Table 1: Revision History

Date	**Developer(s)**	**Change**
Jan 26, 2020	Alexander Samaha	Set-up document, initial changes

SE 3XA3: Development Plan
Title of Project

Team 203, Abstract Connoiseurs
Daniel Noorduyn, noorduyd
David Jandric, jandricd
Alexander Samaha, samahaa

Put your introductory blurb here.

1 Team Meeting Plan

Need to include : When? Where? Frequency? Roles? Rules for agendas? Every Tuesday and Wednesday in ITB 236 from 2:30PM to 4:30PM. Informal meeting every Thursday at 6:00PM, if required.

David and Alex will log, and Dan will be chair.

Rules for agendas:

1.

Rubric: 5 marks.

2 Team Communication Plan

How we communicate as a team. Git issues (through merge requests and wiki posts?), email, group chats, phones. What contact information is exchanged and what is used for what information (issues through git, updates on groupchats, phone for meetings?)

Rubric: 3 marks.

3 Team Member Roles

Name a team leader, scribe. Then we identify experts based on git, LATEX, technology, perhaps domain. Roles are fluid and you should know which hat you wear.

Rubric: 2 marks

4 Git Workflow Plan

Pretty simple, how are our branches organized, is our repository centralized? How do we label everything, how will milestones be used?

Rubric: 3 marks

5 Proof of Concept Demonstration Plan

Validate technical feasibility. https://sensinum.com/proof-of-concept-in-software-development/

Identify risks, is testing going to be difficult? what are difficult implementation aspects. Are there required libraries that have some concerns (pygame and MacOS for example). Will portability be a concern (yes for new Macs I guess, out of our control but lets talk about it).

What will we demonstrate to overcome the risks. (talk about what we might show in terms of testing, what our product will be? demos of levels?).

Rubric: 5 marks. Gonna be a biggie. Maybe we have a table here? some figures.

6 Technology

Talk about the Programming languages, are we using an IDE? Testing framework, how are we testing (maybe something to do with pytest). Document generation, how are we making documentation in Python?

Rubric: 4 marks, we need to write a bit to really bring this one out.

7 Coding Style

We should follow a coding style, this can involve but it's important that the code is consistent. Someone needs to research some python coding styles. Then we talk about it and how it can be helpful!

Rubric: 2 marks. Should be enough to just introduce the style and what advantages it brings.

8 Project Schedule

Provide a pointer to your Gantt Chart.

Rubric: 5 marks. Needs to be detailed and accurate (realistic).

9 Project Review

Looks like this is necessary for our revisions.

Rubric: 1 mark