



Weather App Präsentation

Andreas Enk, Leonardo Pedri, Christopher Stief, Simon Fröhlich, Iolanthe Hochleitner

Vienna, AT

Current Weather

Description: clear sky
Temperature: 0 °C
Feels Like Temperature: -2 °C
Wind Speed: 7 km/h
Wind Direction: SSE
Pop: 0

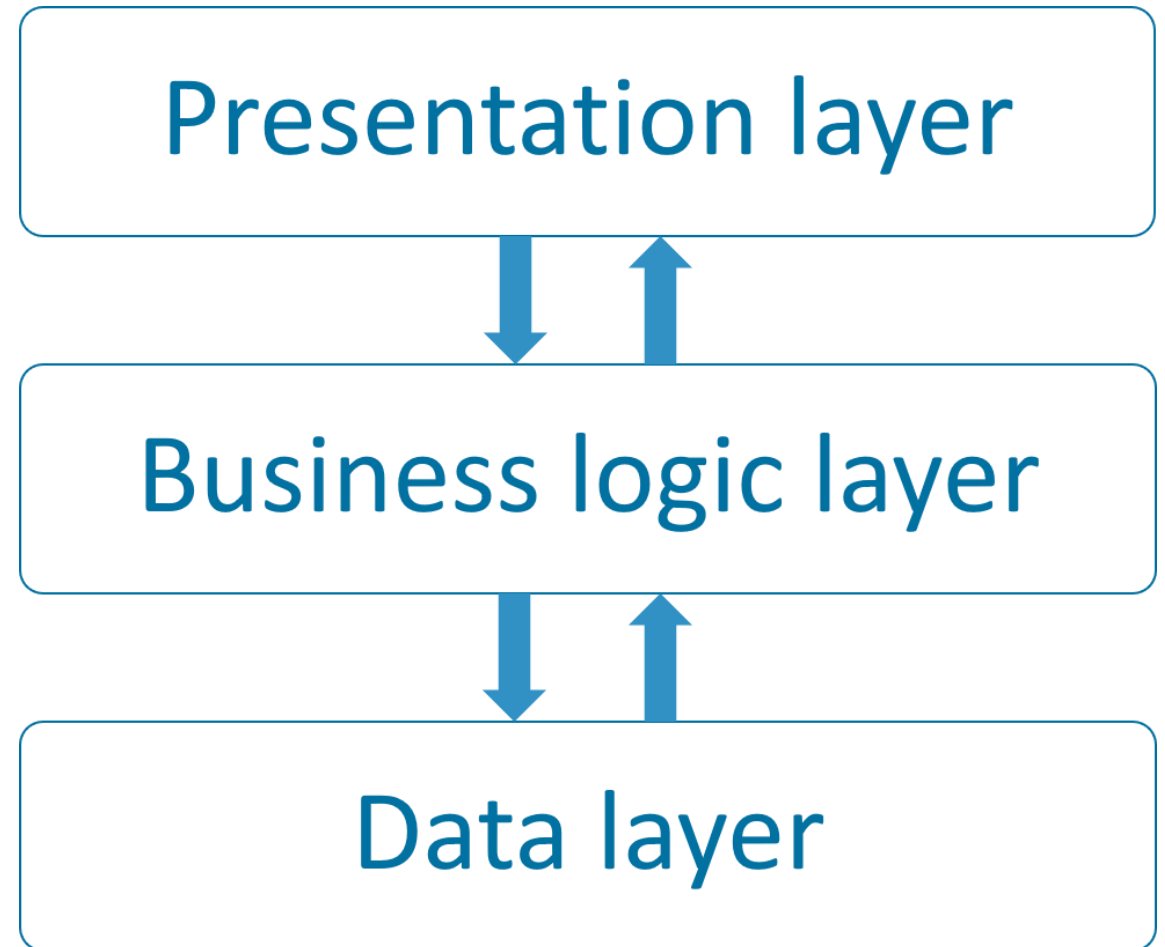


Weather for the next hours

| Time | Weather | Temperature [C] | Feels Like Temperature [C] | Wind speed [km/h] | Wind direction | Pop |
|-------------------|------------|-----------------|----------------------------|-------------------|----------------|-----|
| 29.01.2024: 00:00 | clear sky | 0 | -1 | 5 | SE | 0 |
| 29.01.2024: 01:00 | few clouds | 1 | -1 | 5 | ESE | 0 |
| 29.01.2024: 02:00 | clear sky | 1 | -1 | 7 | ESE | 0 |
| 29.01.2024: 03:00 | clear sky | 1 | -1 | 7 | ESE | 0 |
| 29.01.2024: 04:00 | clear sky | 1 | -1 | 7 | ESE | 0 |
| 29.01.2024: 05:00 | clear sky | 1 | -1 | 7 | ESE | 0 |
| 29.01.2024: 06:00 | clear sky | 1 | -1 | 7 | ESE | 0 |
| 29.01.2024: 07:00 | clear sky | 1 | -2 | 8 | ESE | 0 |
| 29.01.2024: 08:00 | clear sky | 1 | -2 | 8 | ESE | 0 |
| 29.01.2024: 09:00 | clear sky | 2 | -1 | 10 | SE | 0 |

3-Schichten Architektur

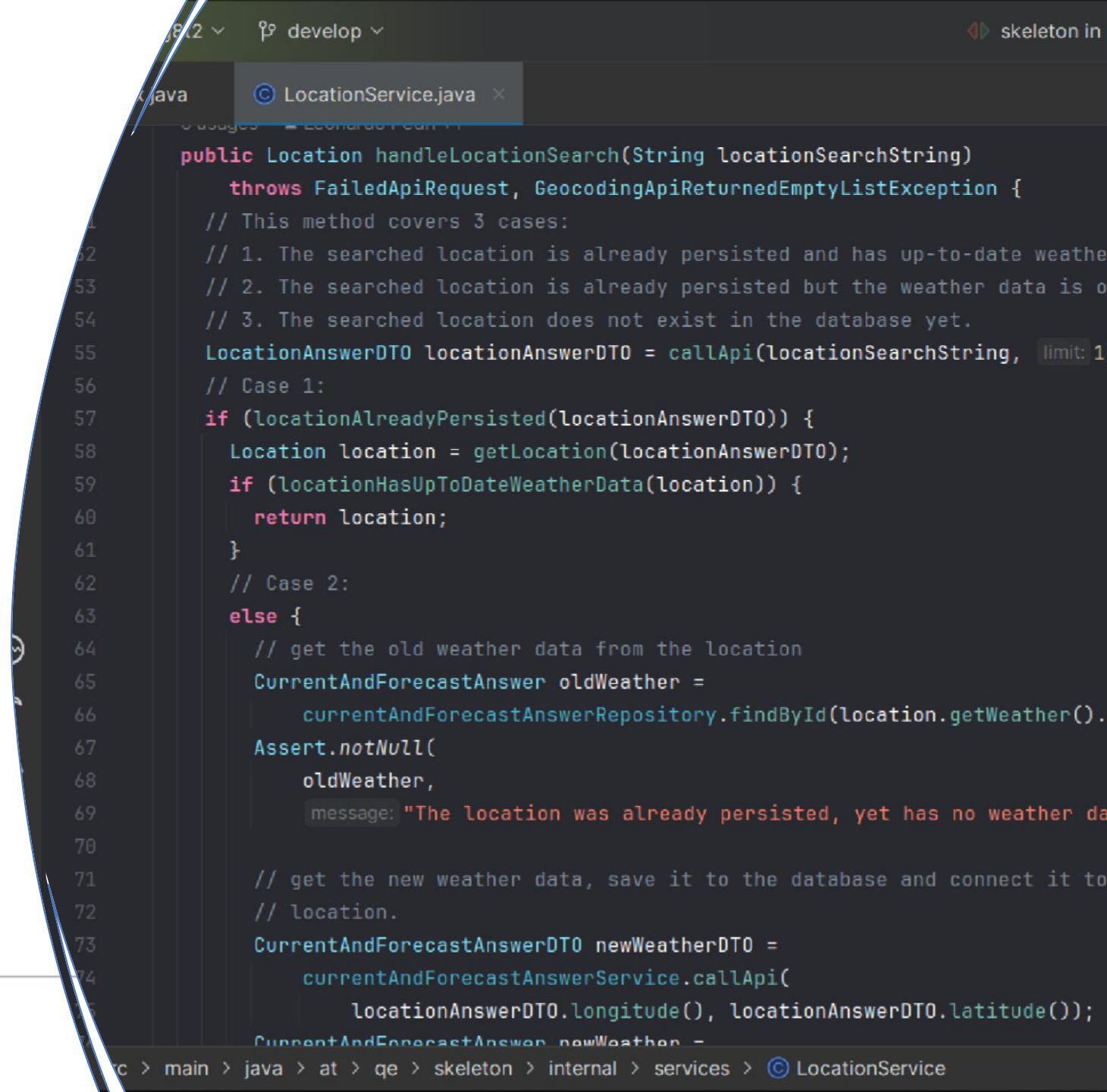
- Persistenzschicht – siehe Model & Repository Klassen
- Anwendungsschicht – siehe Service Klassen
- Präsentationsschicht - siehe Bean & Controller Klassen



Design Patterns

Facade Pattern

- *Convenient access to subsystem's functionality.*
- *Directs the request and how to operate all the moving parts.*



sonarqube

Save and re-use this "Random".

"Random" objects should be reused [java:S2119](#)



Bug



Critical



Closed (Fixed)

Not assigned

5min effort

0 comments

Where is the issue?

Why is this an issue?



60.9%

Coverage on 1.3k Lines
to cover

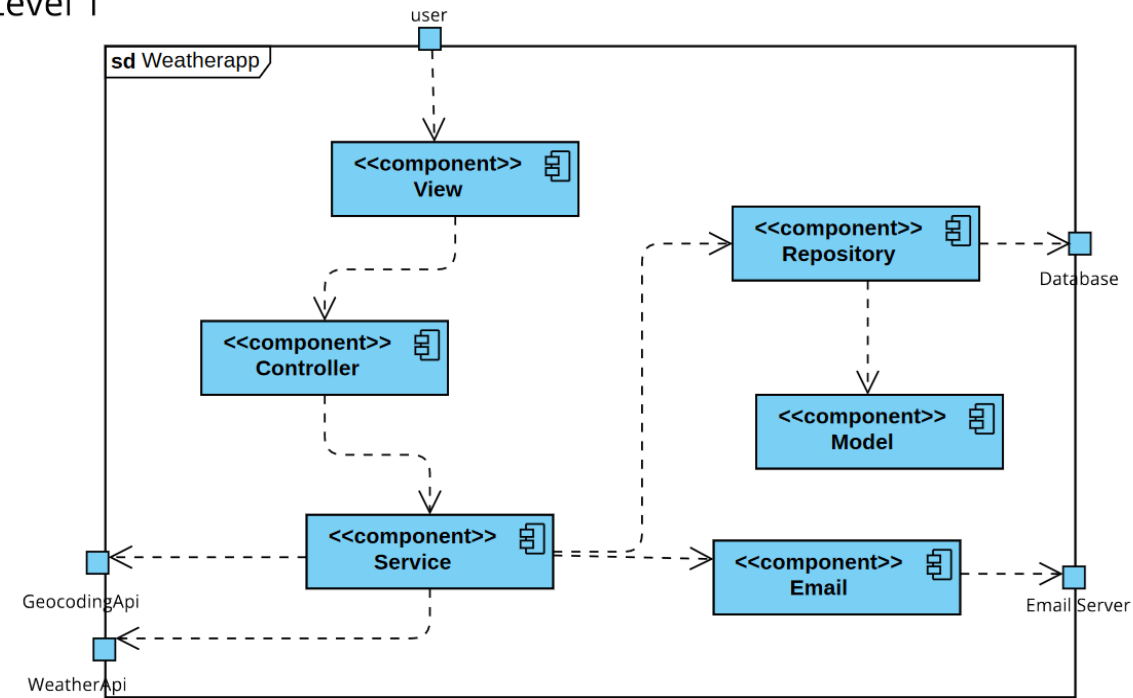
98

Unit
Tests

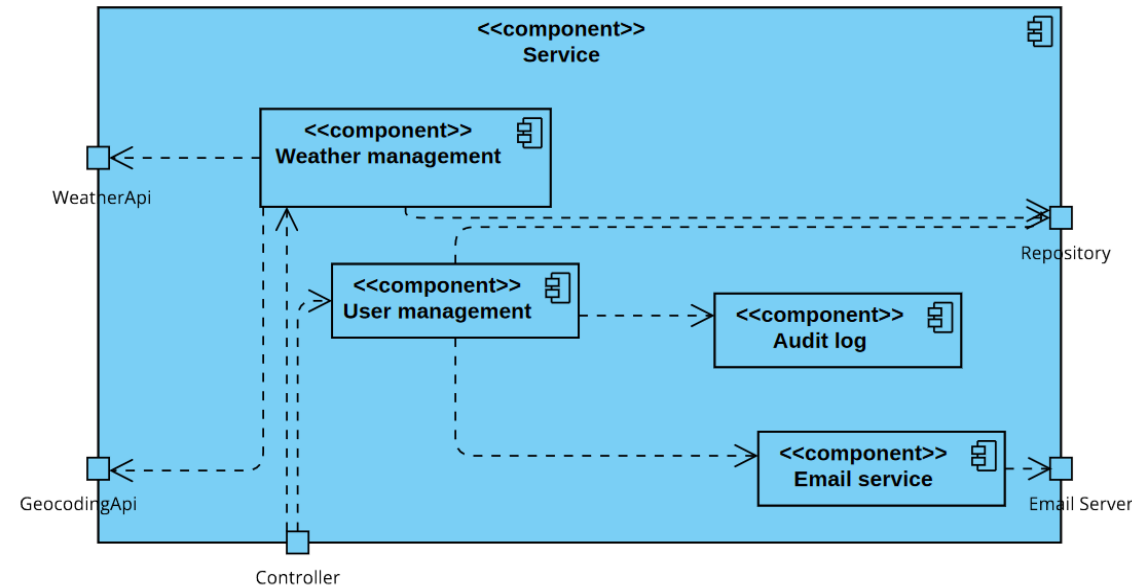
Lessons learned: Diagramme

- Erste Version UML -> groben Projektüberblick
- Einige Änderungen bei tatsächlicher Implementierung
- "Weniger ist manchmal mehr"
 - Reduktion auf das Wesentliche
- Anfängliches Komponenten Diagramm zur Übersicht
- Je weniger Erfahrung, desto hilfreicher sind Diagramme als Orientierungshilfen.

Level 1



Level 2



Lessons Learned: GIT



GIT is your friend:
Nachvollziehbarkeit und
Übersicht über aktuellen Stand
des Projekts



Issues anfangs
grob formulieren
->im aktuellen Milestone Issues
detaillierter ausgestalten



Neuer Issue für Bug fixes



Aufteilung nach features nicht
nach Schichten, weil weniger
Kommunikation nötig war.



Awaiting approval



High priority



In code review



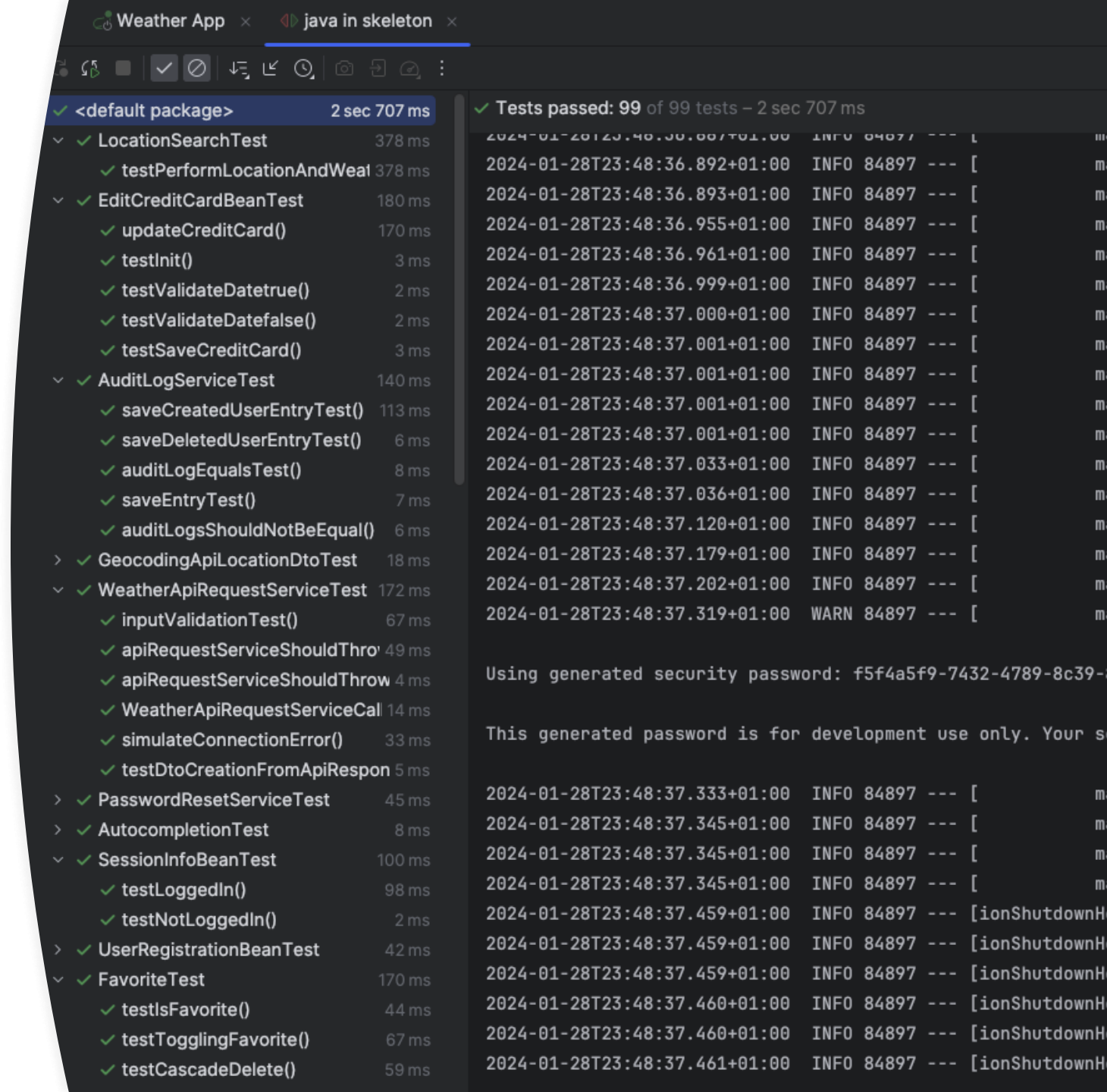
In development



Low priority

Lessons Learned: Testen

- Tests = sinnvoll
- Sicherheit bei Merges & Refactoring
- Was erleichtert das Testen?
 - Kleinere testbare Klassen
 - Spezifische (Custom) Exceptions
 - Constructor Injection



Lessons Learned: Testen

- Constructor Injection:

```
4 usages
33 | @Mock EmailService mockedEmailService;
    11 usages
34 | RegistrationService registrationService;
35 |
36 | @Autowired UserxRepository userxRepository;
37 | @Autowired TokenService tokenService;
38 | @Autowired PasswordEncoder passwordEncoder;
39 | @Autowired UserxService userxService;
40 |
    christophertief
41 | @BeforeEach
42 | public void init() {
43 |     registrationService =
44 |         new RegistrationService(
45 |             this.passwordEncoder, this.mockedEmailService, this.tokenService, this.userxRepository);
46 | }
```

Lessons Learned: Testen

- Field Injection:

```
1 usage
@Mock EmailService emailService;
6 usages
@InjectMocks PasswordResetService passwordResetService;
@Autowired PasswordEncoder passwordEncoder;
@Autowired UserRepository userRepository;

christopherstief
@BeforeEach
public void init() {
    // need to use these because no Constructor Injection
    ReflectionTestUtils.setField(passwordResetService, name: "userRepository", userRepository);
    ReflectionTestUtils.setField(passwordResetService, name: "passwordEncoder", passwordEncoder);
}
```

```
}
ReflectionTestUtils.setField(passwordResetService, name: "passwordEncoder", passwordEncoder);
ReflectionTestUtils.setField(passwordResetService, name: "userRepository", userRepository);
```

Lessons Learned: Testen

- Mockito:

[4 usages](#)

```
@Mock private static WeatherApiRequestService mockedWeatherApiRequestService;
```

```
// verify that the mocked WeatherApiService only got called once  
verify(mockedWeatherApiRequestService, times(wantedNumberOfInvocations: 1))  
    .retrieveCurrentAndForecastWeather(  
        searchedMockLocation.latitude(), searchedMockLocation.longitude());
```

```
when(mockedGeocodingRequestService.retrieveLocationsLonLat(searchedLocationName, limit: 1))  
    .thenReturn(locationDtoListMunich);  
when(mockedWeatherApiRequestService.retrieveCurrentAndForecastWeather(  
    searchedLatitude, searchedLongitude))  
    .thenReturn(searchedMockWeather);
```

Lessons Learned: Testen

- MockRestServiceServer:

```
// stub RestTemplate and Mockserver for mock-api-requests
RestTemplate restTemplate = new RestTemplate();
mockServer = MockRestServiceServer.bindTo(restTemplate).build();
testRestClient = RestClient.create(restTemplate);
```

```
restTemplate = restTemplate.clone();
```

```
mockServer
    .expect(requestTo( expectedUri: "/geo/1.0/direct?q=Innsbruck&limit=" + LIMIT))
    .andExpect(method(HttpMethod.GET))
    .andRespond(withSuccess(apiResponseStringIbk, MediaType.APPLICATION_JSON));
// making the api-request
geocodingApiRequestService.retrieveLocationsLonLat( locationName: "Innsbruck", LIMIT);

// the actual test: verifying if request reached the expected URI
mockServer.verify();
```

```
mockServer.verify();
```

```
// the actual test: verifying if request reached the expected URI
```





lich ☆ main +



Last analysis of this Branch had [1 warning](#)

January 29, 2024 at 12:20 AM Versi

ots Measures Code Activity

Project Settings ▾

Project In

Get pe

27 c

ov



Save and re-use this "Random".

"Random" objects should be reused [java:S2119](#)

...ternal/services/TokenService.java



Bug



Critical



Closed

(Fixed)

Not assigned

5min effort

0 comments

Save and



Bug

Where is the issue?

Why is this an issue?

Save and re-use this "Random".



Bug

...keleton/internal/services/TokenService.java

SonarQube™ technology is powered by [SonarSource SA](#)

9.9 (build 65466) - [LGPL v3](#) - [Community](#) - [Documentation](#) - [Plugins](#) - [Web API](#)