Group Contract

# General

This is a team project, not an I project. If in doubt, ask. No stupid ideas. Be honest with each other.

# Grammar

* Project spelling/grammar is to be American based.
* No use of contractions such as: isn’t, wasn’t, can’t and others.

# Conflict Handling

* Should issues arise professionally in the group, e.g. disagreements over a procedure, a majority win vote should decide it.
* Should someone break the contract, a penalty jar is set up. The penalty will be 5 kr.
* Matthew will keep the money until the group decides to use it.
* A log shall be kept over penalties, in the GitHub folder. The group will vote on whether the penalty will be applied or not, a majority vote is needed to decide.
* In case of disagreement, a die is rolled to decide what we do.

# Attendance and Meetings.

* You are expected to show up every day. If unable to attend, inform the group via the common text chat.
* If unable to attend, you are still expected to attempt to be in the daily scrum meeting via skype or another similar platform.
* We will use a Slack group and events to facilitate organization. If you cannot attend an event, please be honest about why not. Always respond to events ASAP. DO NOT RESPOND MAYBE.
* We meet at 10:00 am, Monday-Friday; 01:30 pm Sunday[[1]](#footnote-1).
* Starting with a stand-up meeting. If late to the stand-up meeting, penalty applies.

# Scrum Usage

* The scrum master will change every sprint, as with working pairs.
* At the end of each sprint we will do a retrospective to reflect on our work
* We will begin each day with a sit-down meeting
* Sprints will be 1 week in length and will begin and end on wednesday.

# Developer Tools

* Organization of time, place, and specials meetings: Facebook and Events
* Organization of project files, code, and version control: Git w/ GitHub and Google Drive.
* Organization of tasks, project backlog etc: Meister Task

# Naming Conventions

* Methods must describe what they do
* Global variables are named with an underscore at the beginning, unless they are constant.
* Constant variables are all upper case.
* Variables always start with lower case and methods always start with upper case, and then camel case is used after.
* Parameters of methods are all lower case.
* Variables should describe what they are.
* Properties named like methods.

# Architecture

* User interface should be MVVM.
* Main class library should be called Core.
* Unit and integration tests will be used, TDD.
* File and Database management will be done by Data Access Layer.
* The code should follow the SOLID principals.

# Quality Assurance

* Completing tasks in the QA section of the scrum board is the top priority
* You cannot check tasks that you created (all grammar is checked by Matthew)
* You must follow the given criteria when checking tasks
* The scrum master will assign group members to check tasks

## How you are expected to use Git / GitHub

* Only change one method per commit.
* Expected to make decent comments.
* Sync whenever you leave your computer and whenever you feel it is necessary.
* Each task should be dealt with in separate branches
* After QA, a pull request to master should be made, then the scrum master reviews, merges, and closes the branch

# Logging

* There should be an entry in the log every day that work is done.
* It is the common responsibility to ensure the log is written.
* During meetings, a secretary is assigned to write a summary of the meetings.
* The scrum master will choose the secretary

# Voting

* All votes are equal
* You cannot abstain

# Amending the contract

* To amend the contract, a 75% vote is needed to approve of the amendment.

# Team Pac-Man members:

* Hedviga Arta Geriņa; hedv0149@edu.eal.dk;
* Matthew Peterson; matt2694@edu.eal.dk;
* Roxana Ion; roxa0188@edu.eal.dk;
* Jonas Laursen; jona8690@edu.eal.dk;

1. Jonas except from meeting Sunday due to work [↑](#footnote-ref-1)