Project Specifications of Blooper App

Whoopie Bard <whoopie@clueless.dev> 15 November, 2024

Contents

1	Fund	ctional	Specification	ns															
	1.1	User R	egistration																
		1.1.1	Requireme	nt Sp	oeci	fica	atio	ns											
		1.1.2	Wireframe	S															
		1.1.3	Level 1.1.3																
0	Dura		:G+:	_															
2	Program Specifications 2.1 User-registration Module																		
	2.1	User-re	egistration l	Modu	ıle														
		2.1.1	Screens .																
	2.2	Level 2	2.2																
		2.2.1	Level 2.2.1																
		2.2.2	Level 2.2.2																
		2.2.3	Level 2.2.3																



1 Functional Specifications

1.1 User Registration

This module defines the user-registration aspect of the Blooper. Anonymous users are allowed to self-register and proceed to use the application.

1.1.1 Requirement Specifications

Following requrements specifications are applicable to the user-registration module of the application.

- New user should be able to self-register to the Blooper application. Using a specified URL, anonymous users would be able to fetch a web-form to fill-up.
- · User should be able to register from certain geographical regions only.
 - Q: How to prevent people from using VPN to beat geographical restrictions?
 - Q: Is there any third party service, which provides such facilities?
- Users with an email address and mobile number should be able to create an account in Blooper.
- · No two user-accounts should be possible to be created using same email address.
- No two user-accounts should be possible to be created using same mobile number.

1.1.2 Wireframes

Following wireframes are proposed for the user-registration screen.

1.1.2.1 User-registration Form

This screen is displayed to a new user whose user-account is yet to be created.

Initially, all fields turn out empty. The user needs to fill the required fields in this form.

↑ paragraph 2.1.1.1

1.1.2.1.1 Form-Fields

Following form-fields are expected to be presented to the user for self-registration.

	Allowed Values	Max Size	Required	Unique
Address	A-Z, a-z, 0-9, - +	char(512)	No	
Email Address	.a-z0-9+@	char(128)	Yes	Yes
Full Name	A to Z, and a to z	char(128)	No	
Mobile Number	+-0123456789	char(12)	Yes	Yes

Email Address

It would be unique across the user-database.

That means, no two users can have same email address.

· Mobile Number

Mobile number should be unique across the application-database.

No two users using this application can have same mobile number.



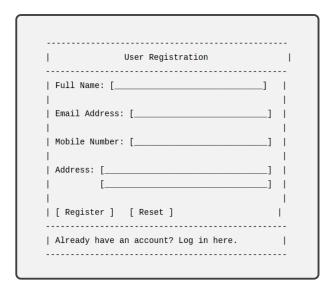


Figure 1: User-registration Form

1.1.2.1.2 Buttons

Following buttons are expected to be made available on the screen.

- · Register: Registers the new user.
 - Normal Flow
 - * System validates supplied data before creating user-account.
 - * If everything is in order, then system create new user-account.
 - Exception Flow

If one or more validations fail, then system displays validation error with following details.

- * If email address is associated with any existing user-account, then system notifies that no new account can be created with same email address.
- * If mobile number is associated with any existing user-account, then system notifies the user that no new account can be created with the same mobile number.
- Reset: Reset all fields of the registration form.

។ subparagraph 2.1.1.1.1

ካ ??

- Normal Flow
 - * Clean-up contents of all fields.

1.1.2.1.3 Links

Following hyperlinks are expected to be present on the screen.

• Login Here: If account is present, click to sign-in.

ጎ ??



1.1.2.2 Message Box for Existing Record

This screen is displayed, when one or more fields which are unique across the application is supplied as part of user-registration, but the same had been supplied earlier for another user-account too.

This could be a pop-up window.

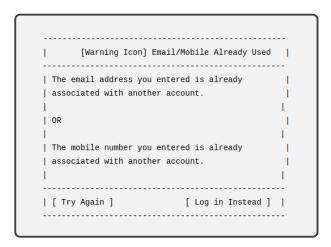


Figure 2: Message Box for Existing Record

1 paragraph 2.1.1.2

1.1.2.2.1 Buttons

• **Try Again**: Re-submit the form by going to the registration form and by modifying its form-fields.

```
Q: Is re-executing the GET request on the form required here?
```

Q: Is it not sufficient enough to handle it locally, by rendering the cached form itself?

• Login Instead: If an account exists already, proceed to the login form.

1.1.3 Level 1.1.3

Q: How can Level 1.1.3 be used as a section heading?

1.1.3.1 Sample Table

A Sample table is given below.

	Header 1	Header 2	Header 3	Header 4
Row 1	1.1	1.2	1.3	1.4
Row 2	2.1	2.2	2.3	2.4
Row 3	3.1	3.2	3.3	3.4
Row 4	4.1	4.2	4.3	4.4
Row 5	5.1	5.2	5.3	5.4



· Row 1

First note for Row 1.

Second note for Row1.

Third note for Row1.

Row 3

First note for Row 3.

Second note for Row 3.

Third note for Row 3.

Fourth note for Row 3.

2 Program Specifications

2.1 User-registration Module

This section elaborates on the program specifications of implementation of user-registration module of Blooper.

The functional requirements of this module is described in

2.1.1 Screens

2.1.1.1 User-registration Screen

Please refer

2.1.1.1.1 Buttons

- Register
 - API Endpoint: /api/v1/users/register
 - Method: POST

HTTP POST method is used on this API endpoint for the intended result. POST method requires input data to be sent in HTTP request-body itself.

Input

Input formats are described below.

* This is the JSON format of input.

```
{
    "fullname": "Jagmeet Chautala",
    "emailaddress": "jagmeetc@example.com",
    "mobilenumber": "1234567890",
    "address": "72, B street, Park Avenue, Sonpat,
    Hryana, INDIA"
}
* It is also possible that some partial data is supplied by ignoring optional details.
{
    "fullname": "Jagmeet Chautala",
    "emailaddress": "jagmeetc@example.com",
    "mobilenumber": "1234567890"
```



```
- Normal Output
```

```
{
    "status": 200
    "message": "Created"
}
- Exception Output
{
    "status": 421,
    "message": "Invalid field supplied"
```

Reset

On-click, it empties all the fields of the form.

- Method: Javascript function to reset all form-fields.

2.1.1.1.2 Links

2.1.1.2 Error-Message Box for Existing Record

2.1.1.2.1 Buttons

- 2.2 Level 2.2
- 2.2.1 Level 2.2.1
- 2.2.2 Level 2.2.2
- 2.2.3 Level 2.2.3
- 2.2.3.1 Level 2.2.3.1