Client Meeting Notes For Requirements

Team Polaris

Existing projects:

* Start to analyze code and documentation before meeting with client
* Make a plan of how you will proceed with this semester, and turn it into a list of requirements

Requirement list should include:

* Priority
  + Prioritize requirements with a rank
* Justification
  + Why is each requirement needed
* Separation of functional, non-functional, and domain requirements
  + Requirements
    - Functional
    - Non-Functional
    - Domain

Client Questions

Nancy’s Notes

1. Plan for semester + List of Requirements: “What requirements are you looking for this semester?”
   1. Documentation wasn’t good. But replace couch and pouch with newer ideas/concept
   2. Want coherence in system so multiple professors can upload ratings together
   3. Can go to the previous version of databases already because they exist simultaneously → concurrency
   4. Improve the 5 second wait time where it goes online and offline. Bugs.
   5. Look up deeper with NoSQ databases (Not higher on the priority list)
   6. Angular 1.5 → Needs to be updated.
      1. Scrapped Angular 1
      2. Angular 2 and 6\*\* not that much different
      3. Refactor to updated version if same functionality is possessed
         1. If refactored, maybe firebase??
         2. Gunay doesn’t have confidence in a refactor that has same functionality → says that couch and pouch are up-to-date
   7. Ionic → the mobile app
      1. For mobile feature but mobile isn’t working
      2. It’s a browser application. Not really mobile
      3. Couch works with this
      4. Bug: Node is a problem. Computers are a problem. Mobile apps are a problem.
         1. Everything becomes obsolete very fast with Android
         2. “MAD” grows month to month that changes a lot
      5. One person has to figure it out and update
      6. But stuck in Android 1
      7. Ionic is unable to produce an output. It’s stuck. Not working on IOS but might work on Android
      8. Trying to upgrade ionic → ios gives issues
   8. Judge didn’t want to turn on their phone when doing this
      1. Were given tablets.
      2. Tablet sync doesn’t sync at the end → gets stuck and you have to log out and go to the home screen so last stuff not entered
      3. Pouch isn’t syncing with couch
   9. Security issue
      1. Passwords in a public repo
      2. Open public resource to passwords → easy access to database
      3. App token is stored inside of the code
         1. Needs to be put into a JSON file outside that’s not posted onto GIT
      4. Generate a variable not a password
      5. Mobile doesn’t have email capability
      6. Create a file and move stuff into the file. File should not be pushed to GIT and change all those passwords
   10. More admin capability in the Web
2. Are there items or priorities that you want us to focus on over others?
   1. Fix Tablet bug/coding bug \*\*\*\*
   2. Android/Mobile
      1. Requires closing and opening
   3. Couch/Pouch
   4. Ionic
3. Is there a specific platform you would like us to focus more on?
   1. New Angular
4. Do you want views for both students, teachers, and admins
   1. Students don’t have accounts
   2. This is judging only for professors/judgers
   3. Judges are given single password to everyone and GGC credentials is sent
5. Should old data be accessed on the web
   1. Old semesters
   2. Can’t delete the tables
   3. Eventually will dump the older versions.

Nancy’s Extra Notes

* Log-in page is decoration only
  + “Viewer” started but never became anything
* Security
  + Forever link invitation
  + GGC account verification process set up on work phone for Profs
* Students can be sent links as well to see their feedback
  + Lots of manual work for it
* Maybe e-mail functionality
  + No php → node, Angular, react, view, JQuery
* Do you want to update or do you want to split the tasks with the other team
* Look up different technologies
* -App started because a professor saw that google forms was an issue because of GGC wifi
* -This APP should withstand being able to be offline. Some kind of caching
* -”CouchDB” → NoSQL
* -2 Databases
* -”Pouch” → Caching on the client side on javascript. Creates a huge mess. Works when works.
* -”Couch” → Syncs with the pouch. But if you delete a table on the database, it appears on the other side.
* -Why
* -Pouch + Couch was most popular and is in the language “ “ for parallel and co-current systems. Parallel use
* -Want coherence in the system so that multiple professors are able to upload together. When it goes offline, everything is synced, and when it goes online then everyone can submit again at the same time.
* -Gunay will give us 2 documents to fix. But don’t change the passwords

Andy’s Notes

* Caching offline application
* CouchDB - noSQL
* Pouch Client side js
* Synchronization
* Distributed systems databases
* Refactor
* Keep couch and pouch supposedly
* Invitation
* Results from Judging
* Email functionality no php

Juan’s Notes

* The app main created to be able to use offline.
* The database is NoSQL
* Pouch runs on the client side
* Both Database are made to run as one.
* Why
  + couchDB - made for asynchrony usage - multiple entries
* New technology - replacing couch and pouch
* Google noSQL that is better than couch and pouch
* Heavier focus on the database
* Major: Created with angular 1.5 but google restructured in 2
* REFRACTURE
  + Check new database if we refracture
* Coding Bug: The biggest bug is sending information from a mobile to one database then getting it to the other database
* DON’T CHANGE PASSWORD!!!
* **Super important:** Security issues: passwords are on a public repo
* Mobile needs security file created and it is not allowed to be pushed to git

Jacob’s Notes

* App that can run offline (Couch and Pouch DB)
* Pouch runs on client and caches everything
* Problem with Couch and Pouch is that when you delete a table it keeps reappearing
* Synchronized, made for parallel use
* All states of the DB exist simultaneously
* NoSQL DB
* Angular 1.5 outdated need to go to update to a newer version of Angular (6 or 7)
* The mobile app has trouble syncing reports at the end
* Android app must kill/restart to sync changes
* Need for better security
  + Email link that emails faculty passwords to access voting
* Passwords are in the public repo, need to use external JSON file to keep the passwords safe
* Sendgrid system used to send emails
* On a large scale, want this system to be used between other colleges