

Start document PiJukebox

Before starting to build the project, make sure all versions are equal to the ones used.

IntelliJ

IntelliJ IDEA 2018.3.3 (Ultimate edition)
Build: #IU-183.5153.38 (January 9, 2019)

Polymer

Polymer 3.1.0

Polymer-CLI

Polymer 1.9.1 or below 2.0.0

Java JRE

Java JRE 1.8.0_152-release

XAMPP

XAMPP Control Panel version 3.2.2

MySQL

Version: 10.1.8-MariaDB

NodeJS

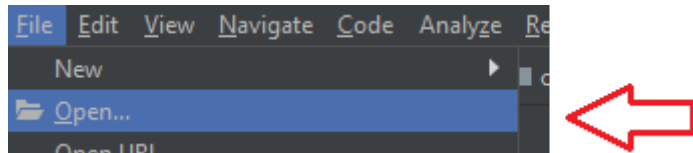
Download latest version from 'Current':
<https://nodejs.org/en/download/current/>

NPM – Polymer

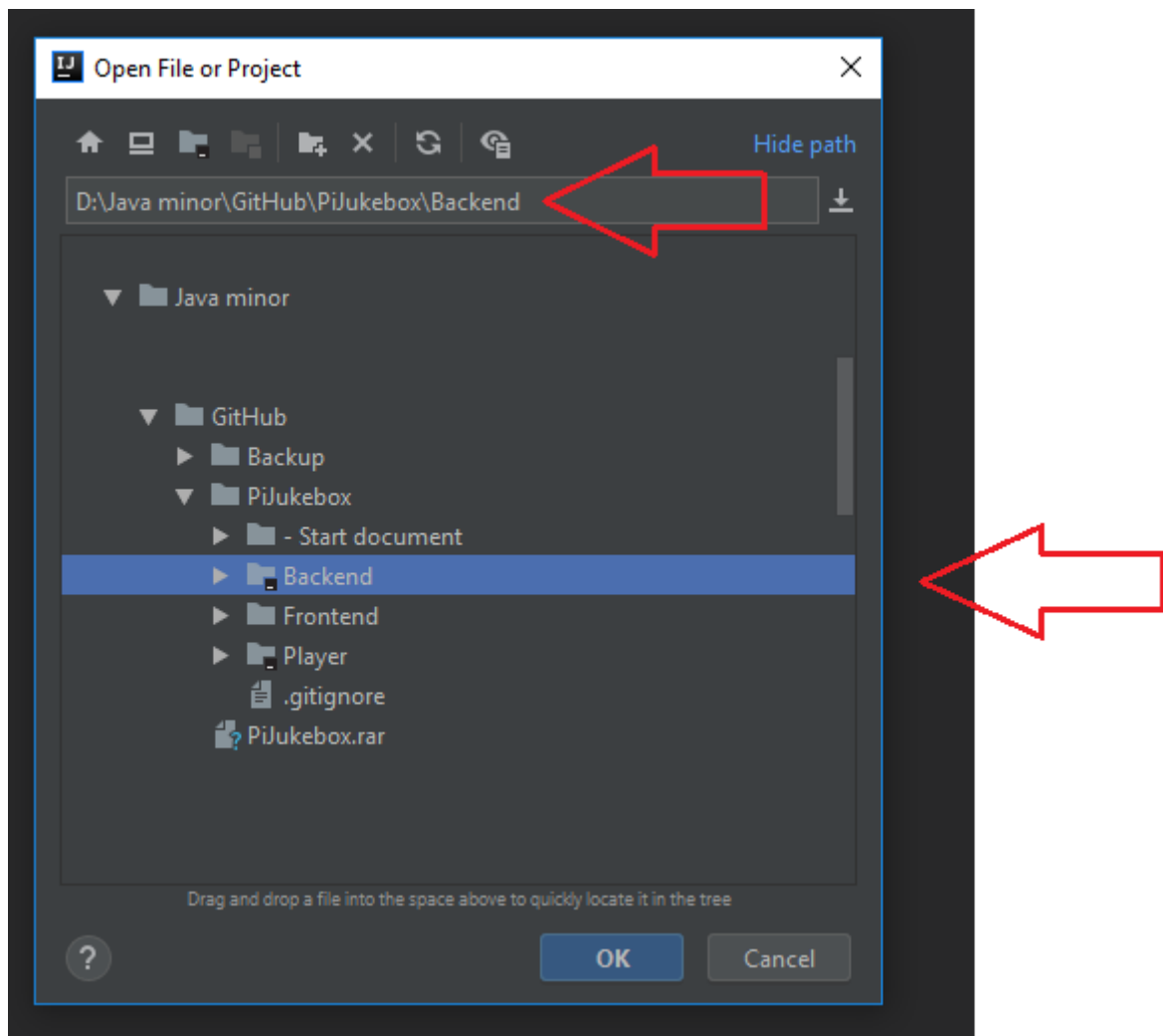
Version: 6.8.0
npm install -g polymer-cli

How to build the back-end

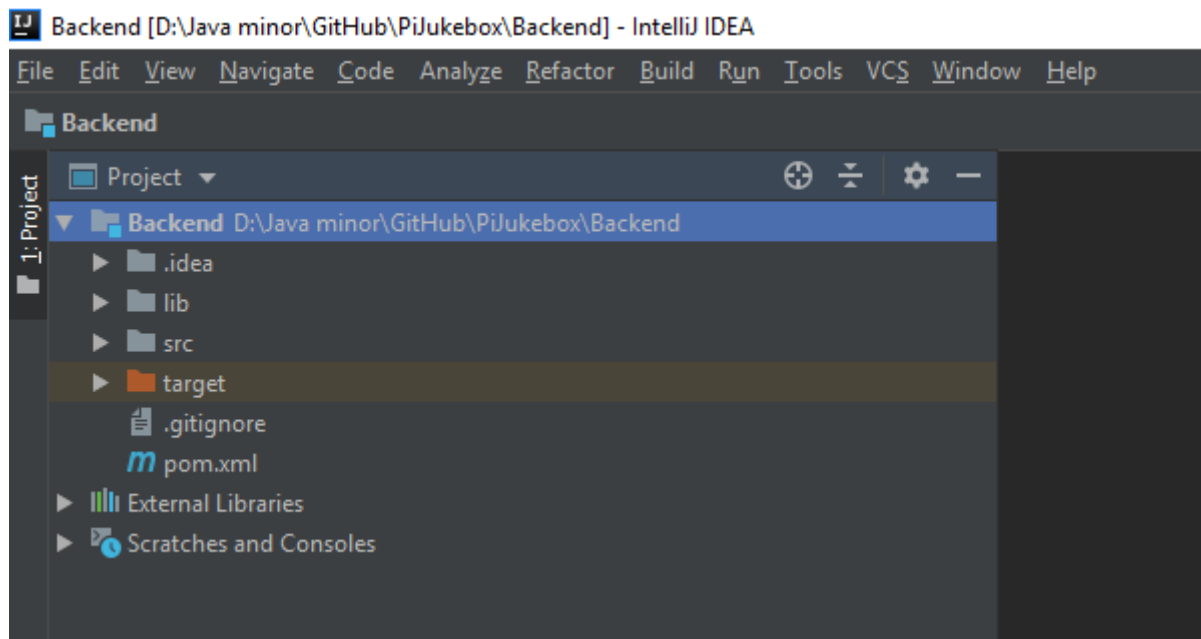
Open up the IntelliJ IDEA, go to:



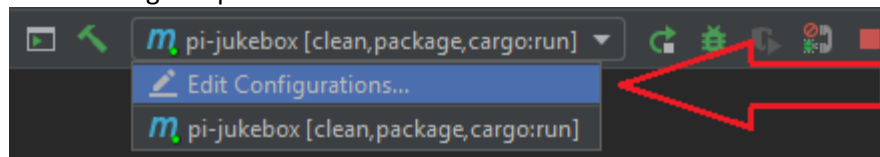
Then browse to the location where the project is saved:



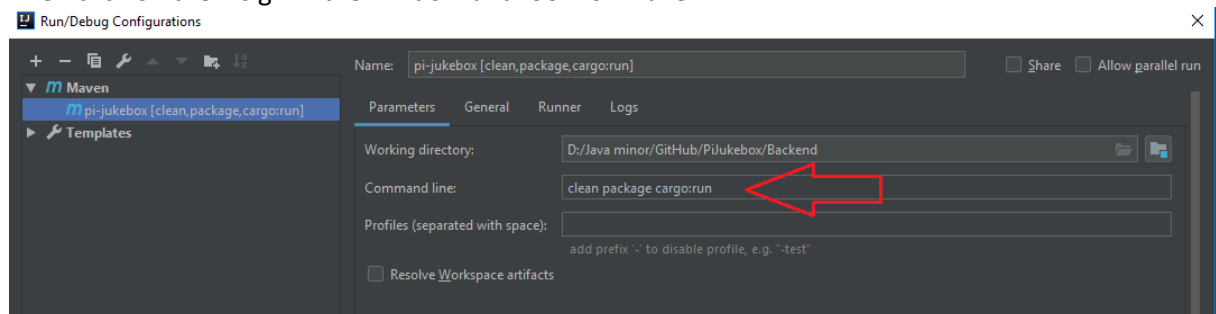
The moment the project is loaded in it'll look similar if not equal to the next image:



If the project cannot be run at this moment, thus it needs to be configured. Go to the right top corner and click on:



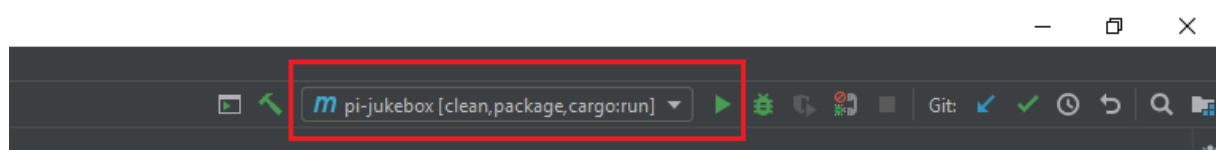
Then click on the + sign in the window and look for Maven.



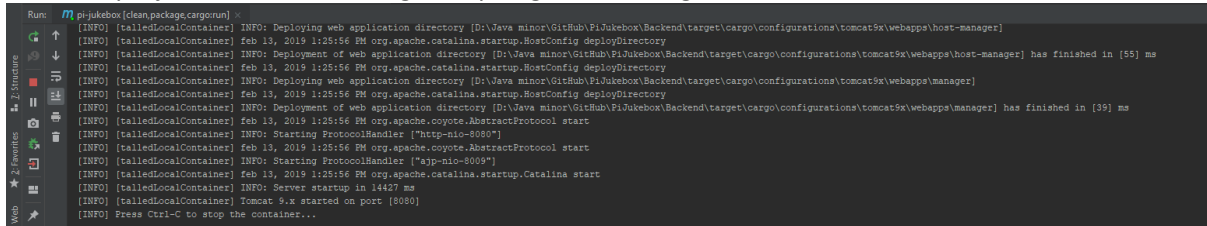
When maven is added make sure the Command line states: clean package cargo:run

This way the IDEA knows it's talking to an Maven Project and knows how to handle the POM file in the PiJukebox project. **If it asks to auto-import Maven related dependencies, click on it**, this way it'll download all the dependencies we used stated in the POM file. If the project is asking to use a valid SDK, use Java JRE 1.8.0_152-release

Now the project can be run by pressing the Play button: (Next image as reference)



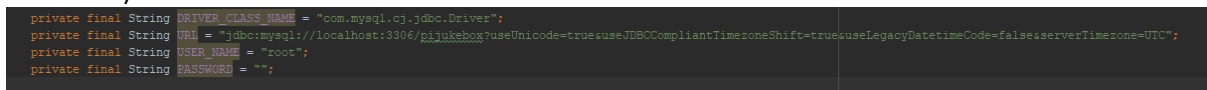
Now the project will start building, compiling and running:



```
Run: pi-jukebox[clean,package,cargo:run] x
[INFO] [talledLocalContainer] INFO: Deploying web application directory [D:\Java\minor\Github\pi-jukebox\Backend\target\cargo\configurations\tomcat9x\webapps\host-manager]
[INFO] [talledLocalContainer] feb 13, 2019 1:25:56 PM org.apache.catalina.startup.HostConfig deployDirectory
[INFO] [talledLocalContainer] INFO: Deployment of web application directory [D:\Java\minor\Github\pi-jukebox\Backend\target\cargo\configurations\tomcat9x\webapps\host-manager] has finished in [55] ms
[INFO] [talledLocalContainer] feb 13, 2019 1:25:56 PM org.apache.catalina.startup.HostConfig deployDirectory
[INFO] [talledLocalContainer] INFO: Deploying web application directory [D:\Java\minor\Github\pi-jukebox\Backend\target\cargo\configurations\tomcat9x\webapps\manager]
[INFO] [talledLocalContainer] feb 13, 2019 1:25:56 PM org.apache.catalina.startup.HostConfig deployDirectory
[INFO] [talledLocalContainer] INFO: Deployment of web application directory [D:\Java\minor\Github\pi-jukebox\Backend\target\cargo\configurations\tomcat9x\webapps\manager] has finished in [39] ms
[INFO] [talledLocalContainer] feb 13, 2019 1:25:56 PM org.apache.coyote.AbstractProtocol start
[INFO] [talledLocalContainer] INFO: Starting ProtocolHandler ["http-nio-8080"]
[INFO] [talledLocalContainer] feb 13, 2019 1:25:56 PM org.apache.coyote.AbstractProtocol start
[INFO] [talledLocalContainer] INFO: Starting ProtocolHandler ["ajp-nio-8009"]
[INFO] [talledLocalContainer] feb 13, 2019 1:25:56 PM org.apache.catalina.startup.Catalina start
[INFO] [talledLocalContainer] INFO: Server startup in 14437 ms
[INFO] [talledLocalContainer] Tomcat 9.x started on port [8080]
[INFO] Press Ctrl-C to stop the container...
```

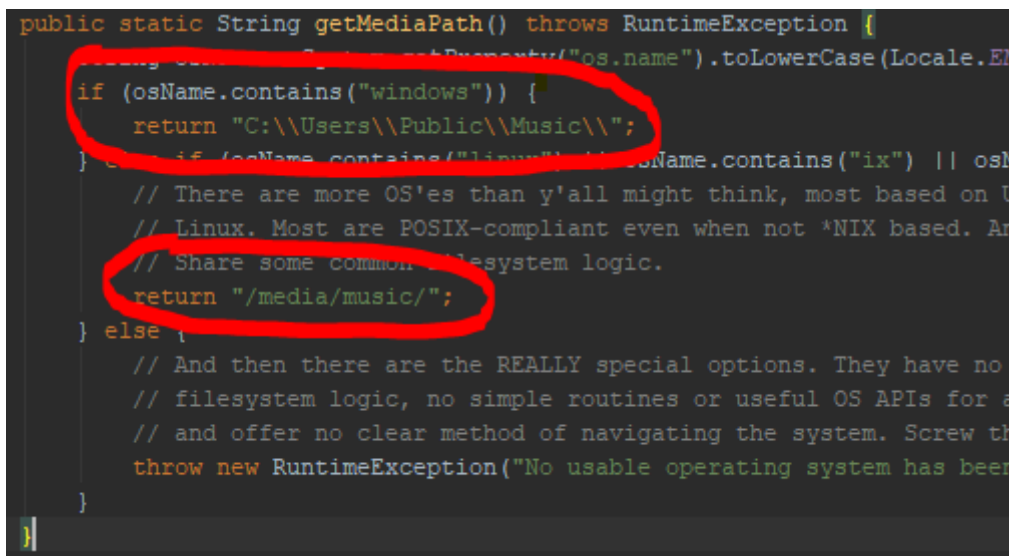
The moment it's showing (Press Ctrl-C to stop the container...) it's done building and running. There is a high probability small things aren't configured correctly, for example the Database and music path.

To change the Database to the one active now, open the project file "DatabaseConfig" through IntelliJ and check if the USER_NAME and PASSWORD are correctly set. (Check next image for reference)



```
private final String DRIVER_CLASS_NAME = "com.mysql.cj.jdbc.Driver";
private final String URL = "jdbc:mysql://localhost:3306/pi-jukebox?useUnicode=true&useJDBCCompliantTimezoneShift=true&useLegacyDatetimeCode=false&serverTimezone=UTC";
private final String USER_NAME = "root";
private final String PASSWORD = "";
```

The music path **doesn't have to** be edited to the correct location. On a windows system it'll make a call to the folder: "C:\Users\Public\Music\ ". But if you're using Linux, it'll go to: "/media/music/"



```
public static String getMediaPath() throws RuntimeException {
    String osName = System.getProperty("os.name").toLowerCase(Locale.ENGLISH);
    if (osName.contains("windows")) {
        return "C:\\Users\\Public\\Music\\";
    } else if (osName.contains("linux") || osName.contains("ix") || osName.contains("mac")) {
        // There are more OS'es than y'all might think, most based on UNIX
        // Linux. Most are POSIX-compliant even when not *NIX based. And
        // Share some common filesystem logic.
        return "/media/music/";
    } else {
        // And then there are the REALLY special options. They have no
        // filesystem logic, no simple routines or useful OS APIs for a
        // and offer no clear method of navigating the system. Screw them.
        throw new RuntimeException("No usable operating system has been found");
    }
}
```

Due to the high volume needed for the songs the team decided to use Google Drive to separate it from the Github branch. Hence the songs can be downloaded through the following link:

https://drive.google.com/drive/folders/1sdjMxZVXQtiJAYcve6CkkSFFfXI_pISi?usp=sharing

How to build the front-end

Before you can work properly with the front-end it needs to be setup and configured.

Download the latest version of Node.js from the 'current' branch.

To make sure you have it installed, open CMD and type: "node --version".


Also make sure you have NPM installed with the command: "npm install -g polymer-cli"

To see if you have the latest version installed, open CMD again and type: "npm --version".

If you see the versions showing up from both of them you're good to go on building the front-end.

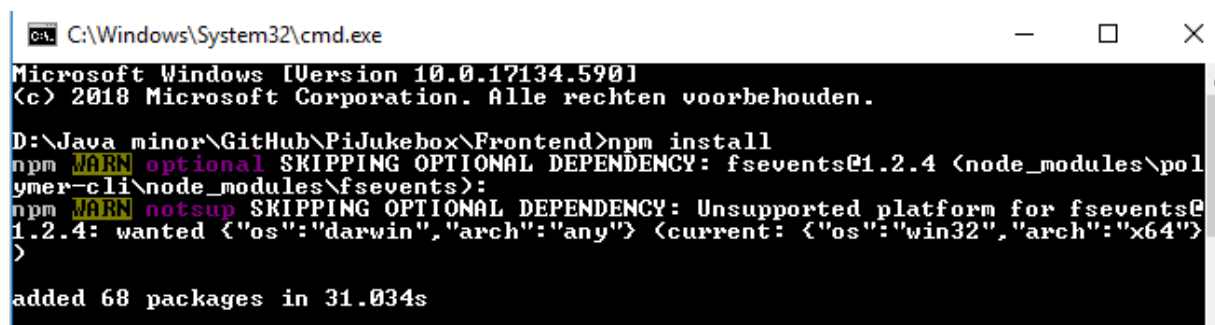
Navigate to the place you have the project saved and go to the folder called "frontend".

When you are in the folder "frontend" open CMD and use the command: "npm install"



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.17134.590]
(c) 2018 Microsoft Corporation. Alle rechten voorbehouden.
D:\Java minor\GitHub\PiJukebox\Frontend>npm install
```

This is fetching all the latest versions and when it's done, it should look similar:



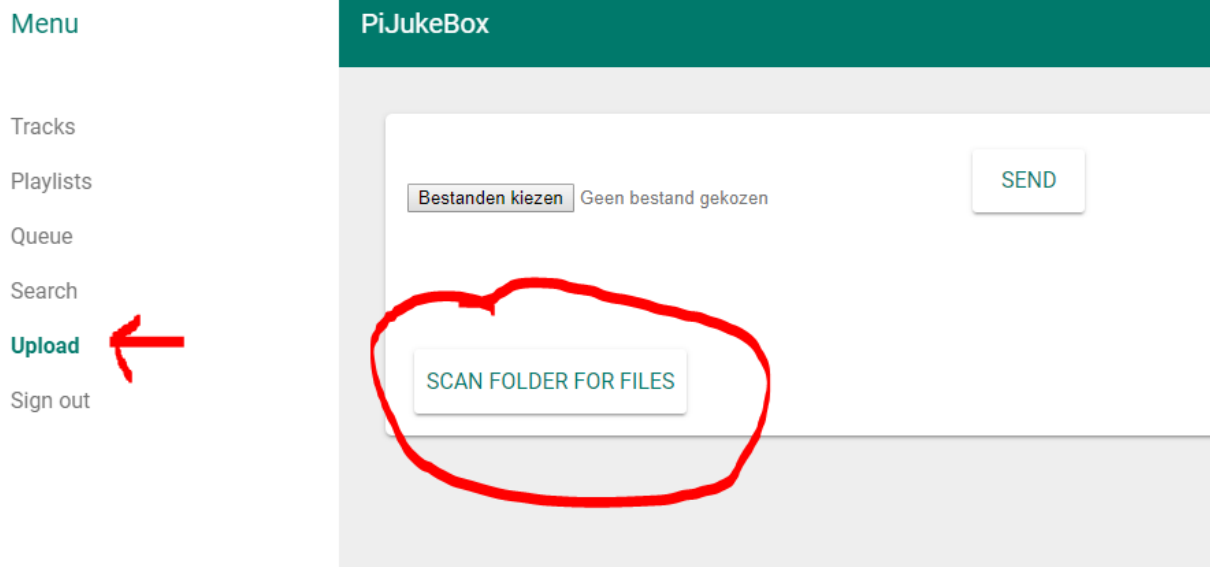
```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.17134.590]
(c) 2018 Microsoft Corporation. Alle rechten voorbehouden.
D:\Java minor\GitHub\PiJukebox\Frontend>npm install
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.4 <node_modules\polymer-cli\node_modules\fsevents>:
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.4: wanted <"os": "darwin", "arch": "any"> <current: <"os": "win32", "arch": "x64"> >
added 68 packages in 31.034s
```

When that's done, you can use the command in CMD: "Polymer serve" , it will look like this:



```
polymer
Microsoft Windows [Version 10.0.17134.590]
(c) 2018 Microsoft Corporation. Alle rechten voorbehouden.
D:\Java minor\GitHub\PiJukebox\Frontend>polymer serve
info: Files in this directory are available under the following URLs
applications: http://127.0.0.1:8081
reusable components: http://127.0.0.1:8081/components/polymer-starter-kit/
```

Before you try to use the function “SCAN FOLDER FOR FILES” under “Upload” (Picture for reference)

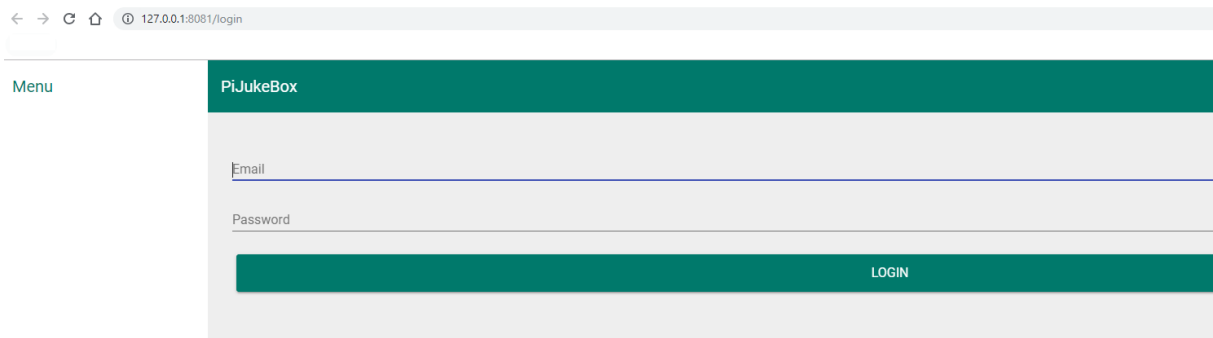


You need to create an Upload folder to your public folder.

Name for reference: “C:\Users\Public\Uploads” , the function will search in that folder for newly added files.

HAVE FUN TESTING! 😊

Open your browser now, and navigate to: <http://127.0.0.1:8081> (Default polymer)



You can create your own user through the database or login with:

Email: john.doe@example.com

Pass: johndoe