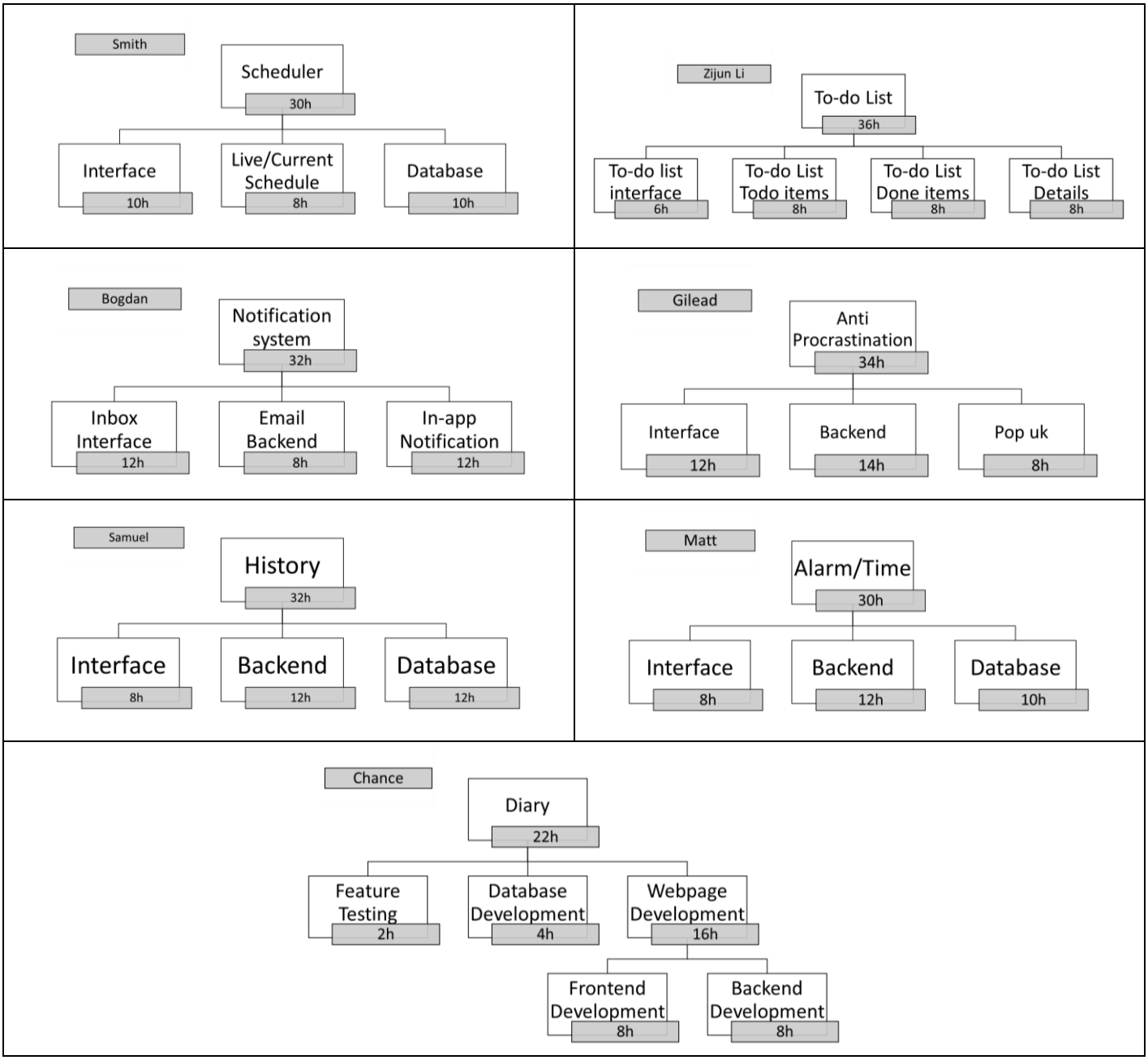
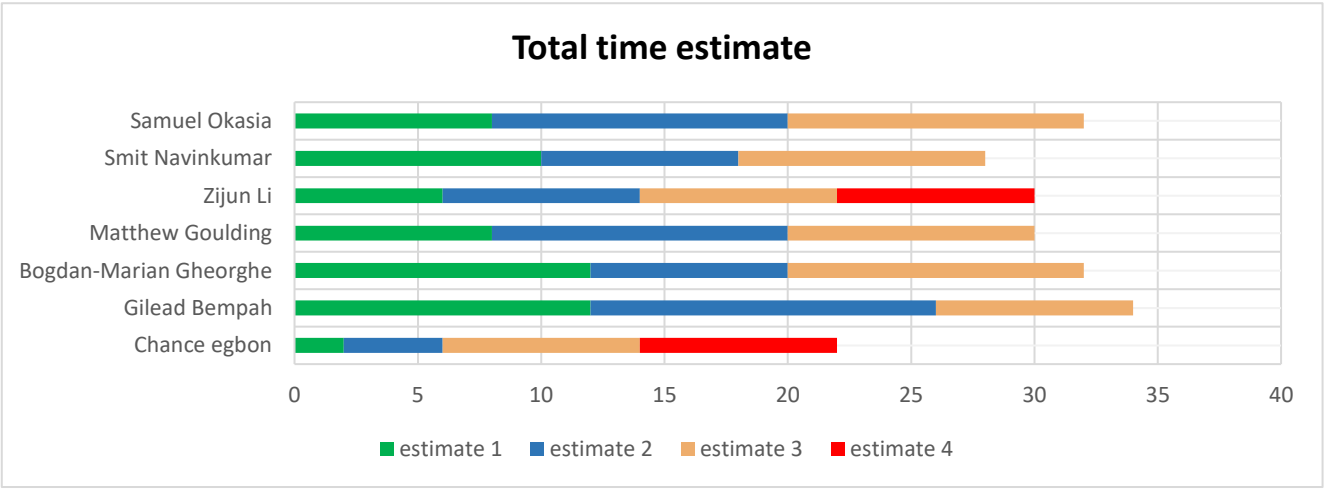
Chance Egbon 2194210

S2: MVP - Individual Submission

Chance Egbon 2194210

Overall agile estimation of feature cards



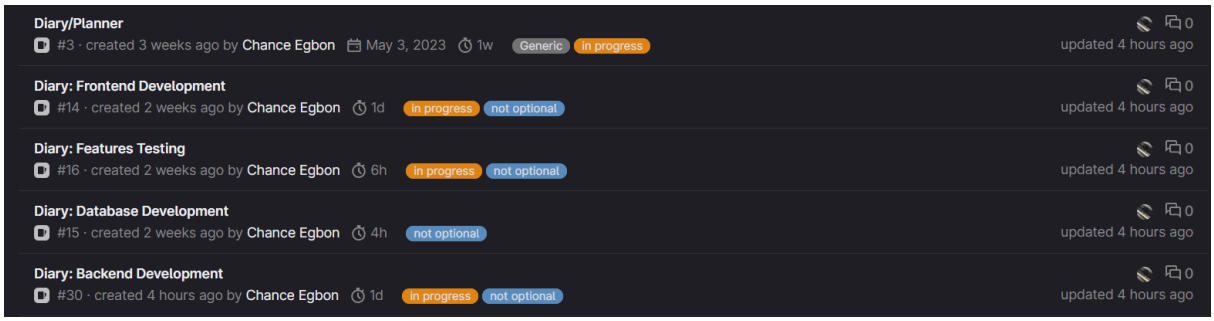


Chance Egbon 2194210

Agile estimation of feature cards

Diary agile cards

- **Diary: Interface Development (8 hours)**
  This task involves implementing the interface of the Diary page. Through the use of HTML and CSS, I will develop what the user sees when making use of the diary feature on the web application. These are the graphical and interactive aspects of the Diary webpage and it includes the navigation linking all the webpages, the title of the page and various texts, and the diary section where the user will able to create, edit and delete daily annotations. I estimated that it might take me a minimum of 8 hours to complete this task because I will need to refresh my HTML and CSS knowledge, implement responsive features to my page, and understand how to position all the graphical elements in a coherent and eye-pleasing way and implement more interactive features using JavaScript.
- **Diary: Backend Development (8 hours)**
  The back end of the Diary feature represents a portion of the webpage that is hidden from the user. I estimated 8 hours for this task because I will have to implement what is needed to store and organise the data inputted by the user and also make sure that every single issue on the client side is dealt with.
- **Diary: Database Development (4 hours)**
  I estimate 4 hours for this task as I will have to design and develop my webpage database and merge it with the others group member's databases.
- **Diary: Testing (2 hours)**
  I estimated 2 hours for the testing task. This task will involve testing all the graphical and interactive aspects of the interface, all the operations running on the page background and testing how the data is stored in the database.
- **Diary/Planner (22 t hours total)**
  This card represents the total amount of time I will spend developing the various sections of the diary feature



Chance Egbon 2194210

Web application testing

Traditional applications are being simplified migrating to web-based interfaces. To successfully test web applications it's important to understand what is testing and why it's beneficial.

For web applications to be of high standards they must work on different operating systems and devices such as mobile phones and browsers and browsers versions. As there are many types of devices and browsers-OS combinations testing a web page to ensure its usability and functionality are preserved in most scenarios is essential.

This software practice is a crucial stage of software development and essential to find bugs. Depending on when and how often practice is carried on bugs can be found could be found prior to a release or on a daily basis. Bugs can appear any time there is a change in your software even if the change is very small. Finding and fixing these bugs as soon as possible is important because fixing bugs at a later date can be more costly and bugs might be used as a way to abuse the system.

Intense testing is vital to create a first-class web application. As manually carrying out functionality testing on a web application can be tedious, QA teams make use of automated testing to build quicker, more efficient and robust test cases. Automated testing makes use of human and machine-driven routine and repetitive testing, comparing actual outcomes with predefine expected outcomes to find bugs that might reside in simpler operations such as login in or deleting an account.

Automating web application tests release testers from repetitive boring tasks allowing them to focus on testing that requires human prospective. These tests can be run without stopping or following a time-based pattern. Automated testing requires testers to fully understand the software that is being tested and have an “automation first!” mindset. Testers are required to tools that enable these tests.

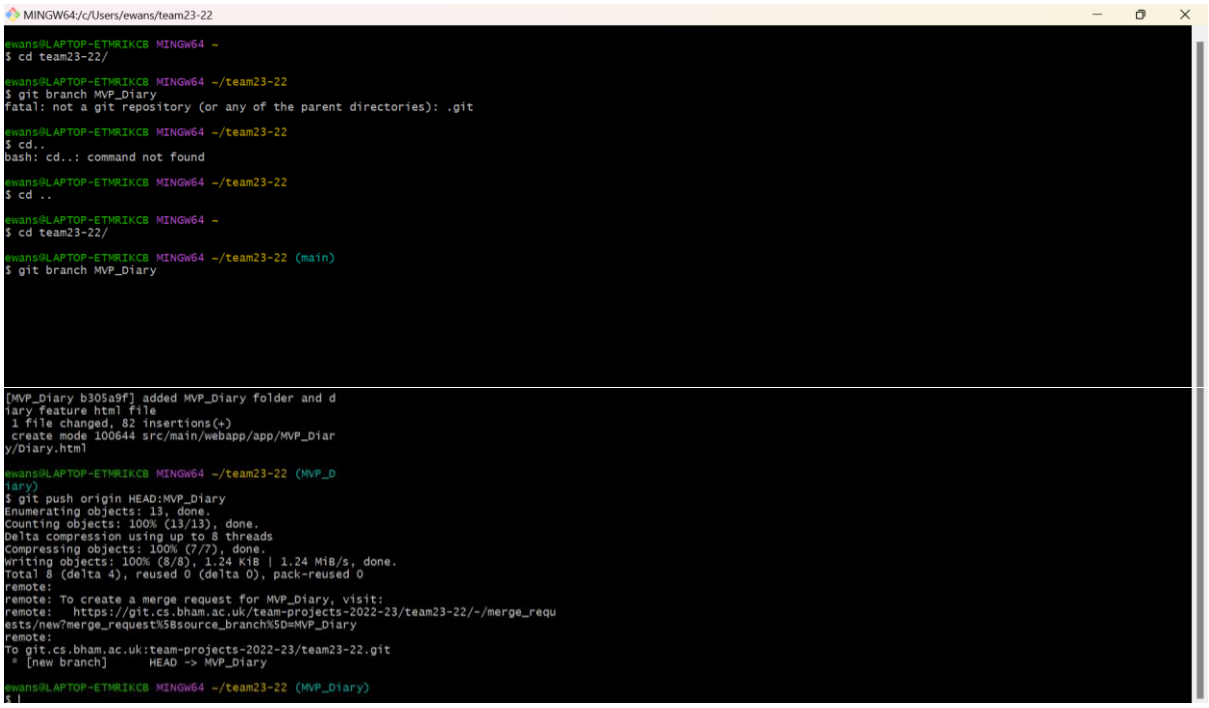
Not all tests are suitable for automation, there will always be a need for manual testing. the most suitable tests/task for automation are repetitive with predictable outcomes. The main type of tests that can be automated are:

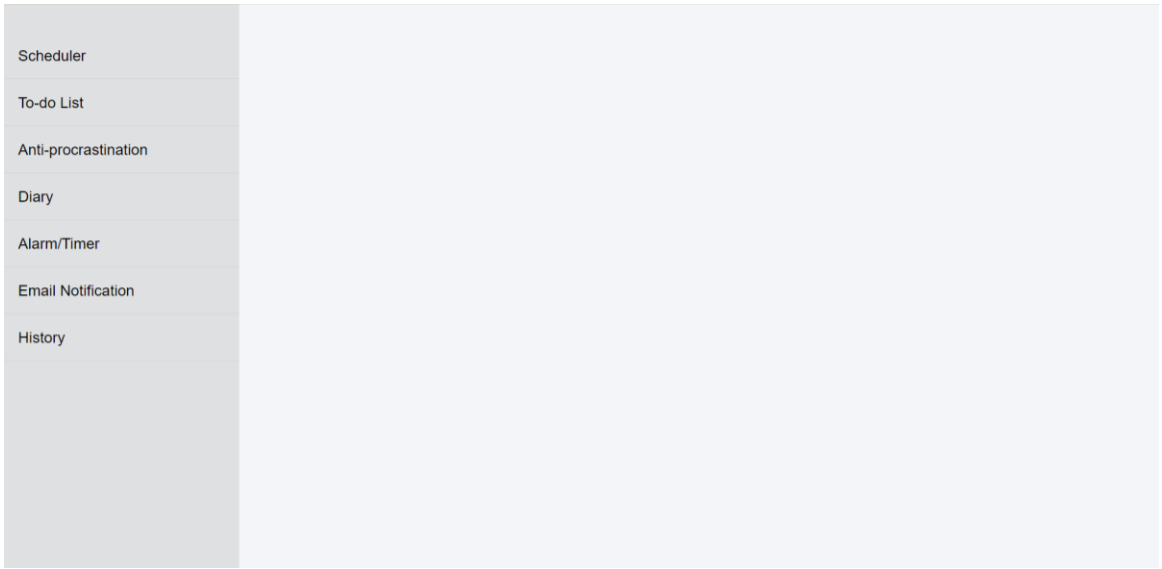
- Functional testing
- Regression testing
- Performance testing
- Cross browsers testing

When it comes to our website before we start automating our web application test we will have to draft a test strategy to decide what are the specific requirement of our website? What do we need to automate? Which test automation is best suitable for the purpose of our web application? and how much maintenance will automation require?
Only after answering these questions, we will then decide on what type of automation test our website might require. There are non-automated tests that we will need to carry out before releasing our web application this test will include testing our web application functionality, performance and usability.

Chance Egbon 2194210

evidence of git commits for MVP/CI development.





Chance Egbon 2194210