### Project set up:

1. Clone the project into whatever directory you want on your machine.
2. Run npm install
3. Then, clone .env.example and rename to .env
   1. This is your local copy of the .env file. You will need to manually update this file as add secrets
   2. Do not check this file to source. Add it to your .gitignore file if it’s not already there.
   3. ***Check the Discord channel for secrets as we introduce them.***
4. For now, there isn’t a DB connected. That would be the first pull request merged to main after the initial commit. If you spin up the project using the command *npm run dev*, **and then you attempt to sign in on the demo web site, it will crash the server.**
5. Finally, I left all of the stock boilerplate code in the project. This is there to help you learn the framework. We’ll be removing this as we start to develop the site.

### Familiarize yourself with the project file structure.

Visit <https://create.t3.gg/en/folder-structure> for a breakdown on the file structure of the project.

TL;DR:

1. ~/pages is the front end portion of the project. tRPC calls such as queries or mutations invoked here are resolved on the server side
2. ~/server is the back end. tRPC resolvers are here. These functions talk to Prisma to fetch data from our DB on Railway.
   1. For each page component, we’ll have an accompanying .ts file where we resolve any data queries
3. ~/styles would likely be untouched until we want to start changing global CSS. Most CSS should be tailwind within their components.
   1. Mostly for global colors and text fonts.
4. ~/utils would house project wide utilities.

### Some of my personal notes on the documentation:

* I don’t think we’ll be utilizing the API stuff under **~/pages/** so don’t worry too much about that portion of the framework. That’s mainly for applications that are primarily APIs and not web servers.
* Most of the auth logic is already scaffolded for us. We’ll just need to make adjustments as we go.
* There are some blog posts we can refer to when we integrate Stripe
  + <https://nkrkn.me/writing/t3-stripe>

For VSCode users:

If you have errors on any .ts or .tsx file that is trying to import a module, make sure your editor is configured to use the workspace’s typescript version.

In the first image, on the bottom bar, look for the language VS code is using to interpret the file; the red arrow is pointing to it. Click on that and then on the second image, in the command palette input box, type in “*typescript”* and select typescript react. Then, in image 1, click on the curly braces that the blue arrow is pointing to and click on **select version.** Another command input box will open with two selections. You want to be using typescript version 5.4.2 or higher which would likely be the **workspace** version *if your copy of typescript has not been updated (like mine, which still is version 4.7).*



