

Welcome back...

To get the latest code so that you can follow along, go to github.com/WarwickAI/wai161 to get the latest code (in Tutorial 3)

Once downloaded, open the folder in VSCode and in the terminal run `npm install` to install the packages, and `npx prisma migrate dev` to update the database

Make sure to replace the Hugging Face access token with your own, and set-up the environment variables as describe in [.env-example](#)



WAI261- Introduction to Software Engineering

Tutorial 4 - Deployment and Finishing Touches

Course Breakdown

Tutorial 0: Setup VSCode, Git and NodeJS (Medium article)

Tutorial 1: Introduction to web development, setting up NextJS project, some UI stuff

Tutorial 2: More React; state management, component lifecycle (+hooks), basic querying

Tutorial 3: Creating and developing server in NodeJS

Tutorial 4: Finishing touches and deployment to Railway

Where are we?

- Set up a React project
- Added some UI libraries
- Created a basic UI
- Created message functionality
- Used a Hugging Face model to do NLP analysis and send a message in our chat
- Moved business logic to server and added authentication

Today we are going to...

- Deploy our website / ChatBot to a frontend cloud provider ([Railway](#))
- Deploy our frontend hosting + server to run in the [cloud](#).
- Some additions to our app

Deploying our Frontend

- Currently accessible locally (e.g. `localhost:3000`)
- Want it to be accessible globally (e.g. `my-chatbot-app.com`)
- Need to host our app with a hosting provider



Railway

- Very simple setup for React projects
- Decent free tier

Setup **Railway**

- First create an account on **Railway.app** (make sure to accept the terms)
- Next run **npm run build** to ensure Vercel won't have a problem building your app.
- Now, install **Railway's** CLI (set of tools) with:

```
npm install -g @railway/cli
```

Deploy to Railway

- Login to [Railway](#) locally using `railway login`
- Run `railway init` to setup a project (make sure to import environment variables in this step)
- Add the [Prisma](#) migrations to the `postinstall` script in `package.json`

```
"postinstall": "prisma generate && npx prisma migrate deploy"
```
- Create a domain on the [Railway](#) dashboard for our app
- Run `railway up` to deploy the current code to [Railway](#)

Your app should now be accessible on the domain you created

If you get environment variable issues, try putting them on your [Railway](#) app manually

But try logging in

Problems:

1. Being redirected to localhost (because `NEXT_AUTH` environment variable incorrect in [Railway](#)).
2. Discord not happy with `redirect_uri` (need to add our new domain to our [Discord](#) auth app)

Go ahead and fix these...

Now let's add some finishing touches

- Add profile pictures (could use Gravatar for this)
- Improve formatting e.g. colour chat boxes, resize message metadata.
- Add sending message on `<Enter>` key pressed
- **Advanced:** make chat-room functionality with room IDs

What's Next...

- Deploy your own project/idea, like a blog or portfolio
- Become part of a WAI project