Project Overview

Terminology

Planning problems: Planning problems involve finding a sequence of actions for a problem given its domain to transition a system from an initial state to a desired goal state, often subject to constraints and objectives.

Domain Profile: Define a set of possible actions that can be taken within a problem scenario.

Problem Profile: A problem profile refers to a specific planning task instance, which describes the initial state, goal state, and specific constraints that may be involved. This is a framework based on PDDL, specialized in a specific domain.

Animation Profile: A user-written specification in a PDDL-like language. It instructs the visualizer on how to represent a planning problem's domain visually.

Planimation: A graphical representation of the planning problem-solving process, illustrating through animation each step from an initial state to a goal state.

Background

Planimation is an open-source framework for visualizing the sequential solutions of planning problems specified in PDDL. Planimation was developed as a final-year project by students at the University of Melbourne since 2018. The Planimation helps to reduce the difficulty of interpreting the results of planning problems with the use of an animation profile. To achieve the visualization, three files are required: problem file, domain file and animation profile. Each problem domain in Planimation is associated with an Animation Profile, allowing for customized visualization of different types of planning problems. The animation profile allows the users to describe the animation parameters clearly, specifying the properties of the object such as position, colour, size and image and defining the animation behaviours of the objects by predicates. Here is the summary of the project from our client Nir Lipovetzky: Planimation | Nir Lipovetzky.

Project Scope Statement

Project Name	Planmination
Description	Visualization the planning problem to reduce the difficulty of interpreting the planning process. The website has encountered the problem like missing visualization, unclear presentation of some family of problem and lack of some function such as download and history tracking.

Deadline	7/6/2024
Goal	The primary goal of the project is to create additional animation profiles to visualize a wider range of planning domains, allowing more planning problems can be effectively visualized and analyzed using Planimation. Each animation profile will specify how a particular planning domain should be visualized, including its initial states, goal states, actions, and constraints. Moreover, we want to improve the usability and functionality of Planimation from previous. For instance, develop the play, stop, playback control function for the visualizer, enable the users to download the animations generated by Planimation and implement a traffic handler component within Planimation to detect and mitigate DoS attacks, develop a build in tool that automatically generates Animation Profiles in PDDL (Planning Domain Definition Language) format for visualizing planning domains within Planimation, handle the scenario of overlapping animation objects and implement a history feature.
Deliverable(s)	1. Having more animation profile. 2. Website is working. 3. Able to download the animation. 4. Able to detect the DoS attack. 5. Automated generate the animation profiles. 6. Able to review the history.
Tasks (In-Scope Items)	1. **Creation of additional animation profiles**. By expanding the library of animation profiles, users is able to access the visualizations of more planning problems, enabling them to analyze and understand a wider variety of scenarios effectively. 2. **Create the multi-functional visualizer.** Users can play, stop and playback the animation as they want. 3. **Implement the GIF/images downloading** for users to download animations generated by Planimation, thus users can save the visualizations for further analysis, presentation or other purpose. 4. **Integrating a traffic handler** component to detect the DoS attacks. By monitoring the incoming network traffic, the pattern of DoS attack can be detected and thus mitigate, therefore the website is able to stay reliable and safe. 5. **Develop a build in tool that automatically generates Animation Profiles** in PDDL format for visualizing planning domains, reducing manual effort and ensuring consistency in profile creation. 6. **Implementing dynamic label calculation** to generate labels for animated objects to handle the overlapping of the objects. 7. **Implement a history feature** to allow the users to track and review previous outcomes.
Out-of-scope Items	1. **Support for Non-PDDL Based profiles**: Extending the Planimation visualizer system to support Planimation problem specifications not based on the PDDL format. It would require a fundamental change in how Planimation processes and visualizes planning problems outside the current project framework. 2. **Mobile Application Version**: Developing a mobile application for Planimation visualizer to allow users to create, edit, and view animations on mobile devices. It would require additional resources and expertise in mobile app development, which the current project scope needs to cover. 3. **Advanced Machine Learning for Predictive Analytics**: Implementing machine learning algorithms to predict possible outcomes based on historical data and

	visualize these predictions. This feature is more aligned with predictive analytics and goes beyond the primary function of our visualizing predefined Planimation problems. 4. **Integration with External Databases or APIs for Dynamic Data Retrieval**: Automatically pulling data (image, etc.) from external sources to dynamically update or create animation profiles. This could significantly complicate the current architecture and needs to be planned in the project's scope. 5. **Customizable User Interface for Animation Editing**: A highly customizable user interface allows users to drag and drop elements within the animation profiles. While beneficial, developing such a feature would significantly increase the project's complexity and be outside the intended scope.
Constraints	1. It needs to be live by 7/6/2024. 2. Our development team is not familiar with PDDL content.

Team

Stakeholders

Name	Contact	Role
Eduardo Oliveira	eduardo.oliveira@unimelb.edu.au	Subject Coordinator
Lucy Sparrow	lucy.sparrow@unimelb.edu.au	Subject Coordinator
Nir Lipovetzky	nir.lipovetzky@unimelb.edu.au	Industry Partner
Mylan Li	mylan.li@unimelb.edu.au	Mentor

Development Team

Name	Contact	Role
Huiyi Wang	huiywang1@student.unimelb.edu.au	Scrum Master, Tester
Lantian Yan	lantiany@student.unimelb.edu.au	Scrum Master, Product Owner
Haoran Li	haoran4@student.unimelb.edu.au	Frontend Developer
Yurun Wang	yurunw@student.unimelb.edu.au	Frontend Developer
Ninghai Zhang	ninghaiz@student.unimelb.edu.au	Backend Developer
Zixuan Huang	Zixuhuang@student.unimelb.edu.au	Database Developer

Roles and Responsibility

Role	Responsibilities	Name
Scrum Master	(i) Facilitate Scrum ceremonies (Sprint Planning, Daily Stand-ups, Sprint Review, Sprint	Huiyi Wang Lantian Yan

	Retrospective). (ii) Ensure the Scrum process is followed and remove impediments. (iii) Help the team to self-organize and focus on high-priority tasks. (iv) Act as a liaison between the development team and other stakeholders.	
Product Owner	(i) Define project vision and gather requirements. (ii) Manage and prioritize the product backlog. (iii) Communicate progress and feedback to the team and stakeholders. (iv) Accept or reject work results.	Lantian Yan
Frontend Developer	(i) Develop and implement user-friendly, responsive UIs. (ii) Collaborate with Backend Developers to integrate front-end and back-end code. (iii) Ensure the technical feasibility of UI/UX designs.	Haoran Li Yurun Wang
Backend Developer	(i) Design and implement server-side logic. (ii) Maintain and improve the database and server integrations. (iii) Ensure high performance and responsiveness to requests from the front-end.	Ninghai Zhang
Database Developer	(i) Manage database schema and optimize database operations. (ii) Ensure data integrity and security. (iii) Develop data storage solutions for the application.	Zixuan Huang
Tester	(i) Create detailed, comprehensive, and well-structured test plans and test cases. (ii) Perform thorough manual and automated tests to identify any bugs or issues. (iii) Document and report any defects to the development team. (iv) Monitor debugging process results.	Huiyi Wang

Links

Tools	Communication
GitHub	Slack
Trello	