**Old implementation plan from last week:**

|  |  |
| --- | --- |
| **W6** | * Working with API and understanding its features * Begin making users and creating a task list   + Have simple design of users and viewable task list finished |
| **W7** | * Add features onto task list   + Solidify Group/User relationship   + User ability to add/remove tasks from calendar     - Add priority flags     - Assign a deadline     - Assign to specific user   + Develop UI * Develop Server architecture, bring it closer to the app |
| **W8** | * Finish up, verify, and clean up whatever was not accomplished in the previous weeks * Work establish server interactions   + Joining and creating new groups   + Inviting new users * Look into adding notifications |

**Motivation for implementation plan changes:**

As we learn and develop our app, we found it would be best to try and stay away from the more general implementation goals as they don’t really provide the developers (us) with enough guidance. As such we have tried to make the implementation plan more concrete and broke our more immediate goals down into more understandable pieces. We also needed to push a couple things back, as we had done before, things seem to kind of ripple through and delays one week propagate into the next.

**New implementation plan updated 11/1/18**

|  |  |
| --- | --- |
| **W6** | * Make sure everyone has visual studio and can use their emulator * Create the general class structure and first page of app * Explored the front end aspect, how buttons work, & how text can be outputted * Set up the database structure |
| **W7** | * Create the local connection to the app from the database * Create screen navigation system * Create more screens for app   + Login screen & New User Screen   + Task Screen * Create UI diagrams |
| **W8** | * Finish making basic screens - Sammy   + Edit Chore Screen & Group based screens - Sammy * Implementing php scripts that will send correct queries to database - Kyle * Users should be able to login, new users be registered, chores be registered and updated, figure out what chores are due this week - Michael * Attempt to make the app completely mobile mainly hosting the php files on heroku and calling them from the app - Vicki * User ability to add/remove tasks from calendar - Sammy   + Add priority flags - Sammy   + Assign a deadline - Sammy   + Assign to specific user - Sammy * Update UML diagrams for documentation, add new ones as needed - Team |
| **W9** | * Work establish group server interactions   + Joining, creating, and inviting new groups * Focus on testing and verifying the code and * Creating lower level code specific documentation * Consider adding wants onto task list   + Filter task list by a specific user     - Must be simplistic as this may require ability to view profiles   + Filter through dates and tasks     - History of tasks     - Historical distribution     - repeating chores   + Look into “possible wants” features: weekly/monthly views or chat |
| **W10** | * Anything that is quick and “easy”   + Possibly feature of weekly/monthly view   + Possibly a group chat/ memo feature |

**What we actually did:**

|  |  |
| --- | --- |
| **W6** | * Getting visual studio and emulators working. Figuring out the issues * Created the classes and defined some functions mainly chore class * Created one layout that will take UI and then print out * Attempted to implement sqlite database but failed |
| **W7** | * Pivoted to mysql database and set up connection to a remote database on a personal computer due to sqlite only being local which is not what we wanted * Created the database structure * Figured out that mysql database is not directly supported by android development since it supports sqlite * Researched alternatives figured out needed to create a REST API so that app could call api which queries into mysql database * Created simple php scripts to insert hardcoded data to database. Tested this with a button being pushed but only local therefore needed to set up a wamp server * Researched how to host the php scripts on heroku so that it can be fully online   + Successfully set up hosting for said php scripts   + Found out that perhaps the API needs to follow a structure to communicate with .NET * Learned how to go from screen to screen like from login screen to home screen and implemented this in a few screens * Created more layouts and connected them all. Developed the basic UI plan in drawings * Finished implementing the class functions? |
| **W8** |  |
| **W9** |  |
| **W10** |  |