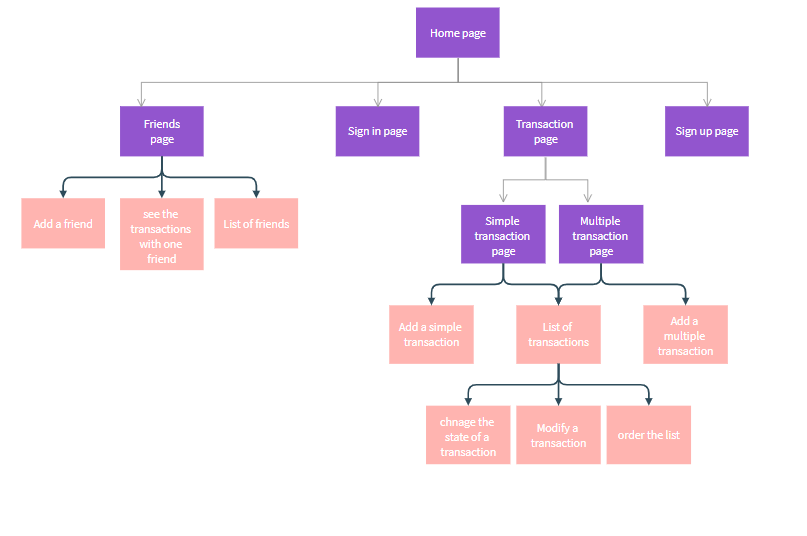
|  |  |
| --- | --- |
|  | **Programmation Web**  ⌘ Documentation template |

**1. Target user profiles**

*Persona profile.*

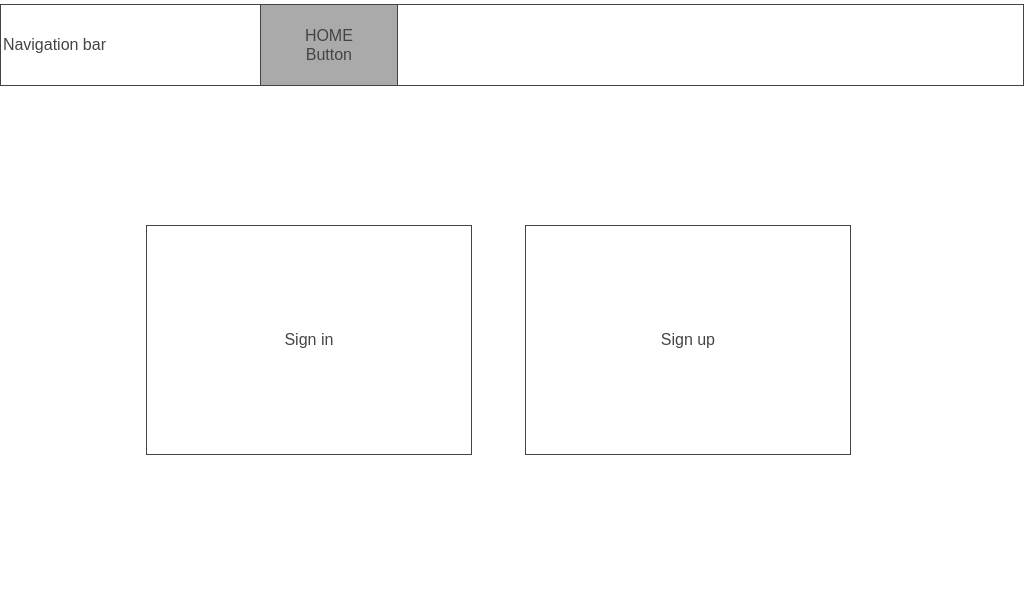
* *Name: John*
* *Age: 22*
* *Commonly used applications: Matlab, Tinder.*
* *Back story: a high-achiever and a Cambridge graduate that specializes in evolutionary algorithms, John moved to Bordeaux for an exciting job opportunity despite barely knowing the local language. John likes to go out and meet people but given his forgetful nature often in a rush leaves his wallet/credit card at home and ends up being unable to pay for his drinks.*
* *Overview:*   
  *- loves maths, numbers & precision,*  
  *- appreciates intuitive design and innovative solutions,*  
  *- is looking for an app that will help him keep track of expenses shared with other people.*

**2. Sitemap**

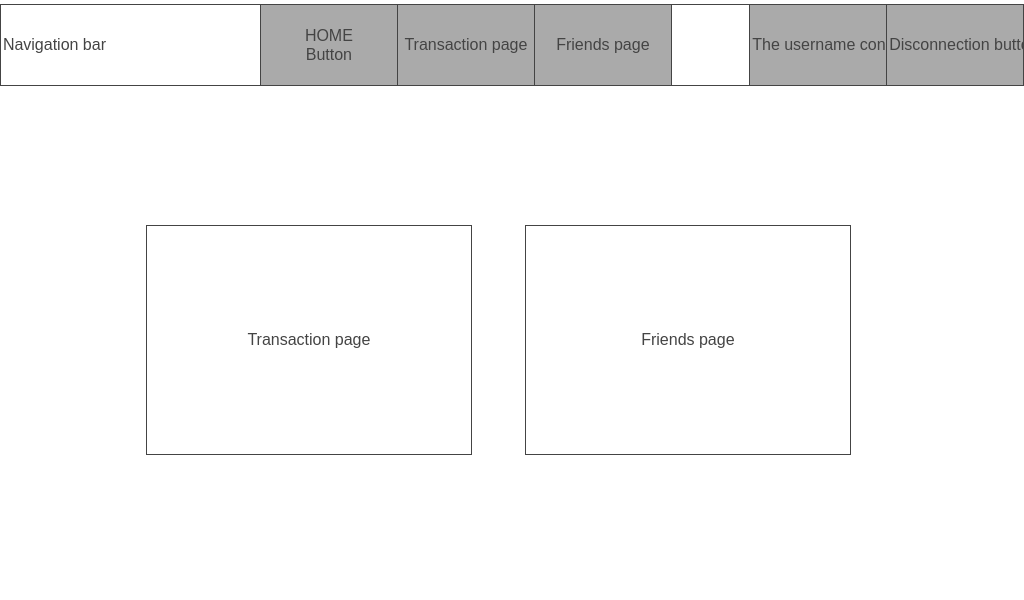


**3. Wireframes**

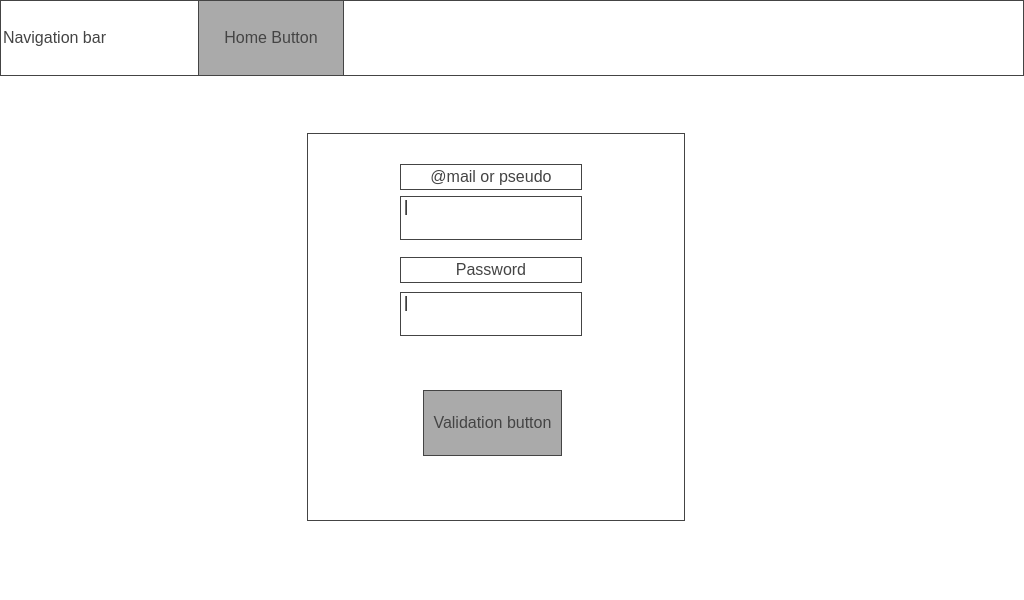
**Home page when the user is disconnected**

****

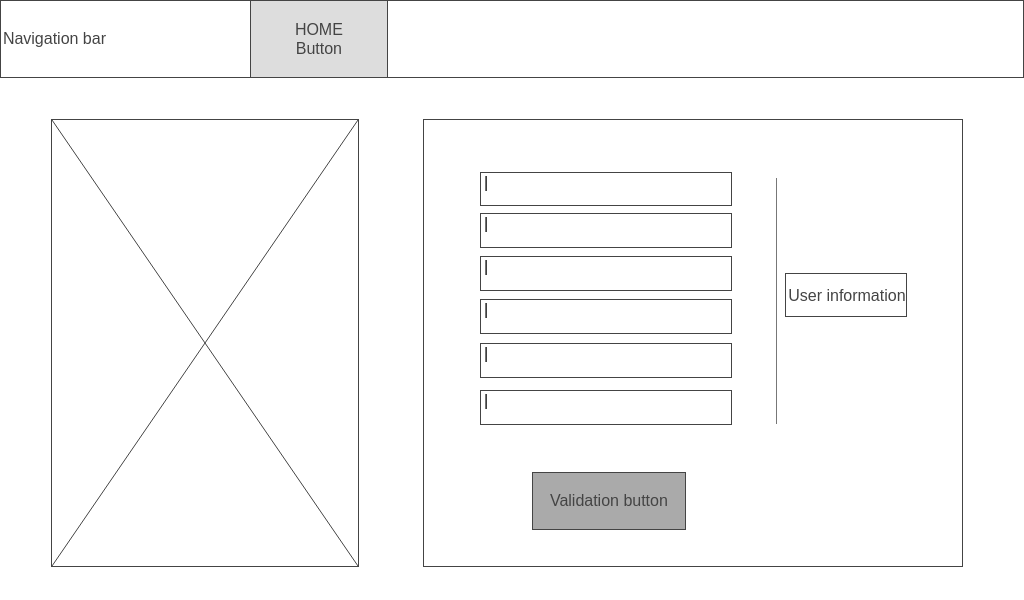
**Home page when the user is connected**

****

**Sign in page**



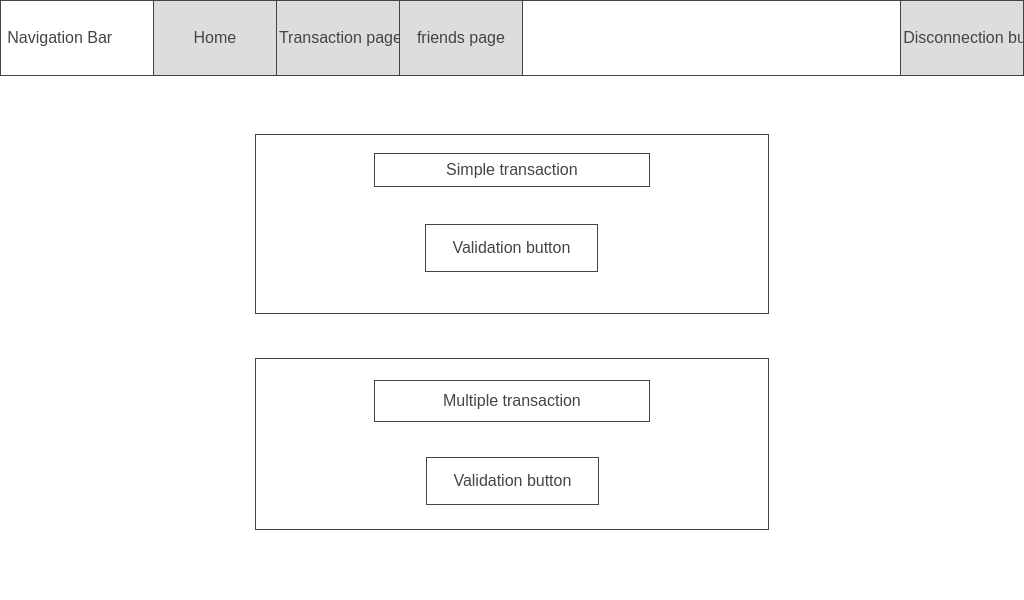
**Sign up page**



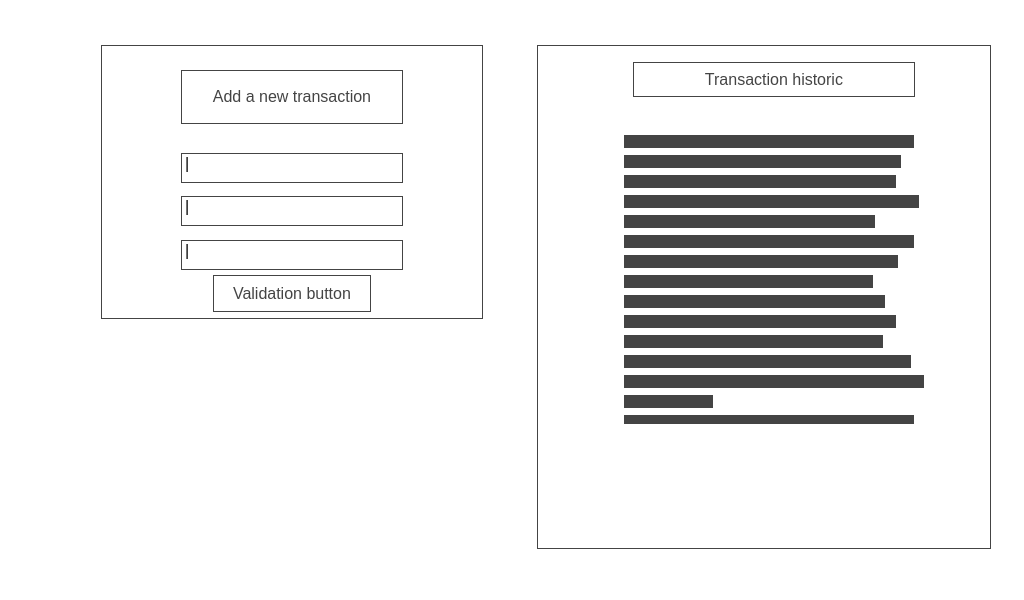
**Friends page**



**Transaction choice page**



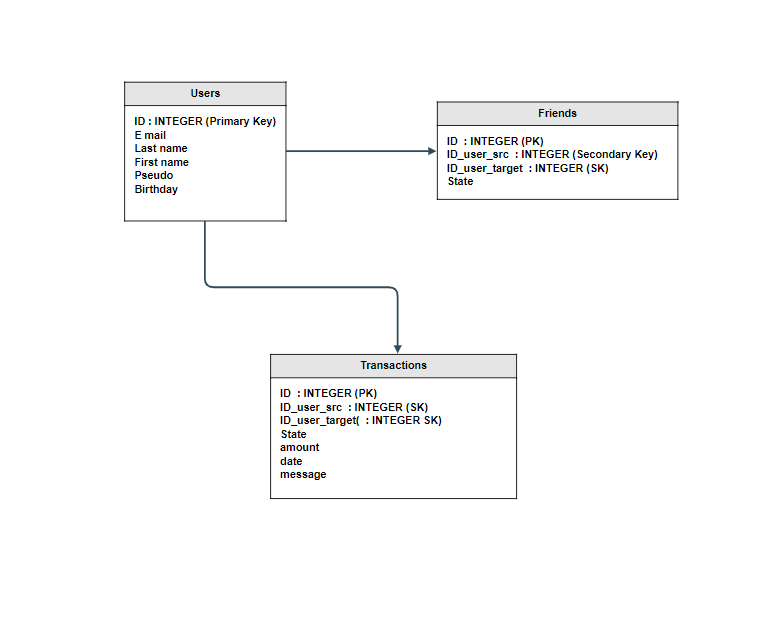
**Simple transaction page**



**Multiple transaction page**



**4. Database model**



**5. System structure - scripts overview**

For every php script created, please specify the following elements:

* php script name and corresponding Wireframe, if applicable,
* a couple of sentences describing the functionality of a given script ,
* HTTP method call and its parameters,
* type of interaction with the database (Insert/Delete/Update), if applicable,
* dependencies with other scripts, if applicable.

Connexion

- sign\_in.php, wireframe 1

- Page de connection à l’application, formulaire simple.

- Méthode POST avec paramètres email, password vers connexion.php

Inscription

- sign\_up.php, wireframe 2

- Page d’inscription à l’application, formulaire demandant toutes les informations

- Méthode POST avec paramètres email, username, firstname, lastname, birthdate, password

vers register.php

- Insert à la table “user”

…...**6. Test strategy**

Test strategy presents the approach to testing taken on a given project. It describes the scope of the tests, the methods used and specification of test cases.

**6A. Test items**

This section presents a list of elements of the system that you intend to test within the scope of the test plan. Essentially, it is a list of what is to be tested - application components.

* Authentication Graphical User Interface
* Friends booklet
* ….

**6B. Test cases**

There exist different types oftests - functional, integration, etc. - you are free to formalize the test cases that you find appropriate to ensure the right level of quality of the project.

The design of the **functional testing** will focus on the adherence of the system to **user requirements**. Integration testing assures that the associated modules communicate correctly, hence tests of inputs and outputs of the system need to be carried out to verify that they conform to the system specifications.

Each test is executed and the result is either a pass or fail.

Test case ID: #1

Test case Name: User Login

Test case Date:

Test case Purpose: To ensure that only valid users are able to log into the application

Test case Function: To verify that only a correct/valid combination of username and password allow a user to log into the application.

Test case Input: An alphanumeric username and an alphanumeric password

Test case expected Output: For a registered user entering the correct combination of username and password, the control panel of the application is shown. For an incorrect combination, the user remains on the login page and an appropriate error message is shown.

**6C. Requirements Traceability Matrix (RTM)**

RTM documents the links between system requirements and test cases that are put in place to ensure an adequate level of testing is achieved. The main purpose of Requirement Traceability Matrix is to see that all requirements have their corresponding test cases so that no functionality is missed out while performing software testing.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **REQUIREMENTS TRACEABILITY MATRIX** | | | | | |
| **Functional Requirement ID#** | **Functional Requirement / Use Case** | **Test Case ID#** | **Test execution date** | **Execution Status** | **Defect description** |
| FR\_1 | As a system user I can add a friend to my friends booklet | TC#001  TC#002 | 20/03/2020 | pass |  |
| FR\_2 | As a system user I can modify existing transactions with a friend | TC#003  TC#004 | 21/03/2020 | fail | The transaction date field can be modified |
|  |  |  |  |  |  |