

Reference: QSE:PROST

Topic: PROST: A Production Planning Game for Humans and AI Agents

Course Type: Bakk-/Master Thesis, ASE or CS Project

Start: As soon as possible

End: To be defined

Contact: Stefan Biffl (stefan.biffl@tuwien.ac.at)

Background

PROST (Simulation der PROduktionsSTeuerung) is a production planning game that has been used in teaching at TU Wien since the early 1990s [1], also at several universities and companies. In the simulation, teams run competing manufacturing companies, period by period: they bid for customer contracts, read the product's *Gozinto* graph to see which parts and materials are required to produce a pump or a motor, buy materials, schedule their machines and deliver on time to earn cash and image. Machine breakdowns and late deliveries challenge the plan from outside, and because everything is connected, one weak decision may spread through the production outcome (Figure 1).

PROST: run a manufacturing company, period by period

Teams compete for contracts and must turn them into delivered products on time — every decision affects the next.

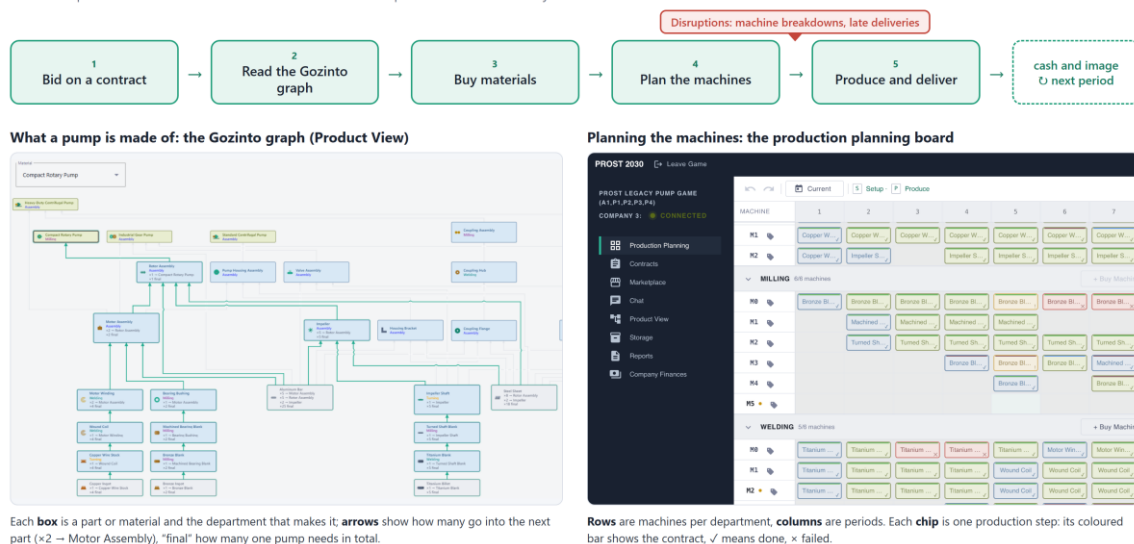


Figure 1: The PROST game loop.

Left: the Gozinto graph of a pump in the Product View.

Right: the production planning board during the first official test game.

In 2026 a student team rebuilt the legacy software as a modern web application (Spring Boot, Angular): modeling real products and materials, teams planning together in real time on a shared board, configurable game templates, and rule-based bots that play as companies. Test games with students showed good learning effects.

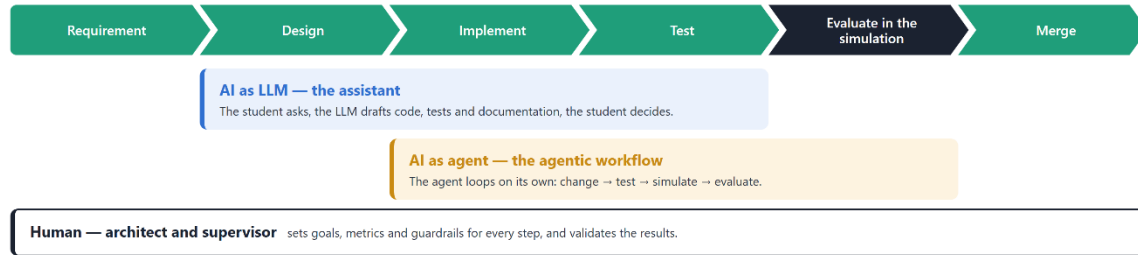
PROST 2030 shall open the game to AI functions on several levels (Figure 2): (i) AI agents play, (ii) help the game master create products and scenarios, (iii) explain after a game why one team did better than another, and (iv) help build and improve the software itself, with humans as architects and supervisors, because AI may fail silently.

Where AI comes into PROST

Students develop PROST with AI — and build AI into the game.

AI fails silently. The human stays the architect:
AI proposes and executes, people decide and validate.

DEVELOPING PROST



AI INSIDE THE GAME

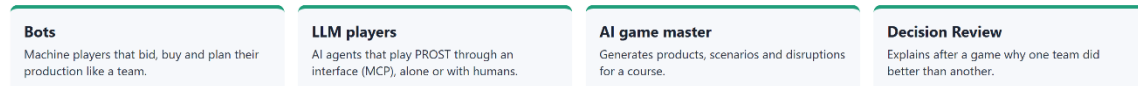


Figure 2: Where AI comes into PROST: developing with AI as an LLM assistant or as an agent under human supervision, and AI inside the game.

Goal

Develop PROST into an **AI-ready platform for teaching and research**. Near-term goal is a stable and polished game on the TU Wien infrastructure for the next PROST course in March 2027. Mid-term goal (June 2027) is a testbed where humans and AI agents plan production side by side, as input to a service that other universities can use. The candidate tasks below are starting points, and your own ideas are welcome [2].

Tasks

Most tasks can be done as a Bakk thesis or as a project, some also as a Master thesis. As a thesis, a task adds a research question and an evaluation, for example a usability study or a benchmark.

Start in October 2026: Work to complete by January 2027

1. **Agentic testing.** AI agents write more and more code, but who checks that it works? Design a skill set for AI agents to write, run and judge tests, and complete the PROST test suite.
2. **UI and player guidance with AI.** Make PROST easier to learn and to play: a clearer interface, including the contracts page and the findings from the test games, handbooks for players and admins, and AI that guides players through the game and what they need to know.
3. **Smart production planning.** Turn the planning board into an assistant that thinks ahead, using real production-planning methods. It simulates the plan into the coming periods, warns before materials run out or a contract is going to be late, and shows how far each contract has come.
4. **Deployment and operations.** Take PROST live for the summer course: a pipeline that tests every change and deploys it automatically to the TU server, with HTTPS, backups and monitoring, and a load test showing that 80 to 100 players can play at once.

Start from February 2027 (building on the course version to be applied in 27S)

5. **LLM players.** Can an LLM run a factory? Give AI agents an interface to play PROST (e.g., via MCP), and let them compete against, or team up with, humans and bots. A first prototype already lets an LLM plan the production schedule.
6. **AI as game master.** Let AI run the game like a good game master: watch how the teams are doing and shape the game while it runs, with disruptions, market changes and new contracts that keep it challenging. Build on a variability model of everything that can change in a game.
7. **PROST as a service.** Bring PROST to other universities as a service instead of sharing code. Work out how partner courses shall get access to their own games, how PROST can be hosted and supported, what it will cost, and how student data will be protected.

Expertise

For this topic, a set of skills is recommended (at least two are mandatory).

- Web application development: Java/Spring Boot or TypeScript/Angular.
- UI/UX design, usability evaluation and accessibility (WCAG 2.2 AA).
- Software testing and quality assurance.
- LLMs and AI agents: prompting, tool use, MCP, agentic workflows.
- Simulation, data analysis and empirical evaluation, e.g., experiments or user studies.
- DevOps: CI/CD, Docker, databases.
- Interest in production planning, industrial engineering or teaching.

References

- [1] D. Biberschick, Matyas, K. (1992). [Planspiel PROST – Simulation der Produktionssteuerung](#). In: Dette, K., Pahl, P.J. (eds) Multimedia, Vernetzung und Software für die Lehre. Reihe Mikrocomputer-Forum für Bildung und Wissenschaft , vol 4. Springer.
- [2] S. Biffli, Marhold, K., Musil, J., Meixner, K.: [Business intelligence for value-based process and systems engineering and operation with procan.do](#). In: Business Intelligence for Multi-Domain Systems Engineering and Operation. Springer Cham (2026). > Section 5.2.x