Gatino - Innovation Gateway for Swinburne Vietnam

Project Plan

GROUP 1

Name	Position	Email	Phone
Ta Quang Tung	Team Leader, Client Liaison	104222196@student.swin.edu.au	0921426803
Nguyen Quang Huy	Development Manager	104169507@student.swin.edu.au	0968751524
Tran Hoang Hai Anh	Support Manager	104177513@student.swin.edu.au	0866899518
Phan Sy Tuan	Quality Manager	104072029@student.swin.edu.au	0888052003
Duong Quang Thanh	Planning Manager	104167828@student.swin.edu.au	0976663084

COS40006, COMPUTING TECHNOLOGY PROJECT B, SPRING 2025

Document Change Control

Version	Date	Authors	Summary of Changes
1.0	6/10/2024	Ta Quang Tung, Nguyen Quang Huy, Tran Hoang Hai Anh, Phan Sy Tuan, Duong Quang Thanh	Initial version of the document.
2.0	6/4/2024	Ta Quang Tung, Nguyen Quang Huy, Duong Quang Thanh	Updated the document to better reflect the process of CTPB.

Document Sign Off

Name	Position	Signature	Date
Ta Quang Tung	Team Leader, Client Liaison	Quang Tung Ta	06/10/2024
Nguyen Quang Huy	Development Manager	Quang Huy Nguyen	06/10/2024
Tran Hoang Hai Anh	Support Manager	Hoang Hai Anh Tran	06/10/2024
Phan Sy Tuan	Quality Manager	Sy Tuan Phan	06/10/2024
Duong Quang Thanh	Planning Manager	Quang Thanh Duong	06/10/2024

Client Sign off

Name	Position	Signature	Date
Dr. Le Minh Duc	Head of IT Department, Swinburne Vietnam Head of Innovation Lab, Swinburne Vietnam	Le, Minh Duc	04/10/2024
Organisation: Innovation Lab, Department of Computer Science, Swinburne Vietnam			

Contents

1. Introduction

This document presents the detailed plan for project Gatino - Innovation Gateway for Swinburne Vietnam, containing key information about the project and its stakeholders. It is written for the development team as well as the client and supervisor (Dr. Le Minh Duc) to help them better understand the project goals and track its progress.

1.1. Background

Swinburne Vietnam has been actively fostering innovation among students to inspire creativity and enhance their learning. Over the years, many innovation projects have been developed by students from a diverse range of majors, namely Software Development, Internet of Things, Data Science, and Artificial Intelligence. However, currently there lacks a unified method to effectively manage these projects and package them for demonstration and reuse. Projects are kept in separate places and setting them up for demonstration has proven time-consuming and difficult. Gatino (short for Innovation Gateway) was thus born to address this logistical need. Proposed by the Innovation Lab of the IT Department at Swinburne Vietnam, this project will provide a means to better manage, showcase, and package innovation projects for external integration.

1.2. Key Project Personnel

The key personnel involved in this project are as follows:

1.2.1. Client

This project was proposed by the Innovation Lab of the IT Department at Swinburne Vietnam, represented and directed by Dr. Le Minh Duc. The lab consists of Swinburne students and lecturers with a primary focus on software research and development.

1.2.2. Other Stakeholders

As Gatino is developed for use within Swinburne Vietnam, students and lecturers are the primary stakeholders of the project.

- **Students** are interested in uploading their projects to the platform and using it to showcase their work. They may also want to reuse features from other student projects.
- **Lecturers** want to use the platform to better manage their students' work. Head of Departments at Swinburne are particularly interested as they can use the information of the uploaded projects to better understand the performance of different majors and the technologies students are interested in.
- **Admission** and **Industry representatives** want to run a demonstration of the students' innovative products to view their functionality.

1.2.3. Project Supervisor, Team Leader, and Key Project Members

Supervisor: Dr. Le Minh Duc

Team:

Name	**Role**
Ta Quang Tung	Team leader, Client liaison
Nguyen Quang Huy	Development manager
Tran Hoang Hai Anh	Support manager
Phan Sy Tuan	Quality manager
Duong Quang Thanh	Planning manager

2. Terms of Reference

2.1. Goals

The goal of this project is to develop the Search and the Demonstration functionalities for Gatino. Our team will be working in conjunction with the core Gatino team, who will develop the other necessary features of the platform such as innovation project deployment and integration.

2.2. Objectives

Our main focus in this project is on the **Search feature** and the **Demonstration (or Demo) feature** of Gatino. For these, we define the following objectives:

1. Allow users to search for projects on the platform with the option to provide specific search filters for more fine-grained results sorted by relevance.
2. Provide users with a smooth and rich project demonstration experience that lets them interact with and view projects in real time. The types of projects that must be supported are web, Android, IoT.
3. Enable users to access the search and demonstration features anywhere, anytime from the comfort of their browsers.
4. Seamlessly integrate the search and demonstration features with the other components of the Gatino system to ease development and maintenance.

2.3. Scope

This project will span two semesters from September 2024 to early April 2025 (7 months). All deliverables are expected to be submitted by April 6th, 2025 (the end of the Spring semester).

Must have:

1. Develop a responsive web interface based on Gatino's chosen theme for the search and demo features.
2. Develop a search engine that can search projects by metadata (name, description, technology used, etc.), source code, and associated documentation.
3. Develop a RESTful API for the search engine whose access is restricted to the Gatino system only.
4. Package the search and demo features into independently deployable services that can be integrated with the other components of Gatino.
5. Research and develop method allowing users to interact with web applications through their browser, and access by dedicated URL.

6. Research and develop methods for users to interact with physical Android app demonstrations through their browsers.
7. Research and develop a broker that supports IoT projects communicating among devices.

Out of scope:

1. Maintain the system after deployment.
2. Develop the means to deploy innovation projects.

2.4. Critical Success Factors

1. Help 100% of users quickly search for and filter innovation projects with an easy-to-write search query and parameters such as programming language, framework used, etc. The search should return results in under 2 seconds.
2. A new user should figure out how to use the search feature in less than 5 minutes of first looking at the interface, as its design and functionality are intuitive and similar to popular search engines like Google or Bing.
3. The user interface should be responsive and should present the search results in a logical and navigable manner.
4. After a new project has been uploaded to the system, it should be searchable after at most 5 minutes.
5. After an existing project has been modified in the system, the search engine should reflect its new information after at most 5 minutes.
6. When many users try to access a demonstration that can only be used by one person at a time, they should be served in a first-come, first-serve manner.
7. If the project supports demonstration on a real device, the user should be able to fully interact with it through the browser as if they were holding the actual device.

2.5. Acceptance Criteria

The result of the project is considered acceptable once it satisfies all of the following criteria:

1. The result fully (or at least nearly) fulfils the 4 objectives defined in 2.2.
2. The result encompasses all items listed in the Must Have section of the Scope.
3. The result fully meets the Critical Success Factors.
4. The result includes all Deliverables as listed in section 4.1.

To verify that the Critical Success Factors listed above are met, we will conduct a user acceptance test after the initial deployment. The target of these tests are students and

lecturers at Swinburne Vietnam, and we will prepare a survey to measure how much the requirements have been met.

3. Establishment

3.1. Processes, Procedures and Standards

In this project, we will follow the Agile development methodology using the Scrum framework. We picked Agile for two main reasons. First, this methodology is also used by the core Gatino team, so the client is familiar with it. Second, it lets us involve the client in every part of the process, thus enabling us to get frequent feedback to improve our product and change course if necessary. To facilitate Scrum development, we use Jira to manage our tasks and sprints.

We will use Git as our version control system, and GitHub as the platform to host our code repository.

We anticipate that we will need to use a variety of programming languages and technologies in this project. For each technology, we will conform to the following standards:

- JavaScript / TypeScript: We will follow the TypeScript style guide prescribed by Google, plus any style guide prescribed by the core Gatino team.
- Frontend (or User Interface): We will develop the user interface with React using the Mantis theme chosen by the core Gatino team to ensure stylistic consistency throughout the frontend.
- Backend: We will develop our API with Express.js following the project structure prescribed by the core Gatino team.

3.2. Project Environment

Our team will use our personal computers for the development process, and we will work online or offline as best suits our schedules. Our design process will be documented on the group's Jira and Google Drive, and any code output will be pushed to a shared private GitHub repository.

Swinburne Vietnam will provide the server for hosting the project. The server will run the Docker Engine on which Gatino's infrastructure and subsystems will be deployed, including our features.

We will build our code output and other services into Docker images that can be easily deployed to the Docker Engine running on the server.

3.3. Project Team Skill Development Requirements

Skill Area	Skill Description	Training Requirement and Suggested Resources
Android Basics	Understanding how to deploy Android apps and basic knowledge of Android device functionalities.	Training Not Required, can be self-taught Suggested Resources: - Android Developer documentation on app deployment - Peer-led sessions, online tutorials
IoT Basics	Understanding core principles of IoT device communication, including protocols (e.g., MQTT, CoAP), network architectures, and data flow between devices and cloud platforms.	Suggested Resources: - Official MQTT documentation - IoT-focused courses (e.g., Coursera's "IoT Systems and Industrial Applications")
Test Basics	Proficiency in test implementation, lifecycle management, and familiarity with modern testing frameworks for TypeScript/JavaScript (e.g., Jest, Mocha, or Cypress). Ability to design unit, integration, and end-to-end tests	Suggested Resources: - Official Jest documentation - "Testing JavaScript Applications" (O'Reilly)
Docker Basics	Familiarization with Docker, including: - Building Docker images from Dockerfiles - Orchestrating multiple containers with Docker Compose - Configuring environment variables and port mapping for containers	Necessary Suggested Resources: - Docker documentation and tutorials - Online courses on platforms like Udemy or Coursera

Independent Learning	Development of skills to independently explore new technologies. Key aspects include: - Reading and understanding documentation - Note-taking to consolidate learning - Discussing and sharing knowledge with team members	Emphasis on self-improvement Suggested Resources: - Documentation for relevant tools and technologies (e.g., Elasticsearch) - Note-taking and knowledge-sharing techniques
User Acceptance Testing	Training for team leaders and quality managers on how to conduct user acceptance testing, an essential step in project evaluation before deployment.	Professional training if possible Suggested Resources: - Training sessions from professionals or university staff, if available - Online UAT guidelines and documentation

4. Deliverables, Activities and Capital Resources

4.1. Deliverables

We will submit the following deliverables during or at the end of the project:

1. The system's services and components:
    1.1. Source code of the project (hosted on GitHub) and test results.
    1.2. Docker images of our custom services/components as well as of third-party services (e.g., content extraction, search, database, etc.).
    1.3. A Docker Compose file to orchestrate multi-container deployment and configuration.
2. The system's documentation, including:
    2.1. Project Plan.
    2.2. SQAP - Software Quality Assurance Plan
    2.3. Software Requirement Specifications.
    2.4. System Architecture Design and Research Report
    2.5. Detailed Design and Implementation Report

2.6. Detailed architecture/design diagrams of the system.

2.7. Test Plan and Test Report.

2.8. Usability Evaluation Plan and Usability Evaluation Report.

2.9. User Manual and Installation Manual.

2.10. Final Project Report.

2.11. Research reports (if any) of the search and demo features.

2.12. Detailed instructions on how to configure the features and deploy them to Docker.

2.13. Final presentation video, Project poster.

4.2. Activities

Following the Agile methodology, we define two Epics corresponding to the search and demo features that we plan to work on.

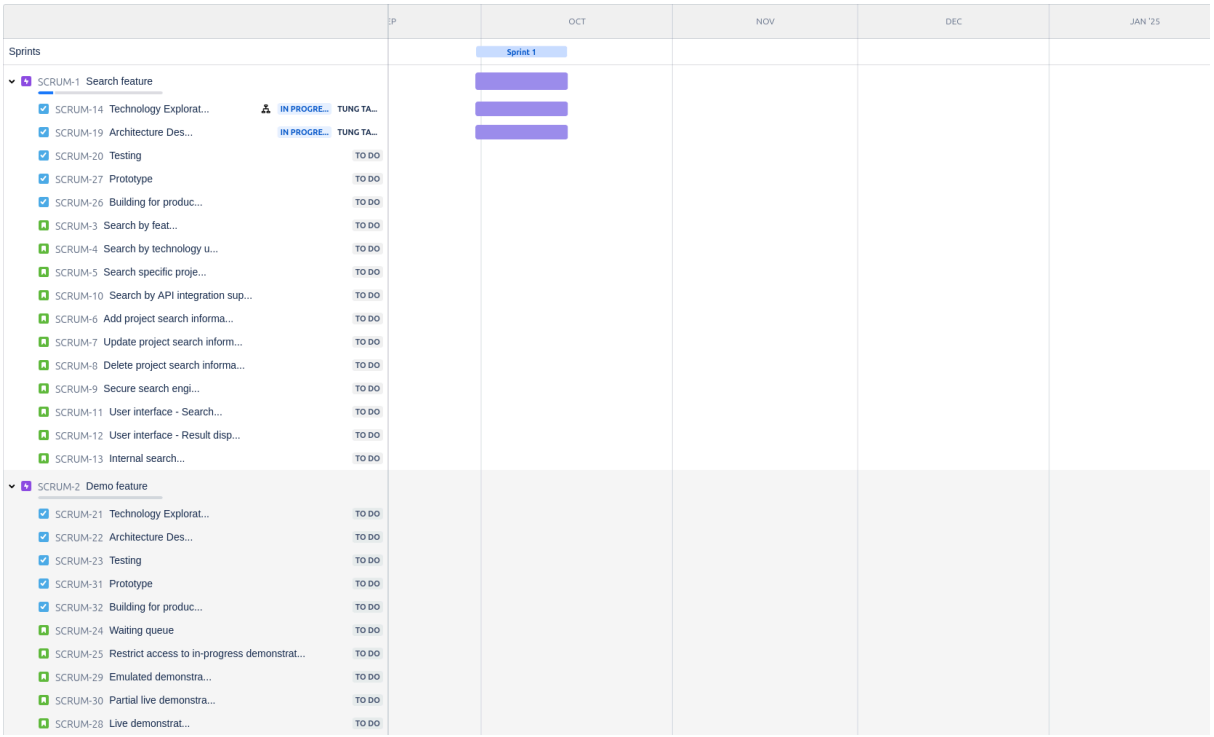

Fig. 1: The epics and initial issues for each epic. The full list can be accessed on Jira.

For each epic, we will do the following activities:

1. **Requirement gathering and analysis**: We will discuss with the client about the feature to better understand its requirements and objectives. Next, we break down the requirements into sub-tasks that the system has to solve. *Deliverable: 2.1.*
2. **Technology exploration**: With the sub-tasks identified, we will research the relevant technologies/frameworks/tools to realize them. The output of this activity is a

research report that details existing technologies for the job along with their strengths and weaknesses. If no technology exists to solve the problem, we will discuss with the client to ease the requirements. *Deliverable: 2.3.*

3. **Architecture design**: We will pick some of the technologies identified in the previous activity and assemble them into the architecture design of the system. We may have to revisit this activity in the future to make any architectural adjustments as necessary. *Deliverable: 2.2.*
4. **Backend development**: With the system design in place, we will begin development, focusing on the backend first as it is more important for our chosen features. We will pull tasks/stories from the Jira backlog and group them into sprints, which last from 1 to 2 weeks. This activity may be revisited if necessary. *Deliverables: 1.1, 1.2, 1.3.*
5. **Frontend development**: After the backend is complete (or in progress if our team is going strong), we will work on the frontend. Again, we will pull tasks from the backlog and put them into sprints. This activity may be revisited if necessary. *Deliverables: 1.1, 1.2, 1.3.*
6. **Prototype building and initial testing**: After the backend and frontend are complete, we will put them together to prepare the prototype of the feature. We will aim to make this prototype as close to the desired product as possible. We will conduct user acceptance testing on the prototype to identify its shortcomings and bugs. These will be added as issues to the Jira backlog. *Deliverables: 1.1, 1.2, 1.3.*
7. **Further development and testing**: This stage is dedicated to improving the prototype and fixing any bugs identified in the initial user acceptance test. At the end of this stage, we will conduct another user acceptance test to verify the system's success. *Deliverables: 1.1, 1.2, 1.3.*
8. **Final product preparation**: In this stage, we prepare the project to submit to the client, including any necessary documentation and configuration. *Deliverables: 2.4, 2.5.*

The above eight activities and their order may not be strictly followed depending on our project status. For example, we may skip step 7 if the client accepts the prototype for deployment, or we may revisit steps 2 and 3 if we later discover that some of the chosen technology is not adequate for the project.

Each activity will be broken down to several sprints if necessary. As we have two features to develop, we may run two different activities in parallel.

For sprints that coincide with a particular assignment of the unit (such as reports, videos, presentations, etc.), we will include these as tasks in the sprint. For example, if the submission deadline for the Project Plan is within Sprint 1, we will include the Project Plan as part of Sprint 1.

4.3. Resources

1. Hardware and Infrastructure
    - Cloud Infrastructure: Servers for hosting the search and demo features and managing storage for indexed content (Swinburne server) .

- Workstations: Computers or laptops with adequate specifications for development, testing, and system management.
- Android devices: A physical Android device to showcase the Android projects.

2. Software and Libraries
    - Search Engine Libraries: Open-source search libraries like Elasticsearch, Solr, or custom-built search algorithms for real-time indexing.
    - Android Display and Control Libraries: Open-source solutions such as Srcrpy.
    - MQTT Broker: Open-source MQTT such as Mosquitto.
    - Docker Engine: To host the system's components and services.
3. Development Tools:
    - Integrated Development Environments (IDEs) such as Visual Studio Code, along with version control tools (Git).
4. Database Systems:
    - Databases such as MySQL solution to support data storage and indexing.
5. Collaboration and Documentation Tools
    - Project Management Software: Jira for tracking tasks and milestones.
    - Documentation Tools: Google Docs for creating project and user documentation.
6. Testing Resources
    - Testing Environments: Dedicated servers or cloud environments for conducting load and performance testing.
    - User Testing Groups: Access to students and faculty for usability testing to evaluate search engine efficiency and effectiveness.

These activities and resources will ensure that the search and demo features are developed efficiently, meet user needs, and are scalable for future growth.

5. Organisation and Structure

This project requires seamless collaboration among various groups and individuals. This section outlines all of the people who were involved in the project, including their roles and responsibilities.

1. Project team:

    - **Ta Quang Tung** (Team Leader and Client Liaison): Oversees the project and maintains communication with the client to ensure their needs are met.
    - **Nguyen Quang Huy** (Development Manager): Leads the development team, ensuring the software is designed, coded, and implemented effectively.
    - **Tran Hoang Hai Anh** (Support Manager): Manages the support team, providing technical assistance and resolving any issues that arise
    - **Phan Sy Tuan** (Quality Manager): Heads the quality assurance team, ensuring the software is thoroughly tested and meets all quality standards.
    - **Duong Quang Thanh** (Planning Manager): Coordinates project planning, managing schedules, resources, and ensuring timely delivery of project milestones.

2. Client: **Dr. Le Minh Duc (also Supervisor in CTPB)**

    In this project, Dr. Le Minh Duc, as the client, plays a crucial role by providing the project's objectives, requirements, and expectations, ensuring the team understands the desired outcomes. He will review progress, offer feedback, and suggest improvements to align the project with his vision. Additionally, Dr. Le Minh Duc will participate in key project reviews and meetings, make critical decisions, and engage in acceptance testing to validate that the final product meets all specified requirements. Finally, he will give the official approval and sign off on the project deliverables, confirming that the project has been completed to his satisfaction.

3. Core **Gatino team** at the Innovation Lab:

    In this project, apart from Dr. Le Minh Duc, the project team must collaborate with the core Gatino team from the Innovation Lab to fully develop the system. The core Gatino team will mostly be responsible for the front-end implementation, system integration, and deployment. They will work closely with the project team to ensure that all technical aspects meet the client's specifications. Additionally, the core Gatino team will assist in acceptance testing to ensure the final product is functional and meets quality standards.

4. Supervisor: **Dr. Pham Van Dai (CTPA)**

    In this project, Dr. Pham Van Dai is pivotal in overseeing progress and ensuring that all activities align with the established goals and standards. His responsibilities include providing guidance and support to the project team, monitoring milestones, and ensuring the project stays on schedule and within the given timeline. Additionally,

Dr. Pham Van Dai will facilitate communication between different teams, address any issues that arise, and ensure that the project meets the quality standards and requirements set by the client.

5. **Students and lecturers** at Swinburne Vietnam

   In this project, students and lecturers at Swinburne Vietnam will participate in the testing process as they represent the target audience for the system. Their involvement will provide valuable user feedback, ensuring that the system meets the needs and expectations of its end users

Table 1 Activities and Deliverable

Deliverable Activity	1.1 Source code & test results	1.2 Docker images	1.3 Docker Compose file	2.1 Plan & SRS	2.2 Architecture design	2.3 Research reports	2.4 Deploy & config manual	2.5 API docs
1 - Req. gathering and analysis				Project team Client Supervisor				
2 - Tech exploration						Project team Client		
3 - Architecture design					Project team Client Supervisor			
4 - Backend dev	Project team Gatino team Client	Project team Client	Project team Client				Project team Client	Project team Client
5 - Frontend dev	Project team Gatino team Client	Project team Client	Project team Client				Project team Client	
6 - Prototype & testing	Project team Students and lecturers Client	Project team Client	Project team Client				Project team Client	Project team Client
7 - Further dev & testing	Project team Students and lecturers Gatino team Client	Project team Client	Project team Client				Project team Client	Project team Client
8 - Final product	Project team Client	Project team Client	Project team Client				Project team Client	Project team Client

6. Risks

Rank	Name	Description	Occurrence Probability (H/M/L)	Severity (H/M/L)	Mitigation Strategy Number	Contingency
1	Limitations of Existing Technologies	A feature required by the client is not feasible using existing technology and developing a custom solution is too costly or time-consuming.	M	H	6	Provide the client with a research report explaining the limitations of existing technologies and suggest modifications to the feature's requirements.
2	Failure To Index Past Projects	The search engine is operational after some projects have already been uploaded to the platform, and thus past projects have not been indexed by the search engine.	H	M	7	Periodically scans the Project table to detect projects that have not been indexed.
3	Project Management Module and Search Module's Communication Failure	An error occurs when the Project Management module tries to notify the Search module of a project upload/update, and thus the information is not added to/updated in the search engine.	M	H	8	Periodically scans the Project table to detect projects that have not been indexed.
4	Technical Challenges with Search Engine	Issues integrating the search engine with the existing system or	L	H	1	Switch to another search engine technology that can accommodate the project's needs.

		limitations with search algorithms.				
5	Data Inconsistencies	Issues with how projects are stored and managed in the database, leading to incomplete or inconsistent search results.	M	M	2	Develop data-cleaning scripts or manual intervention to correct data errors.
6	Team Availability	Team members being unavailable due to personal commitments, illness, or workload, which may cause delays.	L	H	3	Reassign tasks to available team members, and adjust timelines accordingly.
7	Performance Issues	Search engine performance may degrade with a large dataset, affecting response times.	L	H	4	Implement caching or index optimization techniques to improve performance.
8	Client Requirement Changes	The client may request changes to the scope or functionality mid-project.	M	M	5	Prioritize new features and drop less critical functionalities to stay on schedule.
9	Unsatisfactory Feature After Testing	The feature does not satisfy the client's requirements after the acceptance test.	L	H	9	Ask the client for a deadline extension (if possible), and work in the remaining time to correct the feature.

Table 2: Risks

Mitigation Strategy Number	Details
1	Regular progress reviews and early prototyping to identify issues early.
2	Collaborate with the core team to standardize project data fields before starting development.
3	Allocate backup team members for critical tasks. Establish a shared task board (e.g., Jira) to keep track of responsibilities.
4	Conduct performance testing during development and optimize algorithms for scalability.
5	Use Agile methodology to accommodate changes with minimal disruption and keep the client updated with frequent sprint reviews.
6	Ensure that the client sets realistic requirements during the requirement gathering phase.
7	Try to deploy the search feature at the same time as the Gatino launch, thus ensuring that no past projects are not indexed.
8	Use an event streaming platform, such as Apache Kafka, to store Upload and Update events for future retrieval by the search engine.
9	Get the client’s feedback regularly, conduct acceptance tests early to have time to detect mistakes.

Table 3: Mitigation Strategies

7. Schedule

7.1. Project Timeline

The following is the tentative timeline of the project. The actual duration of the sprints may be adjusted depending on the team and project status. The timeline can be found on Jira.

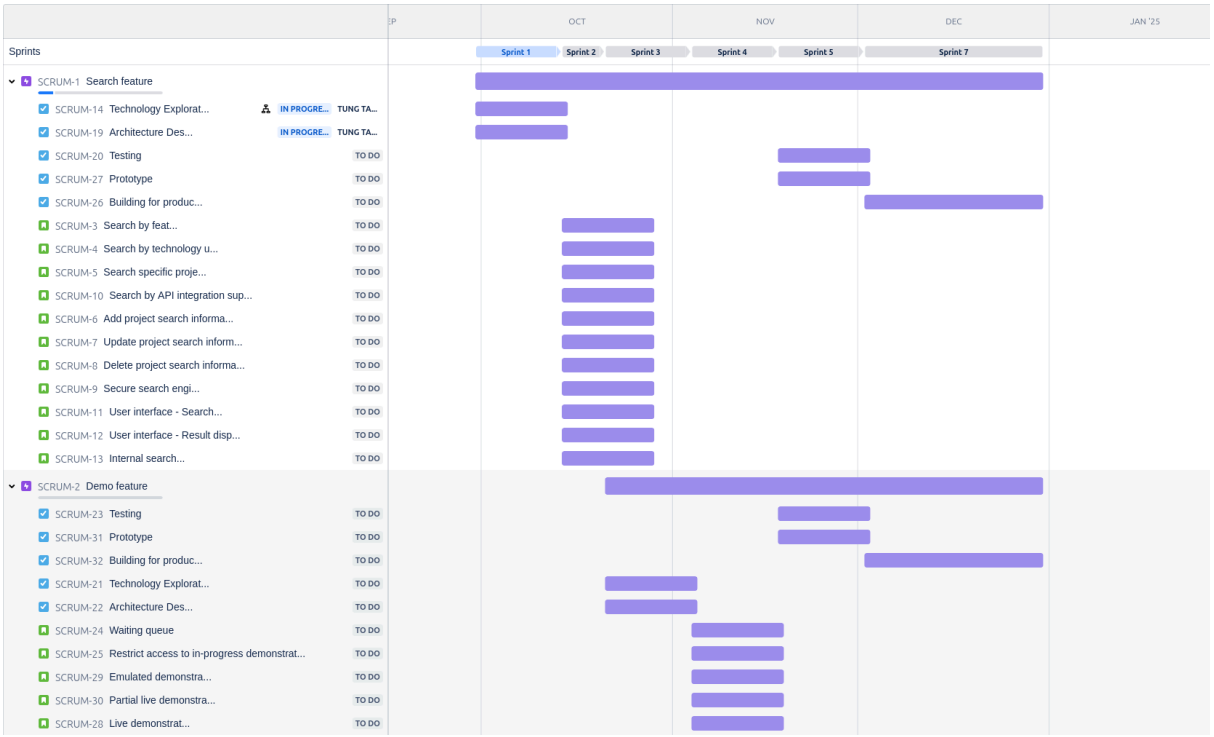

Fig. 2: The timeline of the project.

7.2. External Dependencies

To make sure that the project timeline is met, we have to consider all of the external parties that also involved in the project:

1. **Client:**

   Weekly feedback from our client is critical since our team can adjust the scope of the project and ensure that we keep everything on track.

2. **Core Gatino team's progress:**

   The progress of the core Gatino team is also crucial since during the setup development task at the end of sprint 3, we have to make sure that the upload project feature that the core Gatino team is finished so that we can integrate our search feature.

3. **Gatino's deployment server setup:**

Gatino will be deployed to a server running at Swinburne Vietnam. Before we can successfully deploy our features, we have to wait for the server to be up and running. This is not a concern in the earlier stages of the project, and we will ask the client for progress when it is close to deployment.

7.3. Assumptions

When planning the project schedule, we have made several assumptions that can affect the progress of the plan in consideration:

- **Stable requirements**: We assume that the project requirements provided by the client are stable and won't undergo significant changes during the development process. Any major alterations to the requirements may impact the project timeline.
- **Resource availability**: We assume that team members allocated to specific tasks will be available as planned. However, we acknowledge that unforeseen circumstances (such as illness or other emergencies) can affect availability. To mitigate this, we've allocated backup team members for critical tasks.
- **External dependencies**: Our project plan assumes timely delivery of external dependencies. Specifically:
    - **Client feedback**: We rely on regular feedback from the client to validate our progress and make necessary adjustments. Delays in client feedback could impact subsequent phases.
    - **Core Gatino team**: The completion of the upload project feature by the core Gatino team is crucial for our integration phase. We assume that their progress aligns with our timeline.
- **Infrastructure and environment**: We assume that the necessary development, testing, and production environments will be set up as planned. Any delays in infrastructure provisioning could affect our schedule.

8. Budget

Personnel Cost

Name	Position	Rate per hour (USD)
Ta Quang Tung	Team Leader & Client Liaison	$0.00
Nguyen Quang Huy	Development Manager	$0.00
Tran Hoang Hai Anh	Support Manager	$0.00
Phan Sy Tuan	Quality Manager	$0.00
Duong Quang Thanh	Planning Manager	$0.00

Table 4: Personnel Cost

Time Estimated to Complete Each Task

Activity	Task	Estimated Hours (hrs)	Total per Activity (hrs)
1. Requirement Gathering	Meeting with clients, understanding project requirements	36	66
	Database structure analysis, indexing approach	30	
2. Architecture Design	System architecture design	72	144
	Database and index design	48	
	Setup of development environment	24	
3. Frontend UI Development	Build search UI and filtering	72	168
	Build UI for product demonstration	72	
	Implement autocomplete feature	24	
4. Backend Development	Simple and advanced search with full-text search and filtering implementation	72	228
	Real-time indexing	12	
	Physical Android demonstration feature	72	
	IoT demonstration feature	72	
5. Testing & Optimization	Functional and performance testing	48	96

	End-user feedback gathering	48	
6. Deployment & Documentation	Final deployment and documentation	48	48

Table 5: Task time estimate

9. References

None

10. Tables Index