Software Requirements Specification (SRS) Document

Project name: Reddit for Research (Project 19)

Team number: Team 41

Team members: Karsh, Manvith, Mrinal, Shanmukh

Brief problem statement

An important aspect of collaboration is to read and discuss published literature. Tools that are currently in the market force researchers to do this in isolation. 'Reddit for Research' aims to solve this, by providing a platform where users can upload, read, discuss, and debate literature/research in a seamless manner.

System requirements

The system requirements for our platform (Reddit for Research) are similar to those of a web browser's, as a web browser is all that is required to operate it. Therefore, the system requirements are as follows:

- (Windows)   Windows 7, Windows 8, Windows 8.1, Windows 10 or later
- (macoS) Yosemite 10.10 or later
- (Linux) 64-bit Ubuntu 14.04+, Debian 8+, openSUSE 13.3+, or Fedora Linux 24+
- An Intel Pentium 4 processor or later that's SSE2 capable
- 2GB of RAM

Any other requirements not mentioned can be found at:

- [Mozilla Firefox system requirements](#)
- [Google Chrome system requirements](#)
- [Internet Explorer system requirements](#)

Users profile

Users/members:

- Person that has registered, and is a member of the platform
- Users/members will be able to post, annotate/comment on posts, and reply to other annotations/comments
- As the platform is for research collaboration, users will primarily be researchers/people interested in different domains
- Users should be familiar with using computers/social media, and are expected to be able to follow the easy-to-use UI

Moderators:

- Moderators are users that are granted moderator role by developers
- Moderators are expected to remove irrelevant/inappropriate comments
- Moderators will be able to remove comments/annotations/posts
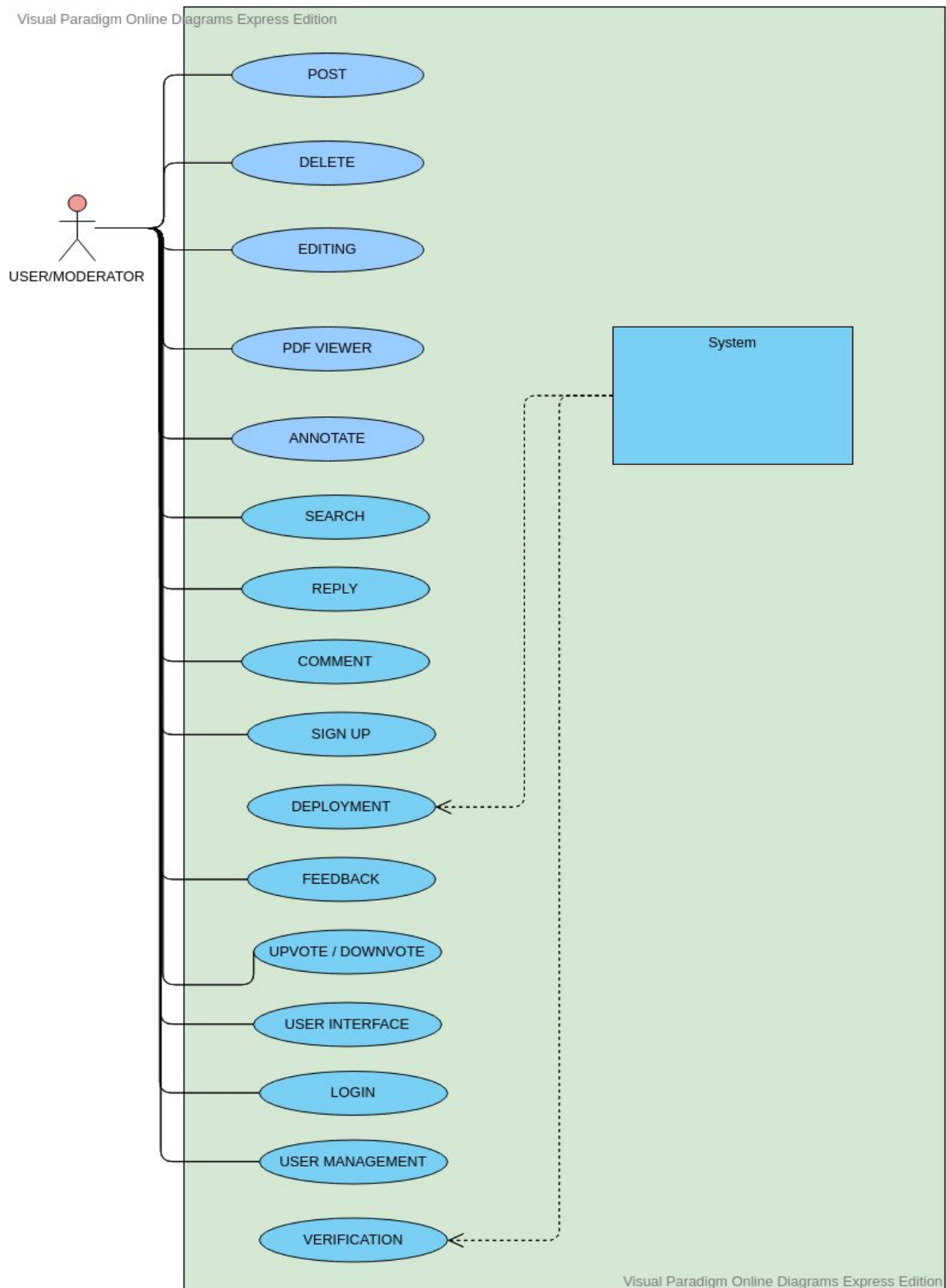
System Administrators/developers:

- System administrators (primarily the developers) are responsible for maintaining the platform
- They can contribute minimally to the platform itself by uploading/commenting
- Their role, however, is to modify system configuration and make appropriate updates

Feature requirements (described using use cases)

No.	User Case Name	Description	Release
1.	User Management	User account (and profile) management (creation, deletion, and modification) and user authentication during sessions. Users will be classified into groups (based on the roles described above) and their permissions and functionalities will be based on their roles.	R2
2.	Posting	Users will have the ability to post articles/PDFs/research papers open for discussion.	R1
3.	Annotation	Users will have the ability to highlight particular subsections of the article and annotate them. This feature can be used on all articles even if the user did not post them.	R1
4.	Search	The Search functionality will involve a search bar in which the user will search through annotations. Most relevant results will be returned.	R1
5.	PDF Viewer	Users will have the ability to view articles and links in the form of an inbuilt PDF viewer.	R1
6.	Commenting	Users will also be able to comment on a post to add his/her viewpoint without the need to annotate/highlight anything in the document.	R1
7.	Reply	The reply functionality allows users to reply specific annotations/comments. This will lead to more focused engagement.	R2
8.	Deleting	Users have the ability to delete Posts/Comments/ Annotations/Replies that they have posted. Moderators have permissions to delete other users posts.	R2

9.	Editing	Users have the ability to edit Comments/ Annotations/Replies that they have posted. An "Edited" tag will show on edited posts.	R2
10.	User Interface	The users will be accessing the WebApp on a web browser, for which a responsive user interface needs to be designed. Templates will be used to display the various information (fathered from backend) and provide functionalities	R2
11.	Deployment	For the product to be delivered to the users, the Web App will be deployed on AWS servers. Product scaling, security, and database management will have to be managed on the cloud servers.	R1 & R2
12.	Feedback	The feedback functionality allows users to input feedback back to the developers. This can be bug/change requests or feature requests.	R2
13.	Night Mode	Night mode functionality will allow the user to switch between a light themed webapp and a dark themed webapp as per convenience.	R2
14.	Profile	User profile will contain details about the user like date of joining, his annotations and comments, etc.	R2
15.	Upvote/Dow nvote	The Upvote/Downvote functionality will allow users to engage with the posts. This will increase user engagement and increase user retention.	R2

Use case diagram

USER/MODERATOR

POST

DELETE

EDITING

PDF VIEWER

ANNOTATE

SEARCH

REPLY

COMMENT

SIGN UP

DEPLOYMENT

FEEDBACK

UPVOTE / DOWNVOTE

USER INTERFACE

LOGIN

USER MANAGEMENT

VERIFICATION

System

Use case description

Use Case Number:	UC-1
Use Case Name:	User Management
Overview:	User account (and profile) management (creation, deletion, and modification) and user authentication during sessions. Users will be classified into groups (based on the roles described above) and their permissions and functionalities will be based on their roles.
Actors:	All Users
Pre condition:	A user trying to log-in
Flow:	1. A user enters his/her credential correctly 2. User is logged in and authenticated for session
Alternate Flow:	If User fails to login 1. A user enters his/her credential incorrectly 2. An error message pops up.
Post Condition:	User details are fetched from the database and based on the group, the user is redirected to the respective dashboard with functionalities based on individual permissions.

Use Case Number:	UC-2
Use Case Name:	Posting
Overview:	Users will have the ability to post articles/PDFs/research papers open for discussion.
Actors:	User logged in and authenticated
Pre condition:	A user trying to post
Flow:	1. A user enters his message about the post 2. User attaches link/pdf of research paper/article. 3. User clicks on the post button
Post Condition:	The post is posted onto the webpage. In the backend, the post and the user who posted it are appended to the database.

Use Case Number:	UC-3
Use Case Name:	Annotation
Overview:	Users will have the ability to highlight particular subsections of the article and annotate them. This feature can be used on all articles even if the user did not post them.
Actors:	User logged in and authenticated
Pre condition:	A user trying to Annotate
Flow:	1. User Highlights certain section of the text. 2. User can annotate this part of the section by using hypothes.is. 3. The annotation is added to the post.
Post Condition:	The annotation is updated to the webapp. The annotation is also added to the database to aid the search engine.

Use Case Number:	UC-4
Use Case Name:	Search
Overview:	The Search functionality will involve a search bar in which the user will search through annotations. Most relevant results will be returned.
Actors:	All Users
Pre condition:	A user trying to search
Flow:	1. A user enters his/her search query in the search bar. 2. The search bar uses elastic search to go through the various annotations in the database. 3. Most relevant results are returned.
Post Condition:	Relevant Results are fetched and displayed.

Use Case Number:	UC-5
Use Case Name:	PDF Viewer
Overview:	Users will have the ability to view articles and links in the form of an inbuilt PDF viewer.
Actors:	User logged in and authenticated
Pre condition:	A user trying to view a document
Flow:	1. User clicks on the file linked in the post. 2. An inbuilt PDF viewer pops up with the file.
Post Condition:	The PDF file is now ready to view.

Use Case Number:	UC-6
Use Case Name:	Commenting
Overview:	Users will also be able to comment on a post to add his/her viewpoint without the need to annotate/highlight anything in the document.
Actors:	User logged in and authenticated
Pre condition:	A user trying to comment
Flow:	1. A user enters his/her comment in the input box. 2. User clicks on the comment button.
Post Condition:	The comment is displayed on the post. In the backend, the comment is added to the database(If required).

Use Case Number:	UC-7
Use Case Name:	Reply
Overview:	The reply functionality allows users to reply specific annotations/comments. This will lead to more focused engagement.
Actors:	User logged in and authenticated
Pre condition:	A user trying to reply
Flow:	1. A user clicks on the reply button underneath a specific comment/annotation. 2. User adds his reply on the input box. 3. User clicks on send button.
Post Condition:	The reply is displayed on the post. In the backend, the comment is added to the database(If required).

Use Case Number:	UC-8
Use Case Name:	Deleting
Overview:	Users have the ability to delete Posts/Comments/ Annotations/Replies that they have posted. Moderators have permissions to delete other users posts.
Actors:	Users who posted (Some special deletions only applicable to moderators)
Pre condition:	A user trying to delete a post/comment/reply/annotation
Flow:	1. A user clicks on the delete button underneath a specific comment/annotation. 2. User confirms his deletion
Post Condition:	The comment/annotation/reply/post is deleted from the webapp as well as the database.

Use Case Number:	UC-9

Use Case Name:	Editing
Overview:	Users have the ability to edit Comments/ Annotations/Replies that they have posted. An "Edited" tag will show on edited posts.
Actors:	Users who posted
Pre condition:	A user trying to delete a post/comment/reply/annotation
Flow:	1. A user clicks on the edit button underneath a specific comment/annotation. 2. User adds his new edited input to the input box. 3. User confirms his edit.
Post Condition:	The comment/annotation/reply/post is deleted on the webapp as well as the database.

Use Case Number:	UC-10
Use Case Name:	User Interface
Overview:	The users will be accessing the WebApp on a web browser, for which a responsive user interface needs to be designed. Templates will be used to display the various information (fathered from backend) and provide functionalities
Actors:	All Users
Pre condition:	No precondition
Flow:	1. The user is given access to functionalities based on his/her roles and permissions. 2. The user-interface follows a template which is simple, interactive, responsive and easy-to-use 3. The User can use the web app interface to access the different functions defined in user roles.
Post Condition:	The frontend user interface uses the backend to correctly perform the desired task

Use Case Number:	UC-11
Use Case Name:	Deployment
Overview:	For the product to be delivered to the users, the Web App will be deployed on AWS servers. Product scaling, security, and database management will have to be managed on the cloud servers.
Actors:	All users
Pre condition:	Internet access
Flow:	1. The user enters the URL to access the web APP
Post Condition:	User is redirected to the web APP which is deployed on cloud servers

Use Case Number:	UC-12
Use Case Name:	Upvote/Downvote
Overview:	The Upvote/Downvote functionality will allow users to engage with the posts. This will increase user engagement and increase user retention.
Actors:	User logged in and authenticated
Pre condition:	A user trying to upvote/downvote
Flow:	1. User goes to a particular post and clicks upvote/downvote.
Post Condition:	The post gets an upvote/downvote. This is updated in the database.

Use Case Number:	UC-13
Use Case Name:	Feedback
Overview:	The feedback functionality allows users to input feedback back to the developers. This can be bug/change requests or feature requests.
Actors:	All Users
Pre condition:	A user trying to give feedback
Flow:	1. A user clicks on the feedback button 2. He/Her is taken to the feedback page where he can type in his feedback. 3. This feedback is sent to a particular email.
Post Condition:	Feedback is sent to the development team.

Use Case Number:	UC-14
Use Case Name:	Night Mode
Overview:	Night mode functionality will allow the user to switch between a light themed webapp and a dark themed webapp as per convenience.
Actors:	All Users
Pre condition:	A user trying to toggle mode.
Flow:	3. User clicks on the Night mode button 4. User is logged in and authenticated for session
Post Condition:	Webapp switches to dark mode.

Use Case Number:	UC-15
Use Case Name:	Profile

Overview:	User profile will contain details about the user like date of joining, his annotations and comments, etc.
Actors:	All Users (admin, moderators, CDC head and owners, parents and therapists)
Pre condition:	User logged in and authenticated
Flow:	1. User clicks on profile tab.
Post Condition:	User details are fetched from the database and displayed in a neat manner in the profile page. His annotations/comments/activity can also be displayed.