# Development environment

## Database

External to environment. Referenced though code.

## Root files

### docker-compose.yml

Creates and launches working environment.

Creation, through

docker-compose up –d

Start through

docker-compose start

Stop through

docker-compose stop  
.gitignore

Crucial in controlling which files are tracked by git. Also which files are pushed to, and pulled from the remote (origin) repository. See sections on “How to work”.

### package.json

Lists all the files required for developing JavaScript. Will be used within the “How to work” section.

### webpack.config.json

Will be used to package all source files for JavaScript and CSS into sing bundles. These bundles with be contained within the “dist” directory.

## How to work (Developer)

This section refers to the file devprocess.pdf found in this directory.  
Begin

If this is a new project

git init

If you are working in an existing project

git pull <location of project>

Next, you will need to make sure that you have all the node development packages required for the project. They are listed in the file package.json. This will be achieved by running  
npm install

### Commit and copy

Webpack with bundle all JS and CSS files into the “dist” directory for display in the browser. The project will also use PHP files which need to be available in the “dist” directory after being committed into the git repository. In order to commit PHP files to the git repository and copy them to the “dist” directory use the examples below.

#### Windows

xcopy src\\*.php dist /i /d /y /s; git add ./src/.; git commit -m “Example message”

#### GNU/Linux (needs testing)

cp -ruT src/\*.php dist; git add ./src/.; git commit -m “Example message”