

UNIVERSITÉ IBN ZOHR AGADIR
FACULTÉ DES SCIENCES



SAHEM: Documentation

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

Realisé Par:
BOUHSINE TAHA

Année universitaire : 2019 - 2020

Contents

1	Business Objective	1
1.1	What	1
1.2	How	1
1.3	Why	1
2	Technologies	2
2.1	Tools	2
2.1.1	Github	2
2.1.2	PlantUml	2
2.1.3	Latex	2
2.1.4	Vs Code	2
2.1.5	Photoshop	2
2.1.6	MongoDB Compass	2
2.2	Frontend	2
2.2.1	Native vs Webview	2
2.2.2	Cross Platform vs Platform Independent	2
2.2.3	Angular	3
2.2.4	NativeScript	3
2.3	Backend	3
2.3.1	NodeJs vs PHP vs J2EE	3
2.3.2	ExpressJs	3
3	Problem Space	4
3.1	Actors	5
3.1.1	A1-User	5
3.1.2	A2-Visitor	6

3.1.3	A3-Registered User	7
3.1.4	A5-Administrator:	8
3.1.5	A6-SahemSystem	9
3.1.6	A7-Payment Interface	10
3.1.7	A8-Project	11
3.1.8	A9-Post	12
3.1.9	A10-Wallet	13
3.2	Use Cases Simplified	14
3.2.1	UC1-Register	14
3.2.2	UC2-Login	14
3.2.3	UC3-ForgotPassword	14
3.2.4	UC4-ShareToSocialMedia	15
3.2.5	UC5-CreateProject	15
3.2.6	UC6-EditProject	15
3.2.7	UC7-DeleteProject	16
3.2.8	UC8-FundProject	16
3.2.9	UC10-VoteOnProject	16
3.2.10	UC11-CommentOnProject	17
3.2.11	UC12-FollowUser	17
3.2.12	UC13-ViewAcivityFeed	17
3.2.13	UC14-CreatePost	18
3.2.14	UC15-EditPost	18
3.2.15	UC16-DeletePost	18
3.2.16	UC17-CommentOnPost	18
3.2.17	UC18-EditComment	19
3.2.18	UC19-DeleteComment	19
3.2.19	UC20-VoteOnUserPost	19
3.3	Use Cases	20
3.3.1	UC1-Register	20
3.3.2	UC2-Login	21
3.3.3	UC3-ForgotPassword	23

List of Figures

List of Tables

Chapter 1

Business Objective

Introduction

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 innovators to share and start with their projects.

Chapter 2

Technologies

2.1 Tools

2.1.1 Github

2.1.2 PlantUml

2.1.3 Latex

2.1.4 Vs Code

2.1.5 Photoshop

2.1.6 MongoDB Compass

2.2 Frontend

2.2.1 Native vs Webview

Native

Webview

2.2.2 Cross Platform vs Platform Independent

Cross Platform

Pros

Cons

Platform Independent

Pros

Cons

2.2.3 Angular

2.2.4 NativeScript

2.3 Backend

2.3.1 NodeJs vs PHP vs J2EE

NodeJs

Pros

Cons

PHP

Pros

Cons

J2EE

Pros

Cons

2.3.2 ExpressJs

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-Registered, A5-Administrator, A6-Reviewer.

Interface Specification:

Author and History:

BOUSHINE Taha

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 benefactor from our system, derived from the A1-User, this actor will be interacting 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 registered 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:

BOUHSINE Taha

Reference Material:

3.1.4 A5-Administrator:

Actor Type and Stereotype:

secondary, direct, Concrete.

Actor Description:

This is user have the most permissions 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:

BOUHSNE Taha

Reference Material:

3.1.5 A6-SahemSystem

Actor Type and Stereotype:

« System » « Business »

Actor Description:

A non human actor that represents our system.

Actor Relationship:

Interface Specification:

Author and History:

BOUHSINE Taha

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 comment section.

have a vote counter.

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

have a progress feed that contains zero or many posts.

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» « Post »

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. have a comment section.

have a vote counter.

have a contributors 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:

3.1.9 A10-Wallet

Actor Type and Stereotype:

«Business» «RegisteredUser»

Actor Description:

A wallet that holds money.

Actor Relationship:

An A3-RegisteredUser have a wallet.

A project have a wallet.

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

A2-Visitor

A3-SahemSystem

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-User

A2-Visitor

A3-SahemSystem

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

A2-Visitor

A3-SahemSystem

Use Case Description:

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

3.2.4 UC4-ShareToSocialMedia**Use Case Thumbnail:**

UC4-ShareToSocialMedia

Actors:

A1-User

A2-Visitor

A3-SahemSystem

Use Case Description:

This 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-RegisteredUser

A3-SahemSystem

Use Case Description:

This use case will handle the act of a registered user creating a project.

3.2.6 UC6-EditProject**Use Case Thumbnail:**

UC6-EditProject

Actors:

A3-RegisteredUser

A3-SahemSystem

Use Case Description:

This use case will handle the act of a registered user editing his existing project.

3.2.7 UC7-DeleteProject**Use Case Thumbnail:**

UC7-DeleteProject

Actors:

A3-RegisteredUser

A3-SahemSystem

Use Case Description:

This use case will handle the act of a registered user deleting his project after entering the edit menu.

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

UC8-FundProject

Actors:

A3-RegisteredUser

A3-SahemSystem

Use Case Description:

This use case will handle the act of a registered user funding a project.

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

UC10-VoteOnProject

Actors:

A3-RegisteredUser

A3-SahemSystem

Use Case Description:

This use case will handle the act of a registered user voting on a project.

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

UC11-CommentOnProject

Actors:

A3-RegisteredUser

A3-SahemSystem

Use Case Description:

This use case will handle the act of a registered user voting on a project

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

UC12-FollowUser

Actors:

A3-RegisteredUser

A3-SahemSystem

Use Case Description:

This use case will handle the act of a registered user following another registered user.

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

UC13-ViewActivityFeed

Actors:

A3-RegisteredUser

Use Case Description:

This is use case will handle the act of a registered user getting all the posts published by other registered users or projects.

3.2.13 UC14-CreatePost

Use Case Thumbnail:

UC14-CreatePost

Actors:

A3-RegisteredUser

Use Case Description:

This is use case will handle the act of a registered user publishing a post.

3.2.14 UC15-EditPost

Use Case Thumbnail:

UC15-EditPost

Actors:

A3-RegisteredUser

Use Case Description:

This is use case will handle the act of a registered user editing his own post.

3.2.15 UC16-DeletePost

Use Case Thumbnail:

UC16-DeletePost

Actors:

A3-RegisteredUser

Use Case Description:

This is use case will handle the act of a registered user deleting his post.

3.2.16 UC17-CommentOnPost

Use Case Thumbnail:

UC17-CommentOnPost

Actors:

A3-RegisteredUser

Use Case Description:

This is use case will handle the act of a registered user Commenting on a post.

3.2.17 UC18-EditComment**Use Case Thumbnail:**

UC18-EditComment

Actors:

A3-RegisteredUser

Use Case Description:

This is use case will handle the act of a registered user editing a comment.

3.2.18 UC19-DeleteComment**Use Case Thumbnail:**

UC19-DeleteComment

Actors:

A3-RegisteredUser

Use Case Description:

This is use case will handle the act of a registered user deleting a comment.

3.2.19 UC20-VoteOnUserPost**Use Case Thumbnail:**

UC16-VoteOnUserPost

Actors:

A3-RegisteredUser

Use Case Description:

This use case will handle the act of a registered user voting on a post.

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 registered

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:

< -information's are not valid/complete. >

Exceptions:

None

Constraints:

None

User Interface Specification:**Metrics:**

medium

Priority:

High

Status:

Minor

Author and History:

BOUHSINE Taha

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:

< -information's are not valid/complete. >

Exceptions:

None

Constraints:

None

User Interface Specification:**Metrics:**

simple

Priority:

High

Status:

Minor

Author and History:

BOUHSINE Taha

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

UC3-ForgotPassword

Use Case Description:

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

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:

< -information's are not valid/complete. >

Exceptions:

None

Constraints:

None

User Interface Specification:**Metrics:**

simple

Priority:

High

Status:

Minor

Author and History:

BOUHSINE Taha

Reference Material: