deti universidade de aveiro
departamento de eletrónica,
telecomunicações e informática

HW1: Mid-term assignment report

Eva Pomposo Bartolomeu [98513], v2022-05-01

1 Introduction

1.1 Overview of the work

Este relatório apresenta o projeto individual de médio prazo necessário para o TQS, abrangendo tanto os recursos do produto de software quanto a estratégia de garantia de qualidade adotada.

A aplicação web desenvolvida cujo o nome é Statistics covid-19, permite aos utilizadores analisar as métricas Covid-19 de sempre por país, além disso permite também ver por data e país.

1.2 Current limitations

Durante a realização deste projeto encontrei alguns obstáculos, alguns ultrapassados e outros não.Por exemplo, um dos problemas que obtive, foi na realização da funcionalidade que mostra ao utilizador todos os dados de Covid-19 registados na história de um país. Esta funcionalidade na maior parte das vezes não funciona para países que têm muitos dados, como é o caso da USA. Esta limitação já foi um problema que esperava ao escolher a API externa que ia usar.

Um outro aspecto que veio dificultar no futuro foi o facto de ter utilizado HashMap para a Cache. Na realização dos testes de integração onde queria testar a Cache e a API, foi difícil porque a API utilizava uma instância da Cache, e era complicado manipular esta, talvez se usasse métodos estáticos, e atributos estáticos na Cache não teria este problema.

2 Product specification

2.1 Functional scope and supported interactions

A aplicação web desenvolvida, como dito anteriormente, permite aos utilizadores visualizar as estatísticas Covid-19 registadas em toda a história, ou de um determinado dia, de um país. O website ainda demonstra estatísticas da Cache, e a procura de países durante a seleção do país que pretende analisar.

As entidades que fazem estudos científicos sobre Covid-19 são um possível ator para esta aplicação, representando assim o cliente do website. Assim os cenários onde este ator pretende saber as estatísticas de Covid-19 de toda a história, ou de um dia, de um país podem ser concretizadas com esta aplicação.
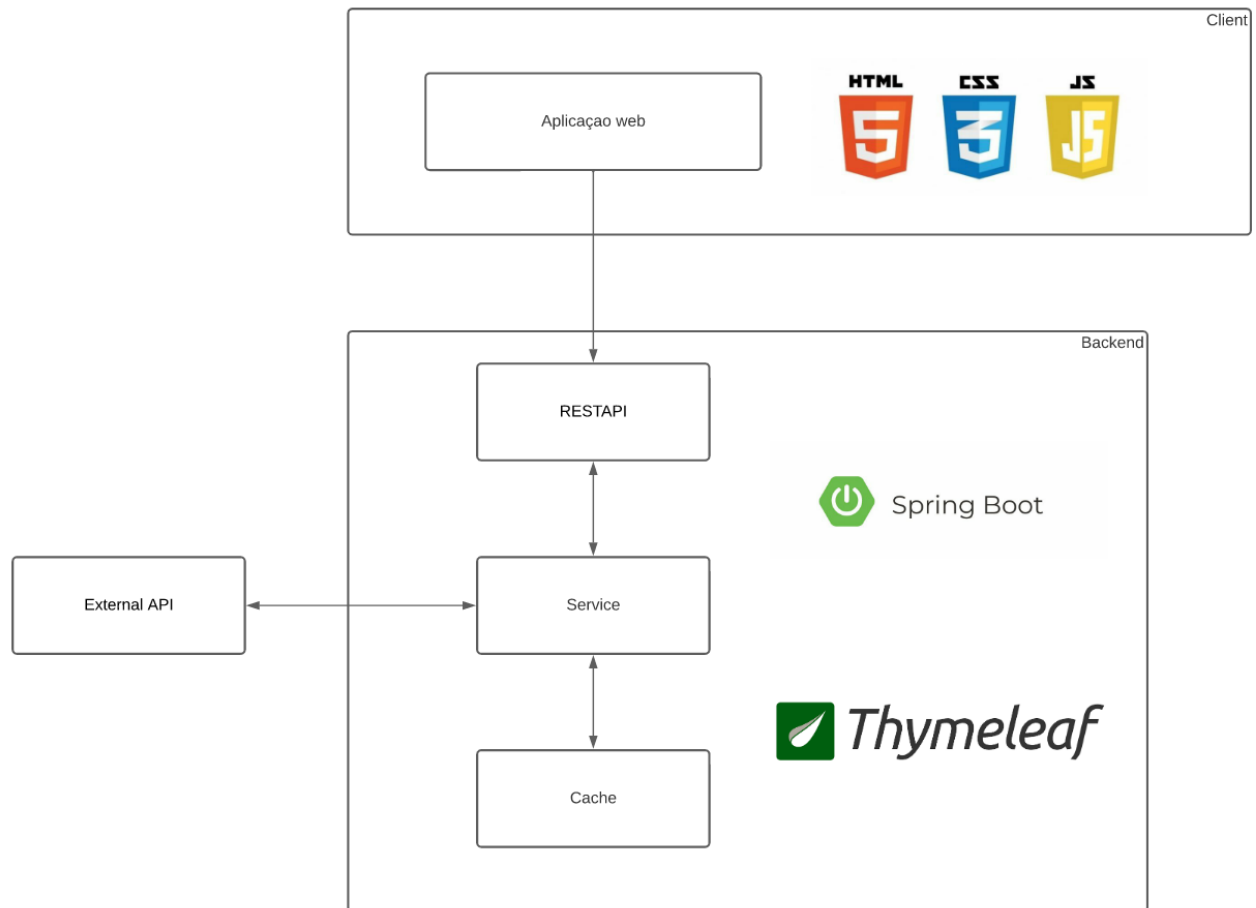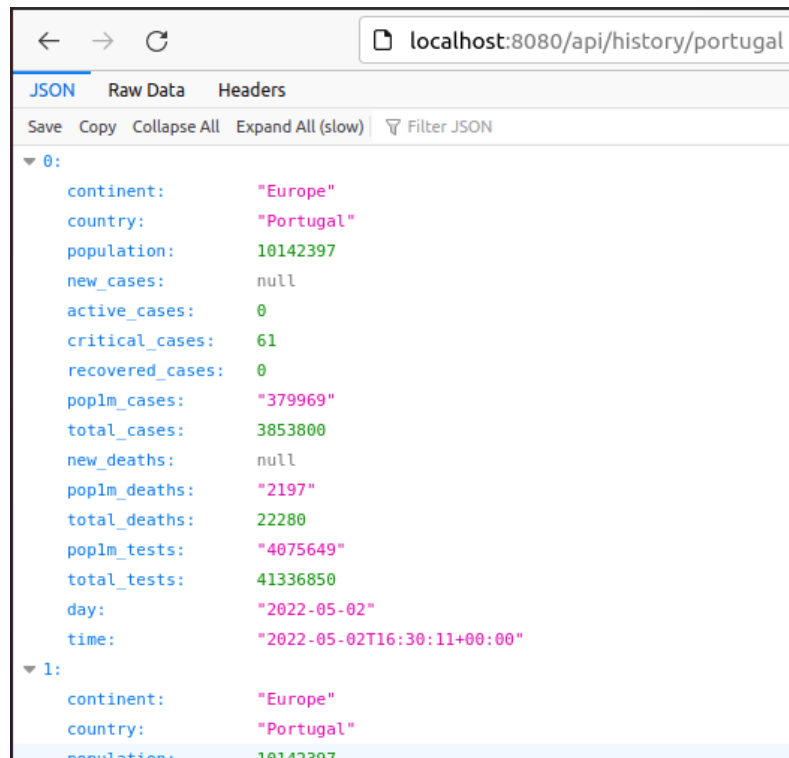
2.2 System architecture

2.3 API for developers

Os endpoints da API (Base URL: http://localhost:8080/api):

- **GET /history/{country}** fornece todos os dados de Covid-19 registados na história de um país

- **GET /history/{country}/{day}** fornece todos os dados de Covid registados num determinado dia de um país
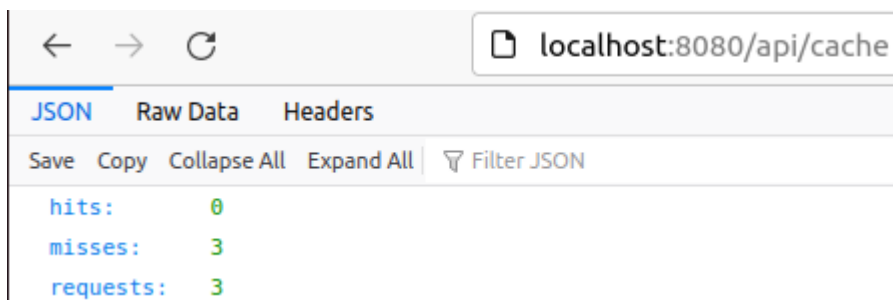
- **GET /countries** fornece todos os países que têm estatísticas na API
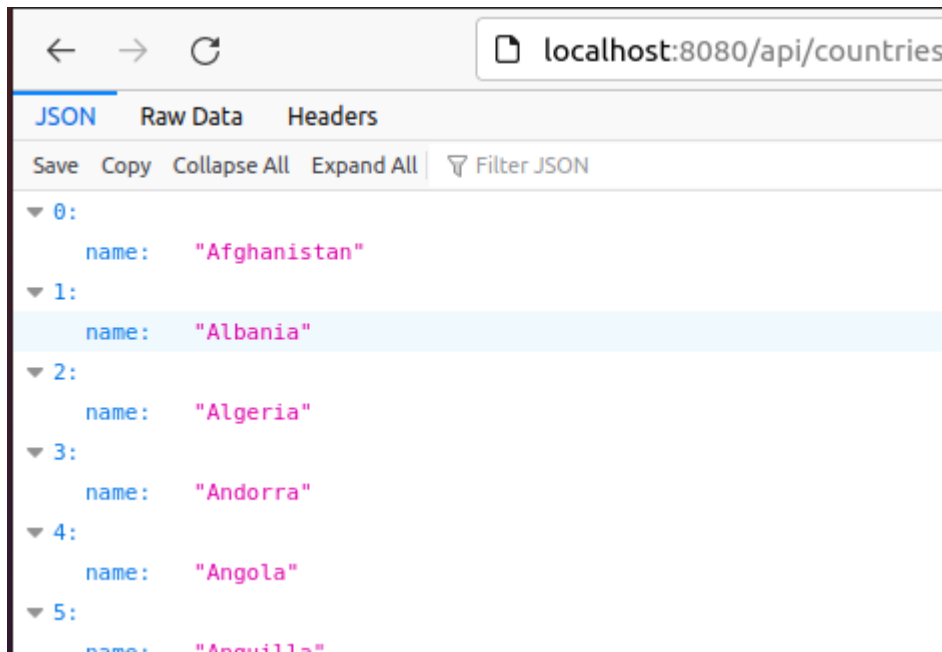
- **GET /cache** fornece todos os países que têm estatísticas na API

3 Quality assurance

3.1 Overall strategy for testing

Neste projeto não foi seguido uma abordagem TDD, ou seja, não realizei os testes antes do desenvolvimento do frontend e do backend. Isto porque inicialmente os requisitos não estavam bem definidos, sendo arriscado declarar os requisitos como testes de unidade. Portanto, adotei a estratégia BDD, onde declarei os requisitos como requisitos.

Em relação às ferramentas de testes usadas, nos testes unitários usei o JUnit5 e MockMvc, nos testes a nível de serviço usei o Mockito e nos testes de integração usei MockMvc e TestRestTemplate.

3.2 Unit and integration testing

3.2.1 Unit Test

- **Covid19ControllerTest**

O teste **Covid19ControllerTest** avalia a API, usa o @WebMvcTest, que simula o comportamento de um servidor da aplicação num ambiente simples, e além disso usa o MockMvc que fornece uma API expressiva. Este teste ainda simula as dependências relacionadas com o service implementation com a anotação @MockBean.

```java
@WebMvcTest(Covid19Controller.class)
class Covid19ControllerTest {
    @Autowired
    private MockMvc mvc;

    @MockBean
    private Covid19Service service;

    @Test
    void givenManyCountries_whenGetCountries_thenReturnJsonArray() throws Exception {
        Country country1 = new Country(name: "Portugal");
        Country country2 = new Country(name: "Albania");
        Country country3 = new Country(name: "Angola");

        List<Country> allCountries = Arrays.asList(country1, country2, country3);

        when( service.getCountries()).thenReturn(allCountries);

        mvc.perform(
            get(urlTemplate: "/api/countries").contentType(MediaType.APPLICATION_JSON))
            .andExpect(status().isOk())
            .andExpect(jsonPath(expression: "$", hasSize(size: 3)))
            .andExpect(jsonPath(expression: "$[0].name", is(country1.getName())))
            .andExpect(jsonPath(expression: "$[1].name", is(country2.getName())))
            .andExpect(jsonPath(expression: "$[2].name", is(country3.getName())));

        verify(service, times(wantedNumberOfInvocations: 1)).getCountries();
    }
}
```

- **Covid19ServiceTest**

O teste **Covid19ServiceTest** avalia de forma isolada o comportamento do Service. Nesta classe usa-se o JUnit5 e o Mockito, logo faz-se os testes necessários para *Covid19Service* sem se testar os métodos da *Cache* e do *Resolver* chamados, fazendo assim mock da *Cache* e do *Resolver*. Nesta interface de teste confia-se apenas no Junit5 + Mockito para controlar testes e para definir expectativas e verificações.

```java
@ExtendWith(MockitoExtension.class)
class Covid19ServiceTest {
    Country country1;
    Country country2;
    Country country3;
    Statistic statistic1;
    Statistic statistic2;
    List<Country> allCountries;
    List<Statistic> allStatistic;
    List<Statistic> emptyStatisticList;
    KeyCacheHistoryWithDay key;

    @Mock( lenient = true)
    private Cache cache;

    @Mock( lenient = true)
    private Resolver resolver;

    @InjectMocks
    private Covid19Service service;

    @BeforeEach
    public void setUp() {

        country1 = new Country(name: "Portugal");
        country2 = new Country(name: "Albania");
        country3 = new Country(name: "Angola");
        allCountries = Arrays.asList(country1, country2, country3);

        statistic1 = new Statistic(continent: "Europe", country: "Portugal", population: 10198931, new_cases: "+195", active_cases: 11590, critical_case
        statistic2 = new Statistic(continent: "Europe", country: "Portugal", population: 10199012, new_cases: "+195", active_cases: 11591, critical_case
        allStatistic = Arrays.asList(statistic1, statistic2);

        key = new KeyCacheHistoryWithDay(country: "Portugal", day: "2020-06-02");

        emptyStatisticList = new ArrayList<>();
    }

    @Test
    void given3Countries_whengetCountries_thenReturn3RecordsThroughCache() throws URISyntaxException, IOException, InterruptedException {
        Mockito.when(cache.getCountries_FromCache()).thenReturn(allCountries);

        List<Country> countries = service.getCountries();
        verifyGetCountries_FromCacheIsCalledOnce();
        assertThat(countries).hasSize(expected: 3).extracting(Country::getName).contains(country1.getName(), country2.getName(), country3.getName
    }
}
```

deti universidade de aveiro
departamento de eletrónica,
telecomunicações e informática

- **CacheTest**

O teste **CacheTest** é um Unit test que avalia unicamente o comportamento da *Cache* usando o Junit5.

```java
class CacheTest {
    private Cache cache;
    Statistic statistic1;
    Statistic statistic2;
    List<Statistic> allStatistic1;
    KeyCacheHistoryWithDay key;
    Country country1;
    Country country2;
    Country country3;
    List<Country> allCountries;

    @BeforeEach
    void setUp() {
        cache = new Cache(lifeTime: 1);

        statistic1 = new Statistic(continent: "Europe", country: "Portugal", population: 10198931, new_cases: "+195", active_cases: 11590, critical_cas
        statistic2 = new Statistic(continent: "Europe", country: "Portugal", population: 10199012, new_cases: "+195", active_cases: 11591, critical_cas
        allStatistic1 = Arrays.asList(statistic1, statistic2);

        key = new KeyCacheHistoryWithDay(country: "Portugal", day: "2020-06-02");

        country1 = new Country(name: "Portugal");
        country2 = new Country(name: "Albania");
        country3 = new Country(name: "Angola");
        allCountries = Arrays.asList(country1, country2, country3);
    }

    @AfterEach
    void tearDown(){
        cache.clearCache();
    }

    @Test
    void addValue_cacheHistory_withoutDayTest(){
        assertEquals(expected: 0, cache.getCacheHistory_withoutDay().size() );
        cache.addValue_cacheHistory_withoutDay(key: "Portugal", allStatistic1);
        assertEquals(expected: 1, cache.getCacheHistory_withoutDay().size() );
        assertEquals(expected: true, cache.getCacheHistory_withoutDay().containsKey("Portugal") );
        assertEquals(expected: true, cache.getCacheHistory_withoutDay().get("Portugal").equals(allStatistic1) );
    }
```

- **ResolverTest**

O teste **ResolverTest** avalia a conexão com a API externa, e a conversão das respostas da API externa para algo que irá ser legível para o service futuramente, para isto usei o Junit5.

```java
class ResolverTest {
    private Resolver resolver;
    Country country;
    Statistic statistic;

    @BeforeEach
    void setUp() {
        resolver = new Resolver();
        country = new Country(name: "Portugal");
        statistic = new Statistic(continent: "Asia", country: "China", population: 1439323776, new_cases: "+1824", active_cases: 26774, critical_cases:
    }

    @Test
    void callEXternalAPI_InvalidUrl() {
        assertThrows(expectedType: ConnectException.class, () -> {
            resolver.callEXternalAPI(url: "xx");
        });
    }

    @Test
    void callEXternalAPI_ValidUrl() throws URISyntaxException, IOException, InterruptedException {
        assertThat(resolver.callEXternalAPI(url: "/countries")).isNotEmpty();
    }

    @Test
    void convertStringtoListCountriesTest(){
        List<Country> countries = resolver.convertStringtoListCountries(string: "{'response':['Portugal']}");
        assertEquals(expected: 1, countries.size());
        assertEquals(expected: true, countries.contains(country));
    }

    @Test
```

3.2.2 Integration Test

- **Covid19RestControllerIT** and **Covid19RestControllerTemplateIT**
  Tanto o teste **Covid19RestControllerIT** como o **Covid19RestControllerTemplateIT** são testes de integração que envolvem vários componentes. Ambos envolvem as componentes: service implementation, a _Cache_ e o _Resolver_. Os dois testes iniciam o contexto da Web completo através da anotação @SpringBootTest e a API é implementada no contexto SpringBoot. A diferença entre estes dois testes é que nesta API implementada, o teste **Covid19RestControllerIT** usa o Mockvc como entry point para suporte de teste Spring MVC do lado do servidor. Já o teste **Covid19RestControllerTemplateIT** usa o TestRestTemplate, um cliente REST para criar pedidos realistas, envolvendo também respostas.

```java
@SpringBootTest(webEnvironment = WebEnvironment.MOCK, classes = Covid19ServiceApplication.class)
@AutoConfigureMockMvc
class Covid19RestControllerIT {

    @Autowired
    private MockMvc mvc;

    @Autowired
    private Covid19Service covid19Service;

    @AfterEach
    public void resetCache() {
        covid19Service.clearCache();
    }

    @Test
    void givenCountries_whenGetCountries_thenStatus200ThroughCache() throws Exception {
        createTestCountry(name1: "Portugal", name2: "Spain");

        mvc.perform(get(urlTemplate: "/api/countries").contentType(MediaType.APPLICATION_JSON))
                .andDo(print())
                .andExpect(status().isOk())
                .andExpect(content().contentTypeCompatibleWith(MediaType.APPLICATION_JSON))
                .andExpect(jsonPath(expression: "$", hasSize(greaterThanOrEqualTo(value: 2))))
                .andExpect(jsonPath(expression: "$[0].name", is(value: "Portugal")))
                .andExpect(jsonPath(expression: "$[1].name", is(value: "Spain")));
    }
```

```java
@SpringBootTest(webEnvironment = WebEnvironment.RANDOM_PORT)
class Covid19RestControllerTemplateIT {

    @LocalServerPort
    int randomServerPort;

    @Autowired
    private TestRestTemplate restTemplate;

    @Autowired
    private Covid19Service covid19Service;

    @AfterEach
    public void resetCache() {
        covid19Service.clearCache();
    }

    @Test
    void givenCountries_whenGetCountries_thenStatus200ThroughCache() {
        createTestCountry(name1: "Portugal", name2: "Spain");

        ResponseEntity<List<Country>> response = restTemplate
            .exchange(url: "/api/countries", HttpMethod.GET, requestEntity: null, new ParameterizedTypeReference<List<Country>>() {
        });

        assertThat(response.getStatusCode()).isEqualTo(HttpStatus.OK);
        assertThat(response.getBody()).hasSizeGreaterThanOrEqualTo(boundary: 2).extracting(Country::getName).containsExactly(...values: "Portugal
    }
```

deti · universidade de aveiro
departamento de eletrónica,
telecomunicações e informática

3.3 Functional testing

- **FrontendTest**

No teste **FrontendTest** avaliei a interface web com uma abordagem orientada pelo comportamento (recursos com cenários), usei o SeleniumJupiter. Foi testado o comportamento de 3 cenários, quando o utilizador visualiza os dados Covid-19 registados em toda a história num país, quando visualiza os dados Covid-19 registados num determinado dia de um país, e quando o utilizador visualiza as estatísticas da Cache.

```java
@ExtendWith(SeleniumJupiter.class)
public class FrontendTest {
    private WebDriver driver;
    JavascriptExecutor js;

    @BeforeEach
    void setup() {
        driver = new FirefoxDriver();
        js = (JavascriptExecutor) driver;
    }

    @AfterEach
    public void tearDown() {
      driver.quit();
    }

    @Test
    public void getHistoryWithoutDayTest() {
      driver.get(url: "http://localhost:8080/");
      driver.manage().window().setSize(new Dimension(width: 1848, height: 1053));
      driver.findElement(By.cssSelector(cssSelector: "h2")).click();
      assertEquals(driver.findElement(By.cssSelector(cssSelector: "h2")).getText(), actual: "Analysis");
      driver.findElement(By.cssSelector(cssSelector: ".btn")).click();
      driver.findElement(By.cssSelector(cssSelector: "h2")).click();
      assertEquals(driver.findElement(By.cssSelector(cssSelector: "h2")).getText(), actual: "Analysis: Afghanistan");
      driver.findElement(By.linkText(linkText: "2")).click();
    }

    @Test
    public void getHistoryWithDayTest() {

      driver.get(url: "http://localhost:8080/");
      driver.manage().window().setSize(new Dimension(width: 1848, height: 1053));
      driver.findElement(By.cssSelector(cssSelector: "h2")).click();
      assertEquals(driver.findElement(By.cssSelector(cssSelector: "h2")).getText(), actual: "Analysis");
      driver.findElement(By.cssSelector(cssSelector: ".fa-calendar")).click();
      driver.findElement(By.cssSelector(cssSelector: "tr:nth-child(1) > .day:nth-child(1)")).click();
      driver.findElement(By.cssSelector(cssSelector: ".btn")).click();
      driver.findElement(By.cssSelector(cssSelector: "h2")).click();
      assertEquals(driver.findElement(By.cssSelector(cssSelector: "h2")).getText(), actual: "Analysis: Afghanistan on 05/01/2022");
    }
```
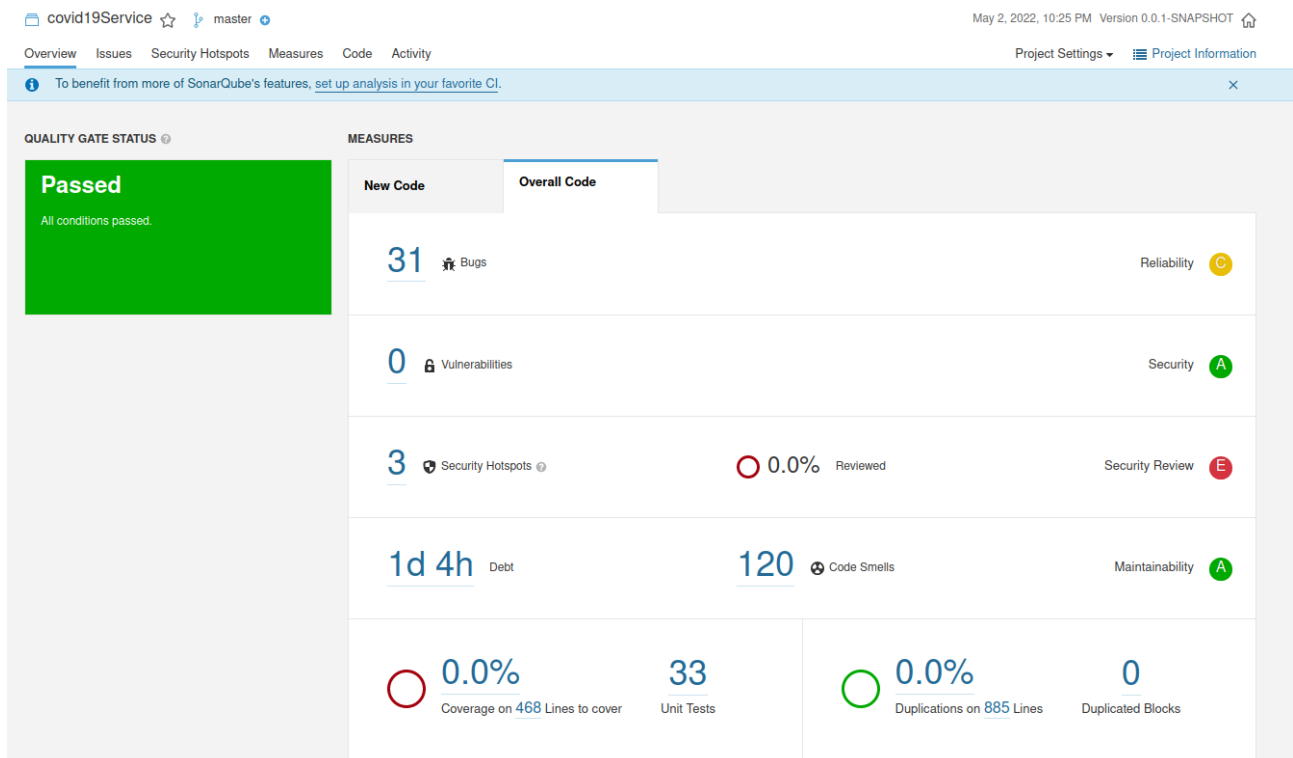
3.4 | Code quality analysis

A ferramenta que utilizei para análise de código estático foi o SonarQube. Na anÁlise que fiz pode se encontrar muitos Bugs, code Smells, mas são quase todos o mesmo problema, esses problemas não foram resolvidos após esta análise por falta de tempo, contudo deixo a minha análise.

Como podemos ver na imagem anterior o meu projeto passou no defined quality gate, usei o default.
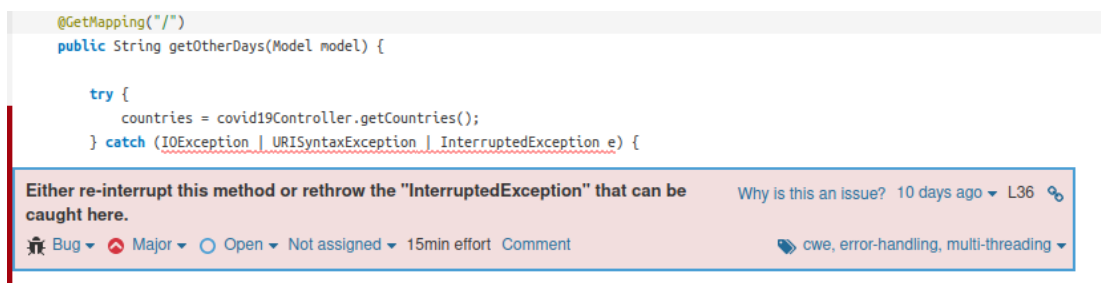
Com o Bug demonstrado nesta imagem, podemos concluir que o Interrupted Exceptions nunca deve ser ignorado no código, deve ser uma exceção tratada.

```
if (!jsonObjectResponse.get("continent").getClass().getName().equals("org.json.JSONObject$Null")) {
```

Use an "instanceof" comparison instead.   Why is this an issue? 6 days ago ▾ L63 🔗

🐞 Bug ▾   ⬤ Major ▾   ◯ Open ▾   Not assigned ▾   5min effort   Comment   🏷️ cert, cwe ▾

```
statistic.setContinent(jsonObjectResponse.getString("continent"));
```

Na imagem anterior usar instanceof porque pode haver mais que um nome possível para uma classe.

```java
@GetMapping("/")
public String getOtherDays(Model model) {

    try {
        countries = covid19Controller.getCountries();
    } catch (IOException | URISyntaxException | InterruptedException e) {
        e.printStackTrace();
```

Make sure this debug feature is deactivated before delivering the code in production.   Comment

A linha de código anterior permite que os programadores e os atacantes encontrem bugs com mais facilidade, o que pode dar acesso a informações detalhadas sobre o sistema que executa o dispositivo e os utilizadores.

📄 src/main/java/com/tqs/covid19Service/model/Statistic.java	4h 4min
📄 src/main/java/com/tqs/covid19Service/model/Cache.java	3h 8min
📄 src/main/java/com/tqs/covid19Service/resolver/Resolver.java	2h 51min
📄 src/main/java/com/tqs/covid19Service/service/Covid19Service.java	41min
📄 src/test/java/com/tqs/covid19Service/model/CacheTest.java	26min
📄 src/test/java/com/tqs/covid19Service/selenium/FrontendTest.java	20min
📄 src/main/java/com/tqs/covid19Service/model/Country.java	10min
📄 src/main/java/com/tqs/covid19Service/controller/Covid19Controller.java	10min
📄 src/main/java/com/tqs/covid19Service/Covid19ServiceApplication.java	10min
📄 src/main/java/com/tqs/covid19Service/inputsForms/FilterOtherDays.java	10min
📄 src/main/java/com/tqs/covid19Service/model/KeyCacheHistoryWithDay.java	10min
📄 src/main/java/com/tqs/covid19Service/controller/Pages.java	10min

O ficheiro Resolver é o que tem o Debt maior porque é aquele que comunica com a API externa, ou seja, lida com muitos dados.

Rename this package name to match the regular expression '^[a-z_]+(\.[a-z_][a-z0-9_]*)*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 10min effort Comment
Why is this an issue? 10 days ago ▾ L1 🔗 ▼▾
🏷 convention ▾

📄 src/.../com/tqs/covid19Service/model/Cache.java

Rename this package name to match the regular expression '^[a-z_]+(\.[a-z_][a-z0-9_]*)*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 10min effort Comment
Why is this an issue? 9 days ago ▾ L1 🔗 ▼▾
🏷 convention ▾

Rename this field "cacheHistory_withoutDay" to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 2min effort Comment
Why is this an issue? 9 days ago ▾ L18 🔗 ▼▾
🏷 convention ▾

Rename this field "cacheHistory_withDay" to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 2min effort Comment
Why is this an issue? 9 days ago ▾ L19 🔗 ▼▾
🏷 convention ▾

Rename this field "timeToLive_cacheHistory_withoutDay" to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 2min effort Comment
Why is this an issue? 9 days ago ▾ L21 🔗 ▼▾
🏷 convention ▾

Rename this field "timeToLive_cacheHistory_withDay" to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 2min effort Comment
Why is this an issue? 9 days ago ▾ L22 🔗 ▼▾
🏷 convention ▾

Rename this field "timeToLive_cacheCountry" to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 2min effort Comment
Why is this an issue? 9 days ago ▾ L23 🔗 ▼▾
🏷 convention ▾

Rename this method name to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 5min effort Comment
Why is this an issue? 5 days ago ▾ L44 🔗 ▼▾
🏷 convention ▾

Rename this method name to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 5min effort Comment
Why is this an issue? 5 days ago ▾ L48 🔗 ▼▾
🏷 convention ▾

Rename this method name to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 5min effort Comment
Why is this an issue? 9 days ago ▾ L56 🔗 ▼▾
🏷 convention ▾

Rename this method name to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 5min effort Comment
Why is this an issue? 9 days ago ▾ L62 🔗 ▼▾
🏷 convention ▾

as=CODE_SMELL # 💡 JUnit5 tes

Return an empty collection instead of null. ⊖
⊗ Code Smell ▾ ⊗ Major ▾ ○ Open ▾ Not assigned ▾ 30min effort Comment
Why is this an issue? 9 days ago ▾ L106 🔗 ▼▾
🏷 cert ▾

Rename this method name to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 5min effort Comment
Why is this an issue? 9 days ago ▾ L109 🔗 ▼▾
🏷 convention ▾

Return an empty collection instead of null. ⊖
⊗ Code Smell ▾ ⊗ Major ▾ ○ Open ▾ Not assigned ▾ 30min effort Comment
Why is this an issue? 9 days ago ▾ L117 🔗 ▼▾
🏷 cert ▾

Rename this method name to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 5min effort Comment
Why is this an issue? 9 days ago ▾ L120 🔗 ▼▾
🏷 convention ▾

Return an empty collection instead of null. ⊖
⊗ Code Smell ▾ ⊗ Major ▾ ○ Open ▾ Not assigned ▾ 30min effort Comment
Why is this an issue? 9 days ago ▾ L128 🔗 ▼▾
🏷 cert ▾

Refactor this method to reduce its Cognitive Complexity from 23 to the 15 allowed.
⊗ Code Smell ▾ ⊗ Critical ▾ ○ Open ▾ Not assigned ▾ 13min effort Comment
Why is this an issue? 5 days ago ▾ L131 🔗 ▼▾
🏷 brain-overload ▾

Rename this method name to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 5min effort Comment
Why is this an issue? 9 days ago ▾ L131 🔗 ▼▾
🏷 convention ▾

This block of commented-out lines of code should be removed.
⊗ Code Smell ▾ ⊗ Major ▾ ○ Open ▾ Not assigned ▾ 5min effort Comment
Why is this an issue? 5 days ago ▾ L152 🔗 ▼▾
🏷 unused ▾

📄 src/.../com/tqs/covid19Service/model/Country.java

Rename this package name to match the regular expression '^[a-z_]+(\.[a-z_][a-z0-9_]*)*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 10min effort Comment
Why is this an issue? 12 days ago ▾ L1 🔗 ▼▾
🏷 convention ▾

📄 src/.../com/tqs/covid19Service/model/KeyCacheHistoryWithDay.java

Rename this package name to match the regular expression '^[a-z_]+(\.[a-z_][a-z0-9_]*)*$'.
⊗ Code Smell ▾ ⊙ Minor ▾ ○ Open ▾ Not assigned ▾ 10min effort Comment
Why is this an issue? 9 days ago ▾ L1 🔗 ▼▾
🏷 convention ▾

📄 src/.../com/tqs/covid19Service/model/Statistic.java

Rename this package name to match the regular expression '^[a-z_]+(\.[a-z_][a-z0-9_]*)*$'. Why is this an issue? 10 days ago ▾ 💡 JUnit5 test cla

deti universidade de aveiro
departamento de eletrónica,
telecomunicações e informática

src/.../tqs/covid19Service/resolver/Resolver.java

Rename this package name to match the regular expression '^[a-z_]+(\.[a-z_][a-z0-9_]*)*$'.
⊗ Code Smell ▾ ☺ Minor ▾ ○ Open ▾ Not assigned ▾ 10min effort Comment
Why is this an issue? 6 days ago ▾ L1 ⚲ ▼▾
🏷 convention ▾

Remove the declaration of thrown exception 'java.net.URISyntaxException', as it cannot be thrown from method's body.
⊗ Code Smell ▾ ☺ Minor ▾ ○ Open ▾ Not assigned ▾ 5min effort Comment
⊖
Why is this an issue? 6 days ago ▾ L24 ⚲ ▼▾
🏷 clumsy, error-handling, redundant, unu... ▾

Define a constant instead of duplicating this literal "response" 4 times.
⊗ Code Smell ▾ ⊙ Critical ▾ ○ Open ▾ Not assigned ▾ 10min effort Comment
Why is this an issue? 6 days ago ▾ L46 ⚲ ▼▾
🏷 design ▾

Refactor this method to reduce its Cognitive Complexity from 40 to the 15 allowed.
⊗ Code Smell ▾ ⊙ Critical ▾ ○ Open ▾ Not assigned ▾ 30min effort Comment
Why is this an issue? 1 day ago ▾ L53 ⚲ ▼▾
🏷 brain-overload ▾

Define a constant instead of duplicating this literal "org.json.JSONObject$Null" 16 times.
⊗ Code Smell ▾ ⊙ Critical ▾ ○ Open ▾ Not assigned ▾ 34min effort Comment
Why is this an issue? 1 day ago ▾ L63 ⚲ ▼▾
🏷 design ▾

Define a constant instead of duplicating this literal "population" 3 times.
⊗ Code Smell ▾ ⊙ Critical ▾ ○ Open ▾ Not assigned ▾ 8min effort Comment
Why is this an issue? 1 day ago ▾ L71 ⚲ ▼▾
🏷 design ▾

Define a constant instead of duplicating this literal "org.json.JSONObject$1" 7 times.
⊗ Code Smell ▾ ⊙ Critical ▾ ○ Open ▾ Not assigned ▾ 16min effort Comment
Why is this an issue? 1 day ago ▾ L71 ⚲ ▼▾
🏷 design ▾

Define a constant instead of duplicating this literal "active" 3 times.
⊗ Code Smell ▾ ⊙ Critical ▾ ○ Open ▾ Not assigned ▾ 8min effort Comment
Why is this an issue? 1 day ago ▾ L81 ⚲ ▼▾
🏷 design ▾

Define a constant instead of duplicating this literal "recovered" 3 times.
⊗ Code Smell ▾ ⊙ Critical ▾ ○ Open ▾ Not assigned ▾ 8min effort Comment
Why is this an issue? 1 day ago ▾ L85 ⚲ ▼▾
🏷 design ▾

Define a constant instead of duplicating this literal "critical" 3 times.
⊗ Code Smell ▾ ⊙ Critical ▾ ○ Open ▾ Not assigned ▾ 8min effort Comment
Why is this an issue? 1 day ago ▾ L89 ⚲ ▼▾
🏷 design ▾

Define a constant instead of duplicating this literal "1M_pop" 6 times.
⊗ Code Smell ▾ ⊙ Critical ▾ ○ Open ▾ Not assigned ▾ 14min effort Comment
Why is this an issue? 6 days ago ▾ L93 ⚲ ▼▾
🏷 design ▾

Define a constant instead of duplicating this literal "total" 9 times.
Why is this an issue? 1 day ago ▾ L
💡 JUnit5 test cl

Por fim é importante dar nomes adequados, e perceptíveis a packages e a variáveis. Além disso, também se deve saber quando se pode criar uma variável para não se repetir muitas vezes expressões.

4   References & resources

Project resources

Resource:	URL/location:
Git repository	https://github.com/eva-pomposo/tqs_98513.git
Video demo	https://youtu.be/8s3qULgKlmi

Reference materials

External API usada: https://rapidapi.com/api-sports/api/covid-193