Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie

AGH University of Science
and Technology

Studio projektowe 2
Get good with
Secure App Practices

Jacek Oleś, Łukasz Stawowy[1]

[1] AGH University of Science and Technology

Faculty of Electrical Engineering,
Automatics, Computer Science
and Biomedical Engineering

ul. Mickiewicza 30
30-059 Kraków
Poland

16.11.2019

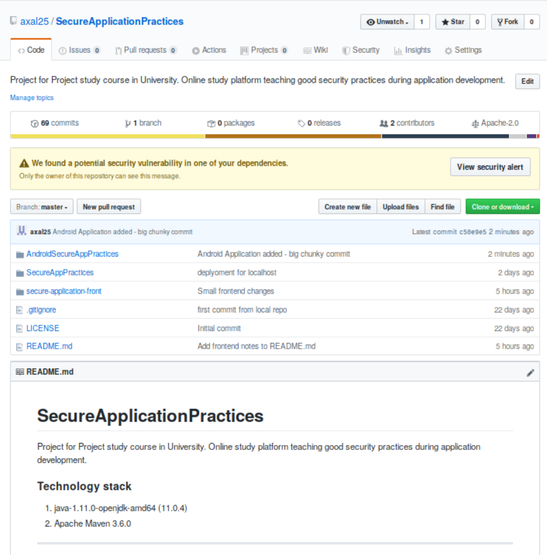
Part I

Ogolne

Github

Dokumentacja i Repozytorium

https://github.com/axal25/SecureApplicationPractices



Project for Project study course in University. Online study platform teaching good security practices during application development.

We found a potential security vulnerability in one of your dependencies.

Only the owner of this repository can see this message.

axal25 Android Application added - big chunky commit — Latest commit c58e9e5 2 minutes ago

AndroidSecureAppPractices	Android Application added - big chunky commit	2 minutes ago
SecureAppPractices	deplyoment for localhost	2 days ago
secure-application-front	Small frontend changes	5 hours ago
.gitignore	first commit from local repo	22 days ago
LICENSE	Initial commit	22 days ago
README.md	Add frontend notes to README.md	5 hours ago

SecureApplicationPractices

Project for Project study course in University. Online study platform teaching good security practices during application development.

Technology stack

1. java-1.11.0-openjdk-amd64 (11.0.4)
2. Apache Maven 3.6.0

Part II

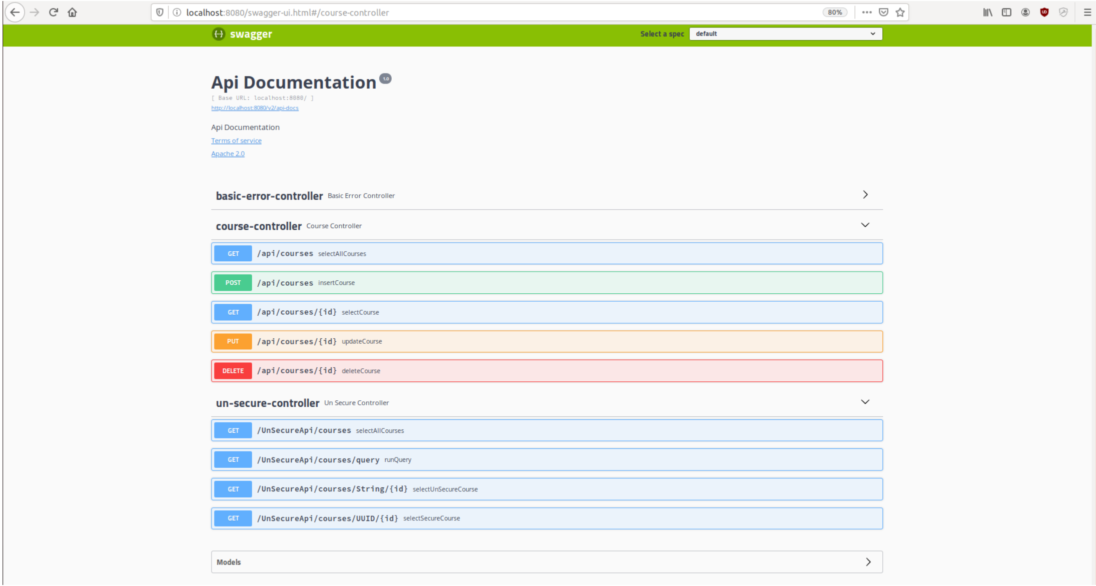
Backend

Backend - Spis treści

2. Swagger - Dokumentacja endpointów, ułatwienie życia frontendowi
3. Bezpieczne API - Przykład endpointów odpornych na SQLi
4. NIEbezpieczne API - Przykład endpointów odpornych na SQLi
5. NIEbezpieczne API - Przykład endpointów odpornych na SQLi
6. Baza danych - PostgreSQL - Skrypt tworzacy baze, schemy, tablice i 3 użytkownikow
7. Docker i Postgres - kontenerowa baza danych
8. Google Cloud Platfrom Postgres Database
9. Google Cloud Platfrom API
10. Wybor bazy danych
11. Test bazy, skryptu, aplikacji back-end'owej
12. Aplikacja androidowa

Swagger

Dokumentacja endpointów, ułatwienie życia frontendowi



Api Documentation 1.0

[Base URL: localhost:8080/]
http://localhost:8080/v2/api-docs

Api Documentation

Terms of service

Apache 2.0

basic-error-controller Basic Error Controller

course-controller Course Controller

Method	Path	Operation
GET	/api/courses	selectAllCourses
POST	/api/courses	insertCourse
GET	/api/courses/{id}	selectCourse
PUT	/api/courses/{id}	updateCourse
DELETE	/api/courses/{id}	deleteCourse

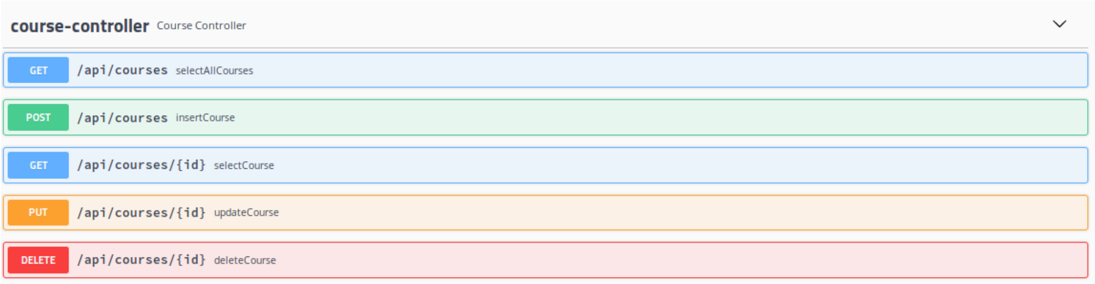
un-secure-controller Un Secure Controller

Method	Path	Operation
GET	/UnSecureApi/courses	selectAllCourses
GET	/UnSecureApi/courses/query	runQuery
GET	/UnSecureApi/courses/String/{id}	selectUnSecureCourse
GET	/UnSecureApi/courses/UUID/{id}	selectSecureCourse

Models

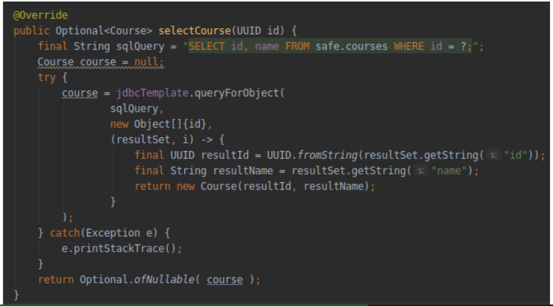
Bezpieczne API

Przykład endpointów odpornych na SQLi

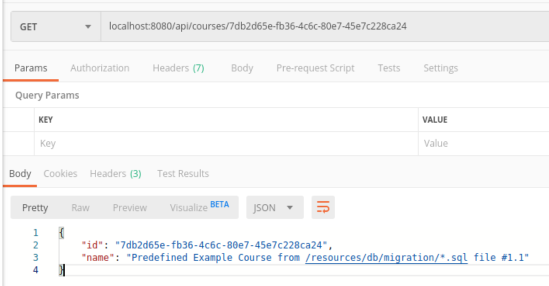


course-controller Course Controller

GET	`/api/courses`	selectAllCourses
POST	`/api/courses`	insertCourse
GET	`/api/courses/{id}`	selectCourse
PUT	`/api/courses/{id}`	updateCourse
DELETE	`/api/courses/{id}`	deleteCourse



```java
@Override
public Optional<Course> selectCourse(UUID id) {
    final String sqlQuery = "SELECT id, name FROM safe.courses WHERE id = ?;";
    Course course = null;
    try {
        course = jdbcTemplate.queryForObject(
                sqlQuery,
                new Object[]{id},
                (resultSet, i) -> {
                    final UUID resultId = UUID.fromString(resultSet.getString( s: "id"));
                    final String resultName = resultSet.getString( s: "name");
                    return new Course(resultId, resultName);
                }
        );
    } catch(Exception e) {
        e.printStackTrace();
    }
    return Optional.ofNullable( course );
}
```



GET localhost:8080/api/courses/7db2d65e-fb36-4c6c-80e7-45e7c228ca24

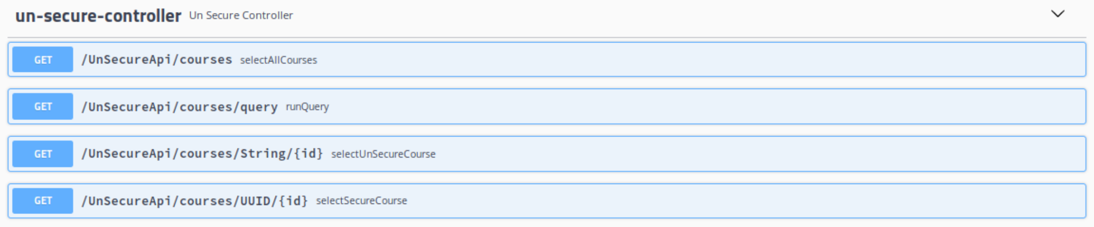
Query Params

KEY	VALUE
Key	Value

```json
{
    "id": "7db2d65e-fb36-4c6c-80e7-45e7c228ca24",
    "name": "Predefined Example Course from /resources/db/migration/*.sql file #1.1"
}
```

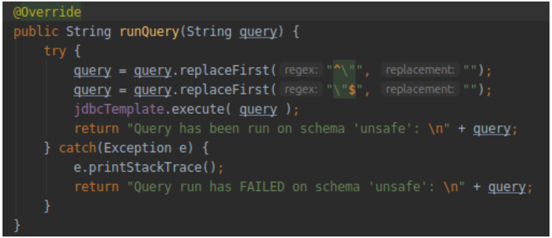
NIEbezpieczne API

Przykład endpointów narażonych na SQLi

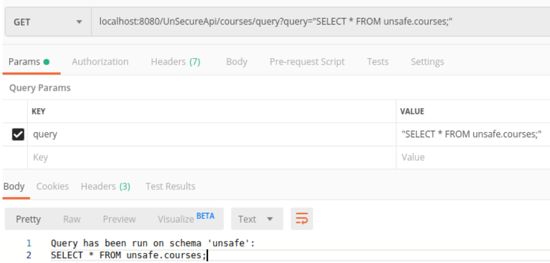


un-secure-controller Un Secure Controller

- GET `/UnSecureApi/courses` selectAllCourses
- GET `/UnSecureApi/courses/query` runQuery
- GET `/UnSecureApi/courses/String/{id}` selectUnSecureCourse
- GET `/UnSecureApi/courses/UUID/{id}` selectSecureCourse



```java
@Override
public String runQuery(String query) {
    try {
        query = query.replaceFirst( regex: "^\"", replacement: "");
        query = query.replaceFirst( regex: "\"$", replacement: "");
        jdbcTemplate.execute( query );
        return "Query has been run on schema 'unsafe': \n" + query;
    } catch(Exception e) {
        e.printStackTrace();
        return "Query run has FAILED on schema 'unsafe': \n" + query;
    }
}
```



GET localhost:8080/UnSecureApi/courses/query?query="SELECT * FROM unsafe.courses;"

Query Params

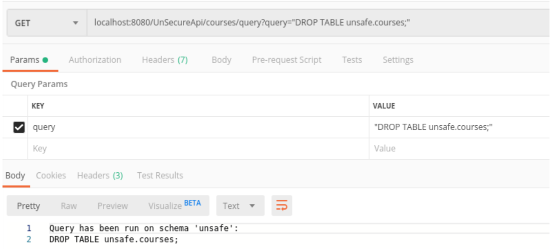
	KEY	VALUE
☑	query	"SELECT * FROM unsafe.courses;"
	Key	Value

Body

```
Query has been run on schema 'unsafe':
SELECT * FROM unsafe.courses;
```

NIEbezpieczne API

Przykład endpointów narażonych na SQLi



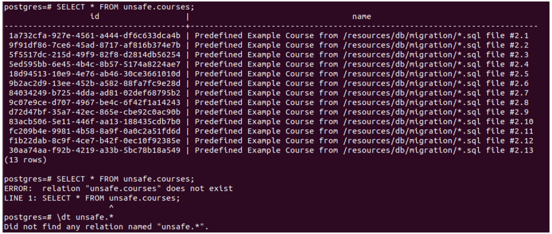
GET localhost:8080/UnSecureApi/courses/query?query="DROP TABLE unsafe.courses;"

Params ● — Query Params

	KEY	VALUE
☑	query	"DROP TABLE unsafe.courses;"
	Key	Value

Body — Text

```
Query has been run on schema 'unsafe':
DROP TABLE unsafe.courses;
```

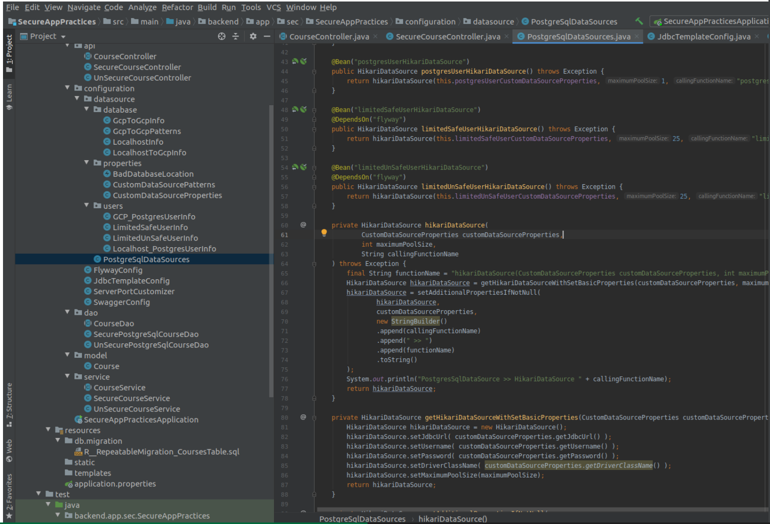


```
postgres=# SELECT * FROM unsafe.courses;
                  id                  |                                   name
--------------------------------------+-------------------------------------------------------------------------
 1a732cfa-927e-4561-a444-df6c633dca4b | Predefined Example Course from /resources/db/migration/*.sql file #2.1
 9f91df86-7ce6-45ad-8717-af816b374e7b | Predefined Example Course from /resources/db/migration/*.sql file #2.2
 5f5517dc-215d-49f9-82f8-d2814db56254 | Predefined Example Course from /resources/db/migration/*.sql file #2.3
 5ed595bb-6e45-4b4c-8b57-5174a8224ae7 | Predefined Example Course from /resources/db/migration/*.sql file #2.4
 18d94513-10e9-4e76-ab46-30ce3661010d | Predefined Example Course from /resources/db/migration/*.sql file #2.5
 9b2ac2d9-13ee-452b-a582-88fa7fc9e28d | Predefined Example Course from /resources/db/migration/*.sql file #2.6
 84034249-b725-4dda-ad81-02def68795b2 | Predefined Example Course from /resources/db/migration/*.sql file #2.7
 9c07e9ce-d707-4967-be4c-6f42f1a14243 | Predefined Example Course from /resources/db/migration/*.sql file #2.8
 d72d47bf-35a7-42ec-865e-cbe92c0ac90b | Predefined Example Course from /resources/db/migration/*.sql file #2.9
 83acb506-5e11-446f-aa13-188435cdb7b0 | Predefined Example Course from /resources/db/migration/*.sql file #2.10
 fc209b4e-9981-4b58-8a9f-0a0c2a51fd6d | Predefined Example Course from /resources/db/migration/*.sql file #2.11
 f1b22dab-8c9f-4ce7-b42f-0ec10f92385e | Predefined Example Course from /resources/db/migration/*.sql file #2.12
 30aa74aa-f92b-4219-a33b-5bc78b18a549 | Predefined Example Course from /resources/db/migration/*.sql file #2.13
(13 rows)

postgres=# SELECT * FROM unsafe.courses;
ERROR:  relation "unsafe.courses" does not exist
LINE 1: SELECT * FROM unsafe.courses;
                      ^
postgres=# \dt unsafe.*
Did not find any relation named "unsafe.*".
```

Baza danych - PostgreSQL

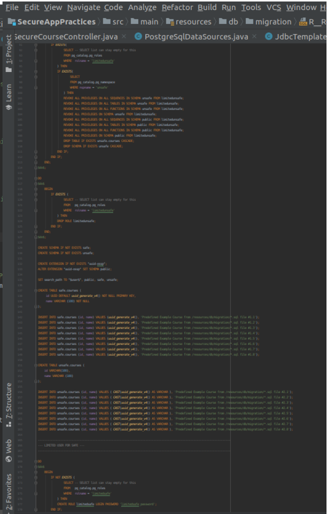
Skrypt tworzacy baze, schemy, tablice i 3 użytkownikow

3 datasource'y dla 3 użytkowników, różne uprawnienia



```java
    @Bean("postgresUserHikariDataSource")
    public HikariDataSource postgresUserHikariDataSource() throws Exception {
        return hikariDataSource(this.postgresUserCustomDataSourceProperties,  maximumPoolSize: 1,  callingFunctionName: "postgres
    }

    @Bean("limitedSafeUserHikariDataSource")
    @DependsOn("flyway")
    public HikariDataSource limitedSafeUserHikariDataSource() throws Exception {
        return hikariDataSource(this.limitedSafeUserCustomDataSourceProperties,  maximumPoolSize: 25,  callingFunctionName: "limi
    }

    @Bean("limitedUnSafeUserHikariDataSource")
    @DependsOn("flyway")
    public HikariDataSource limitedUnSafeUserHikariDataSource() throws Exception {
        return hikariDataSource(this.limitedUnSafeUserCustomDataSourceProperties,  maximumPoolSize: 25,  callingFunctionName: "li
    }

    private HikariDataSource hikariDataSource(
            CustomDataSourceProperties customDataSourceProperties,
            int maximumPoolSize,
            String callingFunctionName
    ) throws Exception {
        final String functionName = "hikariDataSource(CustomDataSourceProperties customDataSourceProperties, int maximumP
        HikariDataSource hikariDataSource = getHikariDataSourceWithSetBasicProperties(customDataSourceProperties, maximum
        hikariDataSource = setAdditionalPropertiesIfNotNull(
                hikariDataSource,
                customDataSourceProperties,
                new StringBuilder()
                .append(callingFunctionName)
                .append(" >> ")
                .append(functionName)
                .toString()
        );
        System.out.println("PostgresSqlDataSource >> HikariDataSource " + callingFunctionName);
        return hikariDataSource;
    }

    private HikariDataSource getHikariDataSourceWithSetBasicProperties(CustomDataSourceProperties customDataSourcePropert
        HikariDataSource hikariDataSource = new HikariDataSource();
        hikariDataSource.setJdbcUrl( customDataSourceProperties.getJdbcUrl() );
        hikariDataSource.setUsername( customDataSourceProperties.getUsername() );
        hikariDataSource.setPassword( customDataSourceProperties.getPassword() );
        hikariDataSource.setDriverClassName( customDataSourceProperties.getDriverClassName() );
        hikariDataSource.setMaximumPoolSize(maximumPoolSize);
        return hikariDataSource;
    }
```

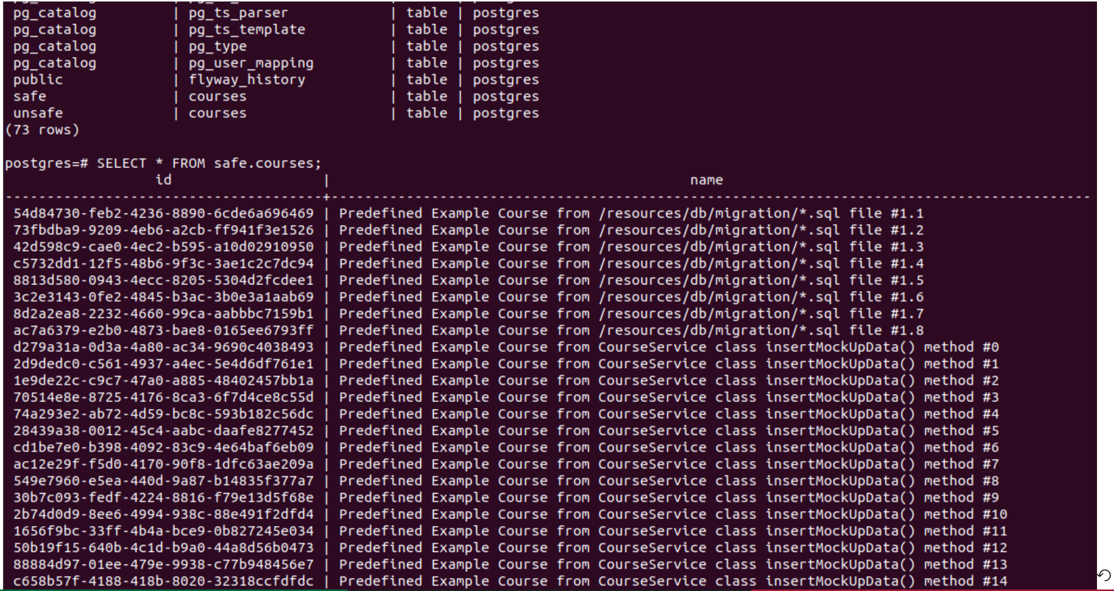
Skrypt tworzacy i niszczacy baze przed każdym uruchomieniem aplikacji


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window H
SecureAppPractices › src › main › resources › db › migration › R__R
SecureCourseController.java
PostgreSqlDataSources.java
JdbcTemplate
1: Project
Learn
Z: Structure
Web
2: Favorites

```sql
        IF EXISTS(
            SELECT -- SELECT list can stay empty for this
            FROM pg_catalog.pg_roles
            WHERE  rolname = 'limitedunsafe'
        ) THEN
            IF EXISTS(
                SELECT
                FROM pg_catalog.pg_namespace
                WHERE nspname = 'unsafe'
            ) THEN
            REVOKE ALL PRIVILEGES ON ALL SEQUENCES IN SCHEMA unsafe FROM limitedunsafe;
            REVOKE ALL PRIVILEGES ON ALL TABLES IN SCHEMA unsafe FROM limitedunsafe;
            REVOKE ALL PRIVILEGES ON ALL FUNCTIONS IN SCHEMA unsafe FROM limitedunsafe;
            REVOKE ALL PRIVILEGES ON SCHEMA unsafe FROM limitedunsafe;
            REVOKE ALL PRIVILEGES ON ALL SEQUENCES IN SCHEMA public FROM limitedunsafe;
            REVOKE ALL PRIVILEGES ON ALL TABLES IN SCHEMA public FROM limitedunsafe;
            REVOKE ALL PRIVILEGES ON ALL FUNCTIONS IN SCHEMA public FROM limitedunsafe;
            REVOKE ALL PRIVILEGES ON SCHEMA public FROM limitedunsafe;
            DROP TABLE IF EXISTS unsafe.courses CASCADE;
            DROP SCHEMA IF EXISTS unsafe CASCADE;
            END IF;
        END IF;
    END;
$do$;

DO
$do$
    BEGIN
        IF EXISTS (
            SELECT -- SELECT list can stay empty for this
            FROM   pg_catalog.pg_roles
            WHERE  rolname = 'limitedunsafe'
        ) THEN
            DROP ROLE limitedunsafe;
        END IF;
    END;
$do$;

CREATE SCHEMA IF NOT EXISTS safe;
CREATE SCHEMA IF NOT EXISTS unsafe;

CREATE EXTENSION IF NOT EXISTS "uuid-ossp";
ALTER EXTENSION "uuid-ossp" SET SCHEMA public;

SET search_path TO "$user$", public, safe, unsafe;

CREATE TABLE safe.courses (
    id UUID DEFAULT uuid_generate_v4() NOT NULL PRIMARY KEY,
    name VARCHAR (100) NOT NULL
);

INSERT INTO safe.courses (id, name) VALUES (uuid_generate_v4(), 'Predefined Example Course from /resources/db/migration/*.sql file #1.1');
INSERT INTO safe.courses (id, name) VALUES (uuid_generate_v4(), 'Predefined Example Course from /resources/db/migration/*.sql file #1.2');
INSERT INTO safe.courses (id, name) VALUES (uuid_generate_v4(), 'Predefined Example Course from /resources/db/migration/*.sql file #1.3');
INSERT INTO safe.courses (id, name) VALUES (uuid_generate_v4(), 'Predefined Example Course from /resources/db/migration/*.sql file #1.4');
INSERT INTO safe.courses (id, name) VALUES (uuid_generate_v4(), 'Predefined Example Course from /resources/db/migration/*.sql file #1.5');
INSERT INTO safe.courses (id, name) VALUES (uuid_generate_v4(), 'Predefined Example Course from /resources/db/migration/*.sql file #1.6');
INSERT INTO safe.courses (id, name) VALUES (uuid_generate_v4(), 'Predefined Example Course from /resources/db/migration/*.sql file #1.7');
INSERT INTO safe.courses (id, name) VALUES (uuid_generate_v4(), 'Predefined Example Course from /resources/db/migration/*.sql file #1.8');

CREATE TABLE unsafe.courses (
    id VARCHAR(100),
    name VARCHAR (100)
);

INSERT INTO unsafe.courses (id, name) VALUES ( CAST(uuid_generate_v4() AS VARCHAR ), 'Predefined Example Course from /resources/db/migration/*.sql file #2.1');
INSERT INTO unsafe.courses (id, name) VALUES ( CAST(uuid_generate_v4() AS VARCHAR ), 'Predefined Example Course from /resources/db/migration/*.sql file #2.2');
INSERT INTO unsafe.courses (id, name) VALUES ( CAST(uuid_generate_v4() AS VARCHAR ), 'Predefined Example Course from /resources/db/migration/*.sql file #2.3');
INSERT INTO unsafe.courses (id, name) VALUES ( CAST(uuid_generate_v4() AS VARCHAR ), 'Predefined Example Course from /resources/db/migration/*.sql file #2.4');
INSERT INTO unsafe.courses (id, name) VALUES ( CAST(uuid_generate_v4() AS VARCHAR ), 'Predefined Example Course from /resources/db/migration/*.sql file #2.5');
INSERT INTO unsafe.courses (id, name) VALUES ( CAST(uuid_generate_v4() AS VARCHAR ), 'Predefined Example Course from /resources/db/migration/*.sql file #2.6');
INSERT INTO unsafe.courses (id, name) VALUES ( CAST(uuid_generate_v4() AS VARCHAR ), 'Predefined Example Course from /resources/db/migration/*.sql file #2.7');
INSERT INTO unsafe.courses (id, name) VALUES ( CAST(uuid_generate_v4() AS VARCHAR ), 'Predefined Example Course from /resources/db/migration/*.sql file #2.8');

-----------------------------------------------------------
--- LIMITED USER FOR SAFE ---
-----------------------------------------------------------

DO
$do$
    BEGIN
        IF NOT EXISTS (
            SELECT -- SELECT list can stay empty for this
            FROM   pg_catalog.pg_roles
            WHERE  rolname = 'limitedsafe'
        ) THEN
            CREATE ROLE limitedsafe LOGIN PASSWORD 'limitedsafe password';
        END IF;
```

Docker i Postgres

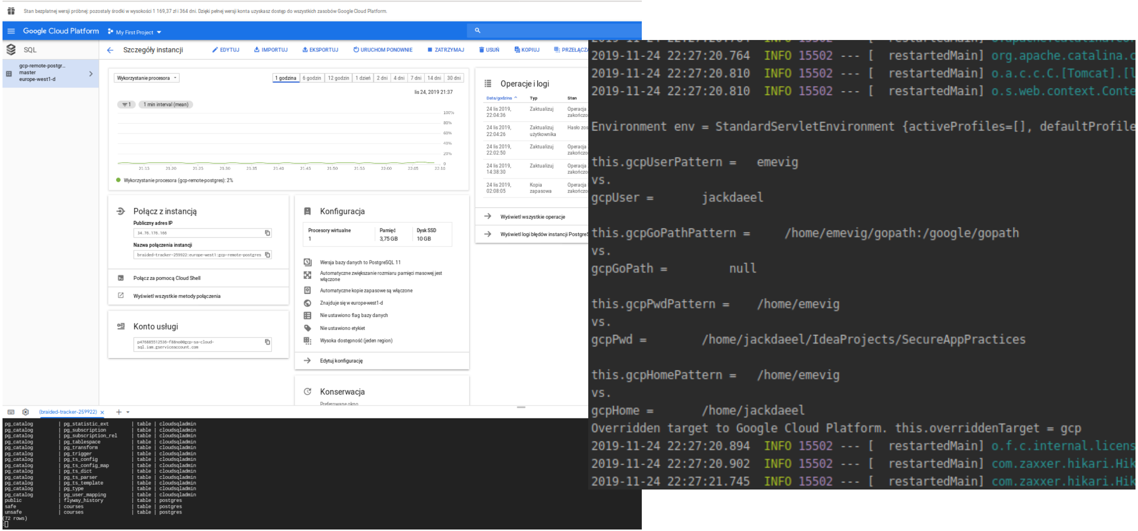
Kontenerowa baza danych



```
 pg_catalog          | pg_ts_parser              | table | postgres
 pg_catalog          | pg_ts_template            | table | postgres
 pg_catalog          | pg_type                   | table | postgres
 pg_catalog          | pg_user_mapping           | table | postgres
 public              | flyway_history            | table | postgres
 safe                | courses                   | table | postgres
 unsafe              | courses                   | table | postgres
(73 rows)

postgres=# SELECT * FROM safe.courses;
                  id                  |                                           name
--------------------------------------+-------------------------------------------------------------------------------------------
 54d84730-feb2-4236-8890-6cde6a696469 | Predefined Example Course from /resources/db/migration/*.sql file #1.1
 73fbdba9-9209-4eb6-a2cb-ff941f3e1526 | Predefined Example Course from /resources/db/migration/*.sql file #1.2
 42d598c9-cae0-4ec2-b595-a10d02910950 | Predefined Example Course from /resources/db/migration/*.sql file #1.3
 c5732dd1-12f5-48b6-9f3c-3ae1c2c7dc94 | Predefined Example Course from /resources/db/migration/*.sql file #1.4
 8813d580-0943-4ecc-8205-5304d2fcdee1 | Predefined Example Course from /resources/db/migration/*.sql file #1.5
 3c2e3143-0fe2-4845-b3ac-3b0e3a1aab69 | Predefined Example Course from /resources/db/migration/*.sql file #1.6
 8d2a2ea8-2232-4660-99ca-aabbbc7159b1 | Predefined Example Course from /resources/db/migration/*.sql file #1.7
 ac7a6379-e2b0-4873-bae8-0165ee6793ff | Predefined Example Course from /resources/db/migration/*.sql file #1.8
 d279a31a-0d3a-4a80-ac34-9690c4038493 | Predefined Example Course from CourseService class insertMockUpData() method #0
 2d9dedc0-c561-4937-a4ec-5e4d6df761e1 | Predefined Example Course from CourseService class insertMockUpData() method #1
 1e9de22c-c9c7-47a0-a885-48402457bb1a | Predefined Example Course from CourseService class insertMockUpData() method #2
 70514e8e-8725-4176-8ca3-6f7d4ce8c55d | Predefined Example Course from CourseService class insertMockUpData() method #3
 74a293e2-ab72-4d59-bc8c-593b182c56dc | Predefined Example Course from CourseService class insertMockUpData() method #4
 28439a38-0012-45c4-aabc-daafe8277452 | Predefined Example Course from CourseService class insertMockUpData() method #5
 cd1be7e0-b398-4092-83c9-4e64baf6eb09 | Predefined Example Course from CourseService class insertMockUpData() method #6
 ac12e29f-f5d0-4170-90f8-1dfc63ae209a | Predefined Example Course from CourseService class insertMockUpData() method #7
 549e7960-e5ea-440d-9a87-b14835f377a7 | Predefined Example Course from CourseService class insertMockUpData() method #8
 30b7c093-fedf-4224-8816-f79e13d5f68e | Predefined Example Course from CourseService class insertMockUpData() method #9
 2b74d0d9-8ee6-4994-938c-88e491f2dfd4 | Predefined Example Course from CourseService class insertMockUpData() method #10
 1656f9bc-33ff-4b4a-bce9-0b827245e034 | Predefined Example Course from CourseService class insertMockUpData() method #11
 50b19f15-640b-4c1d-b9a0-44a8d56b0473 | Predefined Example Course from CourseService class insertMockUpData() method #12
 88884d97-01ee-479e-9938-c77b948456e7 | Predefined Example Course from CourseService class insertMockUpData() method #13
 c658b57f-4188-418b-8020-32318ccfdfdc | Predefined Example Course from CourseService class insertMockUpData() method #14
```

GCP Postgres Database

Zewnetrzna baza danych



Stan bezpłatnej wersji próbnej: pozostały środki w wysokości 1 169,37 zł i 364 dni. Dzięki pełnej wersji konta uzyskasz dostęp do wszystkich zasobów Google Cloud Platform.

Operacje i logi

Data/godzina	Typ	Stan
24 lis 2019, 22:04:36	Zaktualizuj	Operacja zakończo
24 lis 2019, 22:04:26	Zaktualizuj użytkownika	Hasło zos
24 lis 2019, 22:02:50	Zaktualizuj	Operacja zakończo
24 lis 2019, 14:38:30	Zaktualizuj	Operacja zakończo
24 lis 2019, 02:08:05	Kopia zapasowa	Operacja zakończo

Połącz z instancją

Publiczny adres IP: 34.76.176.166

Nazwa połączenia instancji: braided-tracker-259922:europe-west1:gcp-remote-postgres

Konto usługi

p476885512536-f88no0@gcp-sa-cloud-sql.iam.gserviceaccount.com

Konfiguracja

Procesory wirtualne	Pamięć	Dysk SSD
1	3,75 GB	10 GB

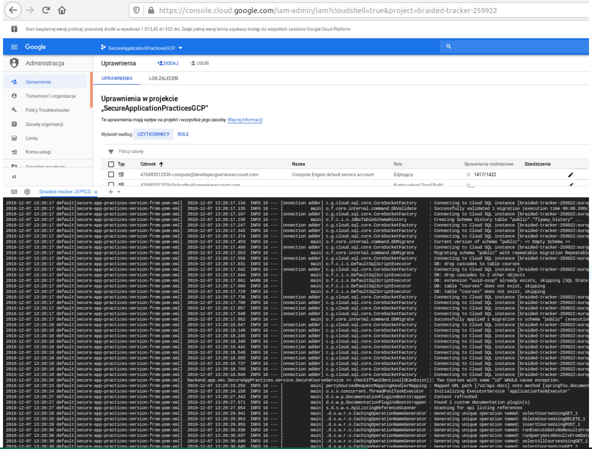
- Wersja bazy danych to PostgreSQL 11
- Automatyczne zwiększanie rozmiaru pamięci masowej jest włączone
- Automatyczne kopie zapasowe są włączone
- Znajduje się w europe-west1-d
- Nie ustawiono flag bazy danych
- Nie ustawiono etykiet
- Wysoka dostępność (jeden region)

```
pg_catalog          | pg_statistic_ext          | table | cloudsqladmin
pg_catalog          | pg_subscription           | table | cloudsqladmin
pg_catalog          | pg_subscription_rel       | table | cloudsqladmin
pg_catalog          | pg_tablespace             | table | cloudsqladmin
pg_catalog          | pg_transform              | table | cloudsqladmin
pg_catalog          | pg_trigger                | table | cloudsqladmin
pg_catalog          | pg_ts_config              | table | cloudsqladmin
pg_catalog          | pg_ts_config_map          | table | cloudsqladmin
pg_catalog          | pg_ts_dict                | table | cloudsqladmin
pg_catalog          | pg_ts_parser              | table | cloudsqladmin
pg_catalog          | pg_ts_template            | table | cloudsqladmin
pg_catalog          | pg_type                   | table | cloudsqladmin
pg_catalog          | pg_user_mapping           | table | cloudsqladmin
public              | flyway_history            | table | postgres
safe                | courses                   | table | postgres
unsafe              | courses                   | table | postgres
(72 rows)
:
```

```
2019-11-24 22:27:20.764  INFO 15502 --- [  restartedMain] org.apache.catalina.c
2019-11-24 22:27:20.810  INFO 15502 --- [  restartedMain] o.a.c.c.C.[Tomcat].[l
2019-11-24 22:27:20.810  INFO 15502 --- [  restartedMain] o.s.web.context.Conte

Environment env = StandardServletEnvironment {activeProfiles=[], defaultProfile

this.gcpUserPattern =   emevig
vs.
gcpUser =       jackdaeel

this.gcpGoPathPattern =     /home/emevig/gopath:/google/gopath
vs.
gcpGoPath =         null

this.gcpPwdPattern =    /home/emevig
vs.
gcpPwd =        /home/jackdaeel/IdeaProjects/SecureAppPractices

this.gcpHomePattern =   /home/emevig
vs.
gcpHome =       /home/jackdaeel
Overridden target to Google Cloud Platform. this.overriddenTarget = gcp
2019-11-24 22:27:20.894  INFO 15502 --- [  restartedMain] o.f.c.internal.licens
2019-11-24 22:27:20.902  INFO 15502 --- [  restartedMain] com.zaxxer.hikari.Hik
2019-11-24 22:27:21.745  INFO 15502 --- [  restartedMain] com.zaxxer.hikari.Hik
```

GCP API

Zewnetrzne API



Stan bezpłatnej wersji próbnej: pozostały środki w wysokości 1 013,45 zł i 352 dni. Dzięki pełnej wersji konta uzyskasz dostęp do wszystkich zasobów Google Cloud Platform.

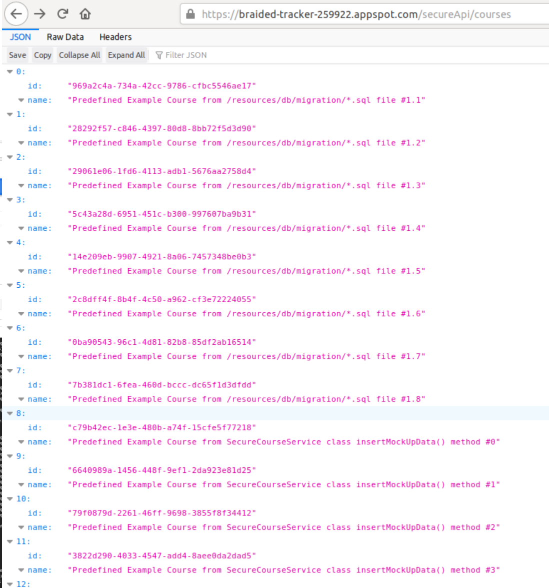
Uprawnienia w projekcie „SecureApplicationPracticesGCP”

Te uprawnienia mają wpływ na projekt i wszystkie jego zasoby. Więcej informacji

Wyświetl według: UŻYTKOWNICY ROLE

□ Typ	Członek ↑	Nazwa	Rola	Uprawnienia nadmiarowe	Dziedziczenie
□	476885512536-compute@developer.gserviceaccount.com	Compute Engine default service account	Edytujący	1417/1422	
□	476885512536@cloudbuild.gserviceaccount.com		Konto usługi Cloud Build		

```
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.134  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.135  INFO 16 --- [           main] o.f.core.internal.command.DbValidate     : Successfully validated 1 migration (execution time 00:00.200s)
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.167  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.238  INFO 16 --- [           main] o.f.c.i.s.JdbcTableSchemaHistory         : Creating Schema History table "public"."flyway_history" ...
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.247  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.342  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.374  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.453  INFO 16 --- [           main] o.f.core.internal.command.DbMigrate      : Current version of schema "public": << Empty Schema >>
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.460  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.553  INFO 16 --- [           main] o.f.core.internal.command.DbMigrate      : Migrating schema "public" with repeatable migration Repeatable
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.556  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.631  INFO 16 --- [           main] o.f.c.i.s.DefaultSqlScriptExecutor       : DB: drop cascades to table courses
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.642  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.644  INFO 16 --- [           main] o.f.c.i.s.DefaultSqlScriptExecutor       : DB: drop cascades to 2 other objects
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.661  WARN 16 --- [           main] o.f.c.i.s.DefaultSqlScriptExecutor       : DB: extension "uuid-ossp" already exists, skipping (SQL State
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.666  INFO 16 --- [           main] o.f.c.i.s.DefaultSqlScriptExecutor       : DB: table "courses" does not exist, skipping
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.729  INFO 16 --- [           main] o.f.c.i.s.DefaultSqlScriptExecutor       : DB: table "courses" does not exist, skipping
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.736  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.768  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.857  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.948  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:17 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:17.952  INFO 16 --- [           main] o.f.core.internal.command.DbMigrate      : Successfully applied 1 migration to schema "public" (executio
2019-12-07 13:20:18 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:18.047  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:18 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:18.148  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:18 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:18.245  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:18 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:18.346  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:18 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:18.441  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:18 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:18.549  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:18 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:18.650  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:18 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:18.737  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:18 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:18.769  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:18 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:18.849  INFO 16 --- [onnection adder] c.g.cloud.sql.core.CoreSocketFactory     : Connecting to Cloud SQL instance [braided-tracker-259922:europ
2019-12-07 13:20:20 default[secure-app-practices-version-from-pom-xml]  backend.app.sec.SecureAppPractices.service.SecureCourseService >> checkIfTwoIdenticalIdCanExist(): Two Courses with same "id" WOULD cause exception.
2019-12-07 13:20:23 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:23.256  INFO 16 --- [           main] pertySourcedRequestMappingHandlerMapping : Mapped URL path [/v2/api-docs] onto method [springfox.document
2019-12-07 13:20:24 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:24.158  INFO 16 --- [           main] o.s.s.concurrent.ThreadPoolTaskExecutor  : Initializing ExecutorService 'applicationTaskExecutor'
2019-12-07 13:20:27 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:27.343  INFO 16 --- [           main] d.s.w.p.DocumentationPluginsBootstrapper : Context refreshed
2019-12-07 13:20:27 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:27.571  INFO 16 --- [           main] d.s.w.p.DocumentationPluginsBootstrapper : Found 1 custom documentation plugin(s)
2019-12-07 13:20:27 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:27.954  INFO 16 --- [           main] s.d.s.w.s.ApiListingReferenceScanner     : Scanning for api listing references
2019-12-07 13:20:29 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:29.941  INFO 16 --- [           main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: selectCourseUsingGET_1
2019-12-07 13:20:29 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:29.953  INFO 16 --- [           main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: deleteCourseUsingDELETE_1
2019-12-07 13:20:29 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:29.955  INFO 16 --- [           main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: insertCourseUsingPOST_1
2019-12-07 13:20:30 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:30.030  INFO 16 --- [           main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: runExecuteGetsNoResultsFro
2019-12-07 13:20:30 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:30.037  INFO 16 --- [           main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: runQueryGetsResultsFromData
2019-12-07 13:20:30 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:30.040  INFO 16 --- [           main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: selectAllCoursesUsingGET_1
2019-12-07 13:20:30 default[secure-app-practices-version-from-pom-xml]  2019-12-07 13:20:30.045  INFO 16 --- [           main] .d.s.w.r.o.CachingOperationNameGenerator : Generating unique operation named: selectCourseUsingGET_2
```

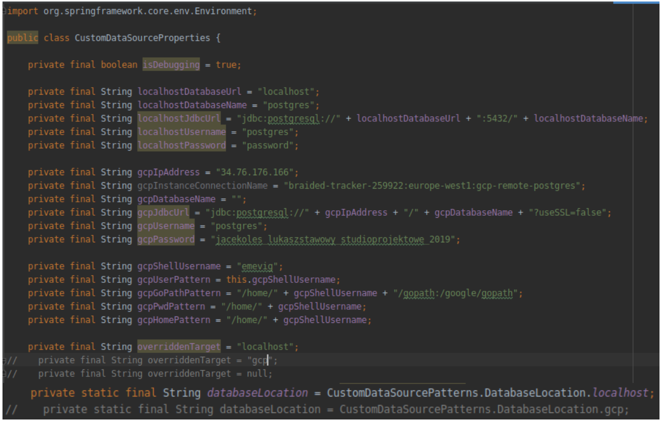

https://braided-tracker-259922.appspot.com/secureApi/courses
JSON
Raw Data
Headers
Save
Copy
Collapse All
Expand All
Filter JSON

```
▾ 0:
    id:     "969a2c4a-734a-42cc-9786-cfbc5546ae17"
  ▾ name:   "Predefined Example Course from /resources/db/migration/*.sql file #1.1"
▾ 1:
    id:     "28292f57-c846-4397-80d8-8bb72f5d3d90"
  ▾ name:   "Predefined Example Course from /resources/db/migration/*.sql file #1.2"
▾ 2:
    id:     "29061e06-1fd6-4113-adb1-5676aa2758d4"
  ▾ name:   "Predefined Example Course from /resources/db/migration/*.sql file #1.3"
▾ 3:
    id:     "5c43a28d-6951-451c-b300-997607ba9b31"
  ▾ name:   "Predefined Example Course from /resources/db/migration/*.sql file #1.4"
▾ 4:
    id:     "14e209eb-9907-4921-8a06-7457348be0b3"
  ▾ name:   "Predefined Example Course from /resources/db/migration/*.sql file #1.5"
▾ 5:
    id:     "2c8dff4f-8b4f-4c50-a962-cf3e72224055"
  ▾ name:   "Predefined Example Course from /resources/db/migration/*.sql file #1.6"
▾ 6:
    id:     "0ba90543-96c1-4d81-82b8-85df2ab16514"
  ▾ name:   "Predefined Example Course from /resources/db/migration/*.sql file #1.7"
▾ 7:
    id:     "7b381dc1-6fea-460d-bccc-dc65f1d3dfdd"
  ▾ name:   "Predefined Example Course from /resources/db/migration/*.sql file #1.8"
▾ 8:
    id:     "c79b42ec-1e3e-480b-a74f-15cfe5f77218"
  ▾ name:   "Predefined Example Course from SecureCourseService class insertMockUpData() method #0"
▾ 9:
    id:     "6640989a-1456-448f-9ef1-2da923e81d25"
  ▾ name:   "Predefined Example Course from SecureCourseService class insertMockUpData() method #1"
▾ 10:
    id:     "79f0879d-2261-46ff-9698-3855f8f34412"
  ▾ name:   "Predefined Example Course from SecureCourseService class insertMockUpData() method #2"
▾ 11:
    id:     "3822d290-4033-4547-add4-8aee0da2dad5"
  ▾ name:   "Predefined Example Course from SecureCourseService class insertMockUpData() method #3"
▾ 12:
```

Bazy danych

GCP vs. (Docker) Localhost

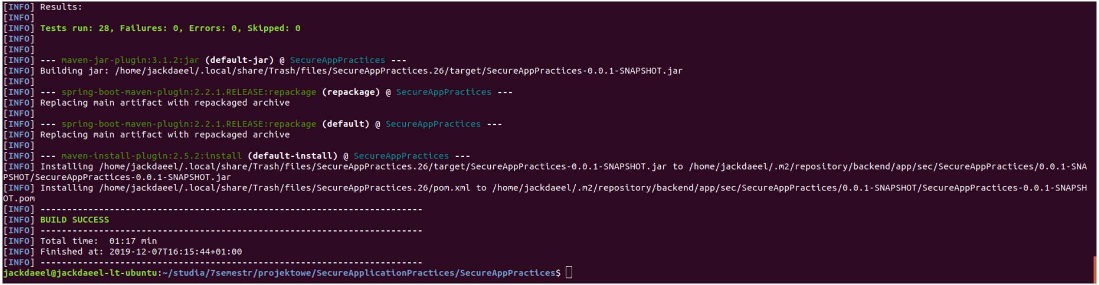
- Wybor bazy danych za pomoca jednej zmiennej lub pozostawienie tego logice aplikacji za pomoca jedej zmiennej



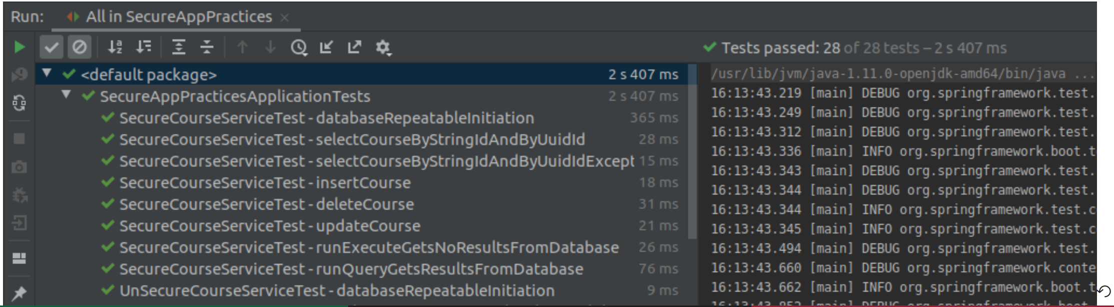
```java
import org.springframework.core.env.Environment;

public class CustomDataSourceProperties {

    private final boolean isDebugging = true;

    private final String localhostDatabaseUrl = "localhost";
    private final String localhostDatabaseName = "postgres";
    private final String localhostJdbcUrl = "jdbc:postgresql://" + localhostDatabaseUrl + ":5432/" + localhostDatabaseName;
    private final String localhostUsername = "postgres";
    private final String localhostPassword = "password";

    private final String gcpIpAddress = "34.76.176.166";
    private final String gcpInstanceConnectionName = "braided-tracker-259922:europe-west1:gcp-remote-postgres";
    private final String gcpDatabaseName = "";
    private final String gcpJdbcUrl = "jdbc:postgresql://" + gcpIpAddress + "/" + gcpDatabaseName + "?useSSL=false";
    private final String gcpUsername = "postgres";
    private final String gcpPassword = "jacekoles_lukaszstawowy_studioprojektowe_2019";

    private final String gcpShellUsername = "emevig";
    private final String gcpUserPattern = this.gcpShellUsername;
    private final String gcpGoPathPattern = "/home/" + gcpShellUsername + "/gopath:/google/gopath";
    private final String gcpPwdPattern = "/home/" + gcpShellUsername;
    private final String gcpHomePattern = "/home/" + gcpShellUsername;

    private final String overriddenTarget = "localhost";
//    private final String overriddenTarget = "gcp";
//    private final String overriddenTarget = null;
    private static final String databaseLocation = CustomDataSourcePatterns.DatabaseLocation.localhost;
//    private static final String databaseLocation = CustomDataSourcePatterns.DatabaseLocation.gcp;
```

Testy

bazy, skryptu, aplikacji back-end'owej



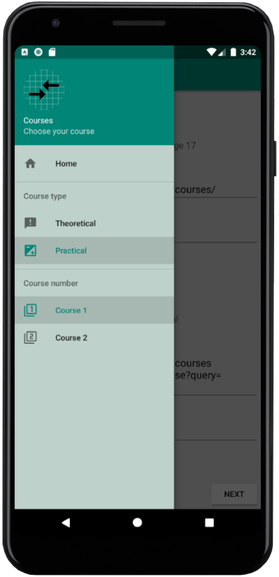
```
[INFO] Results:
[INFO]
[INFO] Tests run: 28, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO]
[INFO] --- maven-jar-plugin:3.1.2:jar (default-jar) @ SecureAppPractices ---
[INFO] Building jar: /home/jackdaeel/.local/share/Trash/files/SecureAppPractices.26/target/SecureAppPractices-0.0.1-SNAPSHOT.jar
[INFO]
[INFO] --- spring-boot-maven-plugin:2.2.1.RELEASE:repackage (repackage) @ SecureAppPractices ---
[INFO] Replacing main artifact with repackaged archive
[INFO]
[INFO] --- spring-boot-maven-plugin:2.2.1.RELEASE:repackage (default) @ SecureAppPractices ---
[INFO] Replacing main artifact with repackaged archive
[INFO]
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ SecureAppPractices ---
[INFO] Installing /home/jackdaeel/.local/share/Trash/files/SecureAppPractices.26/target/SecureAppPractices-0.0.1-SNAPSHOT.jar to /home/jackdaeel/.m2/repository/backend/app/sec/SecureAppPractices/0.0.1-SNA
PSHOT/SecureAppPractices-0.0.1-SNAPSHOT.jar
[INFO] Installing /home/jackdaeel/.local/share/Trash/files/SecureAppPractices.26/pom.xml to /home/jackdaeel/.m2/repository/backend/app/sec/SecureAppPractices/0.0.1-SNAPSHOT/SecureAppPractices-0.0.1-SNAPSH
OT.pom
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  01:17 min
[INFO] Finished at: 2019-12-07T16:15:44+01:00
[INFO] ------------------------------------------------------------------------
jackdaeel@jackdaeel-lt-ubuntu:~/studia/7semestr/projektowe/SecureApplicationPractices/SecureAppPractices$
```

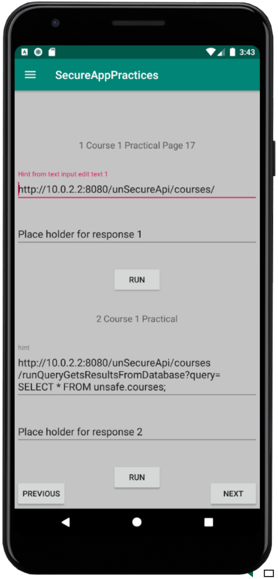


```
✓ Tests passed: 28 of 28 tests – 2 s 407 ms

▼ ✓ <default package>                                                    2 s 407 ms
  ▼ ✓ SecureAppPracticesApplicationTests                                 2 s 407 ms
      ✓ SecureCourseServiceTest - databaseRepeatableInitiation             365 ms
      ✓ SecureCourseServiceTest - selectCourseByStringIdAndByUuidId         28 ms
      ✓ SecureCourseServiceTest - selectCourseByStringIdAndByUuidIdExcept   15 ms
      ✓ SecureCourseServiceTest - insertCourse                              18 ms
      ✓ SecureCourseServiceTest - deleteCourse                              31 ms
      ✓ SecureCourseServiceTest - updateCourse                              21 ms
      ✓ SecureCourseServiceTest - runExecuteGetsNoResultsFromDatabase       26 ms
      ✓ SecureCourseServiceTest - runQueryGetsResultsFromDatabase           76 ms
      ✓ UnSecureCourseServiceTest - databaseRepeatableInitiation             9 ms

/usr/lib/jvm/java-1.11.0-openjdk-amd64/bin/java ...
16:13:43.219 [main] DEBUG org.springframework.test.
16:13:43.249 [main] DEBUG org.springframework.test.
16:13:43.312 [main] DEBUG org.springframework.test.
16:13:43.336 [main] INFO org.springframework.boot.t
16:13:43.343 [main] DEBUG org.springframework.test.
16:13:43.344 [main] DEBUG org.springframework.test.
16:13:43.344 [main] INFO org.springframework.test.c
16:13:43.345 [main] INFO org.springframework.test.c
16:13:43.494 [main] DEBUG org.springframework.test.
16:13:43.660 [main] DEBUG org.springframework.conte
16:13:43.662 [main] INFO org.springframework.boot.t
```

Android

Aplikacja





1 Course 1 Practical Page 17

http://10.0.2.2:8080/unSecureApi/courses/

Place holder for response 1

2 Course 1 Practical

http://10.0.2.2:8080/unSecureApi/courses
/runQueryGetsResultsFromDatabase?query=
SELECT * FROM unsafe.courses;

Place holder for response 2

Part III

Frontend

Frontend

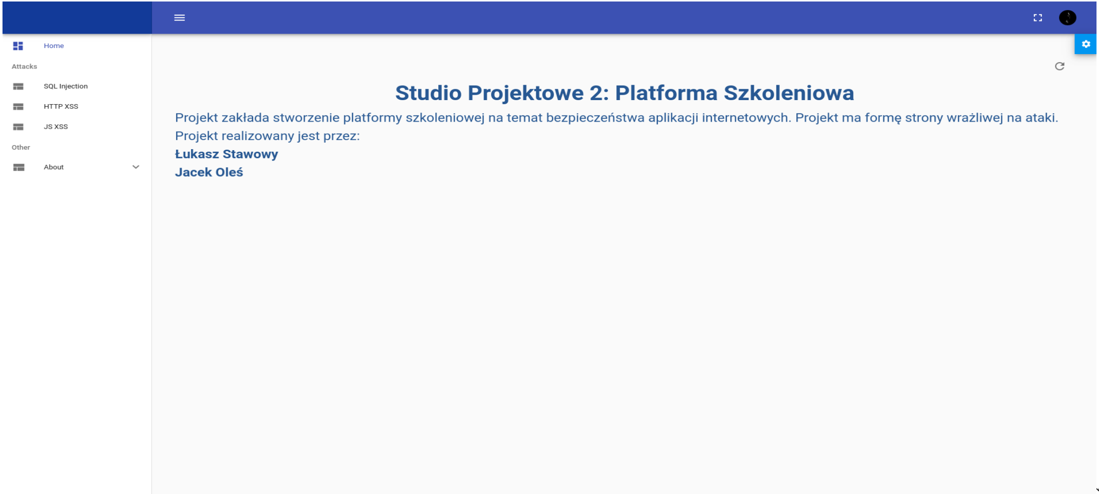
Slowem wstepu

- Jezeli chodzi o front, to udalo sie w pewnym stopniu poprawic jego wyglad.
- Wybor motywu pozostal ten sam, tak jak i ogolny design.

Frontend

HomePage

- Ta sekcja jest tylko dla osób o stalowych nerwach, jeżeli nie możecie znieść brzydko wygladajacych aplikacji, to prosze o zamkniecie oczu.



Studio Projektowe 2: Platforma Szkoleniowa

Projekt zakłada stworzenie platformy szkoleniowej na temat bezpieczeństwa aplikacji internetowych. Projekt ma formę strony wrażliwej na ataki.

Projekt realizowany jest przez:

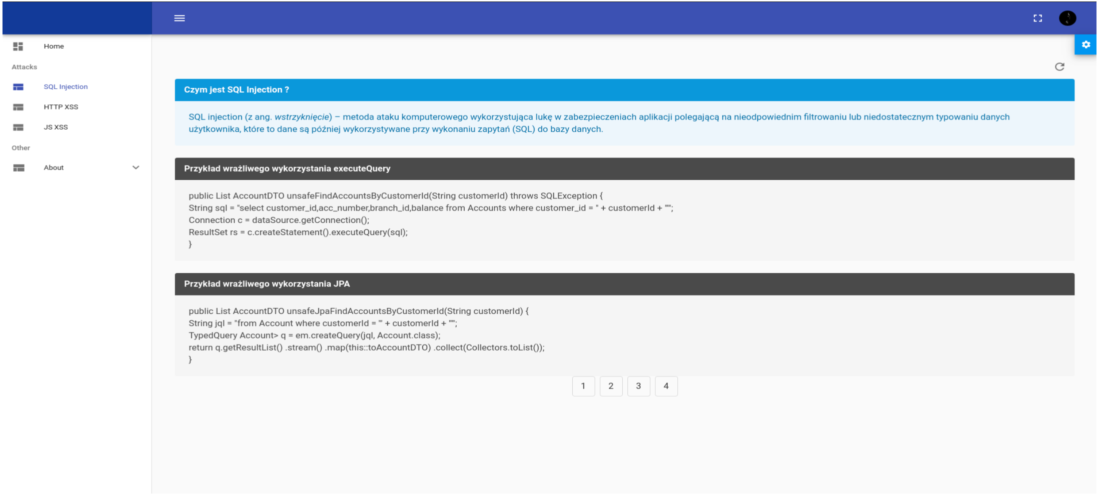
Łukasz Stawowy

Jacek Oleś

Frontend

SQL Injection 1/4

- Widok pierwszej strony kursu SQL Injection



Czym jest SQL Injection ?

SQL injection (z ang. *wstrzyknięcie*) – metoda ataku komputerowego wykorzystująca lukę w zabezpieczeniach aplikacji polegającą na nieodpowiednim filtrowaniu lub niedostatecznym typowaniu danych użytkownika, które to dane są później wykorzystywane przy wykonaniu zapytań (SQL) do bazy danych.

Przykład wrażliwego wykorzystania executeQuery

```
public List AccountDTO unsafeFindAccountsByCustomerId(String customerId) throws SQLException {
String sql = "select customer_id,acc_number,branch_id,balance from Accounts where customer_id = " + customerId + "";
Connection c = dataSource.getConnection();
ResultSet rs = c.createStatement().executeQuery(sql);
}
```

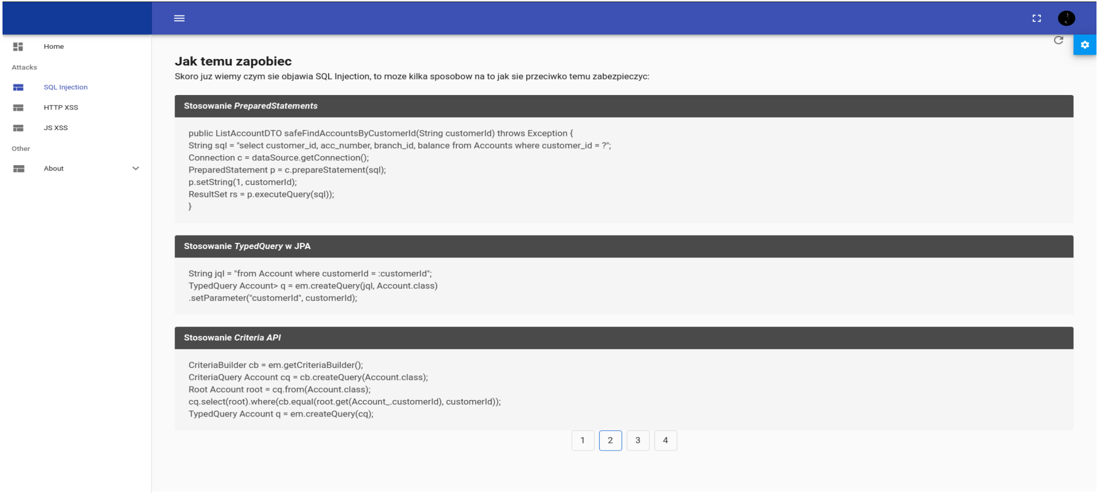
Przykład wrażliwego wykorzystania JPA

```
public List AccountDTO unsafeJpaFindAccountsByCustomerId(String customerId) {
String jql = "from Account where customerId = '" + customerId + "'";
TypedQuery Account> q = em.createQuery(jql, Account.class);
return q.getResultList() .stream() .map(this::toAccountDTO) .collect(Collectors.toList());
}
```

Frontend

SQL Injection 2/4

- Widok drugiej strony kursu SQL Injection



Jak temu zapobiec

Skoro juz wiemy czym sie objawia SQL Injection, to moze kilka sposobow na to jak sie przeciwko temu zabezpieczyc:

Stosowanie *PreparedStatements*

```
public ListAccountDTO safeFindAccountsByCustomerId(String customerId) throws Exception {
String sql = "select customer_id, acc_number, branch_id, balance from Accounts where customer_id = ?";
Connection c = dataSource.getConnection();
PreparedStatement p = c.prepareStatement(sql);
p.setString(1, customerId);
ResultSet rs = p.executeQuery(sql));
}
```

Stosowanie *TypedQuery* w JPA

```
String jql = "from Account where customerId = :customerId";
TypedQuery Account> q = em.createQuery(jql, Account.class)
.setParameter("customerId", customerId);
```

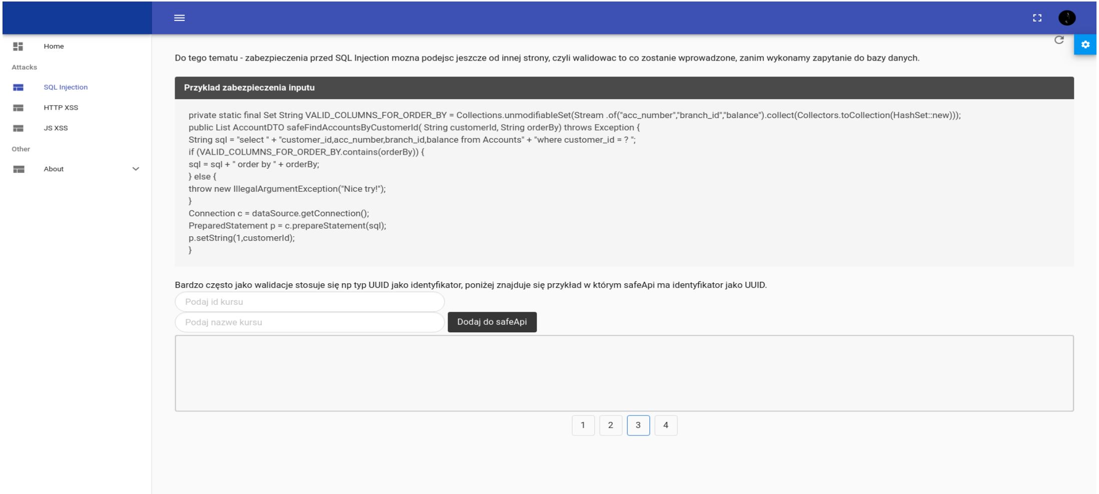
Stosowanie *Criteria API*

```
CriteriaBuilder cb = em.getCriteriaBuilder();
CriteriaQuery Account cq = cb.createQuery(Account.class);
Root Account root = cq.from(Account.class);
cq.select(root).where(cb.equal(root.get(Account_.customerId), customerId));
TypedQuery Account q = em.createQuery(cq);
```

Frontend

SQL Injection 3/4

- Widok trzeciej strony kursu SQL Injection



Do tego tematu - zabezpieczenia przed SQL Injection mozna podejsc jeszcze od innej strony, czyli walidowac to co zostanie wprowadzone, zanim wykonamy zapytanie do bazy danych.

Przyklad zabezpieczenia inputu

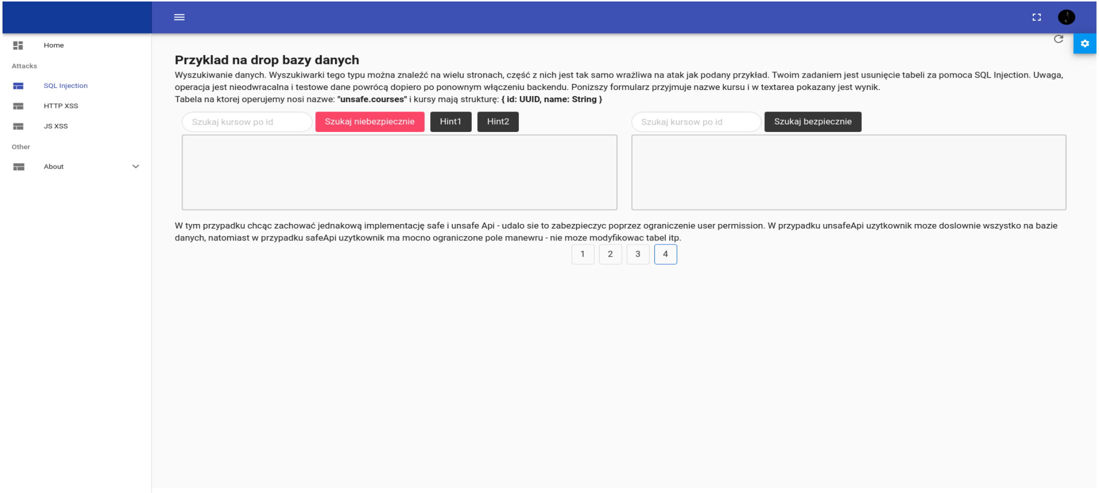
```
private static final Set String VALID_COLUMNS_FOR_ORDER_BY = Collections.unmodifiableSet(Stream .of("acc_number","branch_id","balance").collect(Collectors.toCollection(HashSet::new)));
public List AccountDTO safeFindAccountsByCustomerId( String customerId, String orderBy) throws Exception {
String sql = "select " + "customer_id,acc_number,branch_id,balance from Accounts" + "where customer_id = ? ";
if (VALID_COLUMNS_FOR_ORDER_BY.contains(orderBy)) {
sql = sql + " order by " + orderBy;
} else {
throw new IllegalArgumentException("Nice try!");
}
Connection c = dataSource.getConnection();
PreparedStatement p = c.prepareStatement(sql);
p.setString(1,customerId);
}
```

Bardzo często jako walidacje stosuje się np typ UUID jako identyfikator, poniżej znajduje się przykład w którym safeApi ma identyfikator jako UUID.

Frontend

SQL Injection 4/4

- Widok ostatniej strony kursu SQL Injection



Przyklad na drop bazy danych

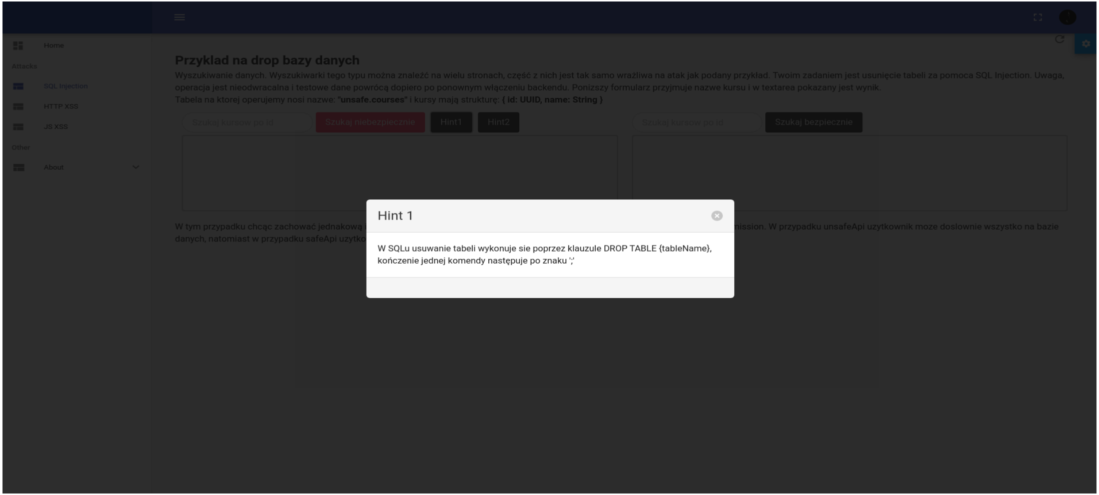
Wyszukiwanie danych. Wyszukiwarki tego typu można znaleźć na wielu stronach, część z nich jest tak samo wrażliwa na atak jak podany przykład. Twoim zadaniem jest usunięcie tabeli za pomoca SQL Injection. Uwaga, operacja jest nieodwracalna i testowe dane powrócą dopiero po ponownym włączeniu backendu. Ponizszy formularz przyjmuje nazwe kursu i w textarea pokazany jest wynik.
Tabela na ktorej operujemy nosi nazwe: **"unsafe.courses"** i kursy mają strukturę: **{ id: UUID, name: String }**

W tym przypadku chcąc zachować jednakową implementację safe i unsafe Api - udalo sie to zabezpieczyc poprzez ograniczenie user permission. W przypadku unsafeApi uzytkownik moze doslownie wszystko na bazie danych, natomiast w przypadku safeApi uzytkownik ma mocno ograniczone pole manewru - nie moze modyfikowac tabel itp.

Frontend

Hint

- Hint z alertu zostal zmieniony na modal



Przyklad na drop bazy danych

Wyszukiwanie danych. Wyszukiwarki tego typu można znaleźć na wielu stronach, część z nich jest tak samo wrażliwa na atak jak podany przykład. Twoim zadaniem jest usunięcie tabeli za pomoca SQL Injection. Uwaga, operacja jest nieodwracalna i testowe dane powrócą dopiero po ponownym włączeniu backendu. Ponizszy formularz przyjmuje nazwe kursu i w textarea pokazany jest wynik.
Tabela na ktorej operujemy nosi nazwe: **"unsafe.courses"** i kursy mają strukturę: **{ id: UUID, name: String }**

W tym przypadku chcąc zachować jednakową i … mission. W przypadku unsafeApi uzytkownik moze doslownie wszystko na bazie danych, natomiast w przypadku safeApi uzytko…

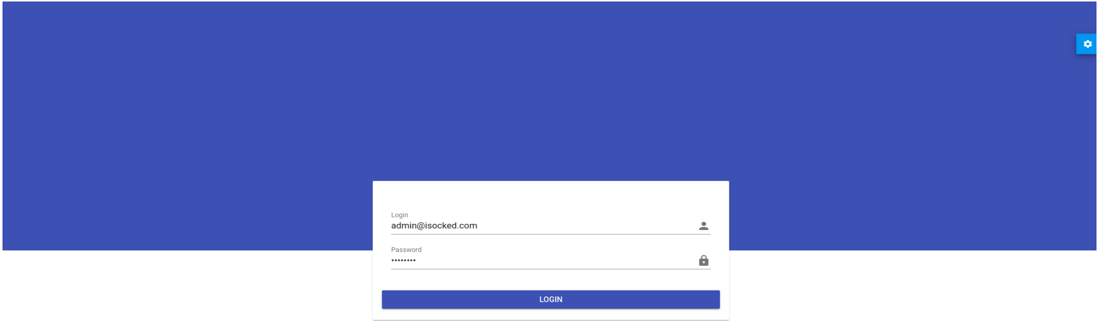
Hint 1

W SQLu usuwanie tabeli wykonuje sie poprzez klauzule DROP TABLE {tableName}, kończenie jednej komendy następuje po znaku ';'

Frontend

Login page

- Strona logowania dalej pozostala zamockowana.



Login
admin@isocked.com

Password
••••••••

Frontend

Scenariusz testu manualnego 1/4

- Nazwa: Hint
- Warunki wstepne: Uzytkownik trafia na strone platformy
- Kroki wykonania:
  1. Uzytkownik wlacza kurs SQL Injection
  2. Uzytkownik wybiera 4 strone kursu
  3. Uzytkownik wlacza hint
- Oczekiwany rezultat: Pokazuje sie modal zawierajacy pomoc w zrealizowaniu kursu

Frontend

Scenariusz testu manualnego 2/4

- Nazwa: Wyszukiwanie kursow
- Warunki wstepne: Uzytkownik trafia na strone platformy
- Kroki wykonania:
  1. Uzytkownik wlacza kurs SQL Injection
  2. Uzytkownik wybiera 4 strone kursu
  3. Uzytkownik wpisuje w input odpowiednie id kursu i naciska przycisk szukaj.
- Oczekiwany rezultat: W textarea pokazuja sie dane podanego kursu.

Frontend

Scenariusz testu manualnego 3/4

- Nazwa: Wykonanie SQL Injection(drop bazy danych)
- Warunki wstepne: Uzytkownik trafia na strone platformy
- Kroki wykonania:
  1. Uzytkownik wlacza kurs SQL Injection
  2. Uzytkownik wybiera 4 strone kursu
  3. Uzytkownik wpisuje w input ”id; DROP TABLE unsafe.courses” i naciska przycisk szukaj.
- Oczekiwany rezultat: Baza danych zostaje usunieta.

Frontend

Scenariusz testu manualnego 4/4

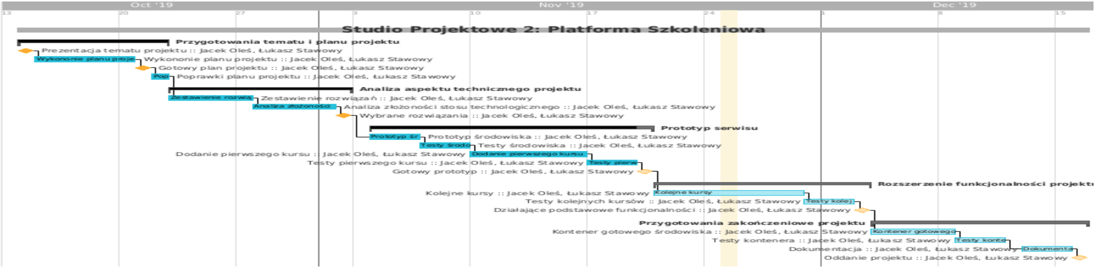
- Nazwa: Dodanie kursu
- Warunki wstepne: Uzytkownik trafia na strone platformy
- Kroki wykonania:
  1. Uzytkownik wlacza kurs SQL Injection
  2. Uzytkownik wybiera 3 strone kursu
  3. Uzytkownik wpisuje w pole id wartosc o formacie uuid np. 9418f043-0e46-484d-9c00-54597d92647d
  4. Uzytkownik wpisuje w pole nazwy dowolna wartosc.
  5. Uzytkownik naciska przycisk dodawania.
- Oczekiwany rezultat: W bazie danych pojawia sie nowy kurs.

Part IV

Podsumowanie

To do

Plany rozwoju



- Dodanie funkcjonalnosci logowania i profilu.
- Dodanie kolejnych kursow.

Bibliography I

Wikibooks
LaTeX/Source Code Listings
https://en.wikibooks.org/wiki/LaTeX/Source_Code_Listings

Till Tantau, Joseph Wright, Vedran Miletić
The beamer class
http://mirror.ctan.org/macros/latex/contrib/beamer/doc/beameruserguide.pdf

Leslie Lamport
LATEX: a document preparation system : user's guide and reference manual
Addison-Wesley Pub. Co., 1994

Pavithra Gunasekara
Latex Installation Tutorial
https://dzone.com/articles/installing-latex-ubuntu
DZone - Open Source Zone

Bibliography II

Stanisław Polak - Polaksta
beamer-AGH
Github - Latex - AGH Template Presentation
https://github.com/polaksta/beamer-AGH