**STATUS REPORT #2**

**Scott & Nicole**

List of scratches

Animation (Mar. 2, 6): Removed all boolean arrays used for animation from render, which had been making the code very messy. Found minimal change between Boolean and Direction change but direction change made it easier to mess with other variables.

Food (Mar. 7): Had to modify the code to make the hamster speed up after eating the pellet, because of the changes in the animation screen. As a result of playing with this code we are considering what we want our game to be.

Tail (Mar. 6-9): Cleaned up a lot of code here, have a head and tail moving around. The movement is still in need of a lot of work as the directional system is fine but the way the head and tail move has yet to be fully thrashed out.

Game (Mar. 1-2, 7-19): Mostly integrated the code we worked on in the other screens; moved the walls to be higher up and invisible. The pellet is there now as well but the sign stuff is still around in case we want to do something with it.

General changes to the overall work (Mar. 7): General cleanup of variables and code in render in all screens; set up the new architecture in GitHub to avoid problems when pushing and pulling binary files (we now have several folders for releases and scratches, so that we never work on the same file at the same time)

We also have scratches for AniHit and Menu, but not too much was done on these screens, except for cleanup and integration perhaps. We ended up putting the scratches into Version 2.0 as this is where we want to go with it next.

Major challenges/setbacks

Github and the school have a conspiracy against us at forest heights, to make sure that we never truly learn how to code, and as a result we are still suffering from the trouble of dealing with that. First, we tried to make a new repository. Clearly, that did not help. We are now at Atomic Hamsters #5 (we skipped 4). Next, we tried downloading the zip. This sort of worked, but is very hard and awkward to work with. Several times, we actually just put our updated work into the “GettoHub” in the classroom folder. This is also very, very awkward to work with. Now, we are pushing and pulling through Git GUI. It is somewhat more work than how we were able to push and pull in the beginning, but it is at least functional. We also set up a new architecture with different folders with Abdullah’s help, so that we never have to work on the same folder at the same time. Again, it is not as seamless and easy to work with as before, but we were able to get at least some code done for this status report, even if it is just a little, as well as deal with all these GitHub issues.

Another challenge we had was in ScrTail. Scott got it working that two separate images for a head and tail move around together. We were now going to add another image for the middle, which would grow as the hamster eats a pellet. However, as we were working with it, we realized that we just did not really like the feel of it and in what direction the game was taking us with this feature. We are now very sure that we would like to deviate from our original plan, and want to keep the animated mouse we still have in ScrAnimation and ScrGame. We really liked the feel of that and really enjoyed playing around with speed. However, we also like the idea of something changing when he eats the pellet, so we decided we would like to just make the whole hamster start growing. We want to add another mouse as well, with separate controls and movement, so that it becomes a two-player game. They will both want to collect the pellets, and both will speed up, collect points, and grow when they eat a pellet. However, if they collide, they blow up. We still also want to add more cool levels, giving it a maze like feel, just to keep them interesting.

The last major challenge we experienced was with Gradle. Gradle didn’t like being in the H drive. As a result it revolted and didn’t allow programs to be run. As soon as gradle got reinstalled in the D drive everything was fine. It sounds like it was a simple fix, but actually cost us several periods.

Source any web site/book that helped you with that concept

We mostly just referenced the ICS3UI website and lots of experimentation. The work done during these weeks was mostly dealing with GitHub, less code.

Our Asana to show everything: https://app.asana.com/0/574180427837604/board

Our GitHub for the actual code: https://github.com/parks2214/Atomic-Hamsters-5

Lessons learned from the last two weeks:

1. You can never run out of redo’s and amendments when changing code or fixing / changing the idea behind what you want to do
2. Asana also works well, even if Freedcamp felt easier at first.
3. GitHub never fails to crash in new ways we have never experienced.
4. Sarcasm is great as seen in #3
5. The problems we experience are new for almost everyone, so fixing stuff takes time. Learning to be patient and understand why things are breaking the way they are has come in handy as a result.
6. Continuing to clean up code makes it easier and clearer to work with.