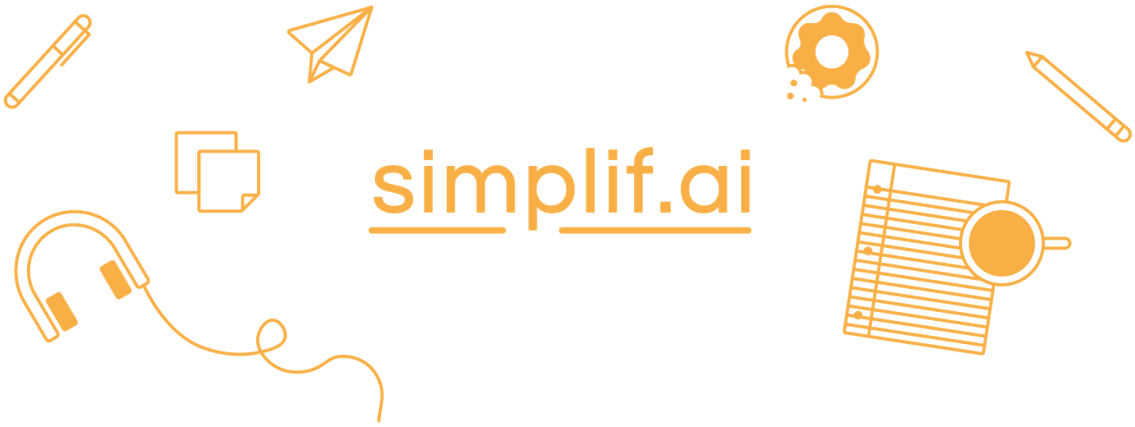

CS 307 Team 3 - Sprint 1 Retrospective

Sawyer Blatz, Ian Renfro, Kevin Xia, Lena Adel, Audrey Vincent, Pretoria (Ria) Chang

1. **What went well?**

Overall as a team, we were very successful with functionality coming together. Our NLP algorithm was extremely functional and we overcame many obstacles and challenges along the way. We were able to improve upon our communication methods and came into more understanding of how the different pieces of our project fit together-- both are lessons we will be fully utilizing moving forward.

User Story #2

As a user, I would like to have functionality for logging into, editing, signing out of, and deleting my account. (User Story #3)

#	Task Description	Estimated Time	Owner	Team
1	Create the signup page	2 hours	Audrey	Frontend
2	Implement ability to log in and maintain a session on web client with cookie	4 hours	Audrey	Frontend
2	Style Sign Up Page	2 hours	Ria	Frontend
3	Implement login page	2 hours	Ria	Frontend

4	Style Login Page	2 hours	Ria	Frontend
5	Implement Logout Navigation flow, updating session cookie	2 hour	Audrey	Frontend
6	Create endpoint for Logout	1 hour	Sawyer	Backend
7	Create and style Profile Page for User	2 hours	Audrey	Frontend
8	Create endpoint for viewing user profile	1 hour	Sawyer	Backend
9	Create endpoint for editing the user	1 hour	Sawyer	Backend
10	Create endpoint for deleting an account	2 hours	Sawyer	Backend
11	Style Profile Page	2 hours	Ria	Frontend
12	Ensure responsive design for both mobile and desktop	1 hour	Ria	Frontend
13	Test the endpoints for getting and editing users	2 hours	Sawyer	Backend
14	Create documentation for user endpoints	1 hour	Sawyer	Backend
15	Test proper functionality for user account functions	1 hour	Ria	Frontend

User Story 2 was successful. However, we should have abstracted tasks into multiple user stories. More properly delegating tasks with a more efficient structure would have made it easier to test our code; it would also have given us a better idea of our workloads-- we did not appropriately estimate the time taken for each task.

User Story #3

As a user, I would like to be able to reset my password if I forget it. (User Story #2)

#	Task Description	Estimated Time	Owner	
1	Implement forget password and reset flow on the webapp	4 hours	Ria	Frontend
2	Set up a mail client	4 hours	Lena	Backend
3	Implement password resetting through the use of the mail client	3 hours	Sawyer	Backend
4	Create an endpoint for forget password	2 hours	Sawyer	Backen
5	Setup change password	2 hours	Sawyer	Backend
6	Create documentation for the mail client and password management	1 hour	Sawyer	Backend
7	Test password management rigorously (resetting, creation, and modifying) to make sure it is secure and bug-free	2 hours	Ria	Backend / Frontend

User Story 3 was successful, as our team members worked effectively and efficiently when it was necessary, and when there were communication issues, we were able to work through them. The endpoints and client are functional.

User Story #4

As a user, I would like to summarize information from plain text. (User Story #5)

#	Task Description	Estimated Time	Owner	Team
1	Implement the organization structure for classes as outlined in our design document	4 hours	Lena	Backend
2	Transform the text to the needed JSON form	3 hours	Lena	Backend

3	Send out the JSON data to NLP summarizer	4 hours	Lena	Backend
4	Receive JSON data from the NLP summarizer	4 hours	Lena	Backend
5	Test the transformation of text to needed JSON form	1 hour	Lena	Backend
6	Test the JSON data to NLP summarizer	1 hour	Lena	Backend
7	Test JSON received from NLP	1 hour	Lena	Backend
8	Create documentation for Text summarizer using NLP summarizer	1 hour	Lena	Backend
9	Implement receiving the response and app routing correctly	2 hours	Audrey	Frontend

User Story 4 was successful, as we strongly pushed communication between the team members involved, especially given the many different relevant parts required.

User Story #5

As a user, I would like to generate a list of condensed information from text. (User Story #10)

#	Task Description	Estimated Time	Owner
1	Build basic class structure for overall summarization module.	2 hour	Kevin
2	Use NLTK to extract nouns from text.	1 hour	Kevin
3	Implement LexWord class.	1 hour	Kevin
4	Implement LexChain class.	3 hours	Kevin

5	Implement LexChainGroup class.	3 hours	Kevin
6	Write code to build basic chains from nouns.	6 hours	Kevin
7	Use lexical chains to rank relevance of sentences (use basic ranking system, improve later).	6 hours	Kevin
8	Add unit tests and documentation for newly added summarization module functions.	8 hours	Kevin

User Story 5 was successful, as the team member who was assigned to build the text summarizer was able to consistently iterate upon his code throughout the sprint, communicated effectively, and has a good idea of what work still needs to be done. Basic functionality was achieved, as planned, and has full compatibility with all the other necessary pieces of our web app.

User Story #6

As a user, I would like to have a place to input text. (User Story #4)

#	Task Description	Estimated Time	Owner	Team
1	Create Home page	3 hours	Ria	Frontend
2	Create and style an interactive text box that toggles on grouped sentences	2 hours	Audrey	Frontend
3	Animation while summary processes	5 hours	Ria	Frontend
4	Routing application based on JWT authentication	3 hours	Audrey	Frontend
5	Navigation Components	2 hours	Audrey	Frontend

6	Navigation Bar Styling	3 hours	Ria	Frontend
7	Test home page, text box, and navigation UI.	2 hours	Ria	Frontend
8	Ensure responsive design for components	1 hour	Ria	Frontend

User Story 6 was successful, and most all kinks in communication were worked out. We were able to achieve the foundation in styling that we had allotted for ourselves.

User Story #7/#8

As a user, I would like to modify the text of sentences/bullets from my generated summary. (User Story #13)

As a user, I would like to use a text editor to finalize the summary of my text. (User Story #15)

#	Task Description	Estimated Time	Owner	Team
1	Parse inputted text into an array of sentences, allowing you to edit one sentence at a time	4 hours	Audrey	Frontend
2	Create a simple text editor page that follows our UI mockups	2 hours	Audrey	Frontend
3	Test the text editor	1 hour	Audrey	Frontend
4	Ensure text editor incorporates responsive design	1 hour	Audrey	Frontend

User Story 7 and 8 were successful. Our designs and implementation matched mockups including a responsive design to easily read the summary.

User Story #9

As a user, I want the web application to have an easy-to-read font. (User Story #26)

#	Task Description	Estimated Time	Owner	Team
1	Select a consistent font to use throughout our web app.	1 hour	Audrey	Design
2	Compare the weights of our different typefaces to determine the font for heading, subtitle, and body.	2 hours	Audrey	Design
3.	Get feedback from people on the font, interface, and layout to see if we should change it	1 hour	Ria	Design

User Story 9 was successful, as not only does our text serve to improve upon the aesthetic of our web app, but also improves readability.

User Story #11

As a developer, I would like to have an API for our service. (User Story #32)

#	Task Description	Estimated Time	Owner	Team
1	Create a flask application and research on Dreamhost	5 hours	Ian	API
2	Create a landing page for redirection to the webapp	4 hours	Ian	API
3	Create route for requesting a summarization	2 hours	Ian	API
4	Link the NLP package to the route for the API	4 hours	Ian	API
5	Test landing page redirect	2 hours	Ian	Testing

6	Test pinging the API	2 hours	Ian	Testing
7	Test NLP connection with the API	4 hours	Ian	Testing
8	Smoke test the API for the NLP specific	3 hours	Ian	Testing

User Story 11 was successful, as the team member who was assigned to these tasks was able to utilize his past experiences to quickly implement all of his tasks, and communicated effectively with other team members when they had relevant questions.

User Story #12

As a developer I would like to have a database where I can store data about the user. (User Story #34)

#	Task Description	Estimated Time	Owner	Team
1	Set up RDS on Amazon Web Services	3 hours	Sawyer	Database
2	Hook up Database with Backend	3 hours	Lena	Database
3	Implement the organization structure for storing data in the database	3 hours	Lena	Database
4	Implement the organization structure for classes as outlined in our design document	4 hours	Sawyer	Database
5	Testing the backend hookup	1 hour	Lena	Testing
6	Testing the organization structure for storing data in the database	1 hour	Lena	Testing
7	Add documentation for this user case	2 hours	Sawyer	Database

User Story 12 was successful, team members were able to setup the database on Amazon web services. We were able to successfully create the tables and store the user's information and summaries in the database.

2. **What did not go well?**

Early on we realized how much time is spent coordinating between different aspects of the application. This extra time was not anticipated. We also did not anticipate time taken to debug. We will accommodate in the next sprint.

User Story #1

As a developer, I would like to have proper documentation, coding standards, and version control standards. (User Story #39)

#	Task Description	Time	Owner
1	Write code templates.	2 hours	Ria
2	Write Github pull request template.	1 hour	Ian
3	Write Github issue template.	1 hour	Ian
4	Write CONTRIBUTING.md.	2 hours	Ian

User Story 1 was somewhat successful. However, some members failed to adhere to our defined coding standards. In addition, when there was pressure to make deadlines, commits were made straight to develop instead of branching first, testing, then making a pull request. Another problem that arose under strict deadlines was the lack of code review on some pull requests. Some members failed to make pull requests when necessary, creating confusion to if the code was actually complete and working. Because we had close communication, we were successful in resolving these conflicts. However, this took more time away from development to resolve issues.

User Story #10

As a developer, I would like user login to be secure. (User Story #33)

#	Task Description	Estimated Time	Owner	Team
1	Hook our web app into the Google Authentication API	3 hours	Sawyer	Backend
2	Use the Google API to securely authenticate the user	2 hours	Sawyer	Backend
3	Test the authentication through Google API and document our usage of it for future use throughout the webapp	2 hours	Sawyer	Backend

User Story 10 was partially successful. As a team, we ran into some last minute issues on getting it to interact correctly with the frontend. On the frontend, we were able to allow the user a choice to connect his/her account with google drive. The request to Google's API was successful, but there is an additional authentication screen we needed to connect back to the front end, which is where the failure occurred last minute. In the next sprint we will ensure that this functionality is working in full.

3. How should you improve?

Overall our team was very successful completing the functionality we sought to implement. We have many ways to improve productivity as a team. This includes communicating better and delivering what is expected of each member on time. Over the course of the sprint, we discovered better ways to communicate and work together.

As a team, we all need to commit more time to the project earlier in the sprint. This was a direct result of failing to realize how much time it takes to coordinate between tasks and teammates. Also, meeting the goals for the week is very important in order to prevent blockers for other members. Next sprint we will focus more on creating realistic goals and executing on what we plan to achieve.