

COSC 4P02: Software Engineering II.

Group 9: Release Planning Document

Scrum Master: Jaden Kuhn (jk21pf@brocku.ca - 7249683)

Product Owner: Dea Kukuqani (dk16qs@brocku.ca – 6196018)

Developers: Shijie Tong (st20az@brocku.ca - 7081201)

Nicholas Caruso (yq20oz@brocku.ca - 7189749)

Thomas Semenak (ts19cp@brocku.ca - 6745038)

Dalton Morris (dm20bq@brocku.ca - 7053184)

Chidera Nwana (cn20rq@brocku.ca - 7078124)

Instructor: Naser Ezzati-Jivan

January 19th, 2025

Release Planning Document - Group 9

Introduction:

The problem we aim to solve is the lack of a streamlined, automated way for individuals and businesses to create personalized newsletters and social media posts tailored to specific preferences. Currently, users often spend hours manually collecting, curating, and formatting content to meet the needs of their audience. Our goal is to implement a LLM which uses aggregated sources to create summarized content for the user. This content is to be presented in a specified format (e.g. Twitter post, Instagram post, Facebook post, newsletter, etc.), or if the user wants, the aggregated sources themselves will be presented.

This document below explains the various user stories for the project and the associated tasks that go with them (Product Backlog). Our Sprint Backlog is at the bottom of this document, and it displays our goals for each sprint and when the sprints occur.

User Stories:

**Note: For priority, P1 is the highest, P5 is the lowest.*

ID	Title	Description	Acceptance Criteria	Priority(P1-P5)
Custom News Aggregation				
1	News Source, Topic, and Keyword Aggregation	As a user, I want to be able to input my favorite news sources, topics, keywords, locations, or industries and have the system gather relevant content so that I can quickly review content from my favorite sources.	<ul style="list-style-type: none"> - Users can input URLs, topics, keywords, locations, or industries representing their interests. - The system will locate and display relevant content directly from these URLs or based on these keywords. 	P1
2	Categorical Tagging of Sources	As a user, I want the system to tag sources with a category so that it can enhance the accuracy of the content displayed.	<ul style="list-style-type: none"> - The system categorizes sources into pre-defined categories to improve the accuracy of the displayed content. 	P4
3	Approved and Prohibited Sources	As a site administrator, I want to be able to manage the list of approved and prohibited sources so that news and content only comes from safe and acceptable sources.	<ul style="list-style-type: none"> - Site administrators can add and remove approved sources. Site administrators can add or remove prohibited sources. - The system will only locate and display content from approved or non-prohibited sources. 	P3
Flexible Output Formats				
4	Formatting Options	As a user, I want to be able to post and share a newsletter to different platforms, and have my content be tailored to the	<ul style="list-style-type: none"> - The user can select a platform to share their generated content. 	P2

		different platforms format restrictions (i.e. character count)	- The system formats these newsletters and content to fit the chosen platform or service (email chain, pdf, etc.)	
5	Format Option Editing	As an administrator, I want to be able to only have approved websites and platforms available.	- An administrator account can add and remove platforms and format options from the list of approved ones.	P2
Dashboard Management				
6	Dashboard Viewing	As a user, I want to view a list of generated content on my dashboard so that I can quickly access past outputs.	- The dashboard displays a paginated list of generated content with clear titles and brief summaries - The list is updated in real-time as new content is generated - Users can mark items as “read” or “favorite” for future reference	P2
7	Filter and Sort	As a user, I want to filter and sort my generated content by date, platform, or relevance to easily find specific items.	- Users can filter content by date, platform, and relevance - Users can sort content by date, platform, performance metrics	P4
8	Download Content Directly	As a user, I want to download generated content directly from the dashboard to use it on other platforms.	- Each piece of generated content has a “Download” button - Supported formats include PDF for newsletters, text files or CSV (for data posts) - Downloads are fast and error-free	P5
9	Customize Preferences	As a user, I want to customize my preferences (topics, template styles, keywords) directly from the dashboard so I can update my settings easily.	- Changes take effect immediately or upon the next scheduled generation	P4
Automated Scheduling				
10	Schedule Automatic Content Generation	As a user, I want to set a schedule for automatic content generation, so that I do not need to do that manually.	- There are predefined intervals from which the user can select from - There is a custom interval option from which the user can enter a custom generation schedule - The model generates content based on the interval and adds it to the user's dashboard (or account)	P3
11	Notifications for the Automatic	As a user, I want to be notified whenever my scheduled content is generated, so that I do not	- When the content is automatically generated (based on the interval), the user gets notified via email	P4

	Content Generation	need to check the website periodically.		
Automated Content Summarization				
12	Generate Summary and Share	As a user, I want to have an easy way to create summarized posts, so I have content to use.	- The main points of the article(s) are summarized	P1
13	Content sharing	As a user, I want to be able to share my content on social media, so I don't need to copy/paste it to my feed.	- Users can share their content on social media from the website	P2
Customizable Templates				
14	Pre-Determined Templates	As a user, I want to be able to share the generated content to social media using pre-determined templates so that I can save time and resources.	- The system should display various templates for the user to select. - The user should be able to select the preferred template for exporting to social media.	P2
15	Custom Template Design	As an editor, I want to edit AI-generated content into custom templates so that I can ensure it aligns with the brand's tone and style.	- Users should be able to manually create their own templates for content sharing. - The system should save the template for future use.	P2
Premium Features				
16	Multi-Language Support	As a user, I want to generate content in different languages so that I can reach a global audience.	- Users can select their preferred language before generating content. - The system generates the content accordingly.	P3
17	API Integration	As a developer, I want to access an API for content generation so that I can integrate this service into my application.	- An API is available for integration with other applications. - The API fully supports the functionality of the application.	P3
Basic UI				
18	Basic UI Setup	As a user, I want to be able to access the website, so I can use the service.	- Users can go to the website without any errors (404, etc.)	P2
19	Basic Signup Logic and Function	As a user, I want to be able to sign up/login, so my data is saved.	- Users can sign up using their email to create a unique account (based on their email) - Users can sign out and re-login whenever they want (after signing up)	P2

Product Backlog:

ID	Task	Associated User Story	Priority
Custom News Aggregation			
1	Create a webpage for which the user enters and saves URLs or keywords for content aggregation.	As a user, I want to be able to input my favorite news sources, topics, keywords, locations, or industries and have the system gather relevant content so that I can quickly review content from my favorite sources.	P1
2	Implement storing and accessing user sources using a database.		P1
3	Implement a content retrieval mechanic to obtain content directly from these sources.		P1
4	Implement a matching algorithm to filter content based on source/keywords.		P1
5	Create a webpage for which the user can view content associated with a relevant category.	As a user, I want the system to tag sources with a category so that it can enhance the accuracy of the content displayed.	P4
6	Implement a categorization system for content sources with pre-defined categories.		P4
7	Create a webpage for which administrators can view and manage approved and prohibited sources.	As a site administrator, I want to be able to manage the list of approved and prohibited sources so that news and content only comes from safe and acceptable sources.	P3
8	Implement functionality for approving and prohibiting sources.		P3
Automated Scheduling			
9	Create a webpage for which the user goes, to set a schedule	As a user, I want to set a schedule for automatic content generation, so that I do not need to do that manually.	P3
10	Create buttons to change the schedule interval (one custom interval button the rest predefined)		P3
11	Utilize the database to store user schedule information		P1
12	Add input field on the page and button to save topic selection	As a user, I want to be able to enter links, and topics for my scheduled automatic content generation, so that the system knows what content to generate.	P2
13	Utilize the database to store user schedule topic information		P1
14	Create an email template	As a user, I want to be notified whenever my scheduled content is	P2
15	Create function to fill the email template with the user's		P2

	information and link to the new generated content (when the content is generated)	generated, so that I do not need to check the website periodically.	
16	Create function to send the filled email to the user		P2
Automated Content Summarization			
17	Research LLMs and different datasets.	As a user, I want to have an easy way to create summarized posts, so I have content to use.	P1
18	Implement and train the LLM.		P1
19	Fine-tune and optimize the LLM.		P1
20	Integrate the LLM with the content aggregation system.		P1
21	Add button to generate the content.		P3
22	Create a list of connected social media platforms from which the user can select from.	As a user, I want to be able to share my content on social media, so I don't need to copy/paste it to my feed.	P4
23	Add button to share/post summarized content to the selected social media platform.		P4
Flexible Output Formats			
24	Add dropdown or list for user to select their desired platform from after the system has checked that articles have been gathered.	As a user, I want to be able to post and share a newsletter to different platforms and not be restricted by platform requirements.	P3
25	Switching between the different dropdowns should change the displayed format without the need to any refresh buttons or "reformat buttons."		P4
26	From an administrator login, a list of available platforms should be displayed and should be able to be edited.	As an administrator, I want to be able to only have approved websites and platforms available.	P3
Dashboard Management			
27	Add sorting functionality for parameters.	As a user, I want to filter and sort my generated content by date, platform, or relevance to easily find specific items.	P3
28	Add an apply/reset filter button to refine content display		P4
29	Implement an API to generate and provide downloadable files (PDF, text, CSV)	As a user, I want to download generated content directly from the dashboard to use it on other platforms.	P5

30	Design a form on the dashboard for updating preferences	As a user, I want to customize my preferences (topics, template styles, keywords) directly from the dashboard so I can update my settings easily.	P5
31	Design and implement the dashboard UI with a section for displaying content	As a user, I want to view a list of generated content on my dashboard so that I can quickly access past outputs.	P3
32	Create a database schema for storing generated content		P2
Customizable Templates			
33	Design a webpage to allow users to select a template for social media sharing.	As a user, I want to be able to share the generated content to social media using pre-determined templates so that I can save time and resources.	P2
34	Implement a preview window so the user can ensure the template meets their requirements before sharing.		P5
35	Create a content editor interface with text formatting options for editing AI-generated content	As an editor, I want to edit AI-generated content into custom templates so that I can ensure it aligns with the brand’s tone and style.	P2
Premium Features			
36	Integrate a payment system to allow users to purchase the premium features.	16 & 17 (both below)	P1
37	Implement the ability to select the language for content generation.	As a user, I want to generate content in different languages so that I can reach a global audience.	P3
38	Develop an API for accessing the content generation functionality.	As a developer, I want to access an API for content generation so that I can integrate this service into my application.	P3
Basic UI			
39	Register a domain/host website through Brock.	As a user, I want to be able to access the website, so I can use the service.	P4
40	Create a basic landing page.		P1

41	Create a basic sign up/registration page.	As a user, I want to be able to sign up/login, so my data is saved.	P2
42	Create a basic login page.		P2
43	Create a table in the database to save user login/registration information.		P1

Sprint Backlog:

Sprint	Tasks
Sprint 1 (Jan. 21 – Jan. 27)	<ul style="list-style-type: none"> - Register a domain/host website through Brock. - Create a basic landing page. - Create a basic sign up/registration page. - Create a basic login page. - Create a table in the database to save user login/registration information. - Research LLMs and different datasets
Sprint 2 (Jan. 28 – Feb. 11)	<ul style="list-style-type: none"> - Develop a system to fetch content from user inputted sources APIs (news sources): <ul style="list-style-type: none"> • register NewsAPI (research familiarize) • Write to fetch data from the API based on user inputted sources keywords (topics, date range, source names) • Database schema for storing fetched content (URL, date, summary, source) • TESTING <ul style="list-style-type: none"> ○ Test fetching content with various user inputs ○ Test cases like invalid inputs or network issues • UI for inputting sources (security) - Start training the LLM on our chosen dataset(s) - Fine-tune and optimize the LLM - Placeholder for displaying aggregated sources or summarized content
Sprint 3 (Feb. 11 – Feb. 23)	<ul style="list-style-type: none"> - Continue training/fine-tuning LLM for chosen dataset(s) - Create a summarize function for users: <ul style="list-style-type: none"> • Add a summarize button • Make sure allow to summarize multiple article - Create a button to find similar news articles gathered automatically - Make our AI able to search news article for same theme. - Add link to webpage
Sprint 4 (Feb. 25 – Mar. 11)	<ul style="list-style-type: none"> - Integrate the LLM for summarizing aggregated content <ul style="list-style-type: none"> • Connect the language model and aggregated content system together - Allow users to set schedules for automatic content (daily, weekly)

	<ul style="list-style-type: none"> • Task scheduler (celery, cron jobs) • email • Favorite articles <ul style="list-style-type: none"> - Display aggregated content (add filters and sorting) - Improve UI for better navigation - Create a share function for users: <ul style="list-style-type: none"> • Add a share button • Add a list for people to share • Add notification part - Create a list which wrote by same author <ul style="list-style-type: none"> • Make a profile for users and make sure users can find their article in their profile. - APIs for post contenting directly to social media platforms (Twitter, LinkedIn, Facebook)
Sprint 5 (Mar. 11 – Mar. 23)	<ul style="list-style-type: none"> - Final testing and polishing - Implement download functionality in multiple formats - Create customizable templates for users

Tool Screenshots:












Epic, User Stories, and Tasks for Custom News Aggregation:

<input type="checkbox"/>	Type	# Key	Summary	Status	Priority	Sprint
<input type="checkbox"/>	▼	C4G9P-7	Custom News Aggregation: Use...	TO DO	P1	
<input type="checkbox"/>	▼	C4G9P-39	As a user, I want to be able to in...	TO DO	P1	
<input type="checkbox"/>		C4G9P-42	Create a webpage for which the...	TO DO	P1	
<input type="checkbox"/>		C4G9P-43	Implement storing and accessin...	TO DO	P1	
<input type="checkbox"/>		C4G9P-44	Implement a content retrieval m...	TO DO	P1	
<input type="checkbox"/>		C4G9P-45	Implement a matching algorithm...	TO DO	P1	
<input type="checkbox"/>	▼	C4G9P-40	As a user, I want the system to t...	TO DO	P4	
<input type="checkbox"/>		C4G9P-46	Create a webpage for which the...	TO DO	P4	
<input type="checkbox"/>		C4G9P-47	Implement a categorization syst...	TO DO	P4	
<input type="checkbox"/>	▼	C4G9P-41	As a site administrator, I want to...	TO DO	P3	
<input type="checkbox"/>		C4G9P-48	Create a webpage for which ad...	TO DO	P3	
<input type="checkbox"/>		C4G9P-49	Implement functionality for app...	TO DO	P3	













Epic, User Stories, and Tasks for Flexible Output Formats:

<input type="checkbox"/>	Type	# Key	Summary	Status	Priority	Sprint
<input type="checkbox"/>	▼	C4G9P-11	Flexible Output Formats: Gener...	TO DO	P1	
<input type="checkbox"/>	▼	C4G9P-30	As a user, I want to be able to p...	TO DO	P2	
<input type="checkbox"/>		C4G9P-32	Create a way to check if articles ...	TO DO	P2	
<input type="checkbox"/>		C4G9P-33	Create a dropdown for users to ...	TO DO	P2	
<input type="checkbox"/>		C4G9P-31	Display the different format opti...	TO DO	P2	
<input type="checkbox"/>		C4G9P-34	Save the currently selected form...	TO DO	P3	
<input type="checkbox"/>	▼	C4G9P-35	As an administrator, I want to b...	TO DO	P3	
<input type="checkbox"/>		C4G9P-50	When logged in to a administra...	TO DO	P3	
# <input type="checkbox"/>		C4G9P-51	The administrator login should ...	TO DO	P3	











Epic, User Stories, and Tasks for Dashboard Management:

<input type="checkbox"/>	Type	# Key	Summary	Status	Priority	Sprint
<input type="checkbox"/>	▼ 	C4G9P-15	Dashboard Management: Regist...	TO DO	P1	
<input type="checkbox"/>	▼ 	C4G9P-55	As a user, I want to customize ...	TO DO	P4	
<input type="checkbox"/>		C4G9P-56	Design a form on the dashboar...	TO DO	P5	
<input type="checkbox"/>	▼ 	C4G9P-54	As a user, I want to download g...	TO DO	P5	
<input type="checkbox"/>		C4G9P-58	Implement an API to generate a...	TO DO	P5	
<input type="checkbox"/>	▼ 	C4G9P-53	As a user, I want to filter and so...	TO DO	P4	
<input type="checkbox"/>		C4G9P-60	Add an apply/reset filter button...	TO DO	P5	
<input type="checkbox"/>		C4G9P-59	Add sorting functionality for par...	TO DO	P3	
<input type="checkbox"/>	▼ 	C4G9P-52	As a user, I want to view a list of...	TO DO	P2	
<input type="checkbox"/>		C4G9P-62	Create a database schema for st...	TO DO	P2	
<input type="checkbox"/>		C4G9P-61	Design and implement the dash...	TO DO	P3	

Epic, User Stories, and Tasks for Automated Scheduling:







<input type="checkbox"/>	Type	# Key	Summary	Status	Priority	Sprint
<input type="checkbox"/>	▼ 	C4G9P-12	Automated Scheduling: Users c...	TO DO	P1	
<input type="checkbox"/>	▼ 	C4G9P-70	As a user, I want to set a schedu...	TO DO	P3	
<input type="checkbox"/>		C4G9P-76	Create new db to store user sch...	TO DO	P1	
<input type="checkbox"/>		C4G9P-75	Create buttons to change the sc...	TO DO	P3	
<input type="checkbox"/>		C4G9P-74	Create a webpage for which the...	TO DO	P3	
<input type="checkbox"/>	▼ 	C4G9P-68	As a user, I want to be able to e...	TO DO	P3	
<input type="checkbox"/>		C4G9P-78	Create new db to store user sch...	TO DO	P1	
<input type="checkbox"/>		C4G9P-77	Add input field on the page and...	TO DO	P2	
<input type="checkbox"/>	▼ 	C4G9P-64	As a user, I want to be notified ...	TO DO	P4	
<input type="checkbox"/>		C4G9P-81	Create function to send the fille...	TO DO	P2	
<input type="checkbox"/>		C4G9P-80	Create function to fill the email ...	TO DO	P2	
<input type="checkbox"/>		C4G9P-79	Create an email template.	TO DO	P2	

Epic, User Stories, and Tasks for Automated Content Summarization:







<input type="checkbox"/>	Type	# Key	Summary	Status	Priority	Sprint ↑ ▼
<input type="checkbox"/>	▼ 	C4G9P-14	Automated Content Summarization: Uses adva...	TO DO	P1	
<input type="checkbox"/>	▼ 	C4G9P-25	As a user, I want to have an easy way to create ...	TO DO	P1	COSC 4P02 - Sprint 1
<input type="checkbox"/>		C4G9P-109	Research possible LLMs and different datasets	TO DO	P1	COSC 4P02 - Sprint 1
<input type="checkbox"/>		C4G9P-108	Implement and train the LLM.	TO DO	P1	COSC 4P02 - Sprint 1
<input type="checkbox"/>		C4G9P-82	Fine-tune and optimize the LLM.	TO DO	P1	COSC 4P02 - Sprint 1
<input type="checkbox"/>		C4G9P-65	Integrate the LLM with the content aggregatio...	TO DO	P1	COSC 4P02 - Sprint 1
<input type="checkbox"/>		C4G9P-66	Add button to generate the content.	TO DO	P4	COSC 4P02 - Sprint 1
<input type="checkbox"/>	▼ 	C4G9P-26	As a user, I want to be able to share my conten...	TO DO	P3	
<input type="checkbox"/>		C4G9P-67	Create a list of connected social media platfor...	TO DO	P4	
<input type="checkbox"/>		C4G9P-104	Add button to share/post summarized content ...	TO DO	P4	

**Note: Only the task, "Research possible LLMs and different datasets" is a part of Sprint 1. In Jira, the user story needs to be included in the sprint that any child tasks are a part of.*

Epic, User Stories, and Tasks for Customizable Templates:

<input type="checkbox"/>	Type	# Key	Summary	Status	Priority	Sprint
<input type="checkbox"/>	▼ 	C4G9P-13	Customizable Templates: Desig...	TO DO	P1	
<input type="checkbox"/>	▼ 	C4G9P-90	As a user, I want to be able to s...	TO DO	P2	
<input type="checkbox"/>		C4G9P-92	Design a webpage to allow user...	TO DO	P2	
<input type="checkbox"/>		C4G9P-93	Implement a preview window s...	TO DO	P5	
<input type="checkbox"/>	▼ 	C4G9P-91	As an editor, I want to edit AI-g...	TO DO	P2	
<input type="checkbox"/>		C4G9P-95	Create a content editor interfac...	TO DO	P2	

Epic, User Stories, and Tasks for Premium Features:

<input type="checkbox"/>	Type	# Key	Summary	Status	Priority	Sprint
<input type="checkbox"/>	▼ 	C4G9P-96	Premium Features: Users can pu...	TO DO	P1	
<input type="checkbox"/>	▼ 	C4G9P-97	As a user, I want to generate co...	TO DO	P3	
<input type="checkbox"/>		C4G9P-100	Integrate a payment system to ...	TO DO	P1	
<input type="checkbox"/>		C4G9P-101	Implement the ability to select t...	TO DO	P3	
<input type="checkbox"/>	▼ 	C4G9P-98	As a developer, I want to access...	TO DO	P3	
<input type="checkbox"/>		C4G9P-102	Develop an API for accessing th...	TO DO	P3	

User Stories and Tasks for Basic UI (Sprint 1):

<input type="checkbox"/>	Type	# Key	Summary	Status	Priority	Sprint
<input type="checkbox"/>	▼	C4G9P-83	As a user, I want to be able to a...	TO DO	P2	COSC 4P02 - Sprint 1
<input type="checkbox"/>		C4G9P-86	Create a basic landing page.	TO DO	P1	COSC 4P02 - Sprint 1
<input type="checkbox"/>		C4G9P-85	Register a domain/host website ...	TO DO	P4	COSC 4P02 - Sprint 1
<input type="checkbox"/>	▼	C4G9P-84	As a user, I want to be able to si...	TO DO	P2	COSC 4P02 - Sprint 1
<input type="checkbox"/>		C4G9P-89	Create a table in the database t...	TO DO	P1	COSC 4P02 - Sprint 1
<input type="checkbox"/>		C4G9P-88	Create a basic login page.	TO DO	P2	COSC 4P02 - Sprint 1
<input type="checkbox"/>		C4G9P-87	Create a basic sign up/registrati...	TO DO	P2	COSC 4P02 - Sprint 1

Market Research:

In analyzing existing SaaS platforms that align with our product planning requirements, we focused on Paper.li and Scoop.it, both of which offer content aggregation, curation, and distribution features. Paper.li provides an easy-to-use interface for automating content aggregation based on user-defined topics, sources, and keywords. It generates personalized newsletters that can be distributed via email, offering an efficient solution for users seeking automated content curation. However, Paper.li's limitations are clear in its inability to easily format content for various social media platforms and its lack of advanced scheduling options for multi-platform publishing. Additionally, the AI-driven content summaries are basic, which may not satisfy users looking for more nuanced, tailored output.

In contrast, Scoop.it offers a more sophisticated approach to content curation and publishing. It allows users to aggregate content from specific sources and topics and supports direct publishing to multiple social media platforms. One of Scoop.it's key strengths lie in its detailed analytics, enabling users to track performance across platforms. However, its pricing model can be prohibitive for smaller teams or students, limiting accessibility. Furthermore, while it allows for some content customization, Scoop.it does not offer extensive options for creating fully branded templates for newsletters, and its scheduling features, though present, are not as intuitive as those found in competing platforms.

Our proposed system seeks to build on the strengths of both Paper.li and Scoop.it while addressing their shortcomings. Unlike Paper.li, which focuses primarily on newsletter creation, our platform will support multiple output formats, including tailored posts for social media platforms such as Twitter, LinkedIn, and Instagram. This multi-format approach will provide greater flexibility for users to share content across various channels. In contrast to Scoop.it's focus on advanced analytics, which may not be a priority for small businesses or individual users, our platform will prioritize accessibility and affordability. By offering automated content generation with customizable scheduling, notifications, and content summaries powered by advanced natural language processing (NLP) and large language models (LLMs), our system will deliver more personalized and accurate content than the basic summaries offered by both Paper.li and Scoop.it.

The SaaS market is experiencing substantial growth, with content management tools being a significant area of demand. In 2023, SaaS adoption reached 85% of enterprise software usage, and organizations continue to allocate significant resources toward SaaS solutions. On average, companies spend \$9,600 per employee annually on SaaS products, demonstrating a strong appetite for innovative content management tools. This market trend underscores the potential success of our platform, particularly with its unique combination of flexibility, user-friendliness, and affordability, which is particularly attractive to the growing number of small businesses, students, and social media influencers in need of accessible content solutions

<https://www.saasacademy.com/blog/saas-statistics>

<https://redline.digital/saas-industry-statistics/>

Release Planning Challenges:

There were several challenges encountered during the Release Planning process:

→ Unclear Requirements:

- Determining tasks for epics and breaking down required tasks into what epics they fit under.
- Starting our project board on GitHub seemed like the best approach initially and turned out to not fit our needs and was not the most user friendly, so we switched to using Jira to define sprint backlogs and tasks.
- It took a bit of work to sort out a clear formatting style for release planning document and backlogs.

→ Team Coordination:

- Coordinating between 7 developers with different schedules, workloads, and availabilities was a challenge encountered during this release planning process. This week two official meetings were held, the first during the scheduled lecture time (attended by all members) and the second on Thursday with the TA (attended by 6/7 members due to scheduling conflicts). Going forward, we aim to have all meetings during the 9-12 block on Tuesday to ensure full attendance by the group.
- To this point, work has been split (relatively) evenly between group members

For the next phase of the project, we aim to:

→ Modify Backlog:

- As the project progresses, small changes to the backlog may be required. User stories may need to be broken down into further tasks, or some tasks may need to be refined. As a group, we will aim to clarify which backlog items require modification and modify these accordingly.

→ Begin Sprint 1:

- A full team meeting will be conducted on Tuesday to divide the tasks for the first sprint. The first sprint will be the shortest, lasting one week in duration. It will also begin on Tuesday after our meeting.

Appendix A - Meeting Minutes:

January 14, 2025 – Meeting 2

Joined: Dea, Shijie, Dalton, Jaden, Nicholas, Thomas, Chidera.

- The group went over the project description and worked together to break the project into several themes and epics. The project was broken down into 6 initial themes:
 - Six group members were allocated a theme and were tasked with breaking down the theme into user stories, and then into further tasks (Shijie, Dalton, Jaden, Nicholas, Thomas, Chidera). A single group member was allocated to reviewing and modifying these themes, stories, and tasks to ensure they were clear and concise (Dea).
- The GitHub Repository ReadMe was modified (Thomas).
- The group also discussed the Release Planning Meeting with the TA. Thursday (1/16/2025) at 11:40 was the timeslot selected, where the most members could attend (6/7 were able to attend, as the meeting is outside of the course hours).
- Next meeting: The group will show the initial themes and user stories to the TA and seek clarification about the report format. Once approved, we will begin breaking down the user stories into tasks and implementing these tasks in GitHub Projects.

January 16, 2025 – Meeting 3 - Release Planning Meeting

Joined: Dea, Dalton, Jaden, Nicholas, Thomas, Chidera.

Absent: Shijie (schedule conflict)

- The group presented the themes and user stories to the TA.
 - Also requested clarification on the format of the report.
 - We were advised that a template was now provided on Brightspace for the report.
 - Suggestions were also made for additional contents in the report. For example, researching a similar market application and including pros and cons to the features it already has.
- After the Release Planning Meeting was finished, the group met to debrief the meeting.
 - Now that the user stories were presented, we would begin breaking down the stories into tasks.

- As discussed in the previous meeting, six group members were allocated a theme, and tasked with breaking down the theme into user stories, and then into further tasks (Shijie, Dalton, Jaden, Nicholas, Thomas, Chidera). One group member would be responsible for performing market research and including this in the report (Dea).
- The group also discussed GitHub Projects, and it was determined that Jira may be the best software to implement our backlog, as it has many options and customizations that are not available within GitHub.
 - The group agreed, and the switch was made. All members of the group were added to the Jira Project.
- Next meeting: The group will have completed the user stories, product backlog, and the sprint backlogs. They will have been implemented in Jira, and the Release Planning Document will have been submitted. The group will meet on Tuesday (1/21/2025) to allocate tasks for Sprint 1 and will formally begin implementing the project.

Appendix B - Team Contributions:

Team Member	Contribution
Jaden Kuhn	<ul style="list-style-type: none"> - Wrote user stories for a feature - Helped write the sprint backlog - Helped set up the Jira board, adding the priority field and epics/user stories/tasks - Decomposed user stories into tasks (product backlog) - Added epic/user stories/tasks into Jira - Wrote the introduction
Chidera Nwana	<ul style="list-style-type: none"> - Wrote user stories for a feature - Contributed ideas for user stories - Decomposed user stories in the Customizable Templates section into tasks with added priorities f - Updated the product backlog for the Customizable Templates section
Dalton Morris	<ul style="list-style-type: none"> - Wrote user stories for a feature - Filled in non-functional requirements - Broke down some features into tasks on Jira - Filled out some of the challenges section
Dea Kukuqani	<ul style="list-style-type: none"> - Verified the user stories for contradictions and accuracy - Did market research and go over extra features we want to include or expand

	on – focused on two similar products that we want to compare to while meeting market needs
Nicholas Caruso	<ul style="list-style-type: none"> - Created the Jira page for the project. - Wrote user stories for features. - Decomposed user stories into tasks. - Implemented stories and tasks into Jira. - Wrote the “Challenges” section. - Included “Meeting Minutes”.
Thomas Semenak	<ul style="list-style-type: none"> - Wrote user stories for a feature and decomposed into tasks - Updated the readme on GitHub - Brain-stormed for sprint backlog - Helped with sprint backlog - Helped set up the Jira board, adding epics/user stories/tasks
Shijie Tong	<ul style="list-style-type: none"> - Wrote user stories for a feature - Helped with Jira (set up story and tasks) - Helped with sprint backlog - Helped with product backlog