

Team: Henrik Larsen, Sean Hansberry, Casey Moher-Crook, Mark Ariniello

Title: Open Source Project Website

Project Summary: A website containing a list of open-source computer science projects from Github. Users will be able to select specific projects and clone the projects from their source repository. Users will also be able to add projects to their favorites for future reference.

Project Requirements:

There are no business requirements

User Requirements		
ID	Requirement	Priority
USR-01	As a user, I want to save projects to my favorites so I don't have to search for them each log-in.	High
USR-02	As a user, I want to search a database for new projects so I can easily find projects.	Medium
USR-03	As a user, I want to be able to submit a project's URL and add it to my favorites for ease.	High
USR-04	As a user, I want to be able to filter search results by different criterias	Critical
USR-05	As a user, I want to be able to view a random project by clicking a button	Low
USR-06	As a Admin I want to be able to see who is currently logged in.	Medium
USR-07	As a Admin I want to be able to track site visits.	Low
USR-08	As a Admin, I want to be able to remove inactive users from login	Low

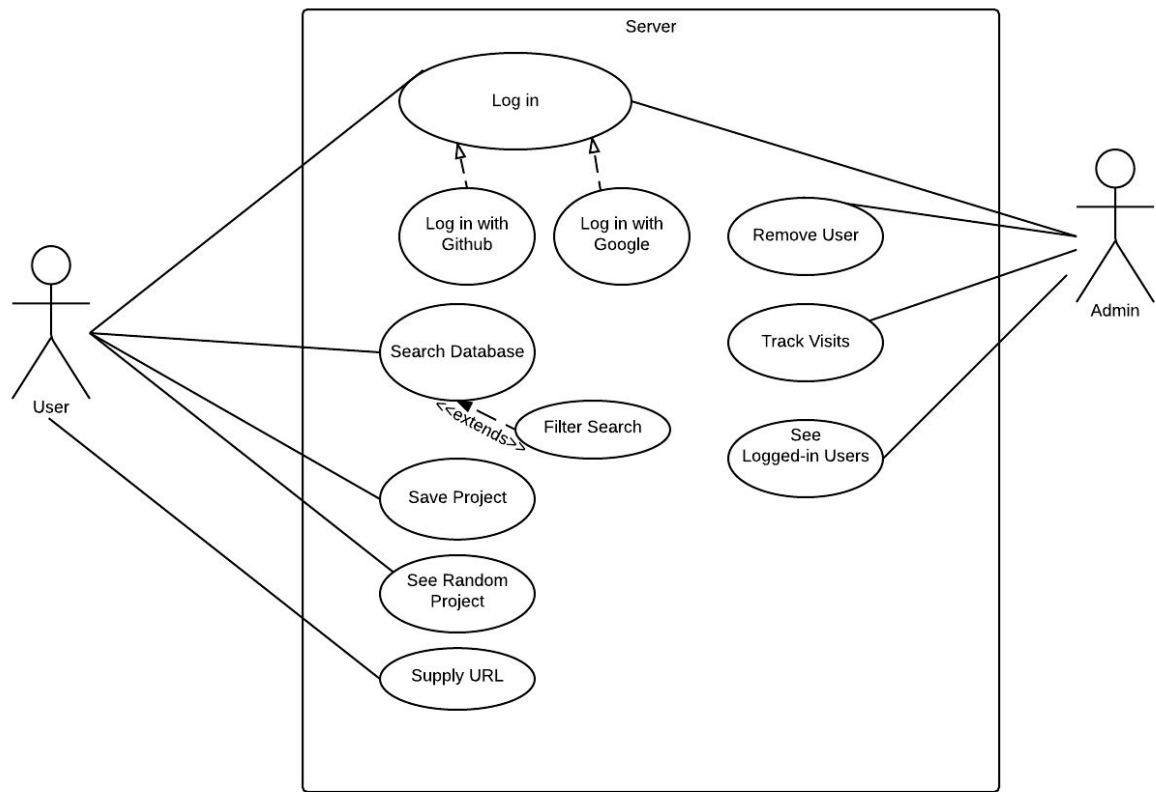
	database	
--	----------	--

Functional Requirements			
ID	Requirement	Topic Area	Priority
FR-001	A link to each source repository should be available to the user	Database	Medium
FR-02	Database populated by pulling projects title, url and readme, # of contributors, and last push from github via the github api	Database	Critical
FR-03	Logins are linked to a third party account, google+ or github	Security	High
FR-04	Database can be searched by any of the fields pulled in FR-02	User experience	Low

Non-Functional Requirements				
ID	Requirement	Topic Area	User	Priority
NFR-01	Experience should be the same across all web browsers.	User experience	All Users	Medium
NFR-02	As a developer I want to be able to reset or recompile the website	Server maintenance	Developer	Medium
NFR-03	As the developer I want the website to run without significant lag time when a user is interacting with the website, so that it is "reliable"	Server maintenance	Developer	Medium
NFR-04	As the developer, I want the website to be	Cryptography	Developer	Medium

	secure, so that user information is protected when using and interacting with the website.			
--	--	--	--	--

Use Case Diagram:



Use Cases:

Use Case ID:	USR-01
Use Case Name:	Save favorites
Description:	Have functionality that allows the user to save projects to their account so they can be viewed later.

Actors:	User
---------	------

Pre-conditions:	User is logged into system viewing the main page														
Post-conditions:	Project is marked as favorite														
Frequency of Use:	Daily by all users														
Flow of Events:	<table><tr><td></td><td>Actor Action</td><td>System Response</td></tr><tr><td>1</td><td>Navigate to open source project list</td><td>Display all open source projects</td></tr><tr><td>2</td><td>Open the specific project for viewing</td><td>New page with specific project being viewed</td></tr><tr><td>3</td><td>Button is pressed to save project to favorites</td><td>Project is saved to account under favorites</td></tr></table>				Actor Action	System Response	1	Navigate to open source project list	Display all open source projects	2	Open the specific project for viewing	New page with specific project being viewed	3	Button is pressed to save project to favorites	Project is saved to account under favorites
	Actor Action	System Response													
1	Navigate to open source project list	Display all open source projects													
2	Open the specific project for viewing	New page with specific project being viewed													
3	Button is pressed to save project to favorites	Project is saved to account under favorites													
Variations:	None														
Notes and Issues for Developer															
Developer Notes															

Use Case ID:	USR-02
Use Case Name:	Search Projects
Description:	As a user, I want to search the database for new projects so I can easily find projects.

Actors:	Site User
Pre-conditions:	User is logged in, Database is not empty
Post-conditions:	User sees a list of projects specific to the criteria given.

Frequency of Use:	Very often														
Flow of Events:	<table><tr><td></td><td>Actor Action</td><td>System Response</td></tr><tr><td>1</td><td>Enters a project title in the search bar</td><td>display a list of relevant project titles</td></tr><tr><td>2</td><td>Uses search criteria to filter search results</td><td>Filters the results according to the criteria</td></tr><tr><td>3</td><td>Clicks the search button</td><td>Displays the list of relevant projects</td></tr></table>				Actor Action	System Response	1	Enters a project title in the search bar	display a list of relevant project titles	2	Uses search criteria to filter search results	Filters the results according to the criteria	3	Clicks the search button	Displays the list of relevant projects
	Actor Action	System Response													
1	Enters a project title in the search bar	display a list of relevant project titles													
2	Uses search criteria to filter search results	Filters the results according to the criteria													
3	Clicks the search button	Displays the list of relevant projects													
Variations:	1 and 2 can be innerchanged or unused (either or both)														
Notes and Issues for Developer															
Developer Notes															

Use Case ID:	USR-03
Use Case Name:	Submit project to favorites
Description:	As a user, I want to be able to submit a project's URL and add it to favorites.

Actors:	User								
Pre-conditions:	User is logged in and has a working URL to a github project								
Post-conditions:	Project is marked as favorite on user’s account								
Frequency of Use:	Whenever the user wants to add their own project								
Flow of Events:	<table><tr><td></td><td>Actor Action</td><td>System Response</td></tr><tr><td>1</td><td>Navigate to</td><td>Show page</td></tr></table>				Actor Action	System Response	1	Navigate to	Show page
	Actor Action	System Response							
1	Navigate to	Show page							

	<table><tr><td></td><td>user account</td><td>listing user account information</td></tr><tr><td>2</td><td>Submit URL to a specific button</td><td>Gets project from URL and saves it under User's favorites</td></tr></table>		user account	listing user account information	2	Submit URL to a specific button	Gets project from URL and saves it under User's favorites
	user account	listing user account information					
2	Submit URL to a specific button	Gets project from URL and saves it under User's favorites					
Variations:	2: User can navigate to other parts of the website before going into their own account						
Notes and Issues for Developer							
Developer Notes							

Use Case ID:	USR-04
Use Case Name:	Filter Search Results.
Description:	As a user, I want to be able to filter search results by either total contributions or most recent contribution.

Actors:	Site User											
Pre-conditions:	User is logged in to site, then selects search.											
Post-conditions:	User is able to rearrange search results by a preferred criteria.											
Frequency of Use:	around half the time a user performs a search.											
Flow of Events:	<table><tr><td></td><td>Actor Action</td><td>System Response</td></tr><tr><td>1</td><td>Select search filters</td><td>marks filter options for query</td></tr><tr><td>2</td><td>Selects search</td><td>display list of projects, default supported by relevance to keyword</td></tr></table>				Actor Action	System Response	1	Select search filters	marks filter options for query	2	Selects search	display list of projects, default supported by relevance to keyword
	Actor Action	System Response										
1	Select search filters	marks filter options for query										
2	Selects search	display list of projects, default supported by relevance to keyword										

			according to filter
	3	click on a result	display result metrics.
Variations:			
Notes and Issues for Developer			
Developer Notes			

Use Case ID:	USR-05
Use Case Name:	Supply Random Project
Description:	As a user, I want to be able to view a random project by clicking a button.

Actors:	User								
Pre-conditions:	User is logged in, database is not empty.								
Post-conditions:	User sees a project and its information.								
Frequency of Use:	Daily								
Flow of Events:	<table><tr><td></td><td>Actor Action</td><td>System Response</td></tr><tr><td>1</td><td>Click “Random Project” button</td><td>display all metrics associated with project</td></tr></table>				Actor Action	System Response	1	Click “Random Project” button	display all metrics associated with project
	Actor Action	System Response							
1	Click “Random Project” button	display all metrics associated with project							
Variations:									
Notes and Issues for Developer									

Developer Notes	
-----------------	--

Use Case ID:	USR-06
Use Case Name:	View logged in users
Description:	Look at who is currently logged into the website as an admin

Actors:	User, Admin							
Pre-conditions:	There are users online, admin is logged in							
Post-conditions:	Admin sees a list of active users							
Frequency of Use:	Daily by admins							
Flow of Events:	<table border="1"> <thead> <tr> <th></th><th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1</td><td>Admin navigates to page to view online users</td><td>Shows a page listing all online users</td></tr> </tbody> </table>			Actor Action	System Response	1	Admin navigates to page to view online users	Shows a page listing all online users
	Actor Action	System Response						
1	Admin navigates to page to view online users	Shows a page listing all online users						
Variations:								
Notes and Issues for Developer								
Developer Notes								

Use Case ID:	USR-07
Use Case Name:	Track Users
Description:	As a Admin I want to be able to track total site visits.

Actors:	Admin
---------	-------

Pre-conditions:	Admin is logged in								
Post-conditions:	Admin finds how many users have visited the site since the counter started.								
Frequency of Use:	Every time a Admin logs in.								
Flow of Events:	<table><tr><td></td><td>Actor Action</td><td>System Response</td></tr><tr><td>1</td><td>Navigates to admin page</td><td>sees site metrcis.</td></tr></table>				Actor Action	System Response	1	Navigates to admin page	sees site metrcis.
	Actor Action	System Response							
1	Navigates to admin page	sees site metrcis.							
Variations:									
Notes and Issues for Developer									
Developer Notes									

Use Case ID:	USR-08
Use Case Name:	Remove users
Description:	Remove inactive users from database so their information is no longer stored

Actors:	Admin, User								
Pre-conditions:	Database is populated with user information, admin is logged in								
Post-conditions:	Deleted user can no longer exist								
Frequency of Use:	Whenever database maintenance is performed								
Flow of Events:	<table><tr><td></td><td>Actor Action</td><td>System Response</td></tr><tr><td>1</td><td>Admin views</td><td>A page</td></tr></table>				Actor Action	System Response	1	Admin views	A page
	Actor Action	System Response							
1	Admin views	A page							

		specific user	showing user information
	2	Admin deletes user from database via a delete button	User is removed from login database
Variations:	2: Admin can view other pages before navigating to user		
Notes and Issues for Developer			
Developer Notes			

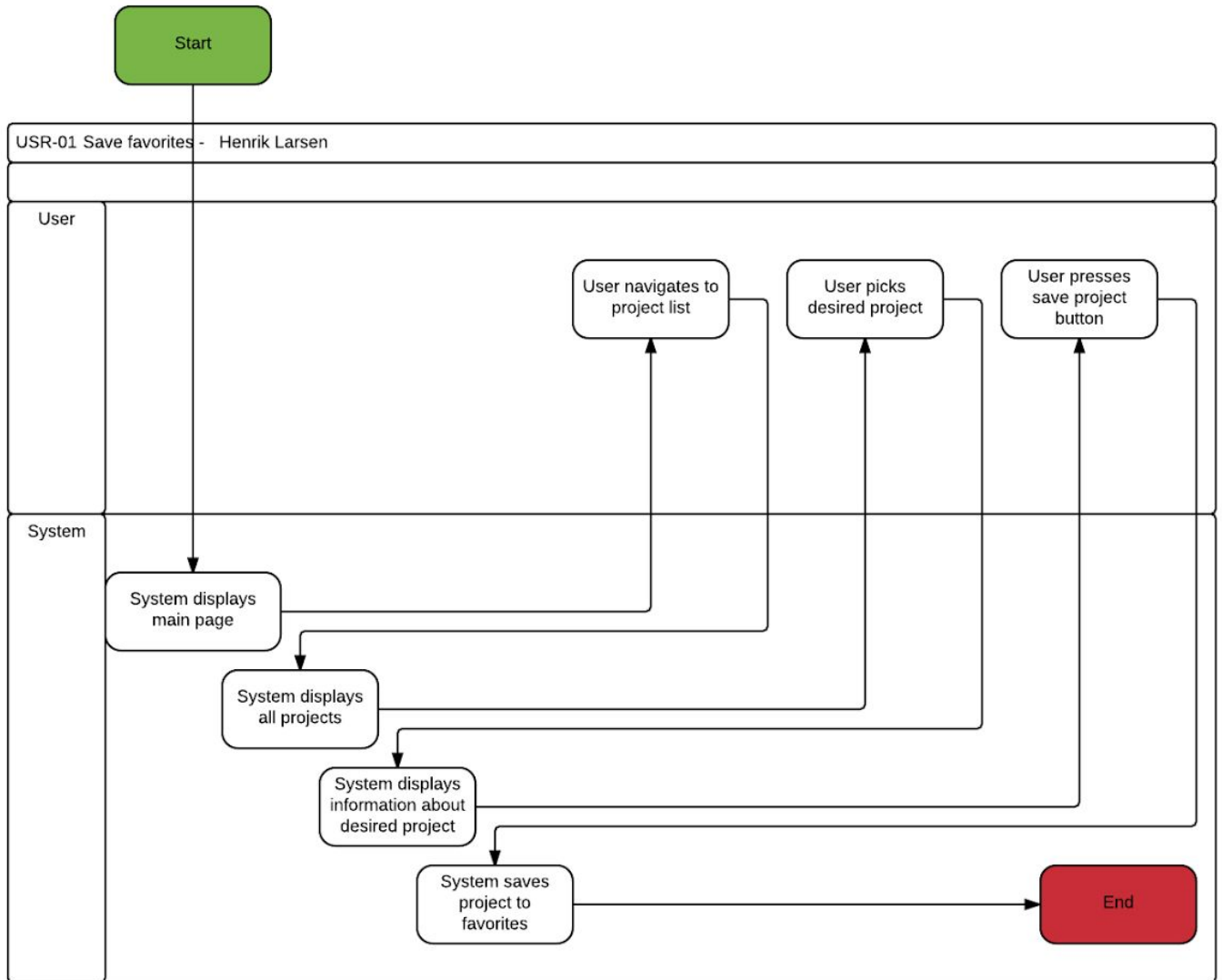
Data Storage:

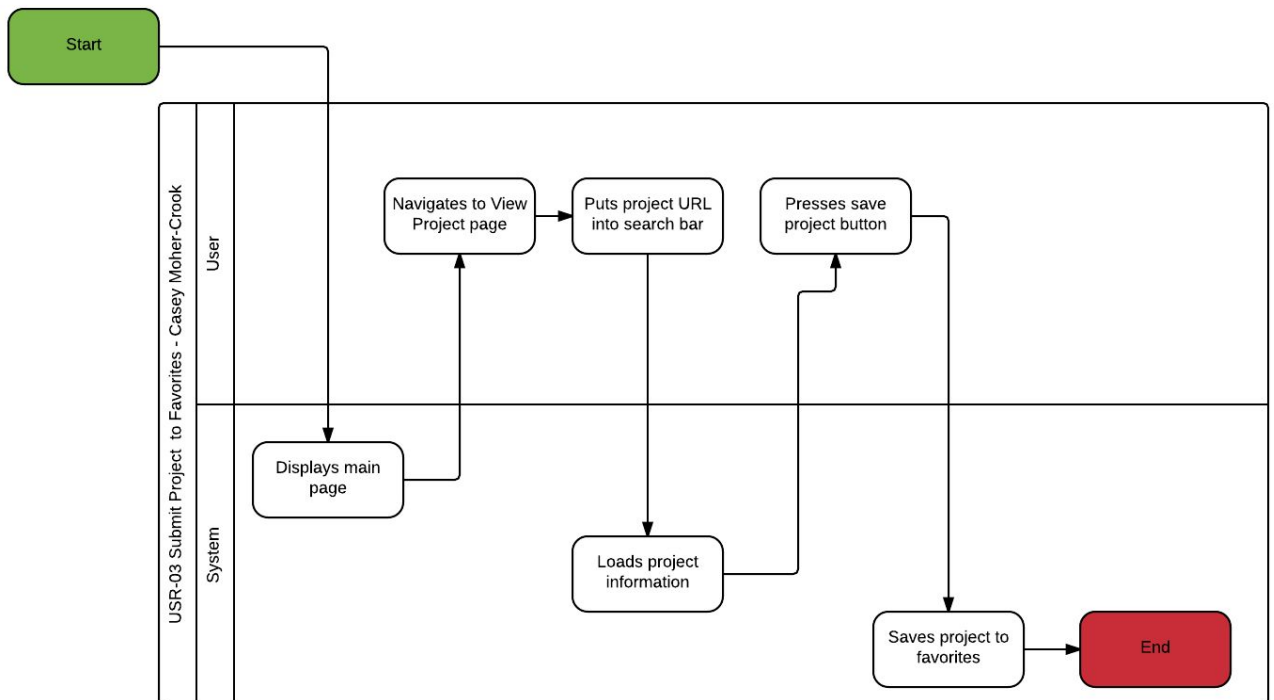
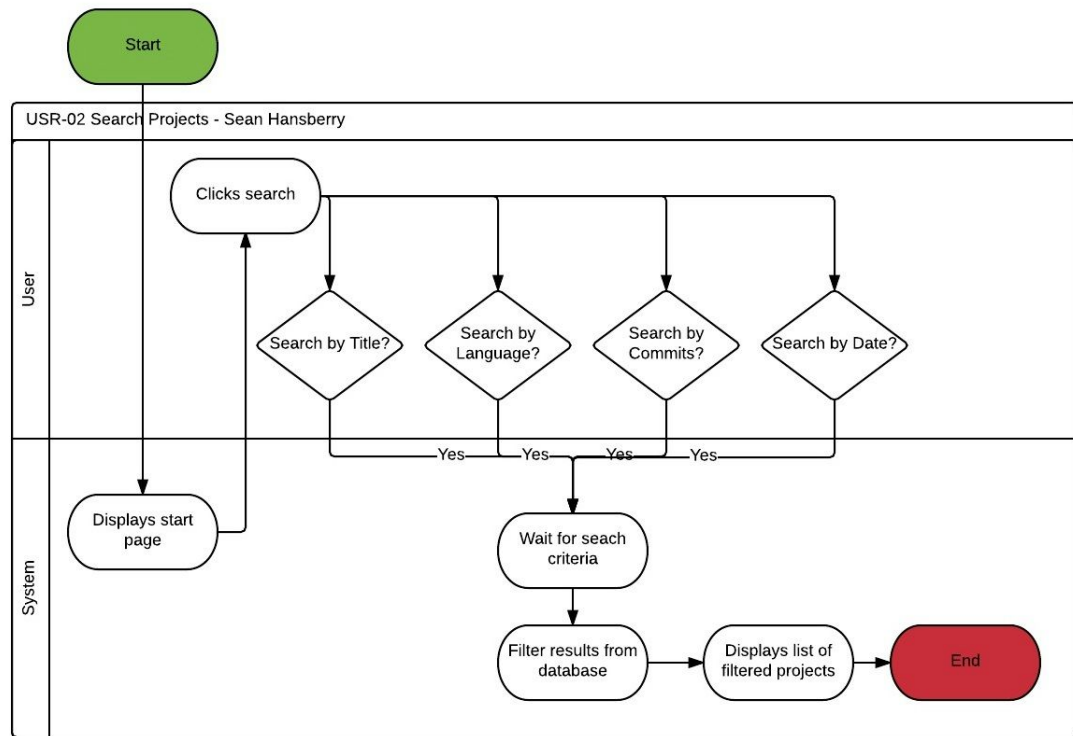
Data Storage: Will store information using MongoDB

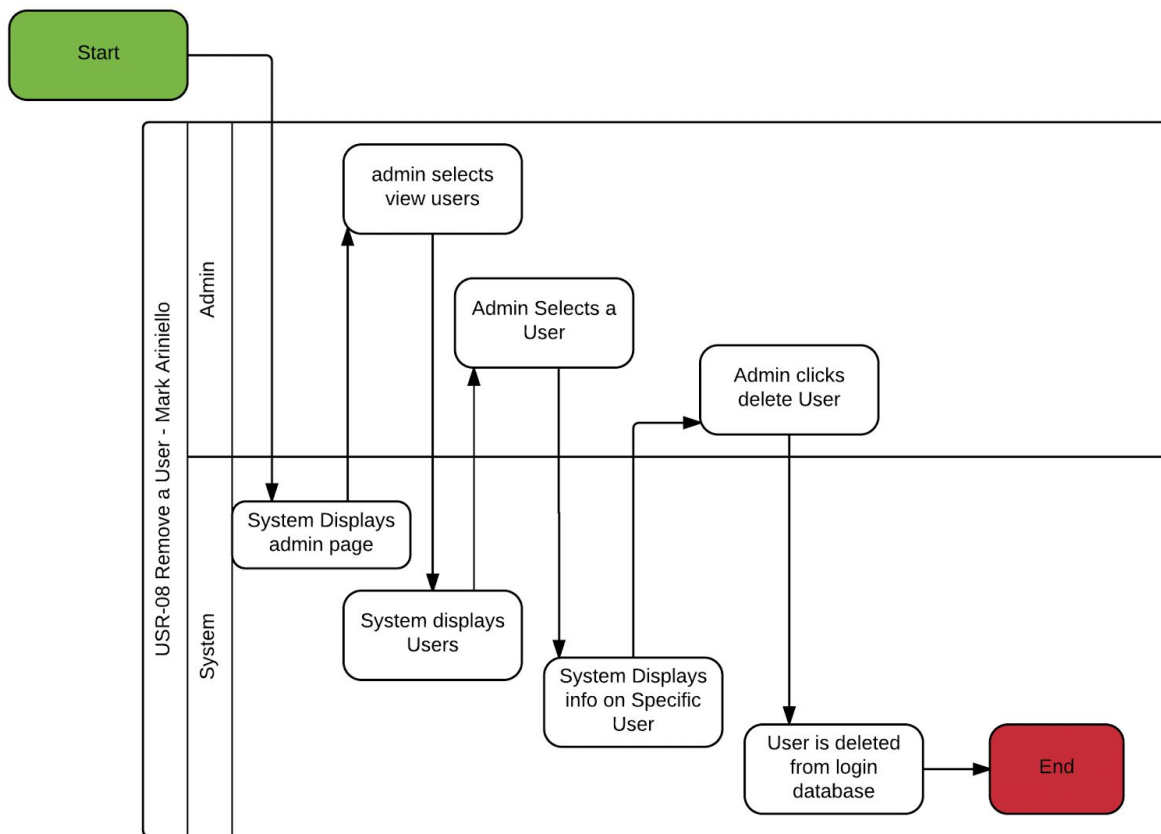
Classes:

- First class to store information about each project; url, readme file, title
- Second class to store information about each user; username, email, password, favorites

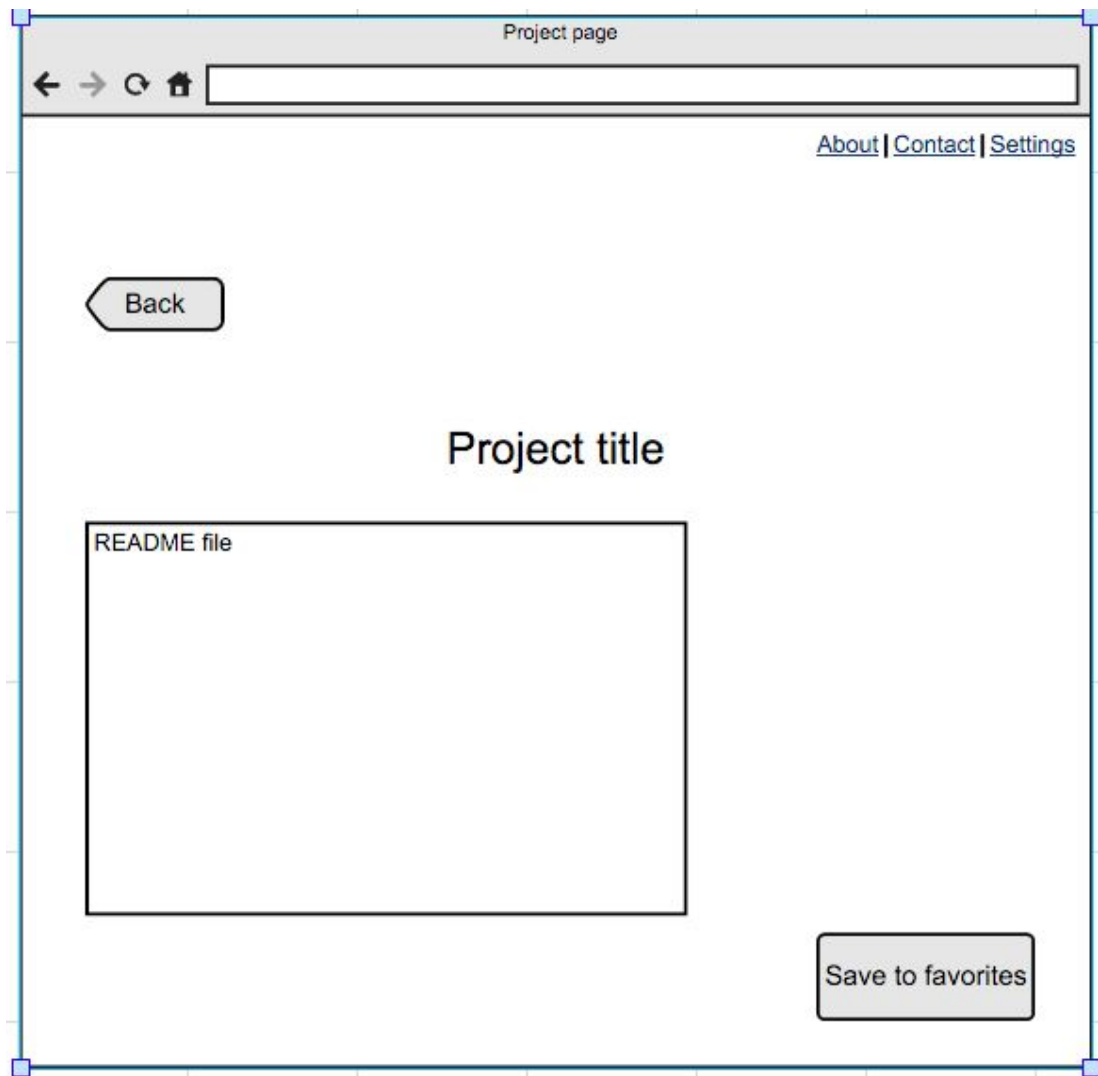
Activity Diagram:

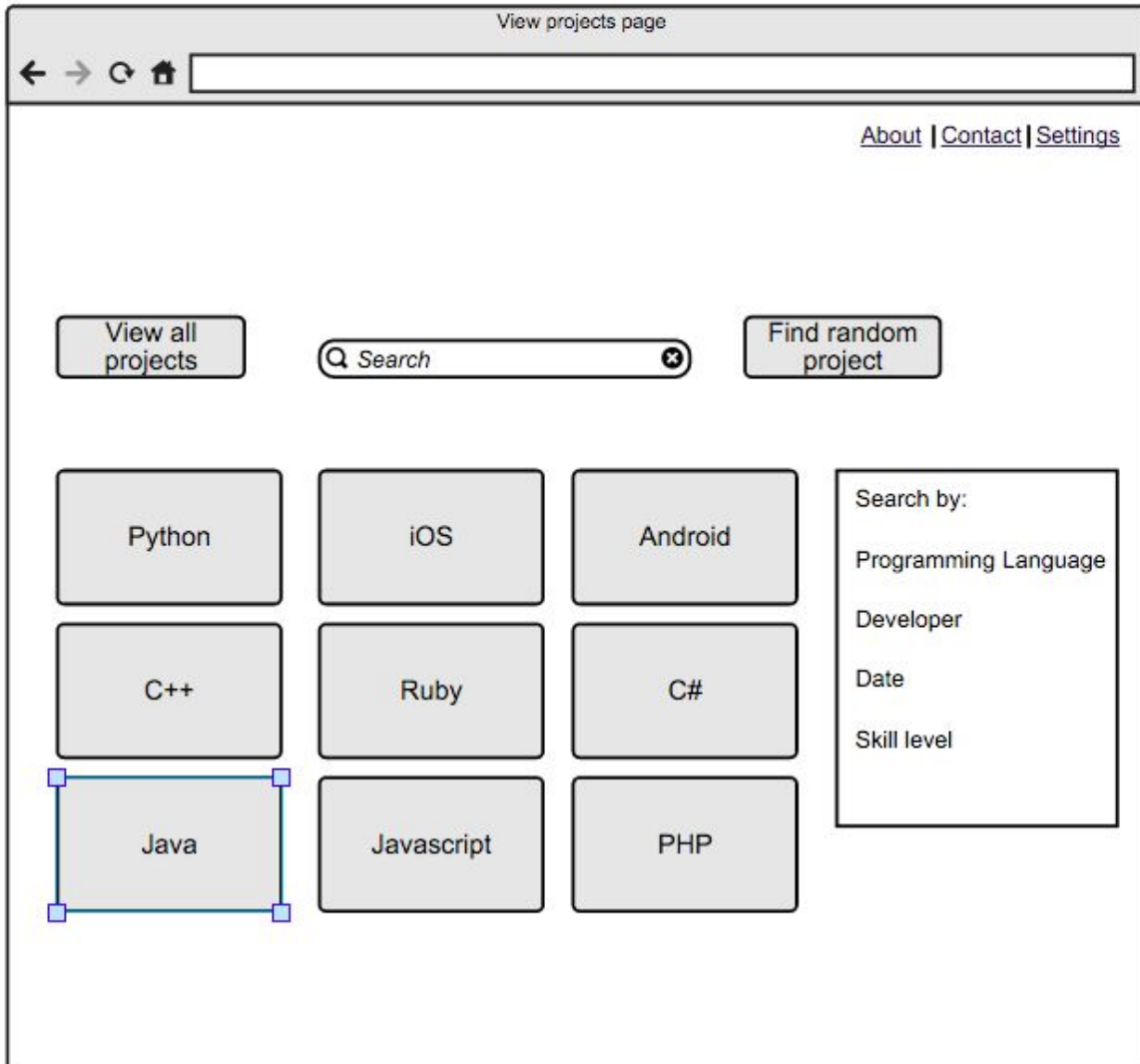


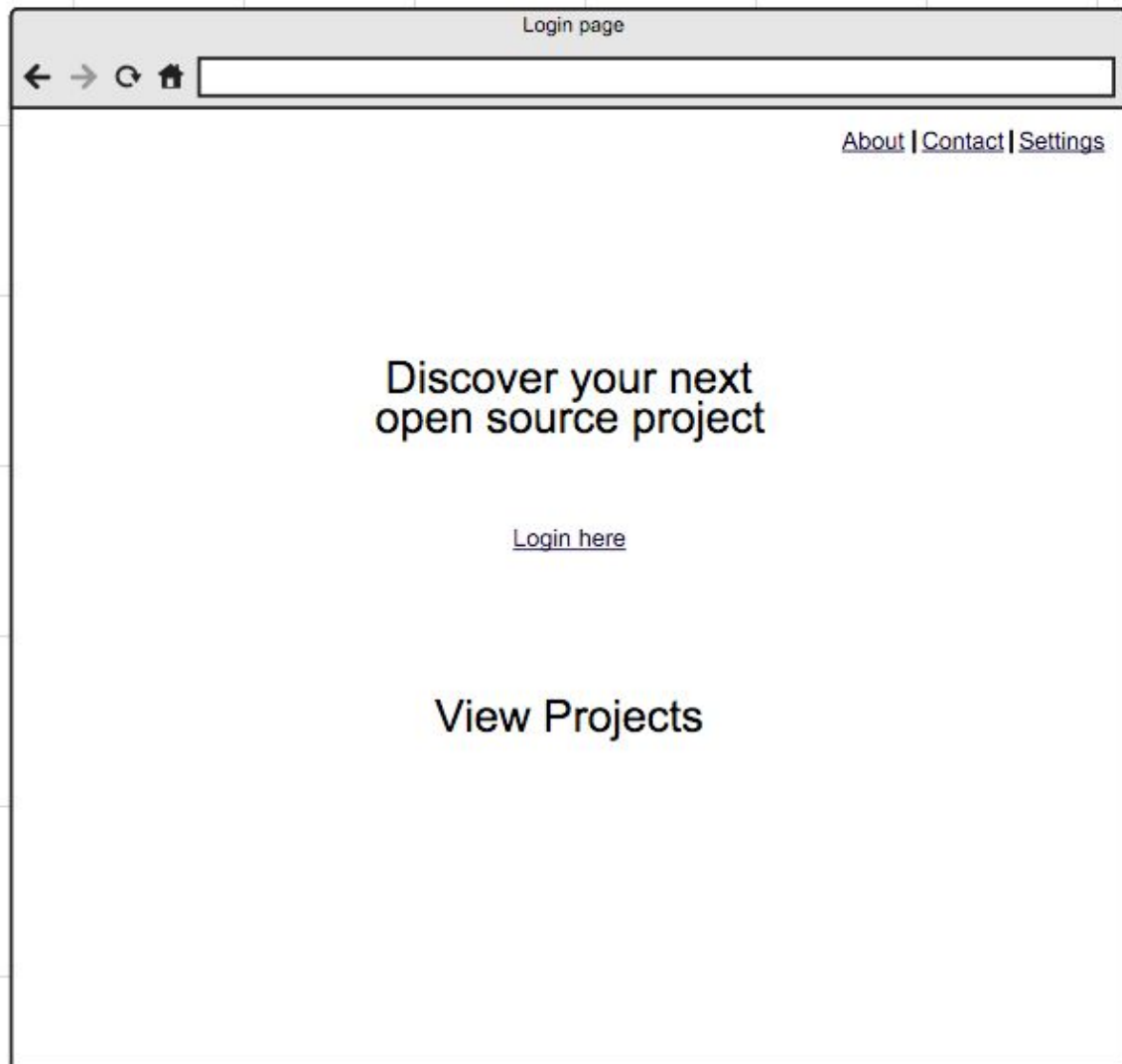




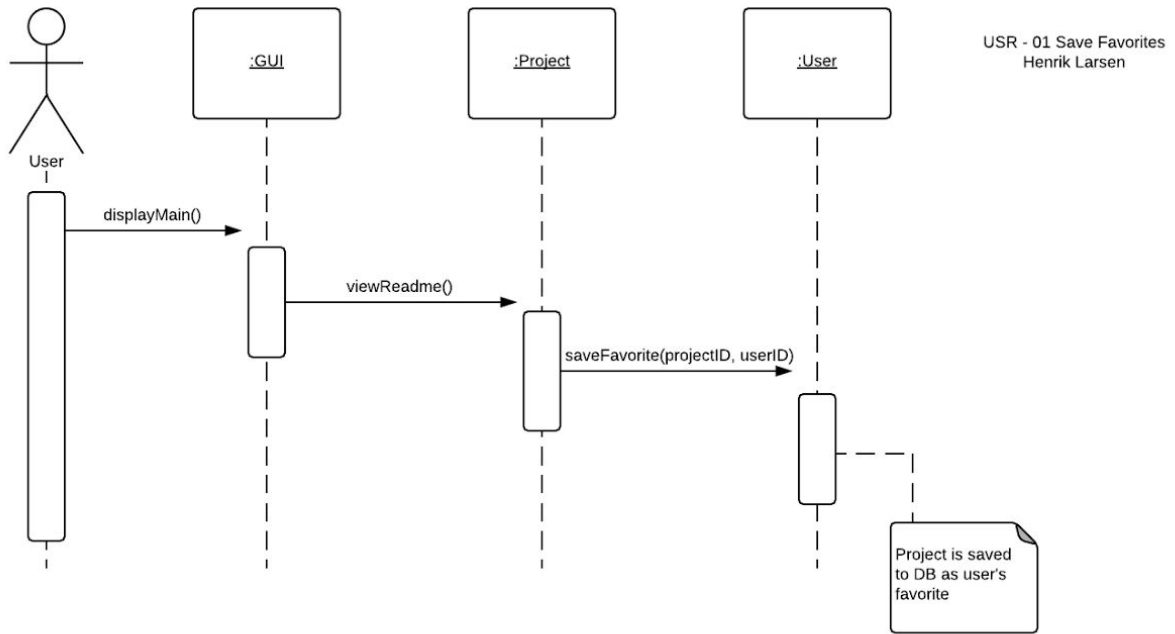
UI Mockups:



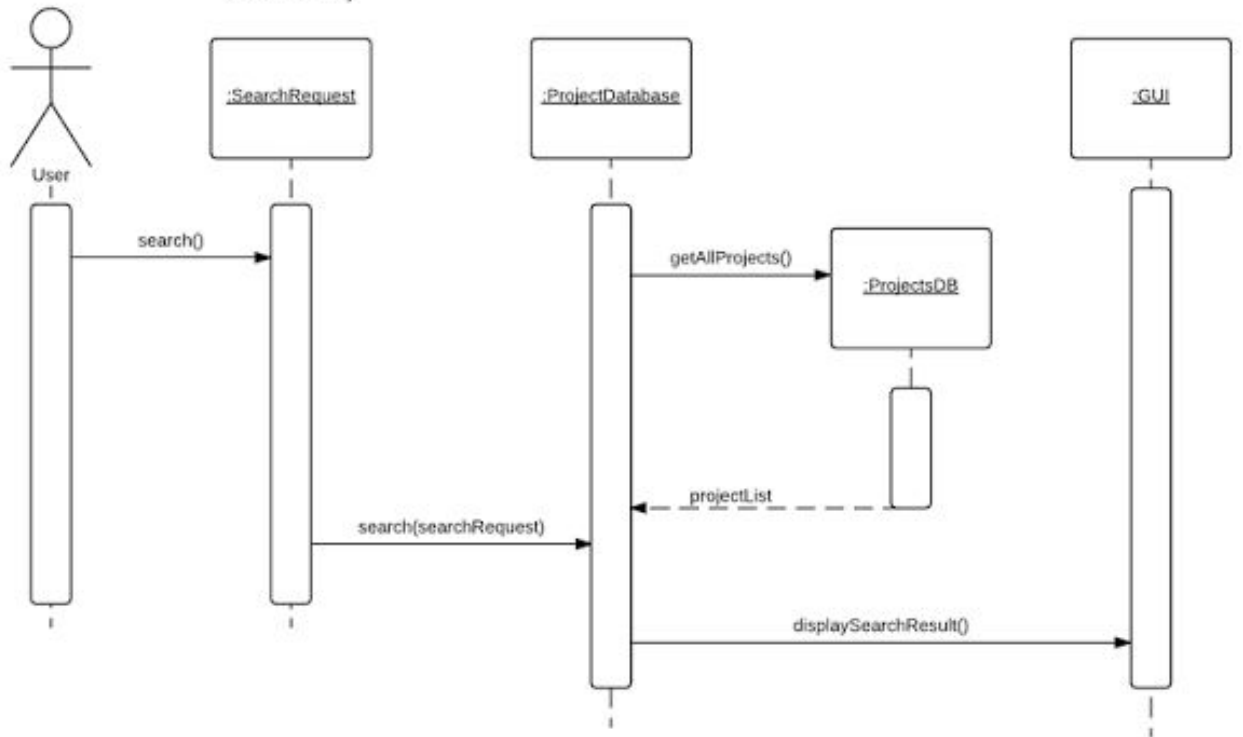




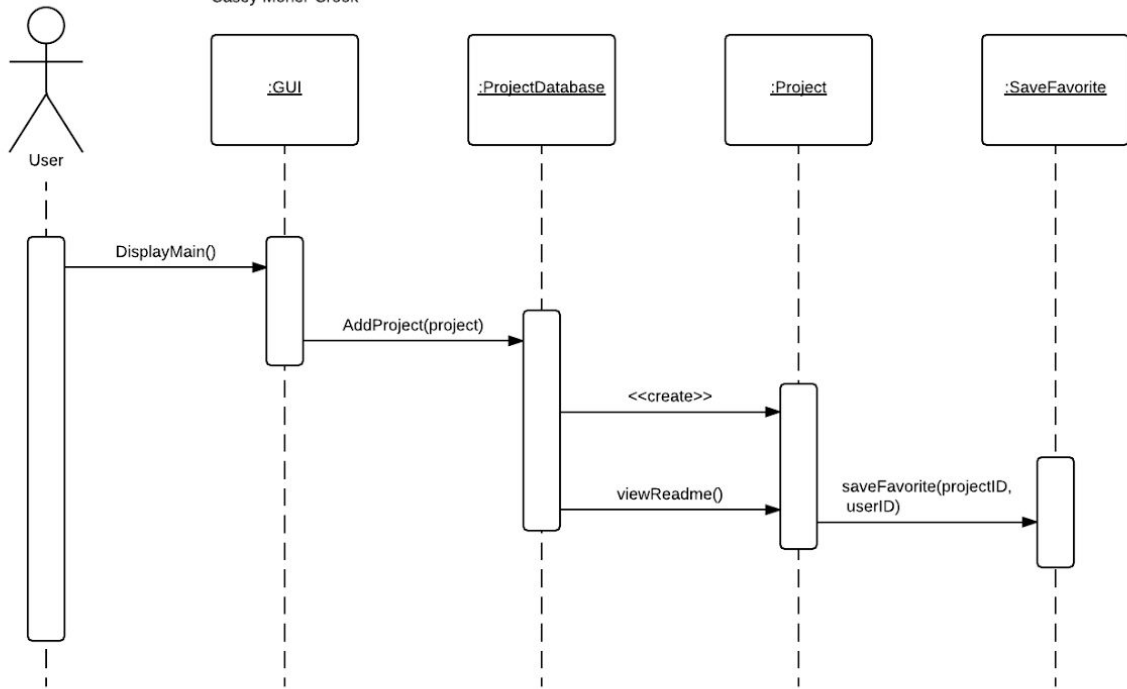
User Interactions:



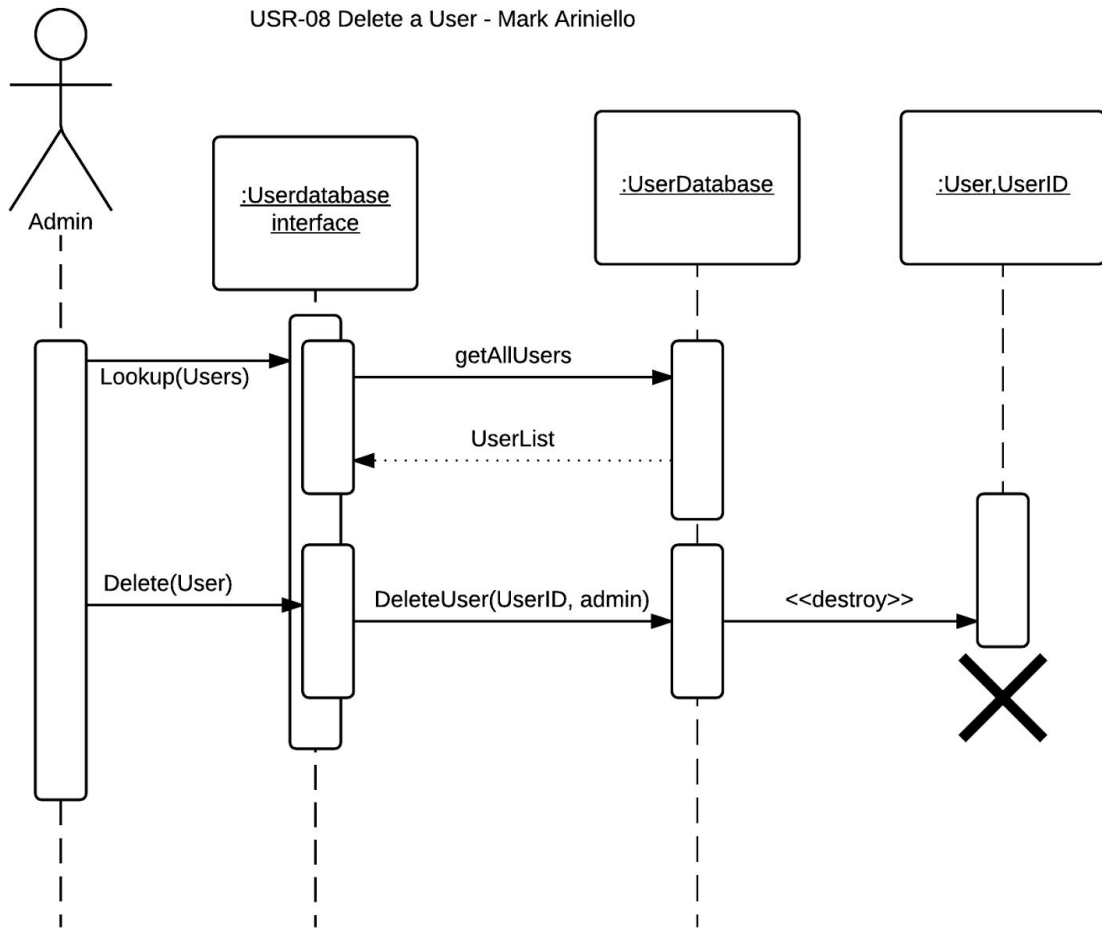
USR - 02 Search Projects
Sean Hansberry



USR - 03 Submit Project to Favorites
Casey Moher-Crook



USR-08 Delete a User - Mark Ariniello



Class Diagram:

