Red Monopoly Scrum

2.1. Structure and Operation of the Scrum Team

- Our Scrum team was originally composed of five members: Rodrigo, Zheng, Helgi, Sasha, and Adam. Originally, the distribution of roles was as follows.
    - Scrum Master: Rodrigo (previously), Helgi (current)
    - Team Advocate: Zheng
    - Documentation Lead: Zheng
    - Product Owner: Sasha
    - Git Expert: Helgi
- With Rodrigo later dropping out, Helgi took on the role of Scrum Master. The rest of the roles that were assigned, stayed the same from sprint 2 until the product release.

Primary Channels

- **Discord Server**: We set up a dedicated Discord workspace with channels for general discussion, backlog refinement, development help, and testing feedback. Early on our voice-call meetings on Discord were sporadic—sometimes we'd hop on for 10–15 minutes when someone got stuck, other times nothing for a week. As the project progressed, we formalized two weekly voice-call slots (typically mid-week and pre-planning), though we never held stand-ups every day.

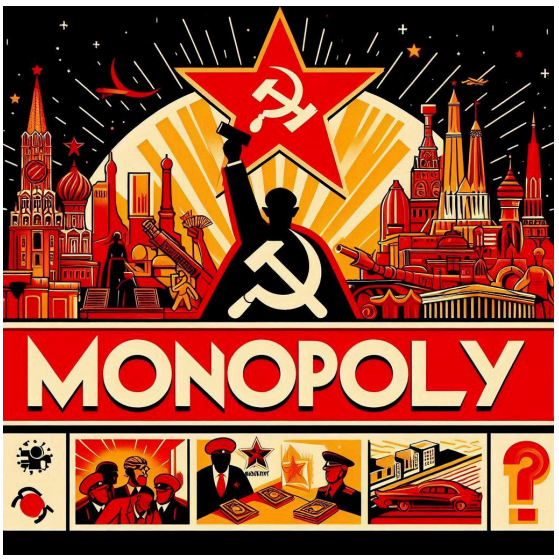
- **WhatsApp Group Chat**:This ensured urgent blockers weren't lost in long Discord threads and allowed mobile notifications when we were away from our desks.

Online vs. In-Person Split

- **Online (75%)**: The bulk of design discussions, code reviews, and backlog grooming took place remotely via Discord or WhatsApp. We found this flexible for juggling classes and personal schedules, and could vet pull requests or brainstorm UI flows without needing everyone in the same room.

- **In-Person (25%)**: Whenever we all happened to be on campus together—most reliably on **Mondays**, when our timetables aligned—we grabbed a corner table in the university cafeteria or reserved a small lab room. These sessions were used for doing paired programming, or preparing pitch presentations.

Project Identity



- Our original logo, and idea.

Project Definition

Own the USSR with strategic moves, historical twists, and satirical fun in this Soviet-themed Monopoly!

Why is it appealing?
1. Historical Twist: It offers a fresh take on the classic game by incorporating significant historical events and figures from the Soviet era.
2. Educational Value: Players can learn about the USSR's history and economics in a fun, interactive way.
3. Strategic Gameplay: The game introduces new mechanics like resource management and economic collapse events, adding depth and complexity.
4. Humor and Satire: The incorporation of humorous and satirical elements makes the game entertaining and light-hearted.
5. Nostalgia: For those who have a nostalgic interest in Soviet-era culture and aesthetics, the game offers a trip down memory lane.

Target audience:
Board Game Fans,History Lovers and Teachers, Families, Students, Retro Fans, Weird Communists. Age: 10+

Features:

- Cities & Landmarks: Replace traditional properties with major Soviet cities (Moscow, Leningrad, Kiev) and famous landmarks (Red Square, the Kremlin, Lenin's Mausoleum).
- Republics: Each color set could represent different Soviet republics (e.g., Baltic States, Central Asian Republics).

- Industrial & Agricultural Sites: Instead of railroads and utilities, use industrial complexes (factories, mines) and collective farms (kolkhozes, sovkhozes).

- Roubles and Vouchers: Replace Monopoly money with Soviet roubles and special vouchers that were used during the Perestroika period.
- Resource Tokens: Introduce resource tokens like coal, steel, grain, and oil that players can trade or use to develop properties.

- Perestroika Perks: Cards with benefits from Gorbachev's policies (e.g., "Your business thrives due to new economic policies, collect 200 roubles").
- Politburo Decisions: These cards could involve unexpected events (e.g., "Summoned to a Politburo meeting, miss two turns", or "Receive a state award, collect 50 roubles").

Unique Game Mechanics
- Economic Collapse Events: Introduce events that reflect the instability and collapse of the USSR (e.g., hyperinflation events that devalue money, or resource shortages affecting property development).
- Spy & Propaganda Missions: Players could draw cards that involve espionage or spreading propaganda to influence other players' positions on the board.

Themed Player Tokens
- Historical Figures: Players can choose tokens representing notable Soviet figures (Lenin, Stalin, Gorbachev, etc.).
- Iconic Items: Other tokens could include a hammer and sickle, a Soviet tank, a Sputnik satellite, and a Matryoshka doll.

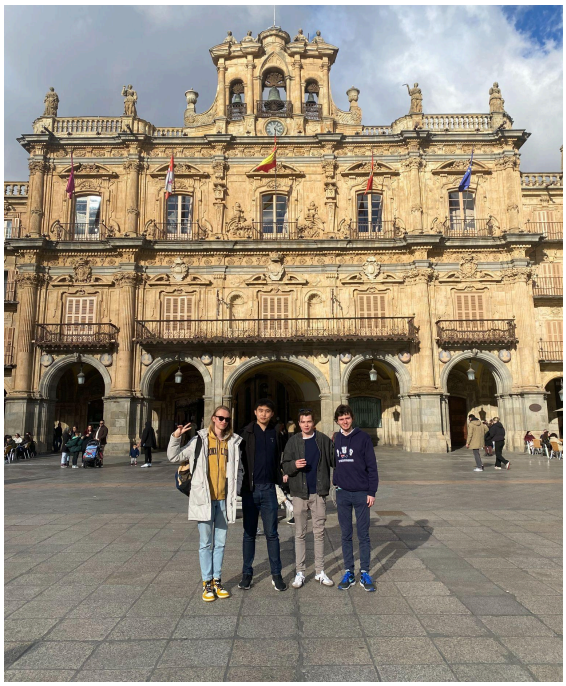
Endgame Scenario
- Collapse Simulation: To win, players must strategically navigate the collapse, maintaining stability in their assets while others lose theirs due to the economic meltdown.

Initial Organisation

As our group doesn’t have any North person, we will collectively make an effort to step out of our comfort zone and ensure the project goes smoothly. To address this, we will establish clear goals and deadlines to keep everyone on track, and designate rotating leadership roles to share responsibilities and foster initiative. Additionally, we will encourage open communication and create a supportive environment where every member feels comfortable contributing ideas, regardless of their natural tendencies. This approach will help us balance our strengths and overcome any potential biases or weaknesses as a team.

Team building activity:

- As our initial team-building exercise, we made a trip to Avila and Salamanca by car, as a day trip.



Evolution from the Original Agile Inception Document

When we drafted our Agile Inception document in January 2025, we identified three key areas to strengthen our team's way of working:

- **Involvement**
  Every team member would take turns facilitating Sprint Planning, Reviews, and Retrospectives, so that no single person became a bottleneck.

- **Clear Goals & Deadlines**
  We agreed to define sprint objectives and hard cut-off dates for story completion, to keep scope creep in check.

- **Inclusive Communication**
  Quieter or less-experienced members would be explicitly encouraged to share ideas, ensuring a wider range of perspectives.

As the project unfolded, new challenges surfaced that our original plan hadn't anticipated:

- **Channel Overload & Fragmentation**
  Managing discussions across Discord, WhatsApp, GitHub, and email led to missed updates and context-switch fatigue.

- **Inconsistent Task Completion**
  Even with deadlines, work clustered toward the end of sprints, creating last-minute

rushes and risking quality.

- **Hybrid Context-Switching**
  Our mix of mostly online and less in-person collaboration fragmented time and sometimes stalled pair-programming momentum, making it harder to work as a team.

- **Other issues**

  With dropouts and/or uneven work load provided by team members, it was required for the other team members to step up and share the uncompleted tasks.

2.2 User Stories

Below is a comprehensive list of our user stories, their priority, going from P0 to P(n), as well as their weight. They are ordered by iterations, with the colors shown indicating the separation of iterations.

User Story	Priority	Weight	Status
As a user I would like to roll a dice	P0	M	DONE
As a user I would like to land on a tile	P0	M	DONE
As a user I would like to have a basic interface to navigate on the game	P0	M	DONE
As a player, if I land on some other's player properties I would need to pay him rent	P0	S	DONE
As a player, I would like to see bank balances of each player and their properties	P0	M	DONE
As a player, I would like to get the option of buying a property yes/no when I land on them	P0	M	DONE
As a developer, I want to implement random chance events when the player lands on chance tile	P1	M	DONE
As a developer, I want to implement jail mechanics	P0	S	DONE
As a player, I would like to see a infobox for each player displaying bank balance, properties owned..	P1	M	DONE
As a player, I want to be able to choose my name before the game starts	P1	S	DONE

As a player, I want the game to end when only one player remains so that a winner can be declared.	P0	M	DONE
As a player, I want to see the game characters moving on the board.	P1	M	DONE
As a user, I would like a basic board interface in order to visualize the current game state.	P0	XL	DONE
As a player, I would like to receive money every time I pass "Go".	P2	S	DONE
As a developer I want to implement railroads logic	P2	S	DONE
As a user, I would like the tiles bigger, with correct colorpattern and correc tile names	P0	L	DONE
As a player, I would like to play against machine players	P1	XL	DONE
As a user, i would like to save/load the game	P0	L	DONE
As a user, i would like to have a automatic players with 3 difficulty settings	P0	L	DONE
As a user i would like the board to look more fitting for a game	P0	M	DONE
As a user I want to replace the swing boxes with images so that we can start designing every tile	P2	M	DONE
As a developer, I want to implement Bank Loans & Interest system	P2	M	DISCARDED
As a user, I would like to undo my last action after my turn	P0	M	DONE
As a player, I want to be able to choose an avatar to play before the game starts	P2	S	DONE
as a user, I would have to pay tax when I land on the tax tile	P2	S	DONE
As a player, I want to see which properties are bought by who on the board in a clear manner	P1	M	DISCARDED

As a user, I would like to play in multiplayer with host/server	P0	XL	DONE
As a player, I would like to interact with the special cards in the game	P1	XL	DISCARDED
As a user, I would like to be able to pay out of jail instead of rolling the dice	P1	M	DISCARDED
As a developer, I would like to fix MVC logic	P0	L	DONE

2.3 Sprint Plannings

Throughout the development of our project, we held regular sprint planning sessions to define clear objectives, prioritize user stories, assign tasks, and set acceptance criteria. These planning sessions evolved as we progressed through each sprint, incorporating feedback and lessons learned from previous reviews and retrospectives.

Early in the project, we selected our Scrum Master (Rodrigo) and Product Owner (Zheng), and collaboratively defined our initial backlog. Our user stories ranged from UI fundamentals (like dice rolling, background music, and navigation) to complex gameplay features (like property ownership, AI behavior, and game state persistence). As our understanding of the game's scope deepened, we adapted our Definition of Done to include both functional correctness and integration with the UI.

In later sprints, planning became more nuanced, with time set aside for UI refinement, AI decision-making, and thematic consistency. We prioritized not only what should be built, but how it should feel — leading to the implementation of a Soviet-themed visual overhaul and player customization features. Planning accuracy and confidence improved as we gained hands-on experience.

2.4 Sprint Reviews

After each sprint, we conducted reviews to evaluate the progress made toward our sprint goals. These meetings were essential to measure what was completed, showcase working features, and identify items requiring further iteration.

Early reviews focused on functionality — validating dice rolling, music playback, menu responsiveness, and initial game flow. Later reviews shifted toward complexity, ensuring game logic (buying property, paying rent, drawing chance cards, going to jail) worked as expected and appeared correctly in the GUI.

By Sprint 3, our reviews placed heavier emphasis on UI feedback, as we had moved from console-based output to full board visualization. We also reviewed visual responsiveness, animation triggers, and user notifications. Sprint 4 introduced save/load functionality, undo features using the Command pattern, and AI difficulty selection — all of which were reviewed both in terms of functionality and user experience.

In our final sprint, reviews centered on theme integration and gameplay polish. We confirmed that Soviet-themed visual assets appeared correctly, AI strategies differed as intended, and custom player icons persisted across save/load cycles.

Across all sprints, our Definition of Done evolved to include:

- Functionality working without exceptions
- Visual confirmation and feedback to the user
- Integration with existing features
- Responsiveness and UI polish

2.5 Sprint Retrospectives

Retrospectives were a critical part of our process, allowing us to reflect on team dynamics, technical challenges, and process improvements. These reflections helped us adapt our use of Scrum to better suit the realities of an academic group project.

In our early sprints, we struggled with communication and time management. Meetings were sparse, and much of the work was done last-minute. However, we still managed to produce a working prototype and distribute responsibilities using branches.

Midway through the semester, we noticed fluctuations in participation and attendance. Some members were more engaged than others, which occasionally slowed integration or forced other team members to pick up additional work. We responded by improving communication, assigning clearer roles, and documenting design decisions.

In the later sprints, especially Sprint 4 and 5, our retrospectives highlighted how much we had matured as a team. We successfully tackled more advanced technical tasks like undo systems, persistent state management, and AI strategy differentiation. However, we also recognized that better testing coverage and more frequent check-ins would have improved stability and helped align implementation details earlier.

Overall, our retrospective process enabled us to learn from our mistakes and build on our successes. While not everything went perfectly, we adapted Scrum in a way that worked for us, and ended the semester with a mostly feature-rich, cohesive, and themed game.

2.6 Product Backlog

Our Product Backlog constantly evolved, reflecting both our ambitions and the realities of time and resources. Midway through the project, new crucial requirements were introduced—save/load state, undo moves via the Command pattern, and AI difficulty selection—which reshaped our priorities and extended the backlog beyond its initial scope.

We also drafted thematic Soviet-inspired user stories—Perestroika reform cards, Sputnik launch bonuses, Victory Day celebration tiles—to deepen the game's historical flavor. However, these extras often vied for attention alongside must-have functionalities: core turn logic, property and railroad transactions, rent payment, chance and community chest mechanics, and robust UI integration.

Through regular backlog refinement sessions, we learned to juggle thematic creativity with core requirements. High-impact features like advanced AI learning behaviors and animated tile zoom were postponed or shelved when deadlines tightened. By focusing on mandatory game mechanics and ensuring thematic cohesion in our UI, we delivered a polished, fully playable Soviet-themed Monopoly—complete with all essential features, even if a few extras had to wait for future iterations or be discarded.

2.7. Sprint Backlog

User Story	Assigned to	Status
As a user I would like to roll a dice	Helgi	DONE
As a user I would like to land on a tile	Zheng	DONE
As a user I would like to have a basic interface to navigate on the game	Zheng	DONE
As a player, if I land on some other player's properties I would need to pay him rent	Sasha	DONE
As a player, I would like to see bank balances of each player and their properties	Sasha	DONE
As a player, I would like to get the option of buying a property yes/no when I land on them	Sasha	DONE
As a developer, I want to implement random chance events when the player lands on chance tile	Sasha	DONE
As a developer, I want to implement jail	Zheng	DONE

mechanics		
As a player, I would like to see a infobox for each player displaying bank balance, properties owned..	Zheng	DONE
As a player, I want to be able to choose my name before the game starts	Sasha	DONE
As a player, I want the game to end when only one player remains so that a winner can be declared.	Zheng	DONE
As a player, I want to see the game characters moving on the board.	Sasha	DONE
As a user, I would like a basic board interface in order to visualize the current game state.	Zheng	DONE
As a player, I would like to receive money every time I pass "Go".	Zheng	DONE
As a developer I want to implement railroads logic	Sasha	DONE
As a user, I would like the tiles bigger, with correct color pattern and correct tile names	Sasha + Helgi	DONE
As a player, I would like to play against machine players	Zheng	DONE
As a user, i would like to save/load the game	Zheng	DONE
As a user, i would like to have a automatic players with 3 difficulty settings	Sasha + Zheng	DONE
As a user i would like the board to look more fitting for a game	Helgi	DONE
As a user I want to replace the swing boxes with images so that we can start designing every tile	Zheng	DONE
As a developer, I want to implement Bank Loans & Interest system	N/A	DISCARDED
As a user, I would like to undo my last action after my turn	Zheng	DONE
As a player, I want to be able to choose an avatar to play before the game starts	Zheng + Sasha	DONE

as a user, I would have to pay tax when I land on the tax tile	Sasha	DONE
As a player, I want to see which properties are bought by who on the board in a clear manner		DISCARDED
As a user, I would like to play in multiplayer with host/server	Helgi/Zheng	DONE
As a player, I would like to interact with the special cards in the game		DISCARDED
As a user, I would like to be able to pay out of jail instead of rolling the dice		DISCARDED
As a developer, I would like to fix MVC logic		DONE

2.8. Description of the Work Performed by Each Group Member

Helgi (Git expert + Scrum Master)

- Served as Scrum Master, facilitated sprint planning, daily stand-ups, and retrospectives to keep the team on track.

- Led some UI/UX efforts—board visualization, dice animations, player info panel styling, and thematic integration of Soviet-era assets, as well as ensuring multiplayer works with the usage of Java Sockets.

- Documented design guidelines and provided peer support for merge conflicts and Git workflows. Ensured correct Git and GitHub usage throughout the project. Presented in Sprint Presentations.

Zheng (Team Advocate + Documentation Lead)

- Acted as Documentation Lead: Authored some of sprint planning, review, retrospective, and backlog evolution sections, and oversaw final report assembly, as well as being key contributor for the Scrum document, the UML document, and the Github README file.

- Provided the generation of UML Class diagram, sequence diagram, required in the homework earlier in the sprints.

- Core Logic Implementation: Developed key game mechanics, such as turn processing (TurnHandler), jail and chance card logic, and Gson-based serialization for save/load functionality. Implemented the GUI aspect of the multiplayer version too. Proposed MVC in earlier sprints, and set up the primordial classes for the project's backbone.

- Additionally, designed the card tile images using Canva that were later used on the board.

- Team Advocacy & Communication: Synthesized feedback, ensured thematic consistency, and championed the player's perspective in design decisions. Presented in sprint pitch presentations.

- Curated and integrated tile imagery, avatar icons, and background music to create an immersive game experience.

Sasha (Product Owner)

- Product Owner focusing on core mechanics: property and railroad purchase logic, rent payments, tax and community-chest flows,dialogue windows, AI strategy difficulty levels and its integration.

- Additionally, added the soviet themed icons for 8 players.

- Command Pattern & Testing: Implemented undo functionality via the Command pattern and designed unit tests for dice rolls, tile actions, and persistence.

- Backlog handling: Owned end-to-end backlog management—refined and weighted user stories, assigned tasks through team collaboration, tracked progress across workflow stages (Ready → In Progress → Review → Done), and ensured adherence to acceptance criteria while resolving bottlenecks to maintain sprint momentum.

- Wiki management: Fully managed wiki documentation in the last sprints.

- Quality Assurance: Identified and resolved gameplay bugs, and ensured code quality and stability. Ensured sprint reviews and sprint retrospectives to be of good quality.

- Sprint Demos: Presented feature demos during sprint reviews and collected team feedback for continuous improvement.

- Did documentation regarding the UML (point 3)

Adam

- Project participation was limited, due being an Erasmus student and having to go back to home country due to family issues.

2.9. Conclusions and Self-Assessment

Over the semester in our Software Engineering course, we transformed from a five-member group into a core team of three active contributors. The gradual attrition, dropping to four, then three regular participants, combined with widely varying individual schedules, made coordinating meetings and maintaining momentum especially challenging during the first few sprints. Nonetheless, these hurdles forced us to refine our communication, planning, and collaboration practices in ways that ultimately strengthened both our process and our product, making it possible to deliver an adequate final product.

Key Lessons Learned

- **Adaptability in Team Size**
  As attendance fluctuated, we learned to re-prioritize backlog items and reassign tasks on the fly, ensuring that essential work always had clear ownership, even when only three of us could attend a planning session.
- **Value of Early Structure**
  Initial sprints suffered from sporadic meetups and unclear agendas. This was gradually fixed in later sprints of the project, with more consistent meet-ups or clearer communication between team members.

Generally, between the active members, communication was appropriate and no major issues arised.