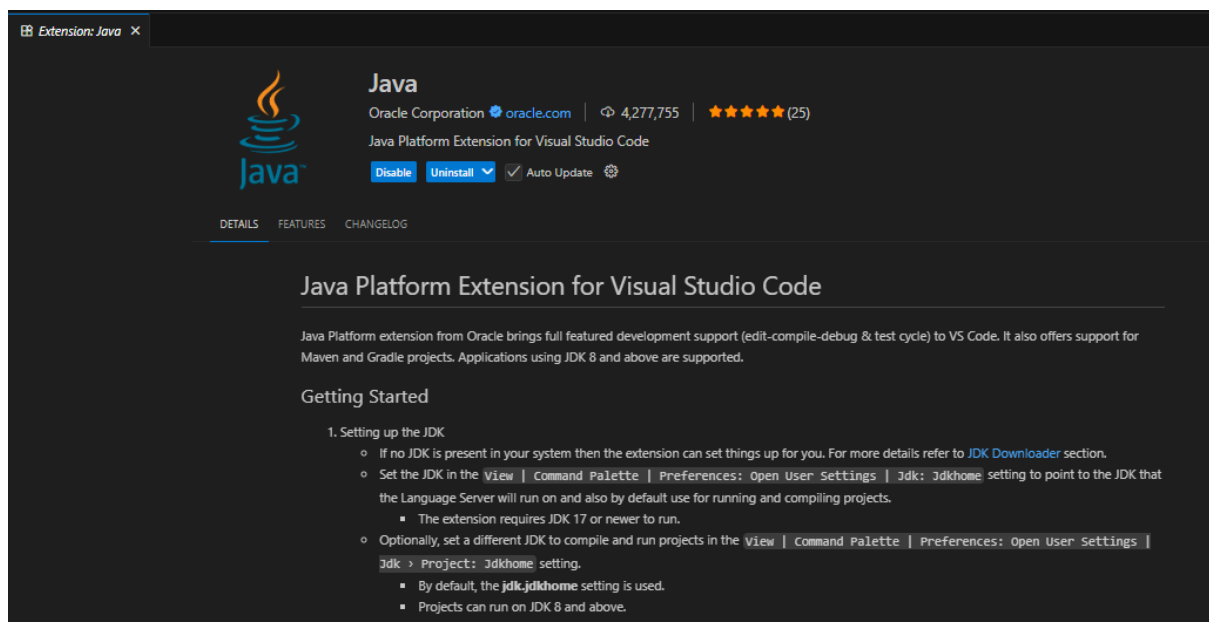


1. java instalado

```
PS C:\Users\thale> java -version
java version "25.0.1" 2025-10-21 LTS
Java(TM) SE Runtime Environment (build 25.0.1+8-LTS-27)
Java HotSpot(TM) 64-Bit Server VM (build 25.0.1+8-LTS-27, mixed mode, sharing)
PS C:\Users\thale> |
```

2. VS code com a extensão java



3. Git

```
PS C:\Users\thale> git --version
git version 2.50.1.windows.1
PS C:\Users\thale> |
```

Project

☐ Gradle - Groovy

☐ Gradle - Kotlin

☒ Maven

Language

☒ Java

☐ Kotlin

☐ Groovy

Spring Boot

☐ 4.0.0 (SNAPSHOT)

☐ 4.0.0 (RC1)

☐ 3.5.8 (SNAPSHOT)

☒ 3.5.7

☐ 3.4.12 (SNAPSHOT)

☐ 3.4.11

Project Metadata

Group

com.deliverylech

Artifact

delivery-api

Name

delivery-api

Description

Demo project for Spring Boot

Package name

com.deliverylech.delivery

Packaging

☒ Jar

☐ War

Java

☐ 25

☒ 21

☐ 17

Dependencies

ADD DEPENDENCIES... CTRL + B

Spring Web

WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

Spring Data JPA

SQL

Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.

H2 Database

SQL

Provides a fast in-memory database that supports JDBC API and R2DBC access, with a small (2mb) footprint. Supports embedded and server modes as well as a browser based console application.

delivery-api > delivery-api > Pesq

Classificar Visualizar Extrair tudo

Nome	Tipo	Tamanho Compact...	Protegido ...	Tamanho	Razão	Data de modificação
.mvn	Pasta de arquivos					10/27/2025 9:46 PM
src	Pasta de arquivos					10/27/2025 9:46 PM
.gitattributes	txtfile	1 KB	Não	1 KB	22%	10/27/2025 9:46 PM
.gitignore	txtfile	1 KB	Não	1 KB	42%	10/27/2025 9:46 PM
HELP.md	Arquivo MD	1 KB	Não	2 KB	56%	10/27/2025 9:46 PM
mvnw	Arquivo	4 KB	Não	12 KB	66%	10/27/2025 9:46 PM
mvnw	Script de Comandos do ...	3 KB	Não	9 KB	65%	10/27/2025 9:46 PM
pom	Microsoft Edge HTML Do...	1 KB	Não	2 KB	66%	10/27/2025 9:46 PM

url github

<https://github.com/zthalesp/delivery-api--Thales-Gabriel-/tree/main>

commit inicial

main


1 Branch

0 Tags

Go to file

Add file


Code

 **zthalesp**

Add initial README with first commit message


d30f086 · 5 minutes ago

2 Commits

 .gitignore


Initial commit

7 minutes ago

 LICENSE

Initial commit

7 minutes ago

 README.md

Add initial README with first commit message

5 minutes ago

README

MIT license

primeiro commit

readme

1 file changed

+1 -0 lines changed

Search within code

README.md

+1 -0

@@ -0,0 +1 @@

1 + primeiro commit

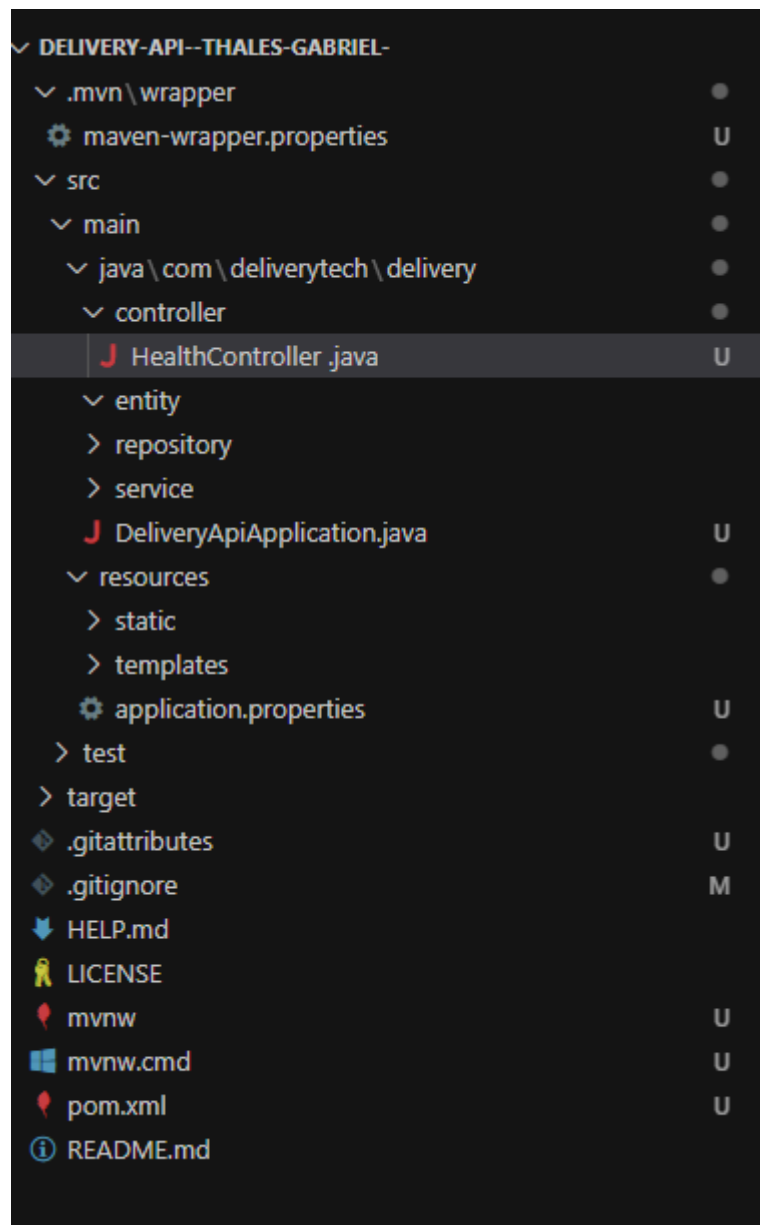
Comments 0

Lock conversation

configuração application.properties

```
src > main > resources > application.properties
1  spring.application.name=delivery-api
2  # Configuração do servidor
3
4  server.port=8080
5
6
7
8  # Configuração do H2 Database
9
10 spring.datasource.url=jdbc:h2:mem:deliverydb
11
12 spring.datasource.driverClassName=org.h2.Driver
13
14 spring.datasource.username=sa
15
16 spring.datasource.password=
17
18
19
20 # Console H2
21
22 spring.h2.console.enabled=true
23
24 spring.h2.console.path=/h2-console
25
26
27
28 # JPA/Hibernate
29
30 spring.jpa.database-platform=org.hibernate.dialect.H2Dialect
31
32 spring.jpa.hibernate.ddl-auto=create-drop
33
34 spring.jpa.show-sql=true
35
36 spring.jpa.properties.hibernate.format_sql=true
37
38
39
40 # Configurações de desenvolvimento
41
42 spring.devtools.restart.enabled=true
43
44
45
46 # Configurações específicas para JDK 21
47
48 spring.jpa.open-in-view=false
49
50 logging.level.org.springframework.web=DEBUG
```

estrutura de pastas do projeto



Código do HealthController implementado

```
src > main > java > com > deliverytech > delivery > controller > J HealthController.java > java > HealthController > info()
1 package com.deliverytech.delivery.controller;
2
3 import java.time.LocalDateTime;
4
5 import org.springframework.web.bind.annotation.RestController;
6
7 import java.util.Map;
8
9 import org.springframework.web.bind.annotation.GetMapping;
10
11
12 @RestController
13 public class HealthController {
14
15
16
17     @GetMapping("/health")
18     public Map<String, String> health() {
19
20         return Map.of(
21
22             "status", "UP",
23
24             "timestamp", LocalDateTime.now().toString(),
25
26             "service", "Delivery API",
27
28             "javaVersion", System.getProperty("java.version")
29
30         );
31     }
32
33
34
35
36
37     @GetMapping("/info")
38     public AppInfo info() {
39
40         return new AppInfo(
41
42             application:"Delivery Tech API",
43
44             version:"1.0.0",
45
46             developer:"Thales",
47
48             javaVersion:"JDK 21",
49
50             framework:"Spring Boot 3.2.x"
51
52         );
53     }
54
55
56
57
58
59 // Record para demonstrar recurso do Java 14+ (disponível no JDK 21)
60 public record AppInfo(
61
62     String application,
63
64     String version,
65
66     String developer,
67
68     String javaVersion,
69
70     String framework
71
72 ) {}
73
74
75
76
77 }
```

endpoint funcionando no navegador

Estilos de formatação ■

```
{"status":"UP","javaVersion":"21.0.8","timestamp":"2025-10-27T19:47:51.648539500","service":"Delivery API"}
```

4