1. **QUESTION**

[PAINLESS]

Proof of concept to demonstrate screen level field validations using regular expression. Expectation is to demonstrate, number, text, PAN, Passport currency, age, restricting special characters, restricting character range, restricting number range, restricting date range etc. solution shall be not limiting the mentioned sample validations.

**Source Code GIT Repository:**

**Web Browser Link:**

[**https://github.com/Ajay-Kumar-Aspiring-Minds-Round-2/userapp**](https://github.com/Ajay-Kumar-Aspiring-Minds-Round-2/userapp)

**GIT Clone Link:**

[**https://github.com/Ajay-Kumar-Aspiring-Minds-Round-2/userapp.git**](https://github.com/Ajay-Kumar-Aspiring-Minds-Round-2/userapp.git)

**Technology Stack:**

****   ****

**Key Features:**

* **Angular UI** for entering user details.
* **Angular** **Reactive Forms** Validation.
* **Required Field Validation, Pattern Validation & Min-Max Range Validations.**
* **Istanbul Code Coverage Report** for Angular Components.
* Use of predefined **Angular Modules** (Ex: **BrowserModule, FormsModule, ReactiveFormsModule**, etc.,)
* Use of **PrimeNG – Angular UI Component Library**.
* Managing **Angular** **Routing**.

**User Details Form:**

* **FormBuilder, FormGroup** & **FormControl** controls are used for **Reactive Forms** design of User Details Form User Interface.
* **Required, Pattern, Min & Max Range Validators** are used for validations.
* **PrimeNG Angular UI Component Library** form elements are used for UI design.
* **Angular Routing** is managed in AppRoutingModule for the routes.
* **Karma configuration:**
  + **Karma** is a testing automation tool created by the Angular team at Google.
  + **Karma** is a task runner for our tests. It uses a configuration file in order to set the startup file, the reporters, the testing framework, the browser among other things.
* **Jasmine Test Framework:**
  + **Jasmine** is a test framework we are going to use to create or write our tests.
  + **Jasmine** has a bunch of functionalities to allow us, to write different kinds of tests.
* **Istanbul Code Coverage Report:**
  + **Istanbul** is a test coverage tool that works with many different frameworks.
  + It tracks which parts of your code are executed by your unit tests. Thus, you can use Istanbul to view and see coverage gaps.

**Field Validation Conditions for FULL NAME:**

The valid **FULL NAME** must satisfy the following conditions:

1. It should allow lower case & upper case alphabets.
2. Space is a allowed character.

**Field Validation Conditions for GENDER:**

The valid **GENDER** must satisfy the following conditions:

1. Options to select are Male or Female.

**Field Validation Conditions for AGE:**

The valid **AGE** must satisfy the following conditions:

1. It should be any number from 1 to 99.
2. Only numbers are allowed.

**Field Validation Conditions for PAN NUMBER:**

The valid **PAN NUMBER** must satisfy the following conditions:

1. It should be ten characters long.
2. The first five characters should be any upper case alphabets.
3. The next four-characters should be any number from 0 to 9.
4. The last(tenth) character should be any upper case alphabet.
5. It should not contain any white spaces.

**Field Validation Conditions for PASSPORT NUMBER:**

The valid **PASSPORT NUMBER** must satisfy the following conditions:

1. It should be eight characters long.
2. The first character should be an upper case alphabet.
3. The next two characters should be a number, but the first character should be any number from 1-9 and the second character should be any number from 0-9.
4. It should be zero or one white space character.
5. The next four characters should be any number from 0-9.
6. The last character should be any number from 1-9.

**Field Validation Conditions for CURRENCY:**

The valid **CURRENCY** must satisfy the following conditions:

1. Only one decimal point is allowed.
2. Only two digits after a decimal point is a valid format (Ex: 123.50)
3. Only one digit after a decimal point is invalid (Ex: 3.5 is invalid)

**Field Validations:**

* **Full Name field - textfield**
  + Required field validation.
  + Pattern Validation - **"^[a-zA-z ]\*$" – Alphabets (a-z & A-Z along with space are allowed)**
* **Gender Field – radio button**
  + Required field validation (**Male or Female options** to select from).
* **Age Field – textfield**
  + Required field validation.
  + Pattern Validation - **"^[0-9]\*$" – Numbers only are allowed.**
  + Min & Max Range validation **– Allowed within range of 1 to 99.**
* **Pan Number – textfield**
  + Required field validation.
  + Pattern Validation - **"[A-Z]{5}[0-9]{4}[A-Z]{1}"**

**Only pattern matched values are allowed, where:**

* + - [A-Z]{5} represents the first five upper case alphabets which can be A to Z.
    - [0-9]{4} represents the four numbers which can be 0-9.
    - [A-Z]{1} represents the one upper case alphabet which can be A to Z.
* **Passport Number – textfield**
  + Required field validation.
  + Pattern Validation - **"^[A-PR-WYa-pr-wy][1-9]**[**\\d\\s?\\d{4}[1-9]$**](file:///\\d\\s%3f\\d%7b4%7d%5b1-9%5d$)**"**

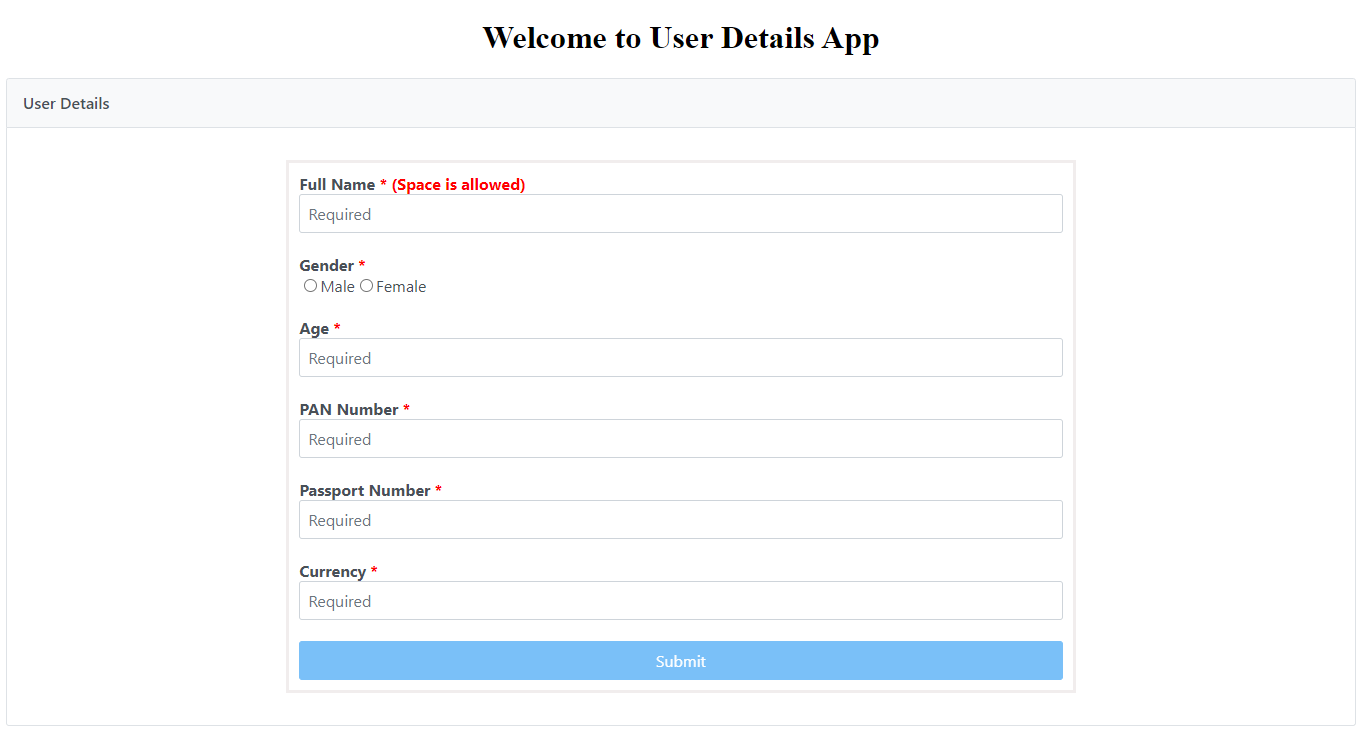
**Only pattern matched values are allowed, where:**

* ^ represents the starting of the string.
* [A-PR-WYa-pr-wy] represents the string should start with A-Z excluding Q, X, and Z.
* [1-9] represents the second character should be any number from 1-9.
* \\d represents the third character should be any number from 0-9.
* \\s? represents the string should be zero or one white space character.
* \\d{4} represents the next four characters should be any number from 0-9.
* [1-9] represents the last character should be any number from 1-9.
* $ represents the ending of the string.
* **Currency – textfield**
  + Required field validation.
  + Pattern Validation – **"^[0-9]+(\.[0-9]{2})?$"**

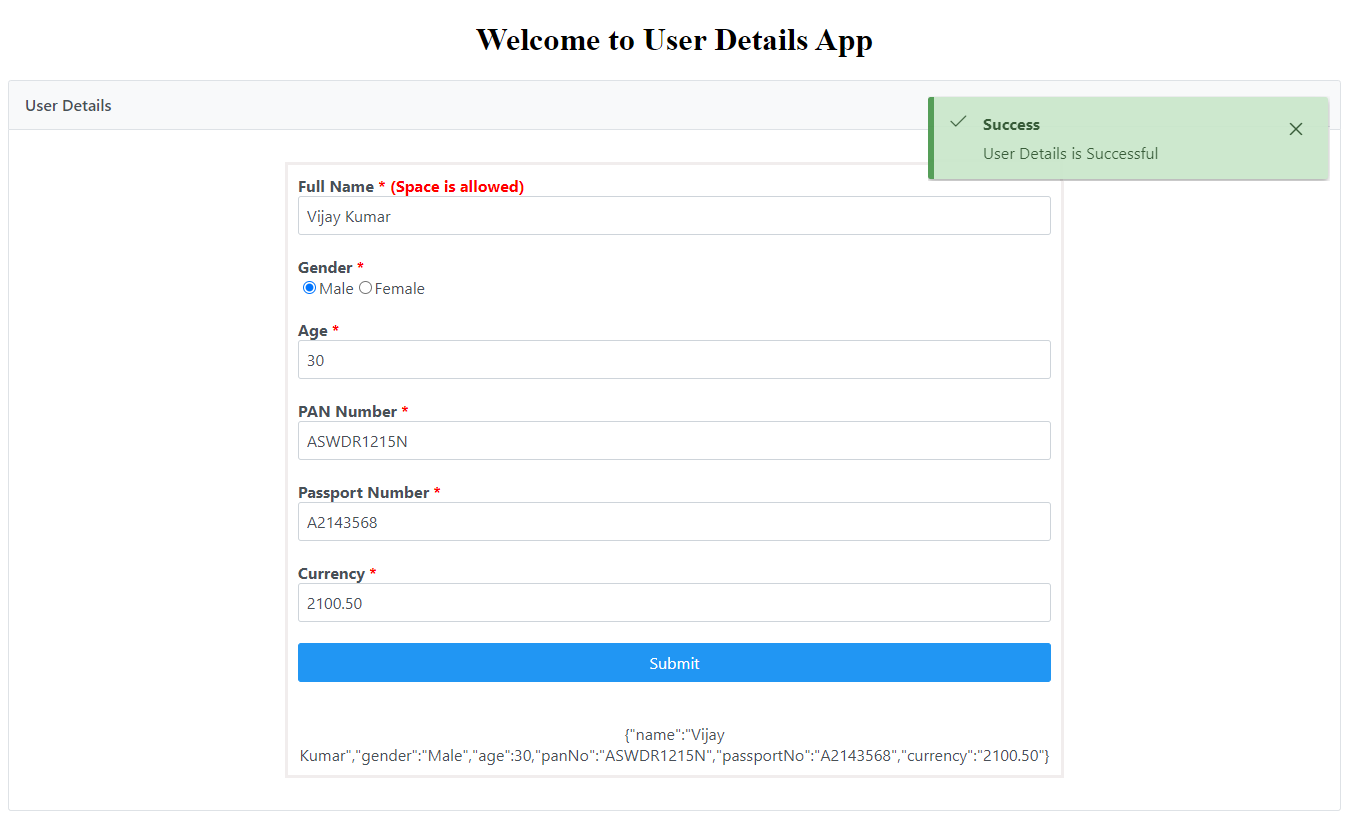
**Only pattern matched values are allowed, where:**

* + - ^ represents the starting of the string.
    - [0-9]+ represents any number of numbers from 0-9.
    - \. represents a allowed decimal point value.
    - [0-9]{2} represents the two numbers which can be 0-9.
    - (\.[0-9]{2})? represents decimal point and two numbers as optional.

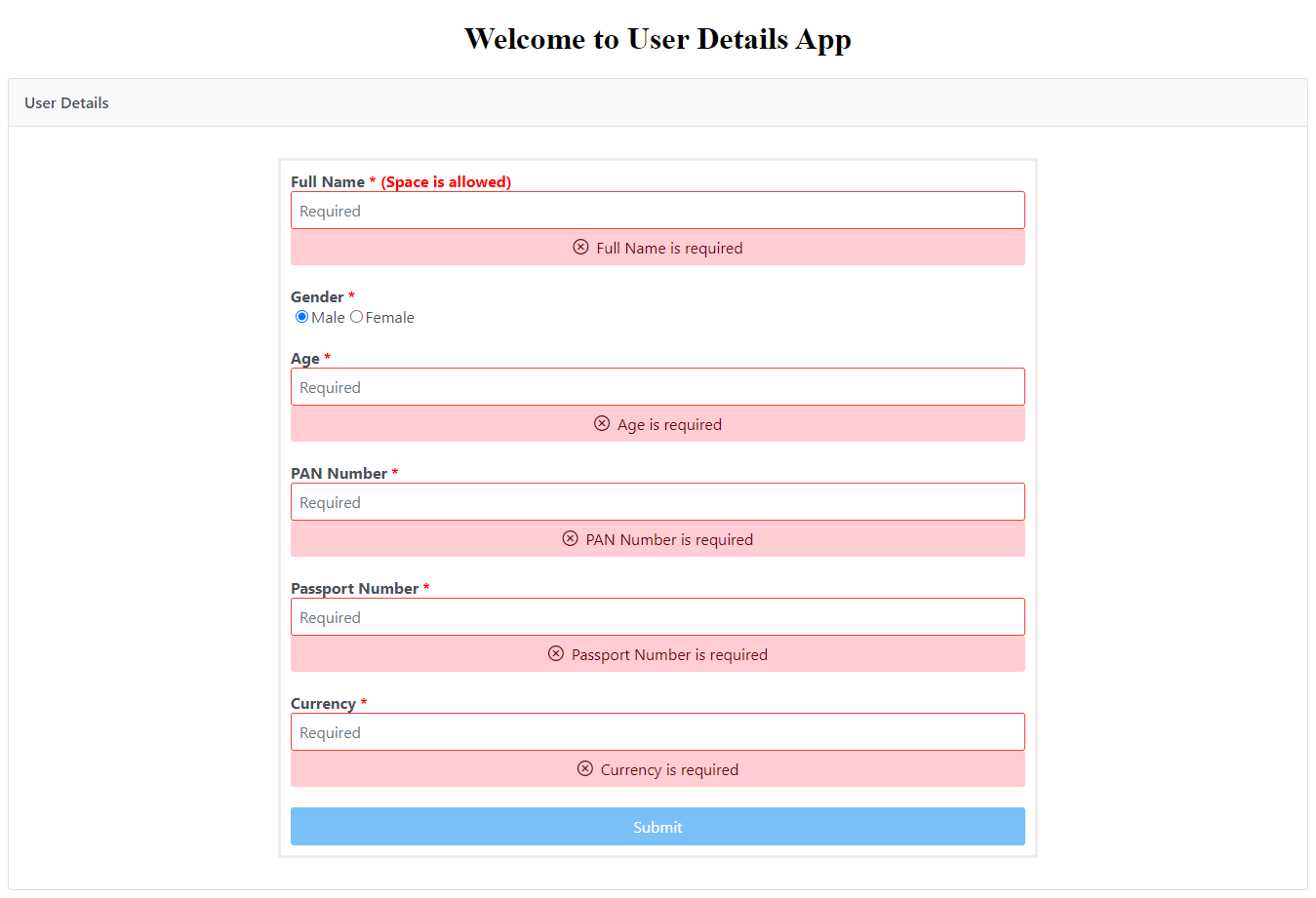
**User Details Form:**

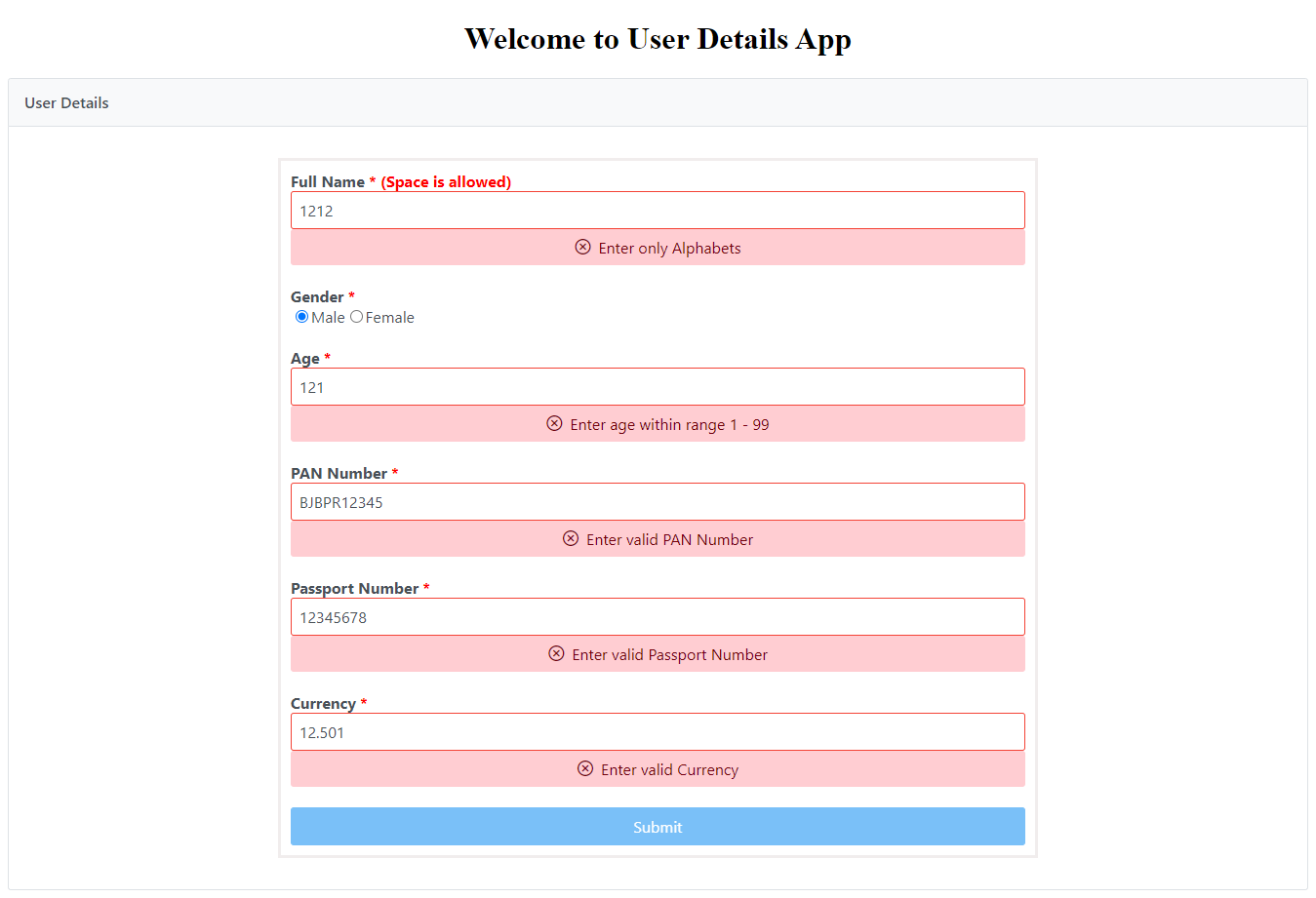


**User Details Form with Successful field validations:**



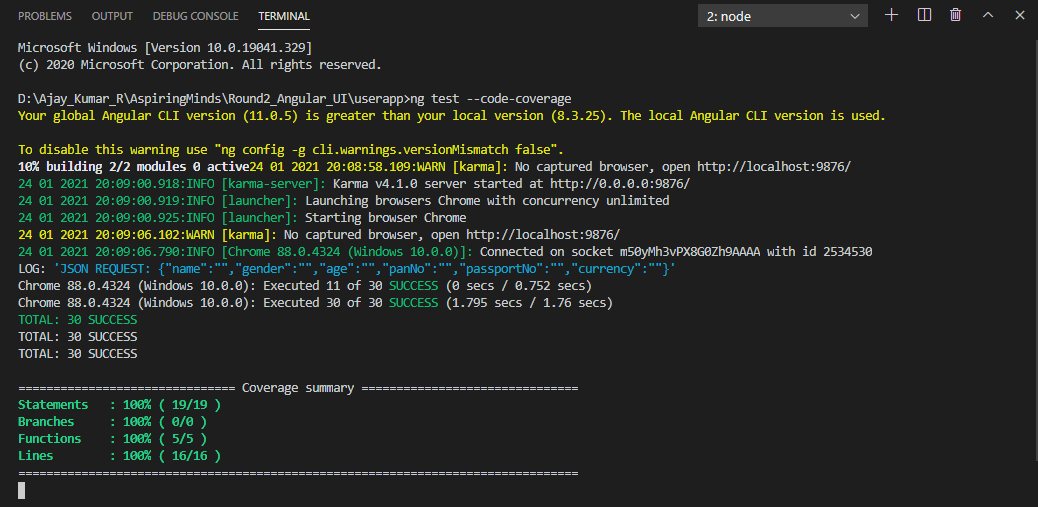
**User Details Form with Error field validations:**

****

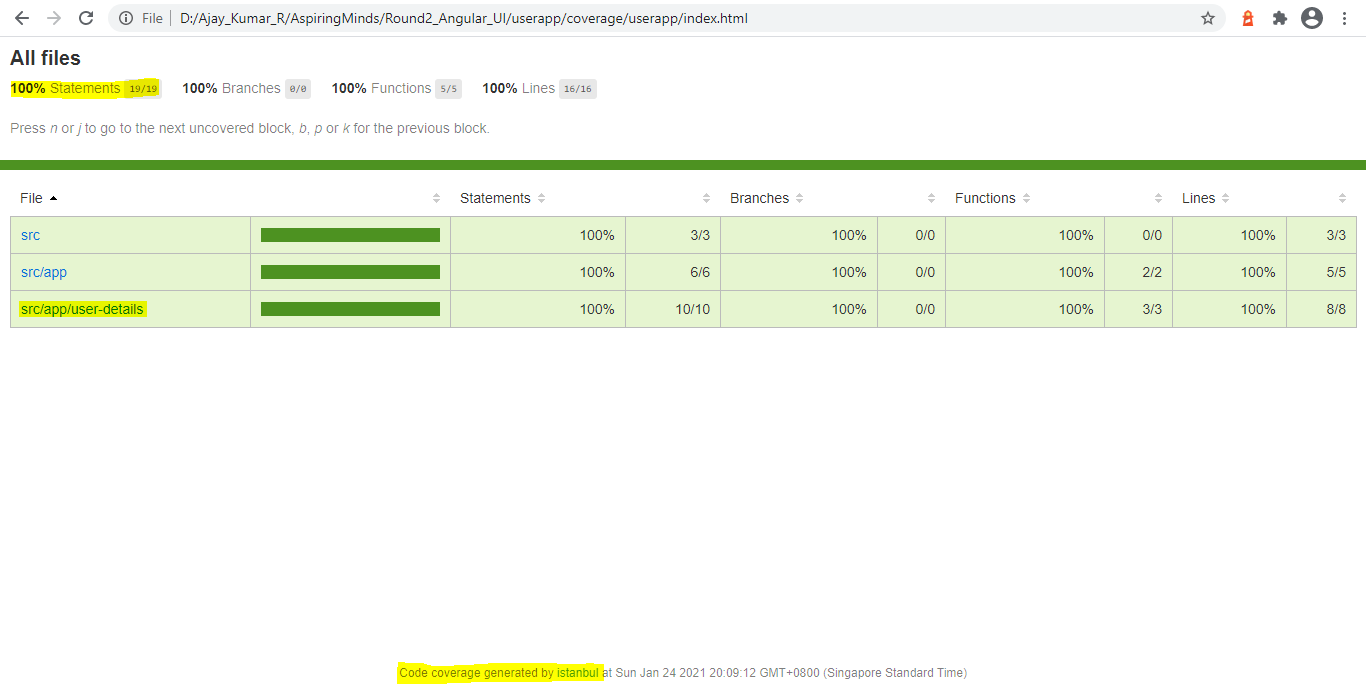


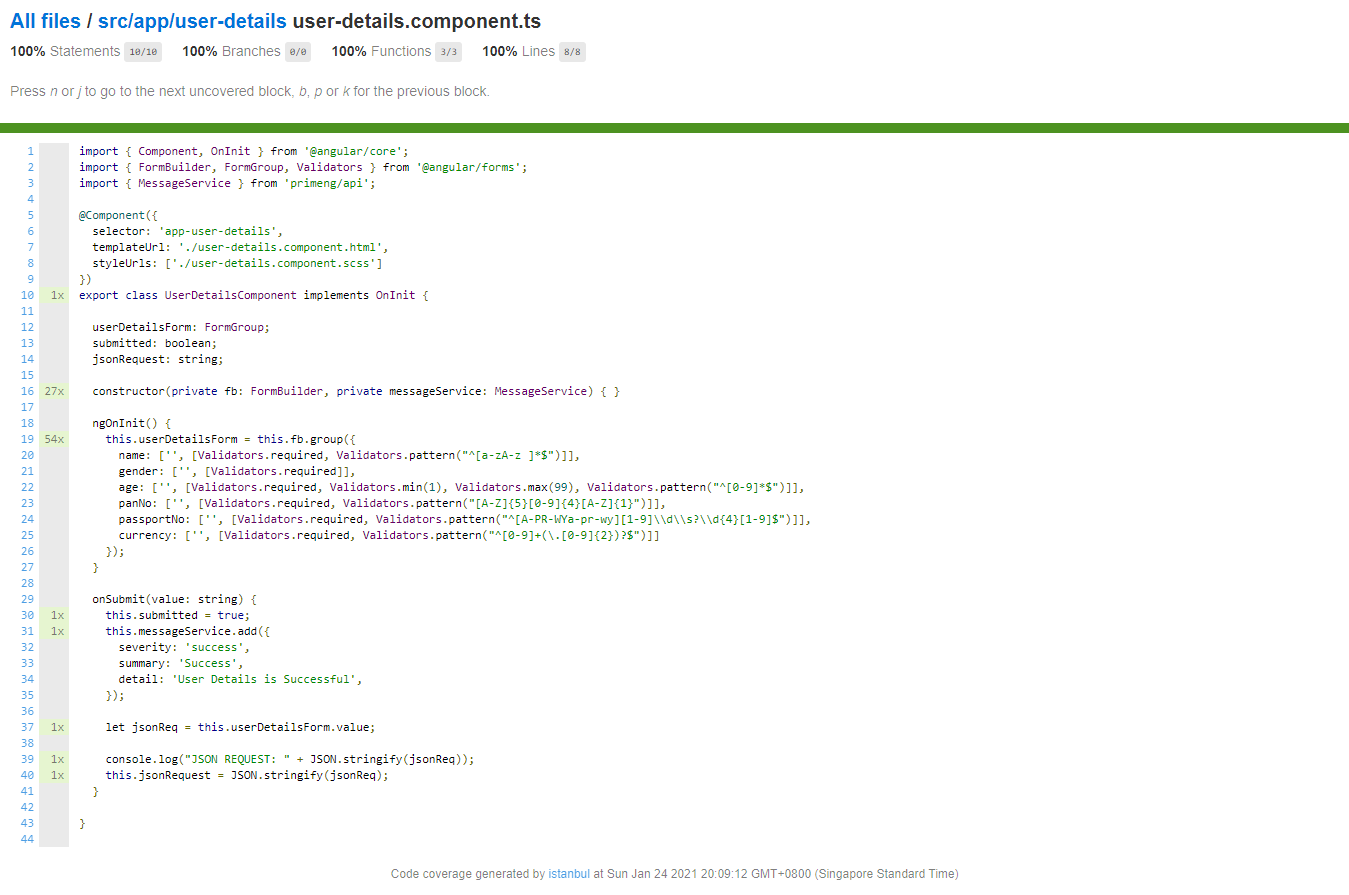
**Istanbul Code Coverage Report for User Details Component:**

* **Unit Test Cases executed : 30 Unit Test Cases.**
* **Passed Unit Tests : 30 Unit Test Cases.**
* **Failed Unit Tests : 0 Unit Test Cases.**

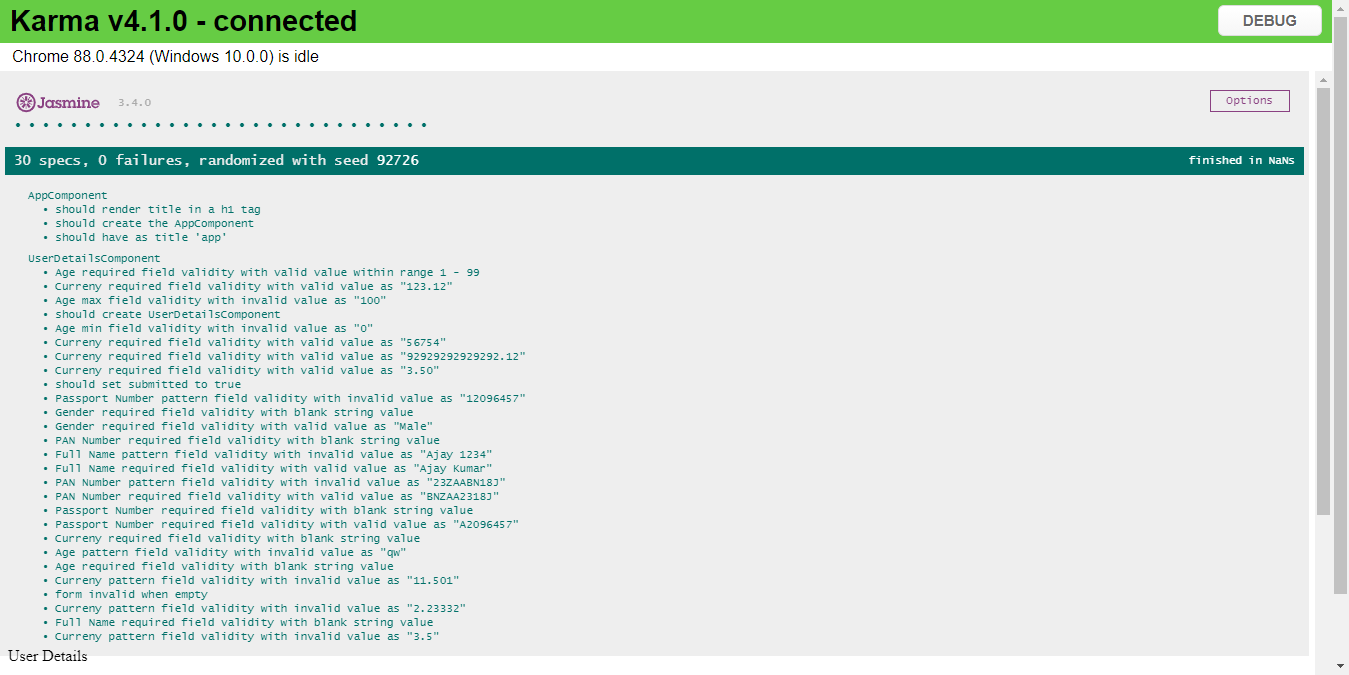


**Istanbul Code Coverage Report:**





**Karma Unit Test Cases Report:**



**Router Tree Structure of Components:**

