**Software Requirements Specification**

**for**

MyBank

**Version 1.0 approved**

**Prepared by Antici Maria-Daniela, Calota Vlad-Marin & Vasile Radu-Mihail**

**Universitatea din Craiova**

**02.03.2021**

**Table of Contents**

**Table of Contents ii**

**Revision History ii**

**1. Introduction 1**

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

**2. Overall Description 2**

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

**3. External Interface Requirements 3**

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

**4. System Features 4**

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

**5. Other Nonfunctional Requirements 4**

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

**6. Other Requirements 5**

**Appendix A: Glossary 5**

**Appendix B: Analysis Models 5**

**Appendix C: To Be Determined List 6**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# **Introduction**

## **Purpose**

The purpose of this software is to enable microtransactions, balances and other financial operations between clients. A client can see his balance in various currencies such as euro, ron and usd. The version of this product is 1.0 as it is in inception.

## **Document Conventions**

The document conventions of the SRS are: Arial style, size 11.

The bold words represent critical technical data or signs important paragraphs.

The italic words represent under discussion terms that may be changed at a later edit. ( are temporary )

## **Intended Audience and Reading Suggestions**

The intended audience of this document is represented by anyone that wants to buy / implement our product **but** a potential user can also read it as it contains important data that is also useful to him.

**NOTE:** If a manager / sales wants to get a better understanding of this document he might need a developer to explain the technical terms to him such as technologies and so on.

## **Product Scope**

The aim of mybank is to facilitate easy transactions and storage of currencies, to have loyal customers by digitizing the banking process and to bring to clients with the offering of a digital variance of the classic banking process.

One more thing that myBank does is that it reduces the number of employees a bank needs.

## **References**

Styles:

<https://www.ed.ac.uk/information-services/help-consultancy/accessibility/creating-materials/word-documents>

Additional guidelines: <https://bit.ly/3b7yuMY>

# **Overall Description**

## **Product Perspective**

The benefits of using our system is instant updates because we will use a dedicated database that is linked to the website. It will have a clear interface with no additional corporate functions that confuse the client. Another benefit is the lack of ads.

## **Product Functions**

* Login
* Balance infos
* Changing the address
* Changing the password
* Financial operations between clients
* Utilities payment
* Transaction history

## **User Classes and Characteristics**

The product will be used by only one class: Users.

It is meant for the daily usage of people without technical expertise or experience.

All the functionalities and only their personal data will be available to them.

Access to another’s account is restricted by credentials.

It is mandatory to have a minimum of 10 years in school in order to be an eligible candidate for an account at MyBank.

## **Operating Environment**

A web application (or web app) is application software that runs on a web server, unlike computer-based software programs that are run locally on the operating system (OS) of the device. Web applications are accessed by the user through a web browser with an active network connection.

Google Chrome Version 89.0.4389.82 is required.

## **Design and Implementation Constraints**

Timing requirement: 11 weeks.

IDE: Visual Studio 2019

Front-end: HTML5, CSS3, JavaScript, Bootstrap 4.6

Back-end: C Sharp, ASP.NET MVC Core 3.1

Database: SQL Server on a Linux Container in Docker 3.2.1

Database Management: Azure Data Studio 1.29.1

Other tools: Postman 7.30.1 for creating and saving simple and complex HTTP/s requests, as well as read their responses.

Testing: NUnit, Selenium

The customer’s organization will be responsible for maintaining the delivered software

## **User Documentation**

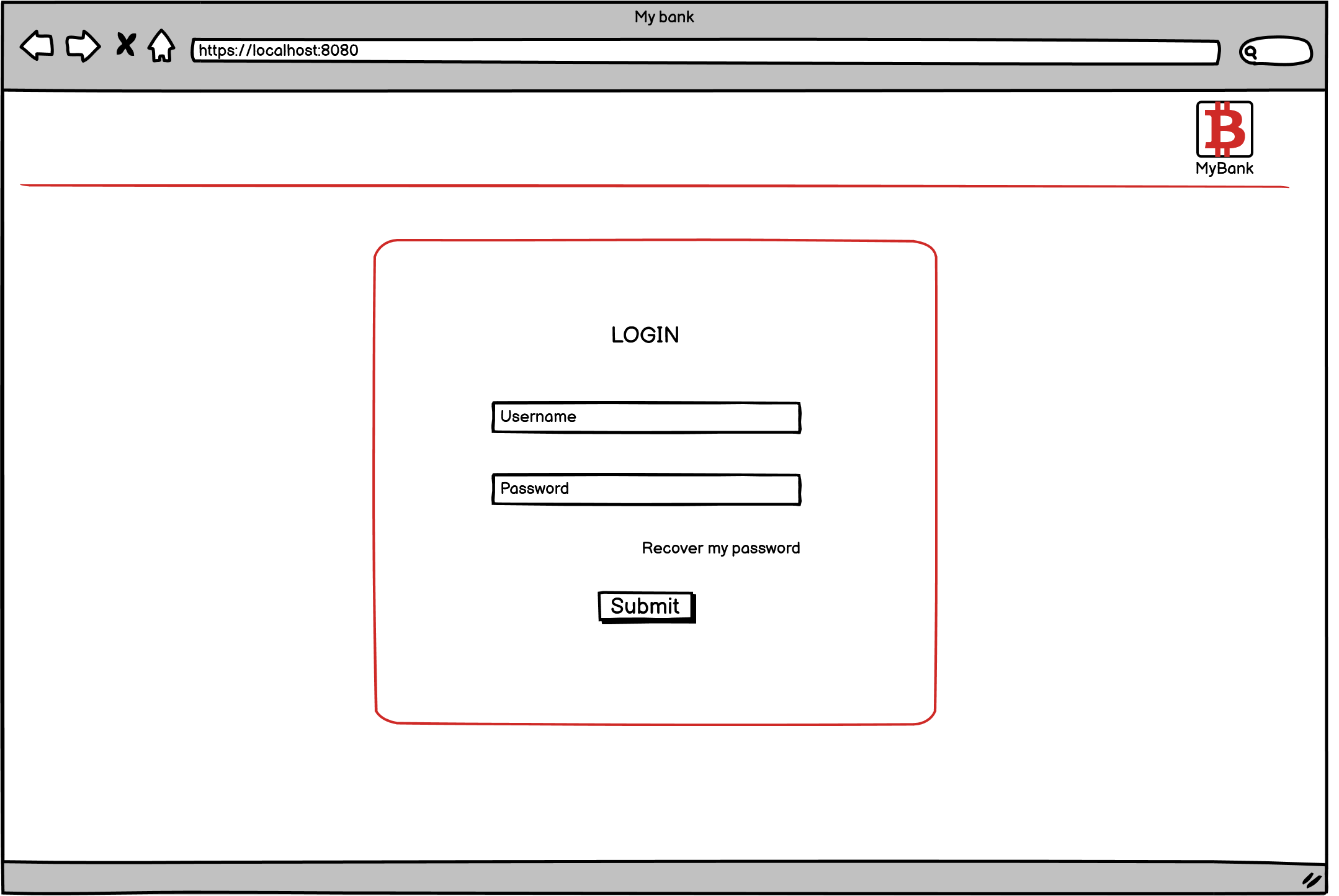
We will provide the users with one user manual and a basic-functionalities tutorial.

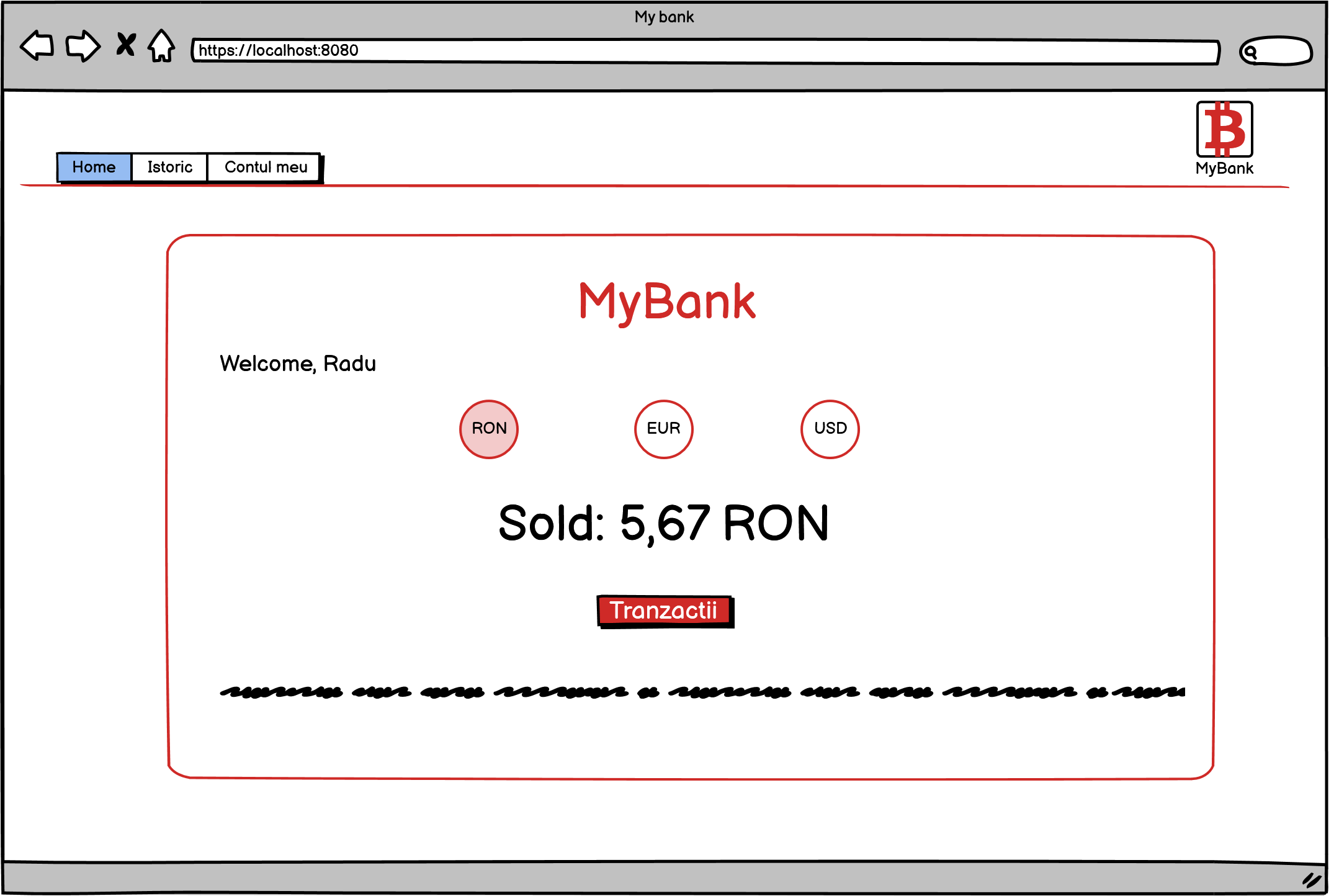
## **Assumptions and Dependencies**

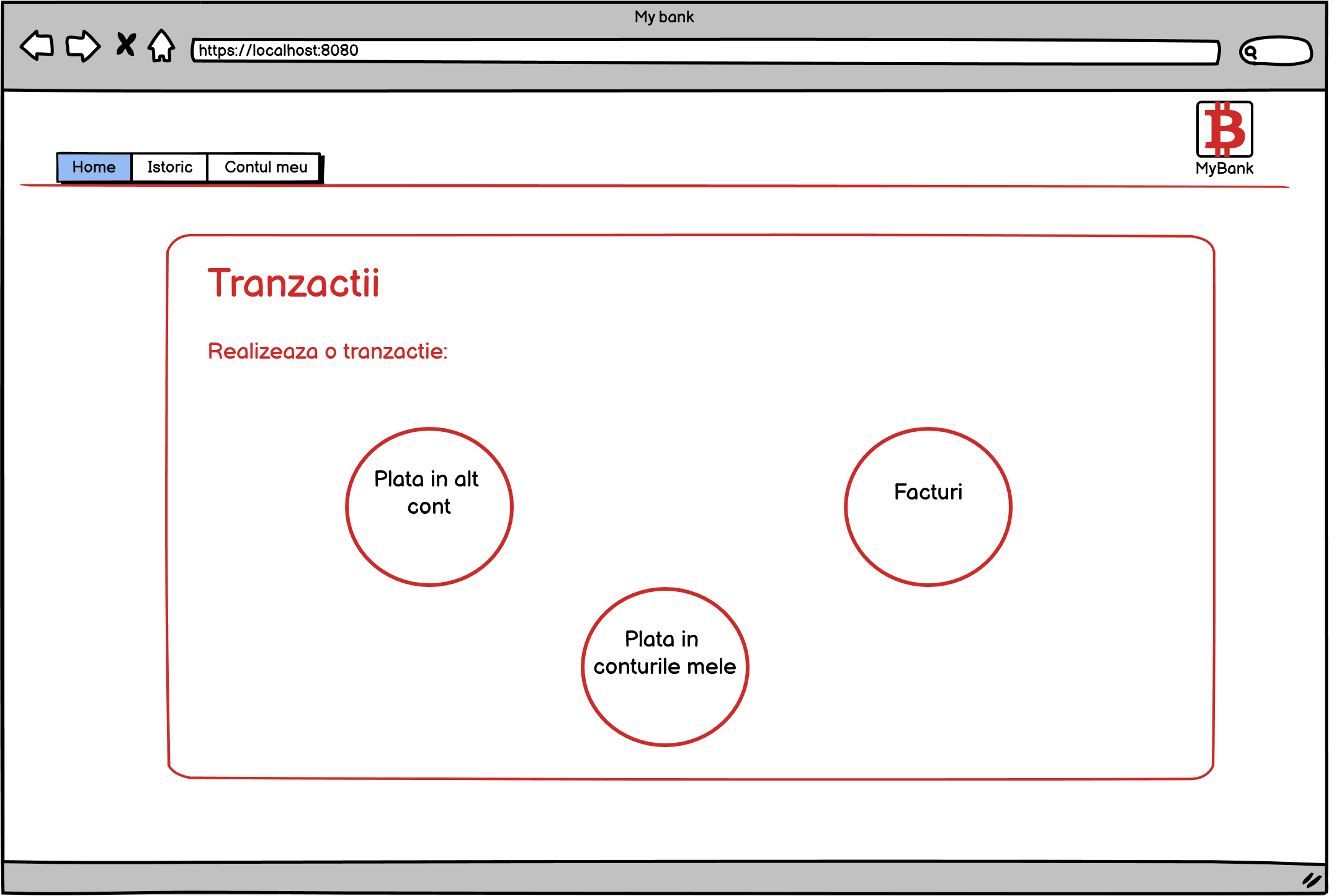
Google Chrome Version 89.0.4389.82 is required.

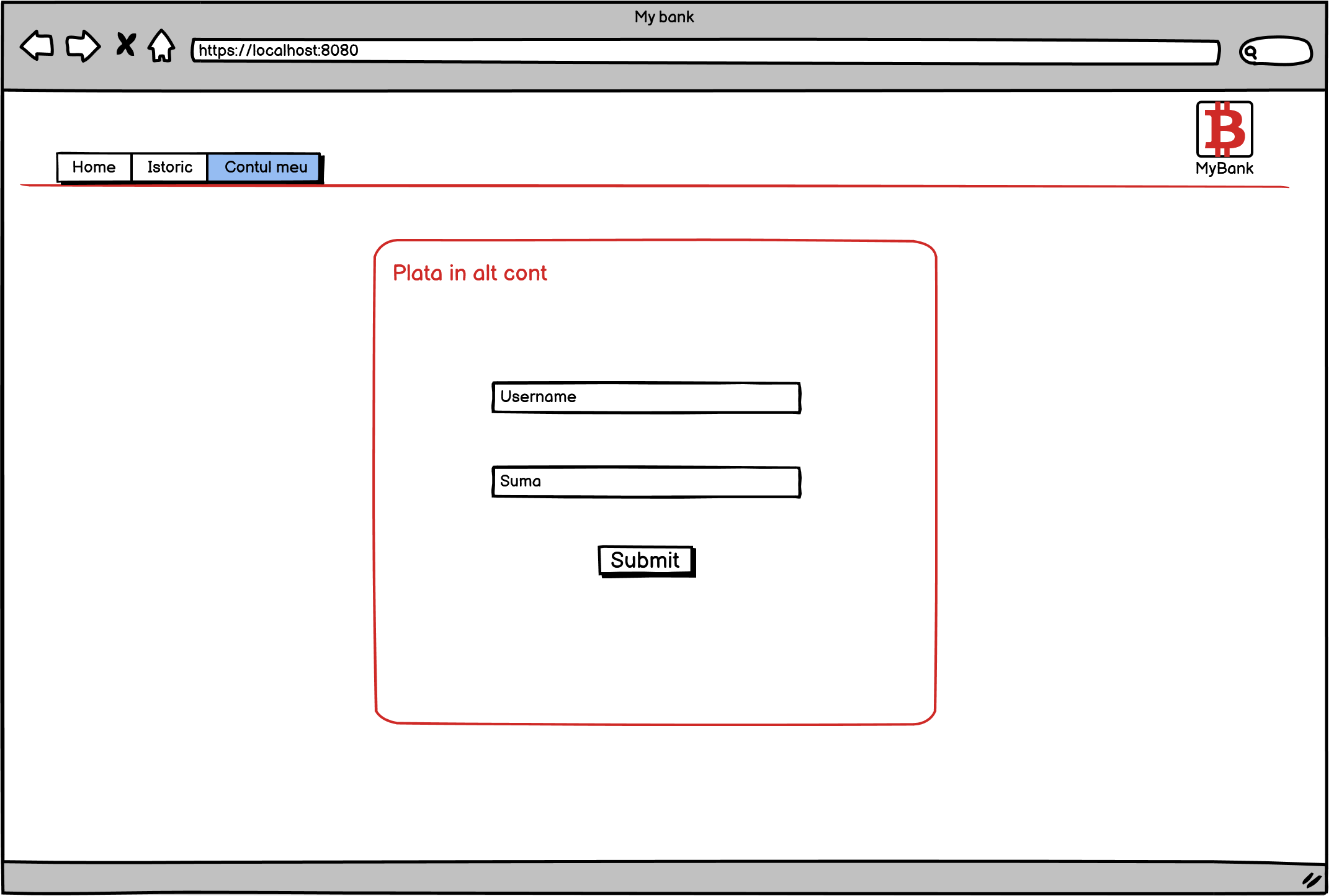
# **External Interface Requirements**

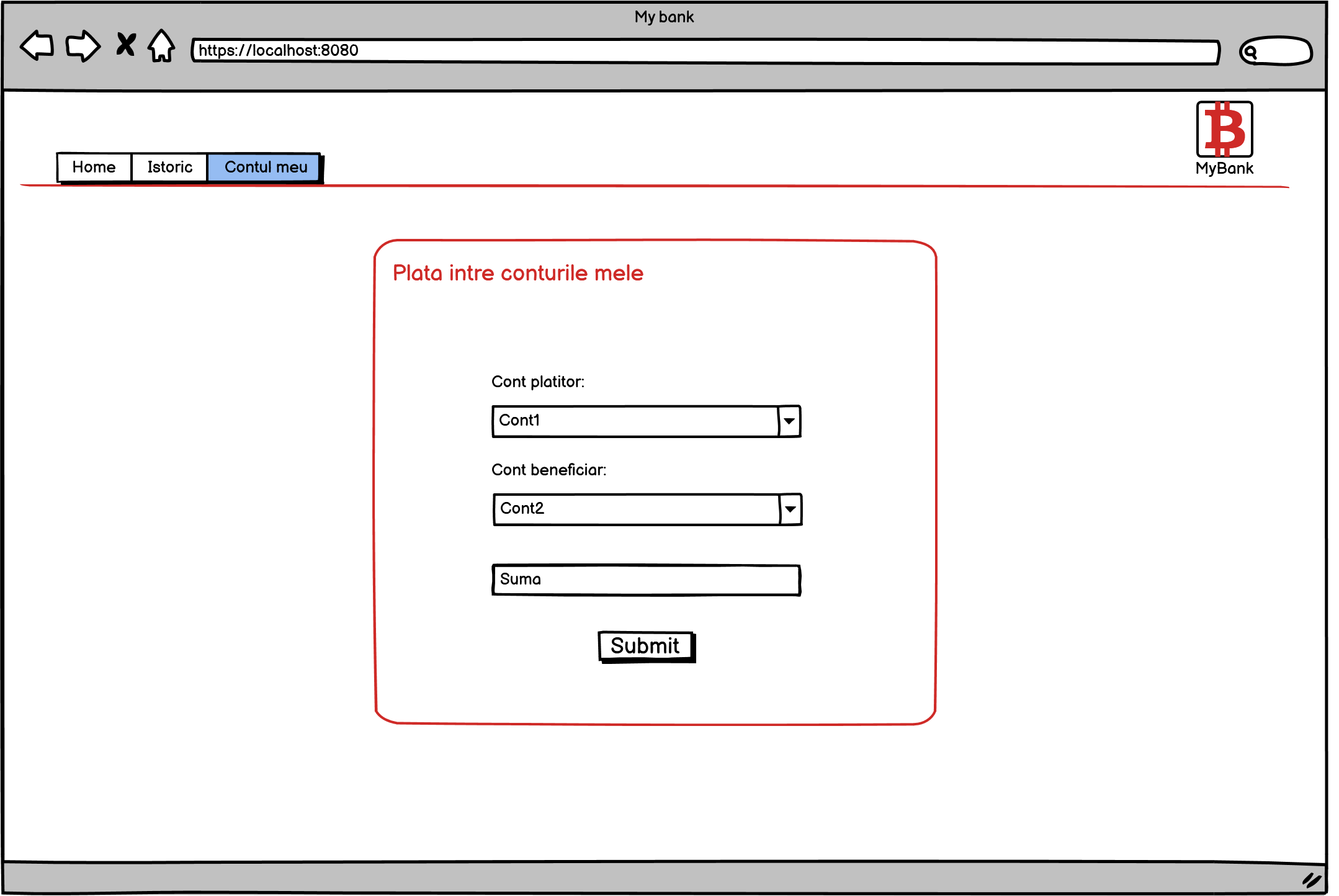
## **User Interfaces**

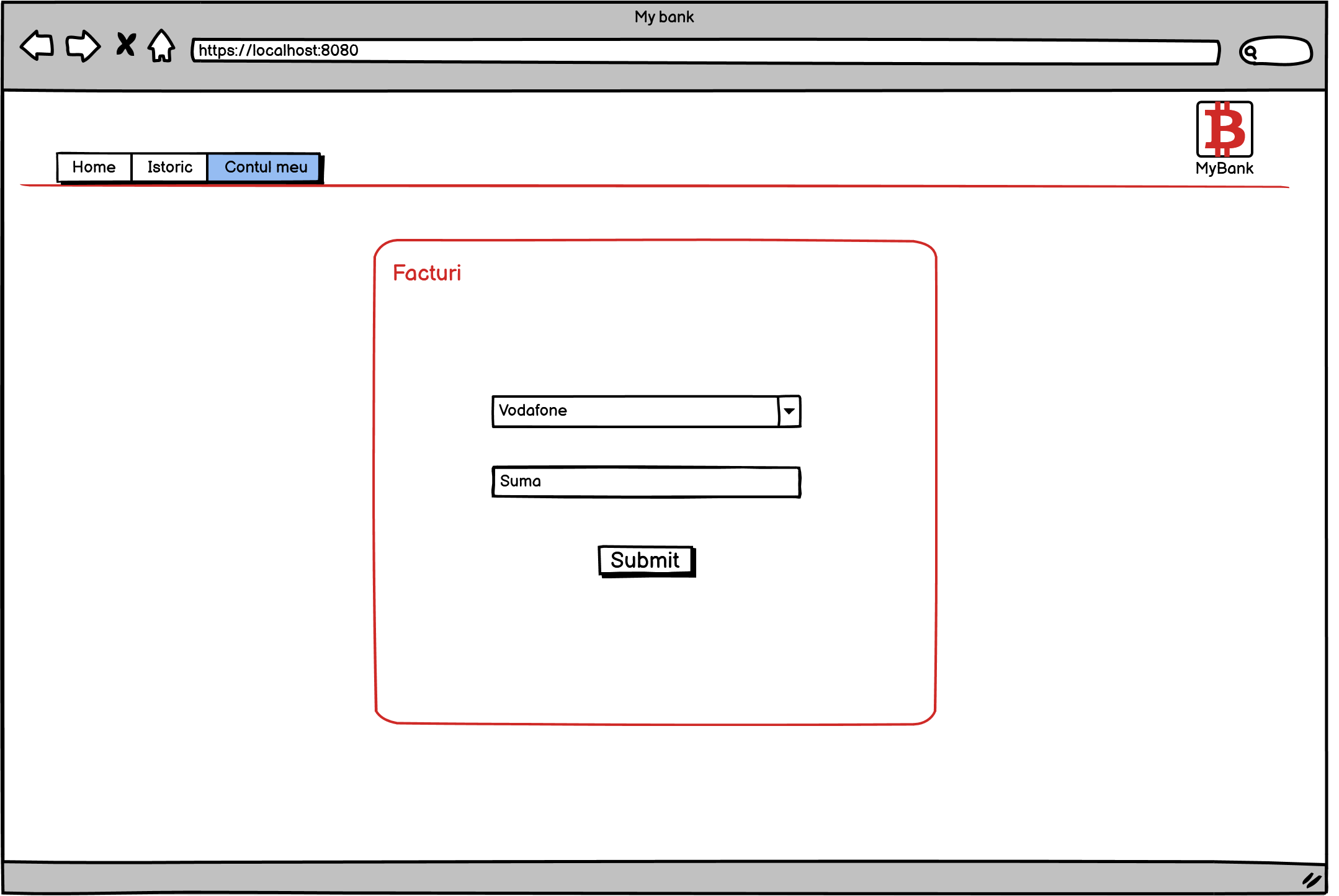
**

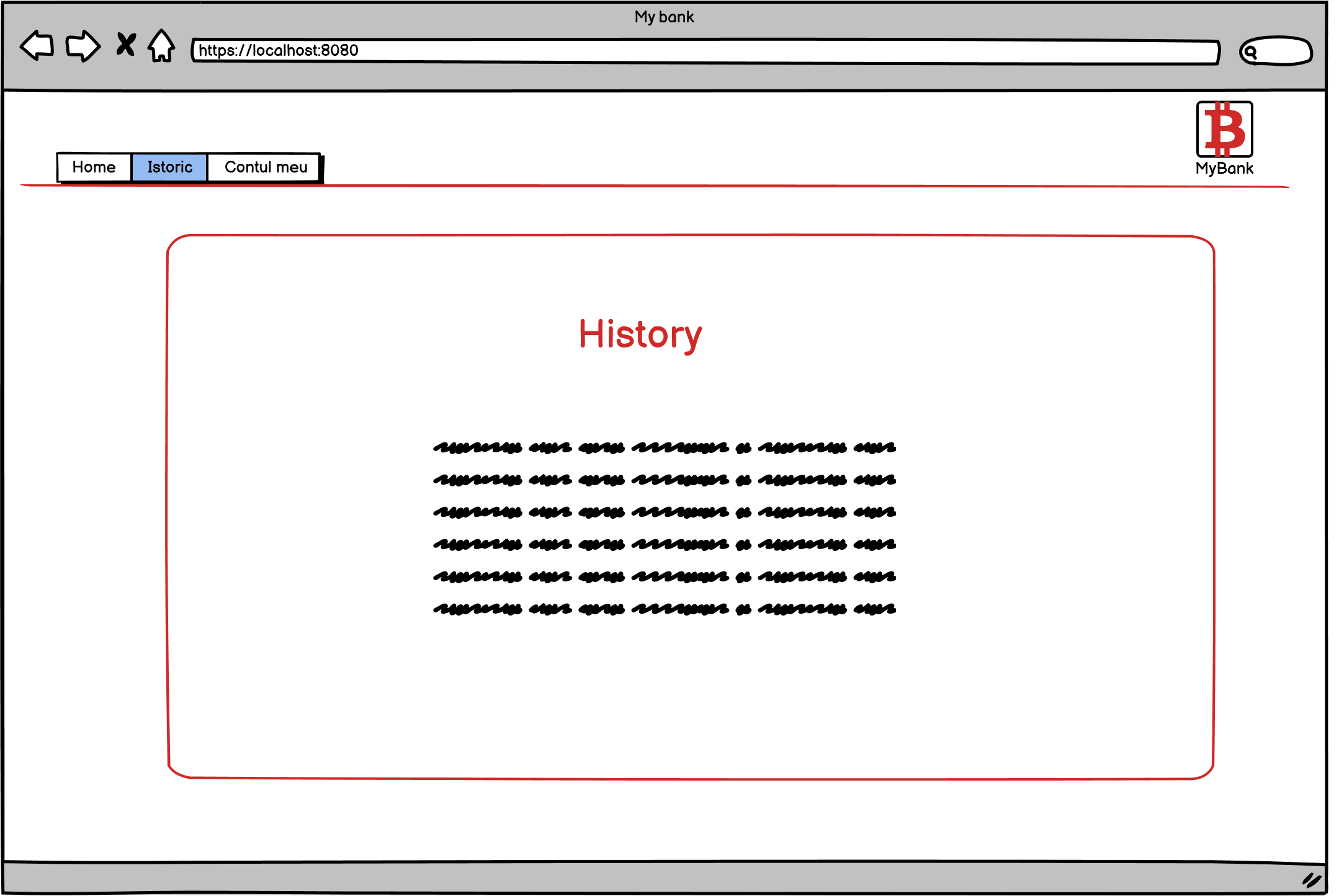
**

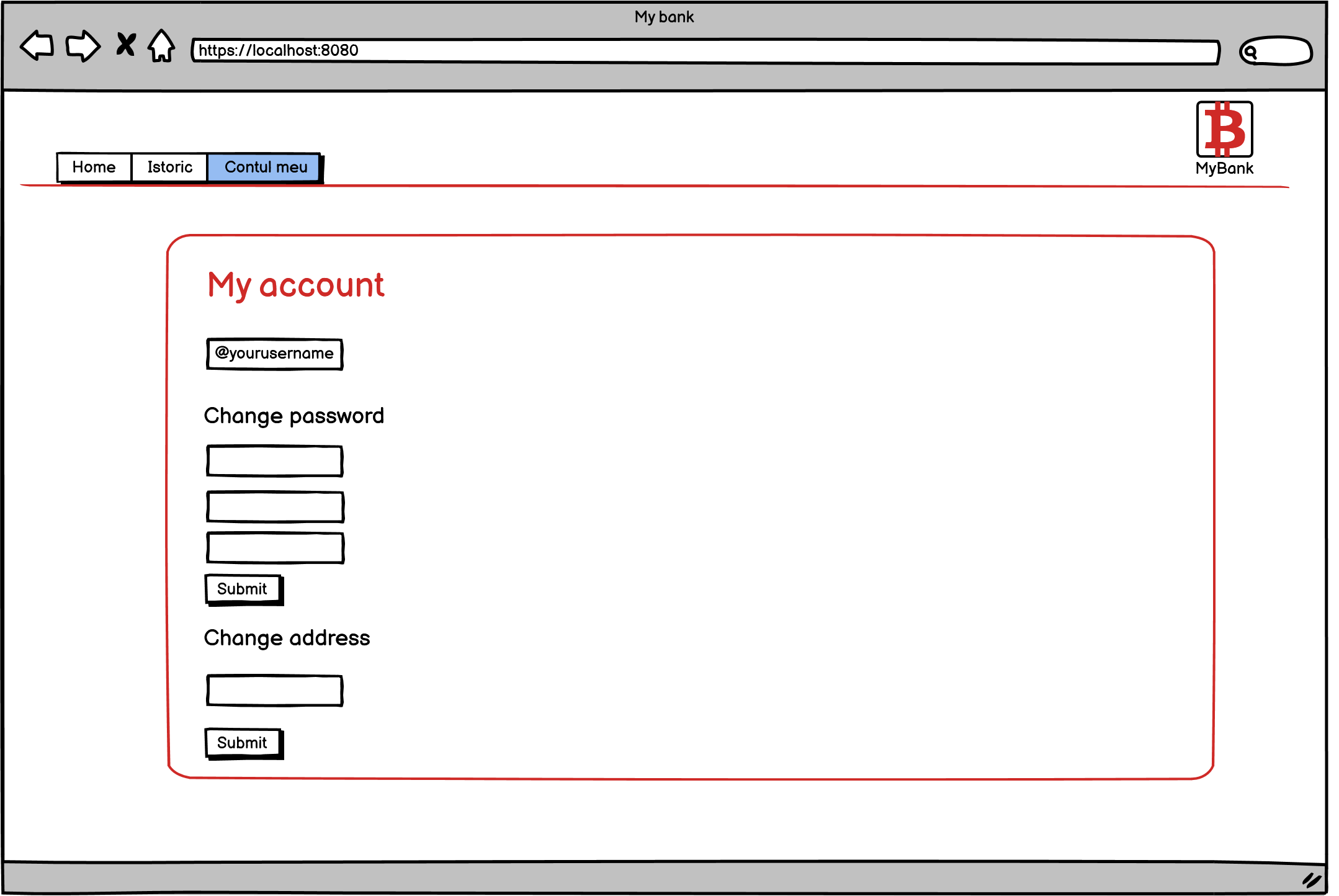
**

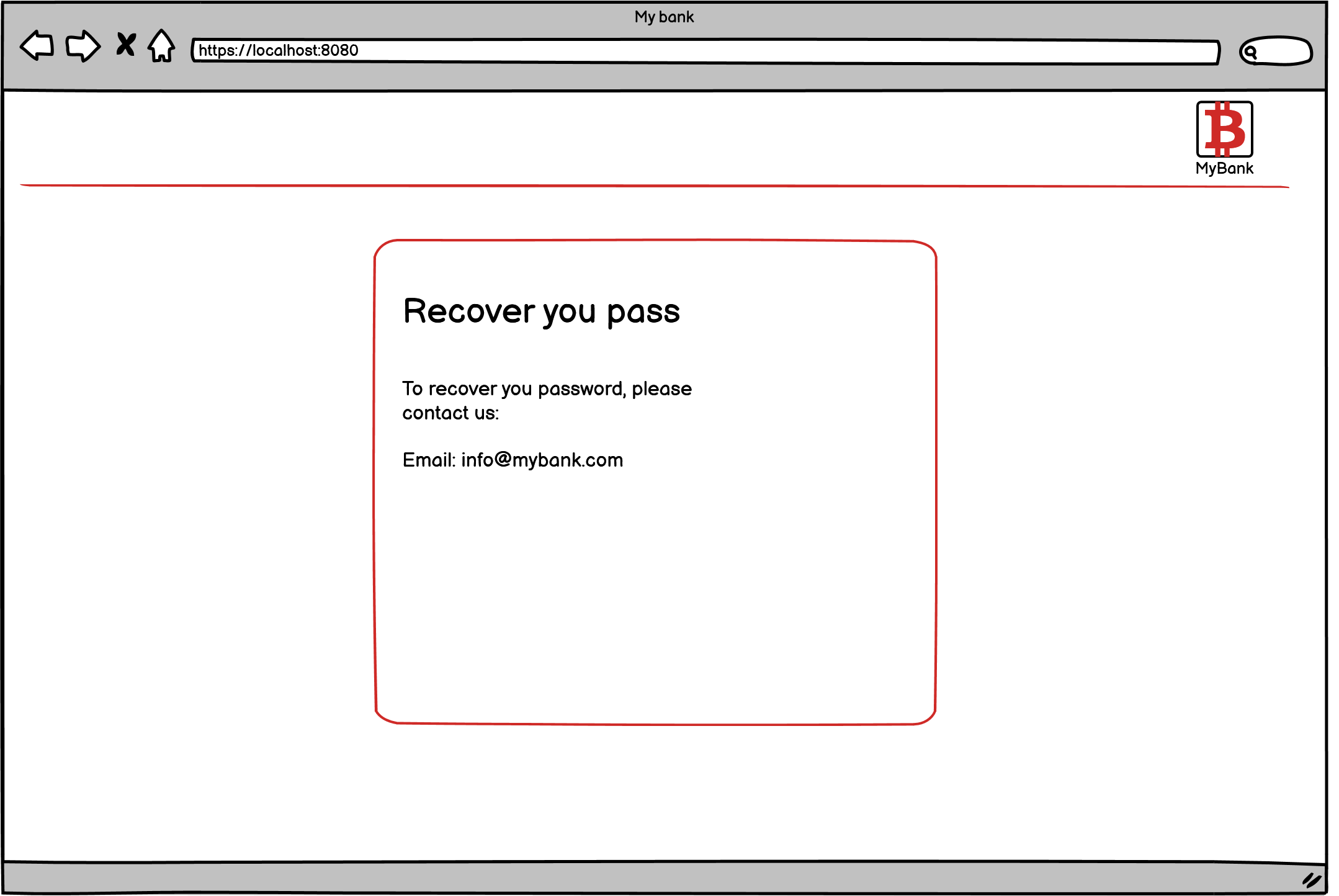
**

**

**

**

**

**

## **Hardware Interfaces**

Not applicable

## **Software Interfaces**

IDE: Visual Studio 2019

Front-end: HTML5, CSS3, JavaScript, Bootstrap 4.6

Back-end: C Sharp, ASP.NET MVC Core 3.1

Database: SQL Server on a Linux Container in Docker 3.2.1

Database Management: Azure Data Studio 1.29.1

Other tools: Postman 7.30.1 for creating and saving simple and complex HTTP/s requests, as well as read their responses.

Communication protocol: HTTP

## **Communications Interfaces**

**HTTP** stands for hypertext transfer protocol. It's a protocol that allows communication between different systems. It will be used for transferring data from a web server to a browser to view web pages.

# **System Features**

* 1. **Login functionality**

4.1.1 Description and Priority

The first page, that is displayed when the website is accessed, will contain a form with 2 input fields: name and password, a submit button and a ‘recover your password’ hyperlink. The priority of this feature is very high, because it is the main gate used by users to access the website.

4.1.2 Stimulus/Response Sequences

*Steps:*Access the website

Add valid credentials

Click on the submit button

*Website’s response:*

The user is redirected to the main interface

*Steps:*Access the website

Add valid credentials

Click on the submit button

*Website’s response:*

The user is redirected to the main interface

4.1.3 Functional Requirements

REQ-1: 2 fields - 1 for password and 1 for code/username

REQ-2: 1 submit button

REQ-3: Validation text - for the scenarios when the credentials are incorrect

## 4.2 Recover my password 4.2.1 Description and priority This functionality is made to help the users to recover their forgotten password. Priority - high 4.2.2 Stimulus/Response Sequences Go to the login form Click on the recover password Expected result: The user is redirected to a contact page 4.2.3 Functional Requirements

REQ-1: Recover my password button in the login page

REQ-2: Contact page with info text

4.3 **Check my balance**

4.3.1 Description and priority  
  
This functionality is available in the homepage and it is used to display the balance according to the selected account type. Priority - medium

4.3.2 Stimulus/Response Sequences

Login

Go to the homepage

By default, the predefined currency is RON

Expected result:

It is displayed the current balance for the RON account

Login

Go to the homepage

Select EUR

Expected result:

It is displayed the balance for the EUR account

4.3.3 Functional requirements

REQ-1: button for selecting the EUR account

REQ-2: button for selecting the USD account

REQ-3: button for selecting the RON account

4.4 **Change my address**  
  
 4.4.1 Description and priority  
 This functionality is available in the user's settings page. Priority - High  
  
 4.4.2. Stimulus/Response Sequences

Login

Go to user's settings page

Erase the text from the address field and add a new one

Submit the changes

Expected result:

The values are saved

Login

Go to user's settings page

Erase the text from the address field and leave it empty

Submit the changes

Expected result:

Validation text displayed

4.4.3. Functional requirements

REQ-1: City input field  
 REQ-2: County input field  
 REQ-3: Street and house number field

REQ-4: Validation text

4.5 **Payments for services - RON**

4.5.1 Description and priority

This is an important feature offered by the bank to help our customers to pay their invoices for different services easy and fast.

The priority of this functionality is very high.

4.5.2 Stimulus/Response Sequences

Login

Select RON currency

Go to Transactions page

Select the services button

Select any service and make the payment

Expected result:

The payment was made

4.5.3 Functional requirements

REQ-1: services button in the transaction page

REQ-2: a dropdown that allows the user to select the service

REQ-3: an input field for the amount of money

REQ-4: insufficient funds validation text

REQ-5: submit button

4.6 **Payments for services - EUR**

4.6.1 Description and priority

This is an important feature offered by the bank to help our customers to pay their invoices for different services easy and fast.

The priority of this functionality is very high.

4.6.2 Stimulus/Response Sequences

Login

Select EUR currency

Go to Transactions page

Select the services button

Select any service and make the payment

Expected result:

The payment was made

4.6.3 Functional requirements

REQ-1: services button in the transaction page

REQ-2: a dropdown that allows the user to select the service

REQ-3: an input field for the amount of money

REQ-4: insufficient funds validation text

REQ-5: submit button

4.7 **Payments for services - USD**

4.7.1 Description and priority

This is an important feature offered by the bank to help our customers to pay their invoices for different services easy and fast.

The priority of this functionality is medium.

4.7.2 Stimulus/Response Sequences

Login

Select USD currency

Go to Transactions page

Select the services button

Select any service and make the payment

Expected result:

The payment was made

4.7.3 Functional requirements

REQ-1: services button in the transaction page

REQ-2: a dropdown that allows the user to select the service

REQ-3: an input field for the amount of money

REQ-4: insufficient funds validation text

REQ-5: submit button

4.9 **Payments to different accounts - RON**

4.9 Description and priority

This is the main functionality of the banking process. It enables the users to transfer various amounts of currencies between them. It is available after the user has logged in at the login page.

4.9.2 Stimulus/Response Sequences

Login

Select RON currency

Go to the Transaction page

Select ‘payment to different accounts’ button

Add the details of the account that are going to receive the money

Add the amount of money

Press on the submit button

4.8.3 Functional requirements

REQ-1: ‘payment to different accounts’ button

REQ-2: an input field for the amount of money

REQ-3: input field for the account’s details

REQ-4: insufficient funds validation text

REQ-5: submit button

4.10 **Payments to different accounts - EUR**

4.10 Description and priority

This is the main functionality of the banking process. It enables the users to transfer various amounts of currencies between them. It is available after the user has logged in at the login page.

4.10.2 Stimulus/Response Sequences

Login

Select EUR currency

Go to the Transaction page

Select ‘payment to different accounts’ button

Add the details of the account that are going to receive the money

Add the amount of money

Press on the submit button

4.8.3 Functional requirements

REQ-1: ‘payment to different accounts’ button

REQ-2: an input field for the amount of money

REQ-3: input field for the account’s details

REQ-4: insufficient funds validation text

REQ-5: submit button

4.11 **Payments to different accounts - USD**

4.11.1 Description and priority

This is the main functionality of the banking process. It enables the users to transfer various amounts of currencies between them. It is available after the user has logged in at the login page.

4.11.2 Stimulus/Response Sequences

Login

Select USD currency

Go to the Transaction page

Select ‘payment to different accounts’ button

Add the details of the account that are going to receive the money

Add the amount of money

Press on the submit button

4.11.3 Functional requirements

REQ-1: ‘payment to different accounts’ button

REQ-2: an input field for the amount of money

REQ-3: input field for the account’s details

REQ-4: insufficient funds validation text

REQ-5: submit button

4.12 Payments between accounts

4.12

Medium priority

4.12.2 Stimulus/Response Sequences

Login

Go to the Transaction page

Select ‘payments between accounts’

Select the sender

Select the receiver

Add the amount of money

Press on the submit button

4.12.3 Functional requirements

REQ-1 : ‘payments between accounts’ button

REQ-2: input field for the amount of money

REQ-3: dropdown for selecting the sender

REQ-4: dropdown for selecting the receiver

REQ-5: insufficient funds validation text

REQ-6: submit button

4.13 Logout functionality

4.13.1 Logging out from the logged-in account is a good practice because if, for example, the customer is using a public or shared computer. This is because, if he does not log out his online account from the browser, the next person using the same browser will be able to access his accounts as well, since you are still technically logged-in into the websites.

4.13.2 Stimulus/Response Sequences

Login

Click on the log-out button

Expected result:

The user is logged out

4.13.3 Functional requirements

REQ-1: Log-out button

4.14. Change my password

4.14.1 Your computer stores and provides access to a lot of sensitive and important data. Keeping this data and all of your accounts safe is a priority. Since this is a banking application, the credentials of the client should be changed often, that’s why this is a key functionality.

4.14.2 Stimulus/Response

Login

Go to my account page

Insert your current password in the first field

Add a new password in the second one

Insert the new password again in the third one

Press on the submit button

Expected result:

The password is successfully changed

Login

Go to my account page

Insert your a wrong password in the current field

Add a new password in the second one

Insert the new password again in the third one

Press on the submit button

Expected result:

Validation text displayed

Login

Go to my account page

Insert your current password in the first field

Add a new password in the second one

Insert a different password in the third one

Press on the submit button

Expected result:

Validation text displayed

4.14.3 Functional requirements

REQ-1: Input field for the current password

REQ-2: Input field for the new password

REQ-3: Input field for adding the new password again

REQ-4: Validation text displayed if the passwords don’t match

REQ-5: Validation text displayed if the current password is incorrect

# **Other Nonfunctional Requirements**

## **Performance Requirements**

The web application will use a maximum of 2 gigabytes ( GB ). It’s loading time will have a variation between 0.05 ms and 60s. These performance requirements are deemed right only when the required software has it’s versions up-to-our-date.

## **Safety Requirements**

Not applicable

## **Security Requirements**

For security requirements we will implement the european GDPR regarding usage of personal data as it is sensible information. The application will not disclose user related data to any other part besides the client and the operator. Data may be used in legal conjunctions as it is needed.

Site operators are not to disclose user information to other parties, this includes other site operators. They may do this in case of cross-departament problems.

The user authentication must be done using real information.

Users must be above 18 years old as the EU states for legal sides.

In case of under-age users, their legal tutor must sign and be a part of their account.

## **Software Quality Attributes**

The main software quality attributes of this product are:

* Reliability – it will be reliable as this type of product **MUST** be 100% sure as it will Involve transactions using currencies
* Robustness – the product is very robust as it has to be in this case

Quality Code Rules:

* Make sure that there shouldn’t be any **project warnings**, treat warning as errors
* Use a CSS reset
* Don’t use @import, it is better to include stylesheets into html documents
* Don’t mix HTML with CSS
* Don’t mix CSS with JavaScript
* Use conditional comments
* Place javaScript files at the bottom
* Test while you build to avoid cross-browser issues
* “Smush” your images
* Explain which div you’re closing
* Code consistency
* Code readability
* Disposing of unmanaged resources

## **Business Rules**

The product will have various business rules that include day-to-day transactions, transactions with low priority and high priority.

We reserve the right to set our currencies trading proportions. We won’t deviate too much from official rates and international trading zones.

# **Other Requirements**

*Not applicable*

**Appendix A: Glossary**

*Not applicable*

**Appendix B: Analysis Models**

Not applicable.

**Appendix C: To Be Determined List**

*2.6 User Documentation*