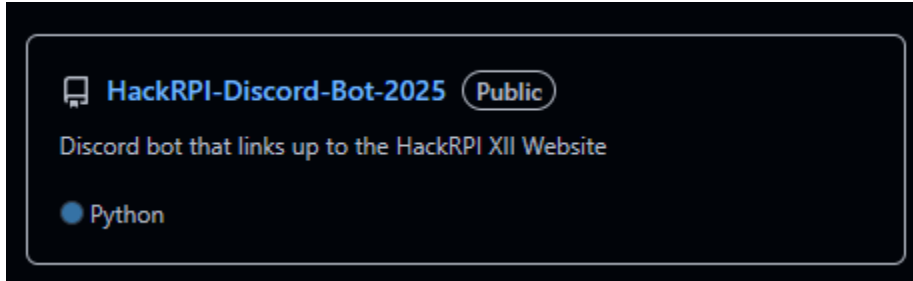


Getting Started

- (1) First, have python installed. You can use this [link](#) to install it if you need to. Here is a [link](#) as well for activating your virtual environment for python.
- (2) Also install [pip](#) for installing the packages to the project.
- (3) Clone the repo in the HackRPI projects list. Go to this [URL](#) to the HackRPI project, then click the HackRPI-Discord-Bot-2025 Repo.



- (4) Create a new folder where you want your local repository store and name it whatever you feel like. Navigate to that folder in your terminal of choice, and type `<git clone https://github.com/hack-rpi/HackRPI-Discord-Bot-2025.git>`

Then `cd` into HackRPI-Discord-Bot-2025 and open this file in your IDE (Assuming you guys use VS Code, PyCharm should work as well but that is not what I develop in so best of luck). In the bottom left, swap your branch to the one under my name (Sean Hyde). Then in the terminal run `<pip install -r requirements.txt>` this will install all of the packages necessary.

```
C:\Users\seanh\Desktop\HackRPI_Example>git clone https://github.com/hack-rpi/HackRPI-Discord-Bot-2025.git
Cloning into 'HackRPI-Discord-Bot-2025'...
remote: Enumerating objects: 20, done.
remote: Counting objects: 100% (20/20), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 20 (delta 5), reused 14 (delta 3), pack-reused 0 (from 0)
Receiving objects: 100% (20/20), 6.96 KiB | 1.39 MiB/s, done.
Resolving deltas: 100% (5/5), done.

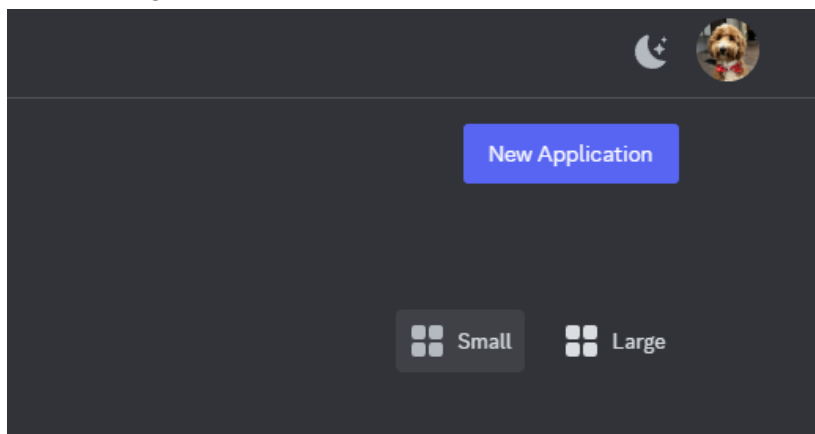
C:\Users\seanh\Desktop\HackRPI_Example>cd HackRPI-Discord-Bot-2025
C:\Users\seanh\Desktop\HackRPI_Example\HackRPI-Discord-Bot-2025>_
```

- (5) Create your own testing Discord server. Create an Administrator role that has all permissions (many of the commands we need to make are role permissive, so testing with roles is necessary). Make a few channels, a private channel called make-announcements, and another where you want the announcements to be posted. You will also have to go into your Discord settings and turn on developer tools so you can access the Channel ID's of the channels.

- (6) **MAKE SURE YOU ARE DOING THIS IN YOUR OWN BRANCH.** In the code, change the two channel IDs to the corresponding ones in your own testing server. Change the first CHANNEL_ID to the general channel id that is accessible by everyone, and the ANNOUNCEMENTS_CHANNEL_ID to the id of the private channel you have created. Once we have finally finished the bot these will probably turn into .env variables so nobody else can see them

```
# Discord Channel ID (Replace with actual channel ID)
CHANNEL_ID = 1337117398322647070
ANNOUNCEMENTS_CHANNEL_ID = 1342590654592847903
```

- (7) You must create your very own Discord bot that is linked to your account (we can't all use the same bot as there would be no way to test your own features you've created) If i find a better solution to this down the line, we can all migrate and do that instead, but for now this is the best I've got (if you find a better way to do this please let me know). Go to the [Discord Development Portal](#), and in the applications tab create your own application in the top right.



(8) Give it a name (“HackRPI Testing Bot”), and go to the OAuth tab on the left. Apply the following permissions (giving it admin for simplicity).

OAuth2 URL Generator

Generate an invite link for your application by picking the scopes and permissions it needs to function. Then, share the URL to others!

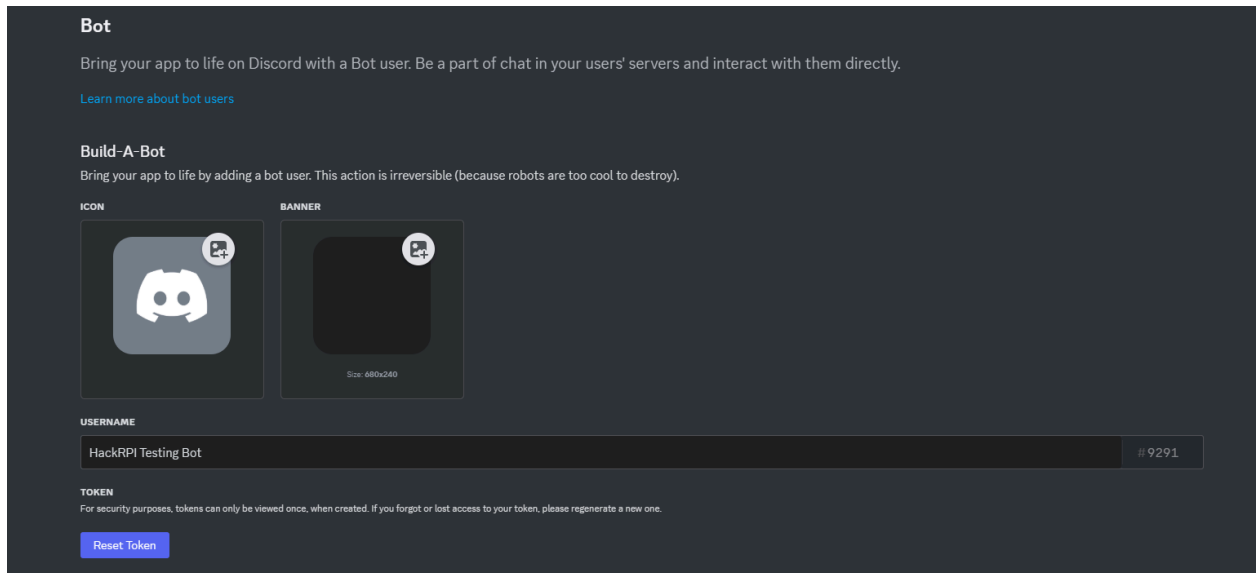
SCOPES

<input type="checkbox"/> identify	<input type="checkbox"/> email	<input type="checkbox"/> connections
<input type="checkbox"/> guilds	<input type="checkbox"/> guilds.join	<input type="checkbox"/> guilds.members.read
<input type="checkbox"/> guilds.channels.read	<input type="checkbox"/> gdm.join	<input checked="" type="checkbox"/> bot
<input type="checkbox"/> rpc	<input type="checkbox"/> rpc.notifications.read	<input type="checkbox"/> rpc.voice.read
<input type="checkbox"/> rpc.voice.write	<input type="checkbox"/> rpc.video.read	<input type="checkbox"/> rpc.video.write
<input type="checkbox"/> rpc.screenshare.read	<input type="checkbox"/> rpc.screenshare.write	<input type="checkbox"/> rpc.activities.write
<input type="checkbox"/> webhook.incoming	<input type="checkbox"/> messages.read	<input type="checkbox"/> applications.builds.upload
<input type="checkbox"/> applications.builds.read	<input type="checkbox"/> applications.commands	<input type="checkbox"/> applications.store.update
<input type="checkbox"/> applications.entitlements	<input type="checkbox"/> activities.read	<input type="checkbox"/> activities.write
<input type="checkbox"/> activities.invites.write	<input type="checkbox"/> relationships.read	<input type="checkbox"/> relationships.write
<input type="checkbox"/> voice	<input type="checkbox"/> dm_channels.read	<input type="checkbox"/> role_connections.write
<input type="checkbox"/> presences.read	<input type="checkbox"/> presences.write	<input type="checkbox"/> openid
<input type="checkbox"/> dm_channels.messages.read	<input type="checkbox"/> dm_channels.messages.write	<input type="checkbox"/> gateway.connect
<input type="checkbox"/> account.global_name.update	<input type="checkbox"/> payment_sources.country_code	<input type="checkbox"/> sdk.social_layer
<input type="checkbox"/> lobbies.write		
<input type="checkbox"/> applications.commands.permissions.update		

BOT PERMISSIONS

GENERAL PERMISSIONS	TEXT PERMISSIONS	VOICE PERMISSIONS
<input checked="" type="checkbox"/> Administrator	<input type="checkbox"/> Send Messages	<input type="checkbox"/> Connect
<input type="checkbox"/> View Audit Log	<input type="checkbox"/> Create Public Threads	<input type="checkbox"/> Speak
<input type="checkbox"/> Manage Server	<input type="checkbox"/> Create Private Threads	<input type="checkbox"/> Video
<input type="checkbox"/> Manage Roles	<input type="checkbox"/> Send Messages in Threads	<input type="checkbox"/> Mute Members
<input type="checkbox"/> Manage Channels	<input type="checkbox"/> Send TTS Messages	<input type="checkbox"/> Deafen Members
<input type="checkbox"/> Kick Members	<input type="checkbox"/> Manage Messages	<input type="checkbox"/> Move Members
<input type="checkbox"/> Ban Members	<input type="checkbox"/> Manage Threads	<input type="checkbox"/> Use Voice Activity
<input type="checkbox"/> Create Instant Invite	<input type="checkbox"/> Embed Links	<input type="checkbox"/> Priority Speaker
<input type="checkbox"/> Change Nickname	<input type="checkbox"/> Attach Files	<input type="checkbox"/> Request To Speak
<input type="checkbox"/> Manage Nicknames	<input type="checkbox"/> Read Message History	<input type="checkbox"/> Use Embedded Activities
<input type="checkbox"/> Manage Expressions	<input type="checkbox"/> Mention Everyone	<input type="checkbox"/> Use Soundboard
<input type="checkbox"/> Create Expressions	<input type="checkbox"/> Use External Emojis	<input type="checkbox"/> Use External Sounds
<input type="checkbox"/> Manage Webhooks	<input type="checkbox"/> Use External Stickers	
<input type="checkbox"/> View Channels	<input type="checkbox"/> Add Reactions	
<input type="checkbox"/> Manage Events	<input type="checkbox"/> Use Slash Commands	
<input type="checkbox"/> Create Events	<input type="checkbox"/> Use Embedded Activities	
<input type="checkbox"/> Moderate Members	<input type="checkbox"/> Use External Apps	
<input type="checkbox"/> View Server Insights	<input type="checkbox"/> Create Polls	
<input type="checkbox"/> View Server Subscription Insights		

- (9) Then take the generated URL at the bottom of the page, copy and paste it into your browser and invite it to the Discord Server you have made. Congrats you just made a Discord Bot.
- (10) Go to the Bot tab on the left, and you must reset the token. This is the token to the bot we have just created. Also at the bottom just toggle administrator. Copy this token and hold onto it. If you lose this token, you must reset it to get a new one.



The screenshot shows the 'Bot' creation page on the Discord developer portal. It includes a 'Build-A-Bot' section with instructions to add a bot user. There are fields for an 'ICON' (a Discord logo) and a 'BANNER' (a dark square). Below these is a 'USERNAME' field containing 'HackRPI Testing Bot' and a 'TOKEN' field. A 'Reset Token' button is visible at the bottom left of the token field.

Bot

Bring your app to life on Discord with a Bot user. Be a part of chat in your users' servers and interact with them directly.

[Learn more about bot users](#)

Build-A-Bot

Bring your app to life by adding a bot user. This action is irreversible (because robots are too cool to destroy).

ICON

BANNER

USERNAME

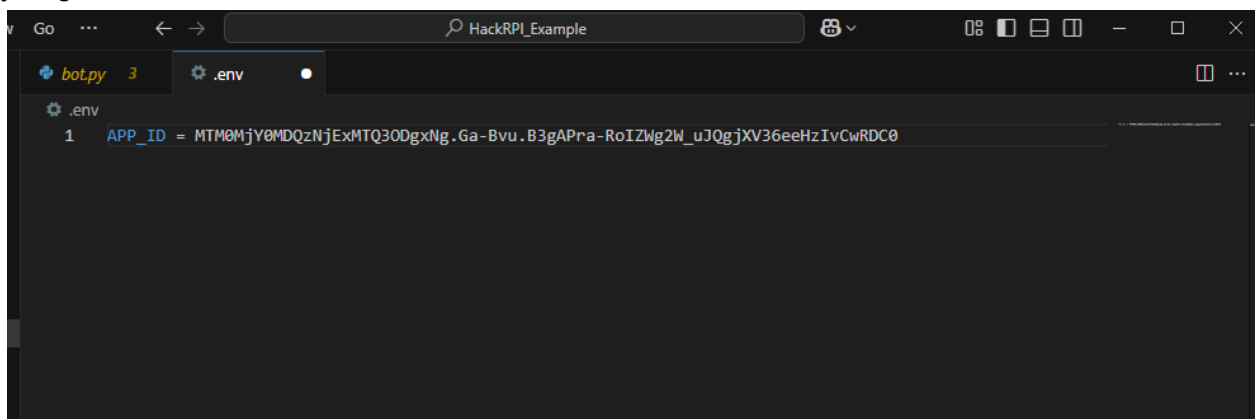
HackRPI Testing Bot #9291

TOKEN

For security purposes, tokens can only be viewed once, when created. If you forgot or lost access to your token, please regenerate a new one.

[Reset Token](#)

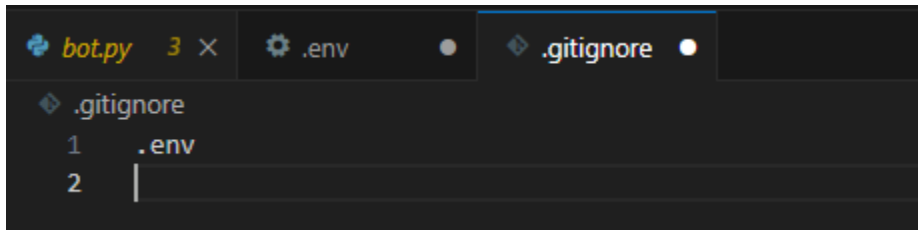
- (11) To actually make it use the code that we want it to, we need to set the Application ID to the bot in the code. In the project directory back in your IDE, create a new file called `<.env>`. In this file, create a new variable called `<APP_ID>` and assign it to the token you just got from Discord.



The screenshot shows a code editor window with a file named `.env` open. The file contains a single line of code: `APP_ID = MTM0MjY0MDQzNjExMTQ3ODgxNg.Ga-Bvu.B3gAPra-RoIZWg2W_uJQgjXV36eeHzIvCwRDC0`.

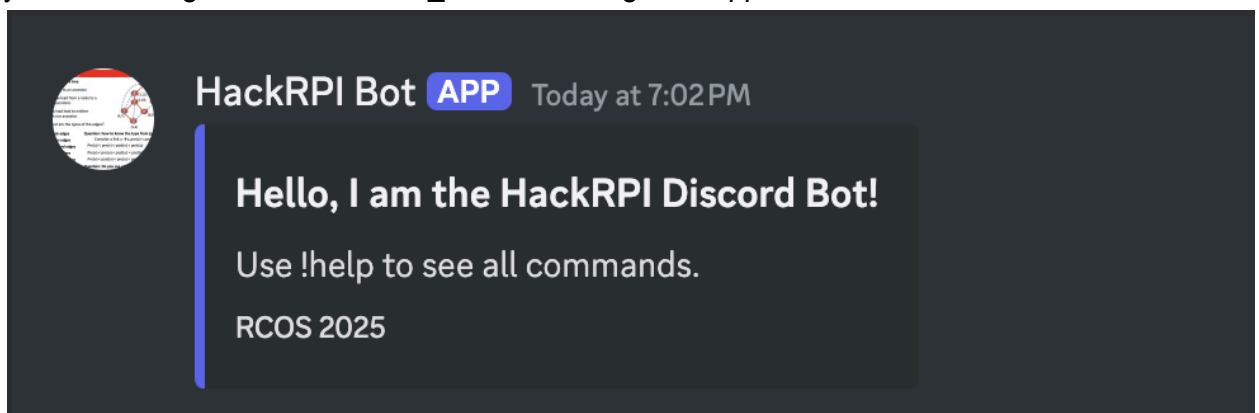
```
.env
1 APP_ID = MTM0MjY0MDQzNjExMTQ3ODgxNg.Ga-Bvu.B3gAPra-RoIZWg2W_uJQgjXV36eeHzIvCwRDC0
```

- (12) Also create a new file called `<.gitignore>` inside of this type `<.env>` If you do not do this, GitHub will freak out because you are attempting to push a secret token up to a public repo.

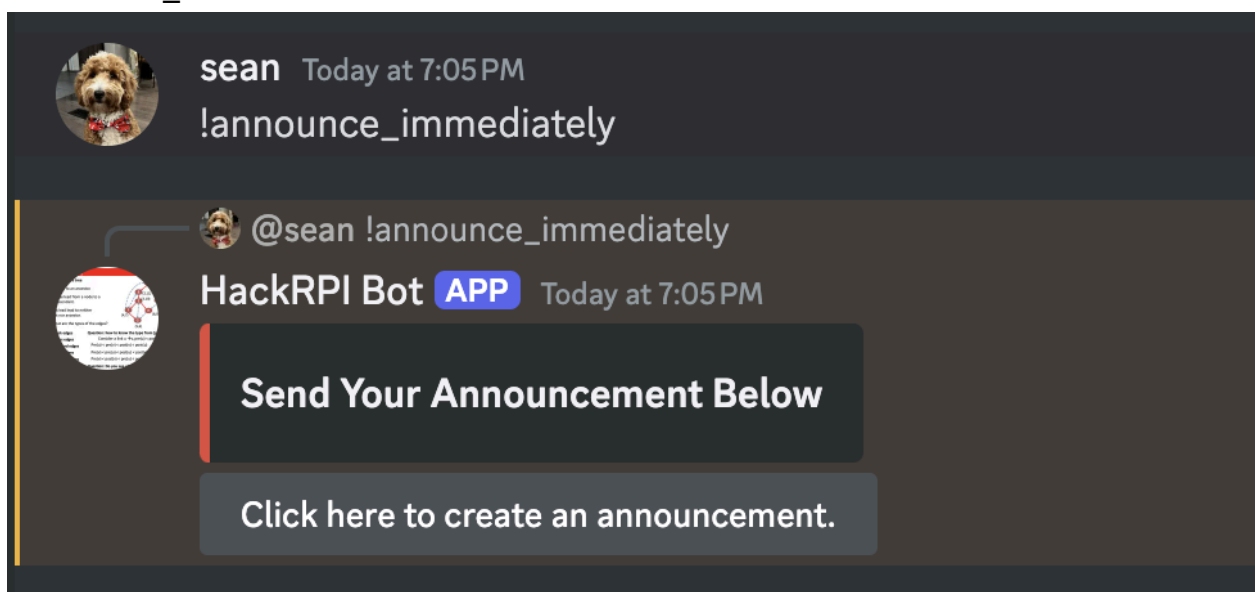


The screenshot shows a code editor with three tabs: `bot.py`, `.env`, and `.gitignore`. The `.gitignore` tab is active, showing a file with two lines: `1 .env` and `2` followed by a cursor.

- (13) The code should now be referencing your own bot you just created. Now click the play button or type `<python bot.py>` in your terminal to run the bot. In whatever channel you have assigned to `CHANNEL_ID` this message will appear.



- (14) You can now start developing on the bot! If you ever make changes, you must cancel your current session running the bot, and save your changes and run again. If you want to test if it's truly working, go into your private make-announcements channel and type `<!immediate_announcement>`



This doesn't quite work yet, but your bot should be able to use this command.

If you're having any trouble with set-up let me know and we can get it figured out.