1DL251 SEPM TEAM-B: D1 - PROJECT DELIVERABLE

Development team and the skills/strengths they can bring to the team:

- Enzel, Viktor (VE) - good with Python and Git
- Gaide, Maxime (MG) - good with prototyping
- From, Gustav (GF) - good with validation/testing
- Ingvast, Pelle (PI) - good with Python
- Kerle, Laurin (LK) - good with organizing and planning
- Ross, Adam (AR) - good in Python, git & team software development

Primary development tools:

Communication: Slack & email	- PI is responsible
Task management: Trello	- GF is responsible
Team meetings: Doodle	- LK is responsible
Designing: DIA	- MG is responsible
Development: GitHub (platform), PyCharm (IDE)	- AR is responsible
Accounting & marketing: The uppsalines website	- VE is responsible

Further information on team member responsibilities:

Each aforementioned team member shares responsibilities for all software development roles/tasks such as management, planning, designing, development, testing, reviewing, and deployment of the game engine development for each sprint. What each developer does in respect to each development role per sprint/story/task can be decided during the planning for said sprint/story/task by the team.

All developers should be involved in the sprint planning, and the initial designing for each sprint, and should all be involved in some degree in all aspects of the software engineering development project when required to and should have completed every type of role at least once during the period of the course.

Team-B rules:

Team work:

- During Meetings members who are behind on their schedule can ask for help. If they don't do this or are constantly behind, the rest of the team can give a warning. After three warnings a vote can be cast to expel this person. Every other member must vote yes for this vote to be successful.

- Every team member is required to attend at least one team meeting per week aside from the scrum and TA meetings.
- Punctuality is a priority at all pre-arranged meetings, whether if it's a full team meeting, or a portion of the team.
- The team aims to attend all activities with a minimum of 80% of the members.
- Team meetings will either be decided during the prior team meeting or by the team member responsible for organizing team meetings. Each team meeting must be at an appropriate time that suits all team members. Each team meeting time and place must be made available to all team members in a comment on the #meetings channel of the team slack workspace. The comment must be made within a reasonable amount of time prior to the meeting by the team member responsible for team meetings planning. Each member is expected to respond to the doodle within a reasonable time-frame.

Git, python and general development conduct:

- Always 'git pull' before you 'git push'.
- Keep git commit headers brief and use imperative grammar (add, not added)
- Include in each commit header either '#implement', '#refactor', '#fix' to define the development (programming) pushed to the git GameEngine repository.
- For peer programming, which is implored, include in each commit a 'Co-authored-by: "user name" <GitHub account email>'. Refer to this webpage: https://github.blog/2018-01-29-commit-together-with-co-authors/
  or use this for a simple work around: https://desktop.github.com/features/co-authors/
- Each development task will require a reviewer that has not worked on that task to validate that the development is completed and passes any and all requirements. Is also responsible for the merging of the branch to the master repository and any and all merge conflict resolutions if task has a branch. Branches will be used where practical.
- Peer programing/task work: each member should work with every other team member at least once on a specific, separate task. That is, complete at least five tasks each, each task being with at least one other team member that you haven't worked with on any other prior task
- Pylint correctness (PEP8 style - PyCharm will always hint with squiggly underlines if the code is not Pylint correct; 1 empty line between methods, 80 chars max per line, etc.) There is also a website for checking pylint correctness: http://pep8online.com
- Write descriptive docstrings for each class and method implemented
- Write inline comments for each significant line of code
- Implement classes in separate files
- Always update in Trello when you are working on a task and when completed a task
- Never do a task without adding the task in Trello

Slack code of conduct:

- Always write messages in the appropriate channels

- #Assignments are either for the tasks or for the deadlines in the course opinions?
- #Random is for fun/non work related stuff
- #Meetings is for meetings. Never write something that isn't a scheduled meeting here. If you have to discuss something in this channel do it in replies. This is to improve readability and have an easy way of checking up on dates.
- #General is for everything general questions or requests for tasks, help or discussion.
- #Python is for more technical issues such as not getting Pylint to work. IE not code related
- #Git is also for more technical issues relative to git.
- Just to reiterate only #Meetings have strict restrictions on what/how to write.

User story estimations:

- The entire team should be involved in a team agreement for each user story estimation
- User story estimations should be decided using hours, a more difficult story get more hours since it takes longer to solve.

Issues between team members:

- If there is ever any issue (such as unresolvable disagreement on task work, etc) occurring between any team members then, if this is not a matter to be addressed by the University, there should be a meeting held with all team members present to best attempt to resolve the issue.