**SWE30010 Managing IT Projects**

**Sprint Setup Task 8 - Intelligent Chatbot Project**

**Name: Derrick Lu Qing Lee**

**Student ID: 101215931**

**Tutor's Name: Sim Kwan Hua**

**Project Proposal: Intelligent Chatbot for Software Project Knowledge Management**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Item | Dependencies | Business Value  (1 least – 10 most) | Release Schedule (Sprint #1 | 2 | 3 | ….) |
| 1 | Set up chatbot framework (create basic structure) | None | 10 | Sprint #1 |
| 2 | The staffs can understand the documents easier since the chatbot will explain it in simpler form. | None | 9 | Sprint #1 |
| 3 | Each user from different departments/group projects are given specific authentication to access their project. | 1 | 9 | Sprint #1 |
| 4 | Staffs can retrieve back the archive chats for future reference. | 1 | 6 | Sprint #1 |

**Task 1: Criteria for Prioritizing Backlog Items**

The following criteria were used to prioritize the backlog items for Sprint #1:

1. **Business Value**
   * Features that provide the highest impact on productivity, usability, and client satisfaction are prioritized. High business value ensures that the feature contributes directly to project goals, such as improving team efficiency.
2. **Development Effort**
   * Estimating development effort helps identify tasks that are manageable within a sprint. This avoids overcommitting and ensures that tasks selected can be realistically completed within the sprint's time constraints.
3. **Feature Dependency**
   * Dependencies between features are considered to ensure smooth progression. Tasks that other features depend on are prioritized to allow for efficient workflow and future development.
4. **Date Needed / Timeline**
   * Time-sensitive features are prioritized to meet deadlines, ensuring that essential functionalities are ready for testing or presentations at specific times.
5. **Risk Involved**
   * High-risk features, such as those involving untested integrations, are tackled early to identify and address issues sooner, reducing potential risks later.
6. **Usability and Accessibility**
   * Features that directly impact user accessibility and ease of use are prioritized to improve the user experience, especially for new team members onboarding with the chatbot.

*Prioritization Decision*: All criteria are important, but **Business Value** and **Development Effort** have slightly higher weight, ensuring that essential features are implemented efficiently and contribute directly to the project’s success.

**Task 2: Select the Highest-Priority Backlog Items for Sprint #1**

Based on the criteria above, the following items are selected as the highest-priority items for Sprint #1:

1. **Set Up Chatbot Framework (Basic Structure)**
   * **Business Value**: 10
   * **Reasoning**: This foundational task is crucial as it establishes the core infrastructure needed to support future features. With no dependencies and high business value, setting up the framework in Sprint 1 provides a stable foundation for subsequent development.
2. **Enable Simplified Document Explanations by the Chatbot**
   * **Business Value**: 9
   * **Reasoning**: This feature aids in onboarding new team members by simplifying complex documentation. It has high usability value, directly impacting user experience and productivity. It can be completed within the sprint, showcasing the chatbot's primary functionality early.
3. **Configure User Authentication for Project-Specific Access**
   * **Business Value**: 9
   * **Reasoning**: Implementing secure, role-based access ensures that only authorized users can access relevant information, aligning with the project’s emphasis on data privacy. This feature provides a critical layer of security for the chatbot, with high business value and feasibility within the sprint.
4. **Enable Archived Chat Retrieval**
   * **Business Value**: 6
   * **Reasoning**: This feature supports continuity and knowledge sharing by allowing users to retrieve past interactions. Although slightly lower in business value, it enhances the chatbot’s functionality and is feasible within the sprint.

**Task 3: Work Breakdown Structure (WBS) and Justification**

The selected items for Sprint 1 have been organized into a Work Breakdown Structure (WBS) to ensure alignment with the project goals and feasibility within the two-week sprint. Each item has been broken down into manageable tasks, with testing incorporated to maintain quality standards.

A diagram of a chatbot framework

Description automatically generated

***Justification***

The first item, **Set Up Chatbot Framework**, lays the foundation for the entire project. This task includes researching and defining requirements (4 hours), which involves identifying the key documentation requirements (2 hours) to ensure the chatbot’s functionality aligns with user needs and selecting appropriate tools and technologies (2 hours) for compatibility and scalability. Developing the chatbot’s basic structure (3 hours) includes creating a simple UI/UX and interaction flow, enabling users to interact with the chatbot easily, enhancing usability from the outset. Backend services are then configured to support data storage and access, including setting up a database for archived data (3 hours) and deploying a server (3 hours) to ensure reliable performance. Finally, testing the core framework (1 hour) ensures that the initial setup functions smoothly and prepares it for further development. This structure provides a stable, high-quality foundation for future functionality, making it an essential first step in Sprint 1.

A diagram of a chatbot

Description automatically generated

***Justification***

The second item, **Enable Document Simplification for Easier Understanding**, focuses on a feature that will help users, particularly new team members, understand complex project documentation in simpler terms. This task begins with researching NLP techniques (4 hours), including identifying relevant NLP methods or APIs (2 hours) to confirm feasibility and creating a function for the chatbot (2 hours) that will parse and simplify complex information. The development of the simplification module (5 hours) involves building the functionality for parsing and simplifying content (2 hours) and integrating it into the chatbot framework (3 hours), allowing users to request simplified explanations directly. Testing this feature (1 hour) ensures that it operates as intended, providing clear and accurate content. This breakdown supports the project’s goal of improving accessibility, especially for new users.

A diagram of a software system

Description automatically generated

***Justification***

The third item, **Configure User Authentication for Project-Specific Access**, aims to implement a secure, role-based access system to protect sensitive project information. This task includes defining authentication requirements (2 hours), specifically choosing a suitable authentication method to establish robust access control. Developing the authentication module (5 hours) consists of implementing basic authentication (2 hours) to allow only authorized users to access the chatbot and setting up department-specific access (3 hours) to restrict users to relevant information, thus enhancing data security. Testing the authentication system (1 hour) verifies that access controls function as expected. This setup provides a critical layer of security for the chatbot, aligning with the project’s emphasis on data privacy.

A diagram of a chat bot

Description automatically generated

***Justification***

The final item, **Enable Archived Chat Retrieval**, allows users to retrieve past conversations, supporting continuity and knowledge sharing. This task begins with setting up a database (3 hours) that includes defining the data structure (1 hour) and implementing a schema for organized archived chats (2 hours). Developing the archive retrieval function (2 hours) enables users to search and retrieve archived interactions, essential for referencing previous information. Integration with the chatbot interface (1 hour) links the retrieval function directly to the chatbot, making it accessible to users. Testing this feature (1 hour) confirms its reliability, ensuring users can reference past interactions as needed. This structure makes archived chat retrieval functional and accessible, meeting the goal of providing valuable reference material.