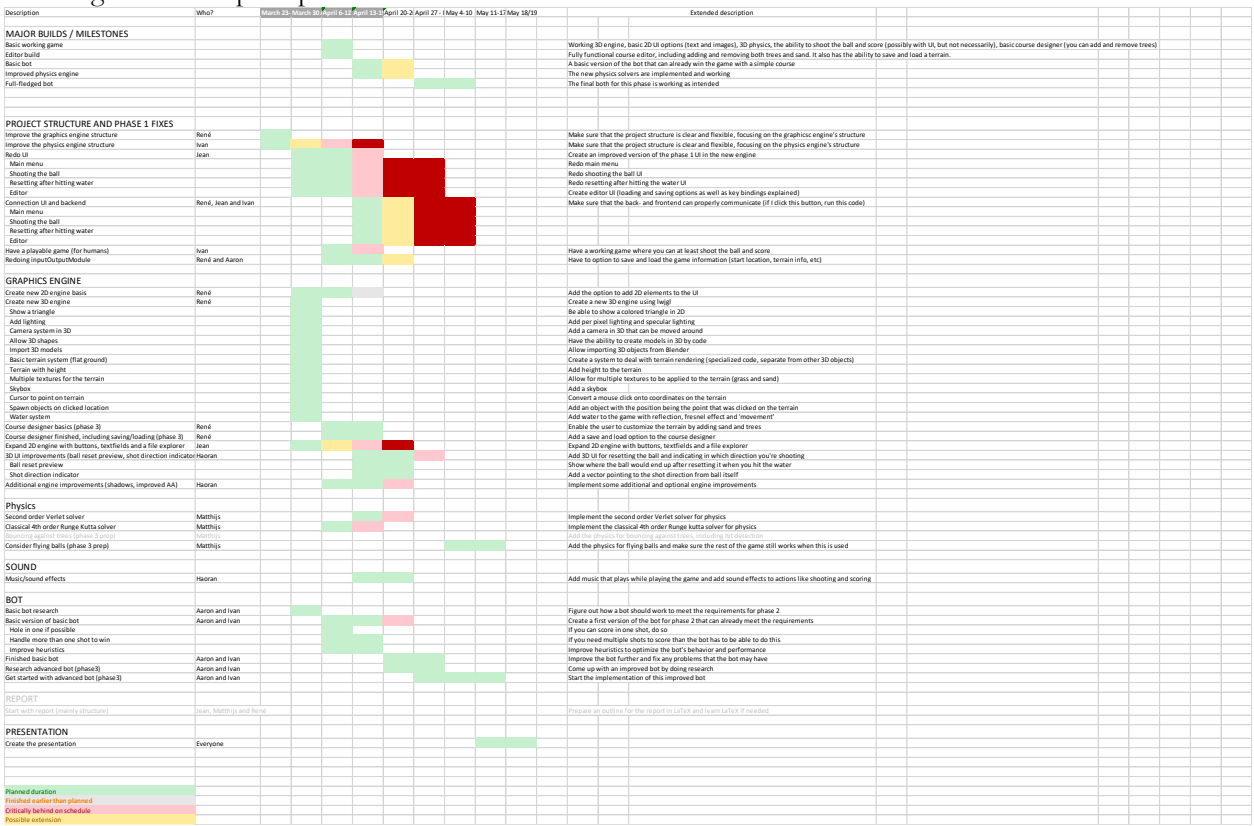
Agenda for meeting on 21-4-2020

Project: 1-2 Crazy Putting
Date: 21/4/2020
Group number: 4
Group members: René Steeman, Aaron Schapira, Ivan Poliakov, Jean Janssen, Matthijs Kusters, Haoran Luan
Chair: Jean
Secretary: Ivan

1. Opening at 12:00
2. Minutes last meeting
   a. Remarks from group: none
   b. Remarks from tutor (1 minute)

3. State of the project (5 minutes)
   a. Completed tasks/milestones: 2D UI system, Runge-Kutta solver, playable game
   b. In progress: Basic bot, improved load/save system, engine improvements, music/sounds

4. Planning (5 minutes)
   a. Things to finish before the next meeting: Improved load/save system, engine improvements, music/sound, UI rework
   b. In progress before next meeting: Basic bot, Verlet solver, advanced bot research, 3D UI
   c. Twice a week, once on Saturday at 10:00 and on Tuesday at 10:00. If needed we'll schedule extra meetings 'on-demand'.
   d. Planning of the complete phase



Description	Who?	March 23- March 30	April 6-12	April 13-19	April 20-26	April 27 - May 4	May 4-10	May 11-17	May 18/19	Extended description
MAJOR BUILDS / MILESTONES										
Basic working game										Working 3D engine, basic 2D UI options (text and images), 3D physics, the ability to shoot the ball and score (possibly with UI, but not necessarily), basic course designer (you can add and remove trees)
Editor build										Fully functional course editor, including adding and removing both trees and sand. It also has the ability to save and load a terrain.
Basic bot										A basic version of the bot that can already win the game with a simple course
Improved physics engine										The new physics solvers are implemented and working
Full-fledged bot										The final both for this phase is working as intended
PROJECT STRUCTURE AND PHASE 1 FIXES										
Improve the graphics engine structure	René									Make sure that the project structure is clear and flexible, focusing on the graphicsc engine's structure
Improve the physics engine structure	Ivan									Make sure that the project structure is clear and flexible, focusing on the physics engine's structure
Redo UI	Jean									Create an improved version of the phase 1 UI in the new engine
Main menu										Redo main menu
Shooting the ball										Redo shooting the ball UI
Resetting after hitting water										Redo resetting after hitting the water UI
Editor										Create editor UI (loading and saving options as well as key bindings explained)
Connection UI and backend	René, Jean and Ivan									Make sure that the back- and frontend can properly communicate (if I click this button, run this code)
Main menu										
Shooting the ball										
Resetting after hitting water										
Editor										
Have a playable game (for humans)	Ivan									Have a working game where you can at least shoot the ball and score
Redoing inputOutputModule	René and Aaron									Have to option to save and load the game information (start location, terrain info, etc)
GRAPHICS ENGINE										
Create new 2D engine basis	René									Add the option to add 2D elements to the UI
Create new 3D engine	René									Create a new 3D engine using lwjgl
Show a triangle										Be able to show a colored triangle in 2D
Add lighting										Add per pixel lighting and specular lighting
Camera system in 3D										Add a camera in 3D that can be moved around
Allow 3D shapes										Have the ability to create models in 3D by code
Import 3D models										Allow importing 3D objects from Blender
Basic terrain system (flat ground)										Create a system to deal with terrain rendering (specialized code, separate from other 3D objects)
Terrain with height										Add height to the terrain
Multiple textures for the terrain										Allow for multiple textures to be applied to the terrain (grass and sand)
Skybox										Add a skybox
Cursor to point on terrain										Convert a mouse click onto coordinates on the terrain
Spawn objects on clicked location										Add an object with the position being the point that was clicked on the terrain
Water system										Add water to the game with reflection, fresnel effect and 'movement'
Course designer basics (phase 3)	René									Enable the user to customize the terrain by adding sand and trees
Course designer finished, including saving/loading (phase 3)	René									Add a save and load option to the course designer
Expand 2D engine with buttons, textfields and a file explorer	Jean									Expand 2D engine with buttons, textfields and a file explorer
3D UI improvements (ball reset preview, shot direction indicator	Haoran									Add 3D UI for resetting the ball and indicating in which direction you're shooting
Ball reset preview										Show where the ball would end up after resetting it when you hit the water
Shot direction indicator										Add a vector pointing to the shot direction from ball itself
Additional engine improvements (shadows, improved AA)	Haoran									Implement some additional and optional engine improvements
Physics										
Second order Verlet solver	Matthijs									Implement the second order Verlet solver for physics
Classical 4th order Runge Kutta solver	Matthijs									Implement the classical 4th order Runge kutta solver for physics
Research advanced bot (phase 3 prep)	Matthijs									Add the physics for hovering-against-bees, resulting in detection
Consider flying balls (phase 3 prep)	Matthijs									Add the physics for flying balls and make sure the rest of the game still works when this is used
SOUND										
Music/sound effects	Haoran									Add music that plays while playing the game and add sound effects to actions like shooting and scoring
BOT										
Basic bot research	Aaron and Ivan									Figure out how a bot should work to meet the requirements for phase 2
Basic version of basic bot	Aaron and Ivan									Create a first version of the bot for phase 2 that can already meet the requirements
Hole in one if possible										If you can score in one shot, do so
Handle more than one shot to win										If you need multiple shots to score than the bot has to be able to do this
Improve heuristics										Improve heuristics to optimize the bot's behavior and performance
Finished basic bot	Aaron and Ivan									Improve the bot further and fix any problems that the bot may have
Research advanced bot (phase3)	Aaron and Ivan									Come up with an improved bot by doing research
Get started with advanced bot (phase3)	Aaron and Ivan									Start the implementation of this improved bot
REPORT										
Start with report (mainly structure)	Jean, Matthijs and René									Prepare an outline for the report in LaTeX and learn LaTeX if needed
PRESENTATION										
Create the presentation	Everyone									

Planned duration
Finished earlier than planned
Critically behind on schedule
Possible extension

5. Any other business (2 minutes)
   a. From group: When do we need to start using Junit for testing, is it phase 3 only?

    b. From tutor:
6. Chair/Secretary for the next meeting. Chair: Matthijs, secretary: Ivan
7. Closing at 12:15