**CSC 667/867**
**Spring 2018**

**<u>Term Project</u>**
**Milestone 2: Heroku Deployment and Database Setup**

**<u>Team-I Members</u>**
Razmik Hakobyan
Kimyou By
Jimmy He

**<u>Team-i Github Repository</u>**
https://github.com/sfsu-csc-667-spring-2018/term-project-team-i

**<u>Heroku Team-I-Chess Application</u>**
**https://team-i-chess.herokuapp.com/**

# Table of Contents

# Problems and Solutions

---

Team-I consists of three members, two of whom are using the Windows operating system. Unfortunately Windows may be the cause behind the majority of our problems. The problems we ran into were the following..

# Nodejs/Expressjs Related

### Nodemon

The issue is probably not with the nodemon package, but rather with Windows. We have added the following line to the 'scripts' section of our package.json file:

```
"start:dev": "DEBUG=term-project-team-i:* nodemon ./bin/www"
```

And we have tried executing our express application using the following command:

```
>npm run start:dev
```

Unfortunately the error states the following;

```
>'DEBUG' is not recognized as an internal or external command
```

As it says, "DEBUG" is not available. There doesn't seem to be anyway around it. However, one of our members are using a macOS and have executed the "DEBUG" command without any issue. Afterward, the full command of `"start:dev"` was also executed without error.

Regardless of the 'DEBUG' command, we also tried executing 'nodemon' directly but that too wasn't recognized as a command either. It was only until it was installed globally that it was recognized as a command.

### Dotenv

The dotenv module was designed to read values in the .env file. We added the following code as instructed to the top of app.js…

```
if(process.env.NODE_ENV === 'development') {
    require("dotenv").config();
}
```

However the 'require' line didn't execute. It was only until after setting the environment variable on Windows to development that it did.

```
>set NODE_ENV=development
```

### Sequelize

The issue with the sequelize module was much similar to nodemon. The command of sequelize was no recognized until we've installed the module globally.

Afterward everything worked as intended. Sequelize, sequelize-cli, and nodemon issues might all be because of Windows only.

# Team Development Setup

---

Team-I development setup is relatively simple. There are two repositories to consider: the Heroku app's repository of which the application is deployed from and the class team's repository of which our code can be viewed on the web. However, every other member who isn't the owner of the Heroku app is marked as a Collaborator on the app's settings which allows them to push directly to the Heroku app's repository.

For testing, we will test locally on own branches and computers. For peer review we'd simply switch to the local branch and run it in order to verify its execution. Afterward, the code will then be merged into the master branch for both Heroku apps' and the class repository.

The commands to deploy code on Heroku can be found here…

Link: https://devcenter.heroku.com/articles/git