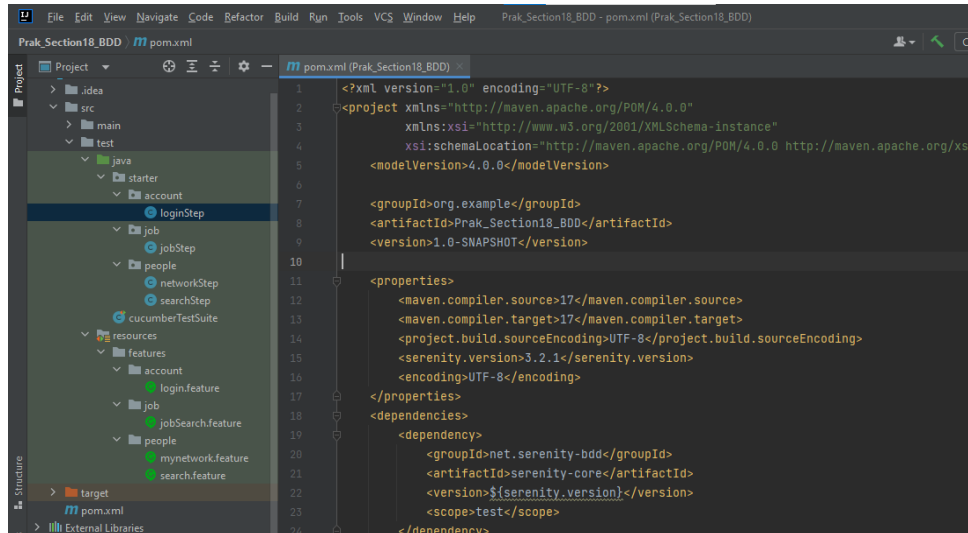


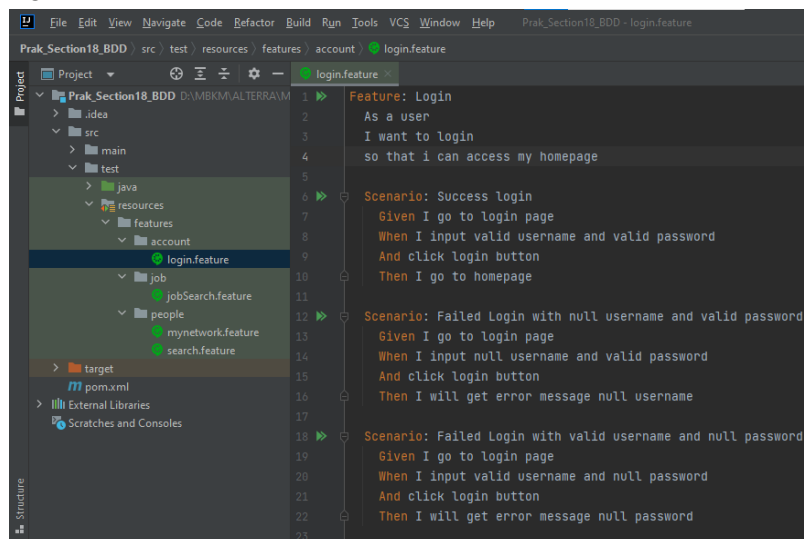
Nama : Sarah Oktavia br Pasaribu  
Kelas : QE-A

## 1. POM.xml



2. Dalam BDD language memakai syntax gherkin, yang ditandai dengan Given, When, Then.

a. login.feature



```

24 >> Scenario: Failed Login with valid username and invalid password
25     Given I go to login page
26     When I input valid username and invalid password
27     And click login button
28     Then I will get error message invalid password
29
30 >> Scenario: Failed Login with invalid username and valid password
31     Given I go to login page
32     When I input invalid username and valid password
33     And click login button
34     Then I will get error message invalid username
35
36 >> Scenario: Failed Login with invalid username and invalid password
37     Given I go to login page
38     When I input invalid username and invalid password
39     And click login button
40     Then I will get error message invalid username and invalid password

```

## b. jobSearch.feature

```

Prak_Section18_BDD  src  test  resources  features  job  jobSearch.feature
Project
  Prak_Section18_BDD
    src
      main
      test
        java
          resources
            features
              account
              login.feature
              job
                jobSearch.feature
              people
                mynetwork.feature
                search.feature
          target
            pom.xml
          External Libraries
          Scratches and Consoles
1 >> Feature: Job Search
2   As a user
3   I want to find a job
4   so that i can get the detail of job
5
6 >> Scenario: Success search job by title
7   Given I go to button job search
8   When I input job title on button search job title
9   And Click the job on button search
10  Then i can see list of job based on title that i input
11
12 >> Scenario: Success search job by city
13   Given I go to button job search
14   When I input the city name on button search job by city
15   And Click the city on button search
16   Then i can see list of job based on city that i input
17
18 >> Scenario: Success search job by city and title
19   Given I go to button job search
20   When I input job title and job city of job
21   And tab enter
22   Then I can see the list of job based on job city and job title that i want
23
24 >> Scenario: Success search job by history
25   Given I go to button job search
26   When I click the job which in history tab
27   Then I can see the list of job based on the history that i choose

```

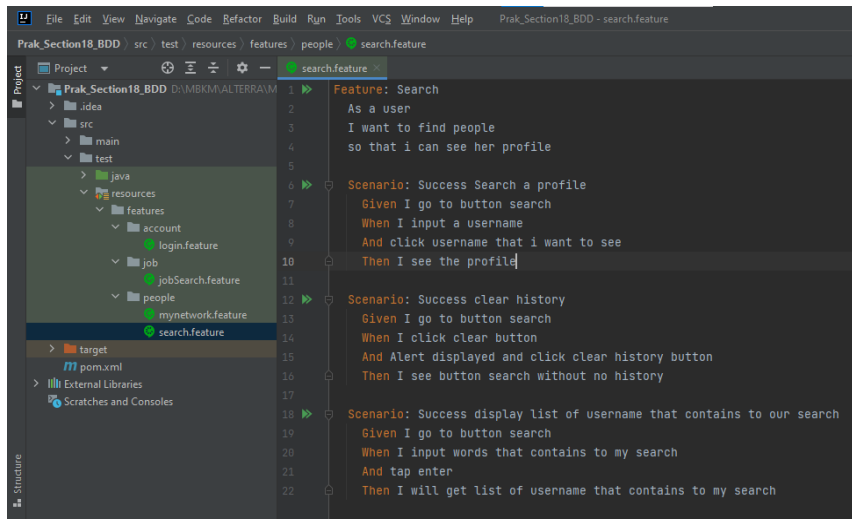
## c. mynetwork.feature

```

Prak_Section18_BDD  src  test  resources  features  people  mynetwork.feature
Project
  Prak_Section18_BDD
    src
      main
      test
        java
          resources
            features
              account
              login.feature
              job
                jobSearch.feature
              people
                mynetwork.feature
                search.feature
          target
            pom.xml
          External Libraries
          Scratches and Consoles
1 >> Feature: My network
2   As a user
3   I want to see my network invitation
4   So that i can connect to them
5
6 >> Scenario: see account profile that just connected with me
7   Given I go to button my network
8   When I click accept
9   And I click the account profile that just connected with me
10  Then I see the account profile that just connected with me
11
12 >> Scenario: see account that i ignored
13   Given I go to button my network
14   When I click ignore
15   And I click the account profile that just ignored by me
16   Then i see the account profile that just ignored by me

```

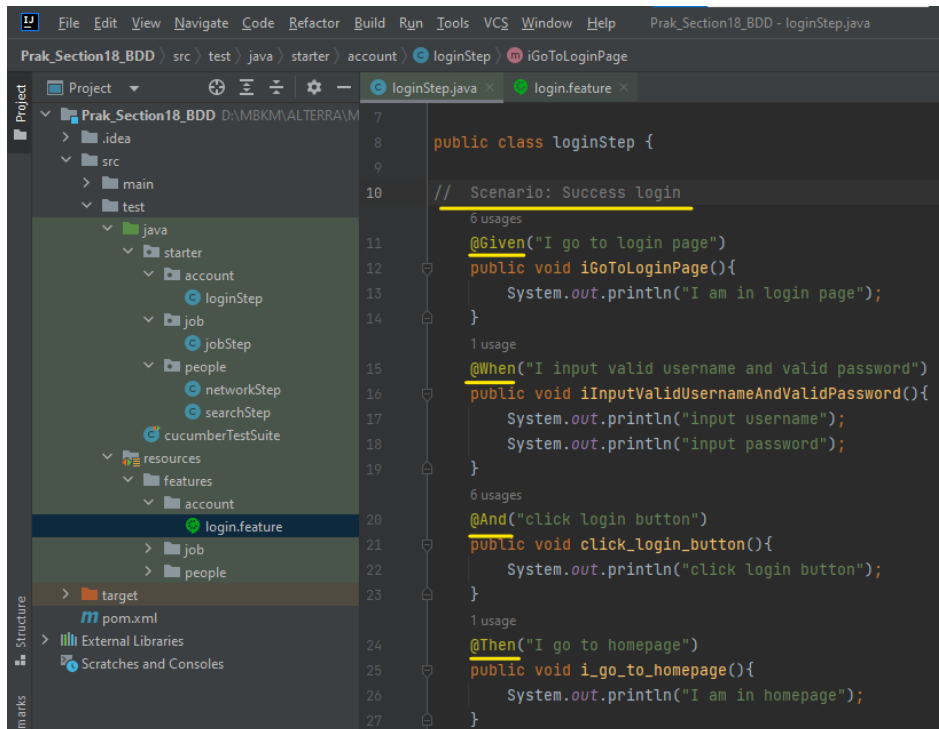
## d. search.feature



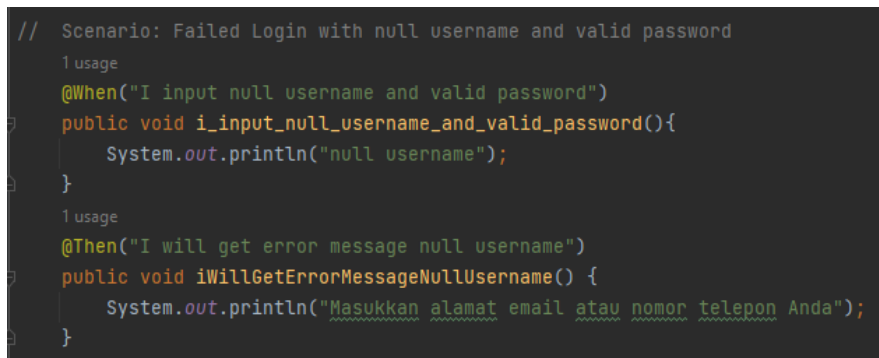
### 3. Step Maps to Java and Configure Stories

#### a. loginStep

scenario 01:



scenario 02:



scenario 03:

```
// Scenario: Failed Login with valid username and null password
1 usage
  @When("I input valid username and null password")
  public void iInputValidUsernameAndNullPassword() {
    System.out.println("null password");
  }
1 usage
  @Then("I will get error message null password")
  public void iWillGetErrorMessageNullPassword() {
    System.out.println("Masukkan password anda");
  }
```

#### scenario 04:

```
// Scenario: Failed Login with valid username and invalid password
1 usage
  @When("I input valid username and invalid password")
  public void iInputValidUsernameAndInvalidPassword() {
    System.out.println("invalid password");
  }
1 usage
  @Then("I will get error message invalid password")
  public void iWillGetErrorMessageInvalidPassword() {
    System.out.println("Kata sandi yang Anda masukkan salah. Coba lagi ");
  }
```

#### scenario 05 :

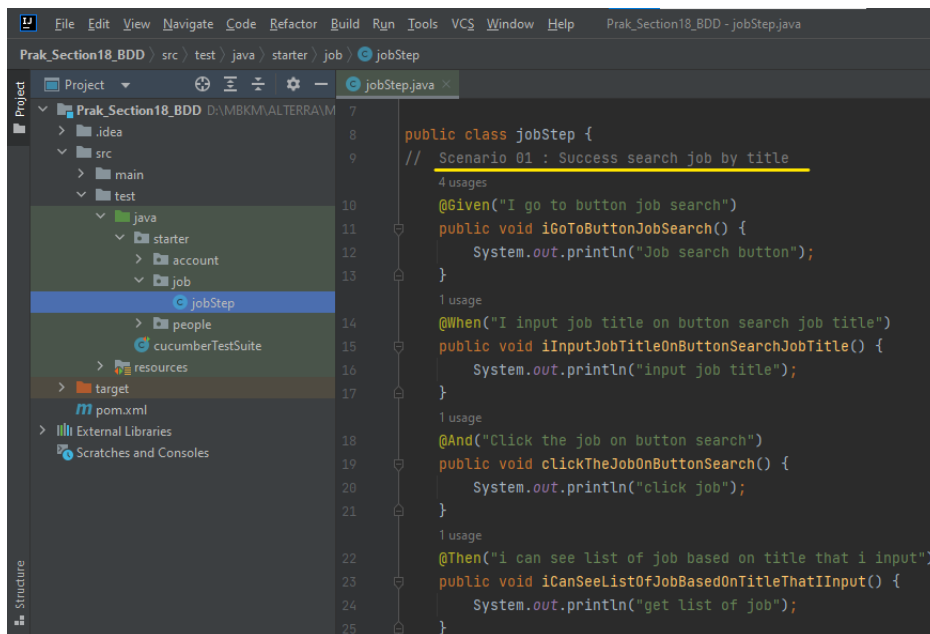
```
// Scenario: Failed Login with invalid username and valid password
1 usage
  @When("I input invalid username and valid password")
  public void iInputInvalidUsernameAndValidPassword() {
    System.out.println("invalid username");
  }
1 usage
  @Then("I will get error message invalid username")
  public void iWillGetErrorMessageInvalidUsername() {
    System.out.println("Tidak dapat menemukan akun LinkedIn yang dikaitkan dengan");
  }
```

#### scenario 06 :

```
// Scenario: Failed Login with invalid username and invalid password
1 usage
  @When("I input invalid username and invalid password")
  public void iInputInvalidUsernameAndInvalidPassword() {
    System.out.println("invalid username and invalid password");
  }
1 usage
  @Then("I will get error message invalid username and invalid password")
  public void iWillGetErrorMessageInvalidUsernameAndInvalidPassword() {
    System.out.println("Tidak dapat menemukan akun LinkedIn yang dikaitkan dengan");
  }
```

### b. jobStep

#### scenario 01:



**scenario 02 :**

```

// Scenario 02: Success search job by city
1 usage
@When("I input the city name on button search job by city")
public void iInputTheCityNameOnButtonSearchJobByCity() {
    System.out.println("input city name");
}
1 usage
@And("Click the city on button search")
public void clickTheCityOnButtonSearch() {
    System.out.println("click city that i input");
}
1 usage
@Then("i can see list of job based on city that i input")
public void iCanSeeListofJobBasedOnCityThatIInput() {
    System.out.println("get list of job that i input");
}

```

**scenario 03 :**

```

// Scenario 03: Success search job by city and title
1 usage
@When("I input job title and job city of job")
public void iInputJobTitleAndJobCityOfJob() {
    System.out.println("input job title and city");
}
1 usage
@And("tab enter")
public void tabEnter() {
    System.out.println("tap enter");
}
1 usage
@Then("I can see the list of job based on job city and job title that i want")
public void iCanSeeTheListofJobBasedOnJobCityAndJobTitleThatIWant() {
    System.out.println("get list job based on the title and city that i want");
}

```

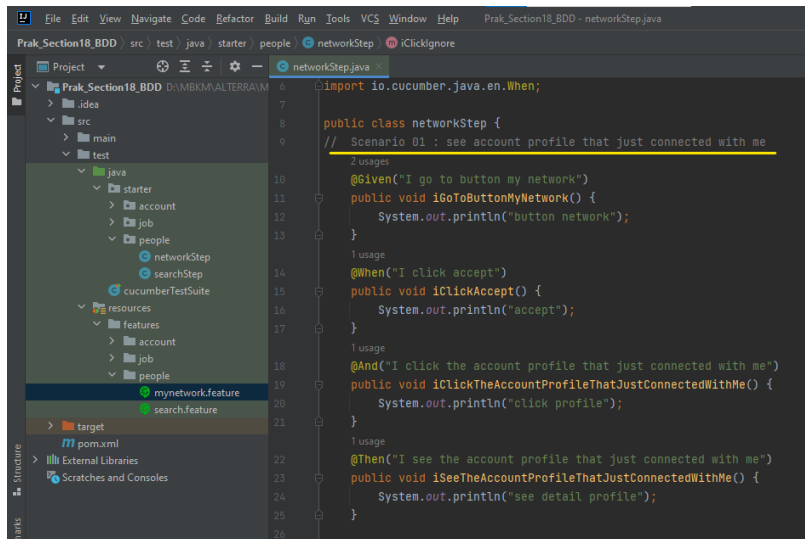
**scenario 04 :**

```

// Scenario 04: Success search job by history
1 usage
@When("I click the job which in history tab")
public void iClickTheJobWhichInHistoryTab() {
    System.out.println("click job in tab history");
}
1 usage
@Then("I can see the list of job based on the history that i choose")
public void iCanSeeTheListofJobBasedOnTheHistoryThatIChoose() {
    System.out.println("get list of job based on the history that i choose");
}

```

c. networkStep  
scenario 01 :

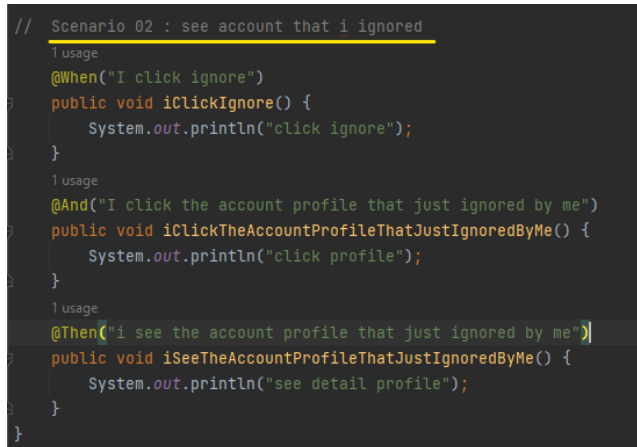


The screenshot shows an IDE window titled 'Prak\_Section18\_BDD - networkStep.java'. The left sidebar displays a project structure for 'Prak\_Section18\_BDD' with folders like 'src', 'main', 'test', 'resources', and 'target'. The 'test' folder is expanded, showing 'java' and 'resources' subfolders. The 'java' folder contains 'starter', 'account', 'job', 'people', 'networkStep', and 'searchStep'. The 'resources' folder contains 'features', 'account', 'job', and 'people'. The 'networkStep' folder is selected, showing 'myNetwork.feature' and 'search.feature'. The main editor displays the 'networkStep.java' file with the following code:

```
import io.cucumber.java.en.When;

public class networkStep {
    // Scenario 01 : see account profile that just connected with me
    2 usages
    @Given("I go to button my network")
    public void iGoToButtonMyNetwork() {
        System.out.println("button network");
    }
    1 usage
    @When("I click accept")
    public void iClickAccept() {
        System.out.println("accept");
    }
    1 usage
    @And("I click the account profile that just connected with me")
    public void iClickTheAccountProfileThatJustConnectedWithMe() {
        System.out.println("click profile");
    }
    1 usage
    @Then("I see the account profile that just connected with me")
    public void iSeeTheAccountProfileThatJustConnectedWithMe() {
        System.out.println("see detail profile");
    }
}
```

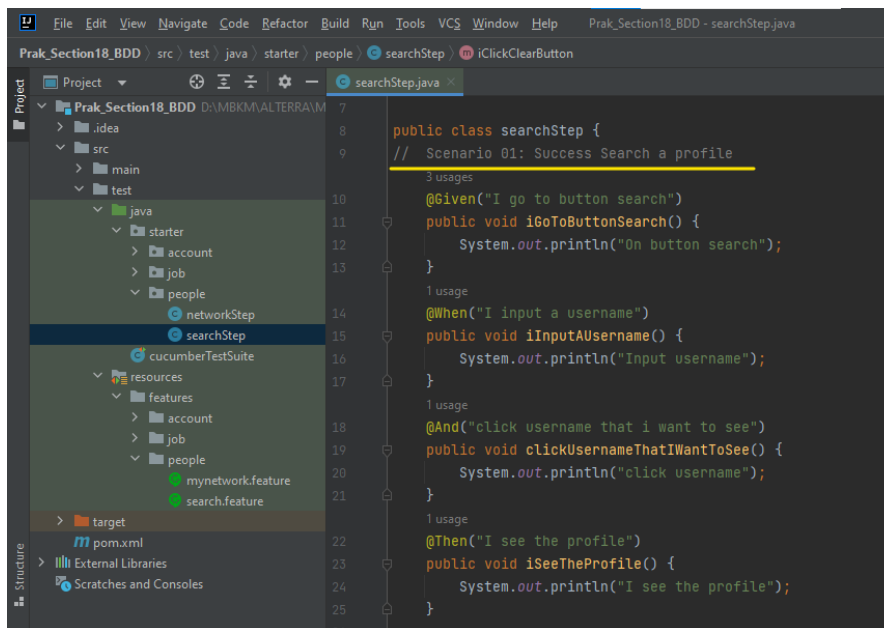
scenario 02 :



The screenshot shows a code editor with the following code for scenario 02:

```
// Scenario 02 : see account that i ignored
1 usage
@When("I click ignore")
public void iClickIgnore() {
    System.out.println("click ignore");
}
1 usage
@And("I click the account profile that just ignored by me")
public void iClickTheAccountProfileThatJustIgnoredByMe() {
    System.out.println("click profile");
}
1 usage
@Then("i see the account profile that just ignored by me")
public void iSeeTheAccountProfileThatJustIgnoredByMe() {
    System.out.println("see detail profile");
}
}
```

d. searchStep  
scenario 01 :



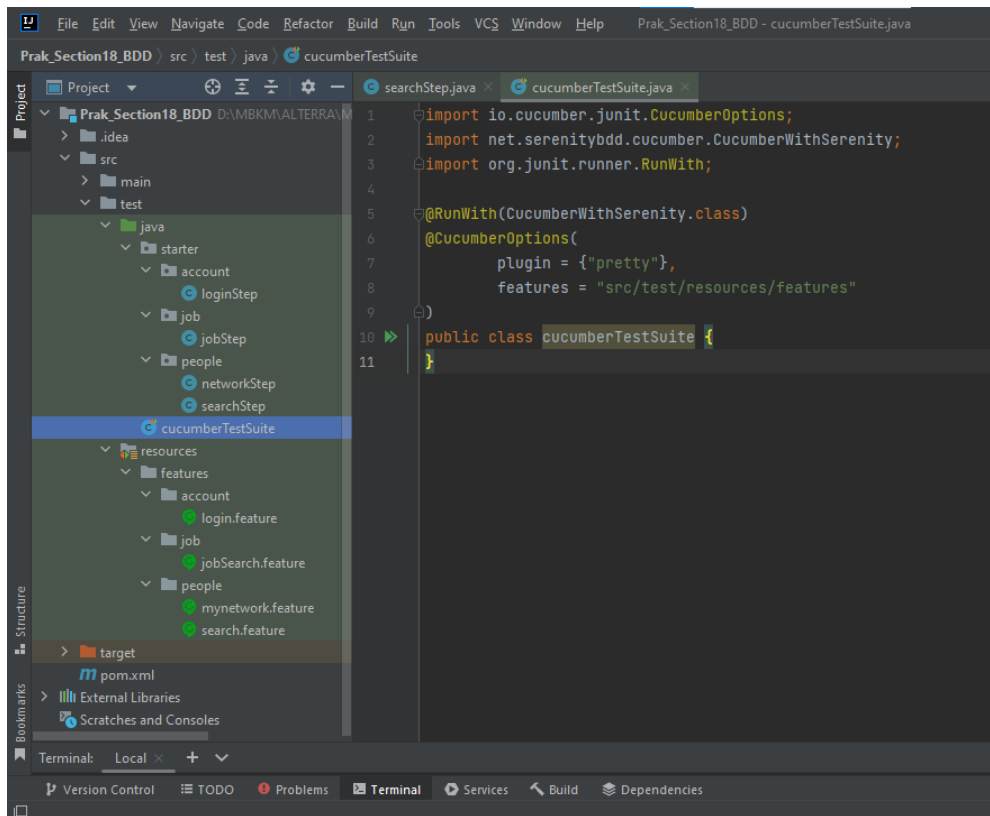
scenario 02 :

```
// Scenario 02: Success clear history
1 usage
@When("I click clear button")
public void iClickClearButton() {
    System.out.println("click clear");
}
1 usage
@And("Alert displayed and click clear history button")
public void alertDisplayedAndClickClearHistoryButton() {
    System.out.println("click clear history button");
}
1 usage
@Then("I see button search without no history")
public void iSeeButtonSearchWithoutNoHistory() {
    System.out.println("get a button without no history");
}
```

scenario 03 :

```
// Scenario 03: Success display list of username that contains to our search
1 usage
@When("I input words that contains to my search")
public void iInputWordsThatContainsToMySearch() {
    System.out.println("input words");
}
1 usage
@And("tap enter")
public void tapEnter() {
    System.out.println("tab enter");
}
1 usage
@Then("I will get list of username that contains to my search")
public void iWillGetListOfUsernameThatContainsToMySearch() {
    System.out.println("get list of username that contains to search");
}
}
```

→ Berikut program cucumberTestSuite.java :



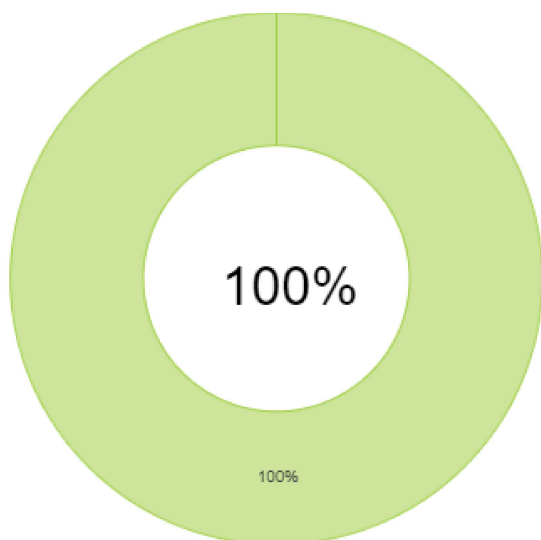


**Hasil Report :**

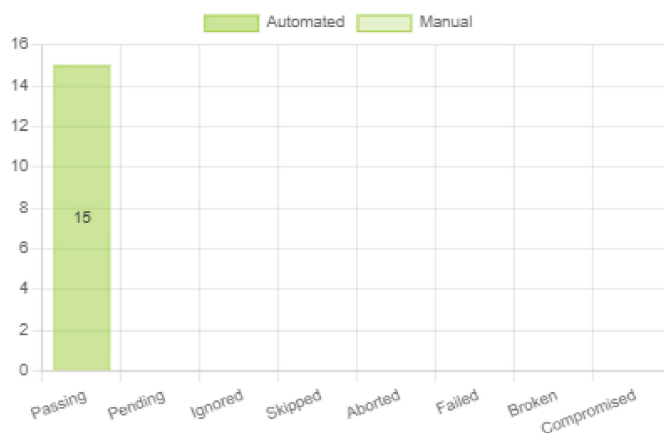
☒ Overall Test Results☐ Requirements (capabilities.html)☐ Capabilities (7ea0a5e2d679b9deeda9910ae4a9bcd3a281e0d908303a9e16e486103e98880c.html)☐ Features (576946480b52ad056d6f5bddf874399c83582ecf90963cc074a14c70580e7d9f.html)☐ (build-info.html) Report generated 08-10-2022 20:13:29

## Test Results: All Tests

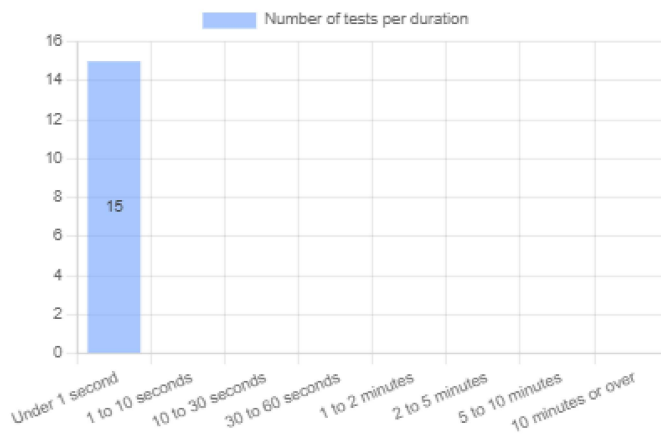
### 15 test cases

☒ Summary☐ Test Results☒ Overview

### ☒ Test Outcomes

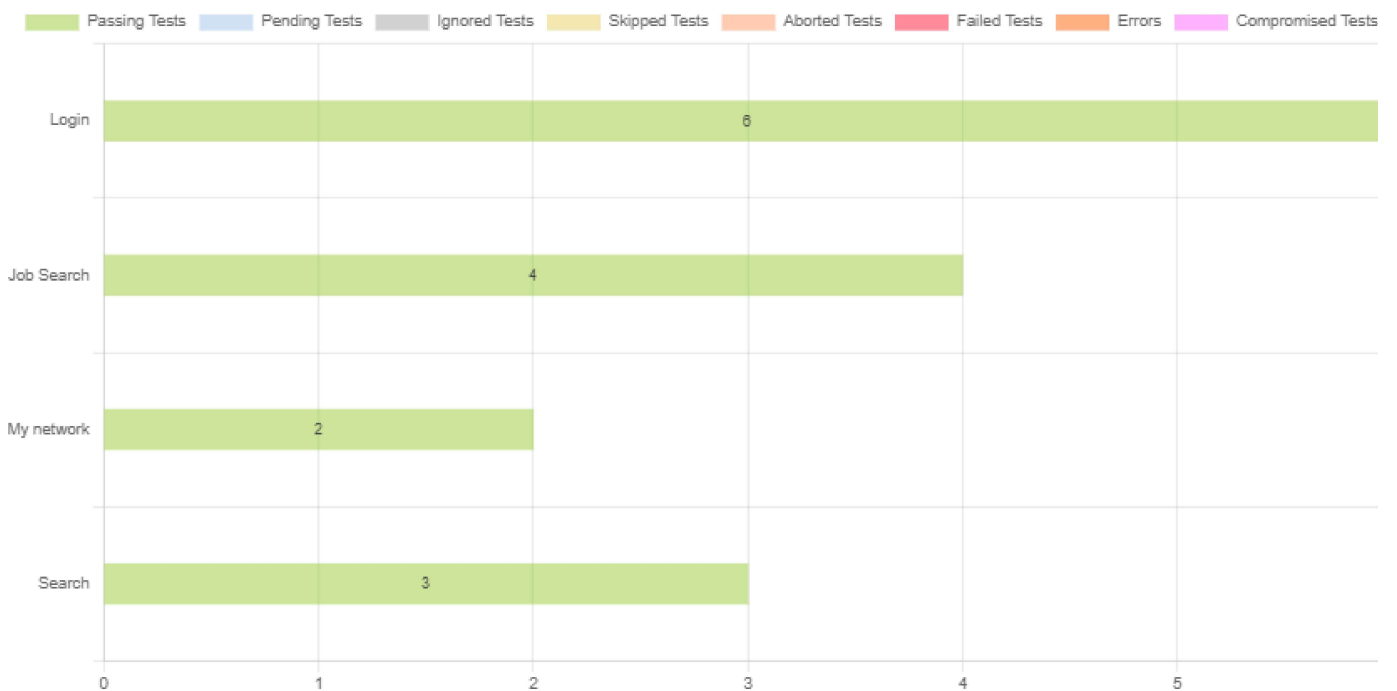


### ☐ Test Performance



## Functional Coverage Overview

### Features



## Key Statistics

Number of Test Cases	15
Number of Scenarios	15
Tests Started	Oct 08, 2022 20:13:21
Tests Finished	Oct 08, 2022 20:13:22
Total Duration	735ms
Fastest Test	16ms
Slowest Test	222ms
Average Execution Time	33ms
Total Execution Time	499ms

## Functional Coverage Details

### Features

Feature	Test Cases	Scenarios	% Pass	Result	Co
---------	------------	-----------	--------	--------	----

Feature	↑↓	Test Cases	↑↓	Scenarios	↑↓	% Pass	↑↓	Result	↑↓	Co
Job Search (2bfddb4e0d6cadce2b44ef5cc67b9fc517d853970764e9e93293a4d378779336.html)		4		4		100%		✓		
Login (a263c1cfd124a91a82912b54ed25885880cd0632629530bae3ab4498ea33fda9.html)		6		6		100%		✓		
My network (53d5435aaada2f25a7cabbccceff941dbef815199ea08e1d9c67bfe26133ff6e6.html)		2		2		100%		✓		
Search (bf1fd9b92b3dd4d6718b6ba986660939ccf14300f2d305171dd3ccc5a7814bce.html)		3		3		100%		✓		

☒ Overall Test Results (index.html)

☒ Requirements

☒ Capabilities (7ea0a5e2d679b9deeda9910ae4a9bcd3a281e0d908303a9e16e486103e98880c.html)

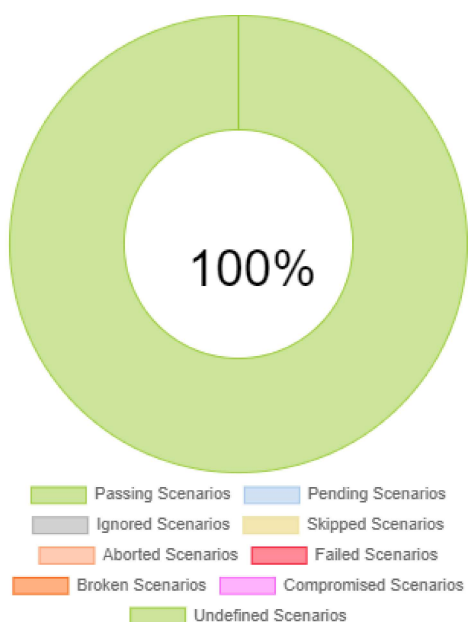
☒ Features (576946480b52ad056d6f5bddf874399c83582ecf90963cc074a14c70580e7d9f.html)

(i) (build-info.html) Report generated 08-10-2022 20:13

☒ Specifications

☒ Test Results

## Feature Coverage By Scenario











## Requirements Overview

<input checked="" type="checkbox"/> Account (183df4364d7df2c67c797dcf08c6014d4d24da4d785b3ffccb836434ef05991.html)	1 feature	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Login (a263c1cfd124a91a82912b54ed25885880cd0632629530bae3ab4498ea33fda9.html)	6 scenarios	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Job (879e6010b55a5f4d1a3b4958dd5ba4858068e545585e743197f4d4403465db58.html)	1 feature	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Job Search (2bfddb4e0d6cadce2b44ef5cc67b9fc517d853970764e9e93293a4d378779336.html)	4 scenarios	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> People (523788e908d3e8edb9b90c3b6769cd8a2314986650ea6d6c1ef3dd5fde6c993.html)	2 features	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> My network (53d5435aaada2f25a7cabbceff941dbef815199ea08e1d9c67bfe26133ff6e6.html)	2 scenarios	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Search (bf1fd9b92b3dd4d6718b6ba986660939ccf14300f2d305171dd3ccc5a7814bce.html)	3 scenarios	<input checked="" type="checkbox"/>

## Test Outcomes for Capabilities

Show  entries

Capability		Test Cases		Scenarios		% Pass		Result		Cove
Account (183df4364d7df2c67c797dcf08c6014d4d24da4d785b3fffccb836434ef05991.html)		6		6		100%				100%
Job (879e6010b55a5f4d1a3b4958dd5ba4858068e545585e743197f4d4403465db58.html)		4		4		100%				100%
People (523788e908d3e8edbf9b90c3b6769cd8a2314986650ea6d6c1ef3dd5fde6c993.html)		5		5		100%				100%

Showing 1 to 3 of 3 entries

Previous

1

Next

## Test Outcomes for Features

Show  entries

Feature	<div>↑↓</div> Test Cases	<div>↑↓</div> Scenarios	<div>↑↓</div> % Pass	<div>↑↓</div> Result	<div>↑↓</div> Cove
Job Search (2bfddb4e0d6cadce2b44ef5cc67b9fc517d853970764e9e93293a4d378779336.html)	4	4	100%	✓	100%
Login (a263c1cfd124a91a82912b54ed25885880cd0632629530bae3ab4498ea33fda9.html)	6	6	100%	✓	100%
My network (53d5435aaada2f25a7cabbccff941dbef815199ea08e1d9c67bfe26133ff6e6.html)	2	2	100%	✓	100%
Search (bf1fd9b92b3dd4d6718b6ba986660939ccf14300f2d305171dd3ccc5a7814bce.html)	3	3	100%	✓	100%

Showing 1 to 4 of 4 entries

[Previous](#) [1](#) [Next](#)