user authentication via Google OAuth 2.0, GitHub OAuth 2.0, and email/password, and manage user accounts and profiles.

- **FR5: Cloud Synchronization (Optional):** The system shall provide optional cloud synchronization of recordings and summaries to Firebase Storage, with user-configurable automatic or manual synchronization settings.
- **FR6: PowerPoint Integration (Optional):** The system shall allow users to upload lecture PowerPoint presentations to provide contextual information for AI processing and summarization.
- **FR7: Multi-User Support:** The system shall support multiple users with individual accounts and data privacy.
- **FR8: Web Interface Access (Viewing & Upload):** The system shall provide a web interface for users to access recordings, summaries, and recommendations via a web browser and upload audio recordings.
- **FR9: Freemium Model Implementation:** The system shall implement the freemium model with tiered access based on user login status and subscription level, as defined in the business model.

3.2.2. Non-Functional Requirements

- **NFR1: Performance:** The system shall provide summarization and recommendation generation in a timely manner after lecture recording or upload, with minimal latency.
- **NFR2: Efficiency:** The mobile application shall be energy-efficient to minimize battery drain during lecture recording.
- **NFR3: Security:** The system shall ensure secure storage and transmission of user data and recordings, utilizing Firebase security features and secure authentication protocols.
- **NFR4: Stability:** The system shall be stable and reliable, minimizing crashes and errors during recording, processing, and synchronization.
- **NFR5: Usability:** The system shall be user-friendly and intuitive, with a clear and easy-to-

navigate interface for both mobile and web applications.

- **NFR6: Scalability:** The server-side infrastructure shall be scalable to accommodate a growing number of users and increasing data volume.
- **NFR7: Offline Capability:** The mobile application shall function effectively in offline environments for recording and local storage.

3.2.3. Documentation

- **3.2.3.1.1. Software Requirements Specification (SRS):** A detailed document outlining all functional and non-functional requirements of the AudioScholar system.
- **3.2.3.1.2. Software Design Description (SDD):** A comprehensive document describing the system architecture, module design, database schema, and technology choices.
- **3.2.3.1.3. Software Project Management Plan (SPMP):** A plan outlining the project timeline, milestones, resource allocation, team roles, and communication strategies.
- **3.2.3.1.4. Software Test Document:** A document detailing test cases, test scenarios, testing methodologies, and test results for all system components.

3.2.4. Usability Testing and User Survey

- Conduct usability testing with target users (students) to evaluate the system's ease of use, effectiveness, and user satisfaction.
- Collect user feedback through surveys and questionnaires to identify areas for improvement and refine the system based on real user experiences.

3.2.5. Final Paper (ACM Format)

- Prepare a final paper summarizing the project, including problem definition, proposed solution, implementation details, evaluation results, and conclusions, adhering to the ACM (Association for Computing Machinery) format for academic publications.

4. Scope and Limitations

4.1. Features and functionalities included:

AudioScholar will include the following key features and functionalities:

- **Mobile Application (Android):**
    - Offline and online lecture recording.
    - Local storage of audio recordings.
    - Optional cloud synchronization (manual/automatic).
    - AI-powered summarization of lectures (post-lecture).
    - Display of structured summaries.
    - Learning material recommendations (YouTube initially, expandable later).
    - User authentication and account management.
    - Optional PowerPoint upload for contextual enrichment.
    - Freemium model implementation (local mode, free logged-in, premium logged-in).
- **Web Interface:**
    - Access to recordings and summaries.
    - Access to learning material recommendations.
    - Audio recording upload (no direct recording via web).
    - User account viewing (management primarily via mobile app).
- **Server-Side Processing:**
    - Audio processing and summarization using AI APIs (Google Gemini AI API).

- Recommendation generation using AI APIs.
- Cloud storage management (Firebase Storage).
- User authentication and authorization (Firebase Authentication, Google OAuth 2.0, GitHub OAuth 2.0).

4.2. Constraints and exclusions:

The project will operate under the following constraints and exclusions:

- **Timeline**: The project development timeframe is constrained to 8 weeks within the academic semester.
- **Resources:** Development resources are limited to the student team and available open-source tools and cloud services (Firebase free tier, potentially limited AI API usage within budget).
- **Platform Limitation:** The initial mobile application development will focus on the Android platform using Kotlin. iOS development is excluded from the initial scope but could be considered for future expansion.
- **AI API Dependency:** The project relies on external AI APIs (Google Gemini AI API). Changes in API availability, pricing, or functionality could impact the project.
- **Language Support:** Initially, the system will primarily focus on English language lectures. Support for other languages may be considered for future development but is excluded from the current scope.
- **Background Recording Limitation (Free Tier):** Background recording on the mobile application will be disabled for free logged-in users as part of the freemium model implementation.
- **Recommendation Engine Scope (Initial Phase):** The initial recommendation engine for free users will be limited to YouTube as the primary source of learning materials. Expansion to other sources is planned for premium users and future development.
- **No Real-time Transcription:** The system will not provide real-time transcription of lectures.

Summarization and note generation will occur after the lecture recording is completed.

- **Web Interface Recording Exclusion:** The web interface will not support direct audio recording. Users can upload pre-recorded audio files via the web interface.

5. Proposed Solution and Methodology

5.1. Overview of the software:

AudioScholar is a client-server application designed to transform lecture audio into actionable insights for students, addressing the inefficiency of traditional note-taking. The mobile application (Android) allows students to record lectures, even offline. Recordings are initially stored locally. Users can optionally synchronize recordings to the cloud (Firebase Storage). The server-side (Spring Boot) processes audio using AI APIs (Google Gemini AI API) to generate summaries and recommendations after the lecture. A web interface (ReactJS) provides browser-based access to recordings, summaries, and recommendations, and allows for audio upload. User authentication is managed via Firebase Authentication and OAuth 2.0. The system operates on a freemium model, offering varying levels of access and features based on user login status and subscription.

5.2. Technologies and platforms:

- **Mobile Application (Client-Side):** Android (Kotlin)
- **Web Interface (Client-Side):** ReactJS
- **Server-Side Application:** Spring Boot (Java)
- **Database and Backend:** Firebase (Firestore/Realtime Database, Storage, Authentication)
- **AI APIs:** Google Gemini AI API (for summarization and recommendations)
- **Authentication APIs:** Google OAuth 2.0 API, GitHub OAuth 2.0 API

5.3. Development approach:

The project will utilize an Agile development methodology, specifically Scrum, to ensure flexibility, iterative development, and continuous improvement. This approach is well-suited for a Capstone project due to its adaptability to changing requirements and its emphasis on collaboration and iterative progress within the 8-week development timeframe.

- **Sprints:** The project will be divided into short sprints (e.g., 2-week sprints) with clearly defined

goals and deliverables for each sprint.

- **Daily Stand-ups:** Daily stand-up meetings will be conducted to track progress, identify roadblocks, and ensure team alignment.
- **Sprint Planning and Review:** Sprint planning meetings will be held to plan sprint goals and tasks, and sprint review meetings will be conducted to demonstrate completed work and gather feedback.
- **Product Backlog:** A product backlog will be maintained to manage user stories, features, and tasks.
- **Iterative Development:** Development will be iterative, with each sprint building upon previous sprints and incorporating feedback.
- **Continuous Testing:** Testing will be integrated throughout the development process to ensure quality and identify issues early.

Agile methodology allows for flexibility in adapting to unforeseen challenges and incorporating user feedback throughout the project lifecycle, which is crucial for a Capstone project with evolving requirements and learning objectives within the limited timeframe.

6. Target Users, Customers, Beneficiaries, and Partners

6.1. Intended users:

The primary target users for AudioScholar are college and university students across all academic disciplines. Specifically, students who attend in-person lectures and seek to improve their note-taking efficiency, lecture comprehension, and study habits are the intended users. The application is designed to be beneficial for students of all learning styles and academic backgrounds seeking more efficient learning tools.

6.2. How the solution benefits them:

AudioScholar provides significant benefits to its target users by addressing the inefficiency of traditional note-taking:

- **Improved Note-Taking Efficiency:** Automates post-lecture note-taking, freeing students from inefficient manual transcription and allowing them to focus on understanding lecture content during class.
- **Enhanced Lecture Comprehension:** Provides structured summaries and recommendations, facilitating efficient review and deeper understanding of lecture material.
- **Reduced Cognitive Overload:** Reduces the mental burden of simultaneous listening and inefficient manual note-taking, leading to a less stressful and more engaging learning experience.
- **Offline Accessibility:** Ensures usability in diverse learning environments, including those with limited internet connectivity, for efficient recording.
- **Personalized Learning Resources:** Offers tailored learning material recommendations, expanding learning beyond the lecture hall and enhancing study efficiency.
- **Time Savings:** Reduces the time spent on inefficient manual note review and organization, freeing up time for other academic activities.
- **Improved Academic Performance:** Ultimately aims to improve student comprehension and

academic performance through more efficient learning and study habits.

6.3. Potential partners or stakeholders involved:

- **Educational Institutions (Universities and Colleges):** Potential partners for pilot programs, integration with existing learning management systems (LMS), content licensing, and distribution to students. Institutions could benefit from improved student learning outcomes and potentially offer AudioScholar as a value-added service to enhance learning efficiency.
- **Faculty and Lecturers:** Stakeholders who can provide feedback on the application's effectiveness and potential integration into teaching practices. Faculty could also contribute to content recommendations and curriculum alignment to further improve learning efficiency.
- **Educational Content Providers:** Potential partners for licensing educational content and integrating it into the recommendation engine for premium users, expanding efficient learning resources.
- **Student Organizations and Associations:** Potential partners for promoting AudioScholar to student communities and gathering user feedback to ensure the tool effectively addresses student needs for efficient learning.

7. Technical Requirements

7.1. Hardware and software needs:

- **Development Environment:**
    - Developer workstations with sufficient processing power and memory for Android Studio, ReactJS development, and Spring Boot development.
    - Android mobile devices for testing and development.
    - Internet access for development, API integration, and cloud service access.
- **Server Infrastructure:**
    - Firebase project setup (free tier initially, potentially scaling to paid tiers as needed).
    - Access to Google Gemini AI API (API keys and usage quota).
    - Reliable internet connectivity for server operations.
- **End-User Devices:**
    - Android mobile devices (smartphones, tablets) for mobile application usage.
    - Computers with web browsers for web interface access.

7.2. Security and infrastructure considerations:

- **Data Security:**
    - Utilize Firebase security rules to protect user data and recordings stored in Firestore and Storage.
    - Implement secure authentication protocols using Firebase Authentication and OAuth 2.0 to protect user accounts.
    - Ensure data encryption in transit and at rest where applicable within Firebase services.

- Adhere to data privacy best practices and relevant regulations regarding student data.

- **Infrastructure Scalability:**
  - Leverage Firebase's scalable infrastructure to accommodate potential user growth and increasing data volume.
  - Monitor server performance and resource utilization to ensure system stability and responsiveness.
  - Design the server-side application (Spring Boot) for scalability and efficient resource management.
- **Offline Functionality Security:**
  - Ensure that locally stored recordings on mobile devices are handled securely and consider potential security implications of offline data storage.
- **API Security:**
  - Securely manage API keys for Google Gemini AI API and authentication APIs.
  - Implement rate limiting and error handling for API calls to prevent abuse and ensure system stability.

8. Evaluation and Success Metrics

8.1. Key Performance Indicators (KPIs):

- **KPI 1: User Adoption Rate:** Track the number of active users (daily/weekly/monthly) to measure user adoption and engagement with AudioScholar as an efficient note-taking tool.
- **KPI 2: Note Review Time Reduction:** Measure the percentage reduction in average note review time reported by users through surveys and in-app time-tracking features, indicating improved efficiency. (Target: 60% reduction - Objective 1).
- **KPI 3: Lecture Comprehension Improvement:** Assess the percentage increase in student self-reported lecture comprehension through pre- and post-usage surveys, reflecting the impact of efficient note-taking. (Target: 25% increase - Objective 2).
- **KPI 4: System Uptime:** Monitor system uptime percentage to ensure system reliability and availability for efficient use. (Target: 99% uptime - Objective 3).
- **KPI 5: User Satisfaction Score:** Collect user satisfaction ratings through usability testing and feedback questionnaires to assess user experience and identify areas for improvement in efficiency and usability. (Target: 4.5/5 stars - Objective 3).
- **KPI 6: Summary Quality Score:** Evaluate the quality of AI-generated summaries through manual review and potentially automated metrics (e.g., information retention, coherence), ensuring summaries are effective and efficient for review.
- **KPI 7: Recommendation Relevance Score:** Assess the relevance and usefulness of learning material recommendations through user feedback and click-through rates, indicating the efficiency of resource discovery.

8.2. Testing and validation strategies:

- **Unit Testing:** Implement unit tests for individual components and modules of the mobile application, web interface, and server-side application to ensure code quality and functionality for efficient operation.

- **Integration Testing:** Conduct integration tests to verify the interaction and data flow between different system components (e.g., mobile app and server, server and AI APIs, database interactions) to ensure seamless and efficient workflow.

- **System Testing:** Perform end-to-end system testing to validate the complete functionality of AudioScholar, including recording, summarization, recommendation, cloud synchronization, and user authentication, ensuring the system functions efficiently as a whole.

- **Usability Testing:** Conduct usability testing sessions with target users (students) to evaluate the user interface, user experience, and overall usability of the mobile and web applications, focusing on ease of use and efficiency. Gather feedback through observation and questionnaires.

- **Performance Testing:** Conduct performance testing to evaluate system response time, scalability, and resource utilization under different load conditions, ensuring efficient performance even under stress.

- **Security Testing:** Perform security testing to identify potential vulnerabilities and ensure the security of user data and system infrastructure, maintaining user trust and efficient data handling.

- **User Acceptance Testing (UAT):** Deploy the application to a small group of target users for a pilot phase to gather real-world feedback and validate user acceptance before wider deployment, ensuring the application effectively addresses user needs for efficient note-taking.

9. Conclusion

9.1. Summary of the proposal:

This proposal outlines the development of AudioScholar, an innovative software solution designed to transform inefficient traditional note-taking into an efficient and effective process for students. AudioScholar addresses the problem of inefficient lecture note-taking by providing AI-powered summarization after lecture recording, offline capabilities, and personalized learning material recommendations. The project leverages cutting-edge technologies and adopts an Agile development methodology to ensure a user-centric and adaptable development process within the 8-week development timeframe. AudioScholar is poised to significantly enhance the learning experience for students, improve academic performance, and revolutionize lecture note-taking by making it an efficient and supportive tool for learning.

9.2. Call to action:

We believe AudioScholar represents a valuable and impactful Capstone project with the potential to make a significant contribution to the field of educational technology by providing an efficient solution to lecture note-taking. We respectfully request approval to proceed with the development of AudioScholar as our Capstone project within the 8-week development timeframe. We are confident in our team's ability to deliver a high-quality, functional, and user-friendly application that meets the outlined objectives and benefits students in higher education by making their note-taking process more efficient. We are eager to embark on this project and bring AudioScholar to fruition.

10. References

[1] T. Gur, T. Dilci, İ. Coskun, and B. Delican, "The impact of note-taking while listening on listening comprehension in a higher education context," *International Journal of Academic Research*, vol. 5, no. 1, pp. 93–97, Jan. 2013, doi: https://doi.org/10.7813/2075-4124.2013/5-1/b.16.

[2] X. Wang, "An analysis of note-taking strategies: The effect of translanguaging on content comprehension and knowledge retention," *Journal of Language Teaching*, vol. 1, no. 3, pp. 1–20, Nov. 2021, doi: https://doi.org/10.54475/jlt.2021.020.

[3] F. Bry and A. Y.-S. Pohl, "Large class teaching with Backstage," *Journal of Applied Research in Higher Education*, vol. 9, no. 1, pp. 105–128, Feb. 2017, doi: https://doi.org/10.1108/jarhe-06-2015-0042.

[4] Artz, B., Johnson, M., Robson, D., & Taengnoi, S. (2020). Taking notes in the digital age: Evidence from classroom random control trials. The Journal of Economic Education, 51(2), 103–115. https://doi.org/10.1080/00220485.2020.1731386

[5] M. Seddik *et al.*, "People's Democratic Republic of Algeria Ministry of Higher Education and Scientific Research An Investigation of Lectures Note-Taking Strategies and Difficulties The Case of Third Year Students of English at the University of Mohammed." Accessed: Feb. 19, 2025. [Online]. Available: http://dspace.univ-jijel.dz:8080/xmlui/bitstream/handle/123456789/7504/420.382.pdf?sequence=1