Fall 2013

# Week ??

* Poster session
* Team presentation

# Week 1

* Read Continuity Report
* Read Developer’s Guide
* Get added to the github repo
* Begin Environment setup
* Begin reading up on technologies

# Week 2

* Make a commit to the github repo
* Complete environment setup
* Continue reading up on technologies
* Pick your subteam

# Week 3

## Backend

* Pupil detection
* Sclera Detection
* Make more robust error handling

# Week 4

## Backend

* Pupil detection
* Sclera Detection
* Acquire some photos that mimic actual input photos

# Week 5

## Backend

* crescent detection
* cataract detection
* Test working functionality on photos
  + ensure high success rate

# Week 6

## Backend

* Cataract Detection
* Finish and polish other feature extraction
  + Ensure high success rate

# Week 7

## Backend

* Integrate with front end
* Error checking and testing
* Develop machine learning plan/strategy/algorithm

# Week 8

## Backend

* Begin implementing machine learning
* Continue testing (if necessary)
* Update documentation

# Week 9

## Backend

* Get a good working copy of machine learning
* Begin continuity report

# Week 10

## Backend

* Finish continuity report
* Hone machine learning

~~Spring 2013~~

# Week 5

## Backend

* ~~finish and test eye detection script~~
* ~~lay out software architecture/structure~~

# Week 6

## Backend

* finish and test pupil detection
* ~~implement software architecture~~
  + ~~refactor code to fit within architecture~~

# Week 7

## Backend

* finish and test sclera detection
* ~~Finish Developer’s guide~~
* ~~Finish draft of pupil detection~~
* ~~Correct eye detection errors~~
* ~~implement controller layer~~
* ~~Continue working on developer’s guide~~

# Week 8

## Backend

* finish and test crescent
* finish and test sclera detection
* Finish Developer’s guide

# Week 9

## Backend

* finish and test cataracts/cloudiness detection

## Everyone

* ~~Write continuity report~~

# Week 10

## Backend

* finish and polish feature extraction

# Next Quarter(s)

* Integrate backend and front end
* Use machine learning with the features we have extracted to differentiate between healthy and unhealty eyes
* Deploy and test application