

Table 1: Revision History

Date	Developer(s)	Change
10/18/2016	Daniel Agostinho	Initial Draft
10/18/2016	Anthony Chang	Initial Draft
10/18/2016	Divya Sridhar	Initial Draft
12/4/2016	Daniel Agostinho	Revision 1

SE 3XA3: Development Plan

PROJECT TETRIS

Team #11, Team Tetra
Daniel Agostinho agostd
Anthony Chang changa7
Divya Sridhar sridhad

PROJECT TETRIS is a game developed by Team Tetra. The members involved in this project are Daniel Agostinho, Anthony Chang, and Divya Sridhar. This project is a reimplementation of the classic puzzle solving game Tetris designed by Alexey Pajitnov. The existing Java implementation that the project was based off of exists at ([link to github.](#)) This development plan will serve as a description of all the actions to be taken to develop this software.

1 Team Meeting Plan

Team Tetra will meet once a week to work on the project and delegate specific tasks to each member. The location of these meetings will be at H.G. Thode library at the McMaster campus. The meetings will take place on Mondays from 5:30 pm to 7:00 pm. Each meeting will have an agenda decided upon by all members of the group throughout the group. Should a member come up with an issue, it will be discussed with the other group members about adding this problem to the following week's agenda. The chair of the meetings will be the team leader Daniel Agostinho. Meeting minutes will be taken down by Anthony Chang and Divya Sridhar. At the end of the meeting, each member will be assigned work with an appropriate deadline.

2 Team Communication Plan

The team members will communicate through various means of communication including SMS/phone, Facebook, and Git Issues. Facebook will be the primary source of collaboration and communication regarding the project deliverables. Git Issues is also used for specific communication regarding certain parts or sections of the project.

3 Team Member Roles

All code and documentation will be divided evenly amongst all group members. Project Tetris will follow an MVC architecture, and each component (i.e. model, view, controller) will be distributed amongst the group. Daniel is the project lead, therefore he will work on developing the code and improving the existing open-source project that is being used as reference. Divya will be responsible for the vigorous testing methods through which the software will be validated. Anthony will be responsible for the modularization of code and the MIS and MG. The software requirements specification and design document will be the responsibility of all group members.

4 Git Workflow Plan

The Git workflow plan that will be used is the Feature-branch workflow. Because most of the work done by the members will be done independently throughout the week, the feature-branch workflow is optimal. Everyone can work on their own branch without causing large merge errors. Also before merging a pull request must be made which allows members to revise the commits. Daniel will be in charge of managing the pull requests. These will be done as often as needed. As the project progresses whenever a group member has finished some code or some part of a documentation, they will submit a request to merge their changes to a develop branch. Once a milestone has been reached and the overall product is ready for a release the develop branch will be merged to master. Any any given point there should be no broken code or incomplete features on the master branch. Labels will be used to identify different types of commits (ie. documentation, code, etc.) Milestones will be used to identify different tasks on Git. For example there will be different milestones for the Requirements Document, etc.

5 Proof of Concept Demonstration Plan

5.1 Will a part of the implementation be difficult?

The most difficult part of the implementation will be the game logic. Getting the Tetris blocks to interact correctly and getting the blocks to be removed from the game once a user clears a row will be the most challenging part.

5.2 Will testing be difficult?

As this software will be a game, testing can be fairly difficult. With regards to unit testing, the software will be validated with JUnit. Aside from that the software will be validated with rigorous manual testing. A group of users will be used to test the software for its functional and nonfunctional requirements. Their feedback will be recorded and used to improve the quality of the software.

5.3 Demonstration

Team Tetra will demonstrate a working prototype of the software including a fully functional game logic. This will correspond to the Controller component in the MVC architecture. The Model and View components will be implemented later as they are not directly related to the risks. The proof of concept will fulfill all the functional requirements of the system including block translation, block collision and clearing the rows. This will be difficult to validate but the Eclipse console will be used to update the user on the state of the model.

6 Technology

The programming language that will be used in this project is Java. Java was the programming language of choice because this is the language that all members were most comfortable with. Java fits well with the object-oriented nature of the game. The IDE to be used is Eclipse. Eclipse is an IDE that all members have been using for years and have the most experience with. The testing framework to be used is JUnit. JUnit was a framework used in previous years and is again a comfortable option for team members to use. For document generation, JavaDoc will be used. The libraries to be used in the software are the Java Swing and Java AWT libraries for the GUI as well as the Java Util library for ArrayLists and Random.

7 Coding Style

The coding style that Team 11 has decided to adopt is the Java style from Google. Link: <https://google.github.io/styleguide/javaguide.html>. The reason for using this standard of coding is that it is very similar to the way that the team members have been coding for years. Using a coding standard is imperative because it allows all of the members of the team to communicate their code effectively and remove the ambiguity that comes with having different members working on the same code.

8 Project Schedule

Provide a pointer to your Gantt Chart.

9 Project Review

Overall the project was a success. We met all of the goals that we set out to achieve. The main goal for this project was to make a fun game that was accessible on all platforms. The game does run on Java, OS X and Linux thus proving its portability. The game is fully functional with almost no bugs. The

overall development of the software went smoothly as well. The decomposition of responsibility worked well as the roles assigned to each member was the role that they were most adept in. Thus the quality of the work completed was optimal. One thing that could have been done differently is the meetings. The workload of the semester proved to be too heavy to have weekly meetings. Although the members made up for it by communicating regularly through SMS/Facebook. An improvement to this would be to have meetings less frequently but for longer duration and to adjust the project schedule accordingly to reflect these meetings.