Realtime Simulation

Necessary requirements

- The term paper will be written in Java >=21 (or another language on top of the JVM if agreed), and managed using Maven.
- There will be demonstrable project progress on GitLab. Ongoing commits (each finished feature with description) will be made by all team members. Caution: inadequately few commits (very tentative: less than 10) will usually raise (justified) suspicions of plagiarism among the practitioners!
- Both project participants must be involved in the creation of the GUI. At least one non-trivial window will be created without the use of Designer in Netbeans or any other "clickable" tool.
- In the project, the student will demonstrate the ability to use threads correctly (this does not include the use of, for example, the Timer class). For example, by creating a real time clock that interacts with the game flow.
- Several non-trivial classes will be covered with unittests or a more comprehensive functional test, using any testing framework.
- Loggers will be used. Log messages will be enabled or disabled by a parameter at runtime, or other user interaction (not by editing the source code).
- All non-trivial public elements in the program must have a meaningful Javadoc. For example, simple getters and setters without side effects are considered trivial; they do not need a Javadoc.
- All non-trivial parts of the code will be commented appropriately.
- The project Wiki must contain an up-to-date and usable user manual and technical documentation of the program - its features, the project structure, the technologies used as described above. Attention, this is not Javadoc!