CMPT 370 Group meeting – Nov 26, 2016

S372

Group Meeting #21

Notes taken by Janelle

Present: Yige, Tushita, Nick, Janelle, Scott

**Highlights and Action Items:**

**NEXT GROUP MEETING TIME: Tuesda, November 29 @ 4:00pm**

**HOMEWORK:**

Scott: Finish construction experience document draft

Everyone: Proofread construction experience document draft

Tushita: Implement changes to the board class from code review

Nick: Continue work on the display

Yige: Flow/AspectJ

Janelle: Review code/Interpreter

**Meeting Minutes**

**1:00pm** – Code review debrief

-taught us a lot about the Board, more all on the same page (shared understanding)

-Found a lot of stuff we might not have found individually

-Length of meeting: Tushita: less exhausting than pair programming

-700~ lines in 2.5 hours Yige: it was worth the time we spent on it

-Good, lively debate atmosphere – positive group dynamics were shown!

-egos were left at the door—everyone’s code, not any single person’s code

-Important style, syntax discussions had: Throwing errors: dialogs vs exceptions, mutation bugs, magic numbers/modularity, where and when things should be validated

What could we have done differently? Actually printed out the right stuff!

Team members could have prepared a bit better – come with more questions, comments, read ahead and mulled things over

Made us consider: we should have done pair design activities, not just pair code construction.

Highlight the importance of documentation: often asked “I wasn’t sure what this does, comments were not present/incomplete”

**1:17pm** – Changes to the Board

-who will implement? Tushita will. Nick will put his notes in the repository

**1:20pm** – Other document components:

-Scott will write the other components, finish the draft tonight

-We will proofread and edit over the course of Sunday on slack/in the lab

Reflection on Code Construction:

-Taking more time than expected

-Some things (like Screens) were more complicated to construct than we originally expected

-Design document made this possible

-Testing too! Interpreter tests were almost copy-paste. Really easy

-We were able to divide up tasks right away and get unit tests passing and individual components created very quickly after the testing document was complete.

-Integration is a huge deal and taking a lot of time. Where we are discovering a lot of small problems and making small changes to make the whole come together.

-Group dynamics are assisting in the code construction

-This could have been even more time-consuming and difficult if we weren’t all communicating well and trying to stay on the same page

-Changes to our process:

-Good architecture but would have preferred an even more detailed design

-A bit backwards, but we need to think about the code construction phase as we design—we were discouraged from doing so but it would have changed things

-Requirements—we should have better estimated how much time each thing might take

Delta from design:

-We needed more fields to keep track of states than we anticipated

-Display change: each screen became its own class

-One board panel and drew the board instead of individual hexagons  
 -Interpreter: timer added as suggested in Testing document

-MVC? We get the appeal—issues with “what knows what” are cropping up. Still stick by our design but we’re learning a lot.

-Added methods to GameScreen

**1:44pm** – Construction issues

-How are we doing the Robot Librarian?

-We are OK to postpone this slightly until after the Sunday milestone

-How does the game screen know how to switch players?

-Do we need a startTurn() function?

-The confirm button in the turn transition screen will be key