# Implementation Details

### How is the Gantt displayed?

The data to be displayed in the Gantt is either stored through the sampleChart function or the userDefineChart function.

The Gantt is then displayed using a function called printChart**. This is a function created in the** additionalFunctions.c **file, which takes the struct array and the number of total tasks as arguments. This function prints the Gantt row by row. It prints the months by using the function** getMonth(x). getMonth(x) **consists of an** enum**. It takes in an integer and prints the corresponding month. After printing the first line of the Gantt, nested** for **and** while **loops are then used to display the rest of the Gantt chart.**

### How to map the task to struct?

In the “editChart” function it maps user input data to a “features” struct by using “fgets” and “sscanf” functions to read user input and populate the fields of the corresponding features “struct” at the specified index in the “ganttInput” array.

### How did you edit and change tasks?

In order to edit and change tasks, we created a function called editChart. After receiving the name of the task which the user would like to change, a for loop is used to find the position number of this task in the struct array. This number is then used to access the corresponding details of this task, in order to edit them.

### How did you implement search for a circular dependency?

To test for a circular dependency the use of a DFS algorithm was used. DFS stands for Depth First Search. At each step of the algorithm it chooses the next unvisited task and adds it to the stack. It then explores the other tasks that are around the main task in question by recursively calling the DFS function on each unvisited task. When it reaches a task that has no unvisited tasks, it backtracks by removing the main task from the stack and continuing to explore the graph from the previous task. DFS can be used to solve a variety of problems on graphs, such as finding connected components, detecting cycles, and searching for a path between two vertices. In this case it was detecting dependencies that would break the code by turning it into a loop unintentionaly.

### What’s your Ascii art about?

The Ascii art in the project is about what me and my project partner like. So one of us enjoys gaming and the other enjoys music.

### Git Repository:

A git Repository was used to upload code and update it which was perfect when working in a group on one project. The link to this projects Repository is:

https://csgitlab.ucd.ie/Sneha-Chhipa/assignment-2-draft2.git