

1. **Name:** Richard Li
2. **Title:** Redium
3. **Project Summary:** Blogging service website with where users may contribute to a public forum, write private/public blogs, comment on and rate others blog posts.

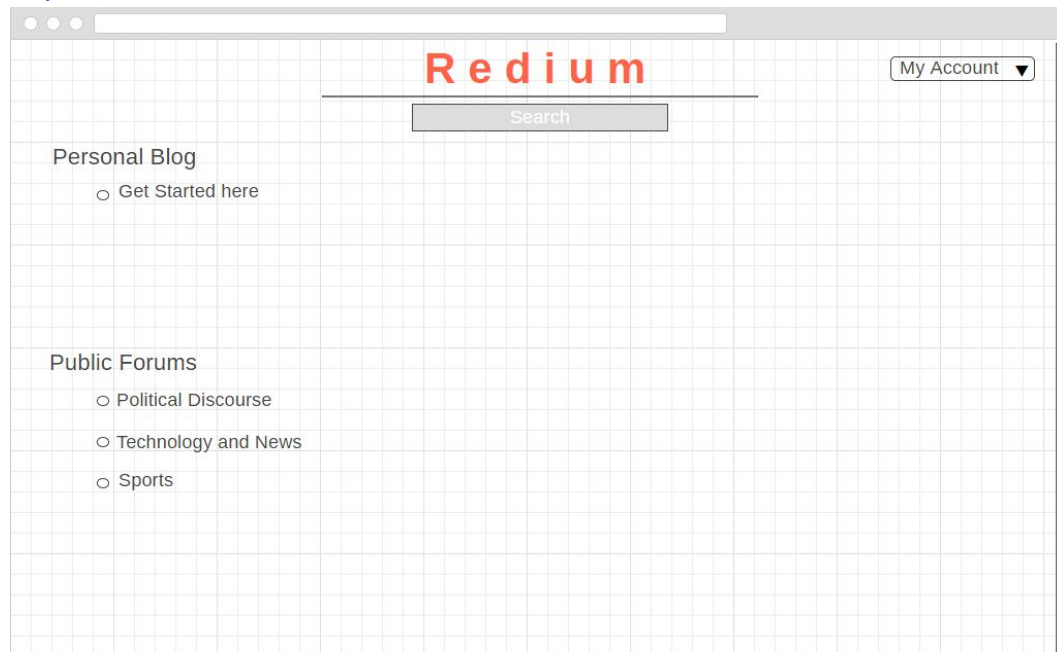
-
4. **Project Requirements:** Based on the project summary, what are the user requirements and responsibilities for your system? List the requirements and their associated responsibilities in a table format. Be sure to break them down into small manageable separate requirements and label each one with a unique ID number.
You can write the requirements in short sentences or in the Agile format.

Req ID	Requirement Description
UR-1A	Blog users may sign up for an account
UR-1B	Users may log in to the system
UR-2	Users may view their account activity amongst all public/private forums or blogs
UR-3A	Users may subscribe to/follow any public forum or blogger
UR-3B	Users may unsubscribe from/unfollow any public forum or blogger
UR-4A	Public and private forums are globally searchable for keywords
UR-4B	Private forums and blogs will only appear in searches if users have read permissions for those particular forums or blogs
UR-5	Users may contribute to (submit text comments) any forum or blog they have access to
UR-6	Users may star forum and blog posts as a voting system to promote more popular content
UR-7	Users may search for other users
UR-8	Users with the most starred content may be identified in a global leaderboard of the most prominent users

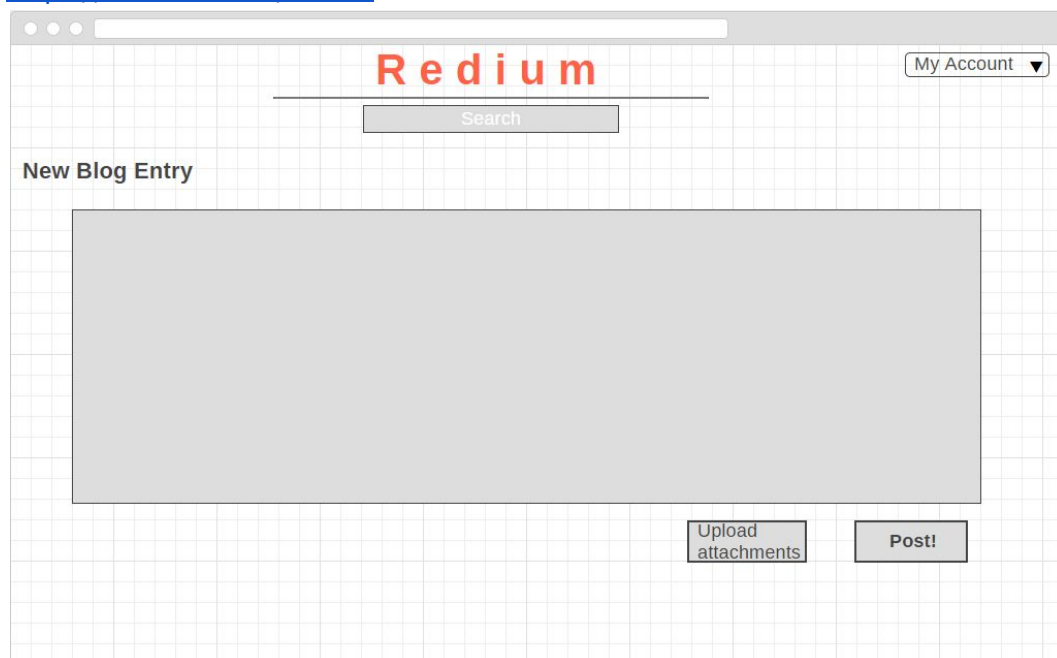
-
5. **UI Mockups:** Create screen mockups for the user interface of various parts of your application.
 - a. What will a user see as they work through the tasks identified in your use cases?
 - b. What is the overall organization of your user interface?
 - c. How will data be displayed?
 - d. How will the user navigate from screen to screen?

Note: it is okay to work on paper for this task and then scan in your work to include in your document. There are also free wire-framing tools online.

<https://wireframe.cc/nxOKNi>



<https://wireframe.cc/lLcn7o>



-
6. **Class Diagram:** Create a class diagram containing: what relationships the classes have, their attributes and (public) methods, what design patterns you may already know about are present in your design, etc. Be sure to show the visibility modifiers and relationships between the classes.

- Focus your class diagram on the classes you need to create.
For classes from Java or another library: You can specify an inheritance or composition relationship if it helps but do not put all the attributes/behaviors of the framework classes into the diagram.
For example,
If I was using Java Swing I could simply specify “submitBtn : JButton” as an attribute in one of my classes without drawing the JButton class. If I want to make use of Java’s Observer interface, I may draw the Observer interface in the class diagram to show a class implements it.
- Show all classes you are coding, including attributes, methods, visibility modifiers, relationships to other classes.

