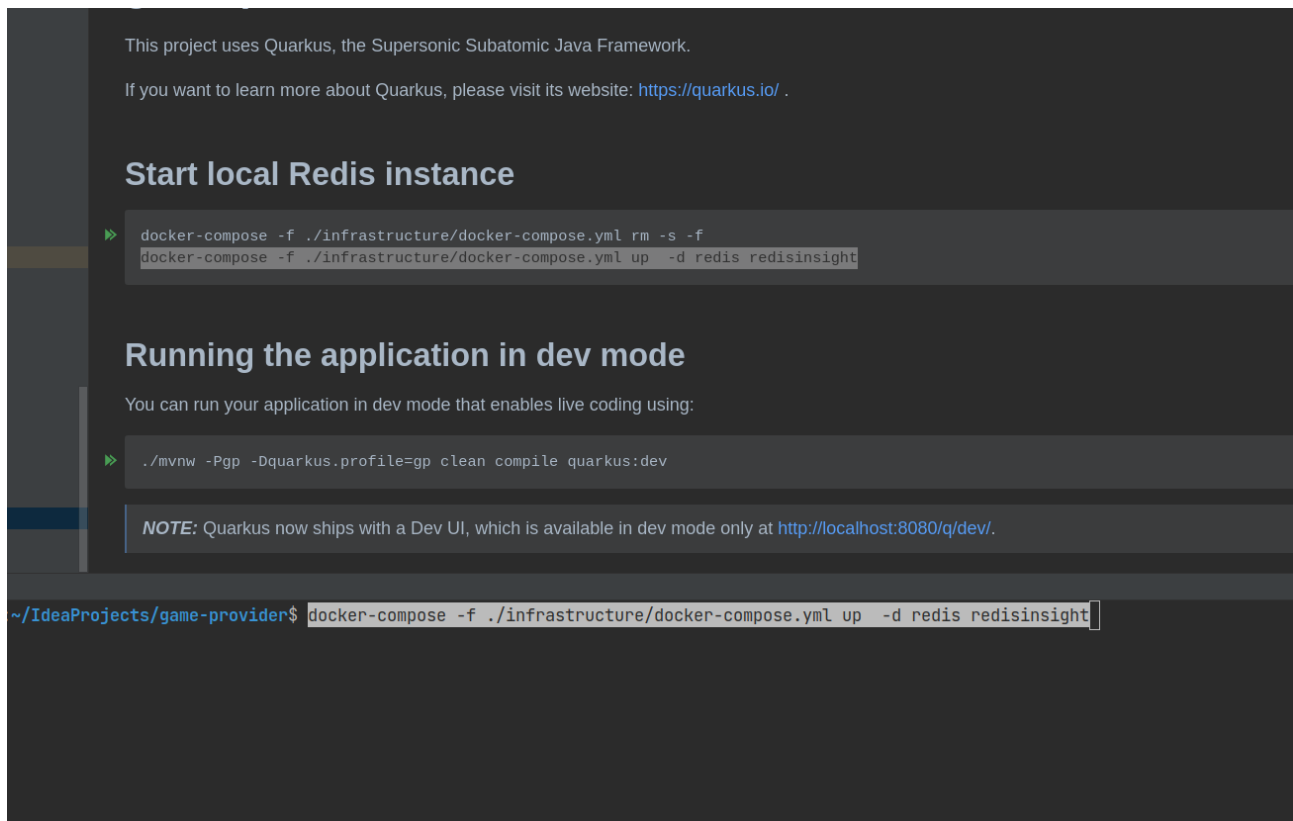
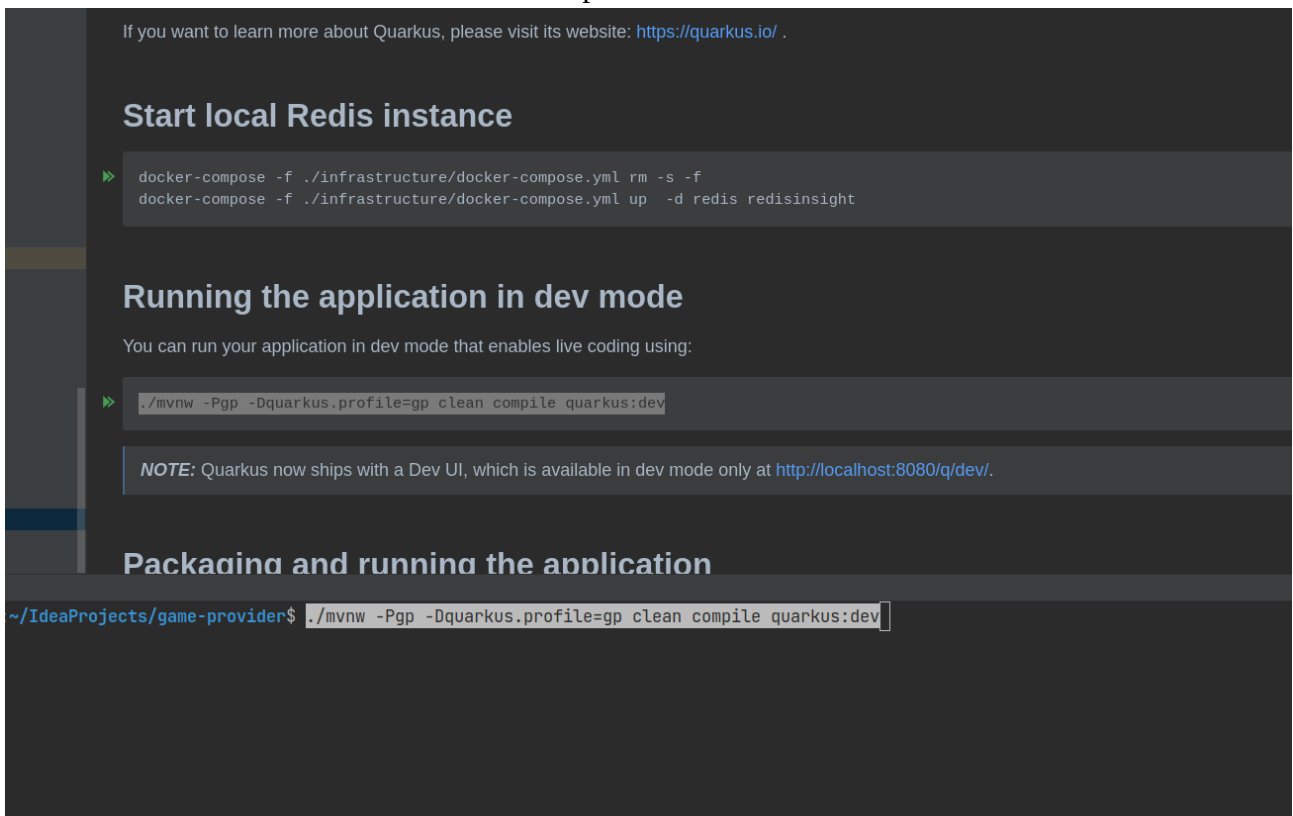


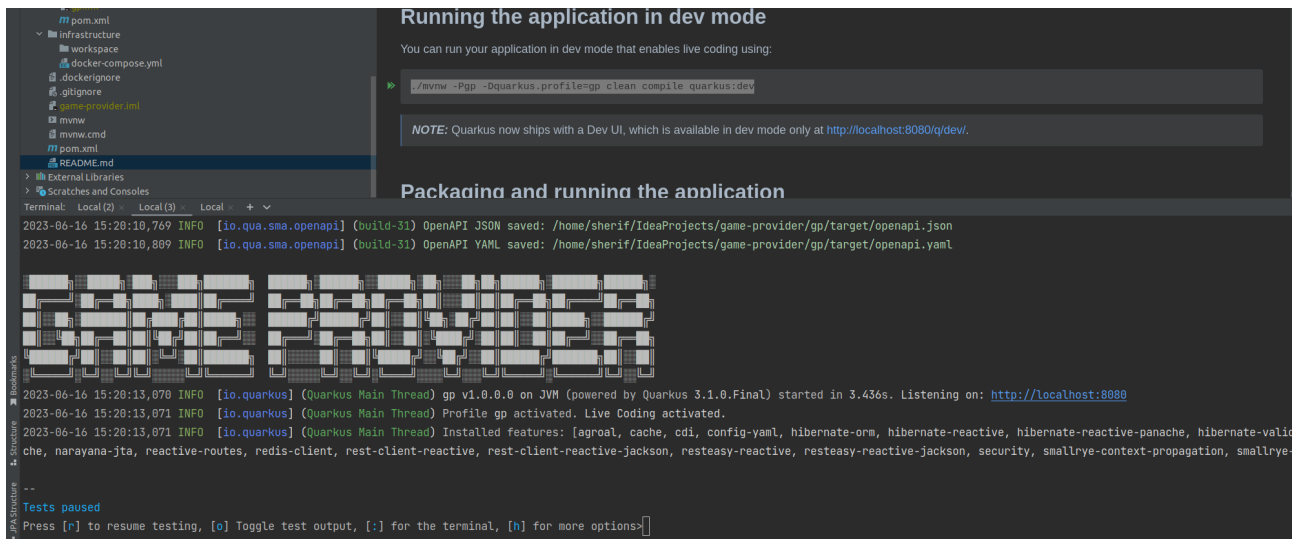
1. In project's root folder open Readme.md file and run the following command in terminal.



2. Copy past the following command to run the application in the dev mode. Make sure you have installed Java 17. Otherwise use this link. <https://sdkman.io/>



3. After successful run you should see something like below.

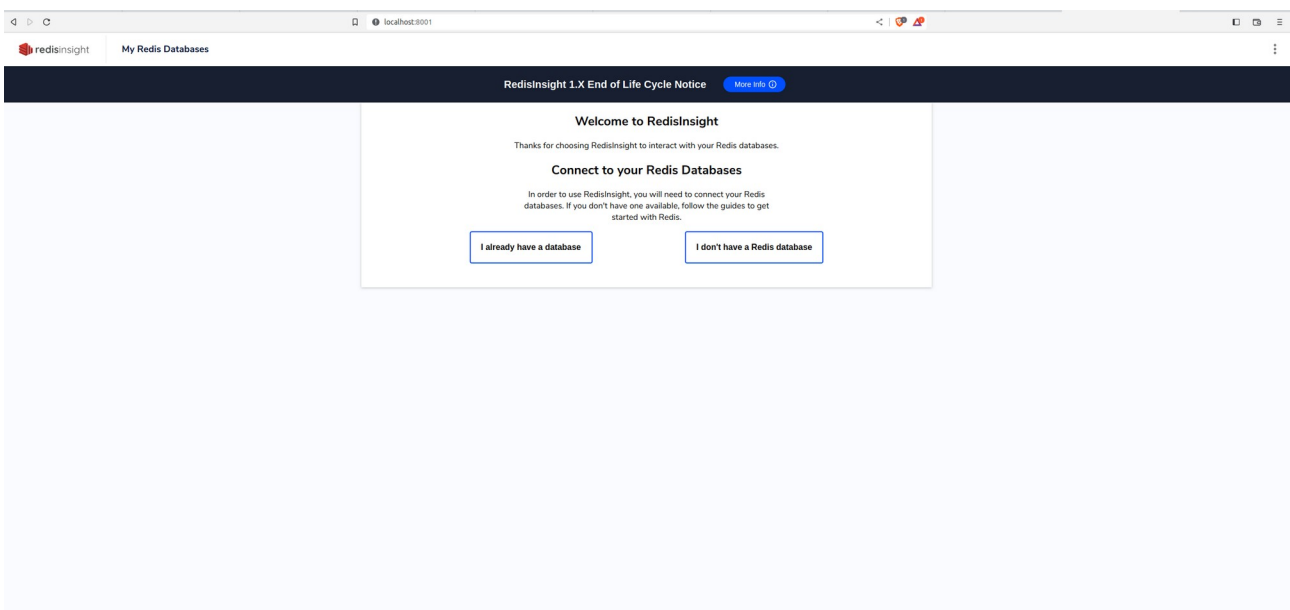


The screenshot shows an IDE with a project named 'game-provider'. The terminal output includes the following logs:

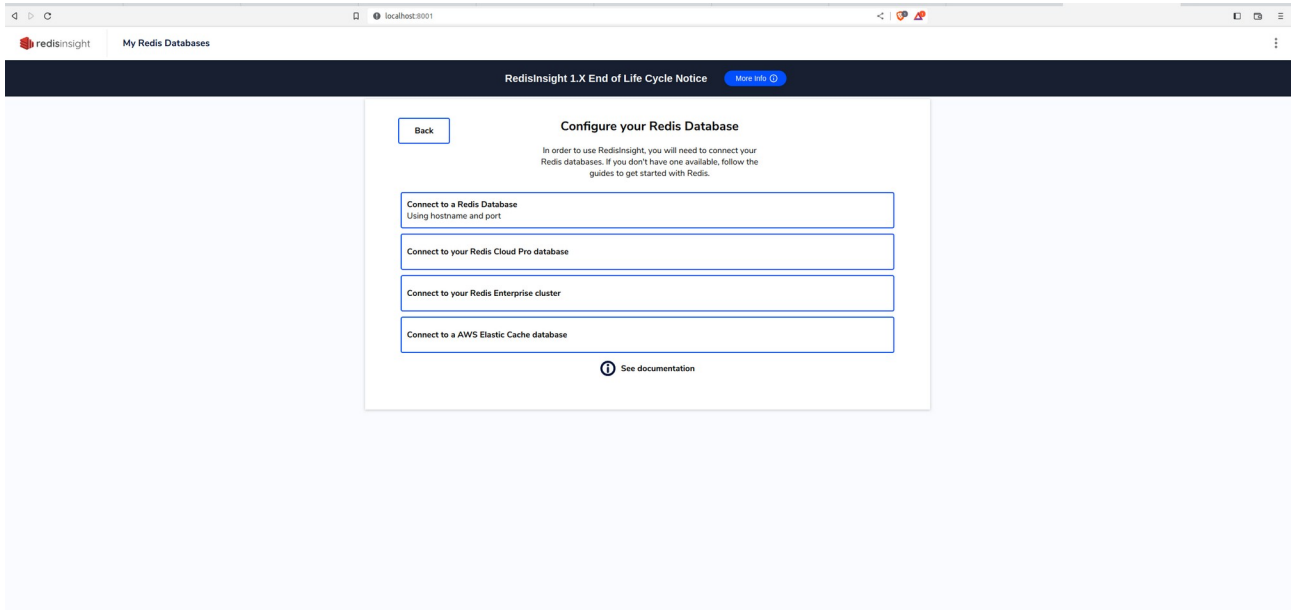
```
2023-06-16 15:28:10,769 INFO [io.qua.sma.openapi] (build-31) OpenAPI JSON saved: /home/sherif/IdeaProjects/game-provider/gp/target/openapi.json
2023-06-16 15:28:10,889 INFO [io.qua.sma.openapi] (build-31) OpenAPI YAML saved: /home/sherif/IdeaProjects/game-provider/gp/target/openapi.yaml
2023-06-16 15:28:13,070 INFO [io.quarkus] (Quarkus Main Thread) gp v1.0.0.0 on JVM (powered by Quarkus 3.1.0.Final) started in 3.436s. Listening on: http://localhost:8080
2023-06-16 15:28:13,071 INFO [io.quarkus] (Quarkus Main Thread) Profile gp activated. Live Coding activated.
2023-06-16 15:28:13,071 INFO [io.quarkus] (Quarkus Main Thread) Installed features: [agroal, cache, cdi, config-yaml, hibernate-orm, hibernate-reactive, hibernate-reactive-panache, hibernate-validation, narayana-jta, reactive-routes, redis-client, rest-client-reactive, rest-client-reactive-jackson, resteasy-reactive, resteasy-reactive-jackson, security, smallrye-context-propagation, smallrye-...
```

The UI shows a grid of application components, including a 'Welcome to RedisInsight' message and a 'Connect to your Redis Databases' section.

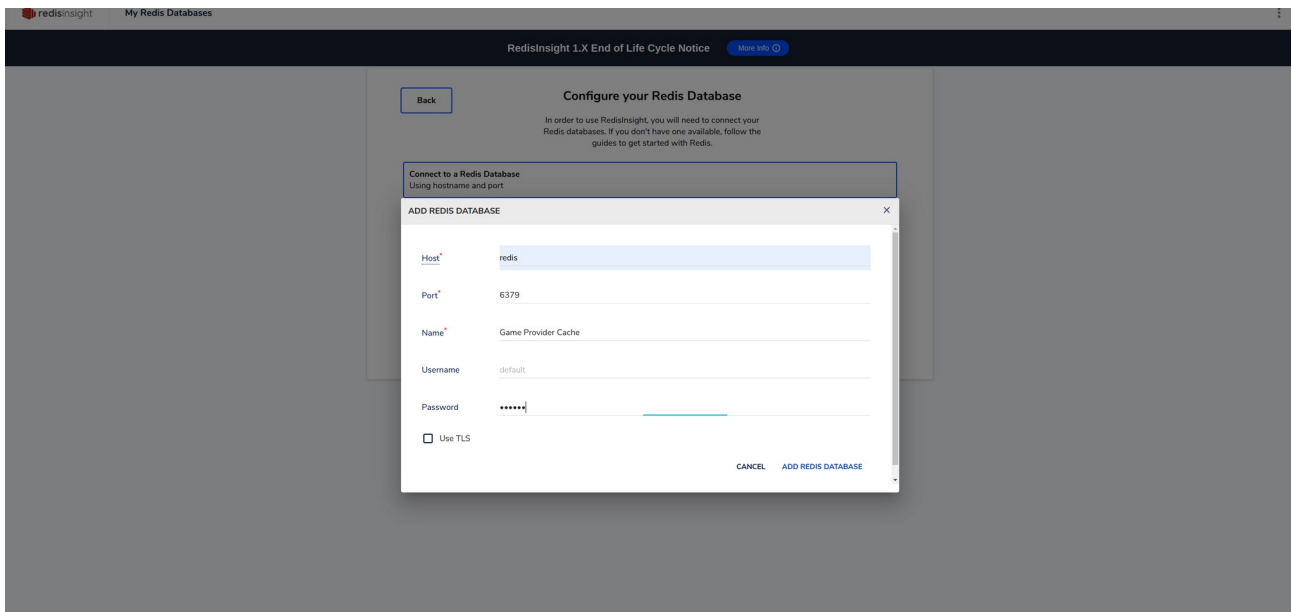
4. Go to localhost:8001 page to open the Redis UI for testing purposes.



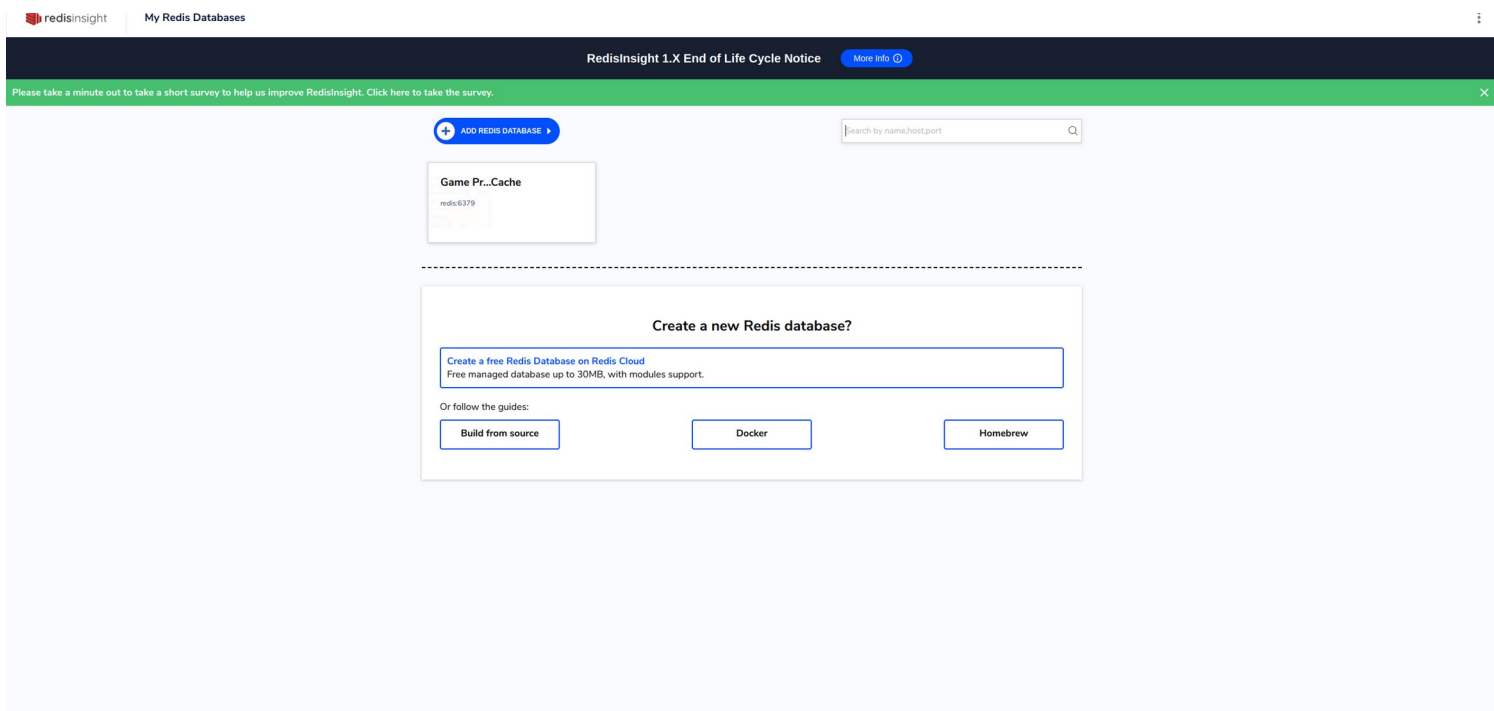
5. Click on “Connect to a Redis Database”



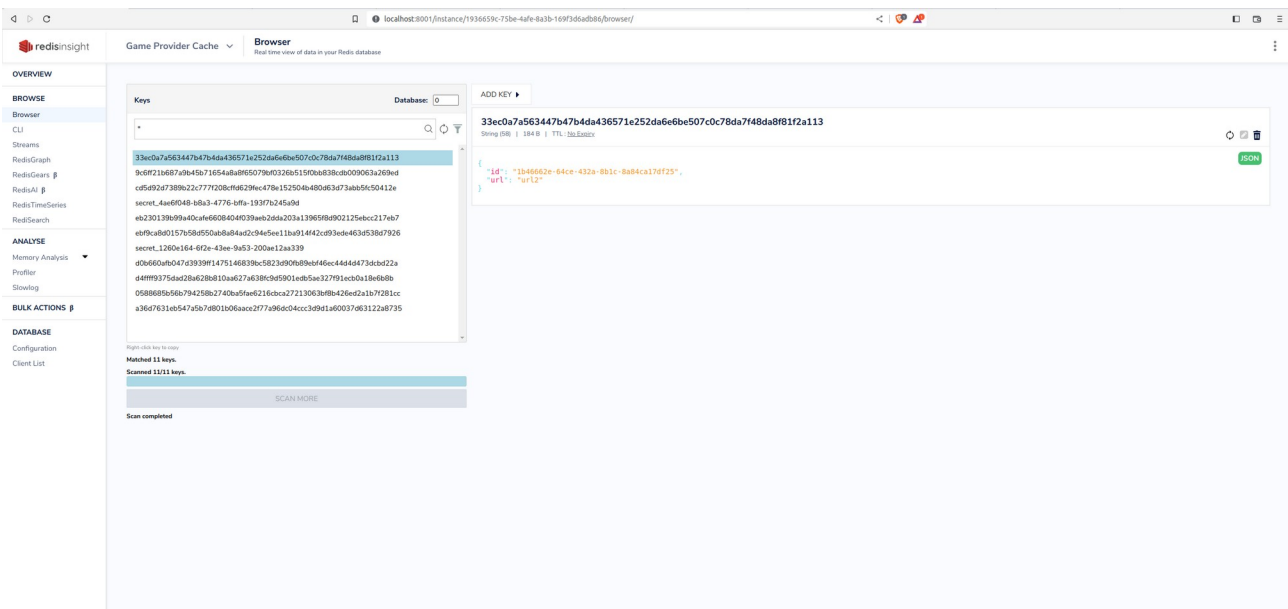
6. Fill the empty fields as described below. Password is 123456



7. Redis cache is ready. Click to the Redis instance with name “Game Provider Cache”



8. From the left pane click to the Browse and you will see predefined platforms, and given game access (platformId + casinoId + gameId as the key). You can see the all predefined ids from the /gp/src/main/java/com.megafair.GameProvider classes. This class created for tentative purposes only.



9. Now you can call Auth And Game modules from the `http://localhost:8080/q/api/`

