



Sprint 1 Testing Document

CS307 - Team 23

Arshpreet Singh
Kaushik Ramachandran
Luciano Handal Baracatt
Pratyaksh Motwani
Sadiq Ahmed A Albinalshaikh

Sprint 1 Overview

During this sprint, we aim to build the bare bones of our web application by setting up the main and local environment. Initially, most of our time will be spent on researching more about setting up Flask locally for every team member and use it with the Firebase wrapper, Pyrebase. Afterwards we will start writing the user interface templates (Web pages) and implement the authentication system. Having the templates ready on Sprint 1 assists us in focusing on the future backend user stories like the search engine and the review system. In addition, the review system is dependent on the authentication where users cannot post unless they are logged in. Therefore, we hope to complete it by the end of Sprint 1. Our goal for authentication is to make only Purdue students sign up and/or log in which will satisfy the Purdue exclusivity goal for the project.

SCRUM master: Sadiq Ahmed A Albinalshaikh

Meeting Schedule: Mondays at 2:30pm - 4:30pm. Additional meetings are planned as needed.

Risks and Challenges

Some of the key challenges we will face during this sprint is setting up the environment and background infrastructure for our web application, and integrating the back end and front end of our application. The biggest challenge for this sprint is the Purdue email verification and ensuring our promise that our reviewers will only be customers from Purdue University. We will ensure that adequate verification and authentication is done to provide a holistic and true academic review of the courses offered at Purdue University

User Story #1

As a user, I would like to sign up for an *easyA* account.

	Task	Estimated time	Owner
1	Create a sign-up page	3 hrs	Luciano
2	Create a user database	4 hrs	Sadiq
3	Connect the database to add new users	3 hrs	Pratyaksh
4	Test if the sign up page submits the valid user input and records it to the database correctly.	3 hrs	Sadiq
5	Test if the user database handles queries correctly and respond as expected per query.	2 hrs	Kaushik
6	Test if user data and encrypted password are being stored securely in the user database.	2 hrs	Kaushik

Acceptance Criteria:

1. If the sign up was implemented and tested correctly, the user interface should be user friendly. If the user did not have an account they should be able to find the sign up page, fill in their credentials, and successfully submit their information easily.
2. If the user database was created and tested correctly, any queries to the user database should be satisfied appropriately when a user logs in, signs up, verifies their account, or changes their password.
3. If the user database was created and tested correctly, when a user signs up, their data and encrypted password should be recorded on the database. If they or someone else tries to sign up with the same email, they will get an error saying that the email was already been used.

Testing Status:

- **Task:** Click On Sign Up Button as a Guest

Nature : Manual Testing

Expected: Redirect guest to Sign Up page with a user interface interactive to accommodate user sign up and provide necessary inputs to procure and parse data from the user.

- **Task:** Click On Sign Up Button as a Guest with a taken email

Nature : Manual Testing

Expected: Display an error message on the sign up page 'As [email] has already been used' and prevent user sign up.

- **Task:** Click on Sign Up Button on Sign Up Page

Nature : Manual Testing

Expected: If the email has not already been taken, user data should be able to successfully submit their information

- **Task:** Data Pipeline to Database Test

Nature : Manual Testing

Expected: Once a user has successfully signed up the database should successfully display the user sign up and cache the data in a systematic manner with password encryption to ensure user privacy

- **Task:** Test 'already have an account, 'Log In' redirection on Sign up Page

Nature : Manual Testing

Expected: Redirect the current user to the sign in page

User Story #2

As a user, I would like to log in to my account.

	Task	Estimated time	Owner
1	Create a log in page	3 hrs	Arshpreet
2	Setup Firebase Authentication for log in submissions and format (Email and Password)	3 hrs	Sadiq
3	Forbid non-verified accounts to log in	2 hrs	Sadiq
4	Test if the log in page submits the valid user input and checks if it exists in the user database	2 hrs	Pratyaksh
5	Test if logging in compares the credentials correctly the recorded data in the user database.	2 hrs	Pratyaksh
6	Test if a non-verified account could login or not.	3 hrs	Kaushik

Acceptance Criteria:

1. If the login page was implemented and tested correctly, the user interface should be user friendly. Whenever the user wanted to log in they should be able to navigate to the log in page, fill in their log in credentials, and successfully log in easily.
2. If the Firebase Authentication was implemented and tested correctly, the user should be able to log in with the correct credentials. If they use the incorrect combination of email and password they should get an error saying so.
3. If the verification system was implemented and tested correctly, the non-verified account should not be able to log in unless they verify their email.

Testing Status:

- **Task:** Click On Log In Button as a User

Nature : Manual Testing

Expected: Redirect User to Sign In page with a user interface interactive to accommodate user sign up and provide necessary inputs to procure and parse data from the user.

- **Task:** Click On Log In Button as a Guest with a incorrect credentials

Nature : Manual Testing

Expected: Display an error message on the sign up page 'incorrect credentials' and prevent the user from signing in.

- **Task:** Click on Sign Up Button on Sign Up Page

Nature : Manual Testing

Expected: If the email has not already been taken, user data should be able to successfully submit their information

- **Task:** Data Pipeline to Database Test

Nature : Manual Testing

Expected: Once a user has successfully signed in, the database should successfully acknowledge the user sign in and create a user session.

- **Task:** Test non-verified account log in unless they verify their email.

Nature : Manual Testing

Expected:

Expected: Unverified users should not be able to sign up and display an error message displaying that the user has not verified account.

User Story #3

As a user, I would like to log out from my account.

	Task	Estimated time	Owner
1	Create a log out button	3 hrs	Arshpreet
2	Set the user as logged out (guest)	3 hrs	Kaushik
3	Refresh the page and re-render the guest user interface after logging out	3 hrs	Kaushik
4	Test if the logout button functions as expected	2 hrs	Pratyaksh
5	Test if the user becomes a guest user after logging out	2 hrs	Sadiq

Acceptance Criteria:

1. If the log out button was implemented and tested correctly, the user should be able to find the log out button easily on the user interface.
2. If the log out functionality was implemented and tested correctly, the user should be able to set as guests whenever they click on the log out button.
3. If the log out functionality was implemented and tested correctly, after the user logs out successfully their user interface should change accordingly to the guest user interface. The guest user interface will show the login and sign up buttons again and limit the user from using any logged in user services.

Testing Status:

- **Task:** Click On Log Out Button as a Signed in User

Nature : Manual Testing

Expected: Redirect User to Sign In page with user interface interactive to accommodate user sign up and provide necessary inputs to procure and parse data from the user.

- **Task:** Check User status after log out

Nature : Manual Testing

Expected: Redirect User to home page as a guest user

- **Task:** Home Page status as a guest User

Nature : Manual Testing

Expected: The User should no longer be able to see his email tile on the top right corner and it must be replaced with log in and sign up buttons and treat the user as a guest.

User Story #4

As a user, I would like to change my password after logging in.

	Task	Estimated time	Owner
1	Include “Change Password” option in the account drop down menu	4 hrs	Arshpreet
2	Create a change my password page	5 hrs	Luciano, Arshpreet
3	Check if the old password matches the users’ credentials	2 hrs	Arshpreet, Pratyaksh
4	Update the account’s password in the database with the new encrypted password.	2 hrs	Arshpreet, Pratyaksh
5	Test if the database updates the correct user’s password	5 hrs	Kaushik

Acceptance Criteria:

1. If the dropdown menu was implemented and tested correctly, when the user clicks on the menu button after logging in they should be able to see “Change Password” option and redirected correctly to the change password page.
2. If the change password user interface was implemented and tested correctly, the user should be able to enter their old and new passwords and submit them successfully.
3. If the user inputs the incorrect old password they should get an error saying so.
4. If the user submits their password change successfully, the user database should select the correct user and updates their old encrypted password with the new encrypted password appropriately.

Testing Status:

- **Task:** Ensure drop down menu allows user to change his password once logged in

Nature : Manual Testing

Expected: Display a password change and redirect user to password change password

- **Task:** Test if change password allows user to input old password incorrect and try to change password

Nature : Manual Testing

Expected: Display an error message on page 'Please enter the correct old password' and prevent user from changing password

- **Task:** Test if change password allows user to input old password correctly and changes password

Nature : Manual Testing

Expected: The user should change his password successful and be able to log in with his new password.

User Story #5

As a user, I would like to have an account menu for changing my password and logging out.

	Task	Estimated time	Owner
1	Create a drop down menu with “Change Password” and “Log out” options.	4 hrs	Luciano, Arshpreet
2	Make the options in the drop down menu redirect the user accordingly.	2 hrs	Luciano, Arshpreet
3	Create a button that shows up the drop down menu to the user.	4 hrs	Luciano, Arshpreet
4	Test if the “Change Password” Button successfully redirects to change password page	1 hr	Pratyaksh
5	Test if the “Log out” button functions correctly and logs the user out.	1 hrs	Pratyaksh

Acceptance Criteria:

1. If the user interface for the drop down menu was implemented and tested correctly, when the user clicks on the account menu button, the drop down menu should pop up and list all the available options.
2. If per option functionality were implemented and tested correctly, when the user clicks “Change Password” they should be redirect to the change password page. When the user clicks on “Log out” they should be logged out successfully.
3. Guests should not be able to see the account menu button this applies to logged out users as well.

Testing Status:

- **Task:** Ensure drop down menu allows user to make necessary modifications to his account or log out

Nature : Manual Testing

Expected: Display a drop down menu and display password change or log out for a signed in user

- **Task:** Ensure drop down menu redirects user to selected option Password Change or Log out

Nature : Manual Testing

Expected: Option “Change Password” should redirect the user to the change password page while “Log out” should log the user out successfully.

- **Task:** Test drop down menu is available only to logged in users and not guest users

Nature : Manual Testing

Expected: Display a drop down menu should not display after user logs out or is a guest.

User Story #6

As a user, I would like to know if I am logged in or not.

	Task	Estimated time	Owner
1	Show user's email and account menu button when the user is logged in	3 hrs	Luciano
2	Show the login and sign up buttons when the user is not logged in	3 hrs	Arshpreet
3	Set up the navigation for both the login and sign up button.	2 hrs	Arshpreet
4	Test the implemented buttons and account status visibility on all pages.	3 hrs	Pratyaksh

Acceptance Criteria:

1. A logged in user should be able to see their username on the top right corner of the web-app if they are logged in.
2. A guest user should be able to see the login and sign up buttons on the top right corner of the web-app. When the user clicks on either button they should be redirected to the correct page.
3. If the user authentication was implemented and tested correctly, the user should be able to post reviews and view ratings.

Testing Status:

- **Task:** Status Display for Users

Nature : Manual Testing

Expected: Display user email username on the right status button if a user is logged in

- **Task:** Status Display for Guests

Nature : Manual Testing

Expected: Display Log in and Sign up buttons, and redirect users to selected option on selection

- **Task:** Post Reviews and Leave Ratings for a signed in user

Nature : Manual Testing

Expected: Signed in users should be able to select courses, view ratings, and post ratings and reviews for the courses.

User Story #7

As a user, I would like to reset my password if I forget it.

	Task	Estimated time	Owner
1	Create a button in the login page for “Forgot Password?”	1 hrs	Arshpreet
2	Create a password reset page	2 hrs	Luciano
3	Check if the submitted email is in the user database	2 hrs	Pratyaksh
4	Generate a unique link that lets the user reset their account password and make it expire in 1 hour.	4 hrs	Pratyaksh
5	Email the unique link to the user’s email	2 hrs	Pratyaksh
6	Debug and test implementations using manual testing and unit tests.	5 hrs	Kaushik

Acceptance Criteria:

1. When the user clicks on the “Forgot Password?” they should be redirected correctly to the password reset page that asks for their email.
2. When the user submits an email for an account that is recorded in the database, they should receive an email with a link that expires in 1 hour. If the email does not exist in the database, then they should get an error saying so.
3. When the user clicks on the link in the email before its expiration, they should be directed to the password reset page where they enter their new password. If the link was expired, they should get an error saying so.
4. When the user submits their new password, the user database should select the correct user and update their old encrypted password with the new encrypted password.

Testing Status:

- **Task:** Forgot Password Redirection

Nature : Manual Testing

Expected: Redirect User to Forgot Password redirection page on clicking forgot password on the log in page.

- **Task:** Forgot Password for users with an account in the database

Nature : Manual Testing

Expected: Send an email to the user to reset password with a link that expires after an hour and allow user to log in with the reset password, if the link is expired they should get an error saying so.

- **Task:** Forgot Password for users with no account in the database

Nature : Manual Testing

Expected: Prevent user from resetting his password and do not send an email to that account.

- **Task:** Test Database for user new password

Nature : Manual Testing

Expected: When the user submits their new password, the user database should select the correct user and update their old encrypted password with the new encrypted password.

User Story #8

As a user, I would like to see a course rating in the course review page.

	Task	Estimated time	Owner
1	Create a page to reflect the current course offered at Purdue University	5 hrs	Luciano
2	Create a UI to reflect the course rating at the course page	3 hrs	Arshpreet
3	Retrieve the course rating from the database.	3 hrs	Kaushik
4	Test if the page correctly updates the the course rating to reflect on the module	1 hrs	Kaushik
5	Test if the page correctly displays the respective course rating	1 hrs	Pratyaksh

Acceptance Criteria:

1. When the user navigates to a specific course page, they should be directed to the correct course page.
2. When the user navigates to the course review page, they should be able to view the 5-star rating of the course.
3. When a specific course receives a review post, its rating should be updated both in the database and the user interface correctly.

Testing Status:

- **Task:** Search for Course

Nature : Manual Testing

Expected: Redirect user to the selected course page

- **Task:** View the ratings on the Course Page

Nature : Manual Testing

Expected: When the user navigates to the course review page, they should be able to view the 5-star rating of the course.

- **Task:** Test Course Rating on Page when a user reviews the selected course.

Nature : Manual Testing

Expected: When the user navigates to the course review page and reviews a course, ratings should be updated both in the database and the user interface

User Story #9

As a user, I would like to see a course description in the course review page.

	Task	Estimated time	Owner
1	Create the course database	5 hrs	Sadiq
2	Create a UI to contain the course description for the respective course	2 hrs	Arshpreet
3	Record the course description in the database	2 hrs	Sadiq
4	Test if the page correctly displays the course description for the respective course	1 hrs	Kaushik

Acceptance Criteria:

1. The course database should contain a table that resembles the course information like name, rating, and description.
2. When the user navigates to a specific course page, the correct description should be requested from the database and rendered in the appropriate location in the page.
3. The web-app administrators should be able to add courses along with their descriptions to the database easily.

Testing Status:

- **Task:** Displayed Course Page attributes

Nature : Manual Testing

Expected: The Course Page should contain a table that resembles the course information like name, rating, and description.

- **Task:** Administrators should be able make modifications to course descriptions

Nature : Manual Testing

Expected: Web-app administrators should be able to add courses along with their descriptions to the database easily.

User Story #10

As a user, I would like to know that all reviews are from Purdue students.

	Task	Estimated time	Owner
1	Create an email verification system to verify that a signed up user owns a functional Purdue email.	12 hrs	Luciano, Arshpreet, Sadiq
2	Create a system to forbid non-Purdue users from signing up.	12 hrs	Luciano, Arshpreet, Sadiq
3	Limit unverified users to login until they are verified.	4 hrs	Kaushik, Pratyaksh
4	Show “Purdue Student” as the author name of all review posts.	2 hrs	Kaushik, Pratyaksh
5	Debug and test the implementations using unit testing for each task.	27 hrs	Sadiq, Pratyaksh, Kaushik

Acceptance Criteria:

1. When a new user signs up they should receive an email for verifying their account. The verification of the account is up to the real owner of the Purdue email.
2. When a user tries to sign up they should not be able to sign up unless they use an email that ends with “@purdue.edu” An error will forbid the user from signing up with an email that does not have the Purdue domain.
3. When a user tries to login with their unverified account, they should get an error saying that they must complete their verification to login.
4. On the course review page, reviews’ authors will be as “Purdue Student.”

Testing Status:

- **Task:** New user verification

Nature : Manual Testing

Expected: When a new user signs up they should receive an email for verifying their account.

- **Task:** Purdue User verification

Nature : Manual Testing

Expected: When a user tries to sign up they should not be able to sign up unless they use an email that ends with “@purdue.edu” and display an error message otherwise if email is incorrect or unverified.

- **Task:** Course Review Page User Display

Nature : Manual Testing

Expected: On the course review page, reviews’ authors will be titled as “Purdue Student.”

User Story #11

As an administrator, I would like only Purdue students with “@purdue.edu” emails to be able to sign up.

	Task	Estimated time	Owner
1	From a front-end point of view: Use regular expressions to only accept “XXX@purdue.edu” formats in sign up, log in, and password reset.	2 hrs	Sadiq, Arshpreet
2	Implement a backend input sanitization to verify the input again and avoid web attacks that would involve code injection.	4 hrs	Pratyaksh, Kaushik
3	Give an error to the user in case the input fails in either front-end or back-end verification,	2 hrs	Luciano
4	Debug and test the implementations using unit testing for each task.	12 hrs	Sadiq, Pratyaksh, Kaushik

Acceptance Criteria:

1. The user should only be permitted to sign up if their entered email ends with “@purdue.edu”
2. In case the user was able to bypass the regular expression conditions, the backend server should check the input again if it was a Purdue email. Also, it should treat any inputs as text where it will not execute any code injections.
3. When the user submits credentials that conflict with the conditions above, then they should get an error saying that the input is invalid.

Testing Status:

- **Task:** Purdue User verification

Nature : Manual Testing

Expected: The user should only be permitted to sign up if their entered email ends with “@purdue.edu and if the credentials don't match the requirements an error should be displayed.

- **Task:** Bypass Verification

Nature : Manual Testing

Expected: In case the user was able to bypass the regular expression conditions, the backend server should check the input again if it was a Purdue email.

User Story #12

As an administrator, I would like newly signed up users to verify their ownership of their Purdue emails.

	Task	Estimated time	Owner
1	Verify that the email exists and working.	4 hrs	Sadiq
2	Send an email confirmation to the newly signed up user with a verification link that expires in 1 hour.	8 hrs	Arshpreet, Sadiq
3	Give an error to the user in case they use an non-existing email, login unverified, or use an expired verification link.	4 hrs	Luciano
4	Delete unverified users after their confirmation links expire.	4 hrs	Arshpreet, Sadiq
5	Debug and test the implementations using unit testing for each task.	16 hrs	Sadiq, Pratyaksh, Kaushik

Acceptance Criteria:

1. When a user sign up, an email should be sent to the user with a confirmation link that is generated by a random token.
2. When the user clicks on the link, the server should update the database to mark the user as verified then show the user a “Successful Confirmation” page.
3. When the user input a non-existing email, they should get an error saying invalid email.
4. If the user uses a link after 1 hour from signing up, they should get an error saying invalid or expired link.

Testing Status:

- **Task:** Sign Up Verification

Nature : Manual Testing

Expected: When a user signs up, an email should be sent to the user with a confirmation link that is generated by a random token and on confirmation the server should update the database to mark the user as verified then show the user a “Successful Confirmation” page. If the user inputs a non-existing email, they should get an error saying invalid email.

- **Task:** Confirmation Email

Nature : Manual Testing

Expected: If the email is expired, the user should get an error saying invalid link.

User Story #13

As a user, I would like to have an easy navigitable user interface throughout the website.

	Task	Estimated time	Owner
1	Design the web and file storage necessary for an intuitive flow	2 hrs	Luciano, Arshpreet
2	Set up the HTML skeleton and navigation between the different pages	4 hrs	Luciano
3	Add necessary CSS to allow user to have an identifiable visual contrast	2 hrs	Luciano
4	Add necessary JavaScript to make the website more responsive for to the user	3 hrs	Luciano
5	Debug and test the responsiveness and syntax of the web pages	4 hrs	Luciano

Acceptance Criteria:

1. Navigation for all users should be intuitive and require minimal guidance.
2. Users should not be stuck on any page without a way to go back to previous pages.
3. The web page theme should look pleasant without having blue-ish or bright colors that induce eye strain.

Testing Status:

- **Task:** Easy to navigate and efficient user experience and interface

Nature : Manual Testing

Expected: Navigation should be minimal for all tasks and the user should be able to easily navigate to the previous page if needed.

User Story #14

As a user, I would like to log in/sign up from any webpage in *easyA*.

	Task	Estimated time	Owner
1	Set up a navigation bar that contains the login and sign up buttons	2 hrs	Arshpreet
2	Set up necessary forms for the user to submit their information	4 hrs	Arshpreet
3	Make buttons disappear according the user's status if they were logged in or not	4 hrs	Arshpreet
4	Test that the buttons work as expected on all pages	1 hr	Luciano, Arshpreet

Acceptance Criteria:

1. Logging in and signing should be easy and intuitive from any page in the website.
2. Users should be redirected to the main page after logging in/signing up.
3. Guests should see the login and sign up buttons on all pages.

Testing Status:

- **Task:** Log in and Sign Up redirections

Nature : Manual Testing

Expected: Users should be able to sign up and log in from the pages that allow the features, at will. Users should be redirected to the main page and have an easy interface to allow login and sign up features.

User Story #15

As a user, I would like to log out from any webpage in *easyA*.

	Task	Estimated time	Owner
1	For logged in users, make the account menu that contains the log out button visible on all pages	2 hrs	Arshpreet
2	Check the status of the users if they are logged in or not	2 hrs	Arshpreet
3	Make the account email and account menu button disappear after logging out	2 hrs	Arshpreet
4	Test that logging out works properly on all pages	1 hr	Luciano, Ashpreet

Acceptance Criteria:

1. Logging out should be easy and intuitive from any page on the website.
2. Logging out should redirect the user to the home page.
3. The account email and account should disappear and the log in and sign up button show up again.

Testing Status:

- **Task:** Logging Out

Nature : Manual Testing

Expected: Logging out should be easy and intuitive from any page in the website and should redirect the user to home page and display the log in and sign buttons.