CPE 0326-1: GCA #2

SOURCE VERSION CONTROL TOOLS WITH GIT

GROUP 5

APRIL 23, 2022

PREMIUM CARD

All-Nighters MediCall™

KYLE GARCIA
MediCall Life Member
REF NO. 16789-23

MAR
21

BUENO, THERON ADRIANNE
CARDONA, ANGELO IAN MICHAEL
NATIVIDAD, JOHN CARLO
PINLAC, LAWRENCE CHYLE
VELASCO, JEROEN AARON

All-Nighters MediCall™

About Us

All-Nighters

Team Name

Medicall

Service Name

Adobe XD

Adobe XD is the software we have used to create a sample prototype for the App's interface we wish to make. Here in Adobe XD, we use different elements that we would like to include in our prototype.
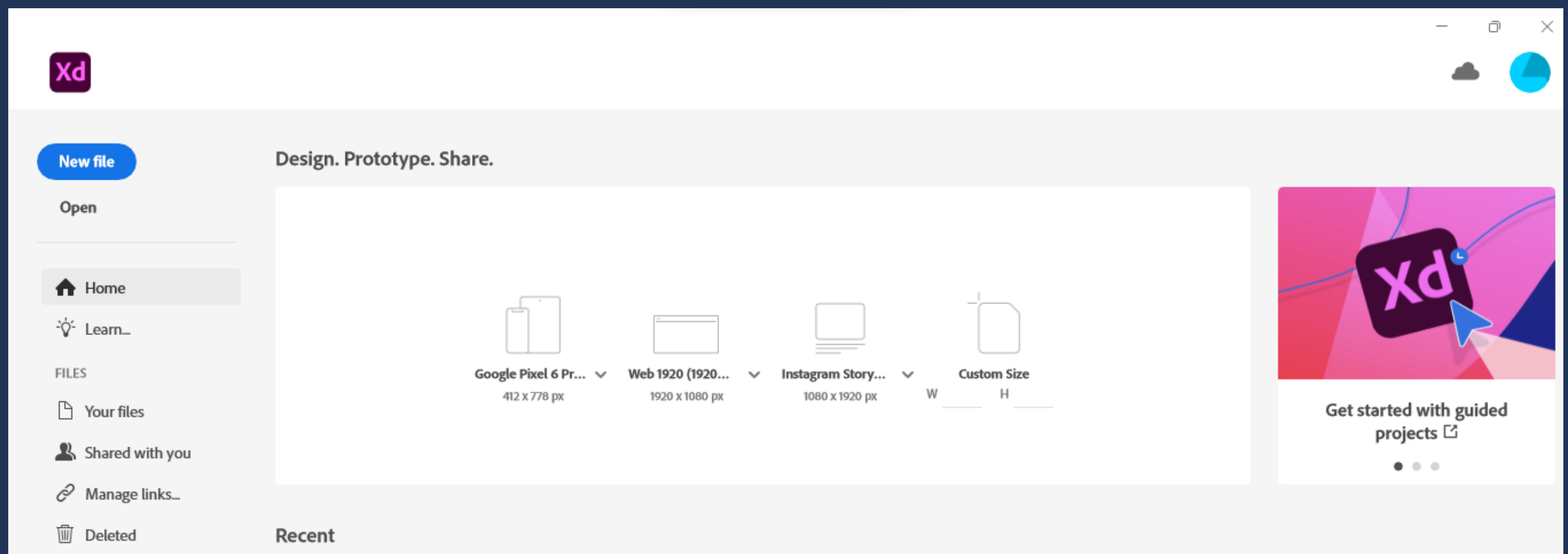

Figure 1: Adobe XD Main Page

Here on the main page, we can see the different artboards that the team used to portray a phone screen and fit in elements that would be present in the features they want to include in the app, as well as the different files that would be used or files that are already done.
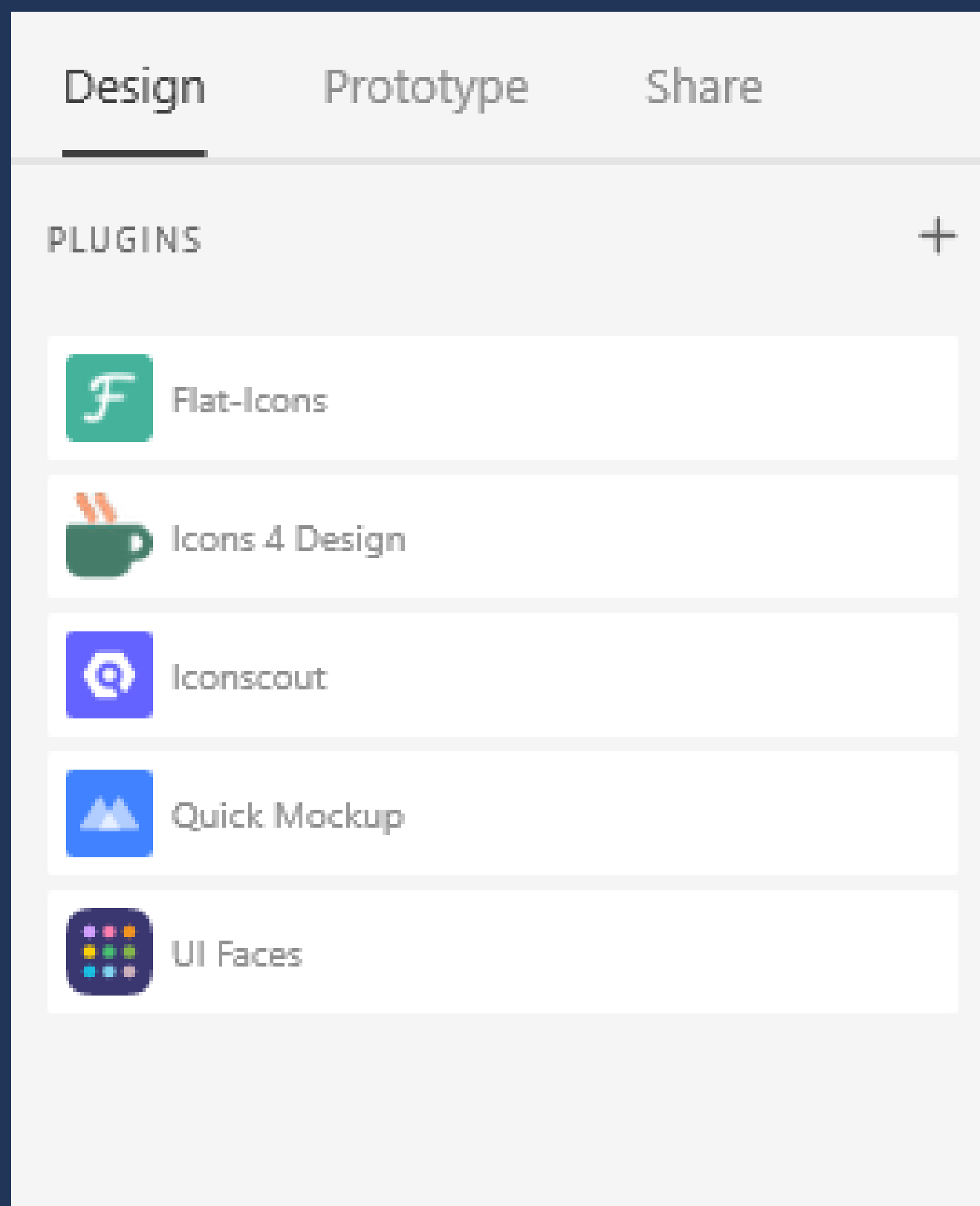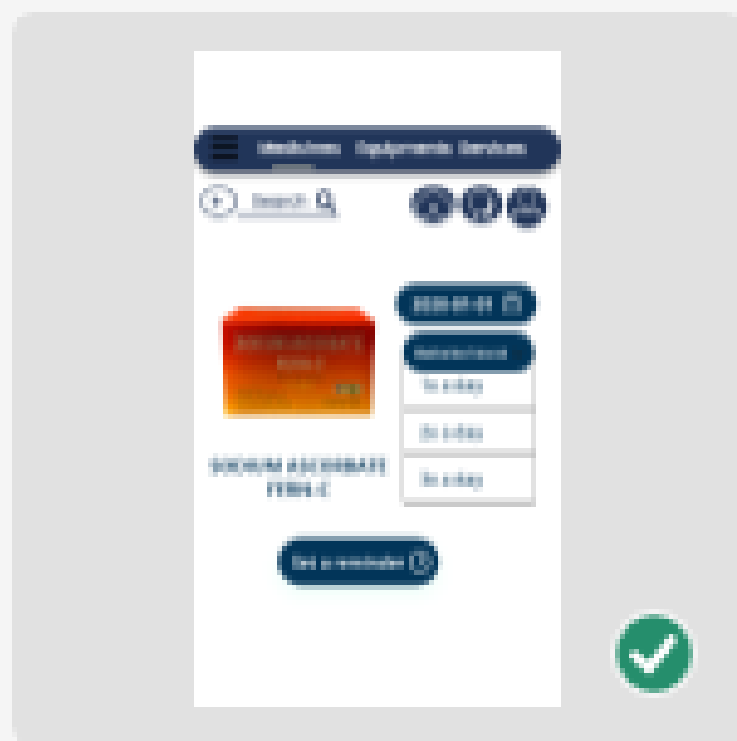
Figure 2: Adobe XD Plugins

In the Adobe XD Plugins, this is where the team used plugins to fetch elements that would be present in the interface they envisioned would be present in the app. With plugins, it would look much closer to elements present in other apps that are already existing since it would make the app look feasible as possible.

Recent

Pill Reminder Feature
CLOUD DOCUMENT
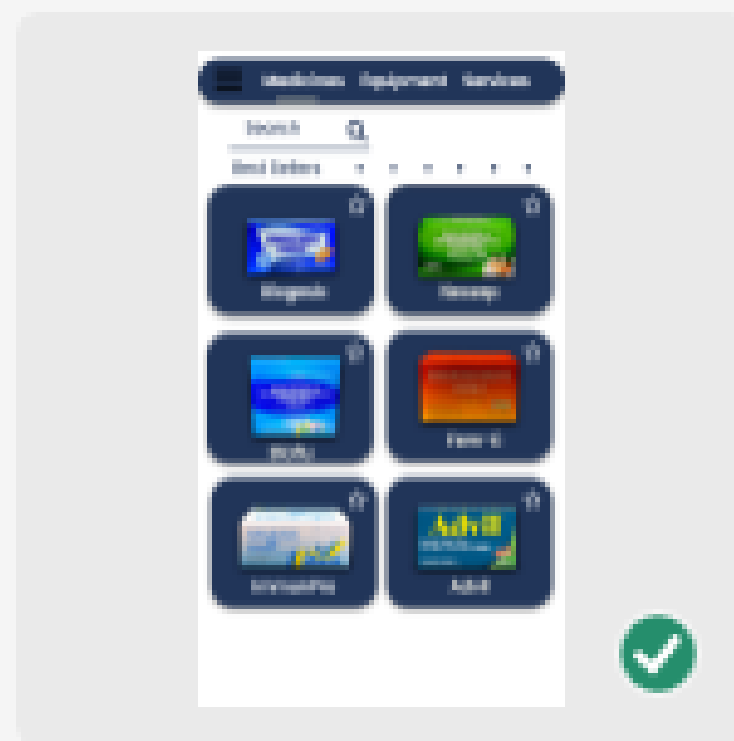
Favorites Feature
CLOUD DOCUMENT
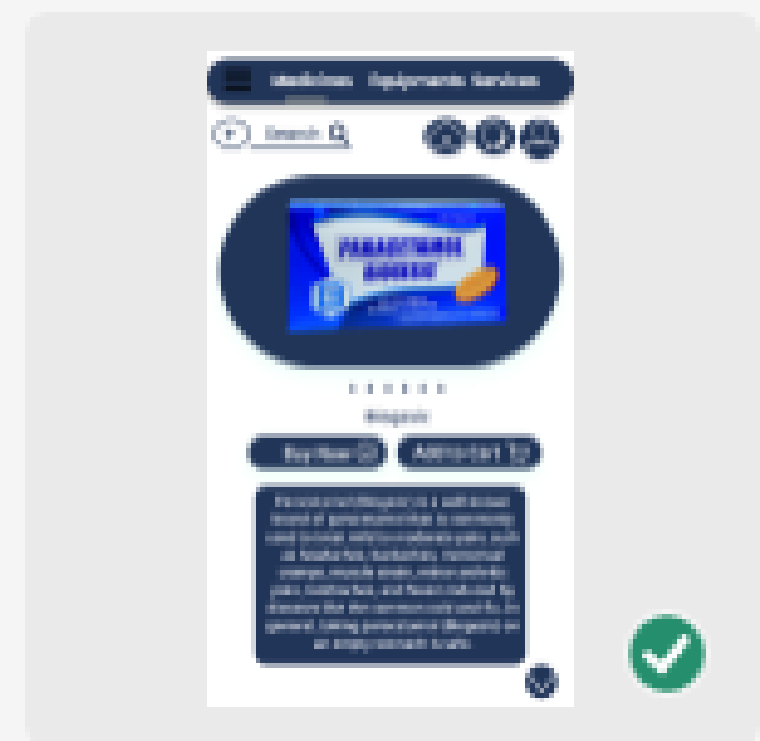
Wishlist Feature
CLOUD DOCUMENT
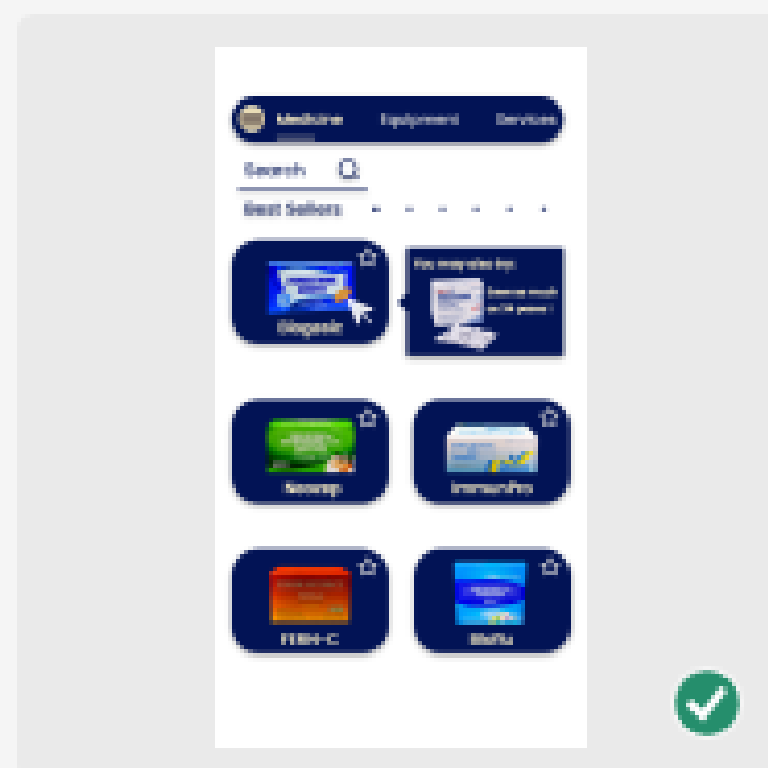
Suggestive Alerts (Fit size)   •••
3 days ago
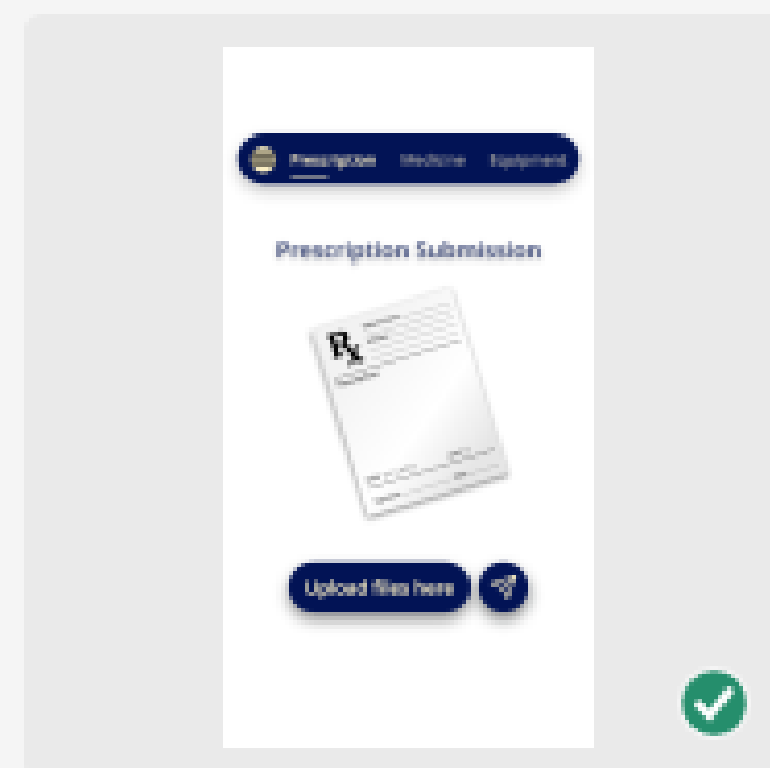
Prescription Subscription
(Fit Size)   •••
3 days ago

Figure 3: Prototype Page

Here are the different prototypes the team has procured for the mock app showcased on the services page. Present here are the different features such as suggestive alerts, prescription subscriptions, wishlists, favorites, and pill reminders.
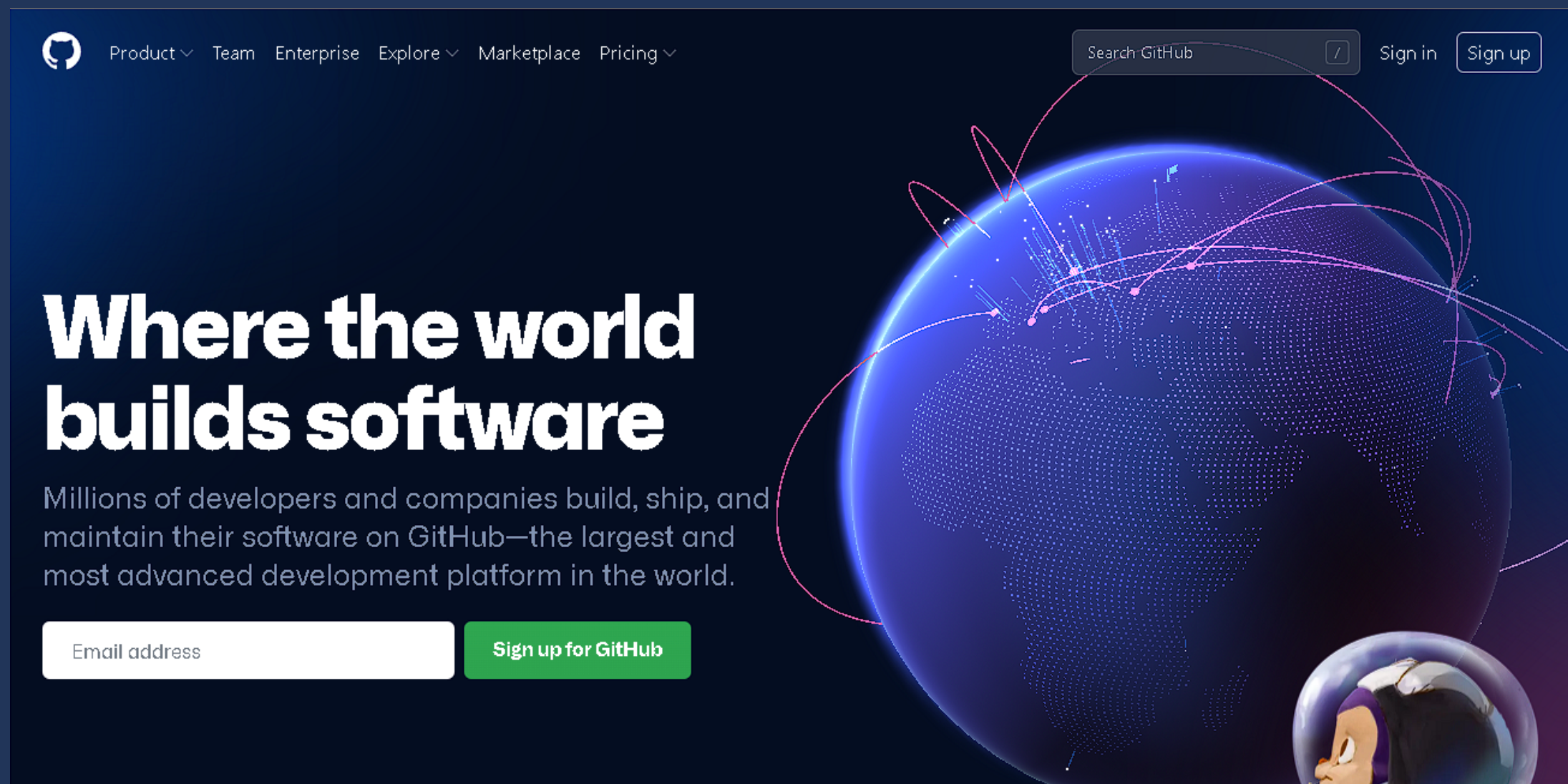
Figure 4: Github Home Page

GitHub is a website and cloud-based service that helps developers store and manage their code and track and control changes to their code, with two connected principles, version control and Git. Version control lets developers safely work through branching and merging. At the same time, Git is a distributed version control system. The entire codebase and history are available on every developer's computer, allowing easy branching and merging.

Figure 5: Github Sign Up Page
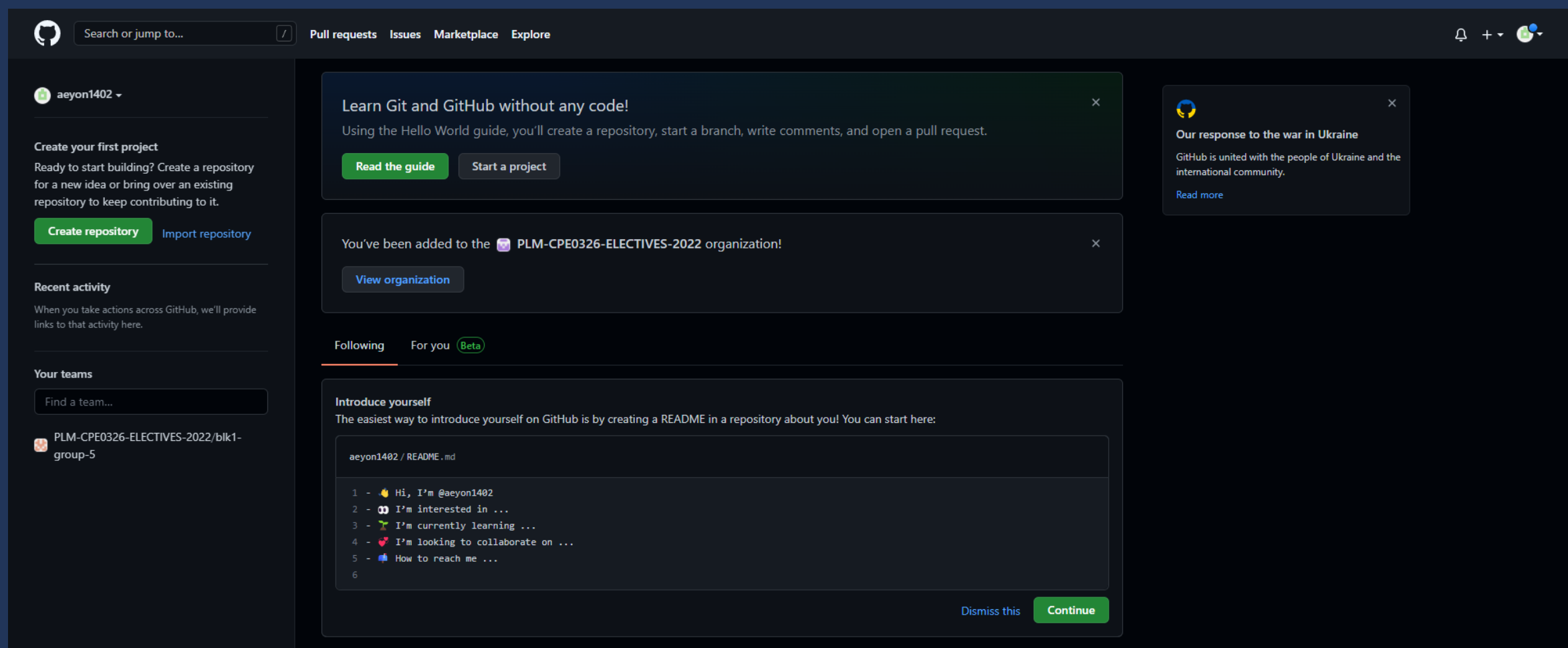
Figure 7: GitHub Main Page

The user may read their tutorial and learn how to get started creating, delivering, and managing software with GitHub, as well as explore their products, on this page. The user may also look through GitHub's news feed, which is tailored depending on the user's choices.
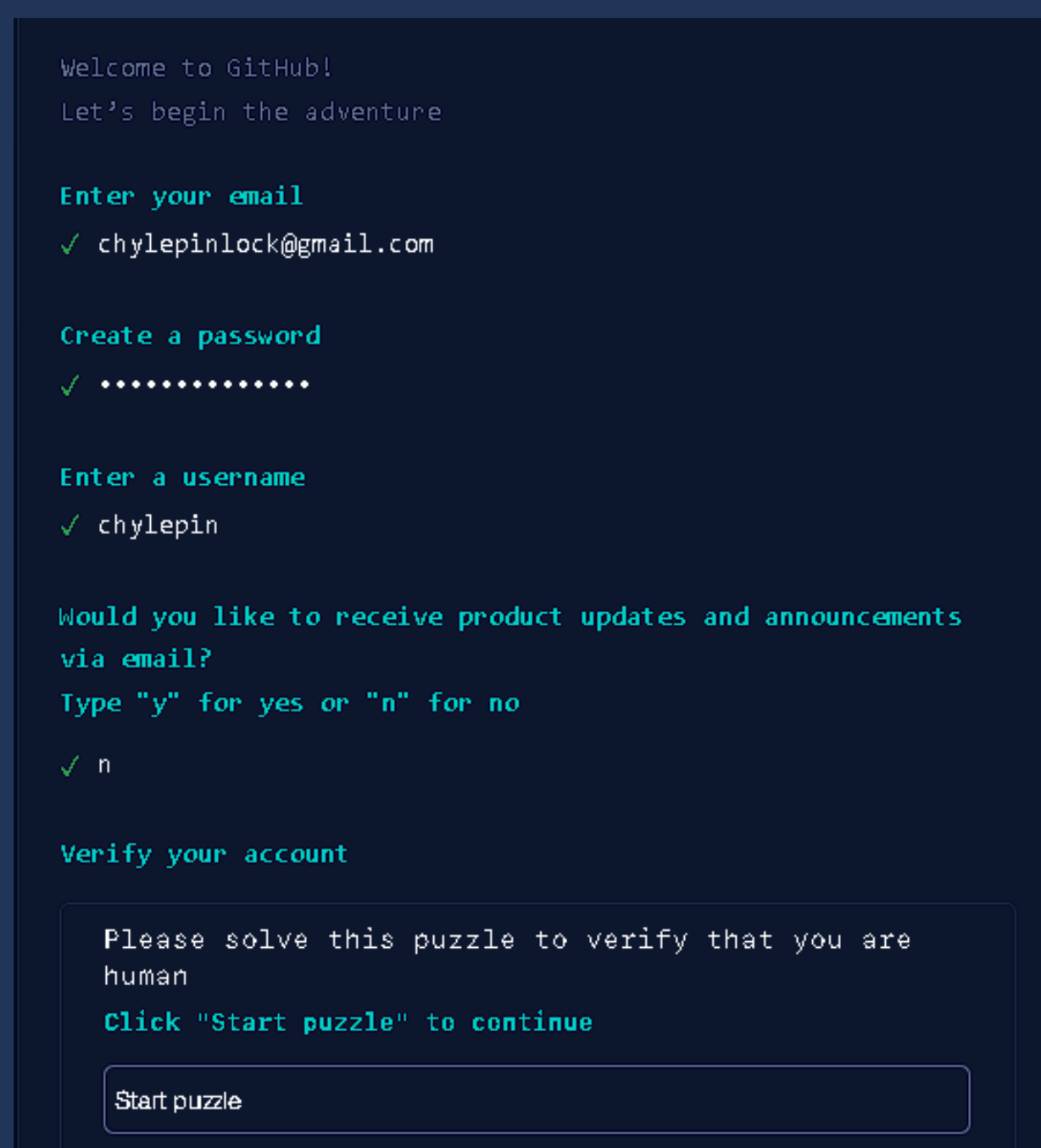

**Figure 8: GitHub
Branches**

As we have mentioned earlier, Branching basically lets you have different versions of a repository at one time. You can use branches to have different versions of a project at one time. This is helpful when you want to add new features to a project without changing the main source of code. The work done on different branches will not show up on the main branch until you merge it

Figure 9: Creating
Fork

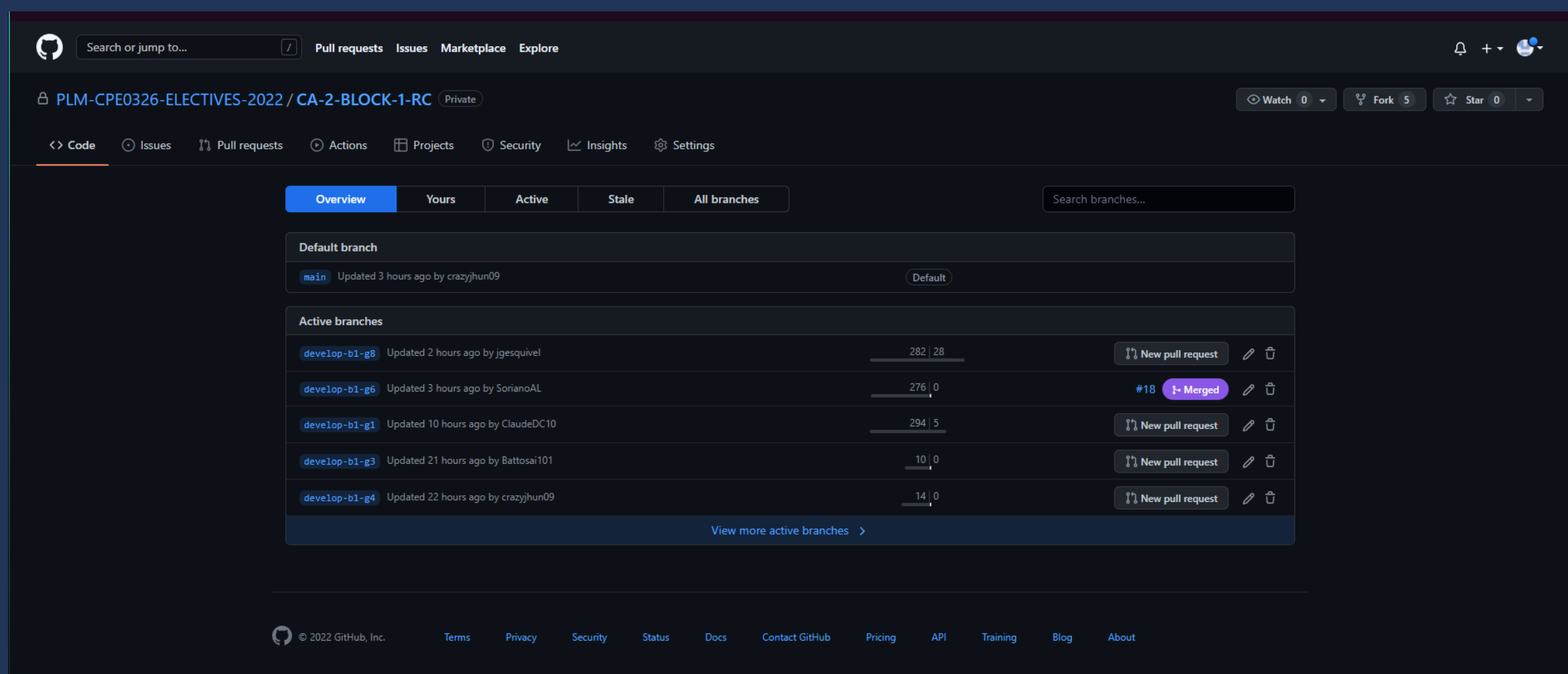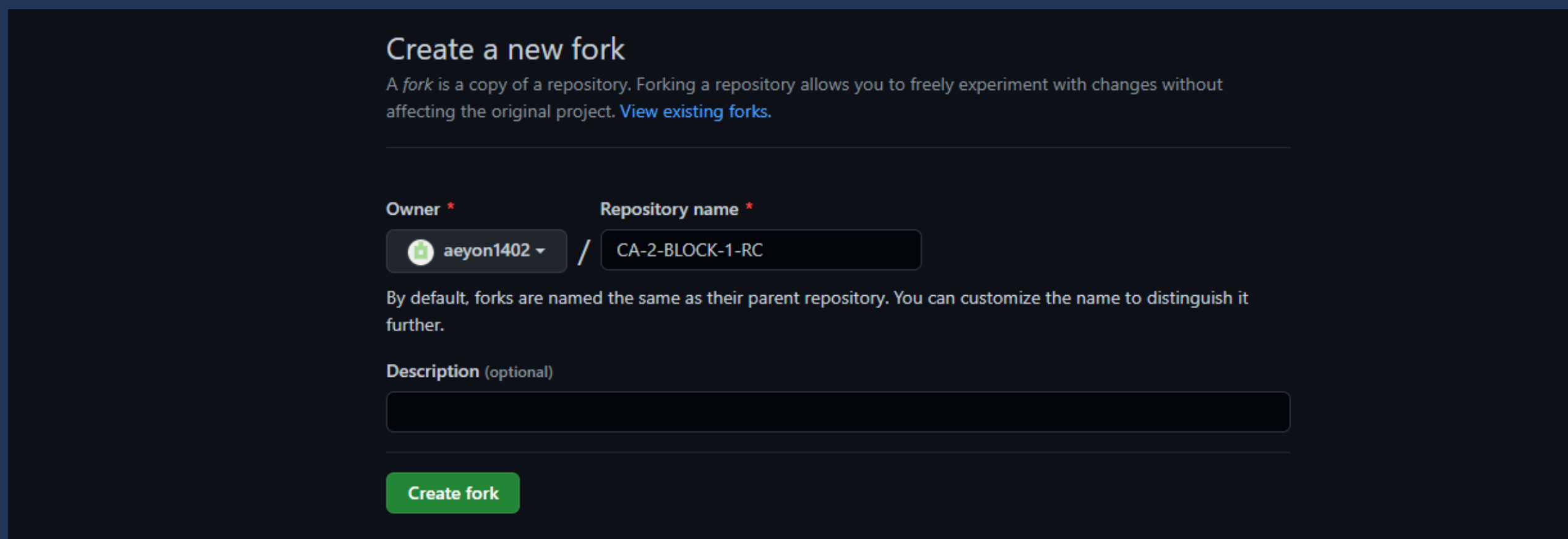With this feature, we can try making changes for improvements and then propose them without directly affecting the main project instantly. Thus, we can safely make changes and tests without creating errors on the main repository.
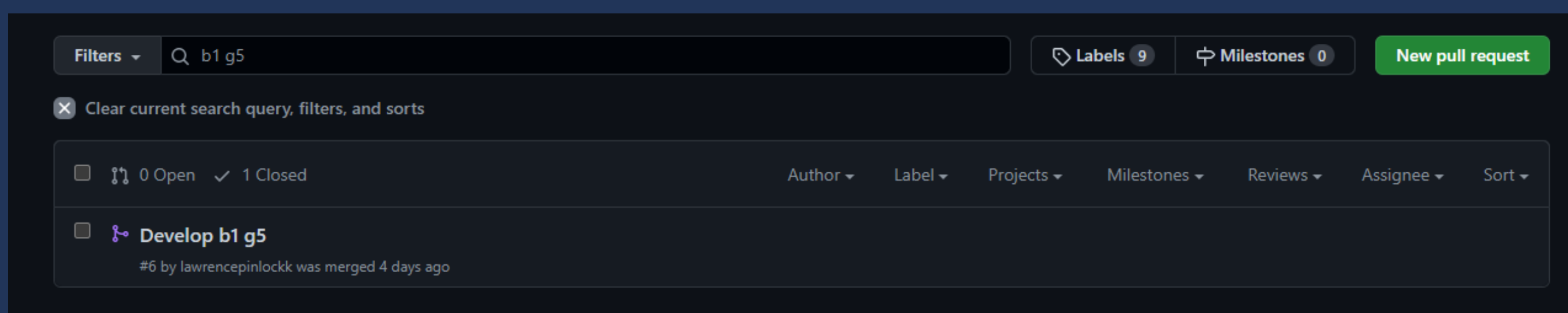
Figure 10: Pull Request in branches

Pull requests are at the heart of GitHub cooperation. When you open a pull request, you're presenting your modifications and asking for someone to evaluate, pull in, and merge them into their branch. Pull requests display diffs or differences between the content in both branches. Changes, additions, and subtractions are shown in various hues. Even before the code is ready, you may initiate a pull request and start a debate as soon as you make a change.

Figure 11: Pull Request in Origin

A pull request is a mechanism for a team to notify the repository's owner that they wish to modify the software. It allowed the owner to check the provided code to ensure that the modifications proposed by a p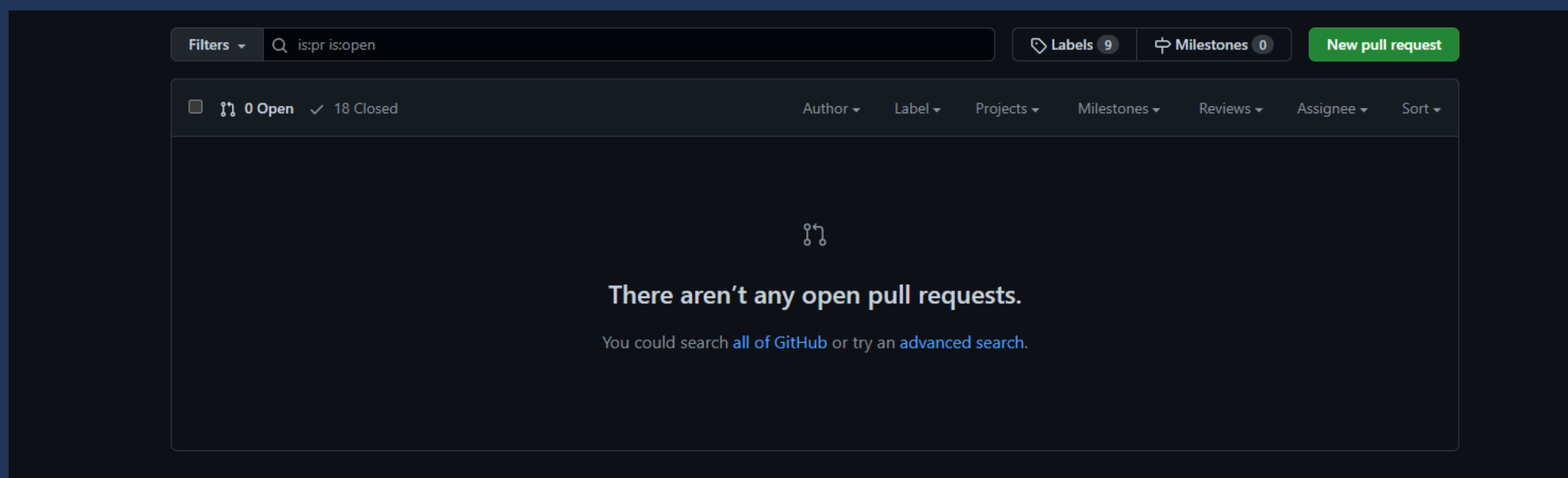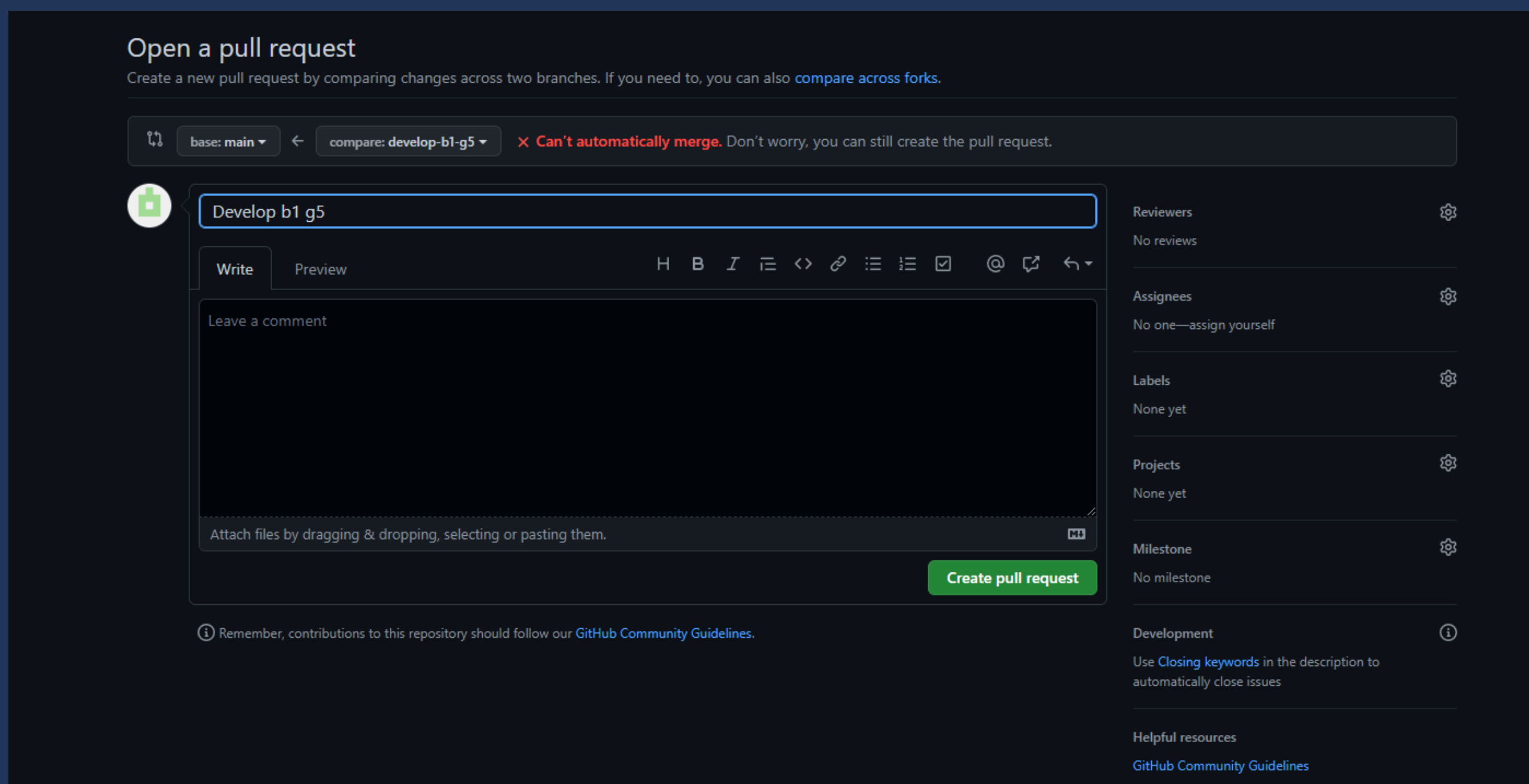rogrammer would not necessitate further work before merging the changes with the primary branch. When you click the 'Merge pull request button, the modifications submitted by a user are immediately merged into the main branch.

The "commits" tag on the right side of the screen displays the number of commits. This may be used to see if the commits were correctly merged into the main branch. When working on the primary branch, users can use the "git pull origin main" command to see the most current modifications made by all team members. In most circumstances, this command may be simplified to "git pull."
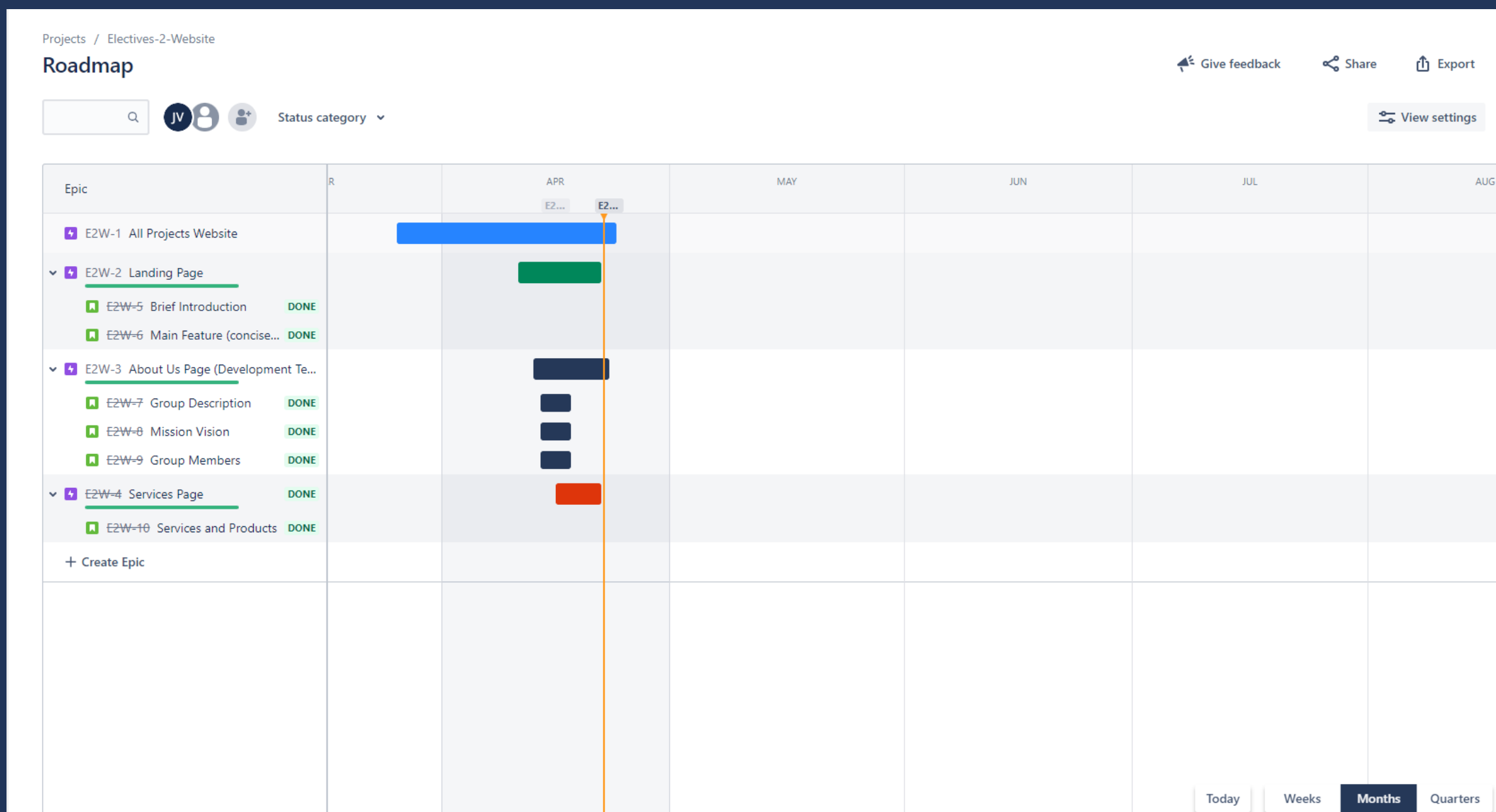
Figure 12: Medicall Project Roadmap

The team has produced four separate epics in the new roadmap that cover all of the work done during the project and the timing of when each is needed. Everything must be completed, from the construction, design, and content of the website, application, and Github to the activities that require care in general.
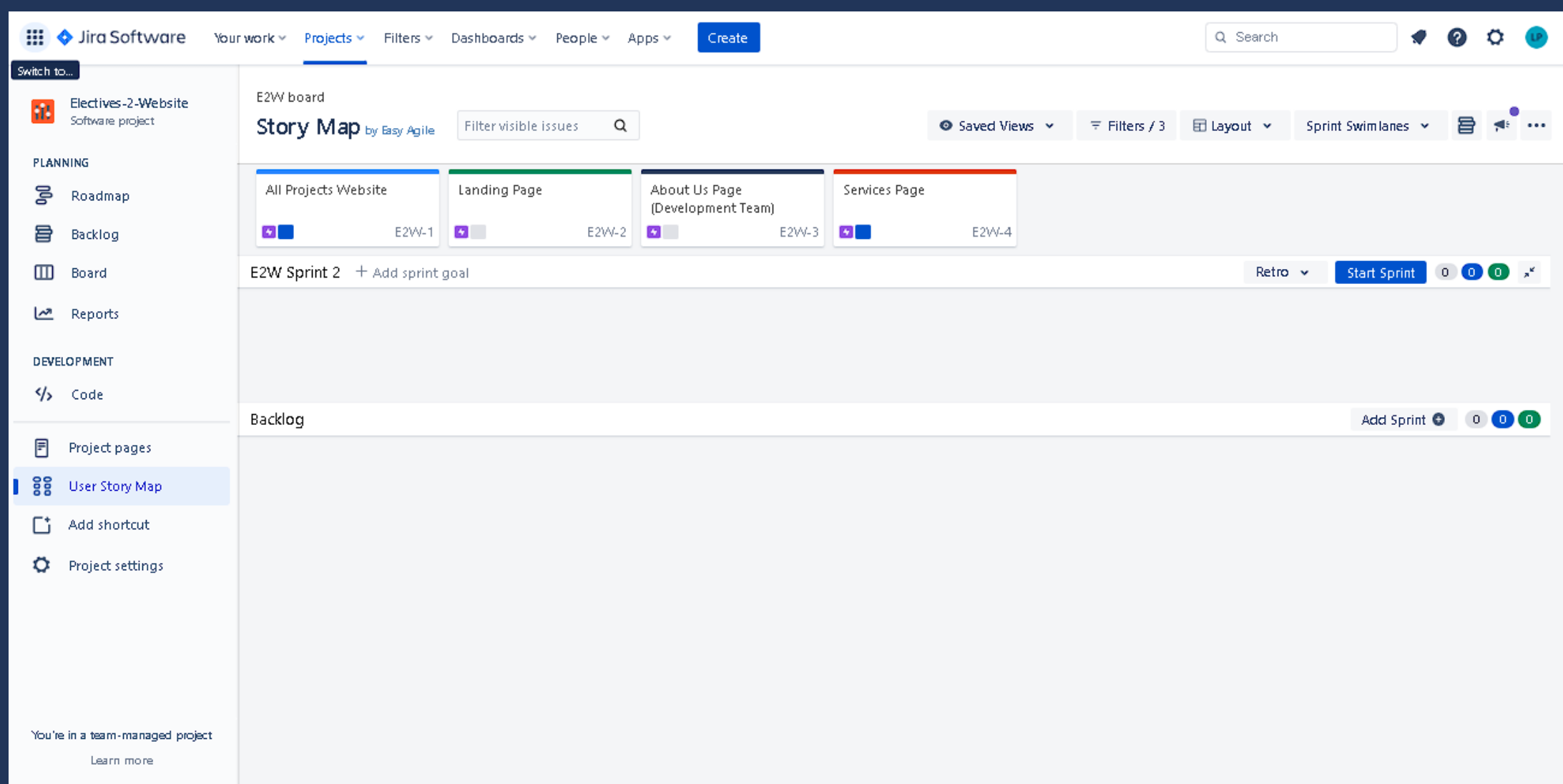

Figure 13: Story Map using Easy Agile

This shows the corresponding epics and the sprint where it belong. In order to create the story map arranging the sprint accordingly is a must.
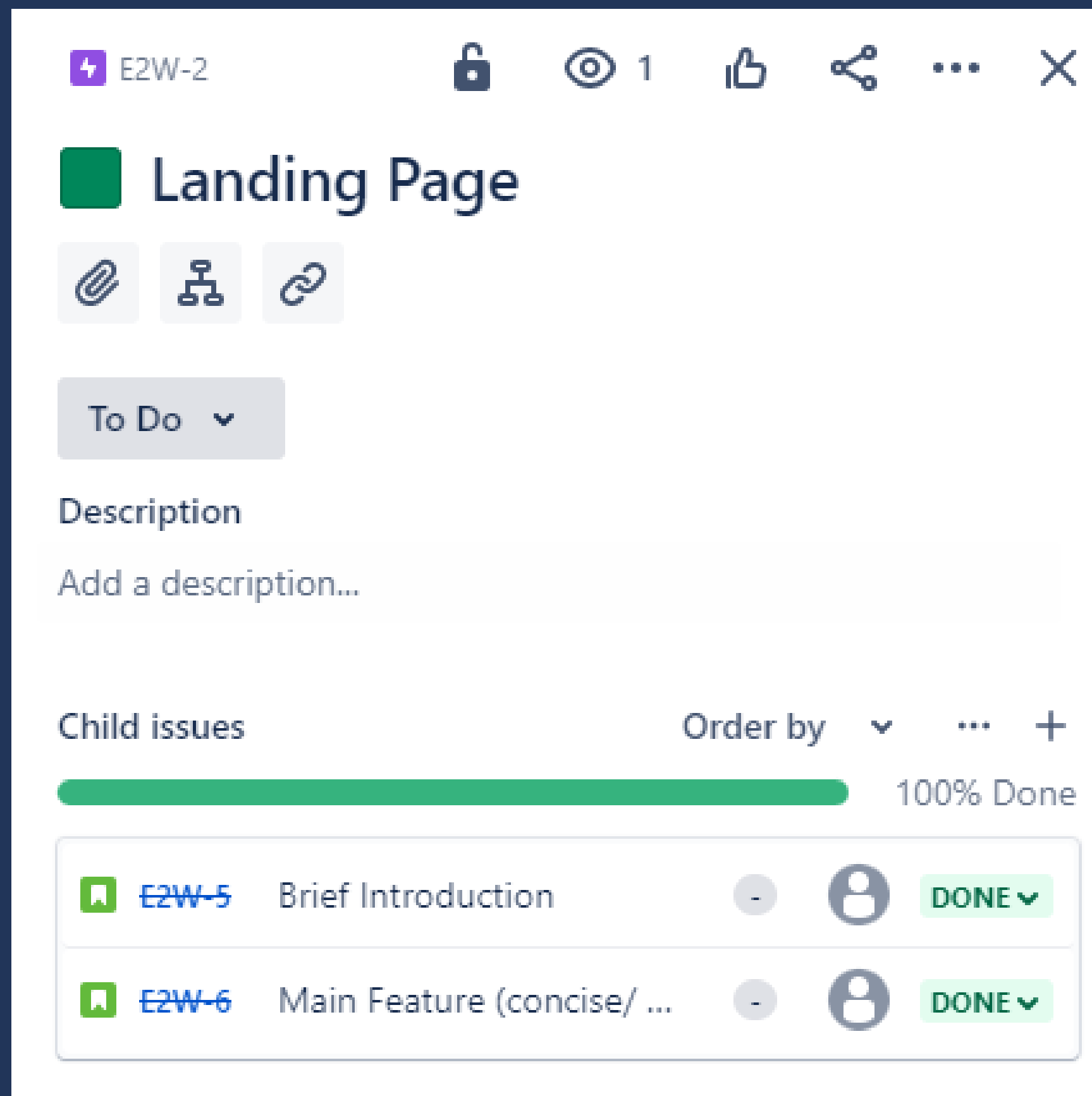
Figure 14: Landing Page Interface

There were two child concerns with the Landing Page Interface. This epic includes the child issues of Brief Introduction and the Main Feature shown for every group on the Landing Page. Child concerns reflect a specific job that a team member must do. All of the children's difficulties that the team encountered were dealt with properly.
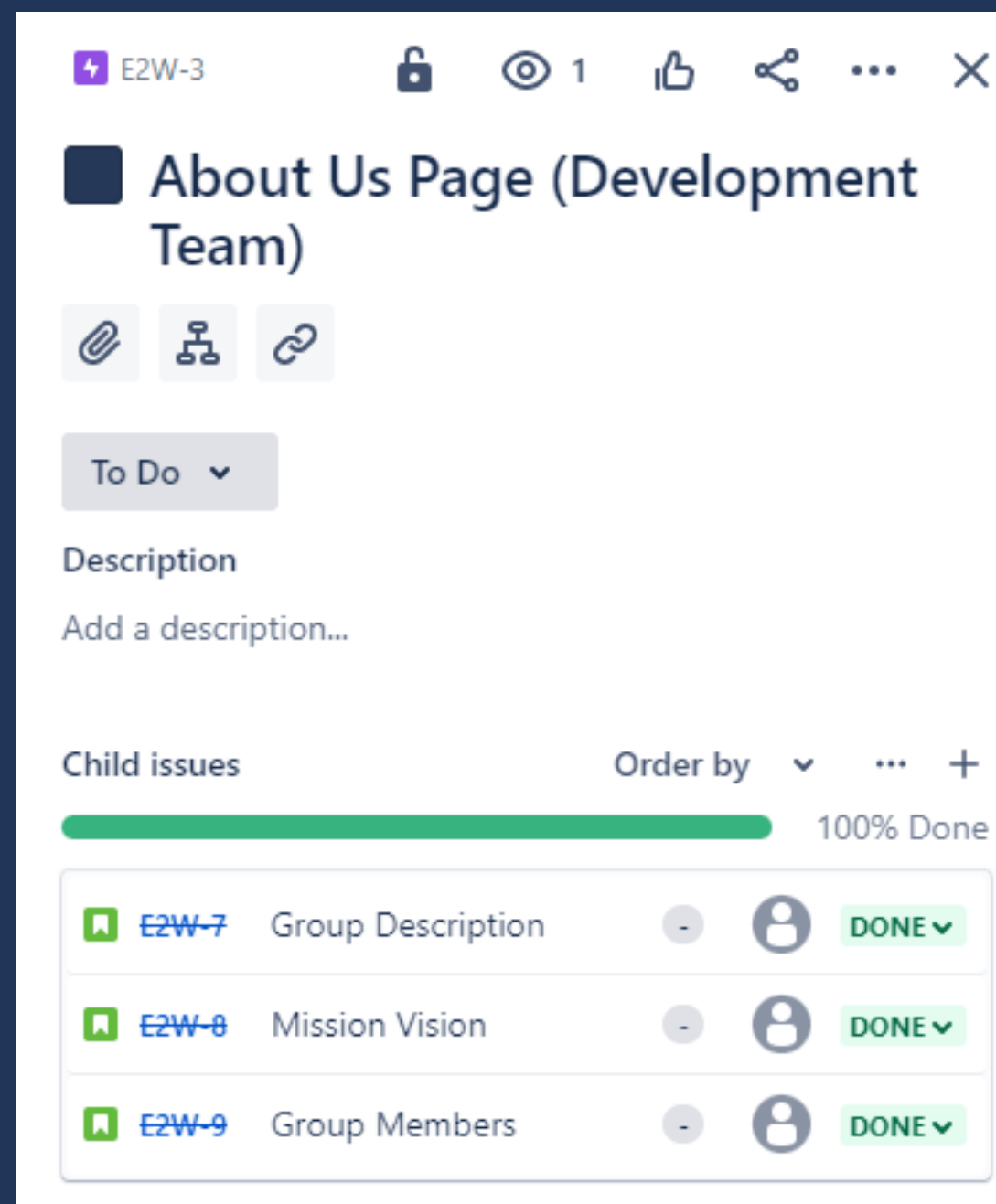
Figure 15: About Us Page Program Interface

All allocated duties are included in the description section of the About Us interface. The information and issues stated in this epic were accomplished according to the specified time frame.
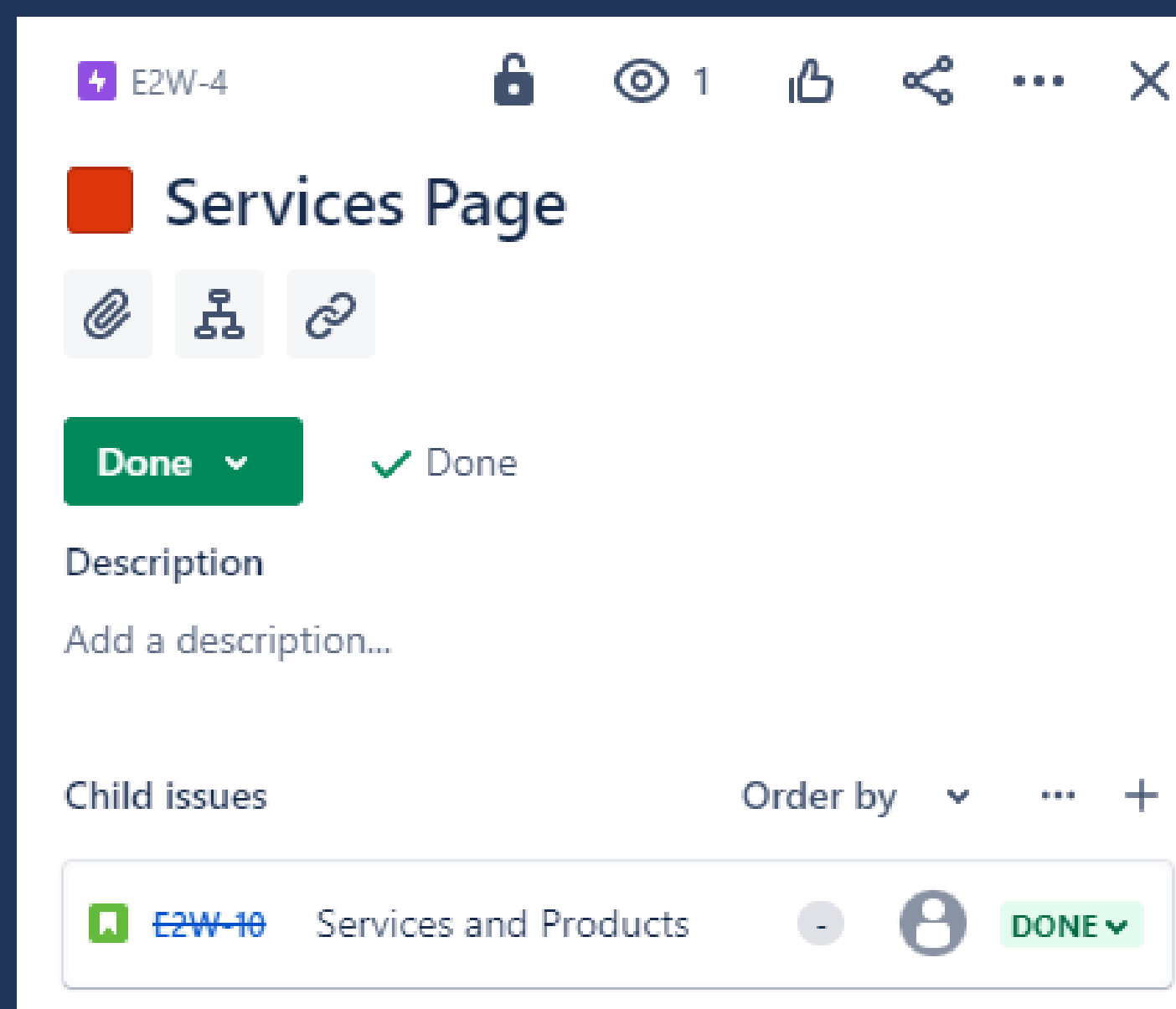

Figure 16: Services Page Program Interface

Only a child problem job was given to be completed in the Services Page Interface, and the creation comprises the many services provided by Medicall and its App Features. The team completed the work and was appropriately marked as done.
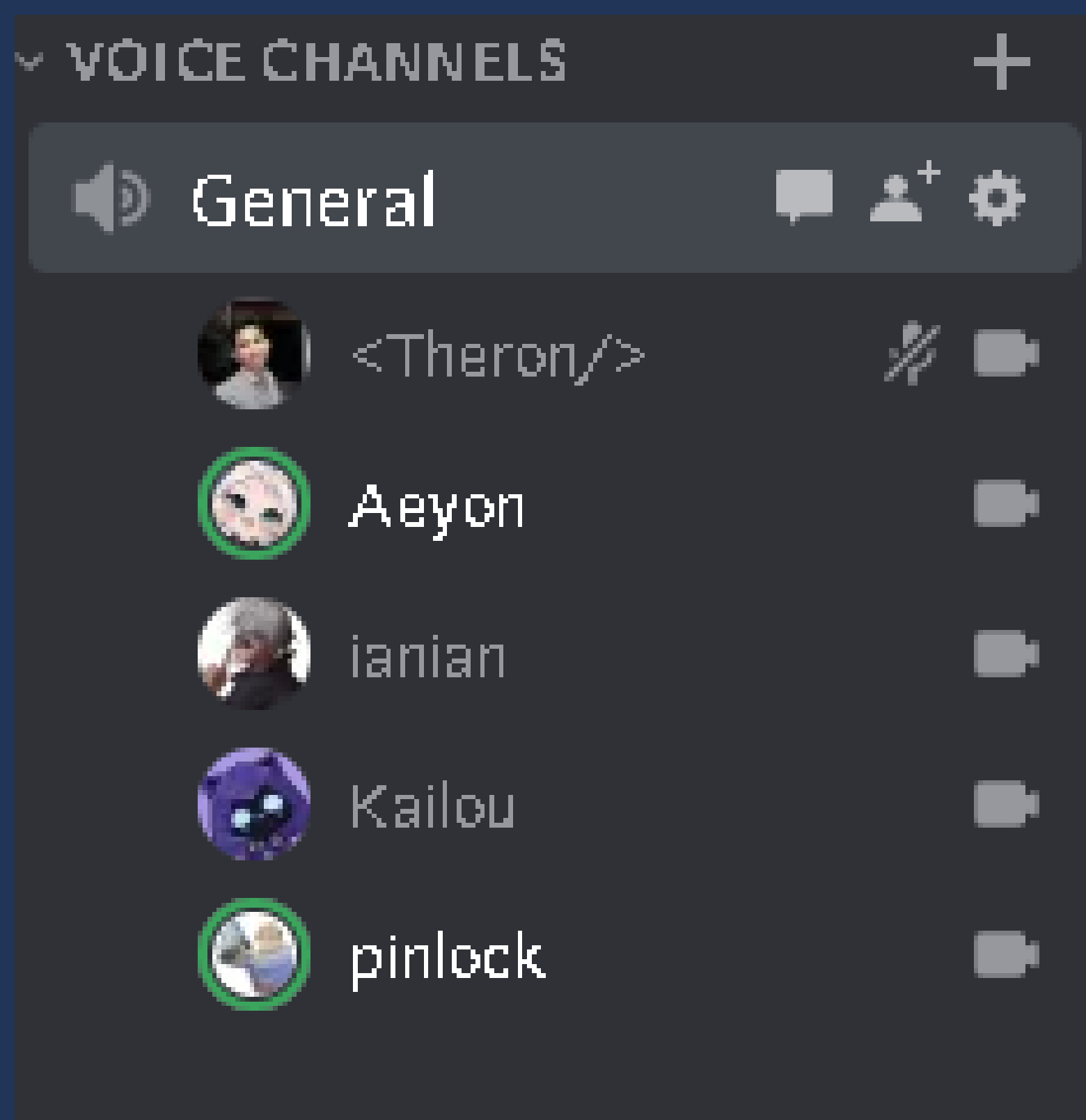
General Observations and Conclusion

The team decided to use Adobe XD for mockups to create realistic app prototypes. Jira, to monitor the progress and set the deadlines on when a part of the project should be done and how much time was left to handle that part. Finalization was done in the last few nights. It was chaotic at first, with all the confusion regarding the push and pull mechanics of GitHub, making the team decide not to tinker much with the push and pull and try to make much of the code before trying to make requests.

Source Code Management, particularly GitHub, was pretty new to some team members, but it was fun exploring it and tinkering with it and the code. It was chaotic at first, but later on, the team decided to delay push and pull requests until they were sure that changes should take place to record progress. It was pretty scary that a team could mess up the code of others or replace them entirely with a misclick.

The group decided to divide the work but continue to share ideas to make the job easier and adequately distribute the workload. At the same time, they also impart knowledge to each other, whether it is in coding or editing and designing the prototype. With this method, the team was able to explore Github and Adobe XD much further and even find ways to execute or push through with the ideas/vision that the team had.

Documentation

References:

- https://kinsta.com/knowledgebase/what-is-github/
- https://docs.github.com/en/get-started/quickstart/hello-world
- https://docs.github.com/en/get-started/quickstart/fork-a-repo