# Purpose of the solution

The purpose of this website is to address the problem of a lack of education or lack of confidence users of Micro:bits may have, by encouragement and an explanation of basic programming principles. This will include demonstrations, programming terms or information about topics. This website will need to be kid friendly as it will be a website for students and teachers, so it has to be upbeat and show micro:bit doing things to help them get enthused about it. The site will also have a section for teachers which will help them understand programming and also help teach their students enthusiastically. This website will focus on teachers and students of New Zealand.

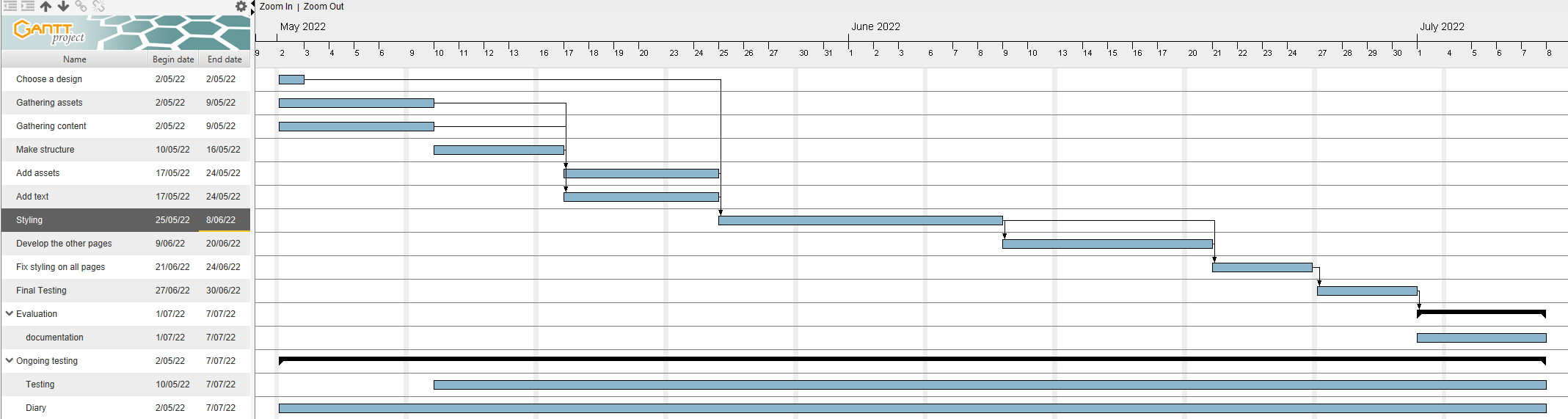
# End user requirement

This website has two very different types of end users. On one hand you have kids that grew up with computers and the internet around them. The other hand you have got teachers that are new to in-depth concepts of computers. This means in terms of content the teachers should have access to something like a computer dictionary, so they can understand terms and differences between things like data types. Their content should also include tutorials the teachers themselves can do for their own education, and to help benefit their students. I must also design content for the students. I will offer the same “dictionary” as there is always a possibility of someone not knowing something. There will also be lesson structures and tutorials that will be set for them, and extra stuff if they complete it faster than normal. For the interface it needs to be suitable for kids to navigate, so a balance of aesthetics and usability will be needed as one may be partially sacrificed to fulfil the goal of keeping the kids engaged. The friendly interface will also be needed for the teachers as they may also struggle with navigation, however it should be ok as long as I stick with the conventions of website design as teachers would have developed internet habits by now.

# Plan

## Planning tools

### Gantt chart



### Why did I use a Waterfall method over an agile method?

I used a waterfall method as It shows me a step by step process of everything I need to do before I can deem the website finished. Being able to see ever step decomposed into a timed list of things to is also easy to sustain as I am only one person, not a full team. This means I can readjust the plan to whatever I need without consequence. The waterfall methods I have gone with is a Gantt chart. The reason why I didn’t use an agile method such as Trello, flowcharts or Kanbans is because I personally like the waterfall method because I can see what must be done by specific dates giving me hard deadlines to ensure I complete them. It also lets me see what steps depends on previous steps being completed and it gives me an order to do them ensuring everything I need will be there on any given step. The agile method seems to require more time to use at its fullest. The agile methods work best in team situations that you get in industry where you need flexibility to take advantage of new possibilities, and you are working often with much bigger projects. The agile methods let the team leader move the project as needed in a way that I do not need to. I do not need a very flexible plan or planning system as I am working alone and don’t have to worry about others, so the stricter structure of the Gantt chart guides me well.

### Why did I use Gantt chart?

I used Gantt chart over other waterfall panning software because I have used in since year 9, so about five years. This means I am very comfortable with the software making laying out plans very simple and easy for me as I know where you find everything in Gantt charts. I also find Gantt charts very easy to read as it not cluttered and I am used to the software. Because of this I felt Gantt charts were my best options for brainstorming and laying out a plan for the development of my website.

## Problem decomposition

### Why did I use this order of steps?

I used this order as I felt it was the most optimal order to complete the steps as this order should have no back tracking and every step should have the appropriate resources to finish it.

1. Gathering Assets and Content: The gathering of Assets is important as I need images as they were a part of the design and they are a key element for the content as it can draw viewers in and inspire them to do something with the website’s information. The same goes for gathering content. I need to have something to talk about and having extra research about the website’s topic is never bad. This will be done first as knowing all the fact will make writing the content easier rather then doing it in the middle of development. Another reason of it being first is because its requirements to do, so I can get it out of the way at the start.
   1. Acquire content.
      1. Find articles and other websites to research and understand micro:bits more in depth.
      2. Gather statistics and opinions to talk about.
      3. Experiment personally with micro:bits to get an understanding of them
   2. Acquire pictures.
      1. Find and pictures and add them to the asset table.
      2. Save original image in case edited versions need to be removed.
      3. Crop, resize, tinting and other types of image editing will be used for each image.
      4. Save in jpg and png. Jpg will be compressed to 7 for optimal quality, and png will only be used if transparency is needed.
2. Make the structure: Making the structure will be the first step in coding the website. I will do this first because its layout outline and boxes of where everything on the website will go, it will also make styling and adding content a lot easier. This is because if I added objects at the start as a part if the structure, they will just be empty containers for content and styling.
   1. Make index page.
      1. Create the main structure of the page using both HTML and CSS.
      2. Add placeholders for images and text, that reserve space for slideshows and content.
3. Add the Assets and Content: This step is where you will insert the images and write the content. You would do this after the structure because all I would have to do is replace the placeholders with the real images and content making this step much faster. This is also required for the rest of the steps to continue.
   1. Add the actual content.
      1. Add images and a fully functional slideshow.
      2. Add all text content in.
      3. Must be spell checked and tested.
4. Style the page: Stylization of the page is the next step and will most likely take the longest as many things can go wrong while getting things to be sized and coloured correctly. The reason why we do this after the content was added is because we can see what the final product will roughly look like with all the images and text in the correct places. This step happens now as all steps before it is needed for it to be started and it is the ground work for the next step.
   1. Stylize the page.
      1. Add personal colours.
      2. Edit text size and font style.
      3. Add colour functions e.g. Dark mode.
      4. Create template for other pages to follow.
5. Develop the other pages: The development of other pages is the next task and will require the use of the last stylesheet and is the reason why the first page must be fully finished before the rest are started. This step should not be long as the first page can be used as a template. All I would need to do is change some of the layout and content which will be short, however it will take time as I will need to make four other pages that fully work.
   1. Develop other pages.
      1. Change some of the layout to suit the page better.
      2. Add content e.g. images and text.
      3. And functional features (Slideshow and Dark mode).
      4. Make links between all the pages.
6. Fix styling on all the pages: This step should take a small amount of time as all you would be doing is readjustments to the stylesheets to make sure everything fits correctly and looks nice. This needs all previous steps to be completed to start.
   1. Skim through all the page looking for issues.
      1. Fix some incorrect sizing.
      2. Incorrect colours.
7. Do final testing: This step is very important and crucial to the success to the website as this step will be all about testing and double checking everything in the website. This is where you will use a test plan to test functions and features of the page to ensure that they work fully. You will also need to proof read everything as well.
   1. Make a test plan
      1. Test all JavaScript based features to see if the work.
      2. Spell check all text.
      3. Make sure every image is in the correct place.
      4. Fix or finish any leftover features.
8. Documentation: Once everything else Is done a document will be made and used to evaluate the website. It will include a summery on how it went, how well it works and what improvements can be made.

Ongoing - Diary and testing: This will be done throughout the entire process as ongoing testing is good to do to ensure sections work and small thing don’t slip through the cracks when I am developing the page. The Diary will be used to Document development as it goes. It will reveal what things were changed, it will show feedback and a weekly update on what the status of the website is. It will also show screenshots of code and the actual page to fully show what has happened each week.