



---

# Mark Allocation for Part 1 Project Plan

---

Marking Criteria	Does not meet the required standard	Meets the required standard	Partially exceeds the required standard	Greatly exceeds the required standard	Feedback
<b>Project Plan:</b>  <b>Realism and Achievability</b>  <b>[25 Marks]</b>	<ul style="list-style-type: none"><li>The project plan is unrealistic or lacks detail, with unclear tasks, dependencies, or timelines.</li></ul>	<ul style="list-style-type: none"><li>The project plan is somewhat realistic and achievable, outlining tasks, dependencies, and timeline with some clarity but lacking detail.</li></ul>	<ul style="list-style-type: none"><li>The project plan is realistic and achievable, providing clear tasks, dependencies, and timeline with sufficient detail.</li></ul>	<ul style="list-style-type: none"><li>The project plan is highly realistic and achievable, presenting clear, detailed tasks, dependencies, and timeline, demonstrating excellent planning skills.</li></ul>	Project plan has:  Tasks, dependencies and timeline, complete and realistic  Whole project Part 1,2,3.
	<b>0 – 12 Marks</b>	<b>13 - 18 Marks</b>	<b>19 - 22 Marks</b>	<b>23 - 25 Marks</b>	

# The Why Behind Jira: An Intro to Agile & Scrum



## The Old Way: The "Waterfall" Model

- Plan everything upfront (big design document).
- Work through phases one by one:
  - Design -> Build -> Test -> Release.
- No room for changes.
- You only see the final product at the very end.

## The New Way: The "Agile" Philosophy

- We accept that we can't know everything at the start.
- Plan and build in small, iterative cycles.
- Deliver working software frequently.
- Welcome and adapt to change.

---

# Choosing a Template

---

The choice of project template is yours just ensure that it can meet the rubric requirements

## Feedback

Project plan has:

Tasks, dependencies and  
timeline, complete and  
realistic

Whole project Part 1,2,3.

## Project templates

Made for you

Custom templates **ENTERPRISE**

Software development



### Kanban

Jira

Visualize and advance your project forward using work items on a powerful board.



### Scrum

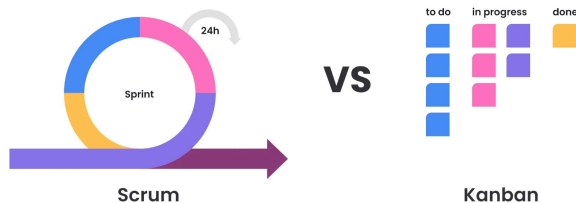
LAST CREATED

Jira

Sprint toward your project goals with a board, backlog, and timeline.



# What is the Main Goal of Each?



## Scrum is about Cadence & Planning

- **Goal:**
  - To deliver a predictable amount of work in a fixed time (a Sprint).
- **Rhythm:**
  - Works in iterative cycles (Sprints).
  - You plan, you execute, you review, you repeat.
- **Asks the question:**
  - "How much can we commit to finishing in the next two weeks?"

## Kanban is about Flow & Flexibility

- **Goal:**
  - To visualize your work, limit work-in-progress, and maximize efficiency.
- **Rhythm:**
  - A continuous, smooth flow of work. Tasks are pulled into the system as capacity permits.
- **Asks the question:**
  - "What is the most important thing to work on right now?"

---

# How the Boards Differ

---

	Scrum Board	Kanban Board
Purpose	Tracks work for a single Sprint.	Visualizes the team's entire workflow.
Columns	Simple, often "To Do," "In Progress," "Done."	Can be highly customized to match the actual steps in your process (e.g., "Backlog," "Design," "Development," "Testing," "Deployed").
Lifespan	The board is often cleared or reset after each Sprint.	The board is persistent and evolves over time. It's a continuous system.

---

---

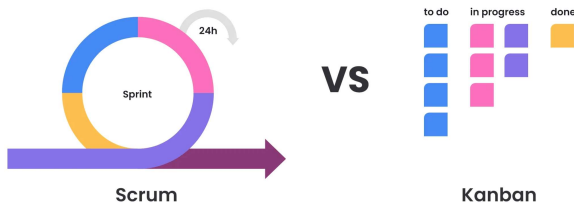
# Key Differences

---

Feature	Scrum	Kanban
Rhythm	Fixed-length Sprints (e.g., 2 weeks)	Continuous Flow
Roles	Prescribed Roles: <ul style="list-style-type: none"><li>• Product Owner</li><li>• Scrum Master</li><li>• Development Team</li></ul>	No Prescribed Roles.   The existing team uses the board to manage their work.
Key Metric	<b>Velocity:</b> How much work is completed per Sprint.	<b>Cycle Time:</b> How long it takes for one task to go from start to finish.
Changing Priorities	Changes are discouraged during a Sprint. New work is added in the next Sprint Planning.	Changes can be made at <b>any time</b> . You can easily re-prioritize the backlog.
Core Principle	<b>Time-boxing:</b> Commit to a chunk of work and deliver it.	<b>Work-in-Progress (WIP) Limits:</b> Don't start too many things at once.

---

# Which one Should You Use?



## Choose Scrum when:

- You are building a new product from scratch.
- You have a project with a clear goal that can be broken down into chunks.
- You need the structure of regular planning and review to keep a team focused.
- (Example: Developing V1 of your class prototype)

## Choose Kanban when:

- Your priorities change very frequently.
- You are in a maintenance, support, or operations role.
- The work is a continuous flow of incoming tasks of various sizes.
- (Example: A helpdesk managing incoming IT tickets, or a team fixing bugs as they are reported)

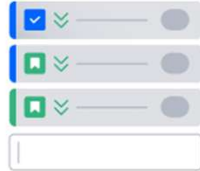


# Create a New Team-Managed Scrum Project

## Scrum

Use template

The Scrum template helps teams work together using sprints to break down large, complex projects into bite-sized pieces of value. Encourage your team to learn through incremental delivery, self-organize while working on a problem, and regularly reflect on their wins and losses to continuously improve.



### Plan upcoming work in a backlog


Prioritize and plan your team's work on the backlog. Break down work from your project timeline, and order work items so your team knows what to deliver first.

[Learn more about the backlog](#)

### Organize cycles of work into sprints

Sprints are short, time-boxed periods when a team collaborates to complete a set amount of customer value. Use sprints to drive incremental delivery, allow your team to ship high-quality work and deliver value faster.

[Learn more about sprints](#)




### Understand your team's velocity

Improve predictability on planning and delivery with out-of-the-box reports, including the sprint report and velocity chart. Empower your team to understand their capacity and iterate on their processes.

[Learn more about agile metrics](#)

Product






 Jira

Recommended for

Teams that deliver work on a regular cadence

DevOps teams that want to connect work across their tools

Work types

-  Epic
-  Story
-  Bug
-  Task
-  Sub-task

Workflow

- TO DO
- IN PROGRESS
- DONE

# Create a New Team-Managed Scrum Project

Feel free to change the key to something else.

The key will become the prefix on all your deliverables



## Add project details

Explore what's possible when you collaborate with your team. Edit project details anytime in project settings.

Required fields are marked with an asterisk \*

Name \*

PROG6212

Key \* ⓘ

PROG

Access \*

🔒 Private

Template

Change template



Scrum  
Jira

Sprint toward your project goals with a board, backlog, and timeline.

Type

Change type



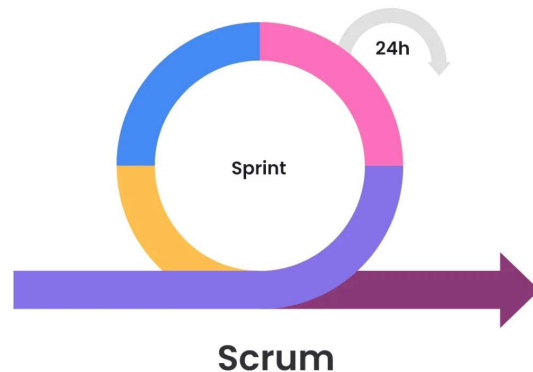
Team-managed

Control your own working processes and practices in a self-contained space.

Cancel

Create project

# Scrums and Sprints








## Scrum

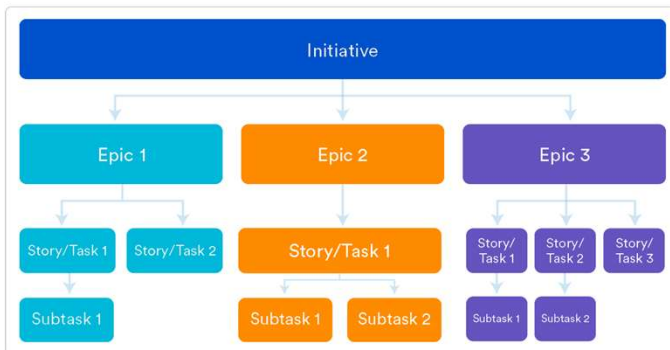
- Scrum is an agile project management framework that helps teams' structure and manage their work through a set of values, principles, and practices.
- Much like a rugby team training for the big game, scrum encourages teams to learn through experiences, self-organize while working on a problem, and reflect on their wins and losses to continuously improve
- Often thought of as an agile project management framework, scrum describes a set of meetings, tools, and roles that work in concert to help teams structure and manage their work.

## Scrum-Sprint

- A sprint is a short, time-boxed period when a scrum team works to complete a set amount of work.

# Work Types in Jira

-  Epic
-  Story
-  Bug
-  Task
-  Sub-task



## Epic

- Epics are large bodies of work that can be broken down into a number of smaller tasks (called stories).
- Epics are almost always delivered over a set of sprints.

## Story

- Stories, also called “user stories,” are short requirements or requests written from the perspective of an end user.
- Example: “Android users need to be linked to apple store.”

## Task

- Main objectives / features / goals to be completed

## Sub-task

- Sub objectives / features / goals to be completed

# Adding Tasks and Sub-Tasks

**Work**

**Sprints**

- ☐ ⚡ PROG-1 Part 1
- ☐ ⚡ PROG-3 Part 2
- ☒ ⚡ Part 3

☐ ⚡ PROG-1 Part 1 + ...

☐ ⚡ PROG-3 Part 2

☐ ⚡ PROG-5 Part 3

Create child work item

Adding a Task

⚡ PROG-1 / ☒ PROG-6

Package and Submit Assignment

+ ⚙

To Do ▾ ⚡ ⚡ Improve work item

Description

Add a description...

Subtasks

Add subtask

Connected work items

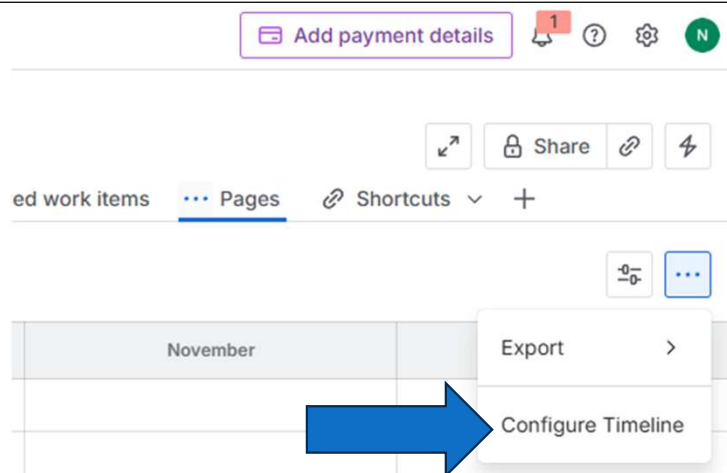
Add connected work item

Click on your task and this pop-up should appear.

Adding a Subtask

Just remember that Subtasks will not show on the timeline

# Editing the Dates for Tasks



For ease of use for this Project you can alter the following setting found in the top right corner of the Timeline sheet

Projects / SCRUM / Project settings

## Timeline

Configure the timeline view for your team.

### Child work item scheduling ☒

Show timeline bars for base-level work such as stories, tasks, or bugs on your timeline.

#### ☐ Schedule child issues by sprints

The schedule bars of child-level issues show on the timeline based on their sprint dates.

#### ☒ Schedule child issues by dates

The schedule bars of child-level issues show on the timeline based on their start and due dates.

This setting also impacts rolled-up dates

# Editing the Dates for Tasks

The screenshot shows a task management interface. At the top, a task bar for 'SCRM-3 Part 3' is visible, with a sub-task 'SCRM-8 Complete Project' marked 'TO DO'. A timeline is shown with a start date of 'Jan 02, 2026' and an end date of 'Jan 15, 2026 (14 days)'. Below this, a details panel for 'SCRM-3 / SCRM-8' is open, showing fields for Assignee (Unassigned), Labels (None), Parent (SCRM-3 Part 3), Due date (None), Team (None), Start date (None), Sprint (None), and Story point estimate (None). Two green arrows point from the 'Method 1' box to the timeline, and two red arrows point from the 'Method 2' box to the 'Due date' and 'Start date' fields in the details panel.

**Method 1:**  
You can hover over the Timeline in the same row as your task to add a timeline

**Method 2:**  
You can set the start and due date by configuring the task.

# Setting up Dependencies

Sprints

☐ ▼ ⚡ PROG-1 Part 1

☐ ☒ PROG-6 Package and Submit ... 🔗 TO DO

☐ ⚡ PROG-3 Part 2 + ...

☐ ⚡ PROG-5 Part 3

+ Create Epic

Create work item >

Move work item >

Change parent work item

Change work item color >

Edit sprint

Edit dates

Remove dates

Edit dependencies

Link to related work ×

Connect work items to indicate the order in which they need to be done.

This work item

⚡ PROG-3 Part 2

Link work item

Is blocked by ⊗ ▼

⚡ PROG-1 Part 1 ⊗ ▼

Link

Work

Just

September

October

November


Sprints

▼ ⚡ PROG-1 Part 1

☒ PROG-6 Package and Submit Assign. 🔗 TO DO

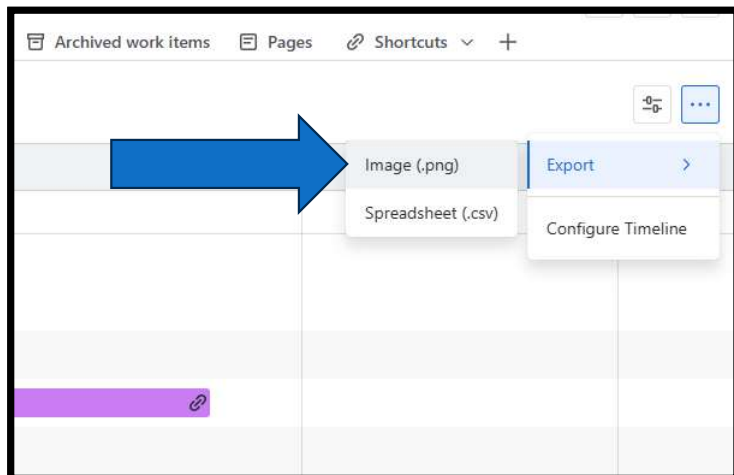
⚡ PROG-3 Part 2

⚡ PROG-5 Part 3

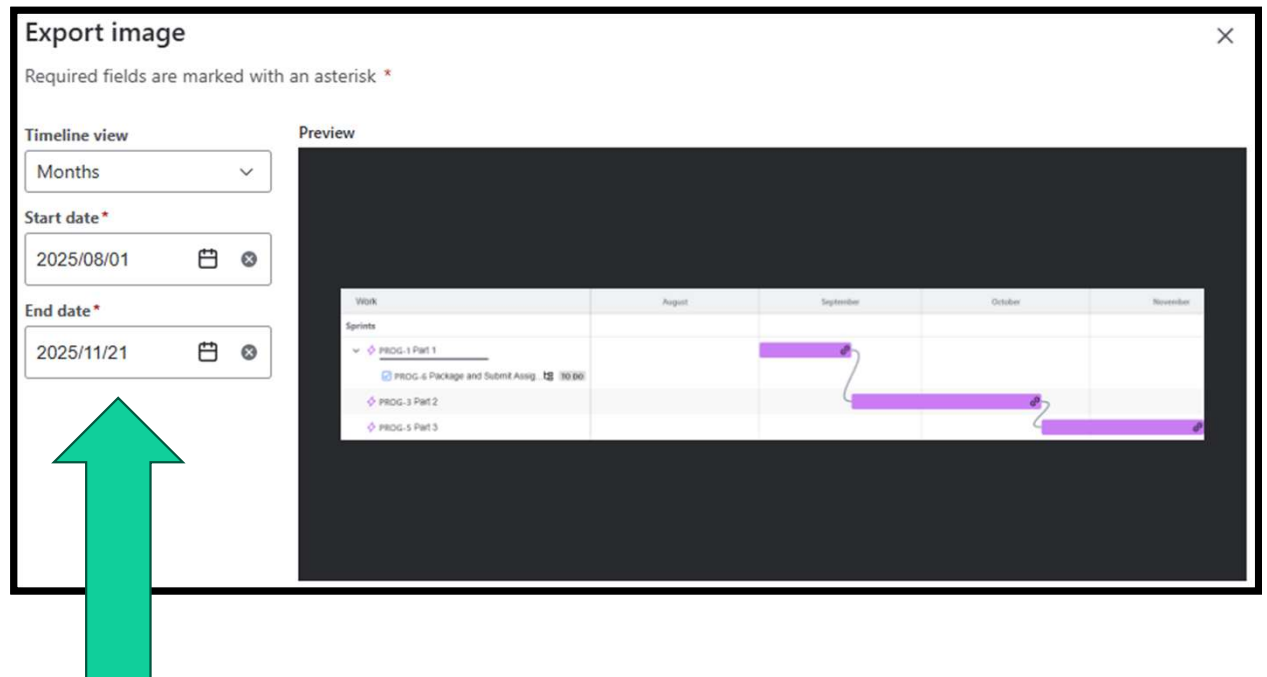


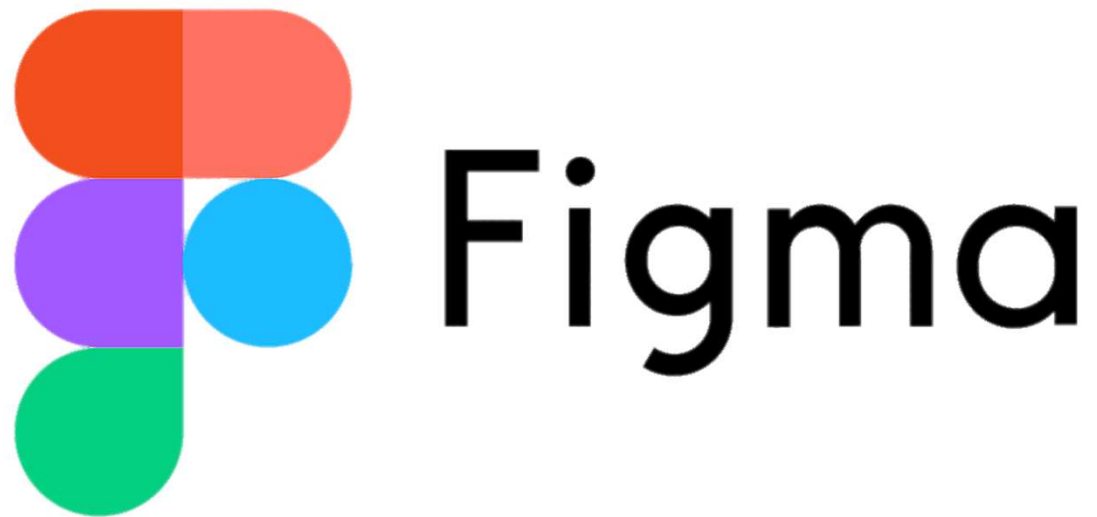


# Exporting as an Image for Submission



Make sure to change the Start & End Date along with the Timeline View





---

# Mark Allocation for Part 1

## GUI Design

---

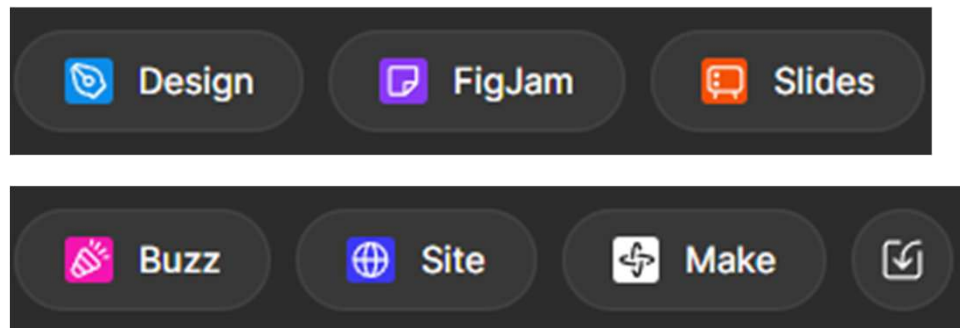
<b>GUI UI:</b>  <b>Design and User-Friendliness</b>  [25 Marks]	<ul style="list-style-type: none"><li>• The GUI design lacks user-friendliness and intuitiveness, with poor layout and usability.</li></ul>	<ul style="list-style-type: none"><li>• The GUI design is somewhat user-friendly and intuitive, with adequate layout and usability but room for improvement.</li></ul>	<ul style="list-style-type: none"><li>• The GUI design is user-friendly and intuitive, with good layout and usability.</li></ul>	<ul style="list-style-type: none"><li>• The GUI design is highly user-friendly and intuitive, with excellent layout and usability, exceeding expectations.</li></ul>	GUI: Has all requirements, layout is good and easy to use, colours / full design
	0 – 12 Marks	13 - 18 Marks	19 - 22 Marks	23 - 25 Marks	

# Getting Started with Figma

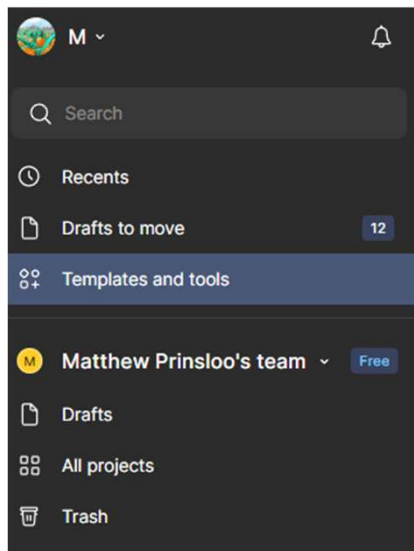


## Opening a New Design Tab

- In the top right-hand corner, you will find a set of buttons.
- You will click on a button to start up a new project
- The one you will be using is called "Design"

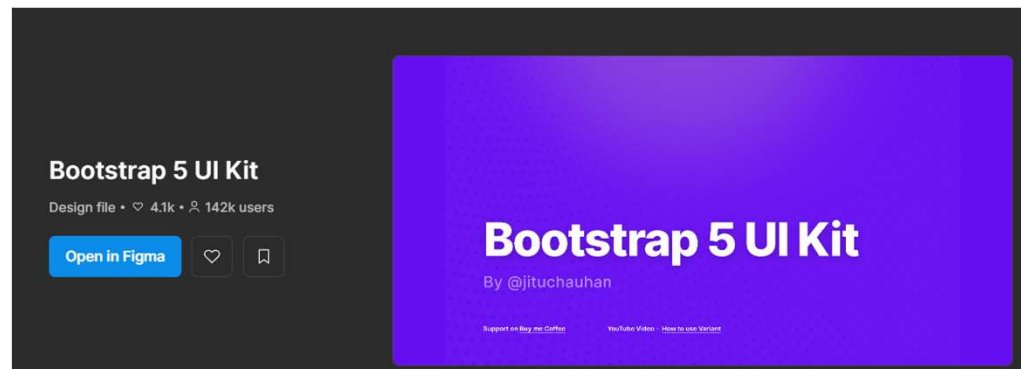


# Getting Started with Figma

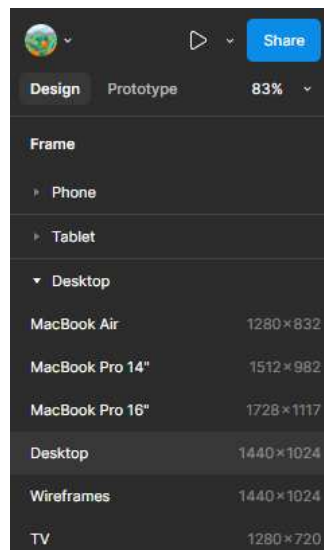


## Installing a Bootstrap 5 UI Kit

- To ensure that you have the same components you would find when building your app, we will need to import a UI Kit.
- To do so we first navigate to the community page.
- Then simply search for a free and well-maintained Bootstrap 5 UI Kit.



# Figma Frames



## Using a Frame

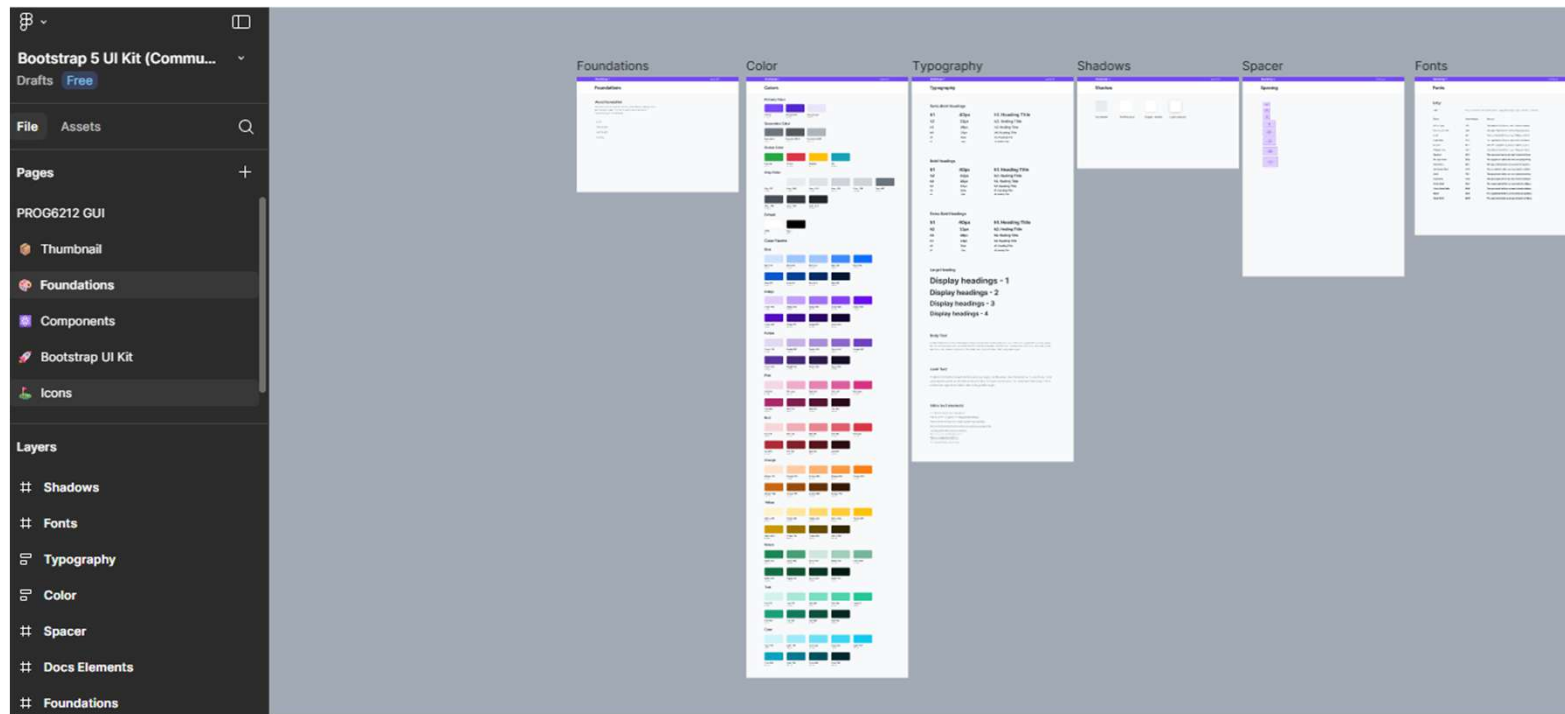
- Either press 5 or select the frame button at the bottom of the website



- The panel on the right should change - allowing you to select from a number of frames.
- For the POE you'll most likely use the Desktop or Wireframe Frame

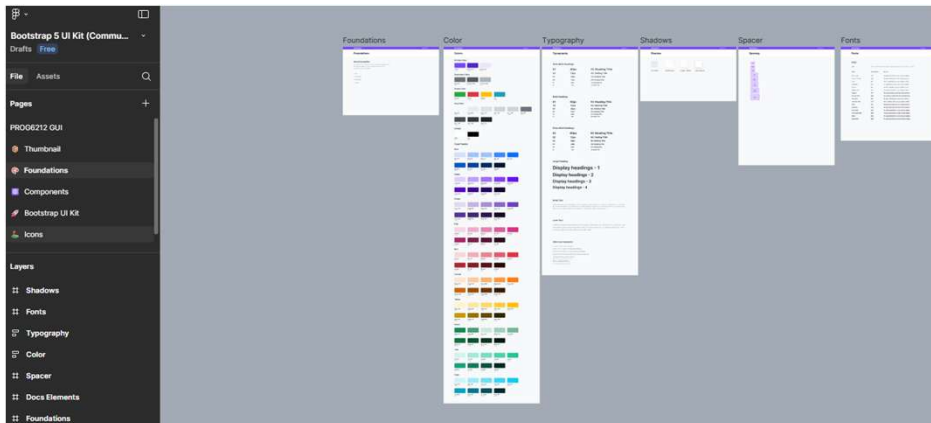
# Altering the Theme

Navigate to the Foundations page to get a better overview of your components



# Altering the Theme

Left click on the Empty space to access the colour styles seen on the right-hand side panel



Now you will be able to change the colour styles of your Figma Design and check how it influences your component / assets

