Problem Statement and Goals
Software Engineering

Team 8, RLCatan
Matthew Cheung
Sunny Yao
Rebecca Di Filippo
Jake Read

Table 1: Revision History

Date	Developer(s)	Change
Sept. 17th	Matthew	Added preliminary problem statement for sections 1-1.4
Sept. 21st	Jake	Added goals and stretch goals sections and polished document
Sept. 21st	Jake	Added personal reflection
Sept. 21st	Matthew	Updated problem statement sections 1-1.4
Sept. 21st	Jake	Rewrote problem statement to focus more on the What than the How
Sept. 22nd	Rebecca	Added personal reflection

1 Problem Statement

1.1 Problem

Settlers of Catan is a strategic board game where players act as settlers, competing to build settlements and cities across the map by collecting and trading resources like wheat, wood, brick, etc. It's one of the most popular modern board games, with around 45 million copies sold. It also has a growing competitive scene, complete with Elo ratings in a manner similar to chess. Unlike chess however, there are no existing AI bots capable of playing *Catan* to the level of a human player. This is due to the large state-space of potential moves on a given turn, and the stochastic nature of the game with its dice rolls, as well as only partially observable information (e.g., opponents' hidden cards). This same nature makes it impossible for humans to calculate optimal strategies, as we can't see far enough ahead. A bot capable of playing *Catan* would be useful for both new and experienced players, aiding in learning the complexities of the game, and providing optimal moves in a given situation. It would also be perfect for *Catan* fans with no one else to play with.

This project proposes the creation of an AI-enabled decision-support tool for the board game *Settlers of Catan*. We aim to overcome the aforementioned bot development limitations by using computer-automated decision-making, supported by deep reinforcement learning (DRL). DRL is a variant of traditional Markovian RL that represents the learned policy as a deep neural network (DNN). This method works particularly well in dense problem spaces, such as seen in *Catan*. Due to the nature of this tool, it may even be able to discover entirely new strategies unknown to human players. The final product will be a digital twin, a tool that uses computer vision to observe a physical game and offer strategic suggestions to the player based on its underlying DRL model.

1.2 Inputs and Outputs

The system's high-level inputs are the game state information of a *Catan* match, including all board and player data. This information is captured in real-time by a camera and processed to create a digital representation of the board's state space. The high-level outputs are the next optimal move(s) for the player, delivered directly to their device. The AI can also simulate gameplay as a competing player. The tool also provides post-game advice, identifying past decisions that could have been made differently to alter the game's outcome.

1.3 Stakeholders

The primary stakeholders for this project are:

- Players of *Settlers of Catan*: The end-users who will use the AI for in-game and post-game analysis to improve their skills, or to train against skilled AI bots.
- Dr. Istvan David: The project supervisor who provides guidance, expertise, and oversight.
- The Project Team: The developers responsible for designing, implementing, and testing the software.
- The Department of Computing and Software (CAS) at McMaster University: The organization hosting the project, which benefits from the academic and technical achievements of its students and faculty.

1.4 Environment

The project requires a hardware and software environment to support its modules. On the software side, the system will need a game simulator for training, a reinforcement learning framework for the AI, a computer vision library to process video, and a user-facing application for the player's device. On the hardware side, the project requires a GPU to train the model, a device with a processor to run the AI, devices for the players to use the application and stream video of the game board, and a physical *Catan* board game for testing.

2 Goals

The following are the major goals for the project:

- Create a simulation environment for *Catan*, encoding game rules such that in a given state, a set of potential moves is returned.
- Devise starting states/confer with experts to find some initial board setups and corresponding strategies to aid the model's early learning.
- Utilize the simulation environment to train a deep reinforcement learning model to play *Catan* to at least the level of the average human player.
- Use computer vision/sensors to transfer the current state of a physical table-top *Catan* game to our simulation (Create a digital twin).
- Send the move generated by the model back to the user in some form.
- Develop a visualisation of the game on the digital twin's side to help explain actions.

3 Stretch Goals

In addition to the project's primary goals the following are stretch goals we will aim to complete if possible:

- Construct personas relating to various kinds of players and the strategies they will typically utilize in the game's trading system in order to improve the model's predictions.
- Improve the model to the extent that it performs better than the average *Catan* player, perhaps even finding strategies unknown to current competitive players.
- Generate explanations of "what could've been" following a game using an LLM.
- Further augment the physical board-to-simulation transfer via the use of multiple camera angles or smart glasses.

4 Extras

The extras for this project include:

- User instruction video
- Performance report

Appendix — Reflection

1. What went well while writing this deliverable?
2. What pain points did you experience during this deliverable, and how did you resolve them?
3. How did you and your team adjust the scope of your goals to ensure they are suitable for a Capstone project (not overly ambitious but also of appropriate complexity for a senior design project)?

Jake Read:

1. Most aspects of this deliverable went quite smoothly. We were able to fairly split the work, in such a way that we all contributed important sections, but were able to work in our own time, reducing time spent in meetings. I believe we were able to balance things in such a way that despite working on different sections, each of us still has a general understanding of the content in each part. What I'm happiest about is that we all seem to be working well together. There hasn't been any conflict between the four of us, and we were upfront about our schedules and availability and have stuck to the expectations we set. Whenever I ran into issues or had questions, it was easy to get a hold of someone in our Discord to help. The Discord server I set up has been perfect for organization, we have a general chat and various channels for sources, notes, resources, etc. Whenever we had any questions none of us could answer, we reached out to our TA or our supervising professor, who were happy to help.
2. We had a couple pain points during the project selection. We knew we wanted to work with AI/ML in the project, but not what project we wanted to do, so we began by running polls in our Discord server on the various potential projects. After voting and a subsequent meeting, we narrowed our options down to two. I was mostly interested in the *Catan* project, while the rest of the team was less certain which they preferred. We decided to schedule a meeting with the supervising profs of both projects, to get a more in-depth idea of what each one involved. This worked, and we ended up going with the *Catan* project. During this whole process, one teammate missed both meetings with little explanation, and was very slow to answer messages. After a discussion with the group, we agreed that we were concerned by the lack of communication, and decided to gracefully part ways with our fifth member.
3. Decisions surrounding scope were quite complex, as none of us had extensive prior experience with reinforcement learning. This made it hard to judge how long certain aspects of the project would take, so we

turned to our supervisor, Professor Istvan David. He was able to give us rough estimates regarding the scope/viability of various goals, which was a great help. The nature of our project made it quite simple to separate goals however, as the design process is rather modular (build simulation, train model, return data, etc.). The existing project description provided in the potential projects document also helped in this regard, as certain milestones were already marked as optional, making them clear contenders for stretch goals.

Rebecca Di Filippo:

1. One thing that went well was how we divided tasks. Each of us focused on our assigned sections individually, and then we reconvened before the deadline to review everything together and compare it against the rubric. This made the process smoother and gave us time to revise before submission. Communication also worked well: using Discord has been especially helpful. We set up polls, channels for resources, and held meetings there, which kept everyone organized and made decision-making easier.

2. The biggest challenge early on was finalizing both a project idea and a stable group. We had several meetings just to settle on which project to choose, and some of the people we initially planned to work with ended up leaving. I had to post multiple times on Avenue and Discord to fill our group, which was frustrating. We also decided to remove one member before the team deadline because they weren't showing up to meetings or contributing. Ultimately, once we solidified the group, things improved. For the project choice, we resolved uncertainty by holding discussions and using Discord polls to weigh different options. Meeting with supervising professors also helped us make an informed decision.

3. At first, we considered including more ambitious features, such as having the agent take in real-time data directly from a physical board using smart glasses or a video stream. After reviewing the overall project scope, we realized these features would be too complex for the time we have. Instead, we focused the core project on developing the reinforcement learning model itself, with simpler inputs like static pictures or descriptions of the game state. More advanced features, such as real-time board capture, multiple camera angles, and post-game explanations generated by an LLM, were moved to stretch goals. This way, the project remains challenging and meaningful, but realistic within the Capstone timeline.