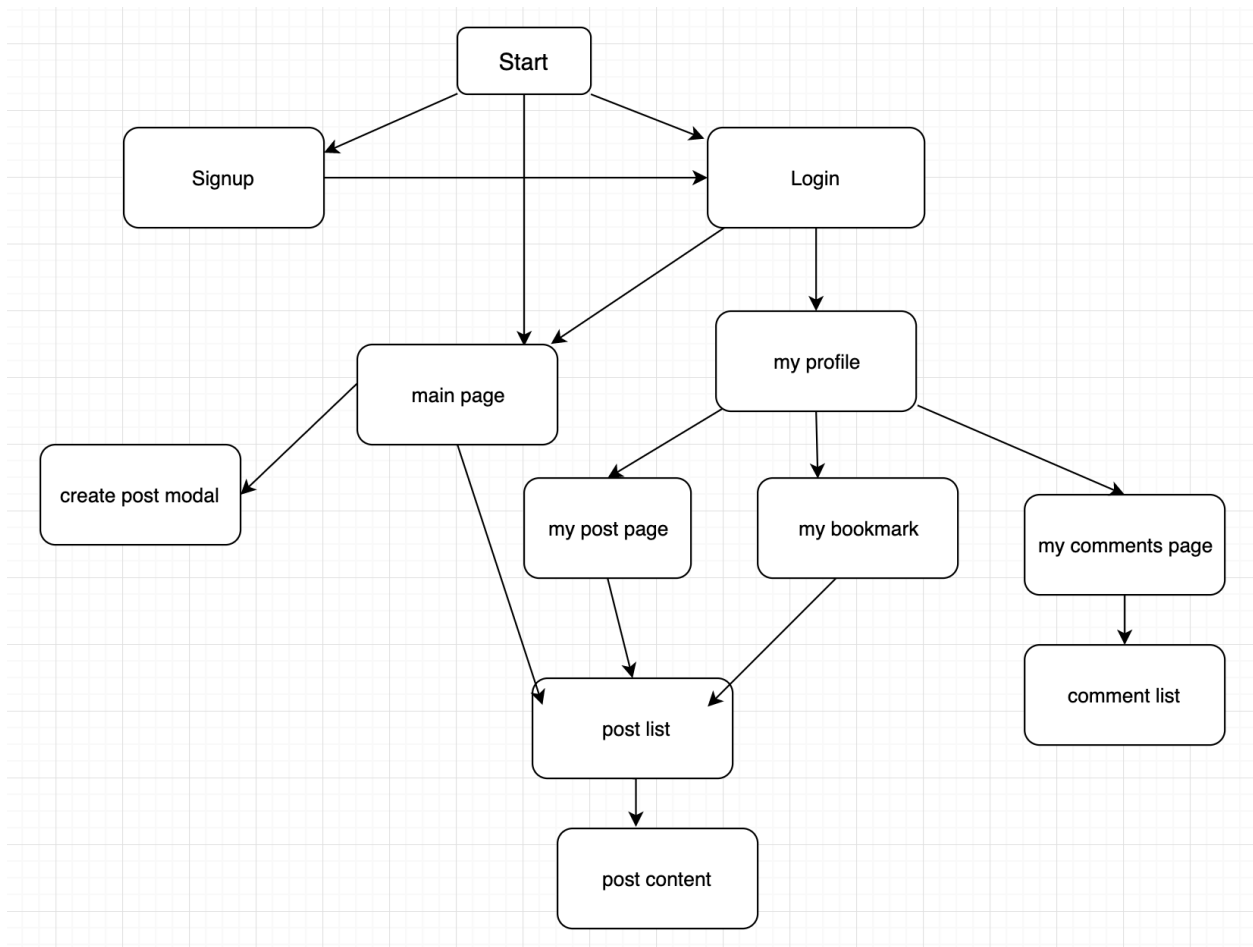


Incremental and Regression Testing

1. Classification of Component

1.1 Component Structure



Start:

- Input: None
- Output: None
- Parent Dependency: None
- Child Dependency: Login, Signup, main page

Signup:

- Input: username, password
- Output: create a new user, update user session
- Parent Dependency: Start
- Child Dependency: Login

Login:

- Input: username, password
- Output: login and update user session
- Parent Dependency: Start
- Child Dependency: main page, my profile, create post modal

main page:

- Input: search query
- Output: post list
- Parent Dependency: Start, Login
- Child Dependency: create post modal, full post list

create post modal:

- Input: post content
- Output: A new item in post list with userid and post id
- Parent Dependency: main page
- Child Dependency: None

My profile:

- Input: username, password
- Output: None
- Parent Dependency: Login
- Child Dependency: create post modal, post list

my post page:

- Input: username, password
- Output: post list
- Parent Dependency: my profile
- Child Dependency: post list

my bookmark page:

- Input: username, password
- Output: post list
- Parent Dependency: my profile
- Child Dependency: post list

my comments page:

- Input: username, password
- Output: comment list
- Parent Dependency: my profile
- Child Dependency: comment list

post list:

- Input: username, password, search query
- Output: a list of posts
- Parent Dependency: main page
- Child Dependency: post content

post content:

- Input: Post id
- Output: content of the post
- Parent Dependency: full post list, my post list, my collected post list
- Child Dependency: None

1.2 Form of Incremental Testing

For this project and our scenario, we decided to go with top-down incremental testing because of the top-down nature of user interface components. We tested modules from the top module where the user can register and login to their account, and subsequently tested features that require a user to be logged in to work properly (like My Posts, My Bookmarks, etc.). This not only follows the logic flow of how our application operates and is structured, but also allows for development to be carried out in a natural and proper order. This testing strategy made sure that our major top level components work and subsequent child components that rely on those parent components also worked when created.

2. Incremental Testing and Regression Testing

Front End

Module	Start Module		
Incremental Testing			
Defect #	description	severity	How to correct
	This module does not contain any defects		
Regression Testing			
Defect #	description	severity	How to correct
	This module does not contain any defects		

Module	Sign Up Module		
Incremental Testing			
Defect #	description	severity	How to correct
1	Username field could be blank on submission	Important	Disabled "Register" button unless field was populated
2	Password field could be blank on submission	Important	Disabled "Register" button unless field was populated
3	Username field could be only whitespaces on submission	Important	Disabled "Register" button unless field contained non-whitespace characters (added "not only whitespace" checking)
4	Password field could be whitespace on submission	Important	Disabled "Register" button unless field

			contained non-whitespace characters (added "not only whitespace" checking)
5	Password field could contain non-alphanumeric characters	Important	Added state logic to display error message when password field contains invalid characters
Regression Testing			
Defect #	description	severity	How to correct
	No defects		

Module	Login Module		
Incremental Testing			
Defect #	description	severity	How to correct
1	Username field could be blank on login submission	Important	Disabled "Login" button unless field was populated
2	Password field could be blank on login submission	Important	Disabled "Login" button unless field was populated
3	Username field could be whitespace on login submission	Important	Disabled "Login" button unless field contained non-whitespace characters
4	Password field could be whitespace on login submission	Important	Disabled "Login" button unless field contained non-whitespace characters
Regression Testing			
Defect #	description	severity	How to correct

	No defects		
--	------------	--	--

Module	Main page		
Incremental Testing			
Defect #	description	severity	How to correct
1	Loaders were not displaying properly while fetching post data	Important	Corrected logic for rendering placeholders, adjust CSS for placeholders
2	Pagination was incorrectly calculating page numbers when result from response was non-numeric	Workaround	Corrected parsing logic for the response data to disregard invalid responses
3	"Create Post" button was not disabled when no user is logged in	Important	Corrected state logic to disable the button when no user session is detected
Regression Testing			
Defect #	description	severity	How to correct
	No defects		

Module	post content page		
Incremental Testing			
Defect #	description	severity	How to correct
1	Placeholder loaders were not displaying correctly while fetching post metadata	Important	Corrected state logic to properly return placeholders while asynchronously fetching data
Regression Testing			

Defect #	description	severity	How to correct
	No defects		

Module	My post page		
Incremental Testing			
Defect #	description	severity	How to correct
1	Loaders were not displaying properly while fetching post data	Important	Corrected logic for rendering placeholders, adjust CSS for placeholders
2	Pagination was incorrectly calculating page numbers when result from response was non-numeric	Workaround	Corrected parsing logic for the response data to disregard invalid responses
3	Users could successfully access "My Posts" even after logging out	Important	Corrected state logic to instead render message saying you must be logged in to access this page
4	Server return empty post list even when there are users in the database and they have posted	Critical	change GET to POST with authentication header
Regression Testing			
Defect #	description	severity	How to correct
	No defects		

Module	My bookmark Page		
Incremental Testing			
Defect #	description	severity	How to correct

1	Loaders were not displaying properly while fetching post data	Important	Corrected logic for rendering placeholders, adjust CSS for placeholders
2	Pagination was incorrectly calculating page numbers when result from response was non-numeric	Workaround	Corrected parsing logic for the response data to disregard invalid responses
3	Users could successfully access "My Posts" even after logging out	Important	Corrected state logic to instead render message saying you must be logged in to access this page
Regression Testing			
Defect #	description	severity	How to correct
	No defects		

Back End

Module	User Module		
Incremental Testing			
Defect #	description	severity	How to correct
1	Removing a user from the database will fail if another user has bookmarked a post from the user being deleted	Critical	Added an algorithm to remove all bookmark entries that match the post ids of the posts created by the to be deleted user
2	When server performs password checking, it should compare the hashed version of input password to the stored password since unencoded password is not stored in the database	Critical	Apply hash() function to the input password when performing password check.
Regression Testing			

Defect #	description	severity	How to correct
1	Removing a user from the database table keeps their posts and bookmarks in other tables (breaks referential integrity of database)	Critical	Added the dependent: :destroy dependency to all of the user posts, bookmarks, comments. This will remove the entries from the database once a user is removed

Module	Home Page		
Incremental Testing			
Defect #	description	severity	How to correct
1	When doing a search query on the homepage, the pagination calculation was incorrect	Important	Changed from posts.all.length to post.length to retrieve the correct post count
Regression Testing			
Defect #	description	severity	How to correct
	No defects		

Module	My Bookmarks		
Incremental Testing			
Defect #	description	severity	How to correct
1	When doing a search query on the homepage, the pagination calculation was incorrect	Important	Changed from posts.all.length to post.length to retrieve the correct post count
Regression Testing			

Defect #	description	severity	How to correct
	No defects		

Module	My Posts		
Incremental Testing			
Defect #	description	severity	How to correct
1	When doing a search query on the homepage, the pagination calculation was incorrect	Important	Changed from posts.all.length to post.length to retrieve the correct post count
Regression Testing			
Defect #	description	severity	How to correct
	No defects		