**2048 Project Reflection**

Date: May 30, 2014

To: Sandeep Raghunandhan {Project Manager}

From: Sandeep Raghunandhan

Subject: “2048 (3D and 2D)” Project Reflection

**Accomplishments.** I handled most of the coding for the 3-D game with respect to the graphics which utilized the Java3D library. I struggled to get what was internally present in the 3-D array of tiles to be consistent with the tiles displayed on the Java3D canvas.

**Learning Experience**: Understanding how GitHub and revision control works. Learning the Java3D library all on our own. Time management was an important skill to have while working on this project.

**Objectives**.

· Challenge I would say too much. In the beginning, we were too ambitious about our abilities because we were hoping for multiple dimension game play and networking. In the end, all we had was a 3-D version

· Effort Considerable bit, but too much work at the last minute. We spent too much time planning and less time implementing. Learning the Java3D api took too long. We should have simultaneously learned the api and coded.

· Quality Decently well. The animation is not obvious because the tiles move too quickly for the human eye to perceive, but you can tell it happened. In terms of user friendliness, the game is quite simple to learn but next to impossible to master.

· Problem Solving Knew a lot about Java3d by the end. Haven’t quite mastered it but got a good hang of how it works. In terms of debugging, I was glad every time I saw a NullPointerException, because these we the easiest exceptions to fix. When a Java3D class, would throw something obscure like “Cannot add BranchGroup to a live/complied scene,” it was much harder to trace the source of the problem

· Results We did not get as far as we hoped, but we are proud to say that we have a functional 3-D 2048 game. The algorithm for the code is admittedly not the best one possible, but it was one that I had thought up on my own.

· Teamwork Contributed significantly to the graphics. Wrote out all the collisions which took about three days of continuous work. Ended up writing some 3000 some odd lines of code. Most of the code involved copy and pasting within the class and making minor changes. The problem with this is that there may be some issues still lingering which we have not accounted for. Our testing went only so far as to test the fundamental functionalities.

**Overall Assessment** 82