

Documentation

Requirements:

Create a Web Interface with 3 subpages - Dashboard, Registration Form, Update Details, to record registered user details

Technologies Used:

1. HTML
2. CSS
3. Flask
4. SQLite
5. SQLAlchemy
6. MaterializeCSS
7. WTForm
 - a. Custom Validators
 - b. Validation Errors
8. JavaScript

Requirements:

1. All 3 pages navigation should within the same browser window/tab
2. Name field should not accept special characters
3. Salary field should accept only numbers
4. Location field should not accept more than 6 characters and should accept numeric only
5. Plot Age vs Salary on Dashboard
6. You can use SQLite, MySQL or CSV file as backend database
7. Only python should be used to work with backend database
8. Make sure you use HTML, CSS, Python and Flask(optional) or Django(optional)
9. None of the file/folder paths in the code should hardcoded

For Requirement 1 the Website can be navigated through the Navbar present
For Requirement 9 none of the paths have been Hardcoded.

Dashboard:

Requirement:

Data from the backend database should be presented in graphs .

Age vs Salary Graph to be plotted.

Solution:

Data from from SQLite Database is fetched using SQLAlchemy and plotted in a bar chart using Chart.JS

Age vs Average Salary per Age plotted as it made the data more readable.



REGISTRATION FORM:

Requirement:

Form should take inputs from the user for Name, Data of Birth, Location, Salary.

Forms should have the following requirements:

1. Name field should not accept special characters
2. Salary field should take only Integers
3. Location field to contain only Integers and Length should be less than 6 characters
4. Data to be inserted in SQLite Database.

Solution:

The above requirements were met.

Screenshots:

A screenshot of a web browser showing a registration form. The browser's address bar displays "127.0.0.1:5000/register". The page has an orange header with navigation links: "Dashboard", "Update", and "Register". The form contains five input fields labeled "Name", "Salary", "Location", and "Date of Birth", each followed by a horizontal line for text entry. A green "SUBMIT" button is positioned below the "Date of Birth" field.

Figure 1: Registration Form without Data

A screenshot of the same registration form, but with a date picker overlay. The form fields are now populated with placeholder text: "Name" is "ISss_[]", "Salary" is "wer56tghyuj", and "Location" is "sew345tghyuj". The "Date of Birth" field is empty. A date picker is open, showing the month of September 2020. The date "Mon, Sep 14" is highlighted in a teal box. The date picker also shows a calendar grid with days 1 through 30. "CANCEL" and "OK" buttons are at the bottom of the date picker.

S	M	T	W	T	F	S
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Figure 2: Date Picker to Insert Date of Birth

The screenshot shows a web form with an orange header bar containing the links 'Dashboard', 'Update', and 'Register'. The form contains four input fields with the following labels and values:

- Name:** !Sss_[]
- Salary:** wer56tghyuj
- Location:** sew345tghyuj
- Date of Birth:** 2020-09-08

Below each input field, there are red error messages:

- Under Name: Special characters not allowed
- Under Salary: Should be numeric only
- Under Location: Should not exceed 6 characters, Should be numeric only

At the bottom of the form, there is a green button.

Figure 3: :Validation Errors as per requirements is shown

UPDATE DETAILS:

Requirements:

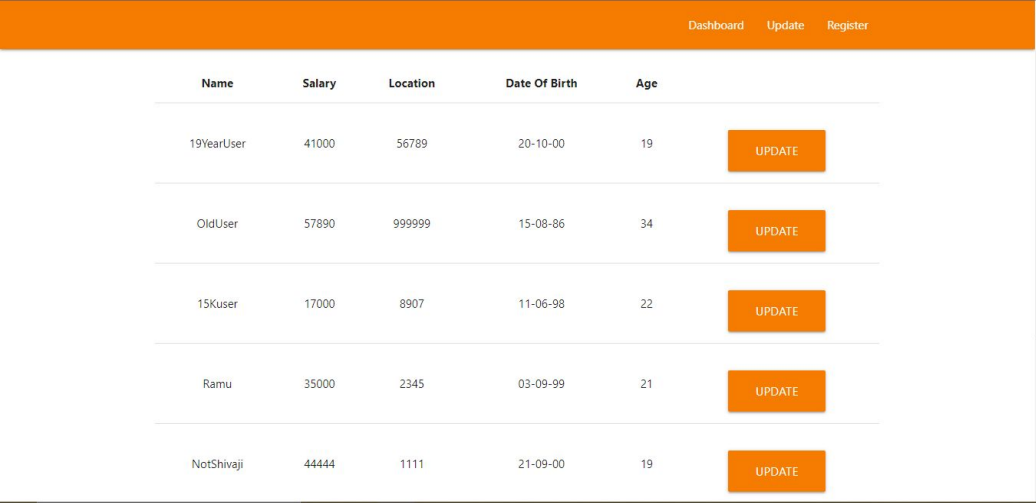
This page should display all the entries from database and should have an option to edit any value and update the database upon savings the changes.

Solution:

The details of all Users are fetched from Database and shown in Tabular Format with a button to Update Data.

User is redirected to another page as data is shown in a concise manner.

Screenshots:

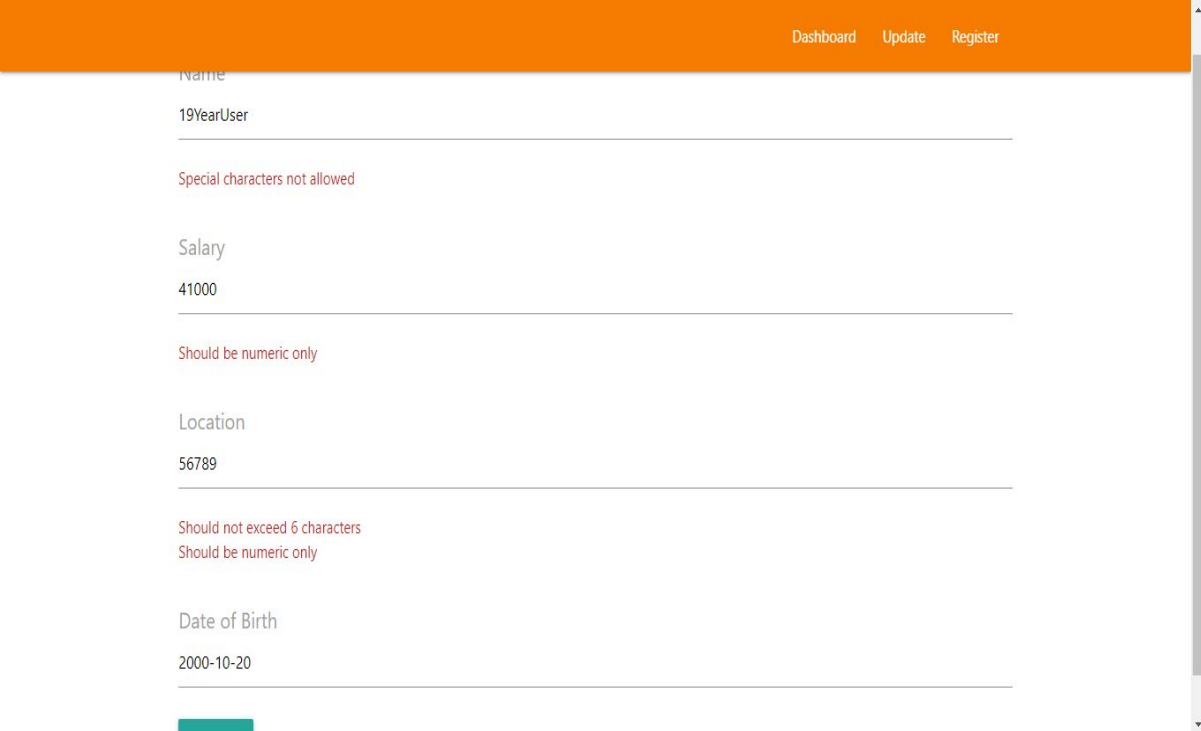


The screenshot shows a web application interface with an orange header bar containing the links 'Dashboard', 'Update', and 'Register'. Below the header is a table with five columns: 'Name', 'Salary', 'Location', 'Date Of Birth', and 'Age'. The table contains five rows of user data. Each row has an orange 'UPDATE' button to its right. The table is styled with horizontal lines separating the rows.

Name	Salary	Location	Date Of Birth	Age	
19YearUser	41000	56789	20-10-00	19	UPDATE
OldUser	57890	999999	15-08-86	34	UPDATE
15Kuser	17000	8907	11-06-98	22	UPDATE
Ramu	35000	2345	03-09-99	21	UPDATE
NotShivaji	44444	1111	21-09-00	19	UPDATE

Figure 4: Table with all the data from backend with an option to update

After clicking on Update user is redirected to Specific User Page



The screenshot shows a web application interface with an orange header bar containing the links 'Dashboard', 'Update', and 'Register'. Below the header is a form for updating user details. The form has five input fields: 'Name', 'Salary', 'Location', 'Date of Birth', and 'Update'. The 'Name' field contains '19YearUser' and has a red error message 'Special characters not allowed' below it. The 'Salary' field contains '41000' and has a red error message 'Should be numeric only' below it. The 'Location' field contains '56789' and has two red error messages below it: 'Should not exceed 6 characters' and 'Should be numeric only'. The 'Date of Birth' field contains '2000-10-20'. The 'Update' button is a green button at the bottom of the form.

Name
19YearUser
Special characters not allowed

Salary
41000
Should be numeric only

Location
56789
Should not exceed 6 characters
Should be numeric only

Date of Birth
2000-10-20

Update

Figure 5 : Custom Validation on Update forms

DashboardUpdateRegister

Name

19YearUserNameChange

Special characters not allowed

Salary

41001

Should be numeric only

Location

56780

Should not exceed 6 characters
Should be numeric only

Date of Birth

2000-10-14

Figure 6: New data is Inserted

DashboardUpdateRegister

Name	Salary	Location	Date Of Birth	Age	
19YearUserNameChange	41001	56780	14-10-00	19	UPDATE
OldUser	57890	999999	15-08-86	34	UPDATE
15Kuser	17000	8907	11-06-98	22	UPDATE
Ramu	35000	2345	03-09-99	21	UPDATE
NotShivaji	44444	1111	21-09-00	19	UPDATE

Figure 7 : Changed data is reflected in Database