As scientific researches have become a lot more collaborative than before due to advancements in tele-communication technology, we want to create a social network for scientists and researchers to facilitate scientific communication. The program achieves the following main objectives:

Each member will have a personal profile page.

Each member can publish posts.

Each member can follow other members.

Each member can comment on others’ posts.

Each member can search for other members and their posts.

The system architecture consists of a frontend server and a backend server. On the frontend side, we used Play! restful server, HTML, CSS, Scala, Ajax, JQuery, and Bootstrap. On the backend side, we used Play! restful server, JPA, and MySQL database.

On the frontend, the following list of functionalities are implemented:

A user has a personal home page.

A user can sign up as a new user or login in as a return user.

Personalized main page has user profile and posts from themselves or people followed by them.

User can edit their own personal data in the front end and save it in the backend.

User can search post using a search bar at the top of the page.

User can search user by using a search page with search options including last name, first name, and research fields.

User can see all of his/her followers.

Personal home page displays posts sorted by timestamp in descending order.

User can post, delete, like and edit a post.

User can set privacy of their post as public or private.

User can comment on a post.

User can follow/unfollow a researcher.

User can share a post.

User get autocomplete search when searching by post.

User can only modify his/her own data and is prevented from modifying others’ data.

On the backend side, the following list of APIs are implemented:

API for posting message.

API for retrieving posts with comments.

API for updating a post.

API for deleting a post.

API for commenting on a post.

API for retrieving followers.

API for retrieving users being followed.

API for following/unfollowing a user.

API for setting security of a post.

API for retrieving user’s location.

In addition to the above basic functionalities, we implemented several innovative features:

REST API with authentication.

"People you may follow" with Spark.

Search with Lucene.