**CMPT 371 – Meeting Notes**

**January 10th /17 🡪 4:00 – 5:00 (1 hour)**

What we, as individuals, can offer to the team:

* Kristof 🡪 Dev Lead 🡪 Java, Python, a bit of web dev, good problem solver
* Jeremy 🡪 Test Lead 🡪 wants to be on the test team, is organized
* Melody 🡪 Dev Team 🡪 Is good with version control, has done significant development, wants to be on the dev team, is not a native English speaker so can’t write documents
* Chris J 🡪 Test Team 🡪 doesn’t want to be a team lead, has some dev experience
* Tushita 🡪 Manager 🡪 wants to be team leader but would also like to code, so isn’t sure at the moment
* Dylan 🡪 Dev Team 🡪 Back end web development, Java, wants to be a developer, also interested in testing because he wants experience, would like to switch between the two
* Chris M 🡪 Dev Team 🡪 doesn’t want to do testing, likes to be active in multiple areas
* Ryan 🡪 Test Team 🡪 good at writing, not the most organized, wants to do development
* Justin 🡪 Dev Team 🡪 not a leader, not a native English speaker
* Arianne 🡪 Dev Team 🡪 UI, Java, C, writing docs, organization

Role Assignment:

* Project Manager: Tushita
* Dev Lead: Kristof
* Dev Team: Melody, Dylan, Justin, Arianne
* Test Lead: Jeremy
* Test Team: Chris J, Ryan
* Build Manager: Chris M

Project Selection:

* **Vendasta**
  + **Email Proxy System**
  + Hesitant because of the server aspect
* **Aura Technologies**
  + **Content Acquisition for Augmented Reality**
  + Questions:
    - Are the models being given to us or do we need to generate them?
    - Will you provide VR headsets or other relevant equipment?
    - Do you want an iOS app or website or both?
    - If website, does it need to be backwards compatible?
    - We are adding onto an already existing app?
    - What are we working with to start off?
* **Kasper**
  + **Canada’s Mortgage App**
  + Looks doable
* **Department of Anesthesia** 
  + Form a database and store information on problems with anesthesia
  + Questions:
    - Is there already a server in place for us to use?
    - What specifically are we storing?
    - Is there already a system in place where everyone already has an account?
* **Promoting Safety in Long-Term Care**
  + Looks like one of the easier projects but the group in general doesn’t seem interested
* **Tool for 3D Visualization of “Embedded” Time Series**
  + Requirements seem relatively finished, so there’s a good sense of direction
  + If we are all committed to learning as much as possible, this would be a top choice
  + Questions:
    - What is the overall presentation (what should it look like)?
    - Where is the data coming from and how can we obtain it?

General Questions:

* Are there more projects coming?

Next Meeting 🡪 Friday January 13th @ 3:30pm

**January 13th /17 🡪 3:30 – 4:30 (1 hour)**

Project Discussion:

* Today we will discuss projects in detail
* Tushita will set up voting for team members to choose their favourite project – all members should vote by tomorrow (Saturday) night
* Project’s Pros and Cons:
  + Kasper Tinder Property Viewing:
    - + easy breakdown of modules
    - + fun to work on
    - + android and iOS experience
    - + flexible
    - Questions:
      * Could we implement either iOS or Android, instead of doing both?
  + Vendasta Proxy Email:
    - + potential job opportunities
    - + professional environment
    - - SMTP proxy server knowledge
    - +/- no UI’s
  + Anesthesia:
    - + non-profit
    - - server requirements
    - - security
    - +/- iOS and Android versions
    - +/- HIPPA Guidelines
    - Questions:
      * Is security something we need to plan for?
  + ~~Osgood’s Visualization:~~
    - ~~+ learn Scala~~
    - ~~+ good breakdown of modules~~
    - ~~+ valuable and practical feedback~~
    - ~~- memory management~~
  + ~~Aura Augmented Reality:~~
    - ~~+ work with Kryztof Voss~~
    - ~~- complicated~~
  + ~~Polytenna:~~
    - ~~+ relatively easy~~
    - ~~+/- lots of UI development~~

Other Topics:

* Jeremy should look into testing tools presented in class
* After project is selected, all group members should start thinking about requirements for phase 1 and bring these ideas to Monday’s meeting
* Inspection will occur on rough draft of requirements
* Dev team and test team start thinking about what the project implications are
* Start thinking about requirements and their implications on Design soon
* Source tree

Next meeting 🡪 Monday January 16th /17 @ 5:00

**January 10th /17 🡪 5:00 – 6:30 (1.5 hours)**

Discussion about Kasper:

* We can do a base outline of the swiping mechanism, buttons, other functionality
* All storage is done in the data base which we don’t have access to yet
* They already have an app which we can view in order to use their coding standard and so that our app works with their system
* Before swiping, there are filter checks (i.e. bedrooms, bathrooms, etc.)
* We can be creative and make it interesting:
  + Buttons vs. swiping if that looks cooler
* Existing apps to look at:
  + Zillow (good example)
  + Realtor.ca (not a good example)
* Back end uses Ruby
* Their code is on github
* Questions:
  + Is there something we can use that converts a code base into both an android and iOS compatible product?
  + We shouldn’t be working with existing data, but we’re not sure.
* Basic needs:
  + Basic set up
  + Filter
  + Swipe feature
  + Database

REQUIREMENTS:

* Requirements for Phase 1:
  + Basic set-up
  + Swipe feature
  + Buttons
  + Fake data
    - Location
    - Price
    - Bedrooms
    - Bathrooms
    - Square feet
* Future Requirements:
  + How can we integrate with google maps?
  + Schedule a viewing feature
  + If a user has liked a house and something about the house changes (change will occur in the database) user needs to get a notification about the change

Next meeting 🡪 Tuesday January 17th /17 @ 4:00

**January 17th /17 @ 4:00 – 5:00 (1 hour)**

Ryan has volunteered to be the Risk Assessment Manager

Risk Assessment Inspection:

* Two aspects to risks:
  + 1) How severe
  + 2) How probable
* Two methods of handling risks:
  + 1) Mitigation
  + 2) Contingency Plan
* 10 risks assessed per phase (minimum)
* Tushita will create a risk channel on slack to jot down risk ideas

**January 20th /17 @ 3:30 – 5:30 (2 hours)**

* 1 code review is mandatory for everyone
  + Everyone needs to sign up on the spreadsheet
* 1 – 2 risk scans for every ID
* Deadline for development needs to be set
  + ID 2 due on Feb 5th /17
* Code due for ID one – Thursday Feb 2nd (day -5)
* Triage meetings – Saturday afternoons (day -2)
* Pair programming sessions:
  + 1 – 2 per person per ID
  + See text file in git repo (called Pair Programming sessions)
  + Good idea to have a dev/test combo
* Project Documentation:
  + On git
* Checklist folder:
  + Check back every couple of days and make sure you’re on top of everything
* Tuesday presentations:
  + Everyone needs to present at least one
  + Sign up in the sign up sheet (371 Kasper)
* Everyone needs to document the work they’ve done
  + Activity log folder on git
  + Everyone gets their own text file
  + Tushita will post an example format file
  + Independent meetings added
* Development and Test Pairs:
  + Assigned on git
* Defect Database
  + Where we keep track of bugs and issues
  + Tracked in Trello

Discussed Infrastructure:

* Ionic2 🡪 framework
* Node JS 🡪 JavaScript framework

Documents:

* Doc team Arianne, Tian, Ryan, and Gaurav
* Requirements Doc (deadline Sunday)
  + Acquire a list of requirements
    - Conrad’s email
  + Finish UI mock up (for people posting)
    - Requirements
  + Use cases (on Git under mini milestones)
  + Platforms iOS, Android, Browser
  + Project documentation 🡪 see syllabus
  + Must haves, should haves, nice to haves
* Design Doc
  + More info to come…

**Dev Team meeting on Sunday @ 2:00pm 🡪 set up dev environments**

Things to check regularly:

* Git
* Google Docs
* Trello
* Slack (for links)

**January 24th /17 - Infrastructure Presentation**

Osgood’s Questions/Advice:

* Cross platform issues
* To what degree is it required to operate offline?
  + Robust in handling disconnection from network
  + Buffer of Listings loaded
* End to end testing
  + Protractor 🡪 more end to end
    - Not fully end to end
  + Truly end to end would be server to app, covering all interfaces
  + Consider bandwidth issues and testing challenges
  + Simulated test requests that would come through a restful interface/webserver, without the need to be running a full client emulator
  + Make sure some end to end testing is supported – will require some coordination
    - Test frameworks may be different on both sides
    - Logging info needs to be recorded
    - Joint tests and separated tests
* What coverage of these do people in the team already have?
  + Angular, node, jengo, ionic, jasmine, karma, trello, etc.
  + Osgood’s preference 🡪 use pre-existing accounts where possible

**January 27th /17 @ 5:30 – 6:00 (0.5 h)**

**Discussed**:

* **Timeline**:
  + Signup for code review, presentations, triage meeting
* **Risk Document:**
  + Ryan is 8/10 done risk document look at risk document on Git
  + Arianne will start editing
* **Upcoming Deadlines** (see Signup Sheet for all deadlines):
  + ID1 due on Sunday at 11:55pm
  + Starting Feb 6th team will start **meeting with Osgood**
    - Dev lead and test lead should be there
  + Feb 7th **ID1 Presentation**
  + Feb 8th **Risk Scan**
    - Put risks on slack if you think of them
  + Feb 10th **Code Review**
    - Dylan with architecture design
  + Feb 17th **ID2 Due**
  + Feb 18th **Bug Party**
* **Other**:
  + ID4 🡪 Everything should be finished by ID4
  + ID5 🡪 Last minute fixes and optimizations
    - Defects Identification should be modified (because things will be fixed)
  + Code review sign offs:
    - Mutual agreement of people at the code review

**January 31st /17 @ 4:00 – 4:50 (50 minutes)**

Dylan and Dev Stuff:

* For implementation, anything the user can input into the app needs to be linked into the code
  + Input must be modelled to an instance variable of that page
  + Example and commenting on Sign-In Screen
* Everyone needs to put loggers in their pages
  + Copy paste the private logger: Logger line from sign up, inject them into the constructor
  + See Sign-In Screen
  + There are different levels of logging
    - Log, debug, warn, error, and info
* Variable names:
  + All camel case (except logger since its global)
* Any time you want to move back a screen, don’t push it because that will create a new instance of it and put it on top, instead pop from the stack

Arianne Documentation Stuff:

* All tasks have been assigned and are due by Friday night for Editing
* Test team stuff due Sunday afternoon