

# Refugee Bnb – Installation

Thank you for trying out Refugee Bnb. Follow these simple steps to set up the system and test it out.

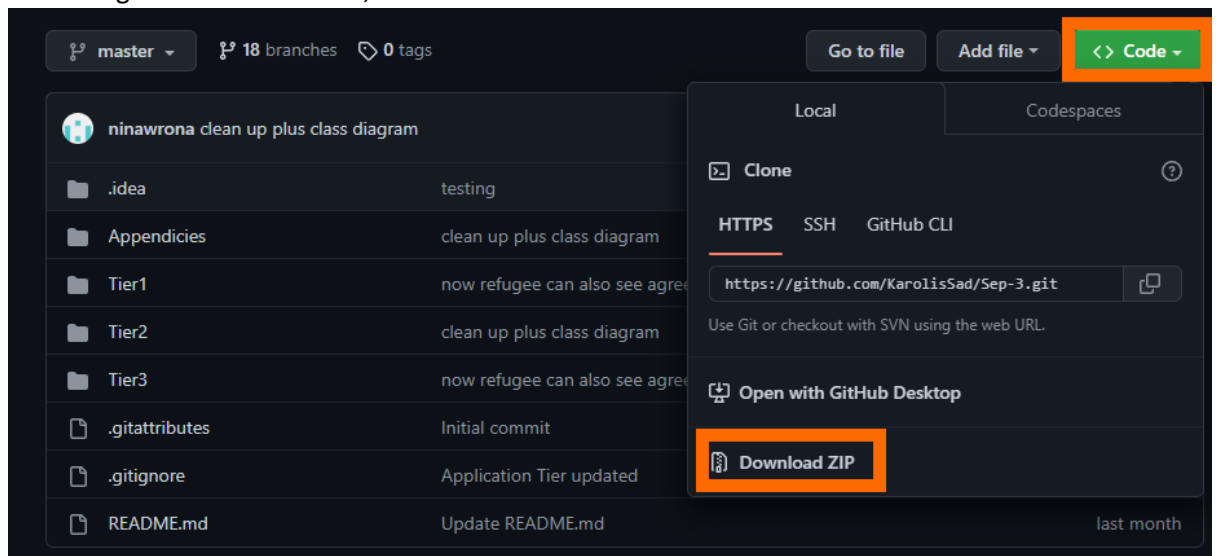
## Pre-requisites:

To install and run this version of Refugee Bnb you will need the following:

- JetBrains IntelliJ installed on your computer
- JetBrains Rider installed on your computer
- PostgreSQL installed on your computer
- An internet browser

## Download sources:

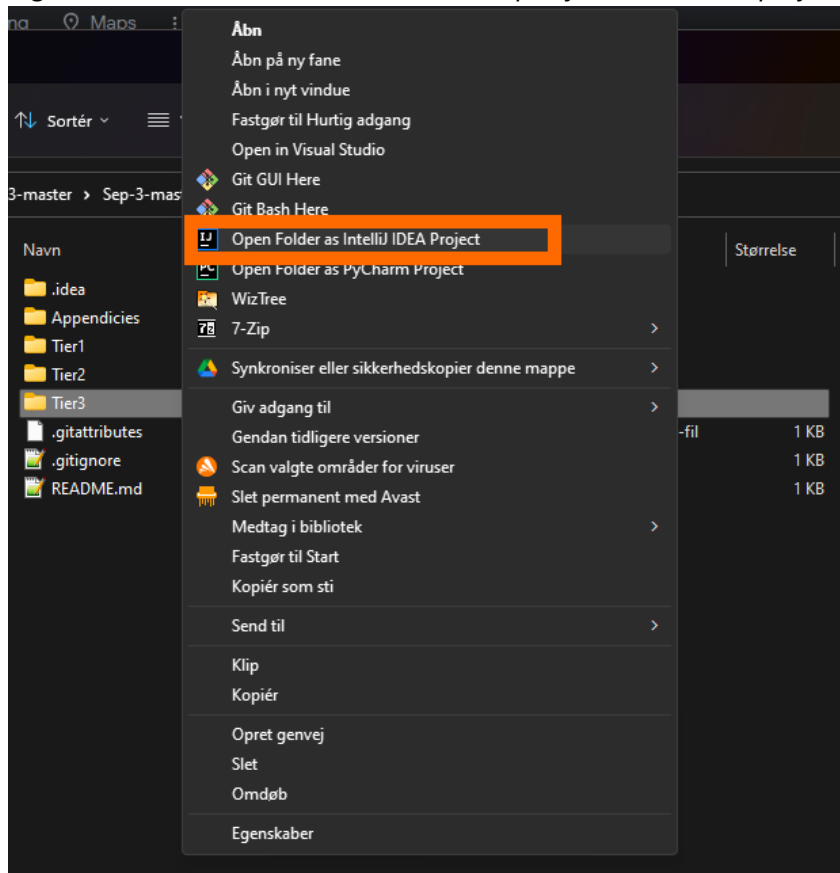
- Follow this link to find the source code: [GitHub](#)
- Click the green “Code” button, and click Download ZIP



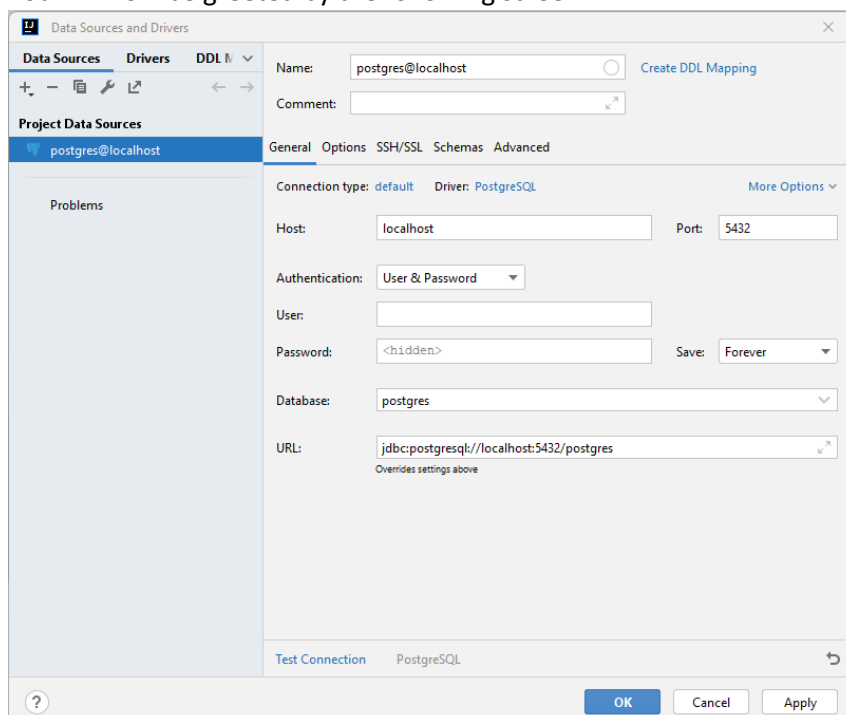
- Unpack the archive to a folder of your choosing

## Database setup:

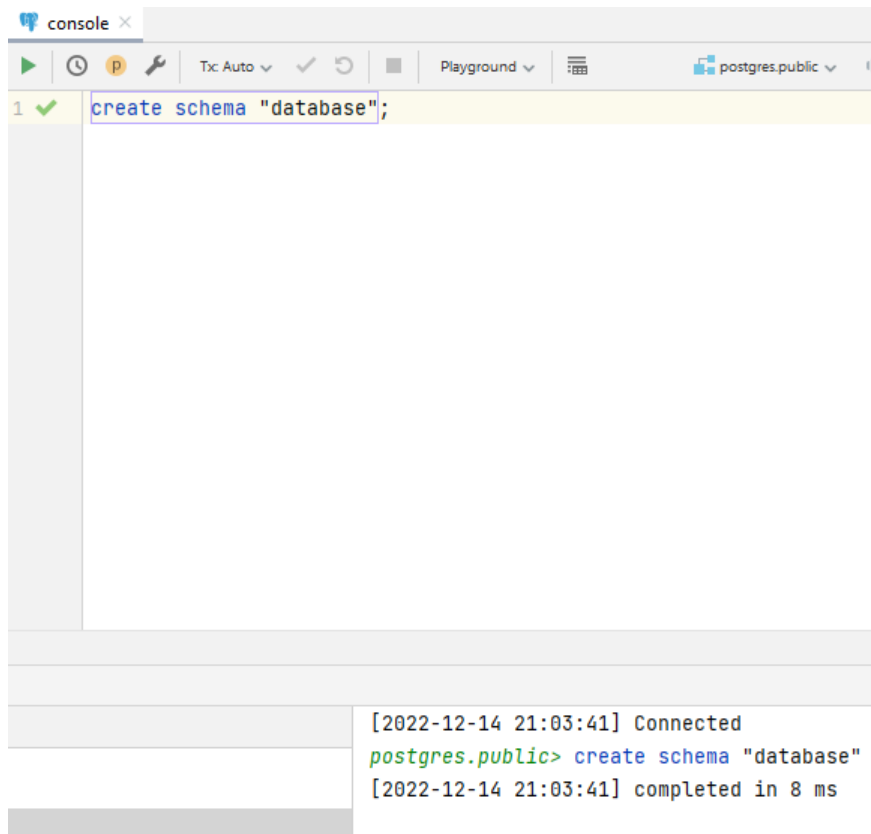
- Right click the “Tier 3” folder and select “Open folder as IntelliJ project”.



- Click the “Database” button on the right side of the screen in *IntelliJ*
- Press the “New” button in the upper right corner of the window that opens up
- Select “Datasource” and then “PostgreSQL” in the menu
- You will now be greeted by the following screen



- In the “User” field, type in *postgres*
- In the “Password” field, type in your personal PostgreSQL password.
- Press “Test Connection” in the lower left corner of the window to test your connection. After that, click apply.
- In the console tab that opens, type in: *create schema “database”;* and click the green *Execute* button. It should look like this afterwards



The screenshot shows a PostgreSQL console window with a toolbar at the top containing icons for running, refreshing, and other actions. Below the toolbar, the SQL command `create schema "database";` is entered in the input field. The console output shows the command was executed successfully, with a timestamp and duration: `[2022-12-14 21:03:41] Connected postgres.public> create schema "database" [2022-12-14 21:03:41] completed in 8 ms`.

- In the project view in the left side of the screen, navigate to: “Tier3/src/main/ressources” and open the file *application.properties*.
- Edit line 7 of this file, *##YOUR PASSWORD HERE##* to be your personal PostgreSQL password instead.



The screenshot shows a code editor with the `application.properties` file. The file contains the following properties:

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
spring.datasource.url= jdbc:postgresql://localhost:5432/postgres
spring.jpa.properties.hibernate.default_schema = database
spring.datasource.username= postgres
spring.datasource.password= ##YOUR PASSWORD HERE##

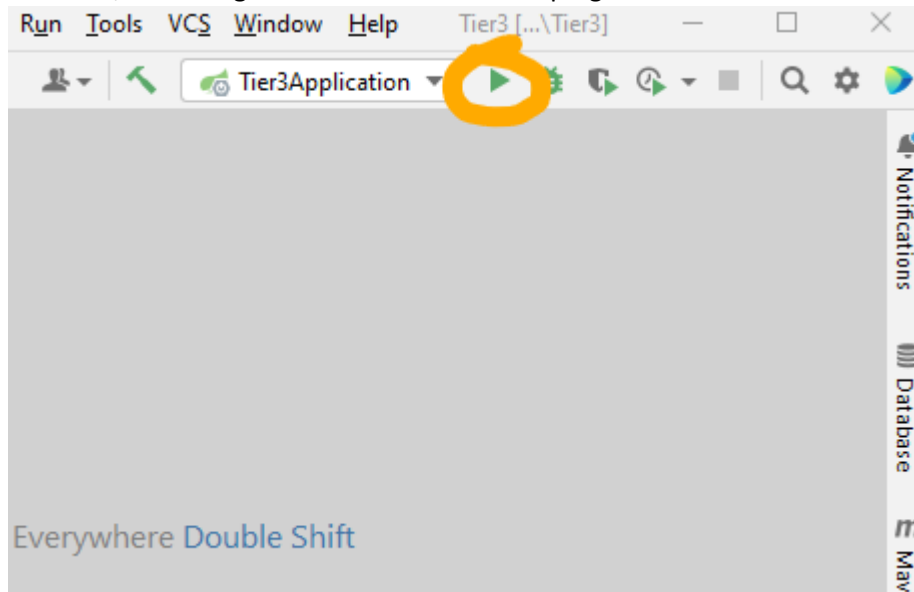
spring.jpa.properties.hibernate.jdbc.lob.non_contextual_creation= true
spring.jpa.properties.hibernate.dialect= org.hibernate.dialect.PostgreSQLDialect

spring.jpa.hibernate.ddl-auto= update

```

## Running the back-ends:

- In IntelliJ, click the green *Run* button in the top right corner of the screen to run the data tier



- With the data tier running, go to the folder where you unzipped the source-code and open tier 2 the same way you did tier 3 (right click the folder and choose “*Open folder as IntelliJ project*”)
- IntelliJ should now ask you where you would like to open the project: Select “*New Window*”.
- Run the Tier2 project like you did for Tier3 (click the green *Run* button in the top right corner of the screen)
- With both IntelliJ projects running, open JetBrains Rider and from the menu, select “*Open*”.
- In the window that opens, navigate to the folder where you unpacked the archive and select the folder called *Tier1* and press *Select Folder*.
- In the upper right corner of the screen, click the green “*Run*” button as you did in IntelliJ
- Now all three tiers are running, and you are ready to try the system.
- After you run Tier1, it should automatically open in your default browser. If it does not, or if you want to use a different browser, simply enter the following in the address-bar:  
[localhost:7073](http://localhost:7073)