**ResumeBuilder**

A resume builder is an online tool that you can use to quickly create a resume. With the help of a resume builder, you may select a pre-formatted resume template, enter your contact and employment details, and instantly have a pdf version of your resume at your disposal.

User can perform below operations,

1. After the registration/login, user will be prompted with a form to fill his career details which will be used to build the Resume.
2. The user will be taken to the profile page after submitting the form and filling out all the required fields so that he may review the information he supplied.
3. In the view profile page, the user can create a resume by choosing any of the templates below.
4. User can download the resume in the form of PDF.
5. User can edit the content of resume and re-download it.

**Instructions:**

* *A Project with the desired folder structure has been already created*. You are only permitted to alter the logical layer; creating files or folders is not permitted.
* The application includes a **database** layer and the required folders that contains the necessary code. You only need to complete the given objectives by writing your logic in the specified controller method.
* You are not allowed to directly insert new data into the database.
* Do not change any schema variables
* The application uses the APIs mentioned below. where some of the controllers have definitions and others have blank definitions, requiring you to develop logic to fulfil the objectives: -
* ***API Name:*** */api/login*

***Method:*** *POST*

***Note:*** *This API is used for Login to application.*

* ***API Name:*** */api/register*

***Method:*** *POST*

***Note:*** *This API is used for new user registration.*

* ***API Name:*** */api /resume/get*

***Method:*** *GET*

***Note:*** *this API is used to get the nutritional details of the food.*

* ***API Name:*** */api/resume/save*

***Method:*** *POST*

***Note:*** *this API can be used to post user details for resume.*

* ***API Name:*** */api/resume/update*

***Method:*** *PUT*

***Note:*** *this API can be used to Update user resume details.*

**Backend** **Objective**:

The application uses the APIs mentioned below. where some of the controllers and services have definitions and others have blank definitions, requiring you to develop logic to fulfil the objectives.

1. We have defined the below methods in the ***Controller*** Layer. You must complete the body of it.

@PutMapping("/update")

**public** ResponseEntity<ResponseMaster> updateResume(@RequestBody ResumeDetailsDTO resumeDetailsDTO) {

@GetMapping("/get")

**public** ResponseEntity<ResponseMaster> getResume(@RequestParam Integer userId) {

1. We have defined below methods in ***Service*** Layer. You must complete the body of them.

**public** ResumeDetailsDTO updateResume(ResumeDetailsDTO resumeDetailsDTO) {

**public** List<ResumeDetailsDTO> get(Integer userId) {

**Note**: -

You are allowed to make changes only in below files.

1. ResumeDetailsController.java

2. ResumeDetailsServiceImple.java

You are not allowed to make any changes in below files.

1. application.properties
2. pom.xml

You are not allowed to make any changes in the files present in below packages.

1. com.example.resumebuider.constant

2. com.example.resumebuider.dto

3. com.example.resumebuider.entity

4. com.example.resumebuider.exception

5. com.example.resumebuider.handle

### 6. com.example.resumebuider.repository

### Frontend Objective:

Instructions:

1. Design a FormArray in projectsForm with controls **“projectDomain”,” projectName” and” projectDescription”**.
2. Prepare Payload and integrate “resume/save” API on submit.

* You are allowed to make any changes to the below files.

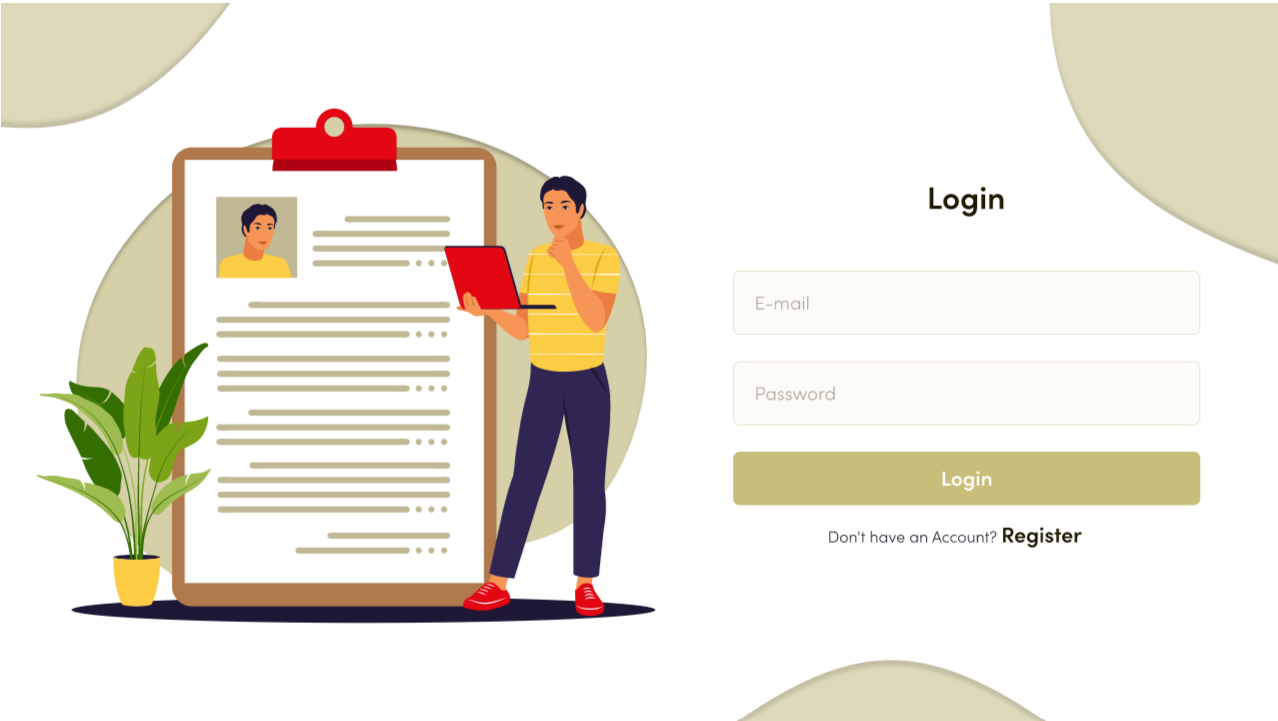
1. collect-user-information.component.html
2. collect-user-information.component.ts

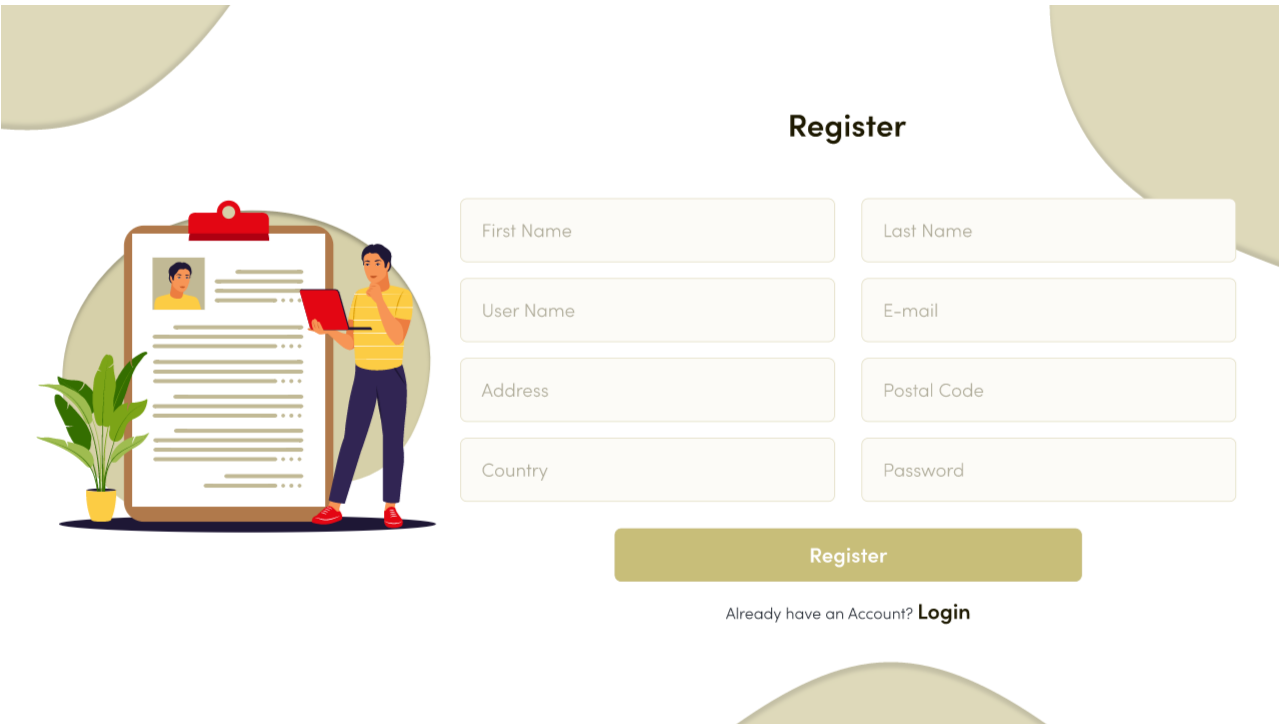
# Build, Run and Deploy

* + Load the project in VS Code.
  + Install the packages.
  + You can use the build command and deploy the application on the server.
  + Start the development server.

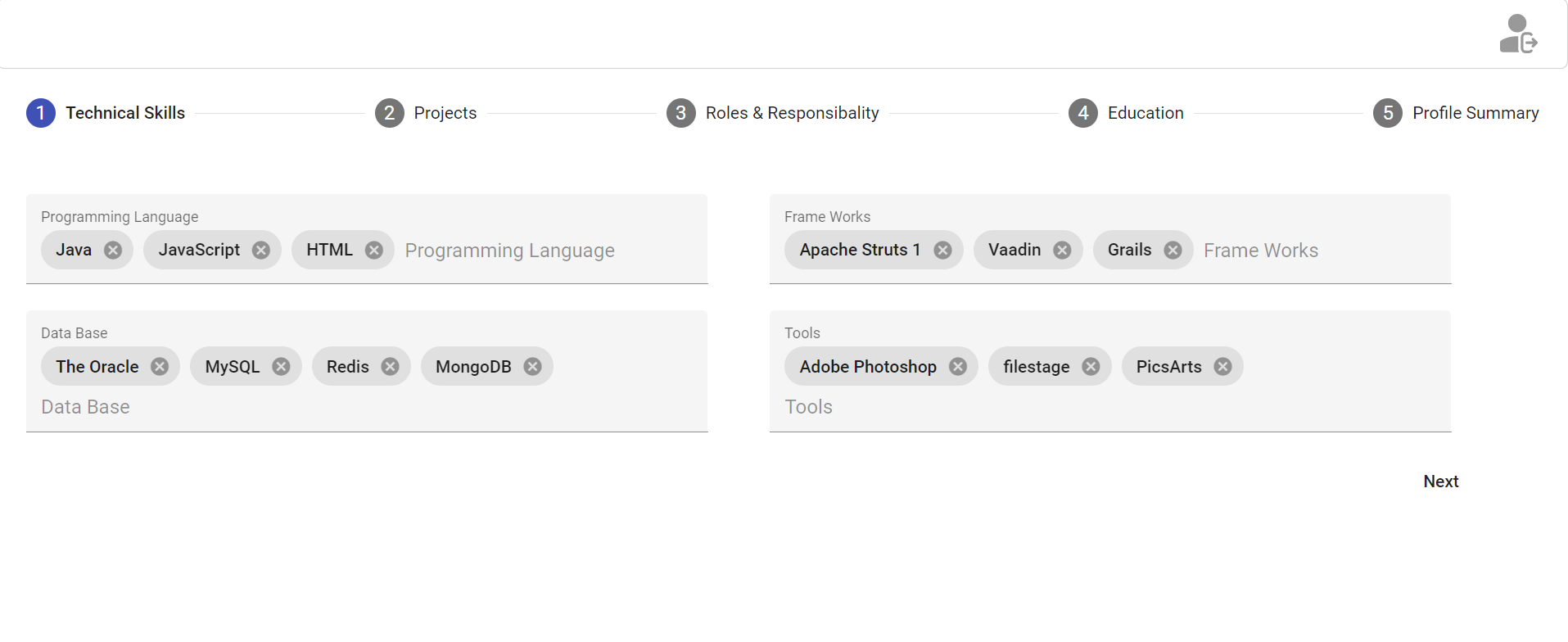
**User Interface Designs**

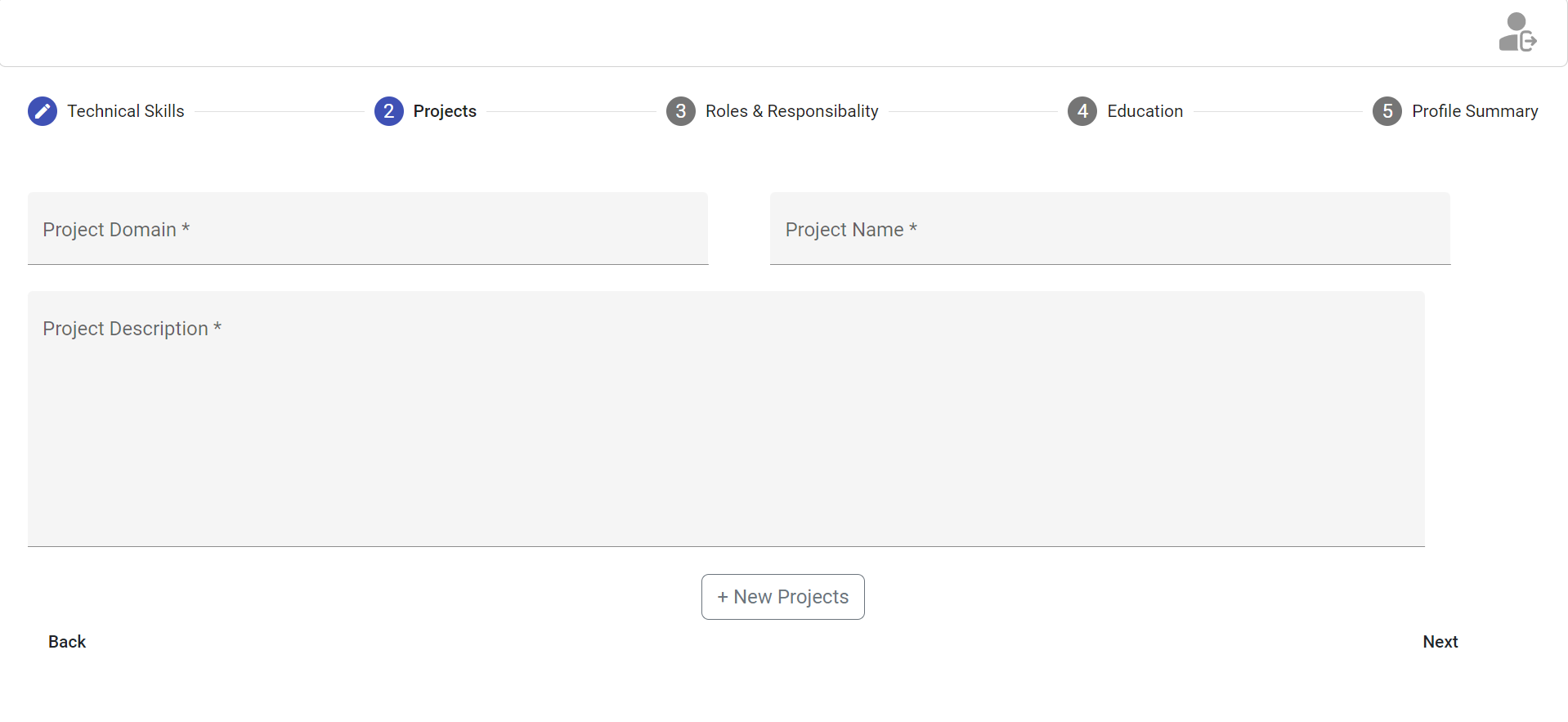
* Login and Register page will look as shown below.

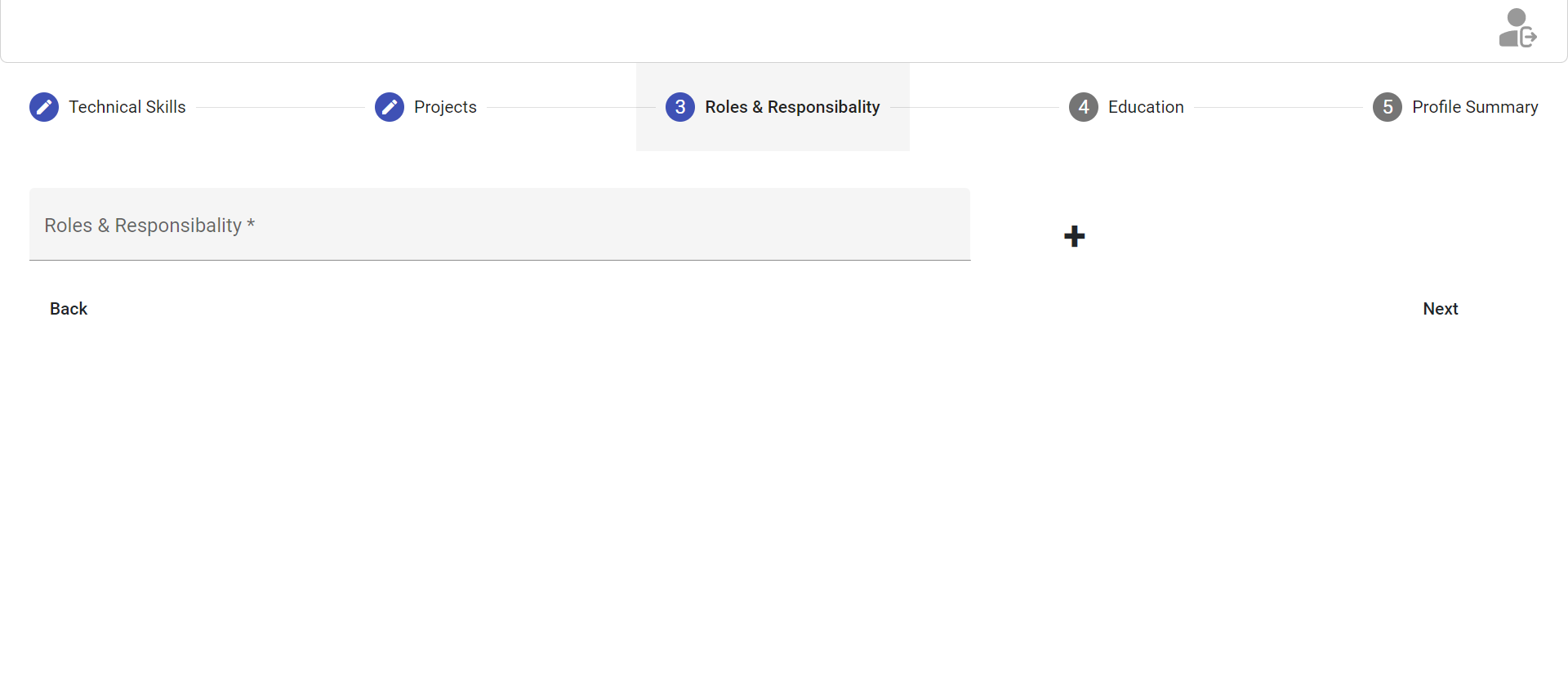


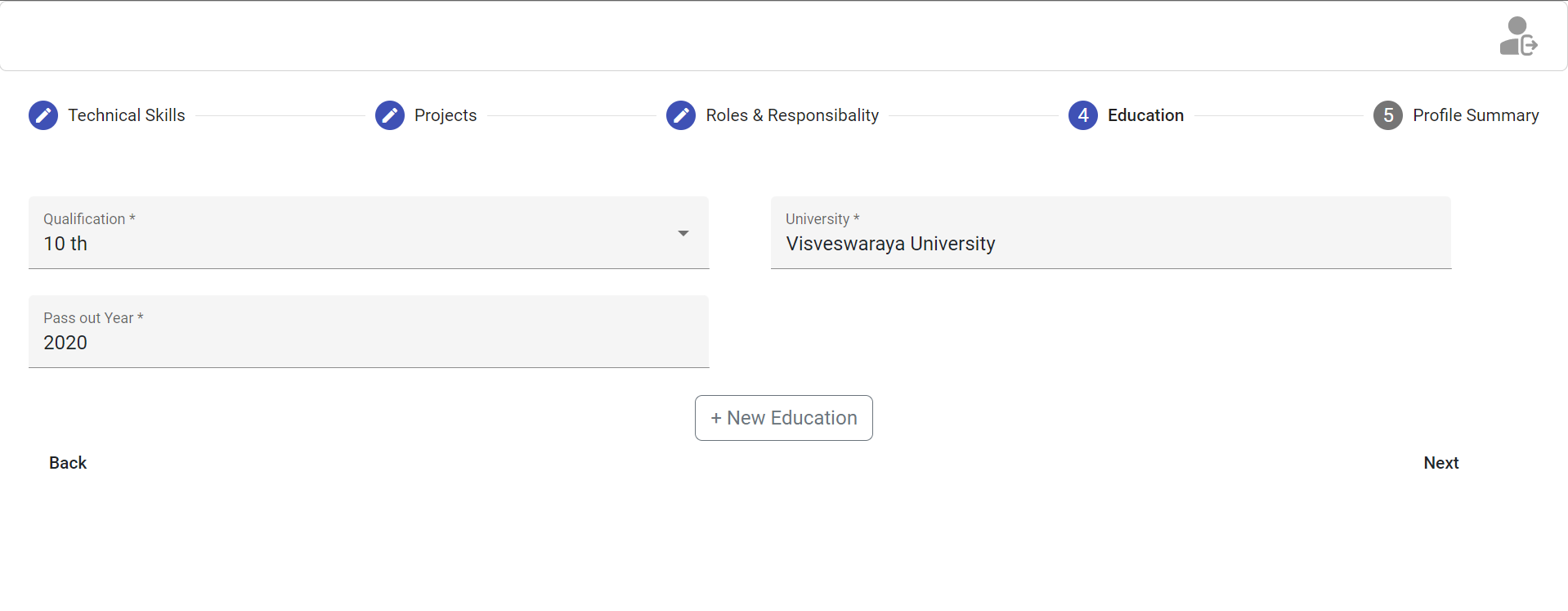


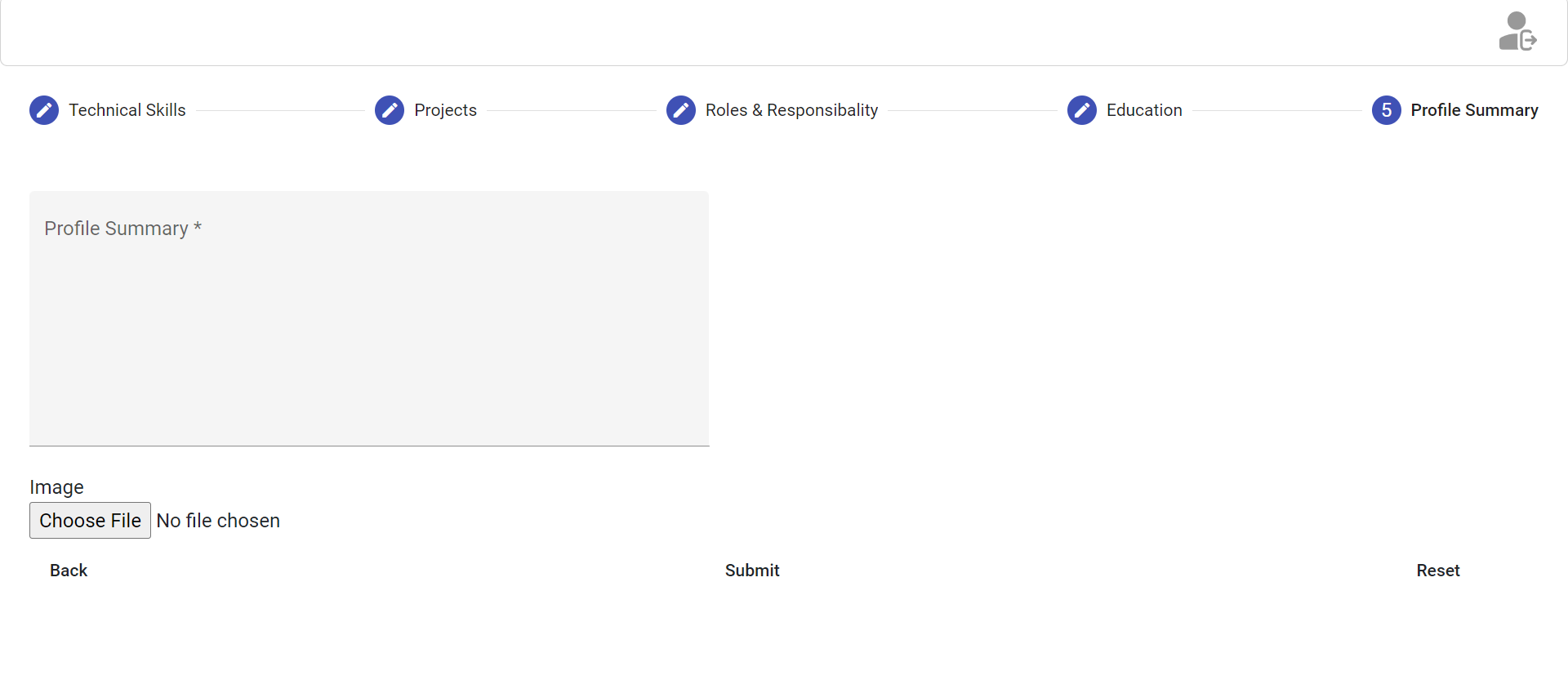
* Below are the screens which has form that user has to fill. Collect-User-Info page.











* User details page will look as shown below.

