

**Iteration 1 Feature Set Plan**

Team 5

Hector Curi, John Politz, Nick Morris, Taylor Bishop, Tyler George

**Week 1**

**Back End: Create Databases and preliminary SQL/SLIM**

Taylor:

* Set up API - We will be using SLIM for the framework. This includes setting up the server’s PHP configuration to work with it
* Create and populate DB - Create the SQL file implementing the database system and populate it accordingly for testing
* Set up login - Set up login function in SLIM API. Will not be able to test this yet until design is done in week 2
* Contact form - Create API functionality to email
* Implement PDF.js and make sure it works accordingly - use dummy site to figure functionality and learn exactly how it works

Tyler:

* Set up registration - Set up registration function in SLIM API. Will not be able to test until week 2
* Create searchterms functionality - Need back-end support to pull database for group information

John:

* Set up presentation pulling - Create API function to pull presentation URL and info from database and send it to front-end
* Establish connection to API from front end - for login, registration, pulling presentations

**Front End: Create all html/php pages and create styling**

John:

* Create site-wide CSS file - Creating the CSS styling structure that the website will use: includes color scheme, placement of navigation bars, and styling for borders / text / etc.
* Implement JavaScript Validation - Implement framework used in Taco Truck to be used for registration and login
* Create front page design to be used a basis for rest of the site
* Create presentation viewer page design with blank spaces for week 2/iteration 2 implementation

Nick:

* Create HTML for:
  + user
  + new
  + invite
  + user-profile
  + contact
* Research ways to implement searchterms and begin creating HTML for it

Hector:

* Clean up paper prototype according to suggestions made by testers and the team
* HTML for:
  + edit
  + present
  + p-afterview
  + afterview
* Find appropriate photo gallery framework - A photo gallery that fits the look of the prototype and has adequate functionality to work with the presentation builder

**Week 2**

**Back End: Finish Databases and link api stuff with front end**

Taylor:

* Test login, registration and fix accordingly - With the front-end design finished we will be able to test back-end functionality
* Set up Drive API support - Create functionality to pull and push documents to and from Google Drive, testing it accordingly on a dummy site and making sure it functions properly on the webserver
* Drive API functionality should be done to upload presentations, upload notes
* Set up group information for users - Create API function to pull information on which groups a user is in and test it accordingly
* Help John implement PDF.js functionality on the front-end

Tyler:

* Continue work on searchterms functionality - By the end of the week, the API should be able to pull all relevant groups from the database and send them via JSON to the front-end
* Set up API functionality to pull a user’s profile information, and to update it accordingly. Test with front-end

John:

* Set up JS functionality to pull which groups a user is a pra
* Implement PDF.js to parse uploaded presentations into various slides
* Create API functionality for code-checking group joins against the database

**Front End: Finish all styling and link more advanced api components**

John:

* Implement advanced api components
* Provide support for back end team for adding elements into existing page designs
* Create JS functionality for code generation when joining groups

Nick:

* Finish HTML for searchterms page to show results from database implementation
* Finish up design on designated webpages from week 1

Hector:

* Implement photo gallery - Make sure the photo gallery works on the pages created the week before, including scrolling and such
* Finish up design on designated webpages from week 1, link it with PDF.js work done by John and Taylor
* Work on notes functionality for viewing presentations and work with Taylor to begin saving it to Google Drive