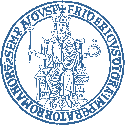
**Event**

**Manager**

**2017**

Web Documentation

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**



Università degli Studi di Napoli Federico II – Corso di laurea di Scienze Informatiche

Software Engineering Project

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Group INGSW2017\_24**

Esposito Alessandro - N86001998

Florino Andrea - N86002104

**Index**

1. **Requirements Analysis** 
   1. Requirements Elicitation
   2. Requirements Analysis
      1. Use Case Diagram
      2. Use Case Description (Cockburn’s template)
   3. Class Diagram (Analysis)
   4. Sequence Diagram
   5. Statechart Diagram
   6. Activity Diagram
2. **Software Design**

2.1. System Architecture

2.2. Design Class Diagram

2.3. Design Sequence Diagram

1. **Testing plan**

3.1. System testing

3.2. JUnit testing

# **Requirements Analysis Document**

1.1. Requirements Elicitation

* **INTERVIEW n.1**

STAKEHOLDER: Prof. Sergio Di Martino

The system provides a website, which has only one language displayed, without any assistance/FAQ area, just some links to social profiles. Such website shows various lists of events, filtered by their category. The user can search for a specific event, artist or a place. Events are shown in a list with their title, image and price, more details can be found once the user clicks on the event. If an event gets delayed or cancelled, all the people who bought a ticket for it must be advised. If an event gets fully booked, it keeps being showed (in a different section) but cannot be bought.

Any customer can browse the website and check the events, but cannot buy any of them until he logs in. If a customer is not registered, he has to do it by clicking the “Sign in” button and filling the fields. Such fields are: Name, Surname, email, Password and Check Password, then a reCaptcha must be checked and the customer can choose if to subscribe to the newsletter. Alternatively, a user can choose to register (and to login) with socials (Twitter, Facebook or Google+). Each user can buy only one ticket per time, then an email with the ticket’s details and QR code is sent to him. This occurs for each ticket bought by any user, without keeping trace of who bought it. The payment is done via external bank service. Once the payment is done, the customer cannot refund any ticket.

* **INTERVIEW n.2**

STAKEHOLDER: Prof. Sergio Di Martino

Mokups have been discussed and accepted by the committer. It is important for the website to be usable and pleasant to see.

1.2 Requirement Analysis

1.2.1 Use case diagram

<<Extends>>

<<Extends>>



Customer

1.2.2 Use Case Description (Cockburn’s template)

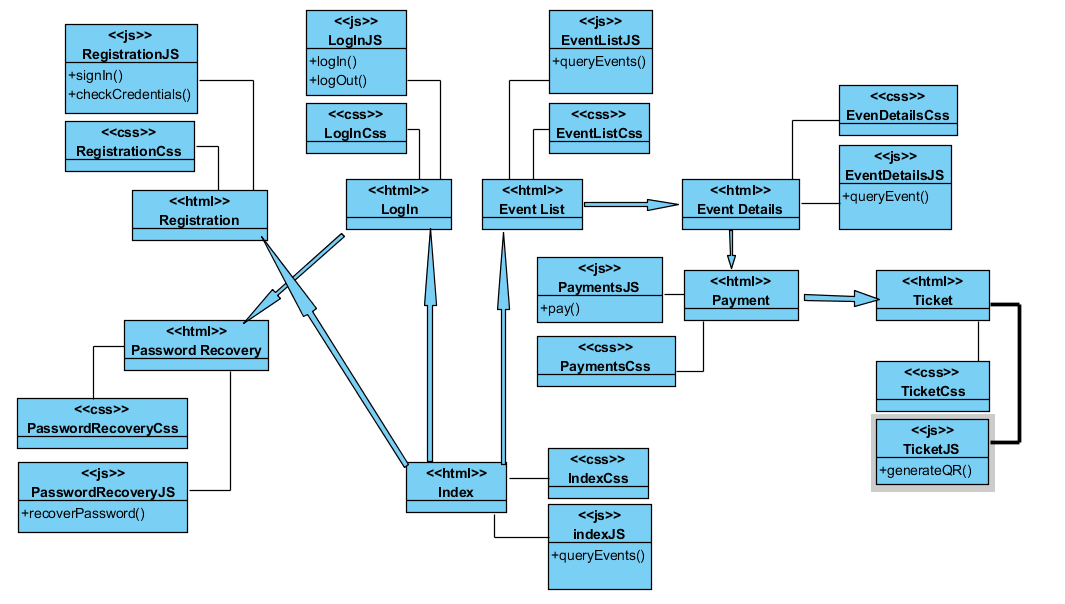
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| #1 | SIGN IN |  |  |  |
| *Goal in context* | The customer signs in successfully | | | | |
| *Success End Condition* | The customer succeeds to sign in | | | | |
| *Fail End Condition* | The costumer fails to sign in | | | | |
| *Primary actor* | Customer | | | | |
| *Trigger* | The customer presses the “sign in” button on the site | | | | |
| DESCRIPTION | **STEP N°** | **CUSTOMER** | **SYSTEM** | |
|  | 1 | Fills in the fields and validates the reCaptcha |  | |
| 2 | Presses confirm button |  | |
| 3 |  | Shows [SIGN IN SUCCESSFUL] page | |
| EXTENSION #1 | **STEP N°** | **CUSTOMER** | **SYSTEM** | |
|  | 1 | Presses social network button |  | |
| 2 |  | Redirects the user to the social network login page | |
| SUBVARIATION #1 | **STEP N°** | **CUSTOMER** | **SYSTEM** | |
|  | 1 | Fails to match password and password confirm fields |  | |
| 2 | Presses sign in |  | |
| 3 |  | Highlights wrong field, Shows a “mismatch user-password” message [REGISTRATION FILED] | |
| SUBVARIATION #2 | **STEP N°** | **CUSTOMER** | **SYSTEM** | |
|  | 1 | Inserts invalid email |  | |
| 2 | Presses sign in |  | |
| 3 |  | Highlights wrong field, Shows the same page and displays a “invalid email” message [REGISTRATION FILED] | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| #2 | LOG IN |  |  |  |
| *Goal in context* | The customer logs in | | | | |
| *Success End Condition* | The customer succeeds to log in | | | | |
| *Fail End Condition* | The costumer fails to log in | | | | |
| *Primary actor* | Customer | | | | |
| *Trigger* | The customer presses the “log in” button on the site | | | | |
| DESCRIPTION | **STEP N°** | **CUSTOMER** | **SYSTEM** | |
|  | 1 | Customer fills in the email and password fields |  | |
| 2 | Presses confirm button. |  | |
| 3 |  | Shows [LOGGED] page | |
| EXTENSION #1 | **STEP N°** | **CUSTOMER** | **SYSTEM** | |
|  | 1 | Presses social networks login button |  | |
|  | 2 |  | The system redirects the user to the social network login page | |
| SUBVARIATION #1 | **STEP N°** | **CUSTOMER** | **SYSTEM** | |
|  | 1 | Fails to match password and email |  | |
| 2 | Presses log in |  | |
|  | 3 |  | Highlights wrong field, Shows the same page but displays a “mismatch user-password” message [REGISTRATION FILED] | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| #3 | SEARCH | |  |  |  |
| *Goal in context* | The customer finds an event | | | | | |
| *Success End Condition* | The customer succeeds to find an event | | | | | |
| *Fail End Condition* | The costumer fails to find an event | | | | | |
| *Primary actor* | Customer | | | | | |
| *Trigger* | The customer clicks on the search bar | | | | | |
| DESCRIPTION | **STEP N°** | **CUSTOMER** | | **SYSTEM** | |
|  | 1 | Write something on the search bar | |  | |
| 2 | Presses enter | |  | |
| 3 |  | | Shows [FIND RESULT] page | |
| 4 | Selects an event from the list | |  | |
| 5 |  | | Shows[EVENT] page | |
| SUBVARIATION #1 | **STEP N°** | **CUSTOMER** | | **SYSTEM** | |
|  | 1 | Writes something on the search bar | |  | |
| 2 | Presses enter | |  | |
| 3 |  | | Shows [FIND RESULT] page | |
| SUBVARIATION #1 | **STEP N°** | **CUSTOMER** | | **SYSTEM** | |
|  | 1 | Writes on the search bar | |  | |
| 2 | Presses enter | |  | |
| 3 |  | | Shows [RESEARCH ERROR] page | |

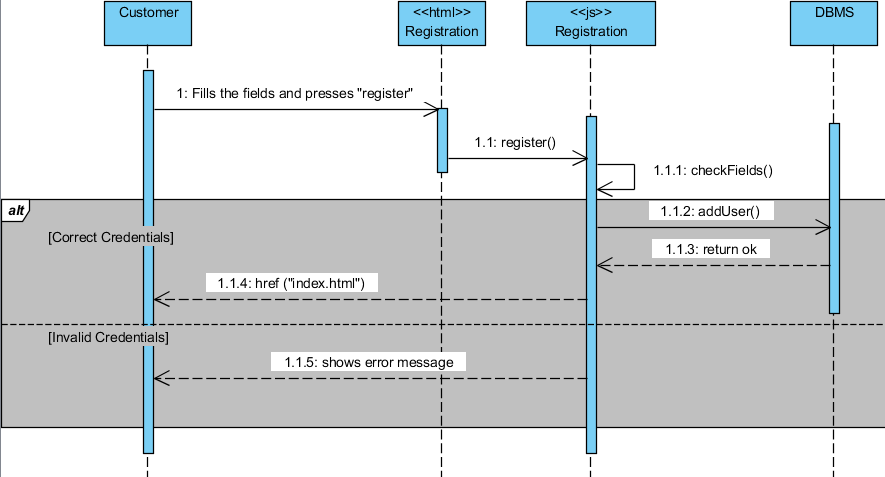
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| #4 | BUY TICKET | |  |  |  |
| *Goal in context* | The customer buys a ticket | | | | | |
| *Success End Condition* | The customer succeeds to buy a ticket | | | | | |
| *Fail End Condition* | The costumer fails to buy a ticket | | | | | |
| *Primary actor* | Customer | | | | | |
| *Trigger* | The customer clicks on “buy” button in [EVENT] page | | | | | |
| DESCRIPTION | **STEP N°** | **CUSTOMER** | | **SYSTEM** | |
|  | 1 | Clicks on “buy” button | |  | |
| 2 |  | | Shows payment page [PURCHASE] | |
| 3 | Fills in correctly all the fields | |  | |
| 4 |  | | Shows the ticket with event’s information and the QR [PURCHASE SUCCESSFUL] | |
| SUBVARIATION #1 | **STEP N°** | **CUSTOMER** | | **SYSTEM** | |
|  | 2 |  | | If the user is not logged in, shows log in page [LOGIN] | |
| SUBVARIATION #2 | **STEP N°** | **CUSTOMER** | | **SYSTEM** | |
|  | 3 | Insert wrong payment data | |  | |
| 4 |  | | Highlights the wrong filled fields, showing the error | |

* 1. Analysis Class Diagram

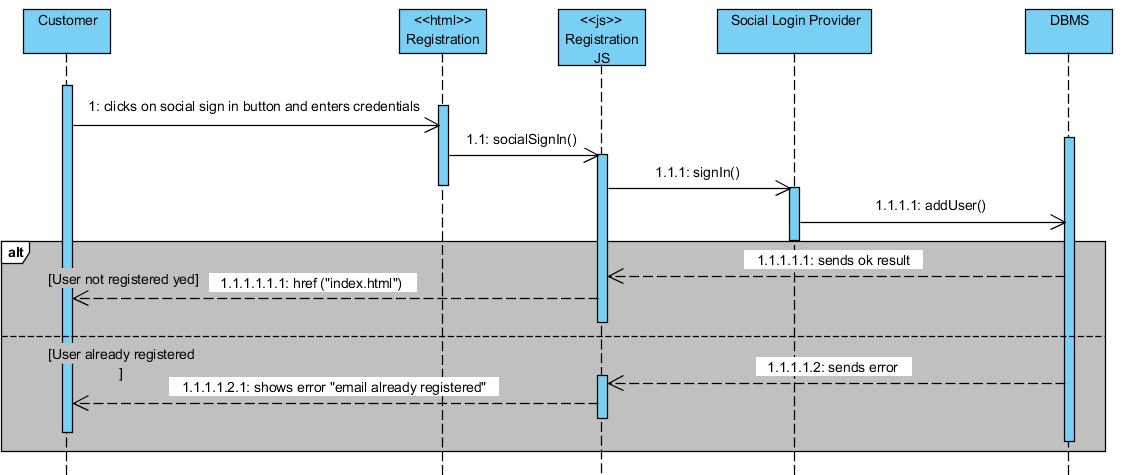


* 1. Analysis Sequence Diagram

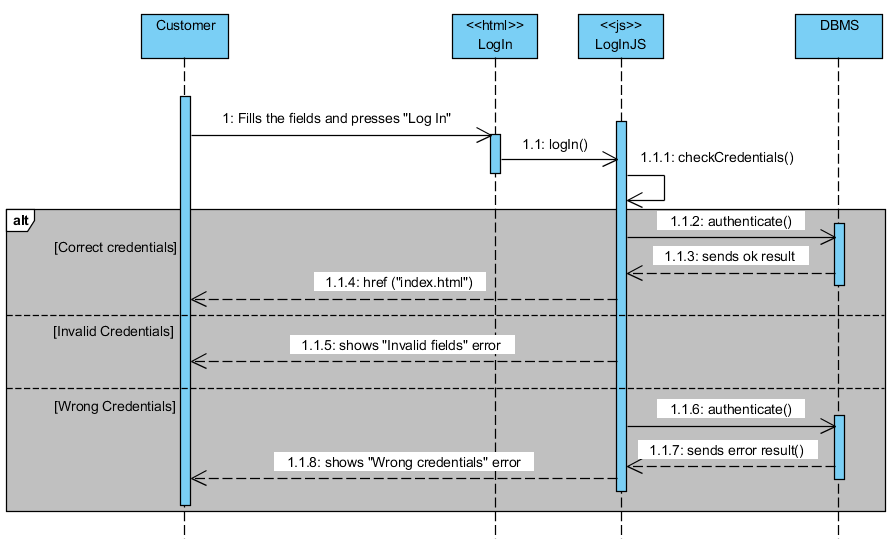
1.4.1. Sign In Sequence Diagram



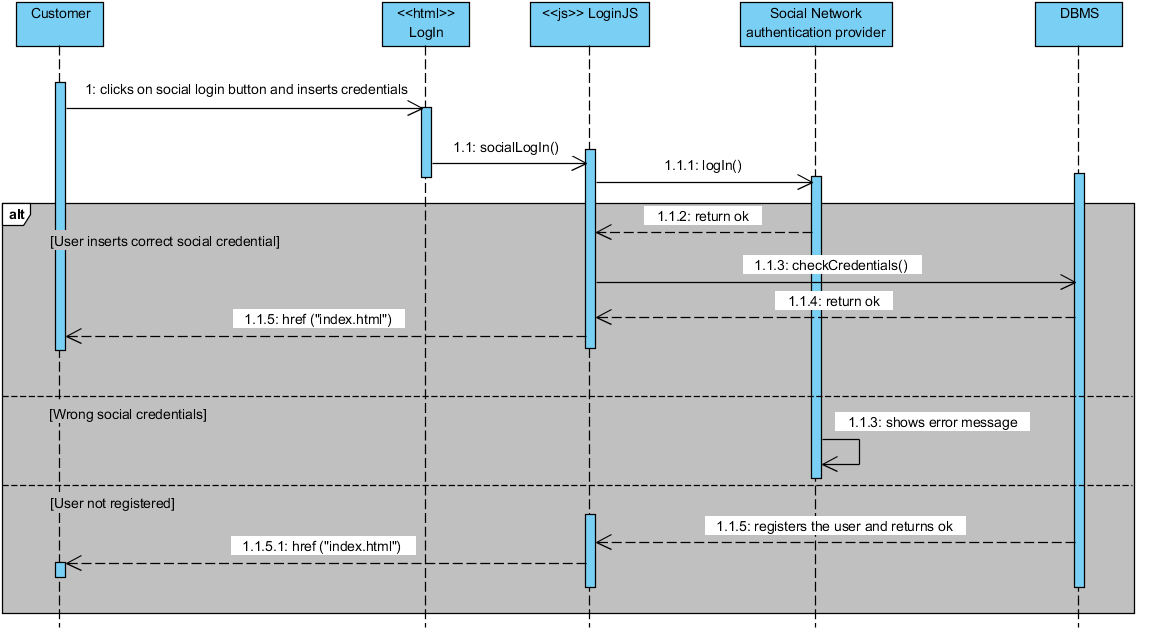
1.4.1.1. Sign In with Social Sequence Diagram



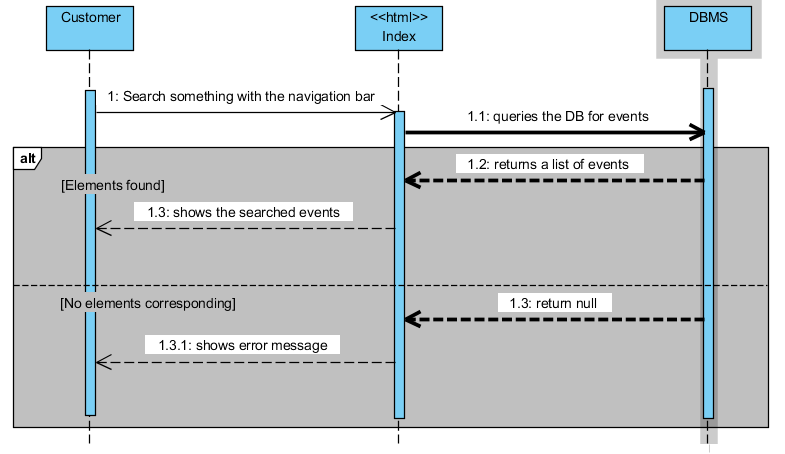
1.4.2. Log In Sequence Diagram



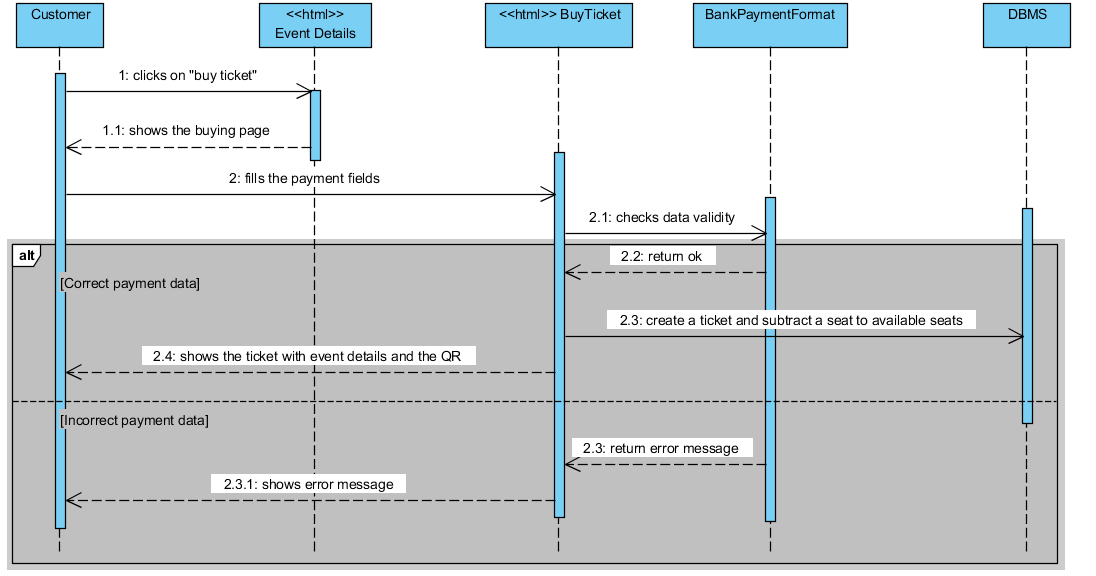
1.4.2.1. Log In with Social Sequence Diagram



1.4.3. Search Sequence Diagram

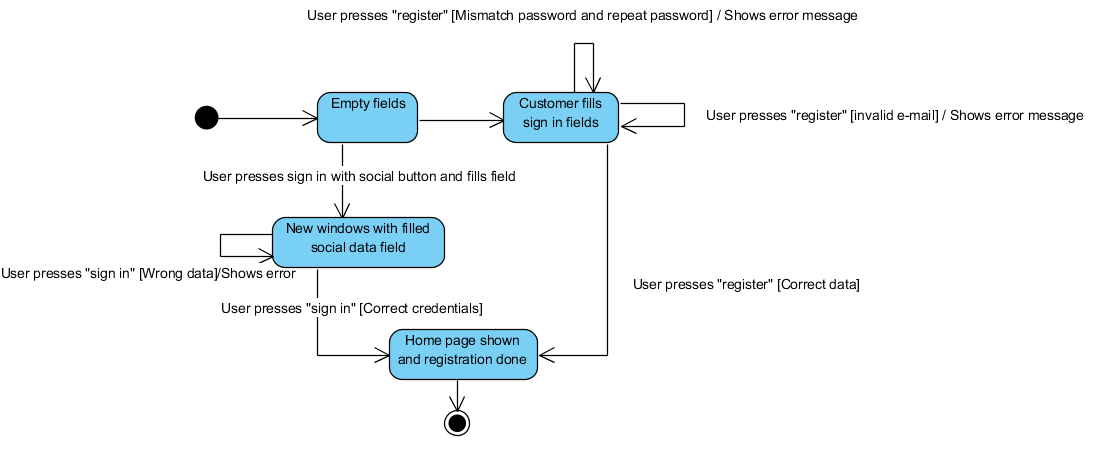


1.4.4. Buy Ticket Sequence Diagram

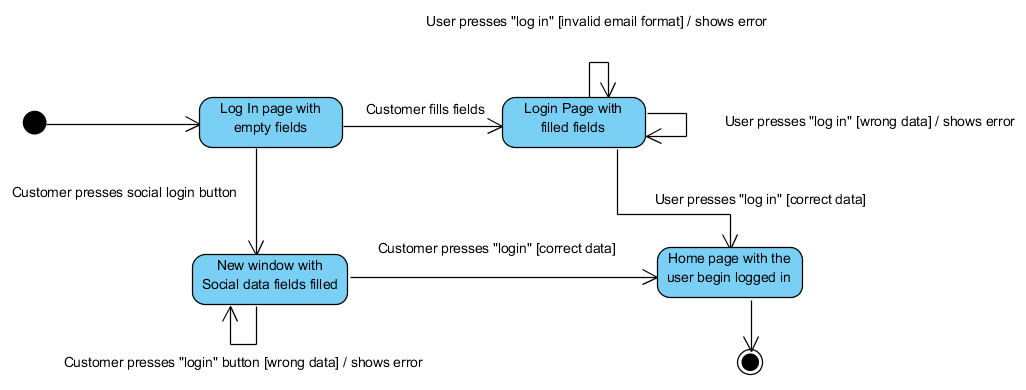


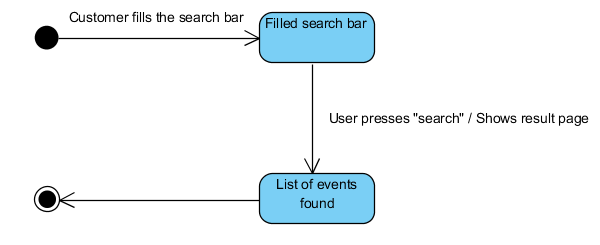
* 1. Analysis Statechart Diagram

1.5.1. Sign in Statechart

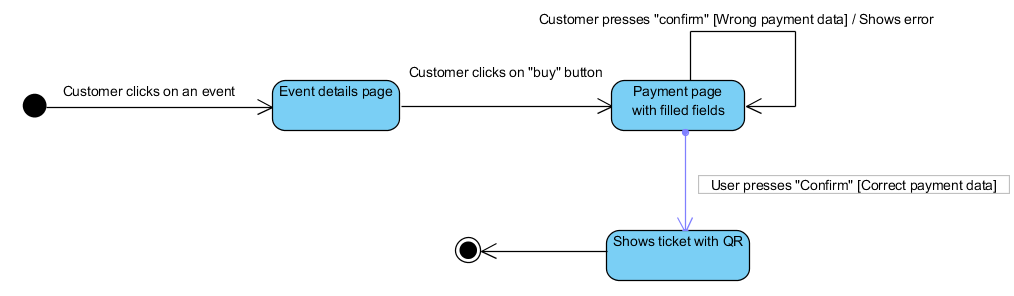


1.5.2. Log in Statechart



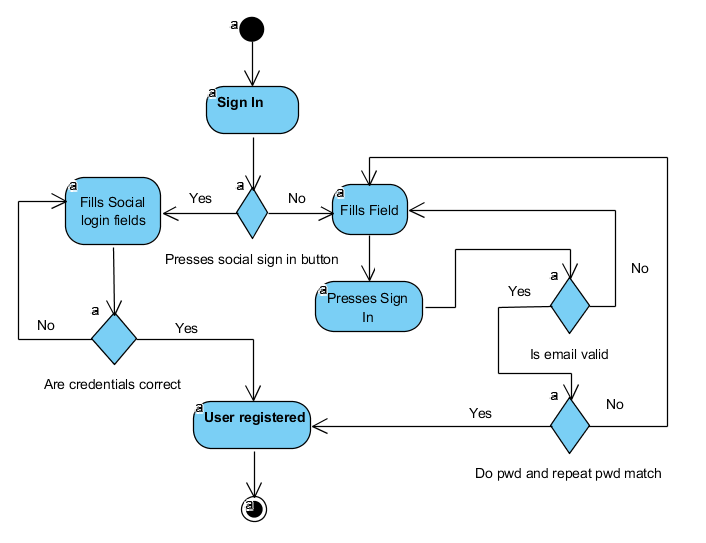
1.5.3. Search Statechart

1.5.4. Buy Ticket

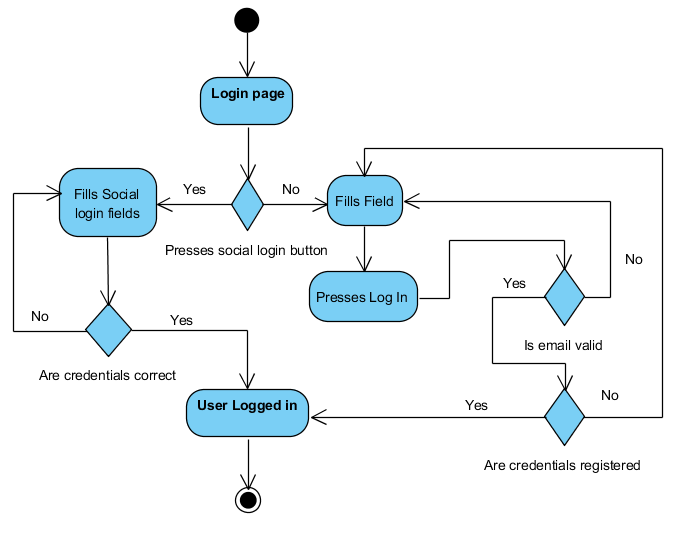


* 1. Activity Diagram

1.6.1. Sign in Activity Diagram



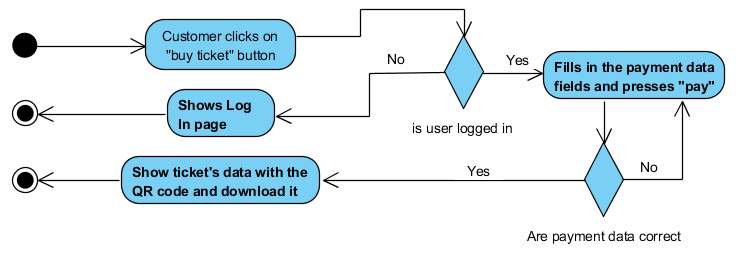
1.6.2. Log In Activity Diagram



1.6.3. Search Activity Diagram



1.6.4. Buy Ticket Activity Diagram

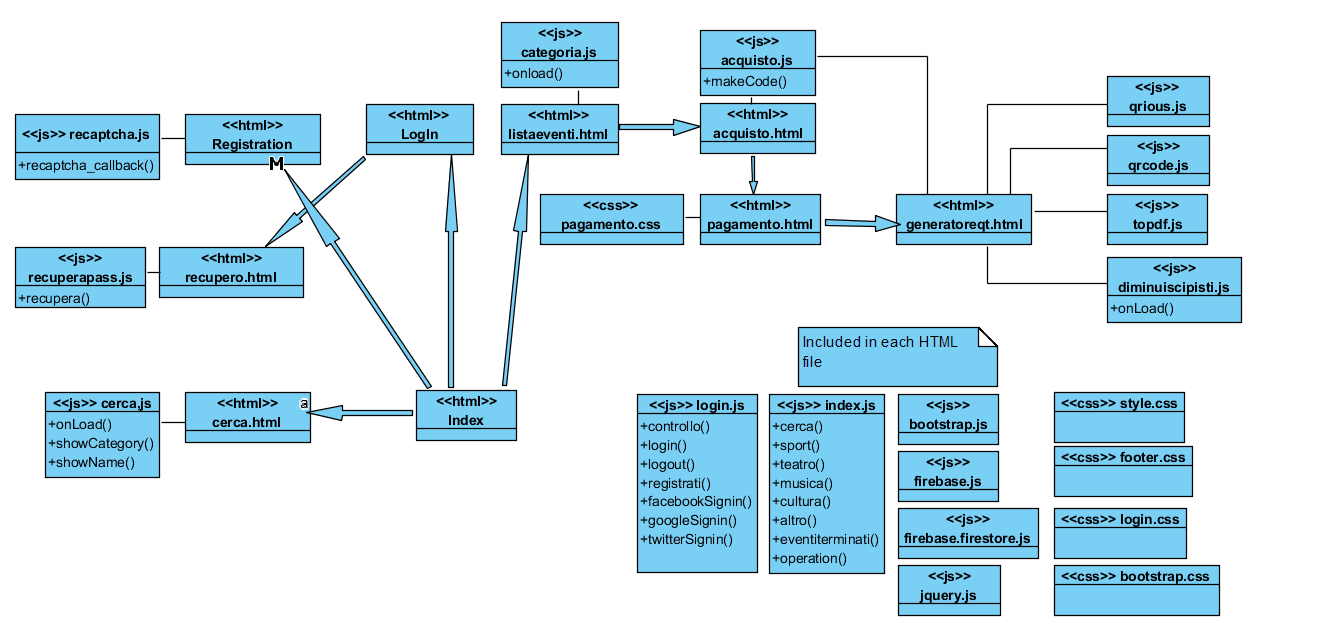


# **System Design Document**

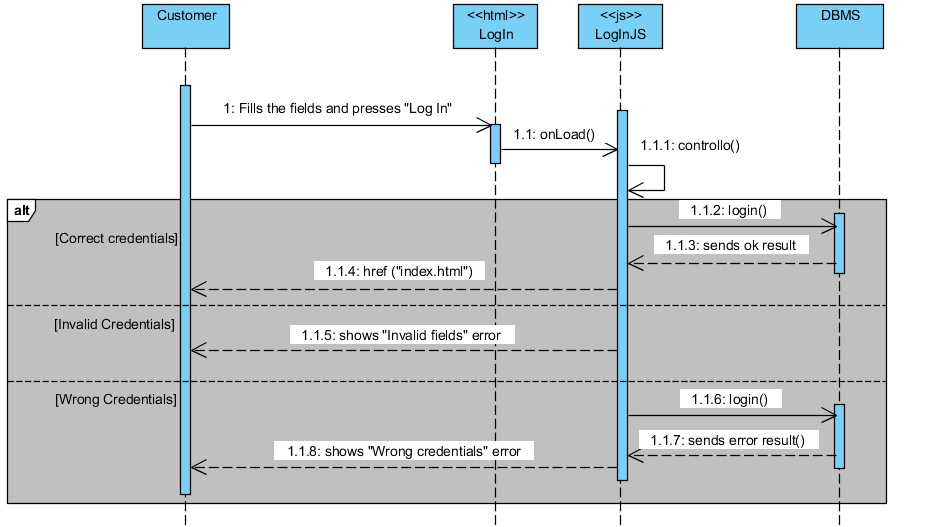
* 1. Architecture Design

This website architecture is focused on assigning all function to .js files, all formatting text to .html files and the graphical elements into .css files. Regarding the Java Script files, each page has to check if the user is logged or not, so that the correct buttons are displayed. All authentication functions are written in the login.js file and all the pages which have specifics functions have their own .js file. This project uses Google Firebase as online database (for demonstration purpose only), the DOCRaptor’s libraries for PDF generation and QRious’ library for QR creation.

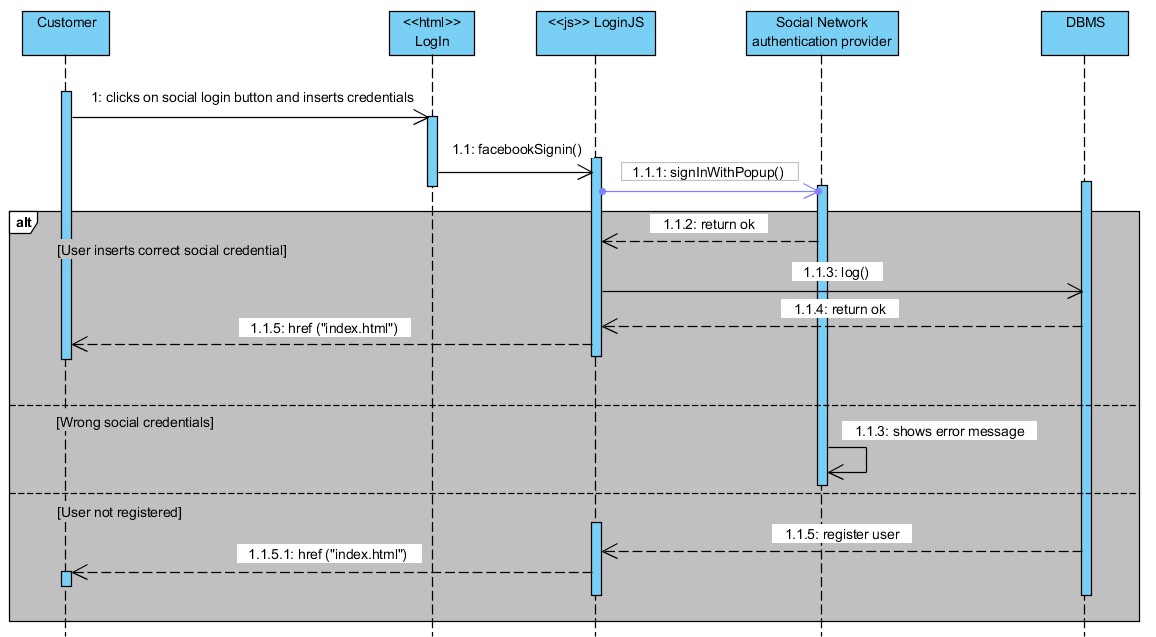
* 1. Class Diagram



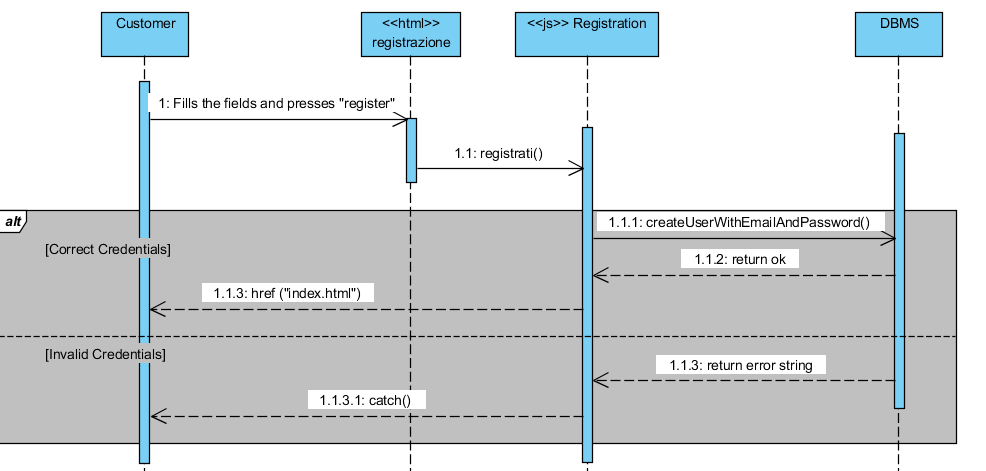
* 1. Sequence Diagram
     1. Log In Sequence Diagram



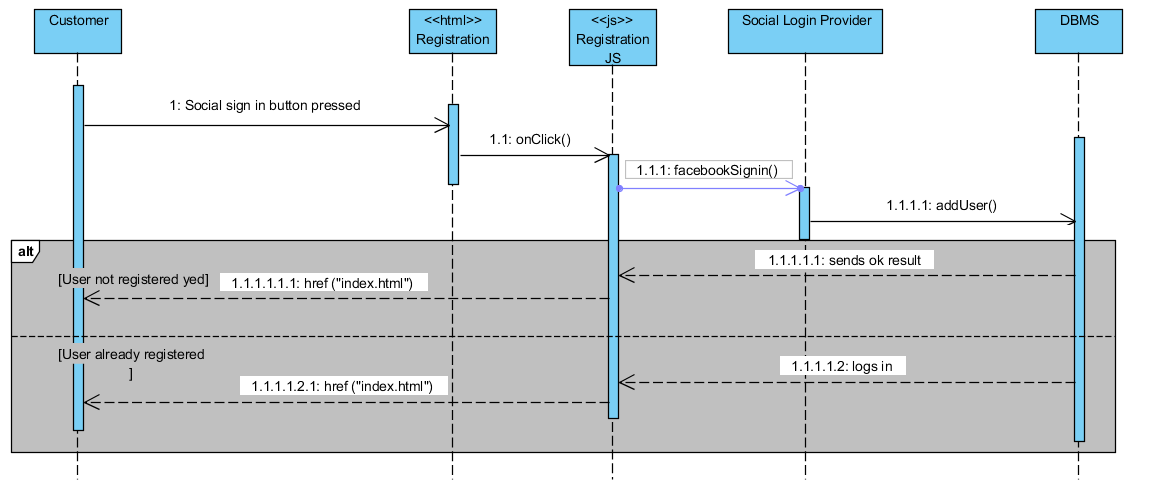
2.2.1.1 Log In with Social Sequence Diagram



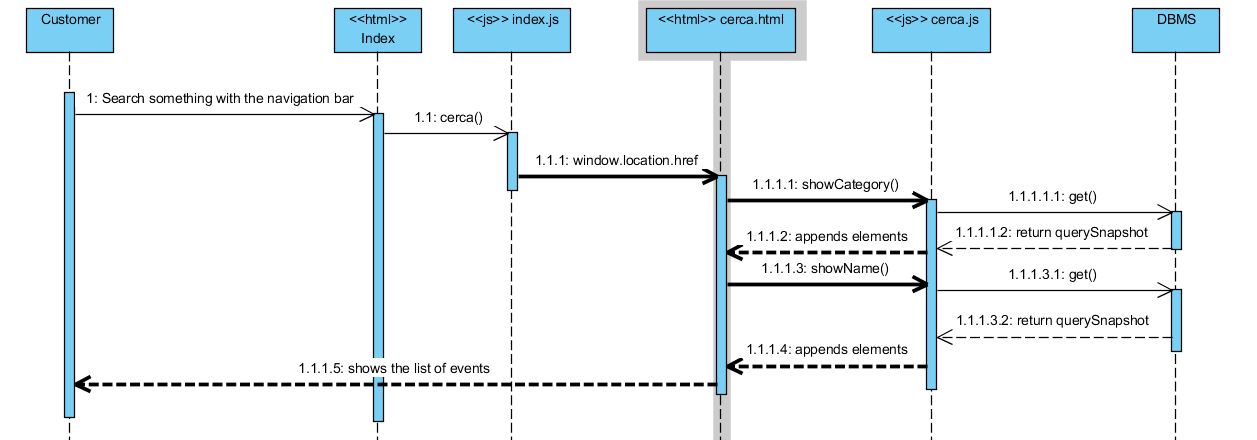
* + 1. Sign in Sequence Diagram



* + - 1. Sign in with Social Sequence Diagram



* + 1. Search Sequence Diagram



* + 1. Buy ticket Sequence Diagram

# **Testing Document**

* 1. System Testing

The following testing plan tests all the user’s input cases so that we can make sure that whatever that input is, the website behaves correctly.

|  |  |  |
| --- | --- | --- |
| Test id | 1 | |
| Test name | Log in test | |
| Test description | Verify that the log in function works properly | |
| Inputs | **Result expected** | **Result achieved** |
| Insert an e-mail and a password which have been registered in the database: [alex19@live.it](mailto:alex19@live.it) and alex123456789 | The user logs correctly in | Correct |
| Insert an e-mail that has been registered in the database but with a wrong password: [alex19@live.it](mailto:alex19@live.it) and a12345 | User does not log in and the page shows an error message “Invalid Credentials” | Correct |
| Insert an e-mail that has been registered in the database : [alex19@live.it](mailto:alex19@live.it) but leave the password field blank | User does not log in and the page shows an error message “Invalid Credentials” | Correct |
| Lave both field blank | User does not log in and the page shows an error message “Invalid Credentials” | Correct |
| Insert a non-valid format email: alex19@live-it | User does not log in and the page shows an error message “Invalid email” | Correct |
| NOTES | The user [alex19@live.it](mailto:alex19@live.it) with the password alex123456789 must be registered in the database | |
|  |  | |
| Test id | 2 | |
| Test name | Sign in | |
| Test description | Verify that the registration function works properly | |
| Inputs | **Result wanted** | **Result achieved** |
| Insert correct data: Andrea Florino, [andrea94@live.it](mailto:andrea94@live.it), andrea12345, andrea12345 and recaptcha | The user registers correctly and gets redirected to the home page | Correct |
| Leave all fields blank and presses register | The register button doesn’t work | Correct |
| Leave all fields blank but validates the recaptcha | The application shows an error message “email address badly formatted” | Correct |
| Writes name, surname and a valid email and validates recaptcha leaving blank the rest of fields | Website shoes error popup “The password must be 6 characters or longer” | Correct |
| Fills all fields except for the “password repeat” one | Website shows error popup “I due campi password non corrispondono” | Correct |
| Fills all fields correctly and checks the newsletter subscription box | User gets registered and subscribed to the newsletter list | Correct |
| Test id | 3 | |
| Test name | Payment test | |
| Test description | Verify that the payment function works properly | |
| Inputs | **Result expected** | **Result achieved** |
| The user inserts correct payment (test) data: 4716108999716531 and 257 for CVV | The payment is successful and a ticket is rendered. | Correct |
| Inserts inexistent card data(“43209482” and “123” for CVV) | Pops up an error message “Wrong owner name” | Correct |
| Test id | 4 | |
| Test name | Search | |
| Test description | Verify that the search function works properly | |
| Inputs | **Result wanted** | **Result achieved** |
| Inserts “sport” and presses search | Shows the result page with the “Juventus – Napoli” event | Correct |
| Inserts “Gli Incredibili 2” | Shows the result page with the “Gli Incredibili 2” event | Correct |
| Inserts “randomquery” | Shows empty page | Correct |