

UNIVERSITÉ IBN ZOHR AGADIR  
FACULTÉ DES SCIENCES



POUR L'OBTENTION DE LA  
LICENCE EN SCIENCES MATHÉMATIQUES ET INFORMATIQUE

---

## SAHEM: Documentation

---

*Encadré par :* Pr. BELAQZIZ SALWA  
*Filière :* Science Mathématique Et Informatique - SMI

*Realisé Par:*  
BOUHSINE TAHA

*Année universitaire :* 2019 - 2020

# Contents

<b>1</b>	<b>Business Objective</b>	<b>1</b>
1.1	What . . . . .	1
1.2	How . . . . .	1
1.3	Why . . . . .	1
<b>2</b>	<b>Technologies</b>	<b>2</b>
2.1	Frontend . . . . .	2
2.1.1	Native vs Webview . . . . .	2
2.1.2	Cross Platform vs Platform Independent . . . . .	2
2.2	Backend . . . . .	2
2.2.1	NodeJs vs PHP vs J2EE . . . . .	2
<b>3</b>	<b>Problem Space</b>	<b>4</b>
3.1	Actors . . . . .	5
3.1.1	A1-User . . . . .	5
3.1.2	A2-Visitor . . . . .	6
3.1.3	A3-Registred User . . . . .	7
3.1.4	A5-Administrator: . . . . .	8
3.1.5	A6-SahemSystem . . . . .	8
3.1.6	A7-Payment Interface . . . . .	9
3.1.7	A8-Project . . . . .	10
3.1.8	A9-Post . . . . .	11
3.2	Use Cases Simplified . . . . .	12
3.2.1	UC1-Register . . . . .	12
3.2.2	UC2-Login . . . . .	12
3.2.3	UC3-ForgotPassword . . . . .	12

3.2.4	UC4-ShareToSocialMedia . . . . .	13
3.2.5	UC5-CreateProject . . . . .	13
3.2.6	UC6-EditProject . . . . .	13
3.2.7	UC7-DeleteProject . . . . .	13
3.2.8	UC8-FundProject . . . . .	14
3.2.9	UC10-VoteOnProject . . . . .	14
3.2.10	UC11-CommentOnProject . . . . .	14
3.2.11	UC12-FollowUser . . . . .	15
3.2.12	UC13-GetAllFollowedUserPosts . . . . .	15
3.2.13	UC14-PostToActivityFeed . . . . .	15
3.2.14	UC15-CommentOnUserPost . . . . .	16
3.2.15	UC16-VoteOnUserPost . . . . .	16
3.3	Use Cases . . . . .	16
3.3.1	UC1-Register . . . . .	16
3.3.2	UC2-Login . . . . .	18
3.3.3	UC3-ForgotPassword . . . . .	19

# List of Figures

# List of Tables

# **Chapter 1**

## **Business Objective**

### **1.1 What**

Provide a platform for crowdfunding startups, Ideas and projects.

### **1.2 How**

Developing and implementing a crossplatform web application.

### **1.3 Why**

Resulting to an international reach for Ideas owners and inovators to share and start with their projects.

# Chapter 2

## Technologies

### 2.1 Frontend

#### 2.1.1 Native vs Webview

Native

Webview

#### 2.1.2 Cross Platform vs Platform Independent

Cross Platform

Pros

Cons

Platform Independent

Pros

Cons

### 2.2 Backend

#### 2.2.1 NodeJs vs PHP vs J2EE

NodeJs

Pros

Cons

**PHP**

**Pros**

**Cons**

**J2EE**

**Pros**

**Cons**



## **Chapter 3**

# **Problem Space**

### **Introduction**

## **3.1 Actors**

### **3.1.1 A1-User**

#### **Actor Type and Stereotype:**

direct, abstract.

#### **Actor Description:**

This is an abstract actor representing all the types of users interacting with SAHEM System. He will be having the ability to see a list of projects with the permission to only see the description of the projects displayed in our web application. Can Register, Login, Forgot Password. Can share links to social media.

#### **Actor Relationship:**

This Actor is specialized into A2-Visitor, A3-Registred, A5-Administrator, A6-Reviewer.

#### **Interface Specification:**

#### **Author and History:**

Taha Bouhsine.

#### **Reference Material:**

### **3.1.2 A2-Visitor**

#### **Actor Type and Stereotype:**

secondary, direct, Concrete.

#### **Actor Description:**

Derived from A1-User, this actor will have a more specialized way of listing projects, will only have the ability to look to the project name, owner, description.

#### **Actor Relationship:**

This Actor is derived from A1-User.

#### **Interface Specification:**

#### **Author and History:**

#### **Reference Material:**

### **3.1.3 A3-Registered User**

#### **Actor Type and Stereotype:**

primary, direct, Concrete.

#### **Actor Description:**

This is the main benifitor from our system, derived from the A1-User, this actor will be intereacting with the system by adding a project, editing, deleting owned projects, funding a project.

List all the projects and their details, the same as for the A2-visitor but can have the access to more information about the project.

Up voting a project that he thinks should be funded by others.

Have a wallet that will be linked to the Payment interface.

can comment, contribute.

Can follow other registred users, follow the progress of projects.

Have an activity feed that will be showing all the posts from other users in the system.

#### **Actor Relationship:**

This Actor is derived from A1-User.

#### **Interface Specification:**

#### **Author and History:**

Taha Boushine

#### **Reference Material:**

### **3.1.4 A5-Administrator:**

#### **Actor Type and Stereotype:**

secondary, direct, Concrete.

#### **Actor Description:**

This is user have the most permission in our system, have the access to everything and can do everything, derived from A3-Registered User. Accept, Delete, Ban Users etc..

#### **Actor Relationship:**

This Actor is derived from A3-User.

#### **Interface Specification:**

#### **Author and History:**

BOUHISNE Taha

#### **Reference Material:**

### **3.1.5 A6-SahemSystem**

#### **Actor Type and Stereotype:**

#### **Actor Description:**

#### **Actor Relationship:**

#### **Interface Specification:**

#### **Author and History:**

#### **Reference Material:**

### **3.1.6 A7-Payment Interface**

#### **Actor Type and Stereotype:**

#### **Actor Description:**

#### **Actor Relationship:**

The interface that will handle every aspects of the user payments, either recharging his account or withdrawing budget after his project it's goal.

#### **Interface Specification:**

#### **Author and History:**

BOUHSINE Taha

#### **Reference Material:**

### **3.1.7 A8-Project**

#### **Actor Type and Stereotype:**

#### **Actor Description:**

A project that can be owned by one A3-RegisteredUser, and can be either funded or voted on by other users of the same type as A3-RegisteredUser. can be funded by a registered user.

have a coment section.

have a vote counter.

have a contributor list that contains all the users that funded the project.

#### **Actor Relationship:**

Owned by the A3-RegisteredUser that can have one or many projects.

#### **Interface Specification:**

#### **Author and History:**

BOUHSINE Taha

#### **Reference Material:**

### 3.1.8 A9-Post

#### Actor Type and Stereotype:

«Business» «RegisteredUser»

#### Actor Description:

A post owned by one A3-RegisteredUser, up or down voted or commented on by other users of the same type as A3-RegisteredUser. can be funded by a registred user.

have a coment section.

have a vote counter.

have a contributor list that contains all the users that funded the project.

#### Actor Relationship:

Owned by the A3-RegisteredUser that can have one or many Posts.

#### Interface Specification:

#### Author and History:

BOUHSINE Taha

#### Reference Material:



## **3.2 Use Cases Simplified**

### **3.2.1 UC1-Register**

**Use Case Thumbnail:**

UC1-Register

**Actors:**

A1-User

**Use Case Description:**

This is use case will handle the user registering request.

### **3.2.2 UC2-Login**

**Use Case Thumbnail:**

UC2-Login

**Actors:**

A1-Visitor

**Use Case Description:**

This is use case will handle the user log in request.

### **3.2.3 UC3-ForgotPassword**

**Use Case Thumbnail:**

UC3-ForgotPassword

**Actors:**

A1-User

**Use Case Description:**

This is use case will handle the user's resetting password request.

### **3.2.4 UC4-ShareToSocialMedia**

#### **Use Case Thumbnail:**

UC4-ShareToSocialMedia

#### **Actors:**

A1-User

#### **Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

### **3.2.5 UC5-CreateProject**

#### **Use Case Thumbnail:**

UC5-CreateProject

#### **Actors:**

A3-RegistredUser

#### **Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

### **3.2.6 UC6-EditProject**

#### **Use Case Thumbnail:**

UC6-EditProject

#### **Actors:**

A3-RegistredUser

#### **Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

### **3.2.7 UC7-DeleteProject**

#### **Use Case Thumbnail:**

UC7-DeleteProject

**Actors:**

A3-RegistredUser

**Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

**3.2.8 UC8-FundProject****Use Case Thumbnail:**

UC8-FundProject

**Actors:**

A3-RegistredUser

**Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

**3.2.9 UC10-VoteOnProject****Use Case Thumbnail:**

UC10-VoteOnProject

**Actors:**

A3-RegistredUser

**Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

**3.2.10 UC11-CommentOnProject****Use Case Thumbnail:**

UC11-CommentOnProject

**Actors:**

A3-RegistredUser

**Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

**3.2.11 UC12-FollowUser****Use Case Thumbnail:**

UC12-FollowUser

**Actors:**

A3-RegistredUser

**Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

**3.2.12 UC13-GetAllFollowedUserPosts****Use Case Thumbnail:**

UC13-PostToActivityFeed

**Actors:**

A3-RegistredUser

**Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

**3.2.13 UC14-PostToActivityFeed****Use Case Thumbnail:**

UC13-PostToActivityFeed

**Actors:**

A3-RegistredUser

**Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

### **3.2.14 UC15-CommentOnUserPost**

#### **Use Case Thumbnail:**

UC15-CommentOnUserPost

#### **Actors:**

A3-RegistredUser

#### **Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

### **3.2.15 UC16-VoteOnUserPost**

#### **Use Case Thumbnail:**

UC16-VoteOnUserPost

#### **Actors:**

A3-RegistredUser

#### **Use Case Description:**

This is use case will handle the act of a user sharing a project or user profile link.

## **3.3 Use Cases**

### **3.3.1 UC1-Register**

#### **Use Case Thumbnail:**

UC1-Register

#### **Use Case Description:**

A request from a A2-Visitor to register in the application.

#### **Use Case Stereotype and Package:**

« Business » - « User »

#### **Preconditions:**

Not registred

**Postcondition:**

Registered

**Actors:**

A2-Visitor

**Use Case Relationships:**

Associated with actors A2-Visitor,

**Basic Flow:**

1-The visitor will click on the register button. 2-Fill the personal information form. 2.1- the system will check if the information are valid.

3-Fill the payments method form. 3.1- the system will check if the information are valid. 4-Confirm Email. 5-System will save the user's information. 6-System will confirm the users's registration. 7-redirect the User to the sign in form.

**Alternative Flow:**

< -Informations are not valid/complete. >

**Exceptions:**

None

**Constraints:**

None

**User Interface Specification:****Metrics:**

medium

**Priority:**

High

**Status:**

Minor

**Author and History:**

Taha Bouhsine

**Reference Material:****3.3.2 UC2-Login****Use Case Thumbnail:**

UC2-Login

**Use Case Description:**

A request from a A2-Visitor to register in the application.

**Use Case Stereotype and Package:**

« Business » - « Visitor »

**Preconditions:**

Logged out

**Postcondition:**

Logged in.

**Actors:**

A2-Visitor

**Use Case Relationships:**

Associated with actors A2-Visitor,

**Basic Flow:**

1-The visitor will click on the Login button. 2-Fill the username and password form. 2.1- the system will check if the information are valid. 3-System will log the user's Session. 7-redirect the User to the main page.

**Alternative Flow:**

< -Informations are not valid/complete. >

**Exceptions:**

None

**Constraints:**

None

**User Interface Specification:****Metrics:**

simple

**Priority:**

High

**Status:**

Minor

**Author and History:**

Taha Bouhsine

**Reference Material:****3.3.3 UC3-ForgotPassword****Use Case Thumbnail:**

UC3-ForgotPassword

**Use Case Description:**

A request from a A2-Visitor to reset the passwod.

**Use Case Stereotype and Package:**

« Business » - « Visitor »

**Preconditions:**

Old Password



**Postcondition:**

New Password

**Actors:**

A2-Visitor

**Use Case Relationships:**

Associated with actors A2-Visitor,

**Basic Flow:**

1-The visitor will click on the Forgot Password button. 2-Fill the Email and UserName form. 2.1- the system will check if the information are valid. 2.2- The System will send an reset email to the User Email. 3- the user will click on the link. 7- the user will fill the form with the new email. 7.1- check if the password is valid. 8- the user will be redirected to the login page.

**Alternative Flow:**

< -Informations are not valid/complete. >

**Exceptions:**

None

**Constraints:**

None

**User Interface Specification:****Metrics:**

simple

**Priority:**

High

**Status:**

Minor

**Author and History:**

Taha Bouhsine

**Reference Material:**