

MINI-WEB PROJECT REPORT

presented in order to obtain the
MINI-WEB SUBJECT NOTE
speciality : software engineering

Bloggy

by YASSINE NEMRI
MOHAMMED YASSINE MATHLOUTHI
MOHAMED AZIZ KBAIER



Academic supervisor: Mrs Sara Essafi
Le:

Signature:

Dedications

We dedicate this report to our teacher, Madame Essafi, for her unwavering support and guidance throughout this period. Her dedication to ensuring that we have the necessary resources and tools to complete this report has been truly inspiring. Her commitment to our education has enabled us to navigate the challenges we faced and develop the skills needed to create a comprehensive report. Thank you for your encouragement, your patience, and for always believing in us. Your influence has made a significant impact on our academic journey, and we will be forever grateful.

Bloggy team

Contents

General introduction	2
1.1 Introduction :	2
1.1.1 Identification of functional needs	2
1.1.2 Identification of non-functional needs	2
1.1.3 Identification of actors	3
1.1.4 Use case diagram	4
1.1.5 Product Backlog	5
1.2 Working environment:	5
5subsection.1.2.1	
1.2.2 Scrum tools	5
1.2.3 Conclusion	8
2 Sprint 0	9
2.1 Introduction	9
2.2 Sprint 0 backlog identification	9
2.3 Sprint refinement 0	9
2.3.1 "Authenticate" use case refinement:	10
2.3.2 "create account" use case refinement	11
2.3.3 "manage blog" use case refinement	12
2.3.4 "Manage account" use case refinement	13
2.3.5 "Add blog" use case refinement	14
2.3.6 "edit blog" use case refinement	15
2.4 Sprint 0 conception	16
2.4.1 "Authenticate" use case conception:	16
2.4.2 "Create account" use case conception:	17
2.4.3 "manage account" use case conception:	18
2.4.4 "edit blog" use case conception:	20
2.4.5 "add blog" use case conception:	22
2.4.6 Global class diagram of sprint 0:	23
2.5 Completion of sprint 0	23
2.5.1 interface "login"	23
2.5.2 interface "create account"	24
2.5.3 interface "manage blog"	24
2.5.4 interface "manage account"	25
2.5.5 interface "add blog"	25
2.6 Conclusion	26

3 Sprint 1	27
3.1 Introduction	27
3.2 Sprint 1 backlog identification	27
3.3 Sprint Refinement 1	27
3.3.1 Refinement of "read blog" use case	28
3.3.2 Refinement of "delete blog" use case	29
3.3.3 Refinement of "delete user" use case	30
3.3.4 Refinement of "search for blog" use case	31
3.4 Conception of sprint 1	31
3.4.1 conception of use case "read blog"	31
3.4.2 conception of use case "delete blog"	33
3.4.3 conception of use case "delete user"	34
3.4.4 conception of use case "search for a blog"	35
3.5 completion of sprint 1	37
3.5.1 use case "read blog" interface	37
3.5.2 use case "delete blog" interface	37
3.5.3 use case "delete user" interface	38
3.5.4 use case "search for a blog" interface	38
3.6 Conclusion	39
general conclusion	40
Bibliographie	41

List of Figures

1	website logo	1
1.2	use case diagram	4
2.3	use case authenticate	10
2.4	use case create account	11
2.5	use case manage blog	12
2.6	manage account use case diagram	13
2.7	add blog use case diagram	14
2.8	edit blog use case diagram	15
2.9	s'authenticate class diagram	16
2.10	s'authenticate sequence diagram	17
2.11	Create account diagram class	17
2.12	create account sequence diagram	18
2.13	manage account class diagram	18
2.14	manage account sequence diagram	19
2.15	edit blog class diagram	20
2.16	edit blog sequence diagram	21
2.17	add blog class diagram	22
2.18	add blog class diagram	22
2.19	sprint 0 global class diagram	23
2.20	login interface	23
2.21	create account interface	24
2.22	home page interface	24
2.23	manage account interface	25
2.24	add blog interface	25
3.25	Case diagram read blog	28
3.26	Case diagram delete blog	29
3.27	Case diagram delete user	30
3.28	Case diagram search blog	31
3.29	read blog class diagram	32
3.30	read blog sequence diagram	32
3.31	delete blog class diagram	33
3.32	delete blog sequence diagram	33
3.33	delete user class diagram	34
3.34	delete user sequence diagram	34
3.35	search blog class diagram	35
3.36	search blog sequence diagram	35
3.37	Deployment diagram	36
3.38	read blog interface	37
3.39	delete blog interface	37

3.40 delete user interface	38
3.41 search for a blog	38

List of Tables

1.1	product Backlog	5
1.2	tools	6
1.3	product Backlog	7
2.4	product Backlog of release 1	9
2.5	“Authenticate” use case refinement	10
2.6	“create account” use case refinement	11
2.7	“manage blog” use case refinement	12
2.8	“manage account” use case refinement	13
2.9	“add blog” use case refinement	14
2.10	“edit blog” use case refinement	15
3.11	product Backlog of release 2	27
3.12	“read blog” use case refinement	28
3.13	“delete blog” use case refinement	29
3.14	“delete user ” use case refinement	30
3.15	“search for a blog ” use case refinement	31

General introduction

Blogs are a popular and effective way to reach out to a specific audience, whether it's a group of like-minded people, potential customers, or the general public. Blog topics can range from personal stories and experiences to industry-specific news and analysis. Some blogs are dedicated to a single topic, while others cover a variety of subjects, so we came up with the idea of hosting a website about blogs so that users can read any specific topic he needs, so our website is about blogs of various topics such as sports cars mechanics etc. Of course, you must be able to create your own account in order to post on whatever topic you desire. For our project's modeling, we used the Agile "Scrum" methodology. This document summarizes our efforts, which are divided into four chapters in order to achieve a reliable and satisfactory application:

- In Chapter 01, titled "Project Framework," we will present a welcome system and conduct a study of the current situation by proposing a solution.
- Chapter 02, titled "Specification of needs," which will specify the actors and their needs, as well as the product backlog.
- Chapter 03, "release 1: Blog management," provides a detailed description of the first release 1 to explore.
- Chapter 04, "release 2: Blog monitoring," provides a detailed description of the second release to explore.



Figure 1: website logo

chapter 1

Specification of needs

1.1 Introduction :

Users can create an account, authenticate, view and like blogs, manage their accounts, post blogs, and manage their activities. These actions allow users to consume and produce content on the platform in an interactive way.

1.1.1 Identification of functional needs

In order for a website to function properly and offer value to its visitors, it must have certain fundamental features and capabilities. Ultimately, a website's success depends heavily on its functional requirements. Using our system, you can:

- Create an account
- authenticate
- see blogs
- like blogs
- manage accounts
- post blogs
- manage activities.

1.1.2 Identification of non-functional needs

Non-functional needs are those features and functions that are essential for the user experience overall but are not directly related to the functionality of the website. In order for the website to win the trust of users and gradually increase engagement and loyalty, it must be accessible to all users, safe, and easy to use [c1]. The following conditions must be satisfied for this:

- **Website responsiveness:** The website should load rapidly and react immediately to user input. Users who are frustrated by a slow website may decide to leave.
- **Accessibility :** This can involve using high contrast colors, offering text transcripts for audio and video information, and providing alternate text for images.

- **Mobile responsiveness:** With more users accessing the internet on mobile devices, it's crucial for the website to be mobile-friendly and responsive to different screen sizes.
- **Security:** The website should be safe from cyberattacks and safeguard user information. Two-factor authentication and the use of SSL certificates are examples of this.

1.1.3 Identification of actors

An actor is a natural or legal person who participates in or is affected by the action or project at hand. As a result, we must begin by clearly specifying which action or sequence of actions we seek to determine who and what the actors are. Our platform includes three actors who interact directly with the system.

Administrator: Administrators manage user blogs and even users by verifying their blogs and managing any errors that occur on the website.

Guest: A guest is a user who can read blogs without registering or contributing.

User: A user is someone who has an account on the website and has the ability to create, edit, or delete blogs, as well as read them.

1.1.4 Use case diagram

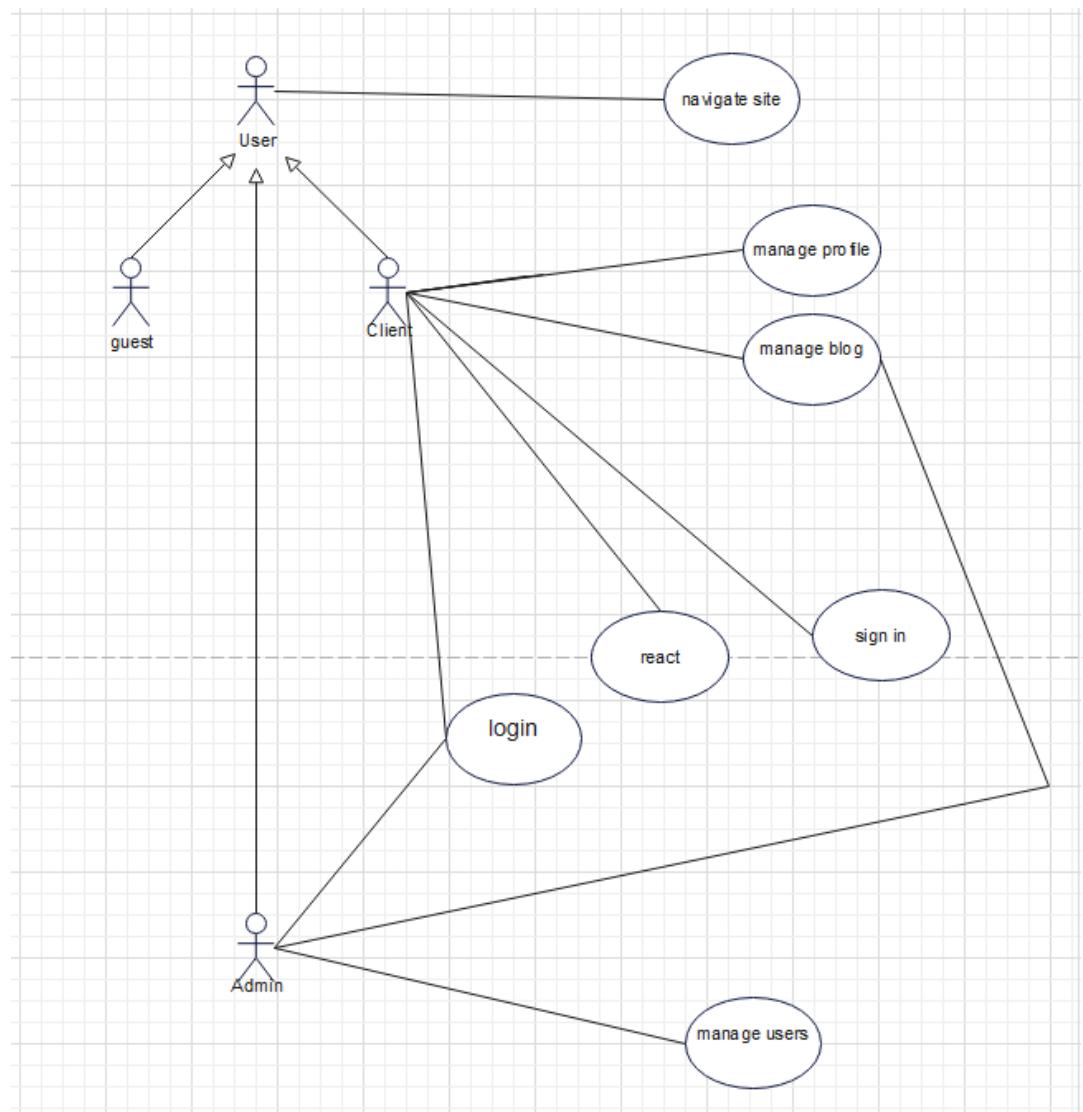


Figure 1.2: use case diagram

1.1.5 Product Backlog

Product Backlog Item	Priority	Estimation	Planned release
As user I can authenticate	1	Medium	Sprint 0
As user I can login	1	Medium	Sprint 0
As admin I can authenticate	1	Medium	Sprint 0
As admin I can login	1	Medium	Sprint 0
As user I can manage blog	1	Medium	Sprint 0
As user I can manage my account	1	Hard	Sprint 0
As user I can navigate blog	2	Medium	Sprint 1
As guest I can react	2	Medium	Sprint 1
As user I can search for a blog	2	Hard	Sprint 1
As guest I can navigate blog	2	Medium	Sprint 1
As admin I can delete blog	2	Medium	Sprint 1
As admin I can delete users accounts	2	Hard	Sprint 1

Table 1.1: product Backlog

1.2 Working environment:

1.2.1 Scrum method [0]

Scrum is a work methodology for developing and maintaining complex products in order to solve complex and changing problems while delivering products of the highest possible value. Scrum employs an incremental approach to optimize predictability and risk control.

Sprint

Sprint planning entails identifying priorities that the team believes they can accomplish during the sprint. Sprint income is earned at the end of each sprint in which the team development demonstrates completed features. The product increment is potentially deliverable, and the next sprint's implementation can be anticipated. By taking stock of the previous sprint, the sprint retrospective allows you to improve the development process of the subsequent sprints.

1.2.2 Scrum tools

Product	Description
	Adobe AI (Adobe Illustrator) is a vector graphics editor used for creating logos, illustrations, and other scalable graphics. It uses mathematical equations and geometric shapes to create precise, scalable vector graphics and includes a variety of tools and features for design and manipulation. [1].
	Adobe XD (Adobe Experience Design) is a design tool for creating and prototyping user interfaces for websites and mobile applications. It offers various features such as wireframing, vector design tools, and collaboration capabilities. [2]
	Adobe PS (Adobe Photoshop) is a digital image editing software used for color correction, image retouching, and compositing. It supports a wide range of file formats and is widely used by graphic designers, photographers, and artists. [3]
	Angular is a popular open-source web application framework developed by Google. It is used for building dynamic single-page applications and supports the development of complex, scalable web applications. Angular provides developers with a variety of features and tools for creating responsive, interactive web applications, including templates, data binding, dependency injection, and component-based architecture. Additionally, Angular has a large and active community of developers and provides extensive documentation and support resources. [4]
	JavaScript (JS) is a programming language used primarily for web development. It is a high-level, dynamically typed language that is often used to create interactive and responsive user interfaces for websites and web applications. JavaScript is executed on the client-side, meaning it is run in the user's web browser, and is used for tasks such as form validation, creating animations, and interacting with server-side APIs. Additionally, JavaScript can be used for server-side programming with Node.js, making it a versatile language for both front-end and back-end development. [5]

Table 1.2: tools

Product	Description
	<p>HTML5 is the latest version of the Hypertext Markup Language (HTML), which is used to create and structure content for the web. It includes new features and improvements that make it easier to create and manipulate multimedia content, handle input and output, and provide better support for accessibility and semantic markup. Some of the new features in HTML5 include audio and video tags, canvas element for dynamic graphics, geolocation API, and improved support for forms and input types. [6]</p>
	<p>CSS (Cascading Style Sheets) is a styling language used for describing the presentation of a web page or document written in HTML. It provides a way to separate the presentation of content from its structure and semantic meaning. With CSS, you can define the look and feel of a web page by specifying the fonts, colors, layout, and other visual aspects of the content. [7]</p>
	<p>XAMPP is a free and open-source software package that provides a local development environment for web applications. The name XAMPP stands for cross-platform, Apache, MySQL, PHP, and Perl, which are the components included in the package. [8]</p>
	<p>EdrawMax is a versatile and easy-to-use diagramming software that allows users to create a wide range of diagrams, charts, and visual representations. It is used by professionals and students in various fields such as business, engineering, education, and more. [9]</p>
	<p>GitHub is a web-based platform that provides hosting for software development and version control using Git. It offers a wide range of tools and features that make it easier for developers to collaborate on software projects, track changes to code, and manage their workflow. [10]</p>
	<p>Git is a distributed version control system that is used to manage and track changes to code and other files over time. It is widely used in software development to enable collaboration among team members, manage code changes, and track progress of a project. [11]</p>

Table 1.3: product Backlog

1.2.3 Conclusion

To summarize, this blog website is used to enlighten people's minds and clear things up by answering users' questions, so people can now enjoy reading on this platform.

chapter 2

Sprint 0

2.1 Introduction

In this chapter, we will discuss the project's first release, which is data management of a blog that can detail user cases of priorities. This sprint's research includes refinement, design, and production.

2.2 Sprint 0 backlog identification

This first release backlog, which we present below, contains a list of backlog items that must be completed in sprint 0:

Product Backlog Item	Priority	Estimation	Planned release
As user I can authenticate	1	Medium	Sprint 0
As user I can create account	1	Medium	Sprint 0
As admin I can authenticate	1	Medium	Sprint 0
As admin I can create account	1	Medium	Sprint 0
As user I can manage blog	1	Medium	Sprint 0
As user I can manage my account	1	Hard	Sprint 0
As user I can add blog	1	Medium	Sprint 0

Table 2.4: product Backlog of release 1

2.3 Sprint refinement 0

In this section, we are interested in the following use cases:

- Authenticate
- create account
- edit blog
- manage account
- Add blog

2.3.1 “Authenticate” use case refinement:

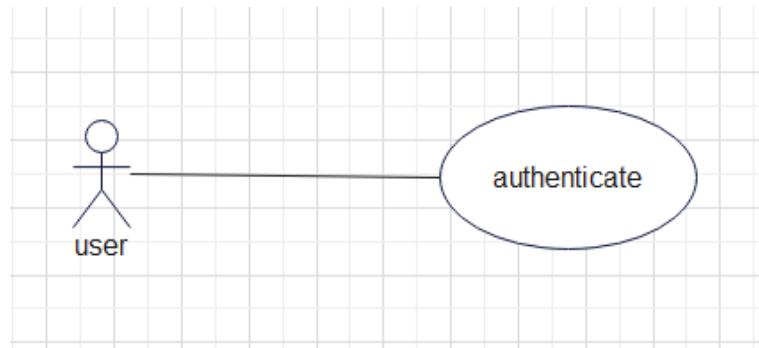


Figure 2.3: use case authenticate

user case	authenticate
actor	user
pre-condition	Systeme running
Post-condition	The actor authenticated and can access to the application.
Main scenario	<ul style="list-style-type: none"> - The system displays the authentication interface. -The actor enters login and password. -The actor clicks on the “Connect” button. -The system checks the login and password combination - The system displays the home page according to the profile of the user
Exception	The system displays an error message if the data is incorrect.

Table 2.5: “Authenticate” use case refinement

2.3.2 "create account" use case refinement

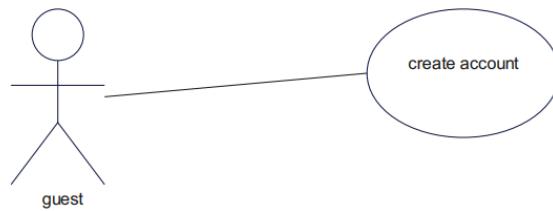


Figure 2.4: use case create account

user case	create account
actor	user
pre-condition	System running
Post-condition	creation was successful
Main scenario	<ul style="list-style-type: none"> - The system displays the registration interface. -The actors entered his personal coordinates. -The actors click on the “Register” button. - The system verifies the data. - The system displays a message of the success of registration.
Exception	The system displays an error message if the data is incorrect.

Table 2.6: “create account” use case refinement

2.3.3 "manage blog" use case refinement

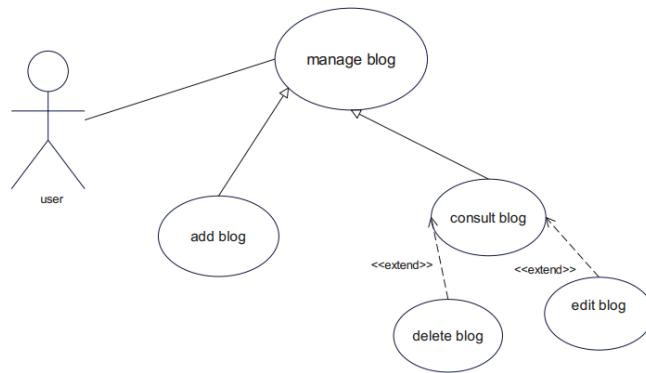


Figure 2.5: use case manage blog

user case	manage blog
actor	user
pre-condition	user should be logged in
Post-condition	A blog should be managed
Main scenario	<ul style="list-style-type: none"> - The system displays the blog management interface. -The actor can edit the blog by clicking on "edit". -The actor can delete the blog by clicking on "delete". -The actor can consult the blog by clicking on "read more" - The system displays the home page according to the profile of the user

Table 2.7: “manage blog” use case refinement

2.3.4 "Manage account" use case refinement

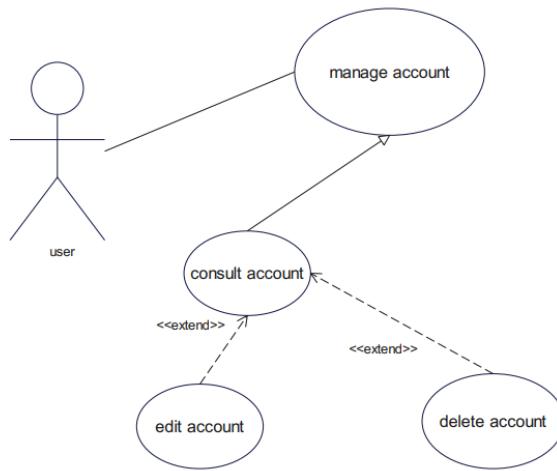


Figure 2.6: manage account use case diagram

user case	manage account
actor	user
pre-condition	user should be logged in
Post-condition	The account should be managed
Main scenario	<ul style="list-style-type: none"> - The system displays the account management interface. -The actor can edit the account information by clicking on "edit". -The actor can change the picture of the account.

Table 2.8: “manage account” use case refinement

2.3.5 "Add blog" use case refinement

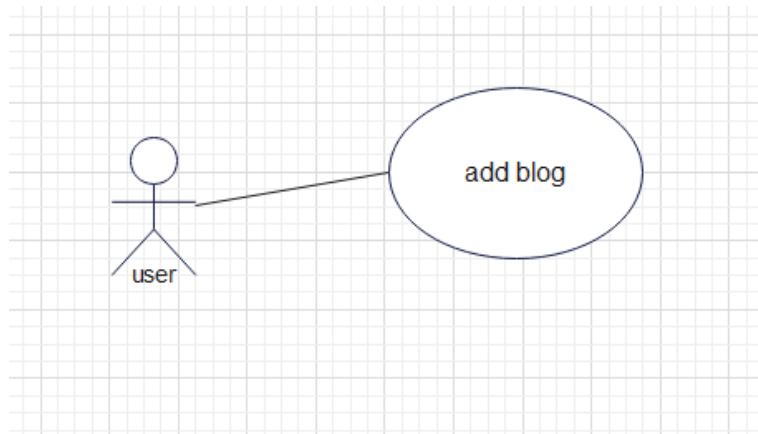


Figure 2.7: add blog use case diagram

user case	add blog
actor	user
pre-condition	user should be logged in
Post-condition	the blog should be properly added
Main scenario	<ul style="list-style-type: none">- The system displays the add blog interface.-The user can add the needed information such as title ,description-The user click on button add so the blog can be added to the data base.

Table 2.9: “add blog” use case refinement

2.3.6 "edit blog" use case refinement

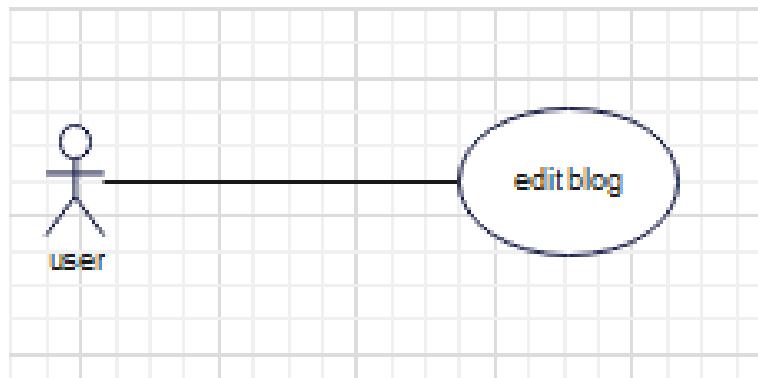


Figure 2.8: edit blog use case diagram

user case	edit blog
actor	user
pre-condition	user should be logged in
Post-condition	the blog should be properly edited
Main scenario	<ul style="list-style-type: none">- The system displays the edit blog interface.-The user can update the needed information such as title ,description-The User click on button confirm to save the changes

Table 2.10: “edit blog” use case refinement

2.4 Sprint 0 conception

Conception is an important activity for comprehending the development of a system in order to make it more reliable and faithful to the needs of the user. As a result, we will specify our system before implementing it.

2.4.1 “Authenticate” use case conception:

Class diagram

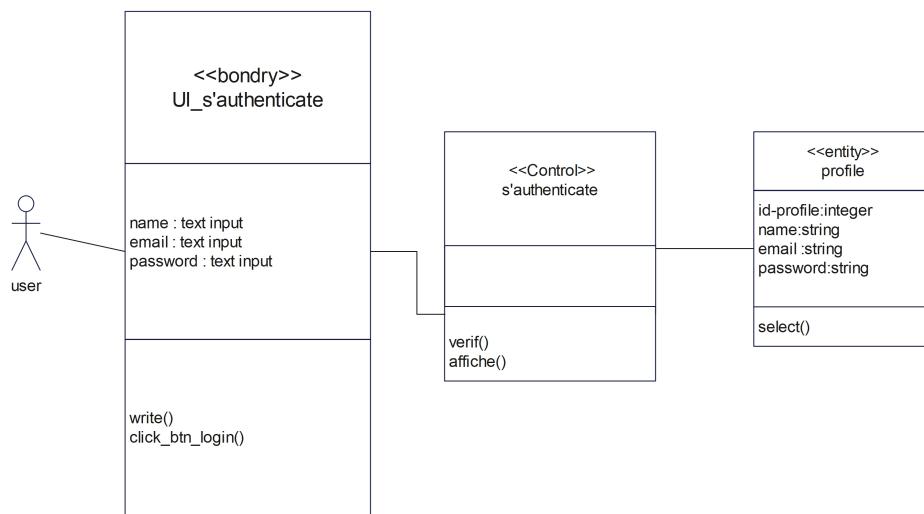


Figure 2.9: s'authenticate class diagram

Sequence diagram

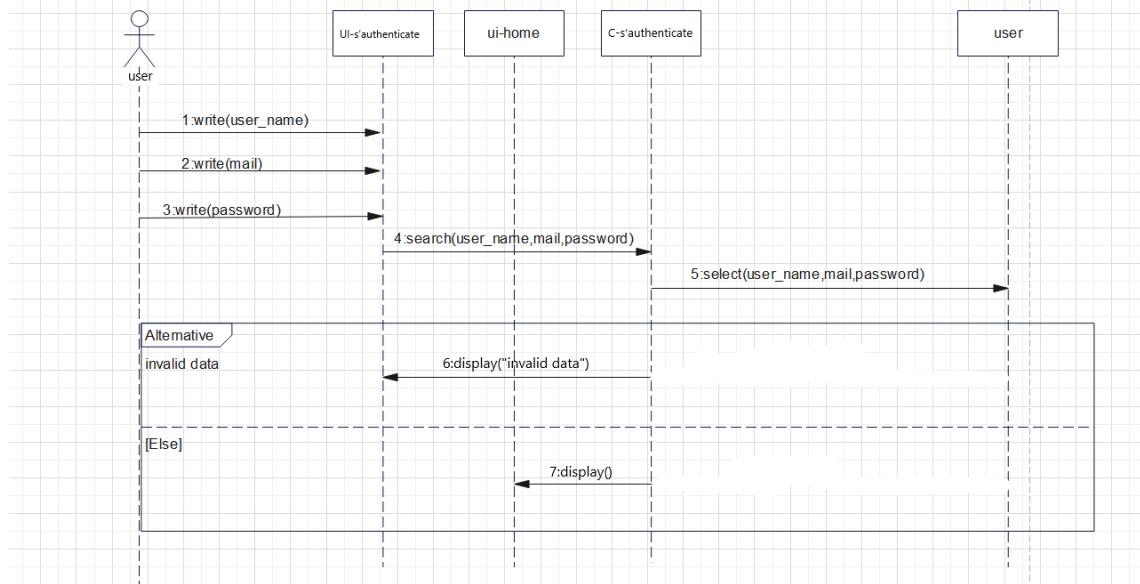


Figure 2.10: s'authenticate sequence diagram

2.4.2 “Create account” use case conception: Class diagram

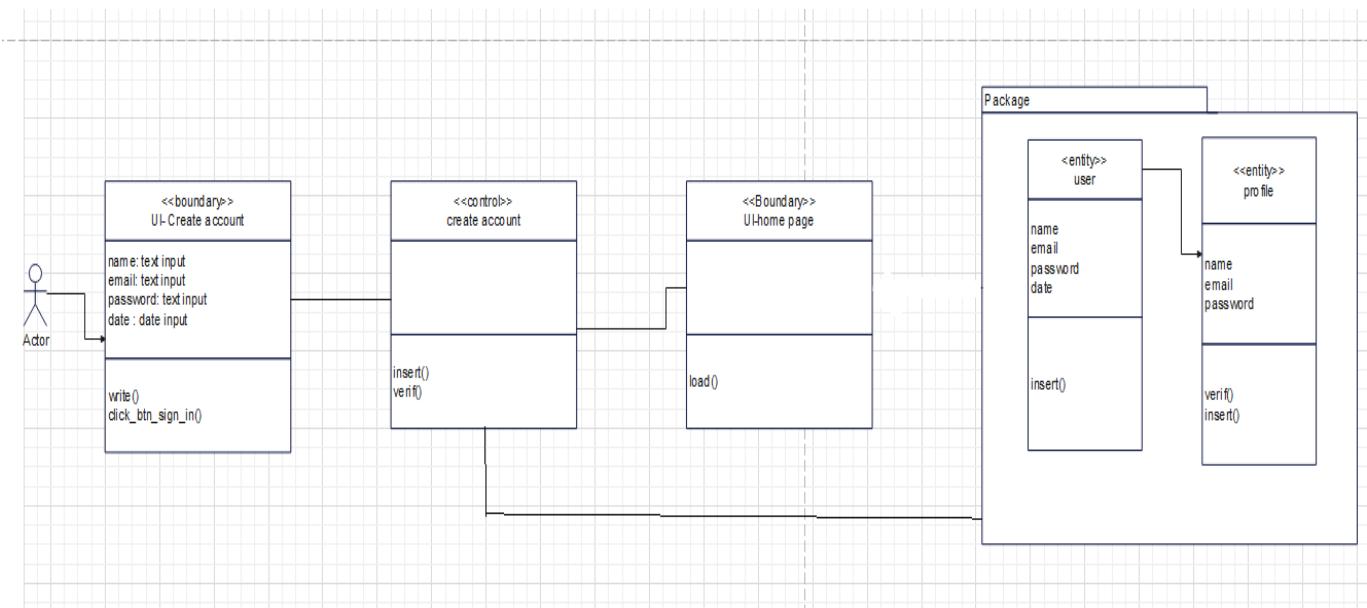


Figure 2.11: Create account diagram class

Sequence diagram

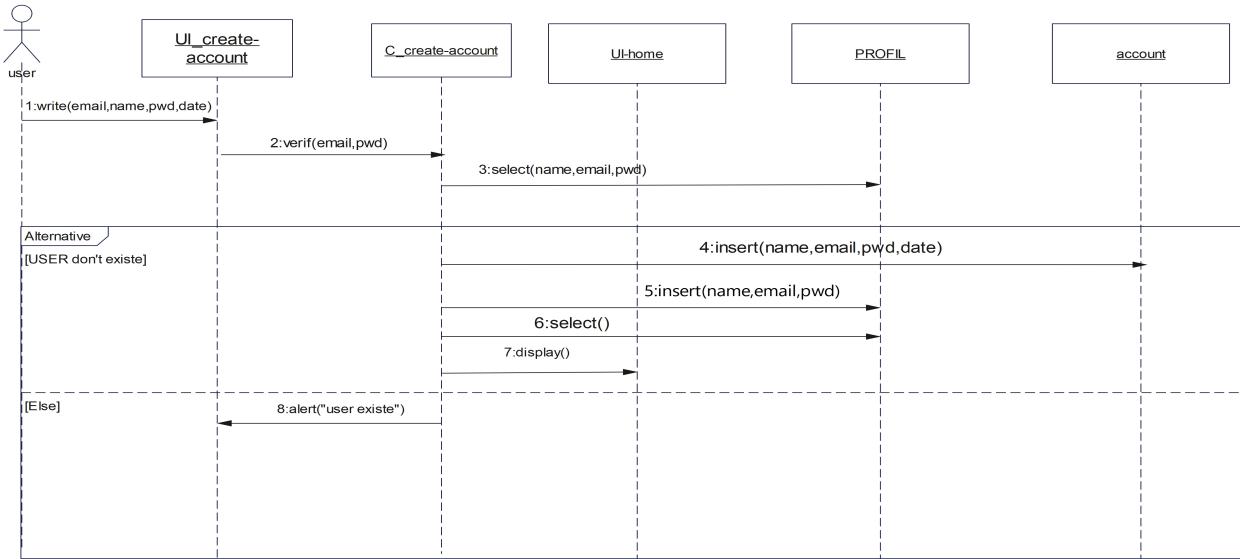


Figure 2.12: create account sequence diagram

2.4.3 “manage account” use case conception:

Class diagram

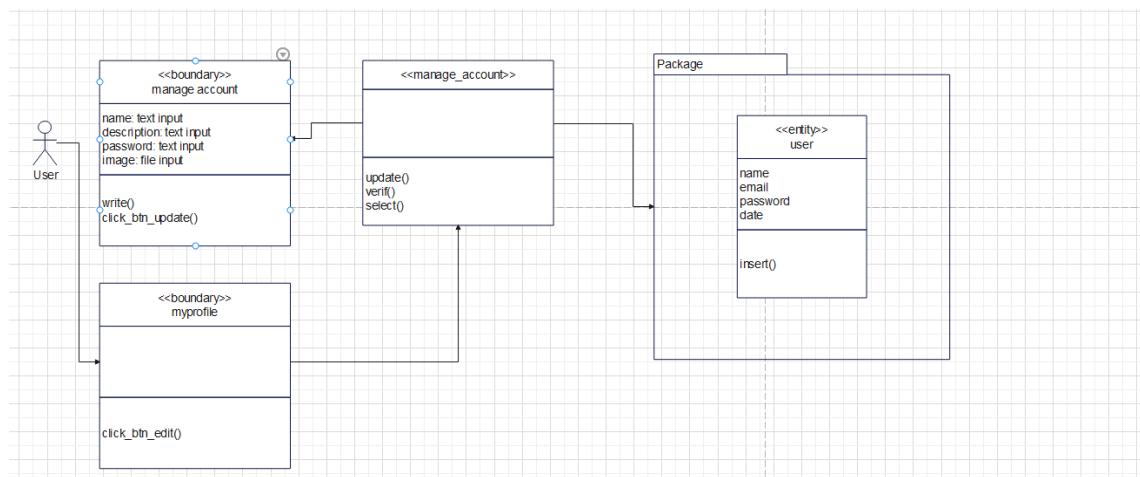


Figure 2.13: manage account class diagram

Sequence diagram

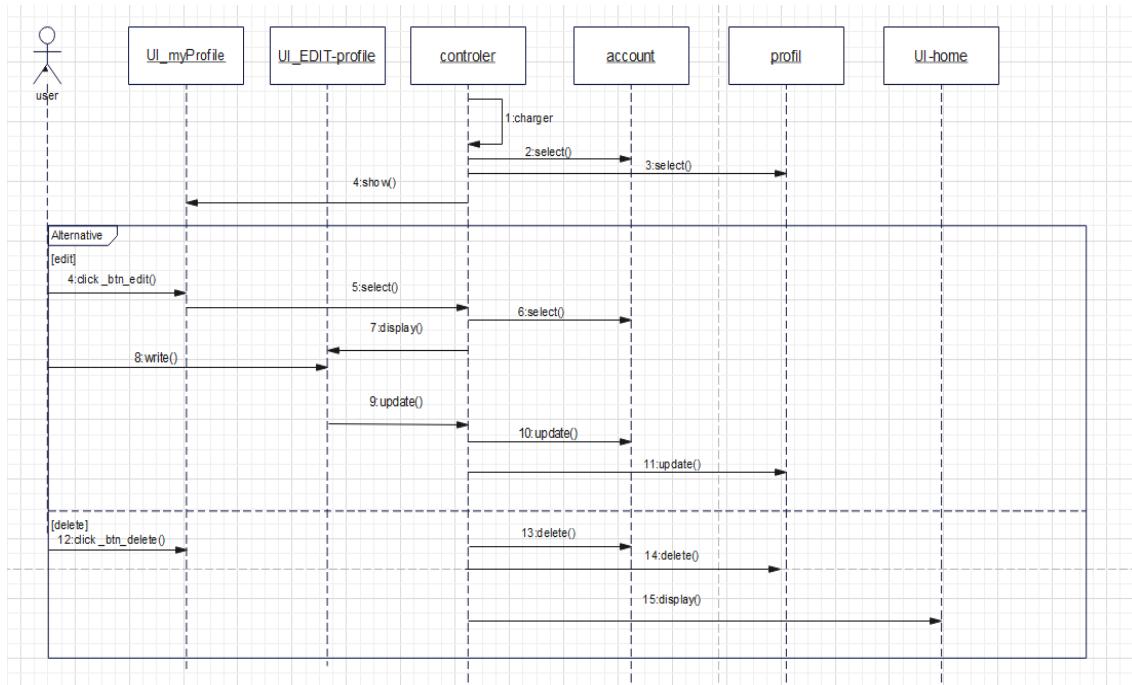


Figure 2.14: manage account sequence diagram

2.4.4 “edit blog” use case conception:

Class diagram

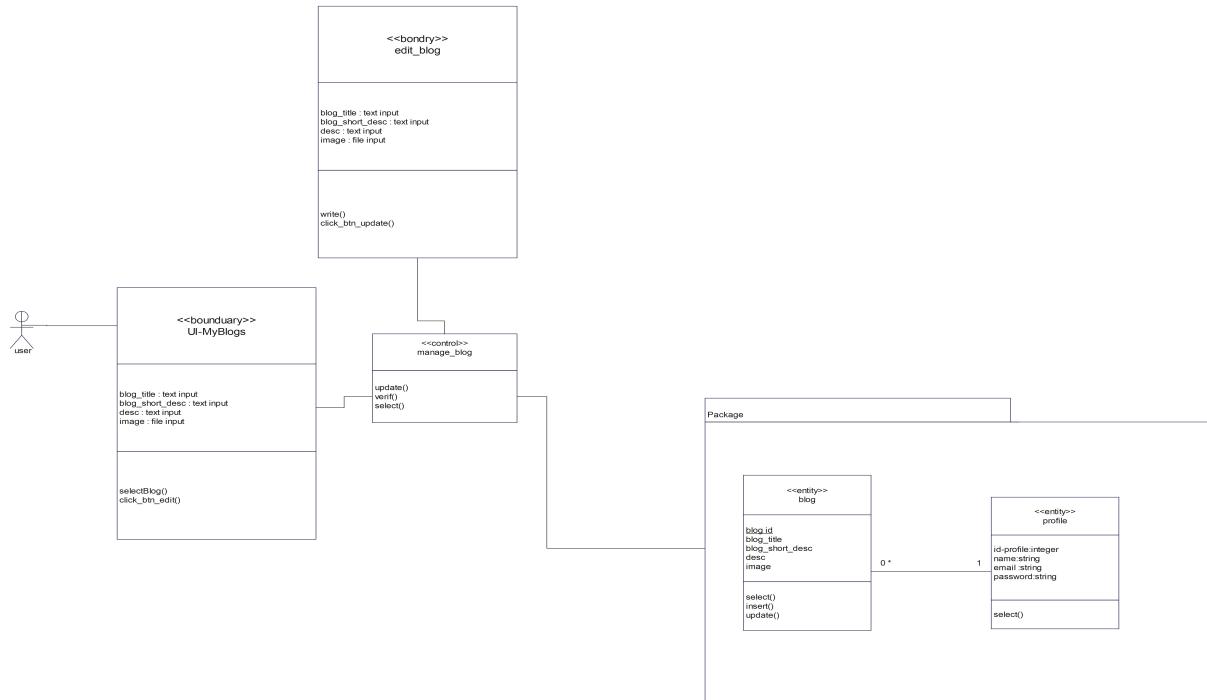


Figure 2.15: edit blog class diagram

Sequence diagram

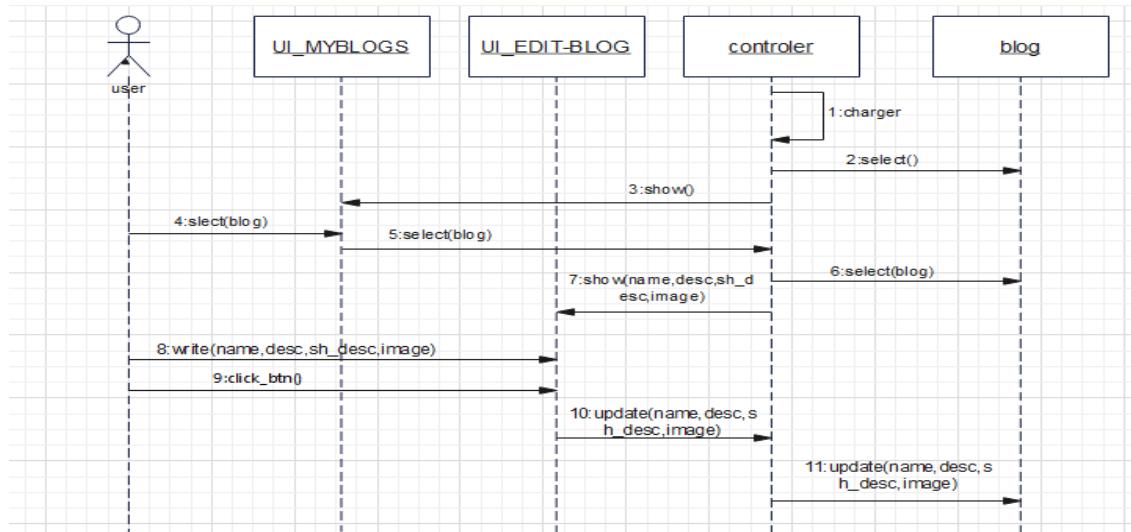


Figure 2.16: edit blog sequence diagram

2.4.5 “add blog” use case conception:

Class diagram

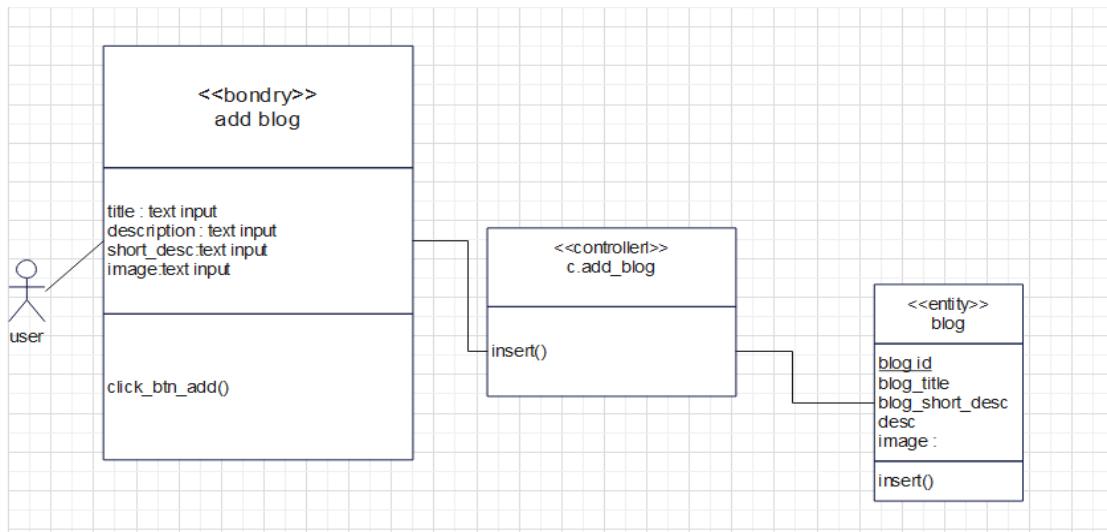


Figure 2.17: add blog class diagram

Sequence diagram

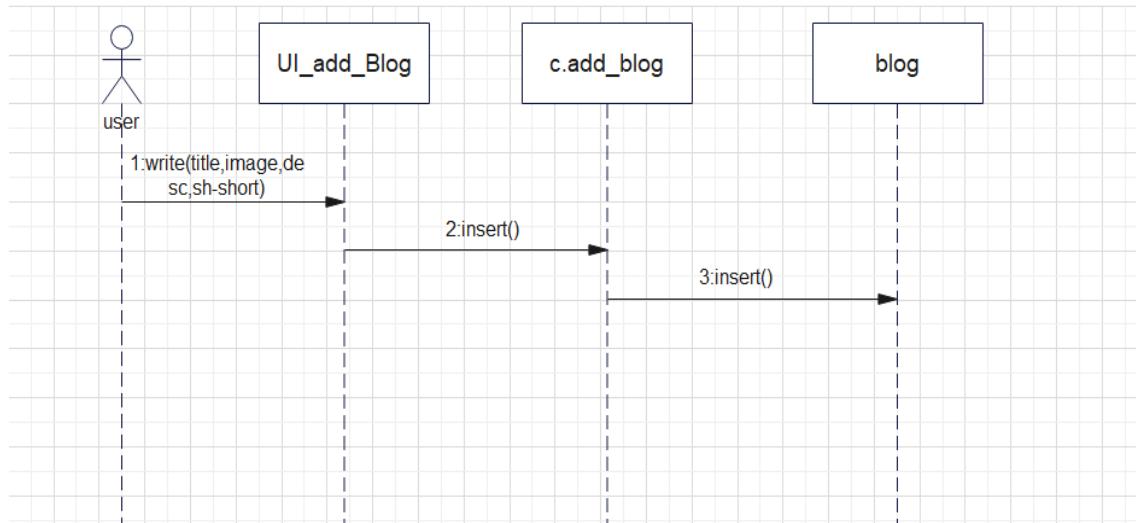


Figure 2.18: add blog class diagram

2.4.6 Global class diagram of sprint 0:

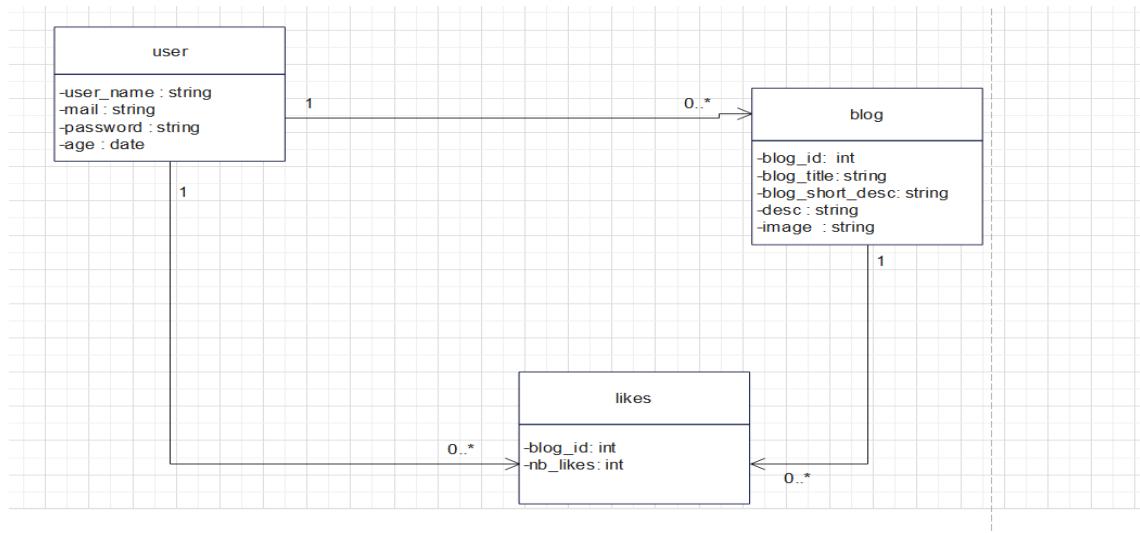


Figure 2.19: sprint 0 global class diagram

2.5 Completion of sprint 0

2.5.1 interface "login"

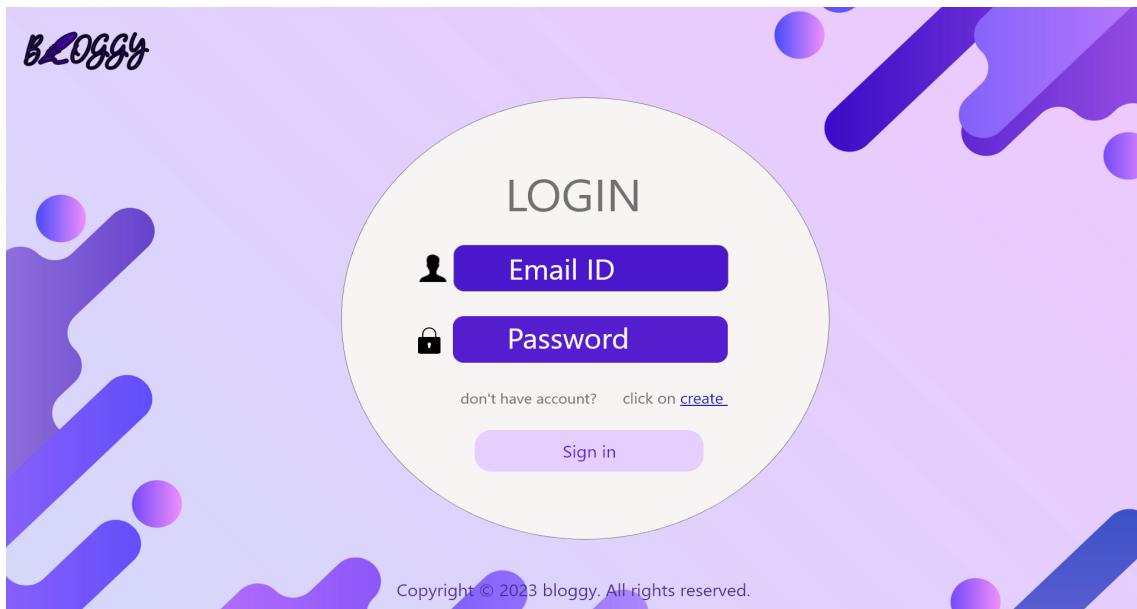


Figure 2.20: login interface

2.5.2 interface "create account"

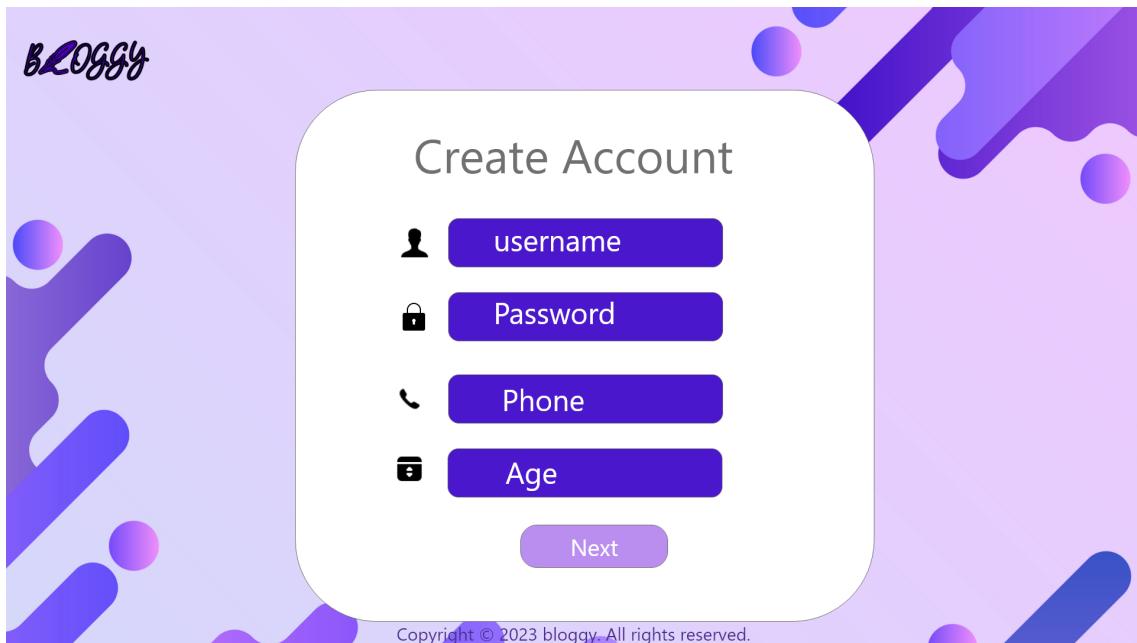


Figure 2.21: create account interface

2.5.3 interface "manage blog"

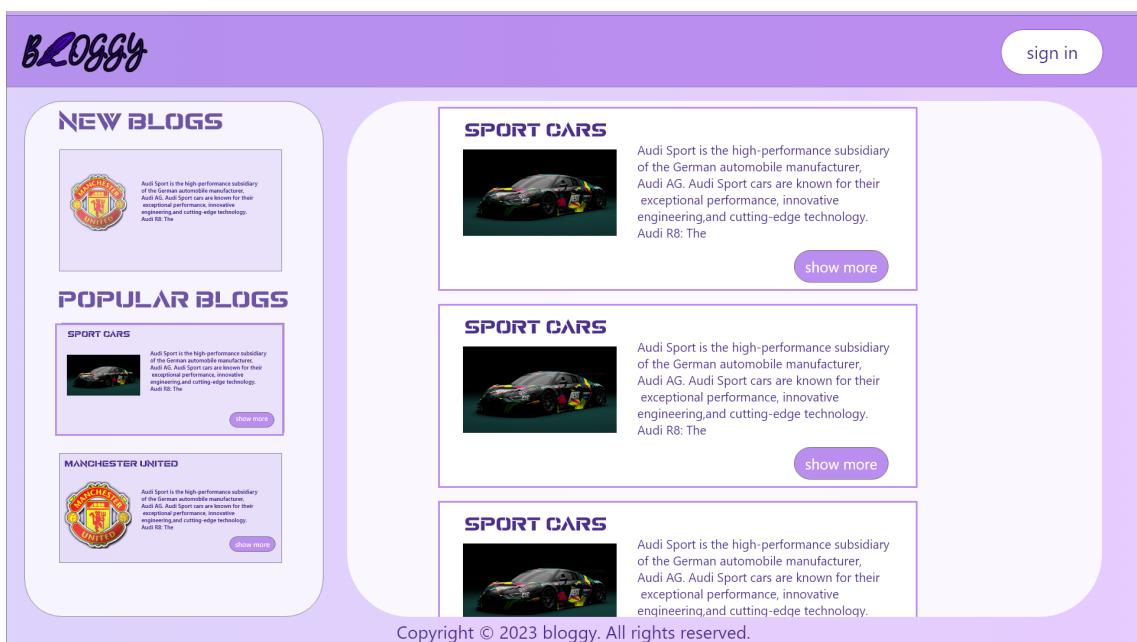


Figure 2.22: home page interface

2.5.4 interface "manage account"

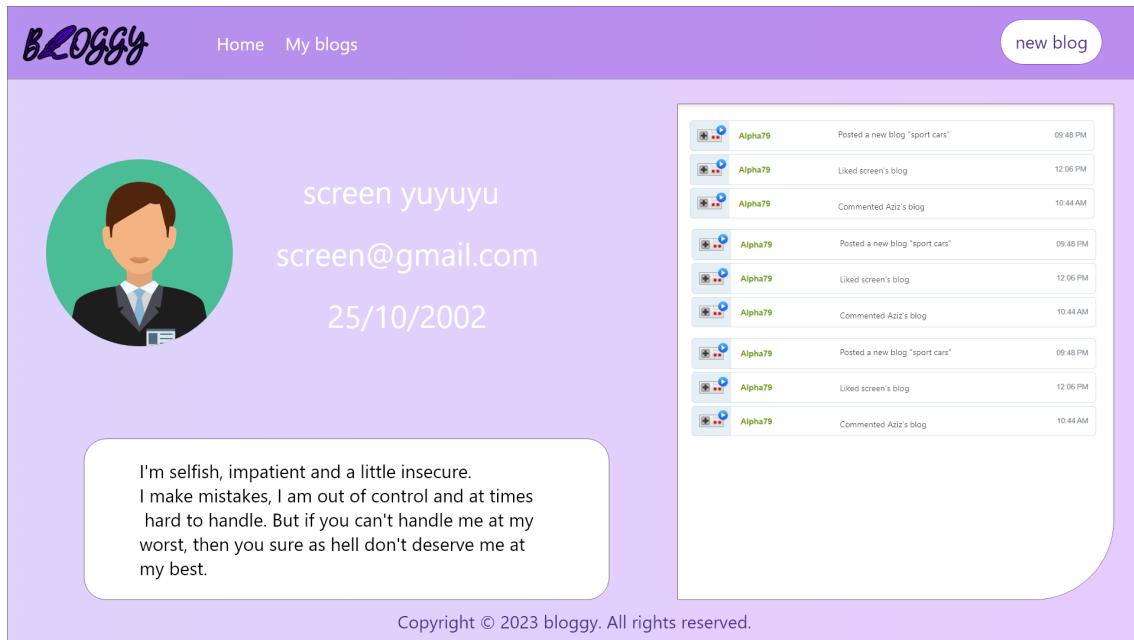


Figure 2.23: manage account interface

2.5.5 interface "add blog"

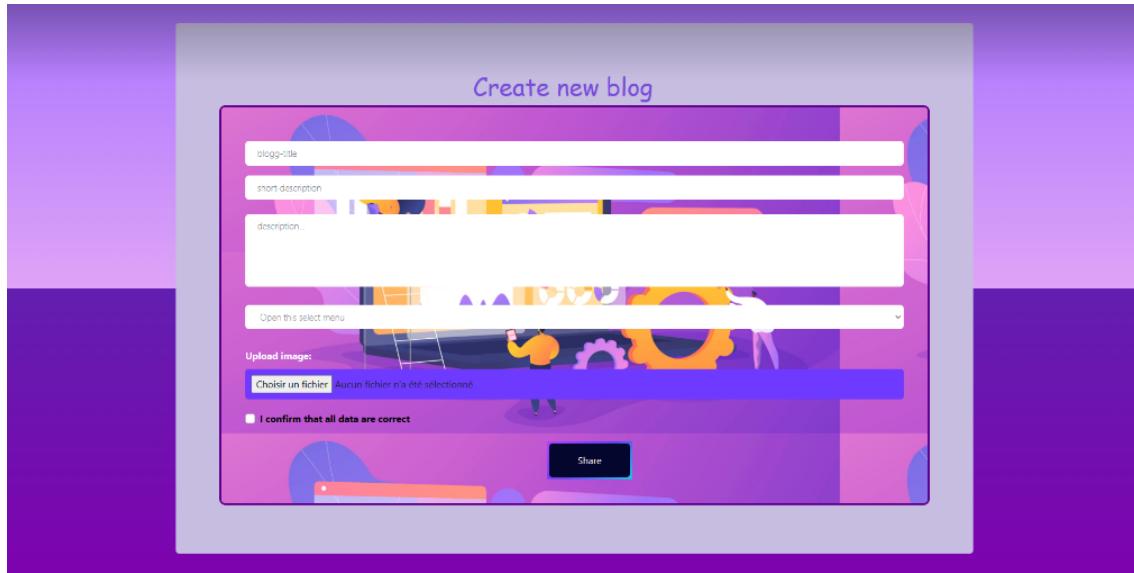


Figure 2.24: add blog interface

2.6 Conclusion

We have realized the use cases of release 1 in this chapter by Then we presented some interfaces and their analyses. The second release devoted to "Blog monitoring" is presented in the following chapter.

chapter 3

Sprint 1

3.1 Introduction

In this chapter, we will present the project's second release, which is: The blog's follow-up, which allows for more detail on the priority 2 use cases. The study of these sprints includes refinement, design, and implementation.

3.2 Sprint 1 backlog identification

Product Backlog Item	Priority	Estimation	Planned release
As user I can navigate blog	2	Medium	Sprint 1
As user I can search for a blog	2	Hard	Sprint 1
As guest I can navigate blog	2	Medium	Sprint 1
As admin I can delete blog	2	Medium	Sprint 1
As admin I can delete users accounts	2	Hard	Sprint 1

Table 3.11: product Backlog of release 2

3.3 Sprint Refinement 1

- read blog
- delete blog
- delete user
- search for a blog

3.3.1 Refinement of “read blog” use case

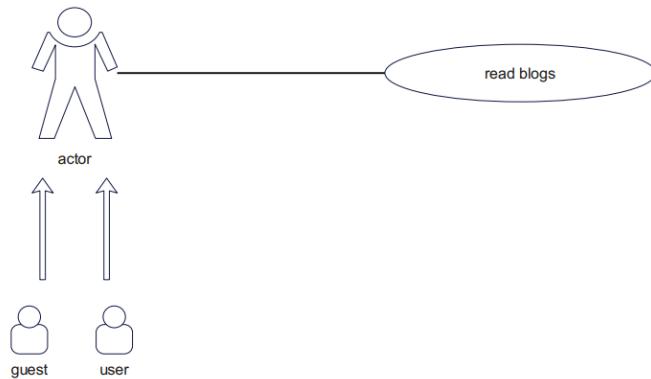


Figure 3.25: Case diagram read blog

user case	read blog
actor	user/guest
pre-condition	System running
Post-condition	blog properly displayed
Main scenario	<ul style="list-style-type: none">- The system displays the home interface.-The actors choose the blog .-The actors click on the “read more” button.- The system displays the blog.

Table 3.12: “read blog” use case refinement

3.3.2 Refinement of “delete blog” use case

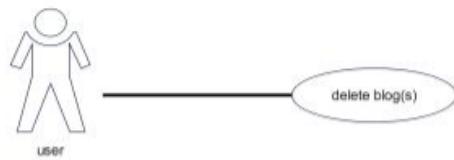


Figure 3.26: Case diagram delete blog

user case	delete blog
actor	user
pre-condition	System running E-mail and password are verified
Post-condition	blog properly deleted
Main scenario	- The system displays my blogs interface. -The actors choose the blog . -The actors click on the “delete” button. - The system removes the blog.
Exception	The system displays an error message if the blog isn't removed

Table 3.13: “delete blog” use case refinement

3.3.3 Refinement of “delete user” use case

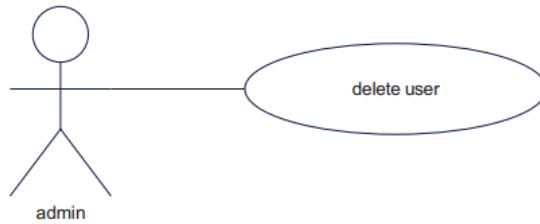


Figure 3.27: Case diagram delete user

user case	delete user
actor	Administrator
pre-condition	System running E-mail and password are verified
Post-condition	user properly deleted
Main scenario	<ul style="list-style-type: none"> - The system displays the user profile . -The Administrator click on the “delete” button. - The system removes the user.
Exception	The system displays an error message if the user isn't removed

Table 3.14: “delete user ” use case refinement

3.3.4 Refinement of “search for blog” use case

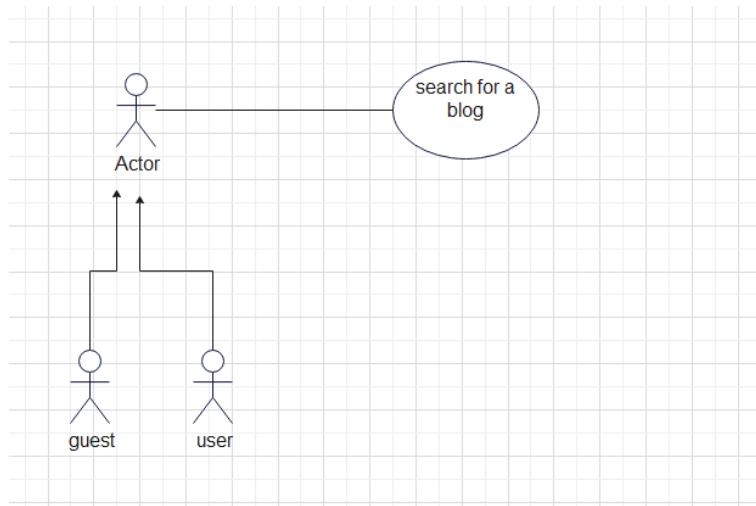


Figure 3.28: Case diagram search blog

user case	search a blog
actor	User/guest
pre-condition	System running
Post-condition	blog found
Main scenario	<ul style="list-style-type: none"> - The system displays the home interface. -The actor write the name of the blog. -The actor click on button search . - The system displays results up on the search of the user .
Exception	The system displays an error message if the doesn't exist

Table 3.15: “search for a blog ” use case refinement

3.4 Conception of sprint 1

Conception is an important activity in understanding the development of a system in order to make it more reliable and responsive to the needs of the customer. As a result, we will specify our system before implementing it.

3.4.1 conception of use case "read blog"

Class diagram

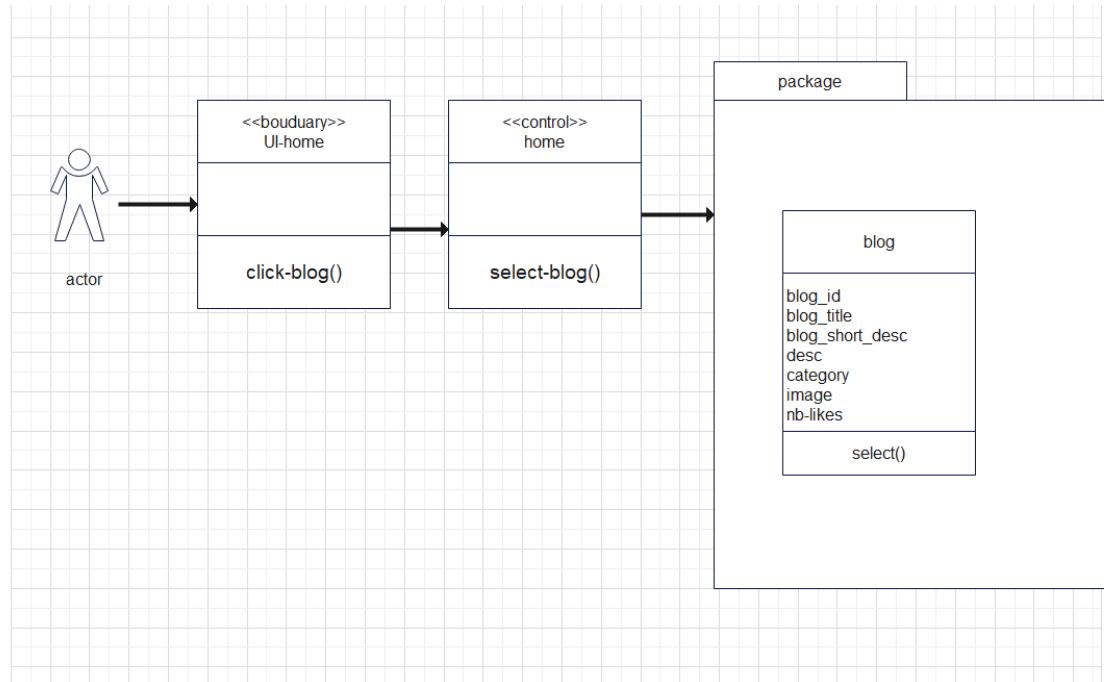


Figure 3.29: read blog class diagram

Sequence diagram

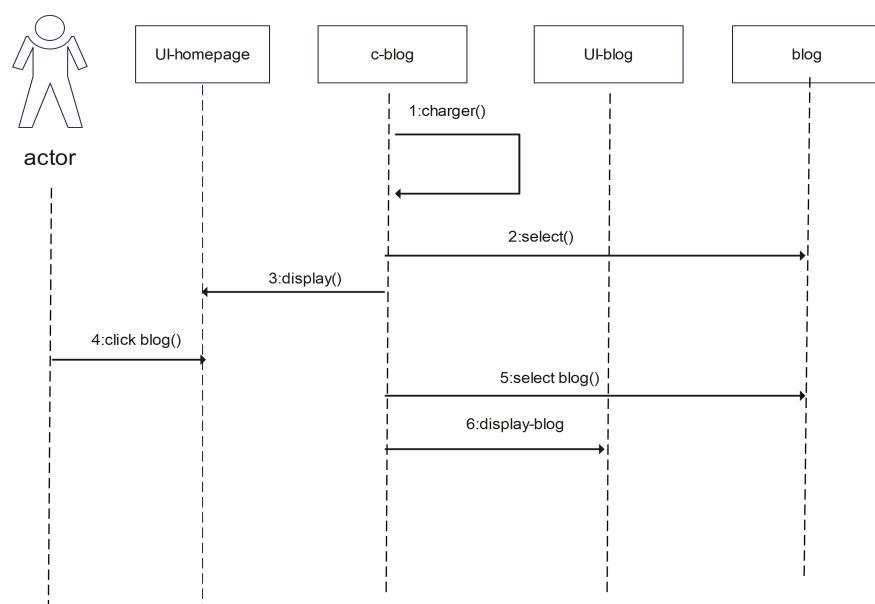


Figure 3.30: read blog sequence diagram

3.4.2 conception of use case "delete blog"

Class diagram

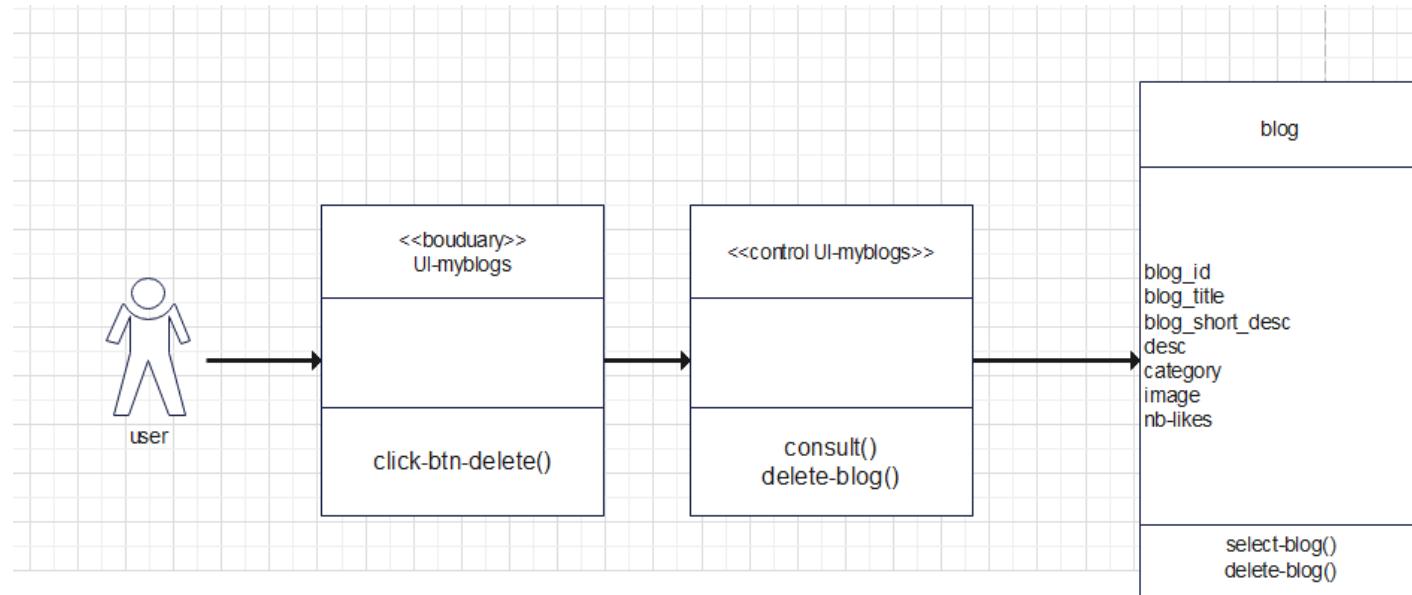


Figure 3.31: delete blog class diagram

Sequence diagram

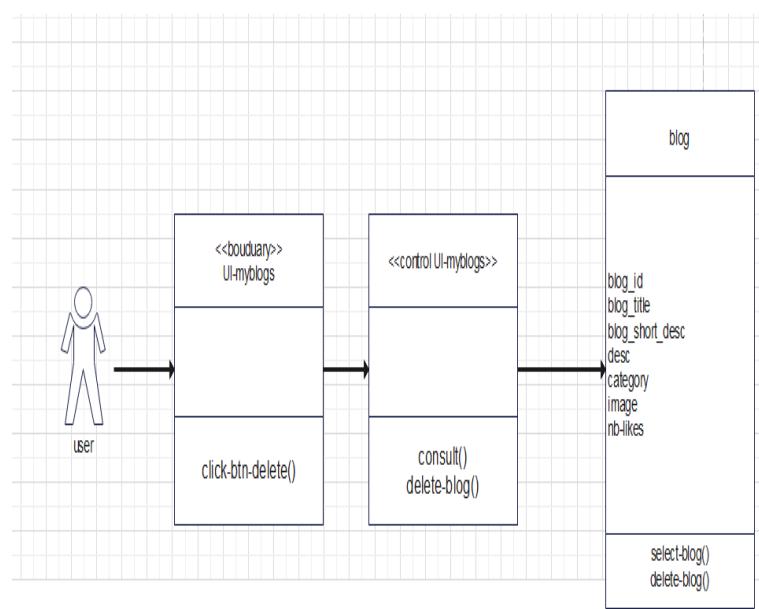


Figure 3.32: delete blog sequence diagram

3.4.3 conception of use case "delete user"

Class diagram

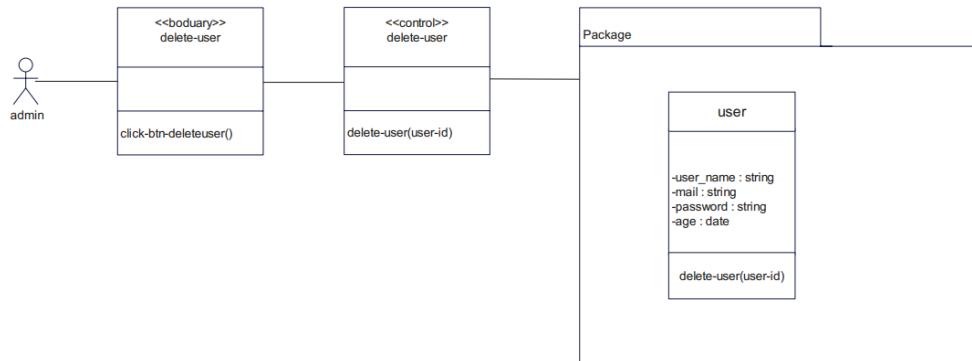


Figure 3.33: delete user class diagram

Sequence diagram

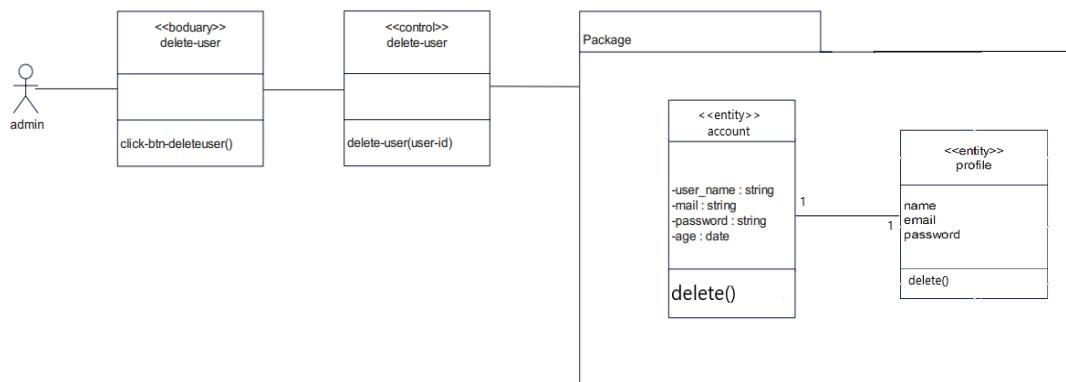


Figure 3.34: delete user sequence diagram

3.4.4 conception of use case "search for a blog"

Class diagram

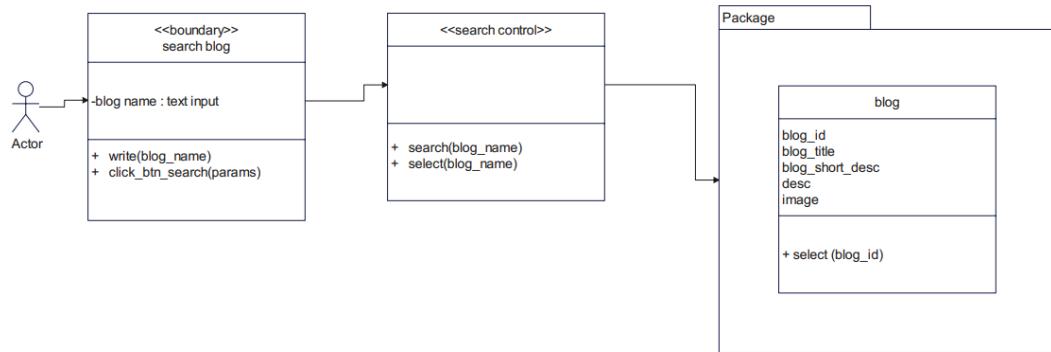


Figure 3.35: search blog class diagram

Sequence diagram

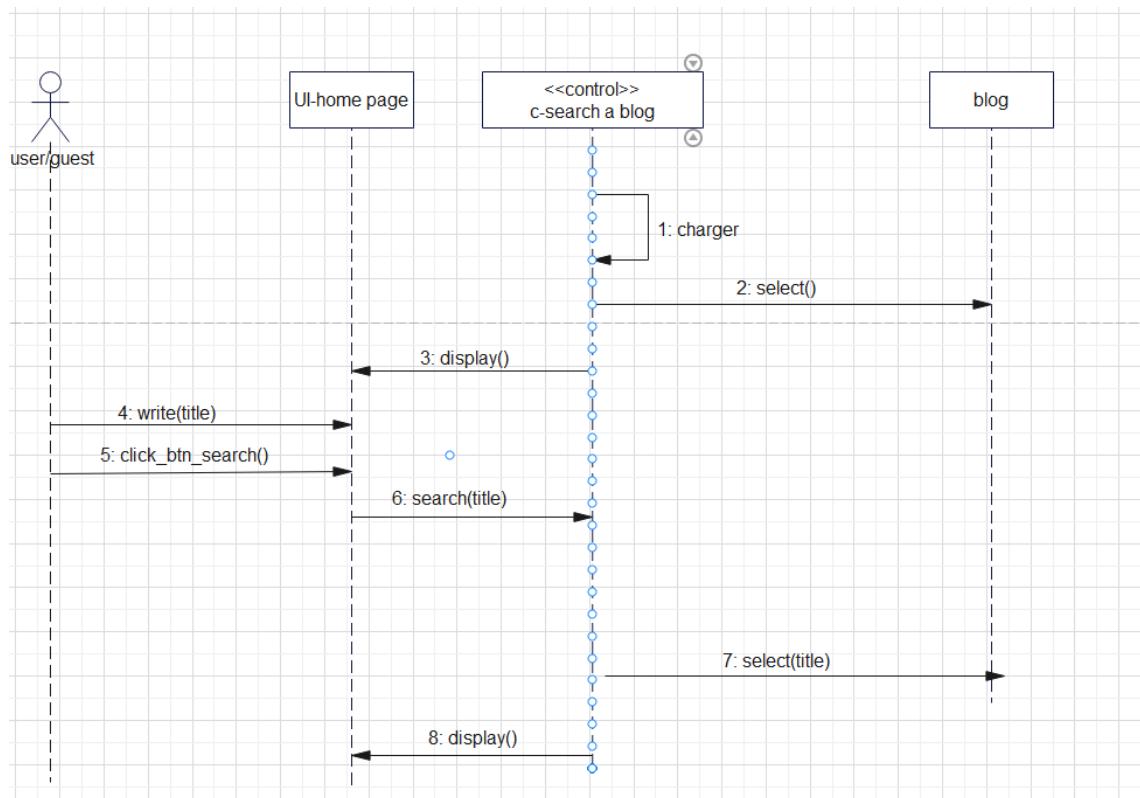


Figure 3.36: search blog sequence diagram

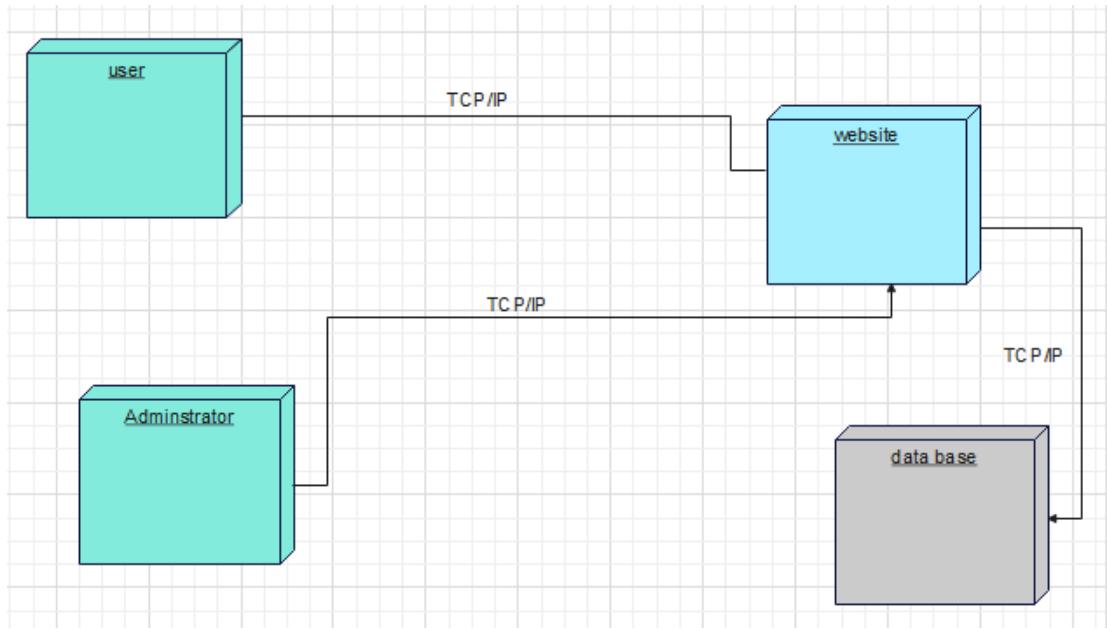
deployment diagram

Figure 3.37: Deployment diagram

3.5 completion of sprint 1

3.5.1 use case "read blog" interface

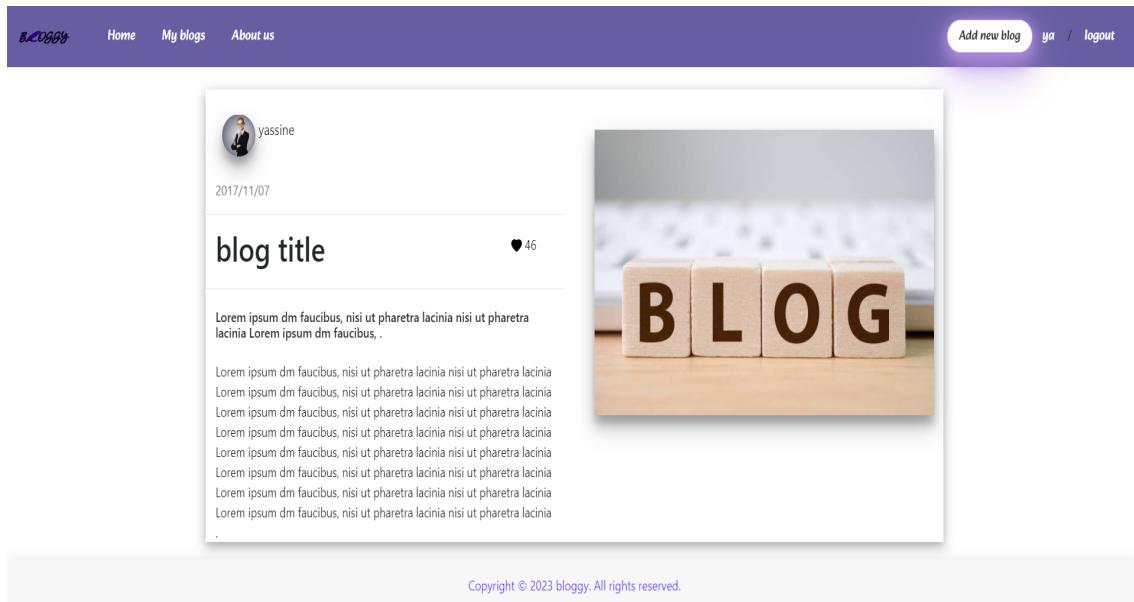


Figure 3.38: read blog interface

3.5.2 use case "delete blog" interface



Figure 3.39: delete blog interface

3.5.3 use case "delete user" interface

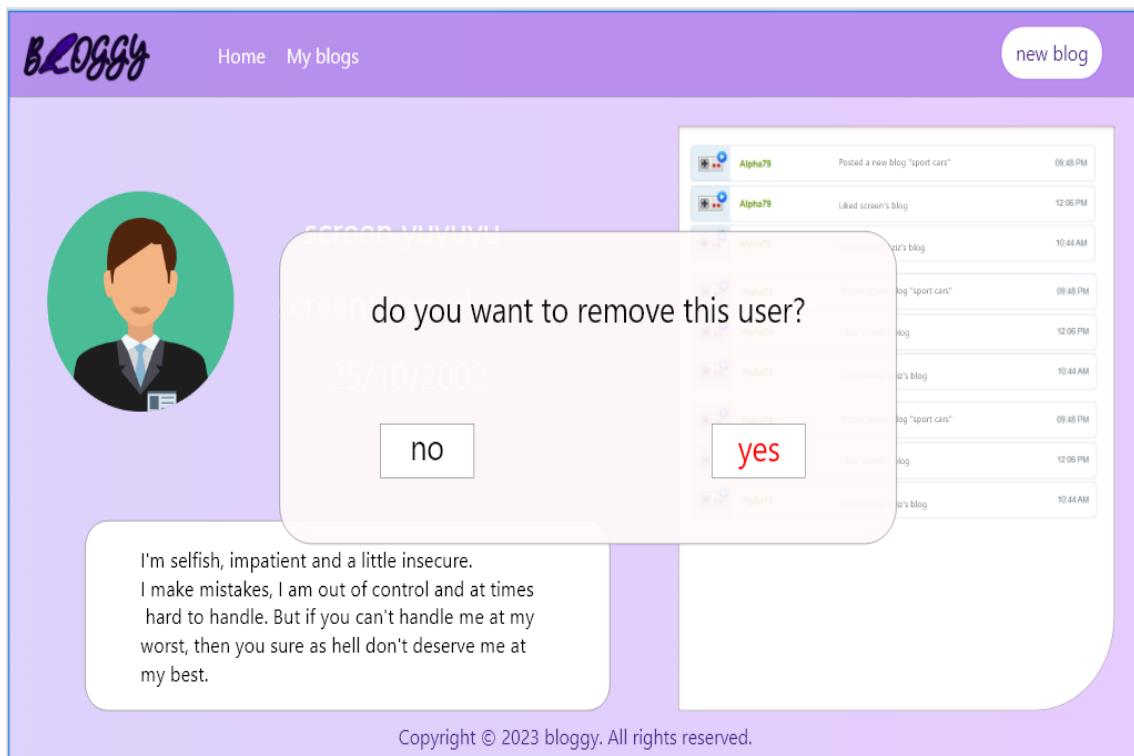


Figure 3.40: delete user interface

3.5.4 use case "search for a blog" interface

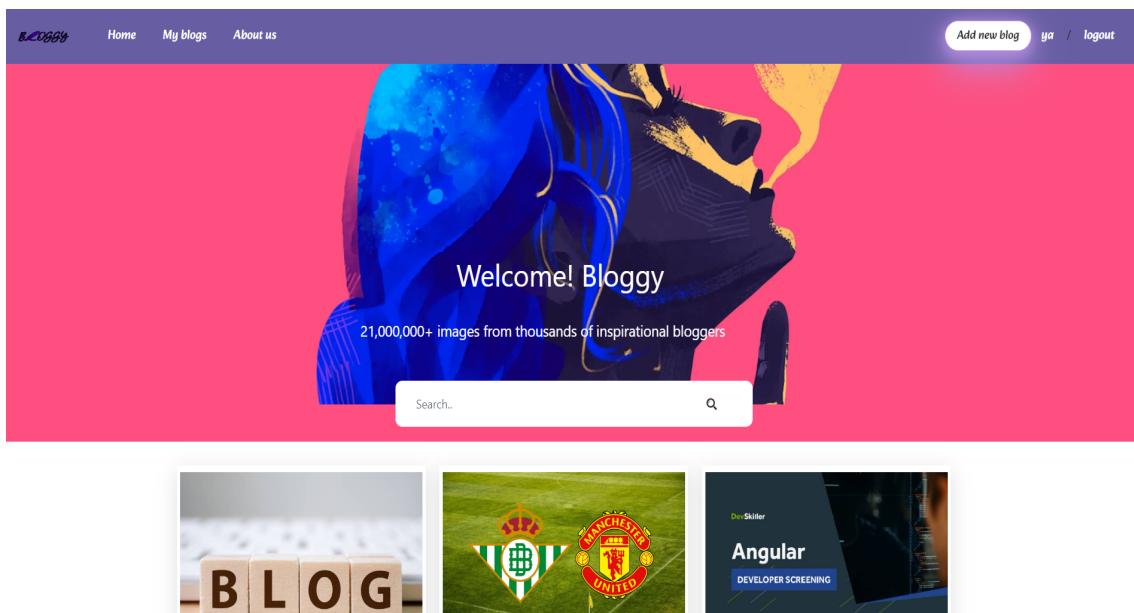


Figure 3.41: search for a blog

3.6 Conclusion

In this chapter, we realized the release 2 use cases, then presented some interfaces and their analyses.

General conclusion

In conclusion, our blog website has the potential to develop into an effective platform for fostering communication and knowledge exchange. Our website is ideally positioned to draw and keep customers over the long term because to its user-friendly interface, broad feature set, and creative design. Our website's emphasis on user participation and interaction is one of its main advantages. We encourage users to engage in meaningful conversations and form connections with others by giving them a choice of methods for sharing and commenting to material. Additionally, our platform allows users to express their thoughts in a variety of media formats, including text and photographs, making it simple for them to do so.

Our website's emphasis on security and privacy is a key component.

In the future, we think that our website has the potential to top the list of platforms for connecting blogs and exchanging content. We can draw a wide variety of users and give them a useful tool for sharing ideas by continuing to innovate and enhance our features and functionality. Our platform has the potential to significantly improve people's lives, and we are dedicated to giving our users the best experience possible.

Bibliography

<https://www.scrumalliance.org/about-scrum>[0]

<https://chat.openai.com>[1][2][3][4][5][6][7][8][9][10]