

Project document

## CFG Student Forum

Amina, Ayisha, Elham, Samantha and Victoria

### INTRODUCTION

Our specialisation project we have developed is a web app for CFG nano degree students. This space is made for students (by students!) and can be utilised on a number of different levels. Our web app will have a number of different features that can be used for effective learning to ensure that students are getting the most while completing the course with Code First Girls. Features of this web app include the following: a login page, forum, study zone, resources page and a user profile. Details of each feature can be found in Specifications and Design.

As CFG Nano degree students we have first hand experience of how much resources, content and learning we have to get through- this may be true no matter what our background is, whether we come from a career in tech wanting to level up our skills or if we are complete newbies and are career changers. **The main objective central to our project is to make the journey through the nano degree as easy as possible.** This can be broken down into the following targets we aim to meet:

- **Functioning Forum** - A space devoted for CFG students where they are able to reach out to other students and use each other as a learning resource. This space fosters socialising amongst students where people are able to learn together in between classes. Although things like stack overflow exist, it is not the most friendly environment for beginners and can prevent engagement. Our forum aims to bring specifically students together where they feel they can openly and easily ask questions that are specific to CFG content. We believe this will be beneficial to students as it not only brings students together but allows a space for relevant discussion and of course support.
- **One space for all resources**- There is a lot of content within the nano degree that we get through both in and out of lessons to help us get the best possible grades and to help us learn as much as possible. Our resource space acts as an organised database that students could go to find all the resources that they needed to support them in this course. The resource area is where CFG admins/students can upload authorised study materials that are relevant to their course (software/data/full stack). This will support students to be more readily prepared for their class and put them in the best position to learn.
- **Creating a focused study zone**- We wanted to go the extra step to ensure that students have a space that encourages efficient learning. This space will 'lock' students in and away from distractions and have features such as a pomodoro timer to promote student productivity.

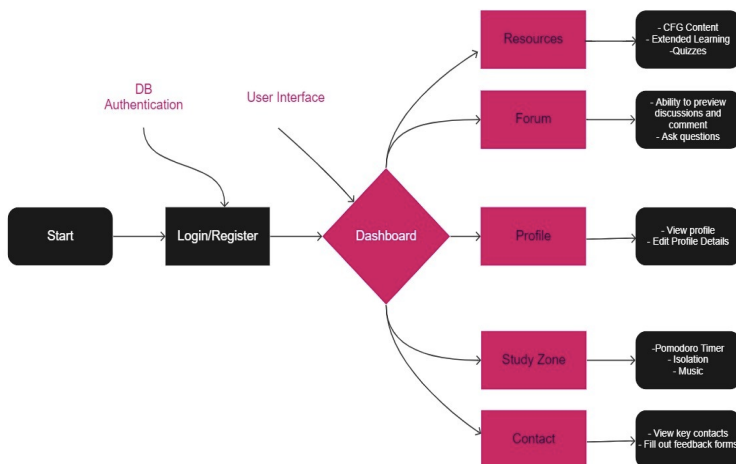
## BACKGROUND

During the development of the web application we had highlighted a number of features as a team that we needed for our project to function effectively. In addition to this we also kept a note of supplementary features that we could add once we had achieved basic functionality. The team used MoSCoW boards to look at how to prioritise between each feature on each of the key pages for the app.

One area that was broadly discussed was 'who the app was for', this meant looking at if our project was for the general public and beginners in tech or specifically CFG students. For the purposes of this project we felt it would be more efficient to tailor this project towards the CFG syllabus and culture as this is what we had the best insight into. Despite this the team did consider what our project would look like if it was non-CFG related, and also prospects of scalability if our project were to expand. As we decided to go with the CFG web app one of the first key elements was the functionality of our login page and determining how we would grant access to our web space. This is where the production of the log in/ register page comes in where only students who have authorised email addresses would be able to log in or register.

## SPECIFICATIONS AND DESIGN

We based our specifications and design of our project around ease of use and functionality for the forums users, we blueprinted software architecture that outlined the flow of how our project would interact with users and what user navigation would look like. During this stage of software design we chose the key elements that would create our project, this included the different pages that a prospective CFG student would have access to and how these pages would relate to each other. For example, how a user's profile would be able to be used when creating posts on the forum page, this was key to the functionality of our project. In addition to this, we discussed back end logic such as the use of a SQL database to hold user information to ensure that only authorised and registered users would be able to access this site.



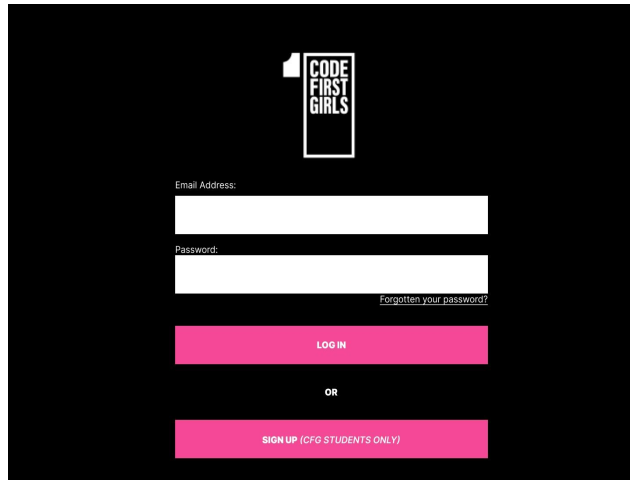
### ← Software architecture design

In this diagram we identified each webpage for our app, and their key features.

To ensure that what we wanted to achieve was possible we researched resources we would be able to utilise to complete our project to a high standard. We looked into using Django as a Python based web framework which was a fit for our needs/ requirements of the project. Other software logic discussed includes MySQL and Bootstrap. See appendix 1 for our database structure.

## Design Templates

As our project was based on the Code First Girls, and specifically developed with nano-degree students in mind the design was important in that it had to continue consistency with already existing CFG branding. We were able to create proposed designs for how our web app could look like using Figma, examples of this are below:



CODE FIRST GIRLS

Email Address:

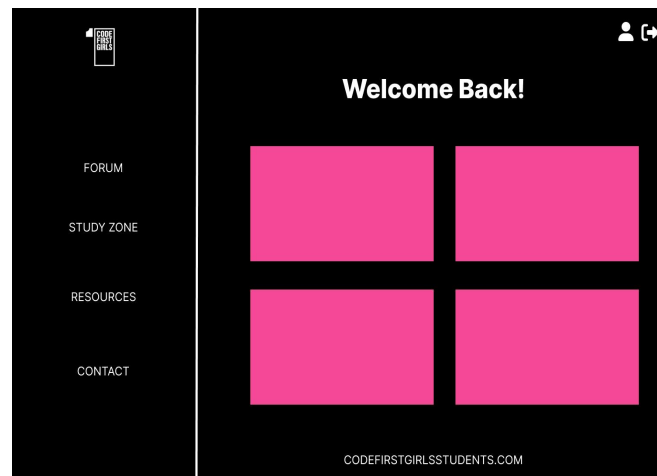
Password:

[Forgotten your password?](#)

LOG IN

OR

SIGN UP (CFG STUDENTS ONLY)



CODE FIRST GIRLS

FORUM

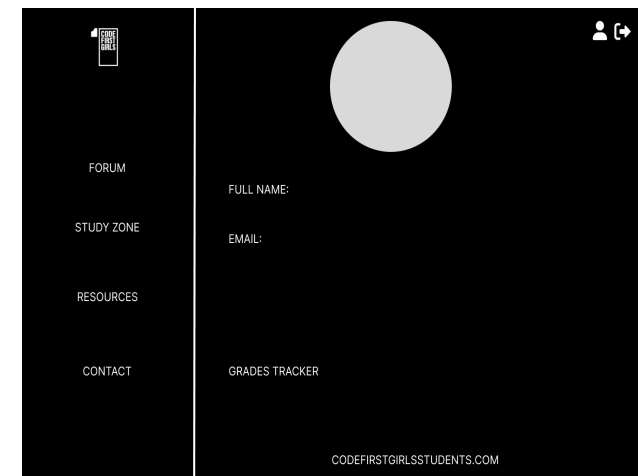
STUDY ZONE

RESOURCES

CONTACT

## Welcome Back!

CODEFIRSTGIRLSSTUDENTS.COM



CODE FIRST GIRLS

FORUM

STUDY ZONE

RESOURCES

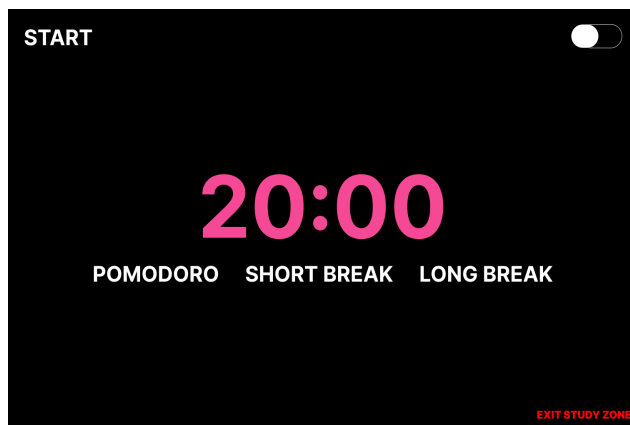
CONTACT

FULL NAME:

EMAIL:

GRADES TRACKER

CODEFIRSTGIRLSSTUDENTS.COM

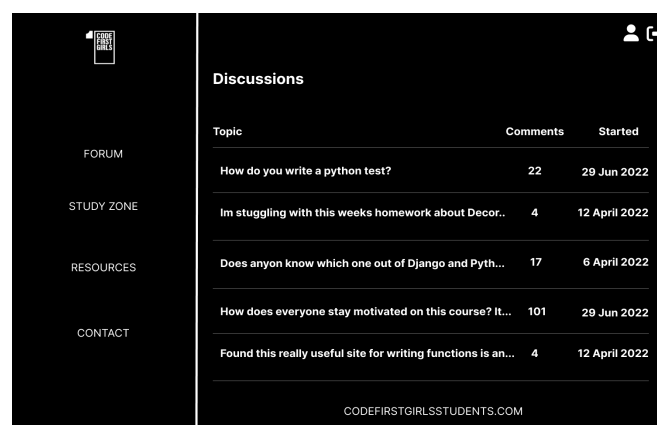


START

20:00

POMODORO SHORT BREAK LONG BREAK

EXIT STUDY ZONE



CODE FIRST GIRLS

FORUM

STUDY ZONE

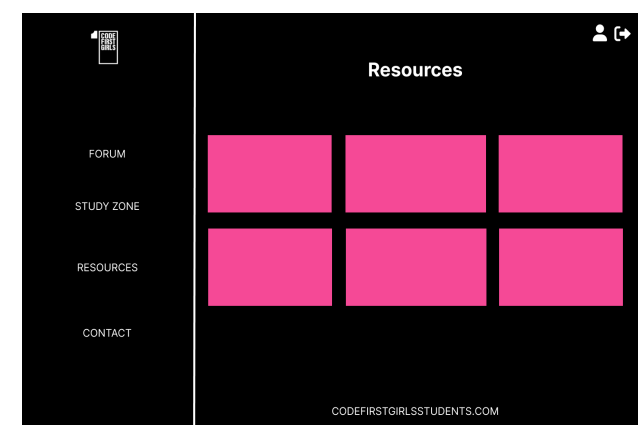
RESOURCES

CONTACT

### Discussions

| Topic  | Comments | Started       |
|--|----------|---------------|
| How do you write a python test?                              | 22       | 29 Jun 2022   |
| Im struggling with this weeks homework about Decor..         | 4        | 12 April 2022 |
| Does anyone know which one out of Django and Pyth...         | 17       | 6 April 2022  |
| How does everyone stay motivated on this course? It...       | 101      | 29 Jun 2022   |
| Found this really useful site for writing functions is an... | 4        | 12 April 2022 |

CODEFIRSTGIRLSSTUDENTS.COM



CODE FIRST GIRLS

FORUM

STUDY ZONE

RESOURCES

CONTACT

### Resources

CODEFIRSTGIRLSSTUDENTS.COM

## User Stories

To support the development of each of each feature on our web application we used User Stories to corroborate what each page is responsible for.

When brainstorming what we wanted to collaboratively create for our project we covered a number of potential features our site could have. This included additional features such as a personal grade tracker which would belong with the users 'Profile' area. Articulation of user stories enabled us to highlight other potential features that would benefit users (students and CFG admins). With User Stories and the use of our MoSCoW boards we were able to determine and assess importance around each feature and prioritise accordingly.

| Requirement              | User Story   | Importance |
|--------------------------|--|------------|
| Must Have a Login System | Sasha must be able to log in with the same email address that is used to enroll on any CFG Course                                      | HIGH       |
| Study Zone               | Laura must be able to escape from the world in her study bubble where she has access to a pomodoro timer and music                     | HIGH       |
| Forum                    | Rachel must be able to view, topic/discussions shared from other users as well as add and delete her own comments                      | HIGH       |
| Contact/Feedback Form    | Michelle can give feedback on lessons, coaches, suggestions, complaints etc as well as the key contacts on their course/cohort         | HIGH       |
| Resources                | Becca must have access to this one stop shop of resources which cover a range of different topics that will help her along her courses | HIGH       |
| Grades Tracker           | Rhianne can use this tracker to keep track of her grades if she chooses to, just so they are all in one place                          | MEDIUM     |
| Forum Interaction        | Lisa can like and save posts she finds useful  | MEDIUM     |

## IMPLEMENTATION AND EXECUTION

Once we established good guidelines for the design and specification of our CFG student space we broke this down into tangible features and requirements for each of our functioning pages. In the table below we worked in sub teams to come up with proposed requirements and agreed as a team on what we would keep to work with the wider site.

The team worked within an Agile Framework, so it should be noted here that throughout this process we maintained flexibility on aspects of our project. Fundamentally as a team we aimed for basic functionality, firstly with the forum and then with our study zone and resource area. With the latter however we were able to remain open on how they would want them to work. For example, whether the Resource Area was able to accept resources from CFG admin only or if students could submit helpful items also. We were able to adapt to what was feasible and make our software and data structures work for us in this way- luckily with this element we were able to achieve the full capacity of this so that CFG admins can authorise student submitted resources therefore making our web-page more user-centred and effective for student experiences.

## Page Breakdown

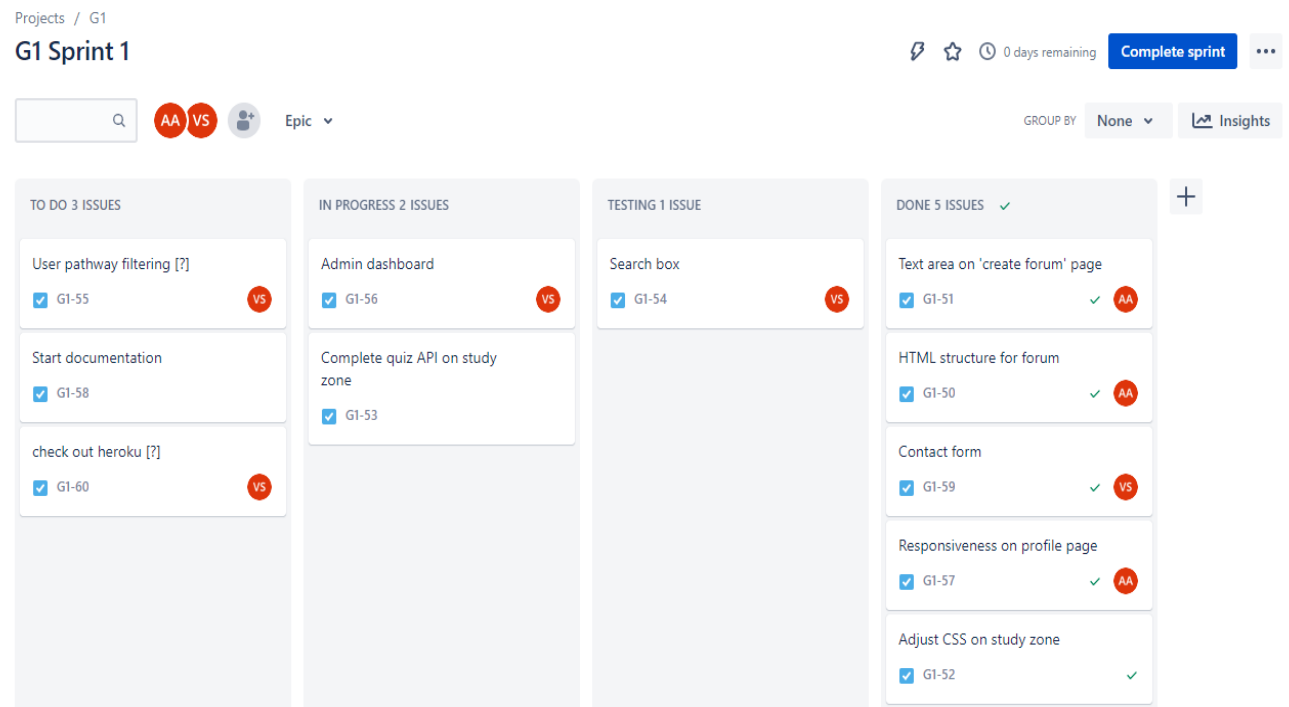
| Features                         | Requirements  |
|----------------------------------|---|
| <b>Log-in/ Registration Page</b> | <ul style="list-style-type: none"> <li>This is the opening page of our app. It is only accessible to CFG users.</li> <li>New users able to register to access site</li> </ul>   |
| <b>Forum</b>                     | <ul style="list-style-type: none"> <li>A space where CFG students are able to openly ask questions. This could be about a particular lesson, wider course experiences or to seek general technical guidance.</li> <li>Users will be able to ask and respond to relevant posts. If users have nothing to post they will be able to browse discussions that are happening in the forum and can also be able to like comments. This will support students in easily accessing relevant solutions to problems they may have</li> <li>Able to 'like' posts/comments that are helpful</li> <li>Potential anonymity if students want to hide identity to post a question, good for shy users.</li> <li>Django framework to create forum</li> </ul>   |
| <b>Study Zone</b>                | <ul style="list-style-type: none"> <li>A space students can go to when they need to get focused.</li> <li>Area contains a pomodoro timer and a selection of focusing sounds (through Youtube API)</li> <li>Area 'locks' user in without any distractions</li> <li>Light and dark mode available- accessibility feature</li> <li>Bonus: Inspiration quotes to keep up revision motivation (API)</li> </ul>   |
| <b>Resources</b>                 | <ul style="list-style-type: none"> <li>A collection of readily available study materials. These materials may be used directly in classes (presentations/ starter code), but may also be extra reading resources for further learning between classes and assessment support such as cheat sheets. This area is overseen and authenticated by CFG.</li> <li>Students able to submit helpful study prompts in the form of flashcards, these are authenticated and approved by CFG admin</li> <li>Students can create personal notes on classes and save them to their profile. These notes can be edited later and the entire personal search history can be searched through</li> <li>Admin Dashboard- table of all available resources, link to submission page and table filtering. Resources available through organised data filing.</li> </ul> |
| <b>Profile</b>                   | <ul style="list-style-type: none"> <li>The user base for our web application. Students will be able to view and change personal details.</li> <li>Able to upload personal photo</li> <li>Students can also view their grades here</li> <li>**Linked to Forum page- user profile name should be seen when posting or commenting</li> </ul>   |

## Agile Working

We aimed to work in a SCRUM methodology, so we would divide each work phase into a separate Sprint. Every few days we hold a team briefing, which equates to the first step: Sprint Planning. We would clarify what needs to be achieved over that period, divide the work and collaborate during the planning stage. Although we were not able to hold a Daily Scrum, we were continuously in conversation over Slack regarding any questions or setbacks we were facing. In addition to this, we were able to catch up in our sub-teams between briefings to work collectively to meet our targets.

During team briefings each member/ sub-team was able to give feedback on what they had been working on. Our 'sub-teams' were flexible in that individuals chose their area of interest based on strengths, and more support was provided for larger/ complex project spaces. In team meetings we were able to come together to showcase any progress, this was crucial throughout our sprint development as the team shared tips on what was working well or not. Additionally, we could collaboratively come together to visualise what each aspect of the web page would look like when it came together. We were also able to review code and different branches of work through our git repository. As a team, we made decisions to maintain consistency throughout the web app and all of the pages and features we created.

We used JIRA to log all of our ongoing tasks and progress on our Sprints, through this the team had an overarching area to follow comprehensive developments of our projects. Tasks in these sprints were addressed during our team briefings so we could uncover any issues and work together on finding solutions to any errors.



## TESTING AND EVALUATION

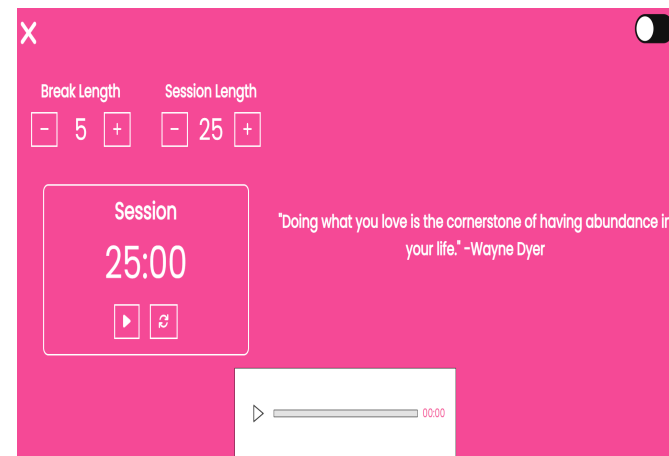
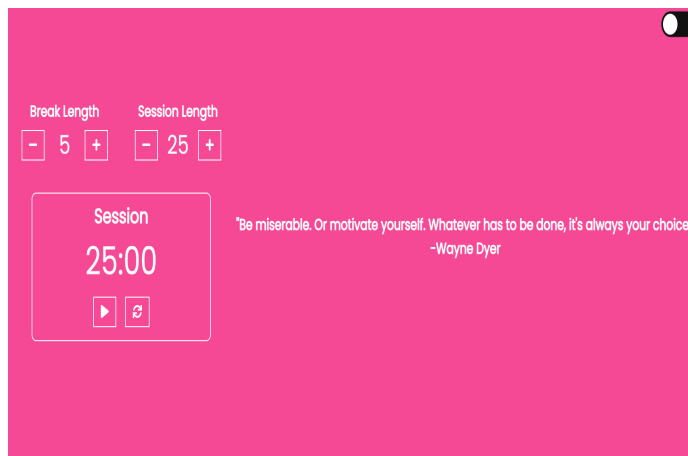
### Testing Strategy

Throughout the development of our project we have been testing to ensure that it works across different operating systems. We have tested this when using github to pull or merge different branches of code that are responsible for different pages. By doing this we have been able to assess how well data has been able to migrate across our different machines and troubleshoot any arising issues. A lot of preparation did go into this as we initially had a number of branches that correlated with each of our features, as a number of these features overlapped with one another it was crucial that the hard code was working for everyone with minimal changes as this may impact other users functionality.

### Functional and User Testing

To minimise the risks of any errors or negative influence on the software we would begin with merging similar branches together. After this we would be able to address any issues with the code and troubleshoot these together as a team during meetings. We had noticed particular issues in Python when working with Django and its consistency between Mac/ Windows operating systems. The issue here was based on Mac OS having challenges with not installing Django correctly, as % of our team were Mac users this was problematic however we were able to troubleshoot this together by installing Brew into our Pycharm which eventually allowed us to use Django effectively.

An example of our testing can be found in our study zone, this page was created with the aim to 'lock' a user in to encourage focus and learning. With this, our intention was to have a small 'X' on the top left corner of the page to allow users to exit without having to close the browser. We had found that when some of us ran the server we did not get this expected feature on this page (pictured left). We found that in this page we had some rendering issues with differences in layout. After working together, we were able to identify this as an issue with our browser and not within our code, we were able to fix (pictured right) this issue by making adjustments related to our individual browsers to ensure that all users were able to see the correct version of our project.



```

class LogInTest(TestCase):
    def setUp(self):
        self.credentials = {
            'username': 'testuser',
            'password': 'secret'
        }
        User.objects.create_user(**self.credentials)

    def test_login(self):
        # send login data
        response = self.client.post('/login/', self.credentials, follow=True)
        # should be logged in now
        self.assertTrue(response.context['user'].is_active)

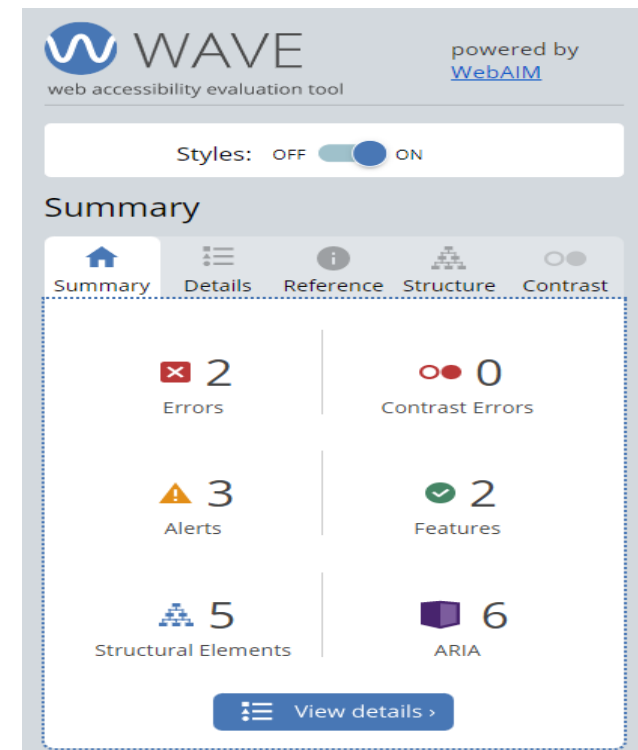
```

← Test cases

We created test cases which assessed how well users were able to log in successfully. In this example we were able to directly test the functionality in one of our key elements which was the ability to log in as a CFG student. This worked effectively as we were able to log in as users to use this site as evidenced in this test case.

## Accessibility Checks →

We used the Web Accessibility Evaluation Tool to assess how accessible our web application is, this was really important as accessibility features are key in being inclusive. This tool concluded a total of 2 errors which we believed was a good score and this score was similar throughout each page on the site.. On further examination we believe that these errors were down to logos/ images that we used which were unavoidable during the course of this project. This is something that could be looked into further in the future development of this project.





## Future Development

One way we could have further tested our server is through the use of a cloud application platform called Heroku. This tool would have allowed us to share our server publicly, similar to having a beta version that we would run and gather feedback from. This would have given us valuable (and unbiased!) insight into the functionality and efficiency of our web application. Through this tool we could also see how our app would work in real life and test how strong our code has worked. Unfortunately, we did not have enough time to utilise this properly as there were some teething issues. When testing to see if we could upload onto Heroku, there were some challenges with this service correctly recognising the language we were using and also identifying where files were placed. As we were at the end of our dev cycle it would have been inefficient to move files around as this would have affected our software performance.

We would also expand on the social media side of things and create profiles where users are able to follow each other and connect via a chat messaging service.. We believe this would give further support to CFG students and allow them to create more meaningful relationships with their cohort but also branch out and converse/meet other students from different cohorts. Through this CFG alumni would be able to connect and support each other on future professional endeavours.

## CONCLUSION

Our team was able to come together to create an exciting and practical application with this project. We identified what could have made our time with CFG even better and thought outside the box for what could be helpful for instructors across the different streams and cohorts. We were able to divide the workload as a team and successfully develop the different pages of our web application as well as all of their individual features. When we came across challenges in our code we addressed this in our team meetings to resolve any discrepancies to ensure that our project was consistent across the site.

As a team we were able to use our learning from the course to make sure our project was running at its full potential, this includes working within an agile framework, implementation of API's , operating github correctly and of course patience building. This in turn, allowed us to produce a functioning fullstack web page which has given us the skills that will most likely be needed within our new careers as software engineers!

## Appendix 1

