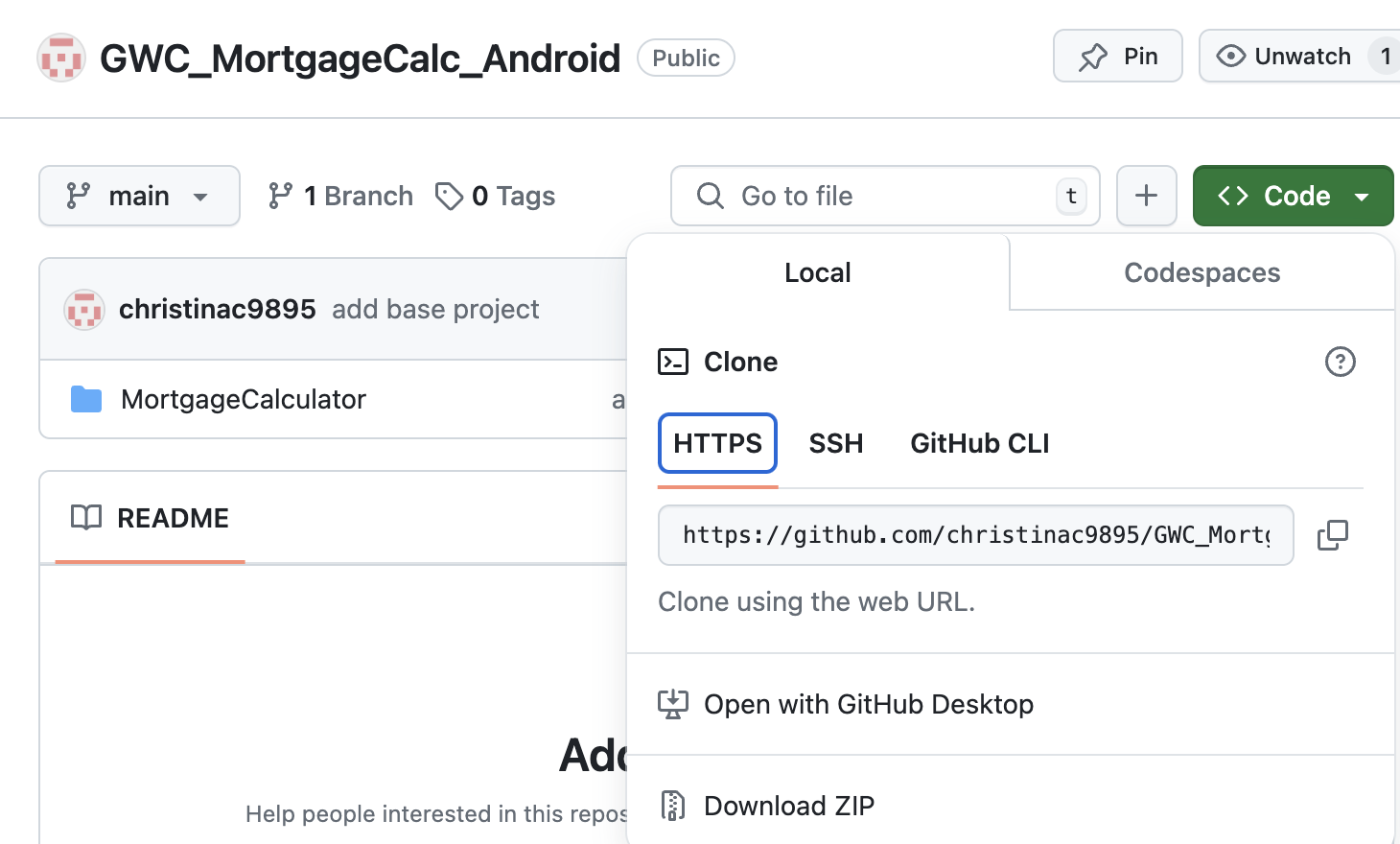
Git Project - Android

## Project Setup

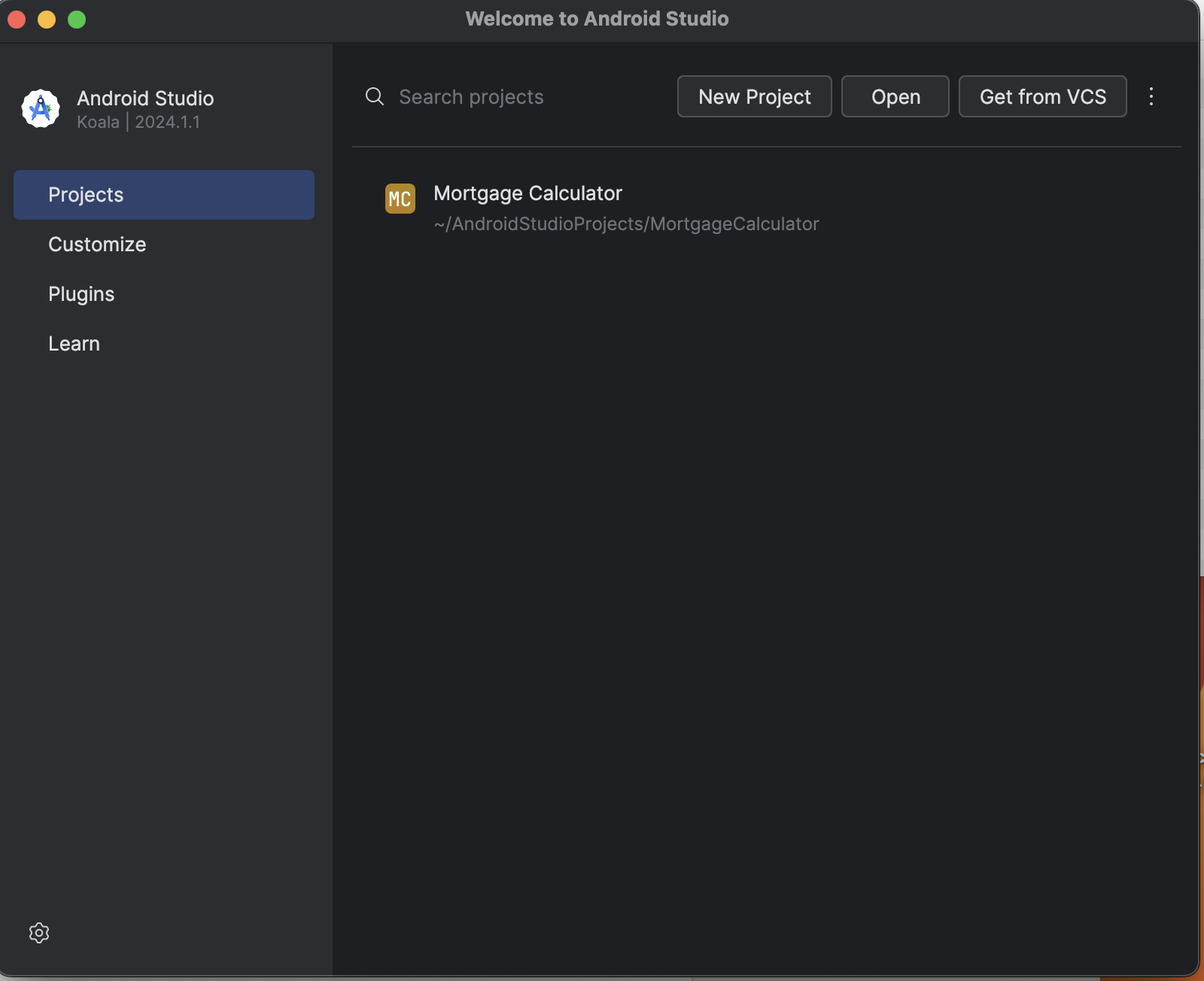
Create a free GitHub account

* 1. <https://github.com/signup?ref_cta=Sign+up&ref_loc=header+logged+out&ref_page=%2F&source=header-home>

2. Click the link below to find your teams base project.Notice there is already an .xcodeproj setup for you. You will use this to develop your app   
[ADD LINK]

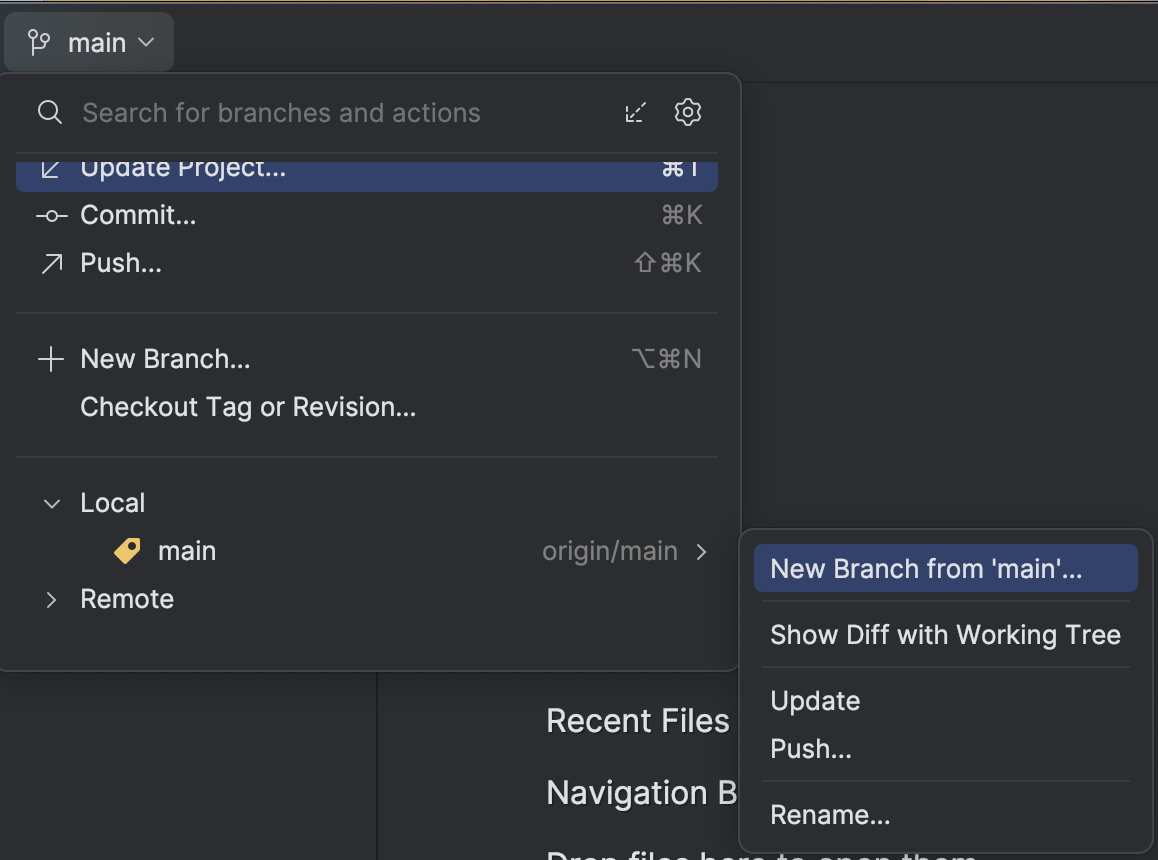
3. Click the Code button, and copy the HTTPS URL.

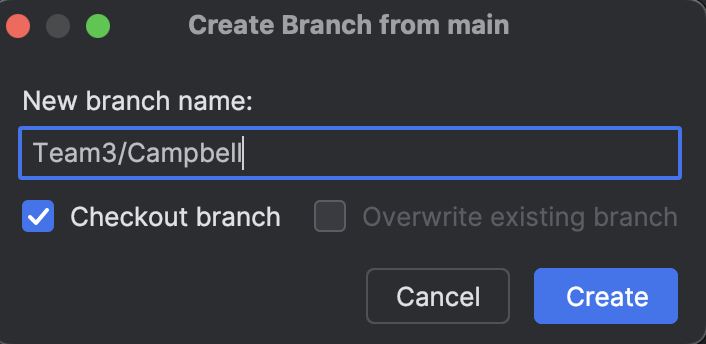
4. Open up Android Studio. Click the option for “Get from VCS”.

5. Paste your projects HTTPS URL in the URL text box, ensure the project is being cloned in the directory you want, and click clone. When the popup appears, choose “Trust Project”. The Project will automatically open, and you are ready to start coding!

## Git Commands through Android Studio

**Create your own branch**

1. You should NEVER work directly off of the main branch or push your changes directly to the main branch. This can lead to merge conflicts and make it extremely messy to merge your code with your teammates. You always want to create a branch of your own and then later pull your code changes into the main branch. To create your own branch from Android Studio, click the “main” drop down at the top of the screen. Click on the origin/main option, and select the “New branch from main…” option.
2. Name your branch your team number followed by your last name. For example: Team3/Campbell. Click “Create”. You should now see the “main” dropdown now says your new branch name instead.

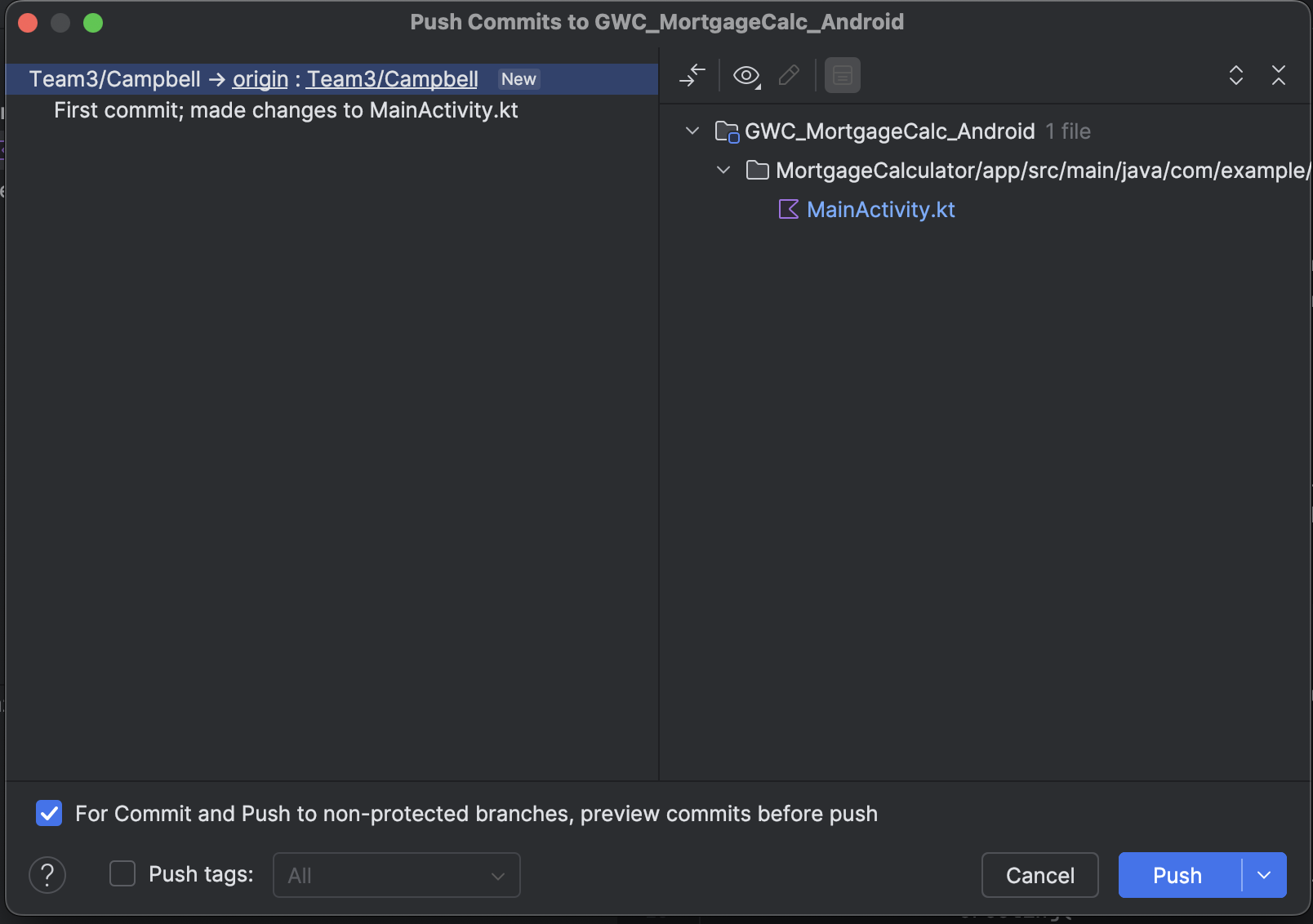
****

**Pushing your code**

1. After you’ve made code changes that you are ready to share with your team, you’ll first need to commit your changes. By committing, you are confirming that you have tested your changes and are confident enough in them to share them with your team. In the top navigation bar, click the “Git” tab and select “Commit…”

2. Select only the checkbox next to the files that you wish to commit, i.e. only the files that you made changes to. The files will be under the “Changes” dropdown. NOTE: you do not need to commit the unversioned files. These are auto-generated caching files. Add a commit message describing what changes you’ve made. Click “Commit and push” to complete the commit, and begin the Push process.

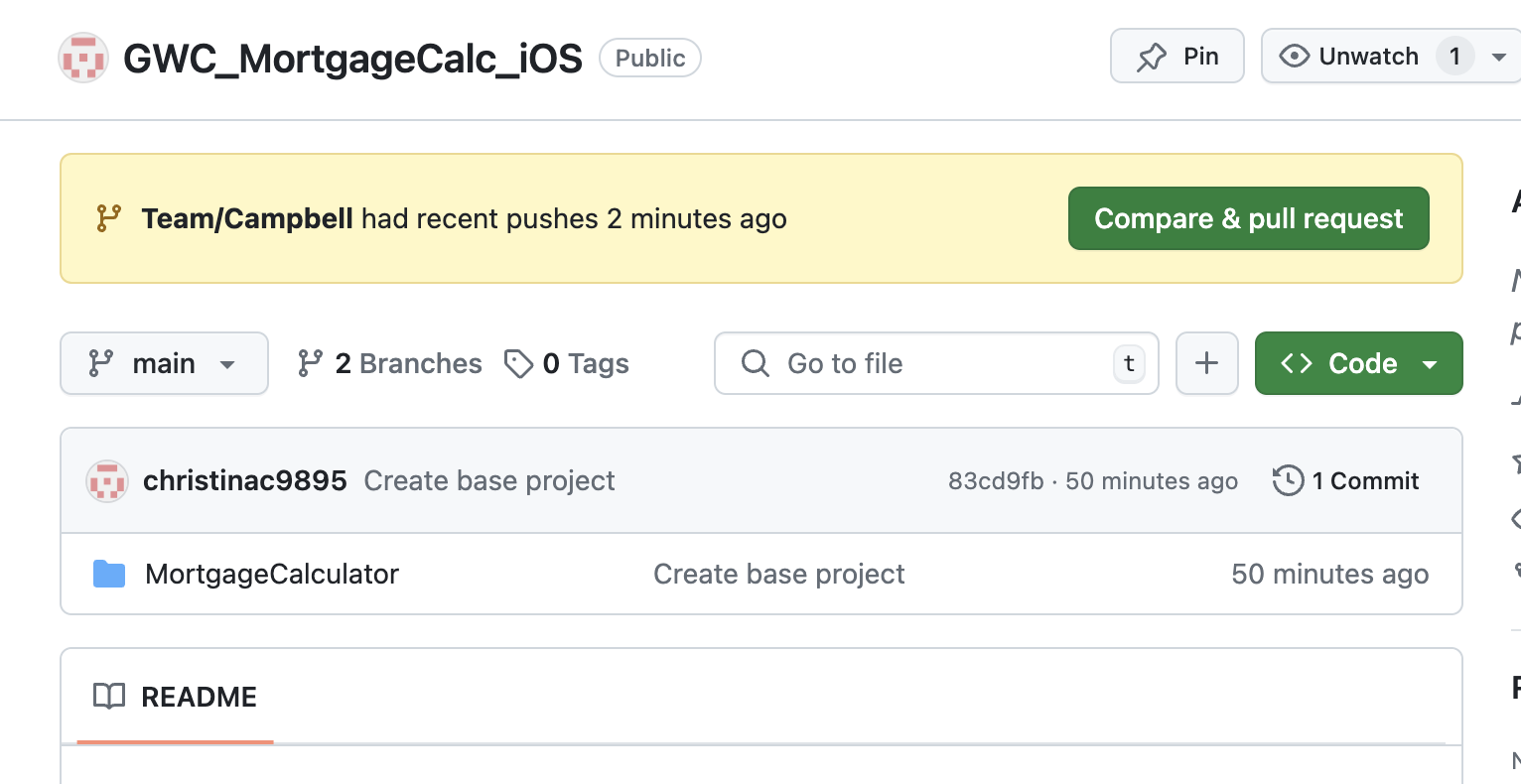
3. On the next popup, confirm the commits you are going to push are correct, and click “Push”.



4. If this is your first time pushing code for this project, you may be asked to authenticate yourself. You may then see a screen asking you to enter a Personal Access Token. Follow these steps to [create your personal access token](https://docs.github.com/en/enterprise-server@3.9/authentication/keeping-your-account-and-data-secure/managing-your-personal-access-tokens#creating-a-personal-access-token) . Make sure you select the scope sets specified below when created your token. Copy and save your personal access token on your computer (NOTE: YOU WILL NOT BE ABLE TO VIEW THE TOKEN AGAIN SO MAKE SURE YOU SAVE IT!). Paste your personal access token into the textfield, and complete the “Push”.

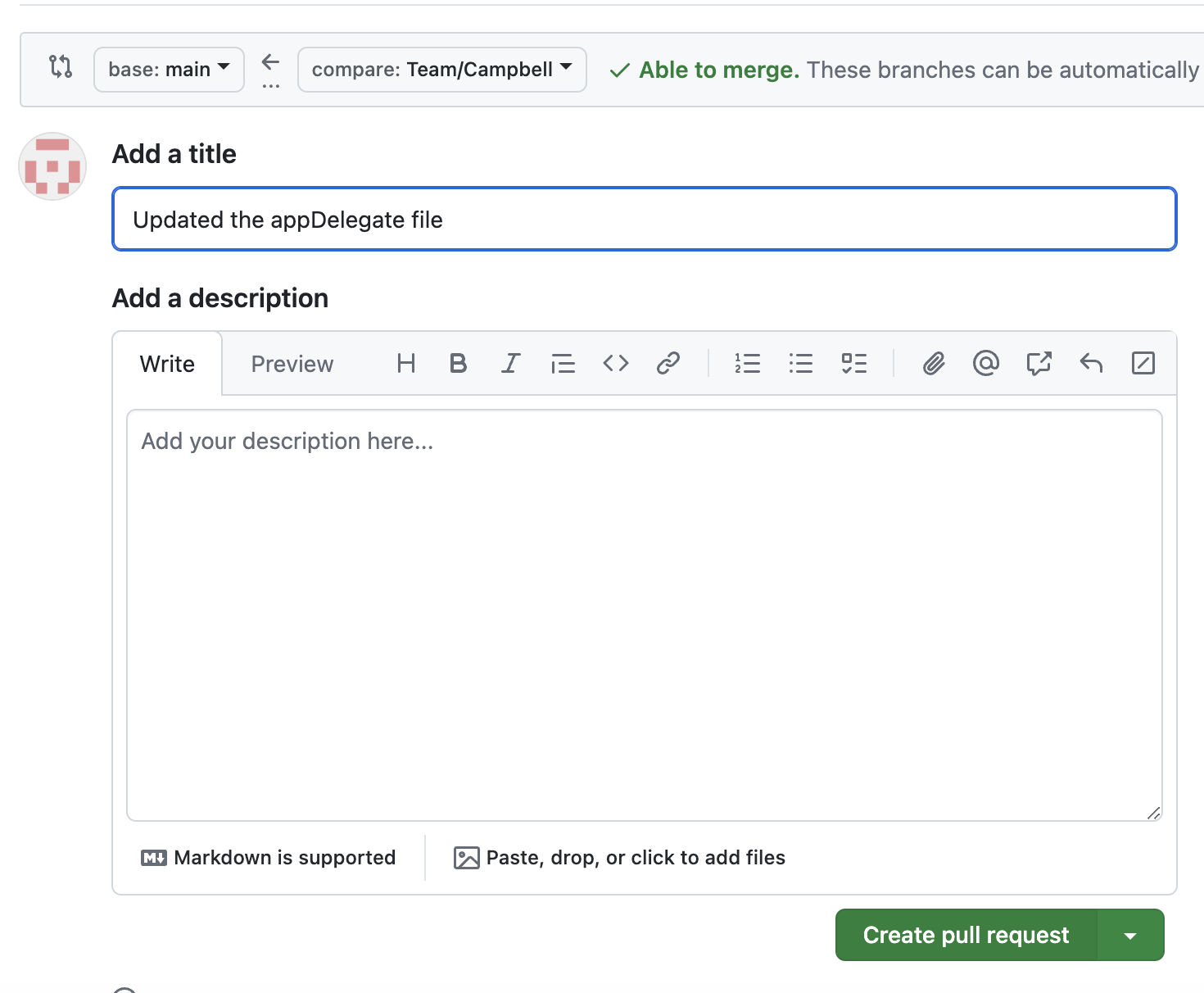
**Opening a Pull Request**

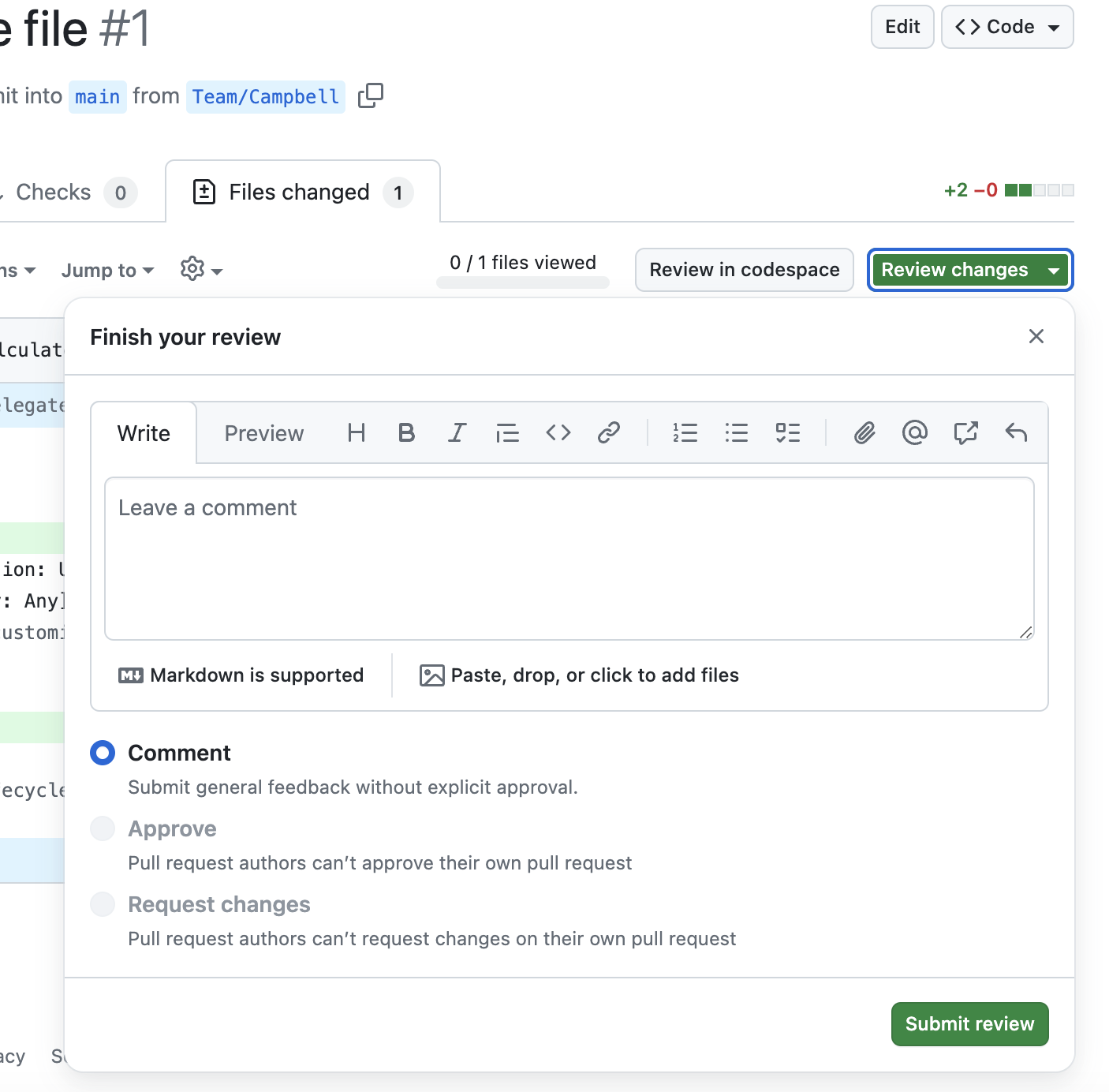
1. Pull requests are used to review your teams code before merging their changes into the main branch. To open a merge request after you pushed your changes to your branch, navigate to your project on GitHub. You should see a prompt that has the option “Compare & pull request”. Click the button.



2. You’ll see a pull request is auto-completed for you. The base branch should be “main” and the “compare” branch should be your branch. If you’d like to add more context for your team, add comments in the description. Once you’re done, click “Create Pull Request”.

**Reviewing Pull Requests**

1. Once your teammate has opened a PR, it is up to your team to properly review and once approved merge the PR into the main branch. To review the PR, on your projects GitHub page navigate to the “Pull Requests” tab. Click on the pull request you want to review.
2. Click on the “Files Changed” Tab, and after reviewing the changes, click on the “Review Changes” button. You’ll see the option to “Comment”, “Approve”, or Request a change”. If there are updates you want your teammate to make -> choose “Request a change”. If you approve of the PR, -> choose “Approve”.

1. Once the PR has been approved, work with the Lead Project Manager to get the PR merged.

**Pulling latest changes**

1. After your teammate’s PRs are merged, you are going to want to pull their changes into your branch on your local. In Android Studio, click the “Gitl” tab again, and select “Pull…”. Change the branch you are pulling from to be the “main” branch. Click Pull.
2. If you currently have any not committed changes on your local, please commit your changes first before trying to pull.
3. If you face any merge issues, please follow the following [tutorial](https://www.youtube.com/watch?v=RxunYSzMNKM)**.**