**(E.1) Glossary**

* Clear and precise definitions of all the vocabulary specific to the application domain, including technical terms, words from ordinary language used in a special meaning, and acronyms.
* It introduces the terminology of the project; not just of the environment in the strict sense, but of all its parts.

Catan – A strategy board game called Settlers of Catan where players collect resources, build roads/settlements, and trade to earn points.

AI(Artificial Intelligence) – Field of computer science / engineering that focuses on creating systems capable of performing tasks that usually require the intelligence of a human.

RL(Reinforcement Learning)- A type of machine learning where an agent learns how to make decisions by interacting with an environment and receiving rewards or penalties for actions taken.

Digital twin- A digital system that mirrors a physical one. In this project, it specially means a digital representation of the physical catan game board, updated in real time.

CV(Computer Vision)- Enables a system to interpret and process visual data. In the case of our project, it’s a camera that can understand and track the physical catan board state.

LLM (Large Language Model)- A machine learning model trained on a very large amount of text data to generate and understand natural language.

Game state- The current configuration of the game, including player resources, board layout (roads, settlements, cities), dice rolls, and ongoing trades.

**(E.2) Components**

* List of elements of the environment that may affect or be affected by the system and project.
* It includes other systems to which the system must be interfaced.
* These components may include existing systems, particularly software systems, with which the system will interact — by using their APIs (program interfaces), or by providing APIs to them, or both.
* These are interfaces provided to the system from the outside world.
* They are distinct from both: interfaces provided by the system to the outside world (<<s3>>); and technology elements that the system's development will require (<<p5>>).

Physical Catan board and pieces – The physical game setup that the system observes.

Players – Human participants who interact with the physical game and receive decision support.

Cameras/Sensors – Hardware that captures the physical board state for the digital twin.

OpenAI Gym – Provides a training/testing environment for reinforcement learning agents.

Reinforcement Learning Agent – Learns strategies through the simulator and connects with the digital twin for real-time decision support.

Visualization Interface – Displays the current game state and AI recommendations to users.

Optional components:

* Smart glasses – Provide players with an augmented view of the game and recommendations.
* LLM service/API – Generates natural language explanations of strategies.

**(E.3) Constraints**

* Obligations and limits imposed on the project and system by the environment.
* This chapter defines non-negotiable restrictions coming from the environment (business rules, physical laws, engineering decisions), which the development will have to take into account.

Game rules of Catan – The RL agent must strictly follow the official rules of the game.

Real-time operation – The system must process board states and provide recommendations fast enough to be useful during live play.

Camera limits (Computer Vision) – Accuracy of game state detection is restricted by available hardware (resolution of camera, field of view, lighting).

Simulator environment – The RL agent is limited to the APIs and mechanics provided by the Catan simulator

Computational resources – Training and running the RL agent is bounded by available GPU/CPU capacity.

Timeframe– The project must be completed within the allocated time frame.

**(E.4) Assumptions**

* Properties of the environment that may be assumed, with the goal of facilitating the project and simplifying the system.
* It defines properties that are not imposed by the environment (like those in <<e3>>) but assumed to hold, as an explicit decision meant to facilitate the system's construction.

**Players will follow standard Catan rules** – you assume human players won’t cheat or make illegal moves.

**Stable lighting and camera angle** – the computer vision module assumes it can reliably see the board, even if real-world conditions could vary.

**Network and device reliability** – you assume the player’s device and connection work well enough for real-time suggestions.

**(E.5) Effects**

* Elements and properties of the environment that the system will affect.
* It defines effects of the system's operations on properties of the environment.
* Where the previous two categories (<<e3>>, <<e4>>) defined influences of the environment on the system, effects are influences in the reverse direction.

**Player decision support** – the AI provides move suggestions, affecting the decisions players make in real-time.

**Learning and adaptation** – the RL agent improves over time, indirectly affecting the level of challenge/advice for players.

**Post-game analysis** – feedback/suggestions might influence how players approach future games.

**Device usage** – the system uses computational resources on player devices or servers for inference and visualization.

**Game pacing** – real-time suggestions could speed up or slow down the flow of the game.

**(E.6) Invariants**

* Properties of the environment that the system's operation must preserve, i.e., properties of the environment that operations of the system may assume to hold when they start, and must maintain

**Board orientation is fixed** – the physical board doesn’t get flipped or moved, so the computer vision module can track it accurately.

**Player order remains constant** – the sequence of turns doesn’t change unexpectedly.

**Player set is fixed** – no new players join, and no existing players leave mid-game.

**Game components stay in place** – settlements, roads, and resources aren’t physically moved outside of normal game actions.

**Camera/viewing angle remains stable** – the vision system continues to see the board from the expected perspective.

**(P.1) Roles and personnel**

* Main responsibilities in the project; required project staff and their needed qualifications. It defines the roles (as a human responsibility) involved in the project.

**Team Leader (Jake)** – schedules meetings, coordinates team, point of contact with supervisor/TA.

**Notetaker (Rebecca) –** creates agendas, takes notes, updates Kanban board.

**IT/DevOps (Sunny)** – manages GitHub repo, handles technical issues.

**Researcher (Matthew)** - responsible for locating relevant academic papers and resources, researching unfamiliar topics, and providing insights to help the team better understand and approach new concepts.

**\*Supervisor – Dr. Istvan David**: provides guidance, expertise, and oversight for the project. Offers advice on technical decisions, project scope, and milestones.

**\*Teaching Assistant – Tiago de Moraes Machado**: provides assistance with course-related questions, clarifications, and support for project development. Acts as a point of contact for feedback and resources.

**(P.2) Imposed technical choices**

* Any a priori choices binding the project to specific tools, hardware, languages or other technical parameters.
* Not all technical choices in projects derive from a pure technical analysis; some result from company policies.
* While some project members may dislike non-strictly-technical decisions, they are a fact of project life and must be documented, in particular for the benefit of one of the quality factors for requirements: "requirements must be justified".

**Programming languages:** Python for backend/AI, JavaScript and React for frontend.

**AI framework:** Reinforcement learning agent trained using OpenAI Gym Catan simulator.

**Computer vision:** OpenCV and/or YOLOv9.

**Version control and collaboration:** GitHub with branches, pull requests, CI/CD pipelines, Kanban board.

**Coding standards:** PEP8 for Python, Google style guide for JS/React.

**License:** MIT license for the project code.

**(P.3) Schedule and milestones**

* List of tasks to be carried out and their scheduling.
* It defines the project's key dates.

**(P.4) Tasks and deliverables**

* **This is the core of the Project book**.
* It details the individual tasks listed under <<p3>> and their expected outcomes.
* It define the project's main activities and the results they must produce, associated with the milestone dates defined in <<p3>>