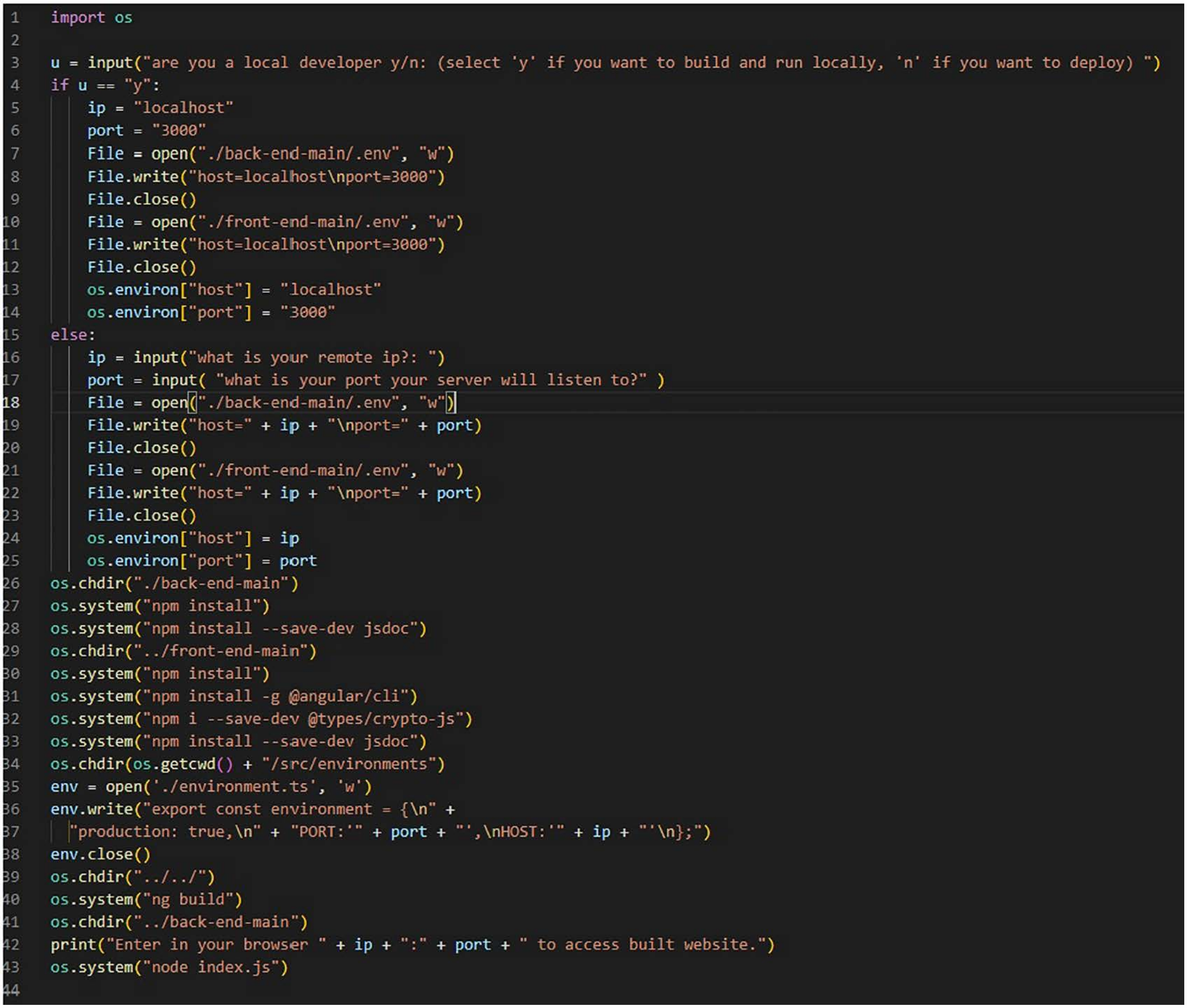
Annotation 9

Title: Automated Setup Script

Description: This artifact is a screenshot of ElectronChat setup script. It represents a team activity to reduce redundant movements to launch a web app for both local and server side cases. The source code presented in the artifact was initially created by Philip Marshal, and further developed by all members of ElectronChat team (James Marshall, Timothy Trusive, and Zakhaddin Khalidov). Early in our development we ran into issues that were stemming from development environment variability. It became evident that we all had different methods and techniques for installing packages, using different local IP and port addresses, and launching the app. To circumvent these issues we had to standardize our methods and automate it so there is no deviation. This exercise and artifact is a great example of software development adversity and the resolution.

Reflection: As a result of this exercise I learned the utility of scripting and automating by removing variability in developer environments. When I started working on the ElectronChat app, I belonged to the back-end team working exclusively with NodeJS on my local machine. Once the app was in a functional state, I had no idea how to build the Angular front end. I was able to communicate with my team to resolve the knowledge curve, but for developers that only want to work on the back end it would of been an annoying nuance that they shouldn't have to deal with. I believe the setup script is a key development in the ElectronChat app that helps reduce dev environment errors and repetitive actions to run the app.

Artifact 9



```python
import os

u = input("are you a local developer y/n: (select 'y' if you want to build and run locally, 'n' if you want to deploy) ")
if u == "y":
    ip = "localhost"
    port = "3000"
    File = open("./back-end-main/.env", "w")
    File.write("host=localhost\nport=3000")
    File.close()
    File = open("./front-end-main/.env", "w")
    File.write("host=localhost\nport=3000")
    File.close()
    os.environ["host"] = "localhost"
    os.environ["port"] = "3000"
else:
    ip = input("what is your remote ip?: ")
    port = input( "what is your port your server will listen to?" )
    File = open("./back-end-main/.env", "w")
    File.write("host=" + ip + "\nport=" + port)
    File.close()
    File = open("./front-end-main/.env", "w")
    File.write("host=" + ip + "\nport=" + port)
    File.close()
    os.environ["host"] = ip
    os.environ["port"] = port
os.chdir("./back-end-main")
os.system("npm install")
os.system("npm install --save-dev jsdoc")
os.chdir("../front-end-main")
os.system("npm install")
os.system("npm install -g @angular/cli")
os.system("npm i --save-dev @types/crypto-js")
os.system("npm install --save-dev jsdoc")
os.chdir(os.getcwd() + "/src/environments")
env = open('./environment.ts', 'w')
env.write("export const environment = {\n" +
    "production: true,\n" + "PORT:'" + port + "',\nHOST:'" + ip + "'\n};")
env.close()
os.chdir("../../")
os.system("ng build")
os.chdir("../back-end-main")
print("Enter in your browser " + ip + ":" + port + " to access built website.")
os.system("node index.js")

```