

Group Contract - GoAyo, Inc

Team members:

Gabriel Brattgård

Alex Phu

Oscar Sanner

Olof Sjögren

Yenan Wang

Contact:

- Olof: olofsjogren97@gmail.com
- Oscar: o.sanner@hotmail.com
- Alex: alex.phu@live.com
- Yenán: nanmu.w4@gmail.com
- Gabriel: gabriel.brattgard@gmail.com

Meetings:

- Meetings are planned and logged in a Google Calendar.
- Weekly mandatory meeting with TA. (Thursdays 14:15 - 15:00, we meet at 13:15)
- Must meet AT LEAST two times.
 - Once on Fridays 13:15 - 15:30.
 - Once on Tuesdays 10:00 - 11:45. (Except w38, then on Monday same time)
 - Once on Mondays, varying times.
- The upcoming meetings are determined at the end of every meeting.
 - The mandatory ones are static but other meetings will be planned when needed.

Roles:

- A secretary role will be rotated amongst the members
 - Mod rotation: Alex, Yenán, Oscar, Gabriel, Olof
- The team will be divided into two development teams and one Scrum master.
 - The development teams will be in pairs and rotated every week.
 - The Scrum Master role will also be rotated among the members.
- The Scrum masters duties:
 - Make sure everyone understands the scrum board.

- Decide which tasks are to be prioritized this week and delegate the work through code-sprints.
- The Scrum master will also be working on tasks if time permits.

Documentation:

- Meeting protocols will be uploaded to the GitHub repository.
- Things that will be documented:
 - Objectives
 - Reports
 - Discussion Items
 - Outcomes and Assignments

Communication guidelines

Fears:

- Conflicts.
 - Keep work and personal issues separated.
- Uneven workload.
- Lack of involvement.

Needs:

- Effective communication.
 - Explicit times, places, and progression.
 - Daily stand-ups as a way to solve this.
 - Communicate what we have done and what our current tasks are.
 - Inform members as soon as an issue arises.
- Clear roadmap and deadlines.
 - Keep to the meeting schedule.

Git Discipline:

- Proper commit messages.
- Use “git -author” during pair programming.
 - Take turns with the commit.
 - Write the other partner’s name in the commit message.


```

■ $ git commit -m "Refactor usability tests.
■ >
■ >
■ Co-authored-by: name <name@example.com>
■ Co-authored-by: another-name <another-name@example.com>"
          
```

Coding conventions:

- Insert a header comment in each class, see examples below.
 - State all users that created/modified the class and date of creation.
 - State purpose and functionality of the class.
 - Update the header comments after each modification with date, authors, and a short message that describes the change.

```

/* @author Simon Gislen

* @author Linda Evaldsson, Johan Swanberg (revised)

* @date 2015-04-02

*

* The map holds all information related to an instance of the game except for the
player details
*

* ---

* 08/04 Modified by Johan Swanberg. Switch to ScreenType from GameState.

* 08/04 modified by Linda Evaldsson. Made Map-class abstract. It is no longer a
singleton.

* 23/04 Modified by Simon Gislen, added Persistent Object management

* 25/04 Modified by Johan Swanberg, adds adding Stack

* 08/05 modified by Linda Evaldsson. Moved functionality to this class; rendering
of towers, updating, and stack-functionality (clear stacks)

* 10/5 modified by Johan Swanberg, fixed creepwithinrange method and added some
comments

* 10/05 modified by Hanna Romer. Added method towersWithinRange that is exactly
like creepsWithinRange.

* 17/05 modified by Hanna Romer. Added method getNew.

* 20/05 modified by Linda Evaldsson. Added ParticleManager as constructor
parameter.

* 22/05 modified by Hanna Romer. Added method isTowerOnPosition.

* */

```

Ambition:

- We strive towards the mark 4+ (5).
- To be in phase with the course curriculum.
- We are hoping that this project will be something that we can show on our resumes. Not only a school project.
- Comfort and safety within the group.

Decision making & Conflict management:

- Clear but limited discussion regarding the issue.
 - If at an impasse: vote on the matter.
 - Accept the outcome, “Kill your darlings”.
- Keep it professional.
- Don’t let problems become “an elephant”, communicate them clearly, and bring them to the surface.

Feedback:

- Don't make it personal & Don't take it personally!
- More is better as long as it's well-motivated.

Deadlines:

- We work in weekly code-sprints, Mondays to Fridays.
- Check deadlines every Monday and Friday.