

2023-11-02

CMPT 276 PROJECT DOC



TABLE OF CONTENTS

01	<u>Project Overview</u>
02	<u>SDLC Selection</u>
03	<u>User Stories</u>
04	<u>Technology Stack</u>
05	<u>Chosen APIS and features</u>
06	<u>Work Breakdown Structure</u>
07	<u>Project Schedule</u>
08	<u>Wireframe and Prototypes</u>
09	<u>Data Flow Diagrams</u>
10	<u>Conclusion and Credits</u>

PROJECT OVERVIEW

This is the document that talks about group Root 9's CMPT 276 project. The group members are Stefan Pricope, Taiga Okuma, Cindy Xiao and Anna Rusinova with proper emails and the GitHub repository link at the [credits](#) section.

This report will largely overview the planning part of our project, and speak on the choices we have made with components in use for the project implementation. Largely the document follows an order of technology and APIs, to data representation, and then some documentation of our work.

The main focus of the project is around the central idea of creating a web application focused on generating color palettes from images. The web application will be able to:

- Generate colour palettes from an uploaded image
- Export individual colours, entire palettes, or the whole board.
- Create collages with images and their corresponding palettes
- Adding generated colour palettes to a board

With this web application, we hope to give users from a variety of backgrounds the option to identify the colours and palettes they need for work, personal life or other reasons.



SDLC SELECTION

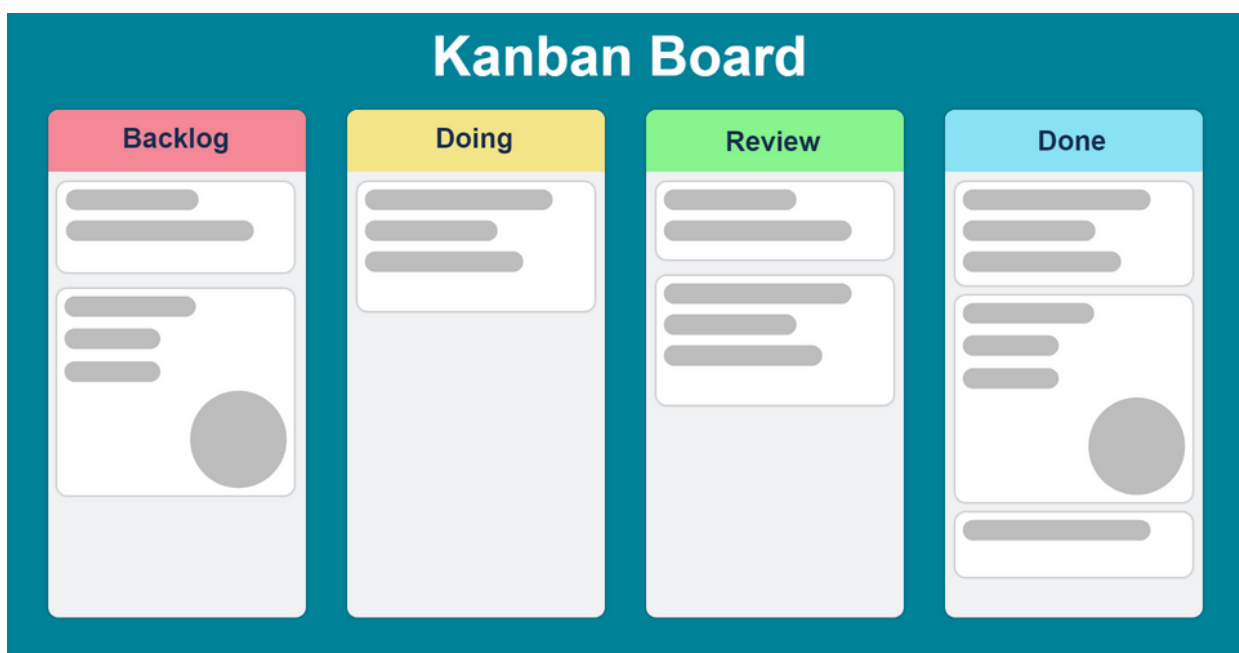
The SDLC we decided to choose was Agile, more specifically the Kanban framework of Agile.

The decision to opt for the Kanban model under the Agile methodology was influenced by its visual nature, which allows for an immediate understanding of the project's status, work in progress, and what tasks are upcoming. By using a Kanban board, team members can easily visualize the flow of tasks, pinpoint bottlenecks, and allocate resources more effectively.

Furthermore, Notion was identified as the optimal platform for implementing the Kanban model for this project. Notion's user-friendly interface, seamless integration capabilities, and flexibility make it a top choice. Its Kanban view not only offers a clean and intuitive visualization of tasks but also allows for easy updates, making the process of tracking and progressing tasks straightforward.

In essence, the combination of the Kanban model with Notion serves a dual purpose. First, it aligns with our goal of maintaining transparency and agility in our development process. Second, it provides an efficient tool that caters to the team's needs, ensuring that tasks are monitored, managed, and executed in a streamlined manner.

For a direct view of our project's progression, you can access our Kanban board in Notion: [Click Here](#)



USER STORIES



No. 01 – Hobbyists and Enthusiasts

Crafters and DIYers: Want to extract color palettes from inspiring images they come across, making it easier to pick paint colors or craft materials for their next project.

Home Decorators: Want to redesign their living space based on a particular image or scenery they love. By uploading images, they can get a color palette to guide their purchasing decisions for paint, furniture, and decor.



No. 02 – Design Professionals

Graphic Designers: Need to consistently create color palettes for branding or design projects. They can upload logos or brand images to extract a cohesive color palette.

Interior Designers: Work with clients who might share an inspiration image (like a sunset or artwork). Designers can upload this image, extract the color palette, and create a mood board that integrates the extracted colors and potential furniture items or decor.



No. 03 – Digital Content Creators

Bloggers and Social Media Influencers: Want to maintain a consistent theme or aesthetic across their content. They can extract color palettes from their favorite posts or images and use them as a guideline for future content.

Photographers: Interested in understanding the color dynamics of their photos. By extracting color palettes, they can better understand and communicate the mood or tone of their photography.



No. 04 – Educational and Commercial Entities

Schools and Universities: May want to create thematic boards for events or functions based on the institution's colors or an event's theme.

Businesses: When launching a product or campaign, businesses might want to set a theme. By uploading product images or campaign visuals, they can create a color-guided mood board that directs other aspects of the campaign.

TECHNOLOGY STACK

Our project leverages a robust technology stack that is both modern and efficient, consisting of HTML, CSS, JavaScript, and React.

HTML:

- **Foundation of Webpages:**

HTML provides the basic structure of our web application, allowing us to define the content and layout with standard text and multimedia elements.

- It is universally supported across all web browsers, ensuring our application's accessibility.

CSS (Cascading Style Sheets)

- **Styling and Presentation:**

CSS is used for the visual styling of our web application. It allows us to create a more engaging and responsive user interface.

With CSS, we can ensure that our application is visually appealing and provides a seamless user experience across different devices and screen sizes.

JavaScript

- **Interactivity and Functionality:**

JavaScript adds interactivity to our web pages, from handling form submissions to dynamic content updates. It's a powerful scripting language that enables us to implement complex features, like real-time updates and interactive mood boards, without the need for a page reload.

React (A JavaScript library)

- **Component-Based Architecture:**

React's component-based architecture makes it easy to manage and maintain our codebase. We can build encapsulated components that manage their own state and then compose them to make complex UIs.

- **Reusability and Efficiency:**

Components are reusable, which enhances development efficiency and reduces the likelihood of bugs. This modularity allows for rapid iteration and development.

- **High Performance:**

React's virtual DOM (Document Object Model) optimizes rendering, allowing our application to perform well even with high user interaction and data updates.

- **Strong Community and Ecosystem:**

With a vast community, React provides a wealth of open-source packages and extensions, enabling us to leverage additional functionalities and tools to enrich our application.

- **Industry Standard:**

React is widely adopted in the industry, ensuring that our technology stack aligns with current professional practices and enhances our team's employability.

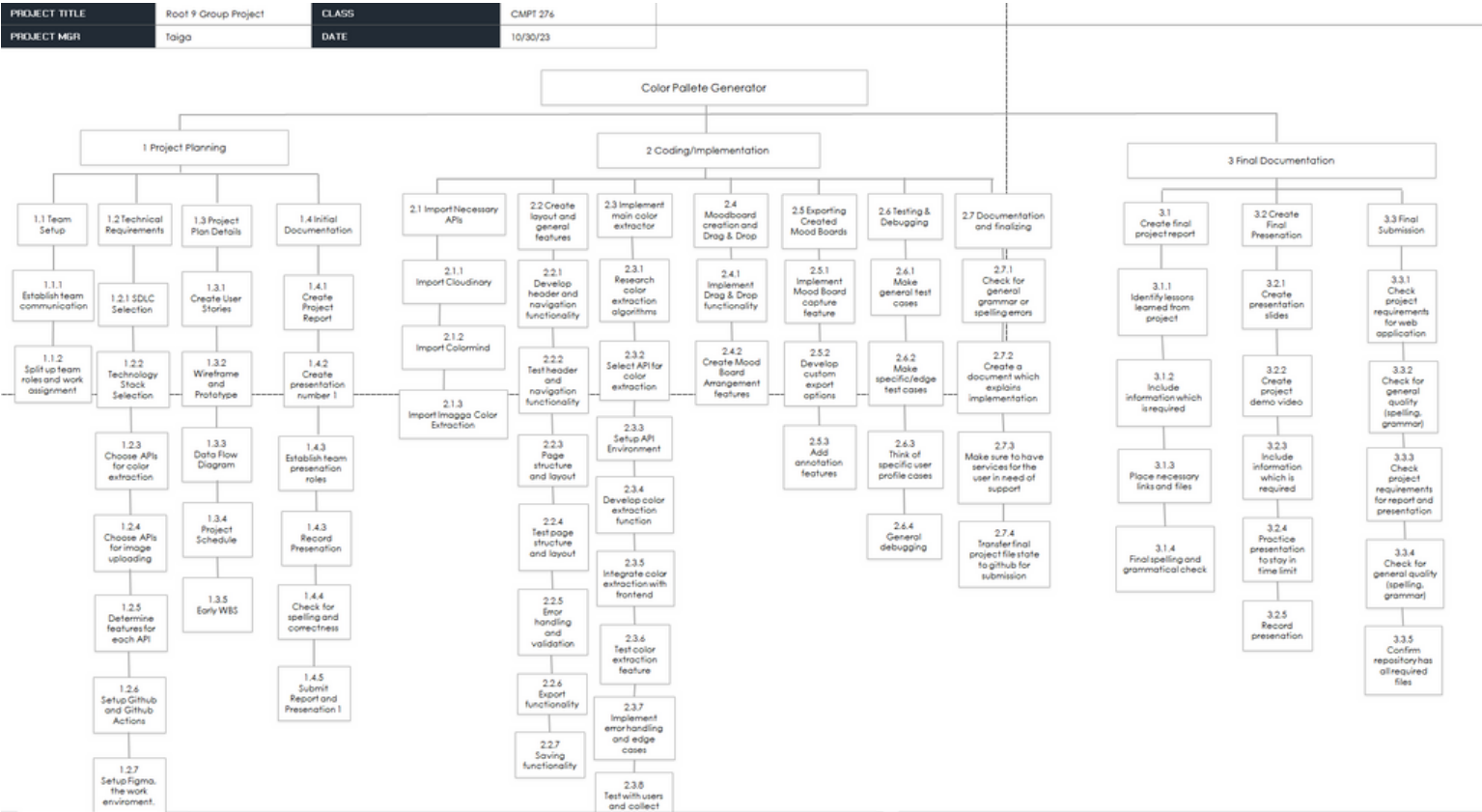
By integrating these technologies, we have built a foundation that is both scalable and flexible, allowing for future growth and development as new needs arise. Our chosen stack balances efficiency in development with delivering a high-quality user experience.

APIS AND THEIR FEATURES

API Name	API Purpose	API Features
Colormind	Color Extraction & Palette Generation	<ul style="list-style-type: none">• Generate color palettes from uploaded images and offer palette suggestions.• Allows user to lock preferred colors and customize the number of colors to extract.• Provides hex codes and RGB values for all colors.
Imagga Color Extraction	Color Extraction & Palette Generation	<ul style="list-style-type: none">• Identify main colors in an image and display the percentage composition of colors.• Automatically tag images with main colors and provide detailed color information.• Allow users to extract palettes from specific parts of an image.
Cloudinary	Image Upload & Management	<ul style="list-style-type: none">• Image uploading with drag and drop with auto resize features.• Automatic optimization of images and converts images to web-friendly formats.• Fast image delivery and dynamic URL generation for easy retrieval of images.
html2canvas	Exporting Created Mood Boards	<ul style="list-style-type: none">• Convert the users mood board into a static image file and allow for high resolution exports.• Provide options for different image formats with export size and quality settings.• Users can add titles, notes and preview their board.
Github Actions	Continuous Integration for Software	<ul style="list-style-type: none">• Compile and build the application on every commit and run automated tests to validate new features or fixes.• Auto deploy the main branch to production after successful tests and rollback if tests fail.• Integrate linting tools to main code quality and flag code that does not meet style guidelines.

WORK BREAKDOWN STRUCTURE

Below you can find a screenshot of the main WBS diagram, and two screenshots from the table format of the WBS. The complete excel file is available at the GitHub repository in the project files named as Root9_WBS.xlsx, or can be found [here](#).



Task #	Task Title	Assigned Group Member	Expected Time (in Hours)	Actual Time (in Hours)
3	Final Documentation			
3.1	Create Final Project Report			
3.1.1	Identify lessons learned from project	Group	2	
3.1.2	Include information which is required	Group	4	
3.1.3	Place necessary links and files	Group	1	
3.1.4	Final spelling and grammatical check	Group	1	
3.1.5	Create Final Project Report Complete		8	0
3.2	Create Final Presentation			
3.2.1	Create presentation slides	Group	1	
3.2.2	Create project demo video	Group	2	
3.2.3	Include information which is required	Group	2	
3.2.4	Practice presentation to stay in time limit	Group	4	
3.2.5	In class presentation	Group		
3.2.6	Create Final Presentation Complete		9	0
3.3	Final Submission			
3.3.1	Check project requirements for web application	Group	3	
3.3.2	Check for general quality (spelling, grammar)	Group	2	
3.3.3	Check project requirements for report and presentation	Group	2	
3.3.4	Check for general quality (spelling, grammar)	Group	1	
3.3.5	Confirm repository has all required files	Group	1	
3.3.6	Final Submission Complete		9	0
3.4	Final Documentation Complete		9	0

1.2	Technical Requirements			
1.2.1	SDLC Selection	Taiga	1	1
1.2.2	Technology Stack Selection	Taiga	1	1
1.2.3	Choose APIs for color extraction	Taiga	1	0.5
1.2.4	Choose APIs for image uploading	Taiga	1	0.5
1.2.5	Determine features for each API	Taiga/Group	2	1.5
1.2.6	Setup Github and Github Actions	Stefan	1	0.5
1.2.7	Setup Figma, the work environment.	Anna	0.5	0.1
1.2.8	Technical Requirements Complete		7.5	5.1
1.3	Project Plan Details			
1.3.1	Create User Stories	Group	2	0.5
1.3.2	Wireframe and Prototype	Anna		
1.3.2.1	Research industry practices	Anna	1	1
1.3.2.2	Low-fidelity wireframing	Anna		
1.3.2.2.1	Create wireframes	Anna	1	1
1.3.2.2.2	Review and iterate wireframes	Anna	1	0.3
1.3.2.2.3	Approve the layout	Anna	0.5	0.5
1.3.2.3	High-fidelity prototyping	Anna		
1.3.2.3.1	Create UI elements and components	Anna	2	3
1.3.2.3.2	Create high-fidelity versions of prototypes	Anna	5	7
1.3.2.3.3	Add interactivity to prototypes	Anna	2	1
1.3.2.4	Test prototypes for usability	Anna	0.5	0.5
1.3.2.5	Document design specifications, interactions, and guidelines	Anna	0.5	0.25
1.3.3	Data Flow Diagram			
1.3.3.1	Sketch Data Flow Diagram Blueprint	Cindy	0.5	0.5
1.3.3.2	Create Data Flow Based on Users' Perspectives	Cindy	1	0.5
1.3.3.3	Connect the Data Flow Process	Cindy	0.5	0.25
1.3.3.4	Format Data Flow Diagram	Cindy	0.25	0.25
1.3.4	Project Schedule	Cindy		
1.3.4.1	Create a Detailed Outline of Schedule	Cindy	1.5	2
1.3.4.2	Duration/Time frame for each task	Cindy	2	1.5
1.3.4.3	Indicate tasks to Each Assignee	Cindy	0.5	0.5

PROJECT SCHEDULE/TIMELINE

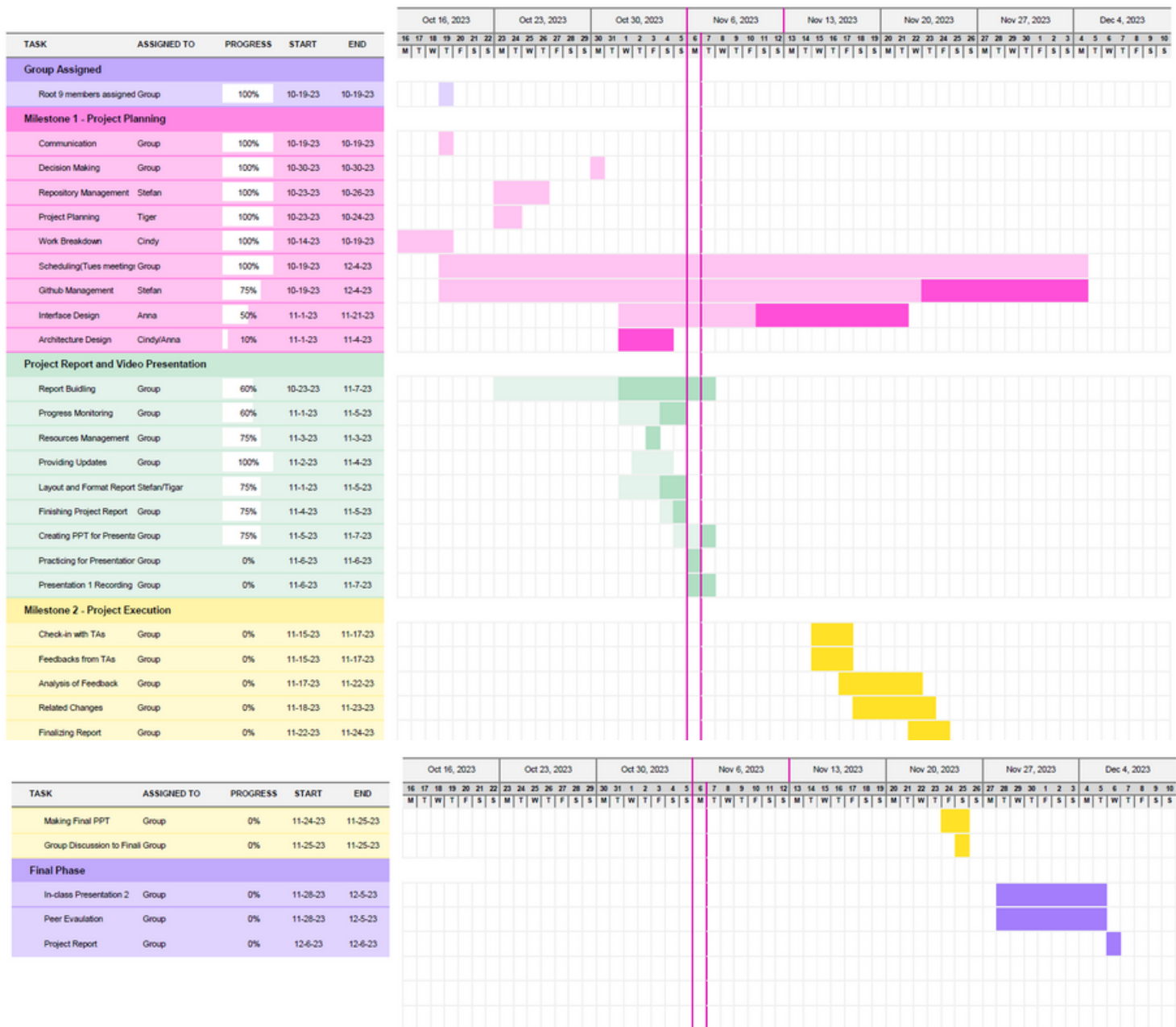
Below you can find screenshots of the project schedule/timeline, and the files for timeline can be found in the GitHub repository in the project plan files folder. To be precise, the timeline excel file is named Root 9 Final Timeline, and can be found [here](#), and the pdf version can be found [here](#). The timeline provides a larger overview of the tasks and objectives we need to accomplish over the period of the project duration.

Group Project

Project Group Root 9

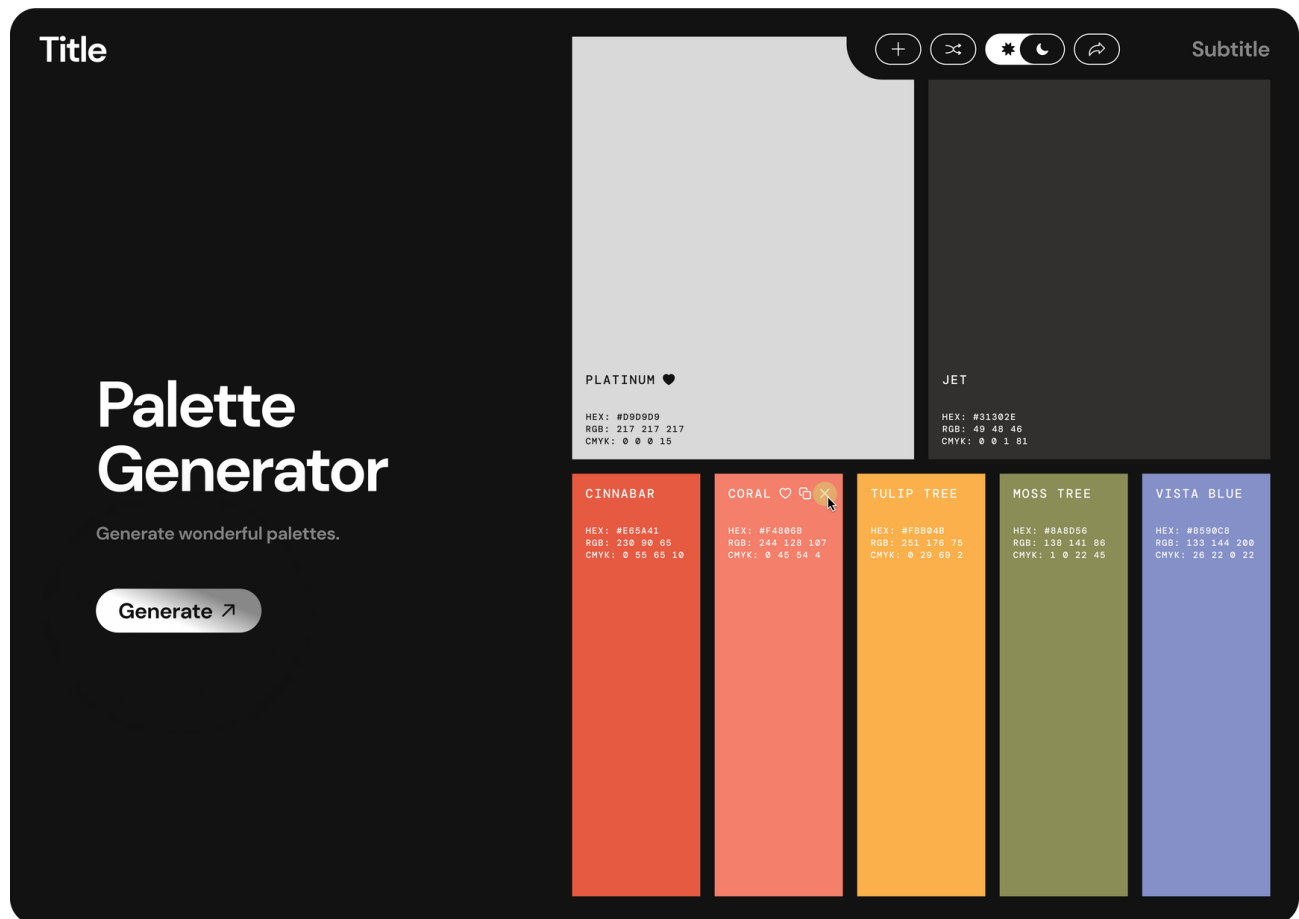
Project start: Thu, 10-19-2023

Display week: 1



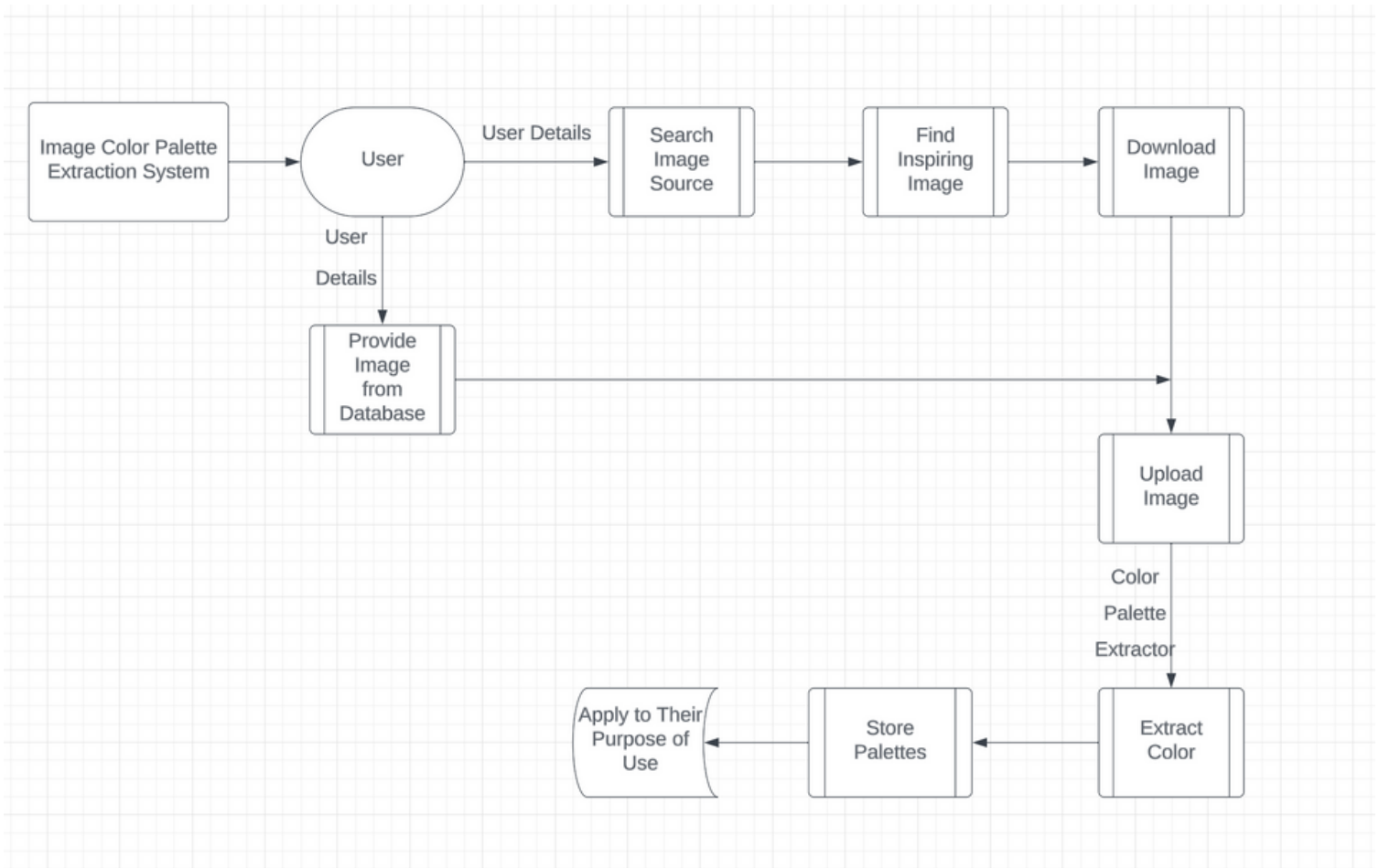
WIREFRAMES AND PROTOTYPES

Below you can find a screenshot of the time/text for the wireframe/prototypes. The screenshots do not include all prototypes/wireframes due to size. The rest can be found at this Figma link [here](#) or at this link for the pdf form [here](#).



DATA FLOW DIAGRAMS

Below you can see the screenshot of our dataflow diagram, which gives an overview of data coming into the system and the way it can flow throughout the processes and features we plan to have. The full file for the data flow diagram can be found in the GitHub repository inside the project plan files folder, or found [here](#).



CONCLUSION AND CREDITS

Below is a list of which of the parts of this document, and who worked on what.

- Project Overview: Anna
- SDLC Model: Taiga
- User Stories: Group
- Technology Stack: Taiga
- API Choices and Features: Group
- Work Breakdown Structure: Stefan
- Project Schedule/Timeline: Cindy
- Wireframes and Prototype Elements: Anna
- Data Flow diagrams: Cindy/Anna
- Report Structure: Stefan
- Presentation: Group



Contact

Stefan Pricope (scp10@sfu.ca)

Taiga Okuma (toa12@sfu.ca)

Cindy Xiao (yxa143@sfu.ca)

Anna Rusinova (ara103@sfu.ca)

GitHub Repository: [Linked Here](#)