# FilmedIn

## Team 8 - Sprint 1 Retrospective Document

Shreya Shankar, Bhavana Kakubal, Parth Kenkere, Prashast Vaidya, Manu Aggarwal

# Table of Contents

## What went well?

1. We were able to set up and successfully operate git while maintaining an efficient workflow.
2. Our repository is now set to reflect the development process of each team member and reflect a working prototype of our product at all times in the master branch.
3. We were able to successfully create a working backend and frontend model for our website. Each portion is completely functional and contains majority of the planned features of the website.

Following are the tasks we accomplished:

- Learnt and understood the MEAN stack for our product
    - Much of the sprint 1 was used to learn and understand these new concepts. As all of our group was new to MEAN stack and MongoDB and AngularJS, etc, we spent the majority of our time reading and watching tutorials in order to start developing our web app.

- Acquired the required knowledge of MongoDB to store our data
    - We found several links that explained the installation and use of MongoDB as well as the process of how to create a web app using MEAN stack.

- Learnt express.js and node.js for the back end development
    - We learnt these web application frameworks in order to use and apply them to our web app

- Learnt HTML and CSS for the frontend development
    - Most of the learning process was through trial and error when we created the necessary html pages for our site. We also relied heavily on bootstrap for our frontend.

- Learnt and understood git flow and how to integrate git with github
    - We understood the core features of git such as branching and merging and created a specific workflow which involves a master branch, each member's individual branch and a product branch used to merge the individual member branches for bug fixes before merging it with the master branch.

- Successfully created a working landing page for the website
    - Using HTML and CSS, we were able to create the landing page for our website. One major challenge with this was the styling and responsiveness of the pages; It was extremely time consuming

- Successfully implemented a login procedure
    - We were able to have a user signup with an email and password, enter information, and later log back into their previously made account.

- Successfully stored a user's information in the database
    - The user's information included their first name, last name, email, password, gender, age, and ethnicity.

- Successfully encrypted a user's password for safety
    - We wrote a logic that encrypted as well as decrypted the user's password.

- Wrote a routes.js file to generate API endpoints for our webpages
    - This file contains all the API calls to execute specific functions on specific web pages.

- Wrote a server.js file to run our server on a local host
    - Creates a local server and runs alongside our mongoDB database.

## What did not go well?

We underestimated the time it would take to learn the required languages and procedures. We were all inexperienced with most of the required material and had to learn everything from scratch. We did not factor the time this would take while planning and calculating the time we allotted for the tasks during the first sprint. The excess time taken cascaded towards the end and resulted in a few unfinished tasks for the sprint. Also, one of our main problems was the unfinished bridge between the frontend and backend. Although the frontend and backend were separately able to complete for most of the tasks, we failed to connect the two. For the next sprint, we plan to complete the unfinished as well as new tasks as well as accomplish the connection between the frontend and backend.

Following are the user stories we partially accomplished:

- As a user, I would like to login to the website.
- As a user, I would like to reset my password (if forgotten) by sending an email to the user.

For the above two stories, we were not able to connect the backend API to render frontend. The major reason for falling short at these tasks for the backend team was that it initially took a considerable amount of time (more than what we had thought) learning/ setting up mongoDB and NodeJS and connecting both of them.

- As a user, I would like the option to edit my profile.
- As a job seeker, I would like to search through all the jobs/opportunities posted.
- As a user, I would like to search through all the available profiles based on age, gender, name and other criterias

For the above three user stories, we were not able to implement the backend side of the architecture. The major reason for this failure was that the backend team already had a backlog of tasks from the first week (making the login API and setting up the mongoDB) because of which everything got pushed by a week on the backend side.

## How we should improve?

For FilmedIn to be completed on schedule we need to individually and as a team complete all our assigned user stories and their sub tasks for the next sprint and catch up on our backlog from the previous sprint. We have arrived at the following resolutions to achieve the above mentioned goals:

- A concern we need to address would be the fact that the team did not know what was happening with regards to the other team members progress in their sub tasks. This can be remedied by the team meeting twice every week, once with our project coordinator, Tori Shurman, and once with just the team members.

- Another hurdle we faced was the miscommunication between the backend team where, on more than one occasion our team has worked on the same files concurrently implementing similar functions. We plan on starting a thread on Slack for each team between all the members of that particular team, on which, we will claim what we plan to do every day and and what we achieved at the end of the day.

- During this sprint we faced a lot of problems setting up the environment. This led us to fall behind schedule and thus resulted in us not completing all the assigned user stories. This was further augmented by the number of user stories we assigned and the number of days we had to implement them in. This could’ve been avoided by providing a well thought out sprint planning document with realistic goals