

الجمهورية الجزائرية الديمقراطية الشعبية

People's Democratic Republic of Algeria

وزارة التعليم العالي والبحث العلمي

Ministry of higher Education and Scientific Research

جامعة أبو بكر بلقايد - تلمسان -

University Aboubakr Belkaïd – Tlemcen –

Faculty of TECHNOLOGY



THESIS

Presented for the **graduation** of **MASTER**

In: Telecommunications

Specialty: Networks and Telecommunications

By:

BENABDALLAH Fatima Youssera

BAHLOUL Rania

Subject

Design and Development of a mobile application for the signalization of road accidents

Publicly supported in 10 /07/2021, in front of the jury composed of:

Mr. MERZOUGUI Rachid	Professor	University of Tlemcen	President
Mr. MOUSSAOUI Djilali	Assistant Professor	University of Tlemcen	Thesis director
Mr. Zerrouki Hadj	Assistant Professor	University of Tlemcen	Examiner

University year 2020/2021

Acknowledgments

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

First of all, we would like to thank Almighty ALLAH who gave us the strength and the will to carry out this work.

We thank all teachers for the quality of the teaching during these past years.

We would like to thank the head of the Telecommunication department Mr. IRID Sidi Mohammed Hadj.

Our sincere thanks in particular to my supervisor Mr. MOUSSAOUI Djilali for his passion for guiding for his encouragement and advising us during the realization of this thesis.

We also thank the member of the jury:

Mr. MERZOUGUI Rachid and Mr. Zerrouki Hadj for having kindly examined and judged this work.

We would like to thank our relatives especially our parents and our siblings who supported us morally and materially and to all those who have helped us from near or far for the preparation of this dissertation.

Dedication

To the best mother in the world AMIMER Fouzia and

to my hero my father BAHLOUL Nadji

To my little sister Douaa Romaissaa

To my brothers Walid Abd Elhaq and Mohammed Cherif

To my grandfather and my grandmother.

For all members of BAHLOUL family and AMMIMER family.

BAHLOUL Rania

Dedication

I dedicate this modest work mainly to the five people who mean the world to me, my parents, my two sisters and my brother. A special feeling of thankfulness to my loving parents, my father, BENABDALLAH MOHAMMED, the mentor of my life and my idol and my mother, ZENASNI FATNA whose words of encouragement and push for tenacity ring in my ears. My little sister, BENABDALLAH FERYEL who has always been the person who understands me and who has always been by my side. My big sister in expatriate BENABDALLAH AMINA, the most strong and inspiring woman I know, my confidante, my other half, despite the distance but you are always in my heart. And my expatriate brother BENABDALLAH BOUMEDIEN who has been a source of great motivation and support. To my friend Ferial who have been always there for me . I also dedicate this thesis to all the people in my life who touch my heart.

BENABDALLAH Fatima Youssera

Abstract

Since developing an application has become a real job, mobile applications are becoming more and more present in our life, as well as road accident.

Our goal was to create an easy and fast application that aim to reduce the number of road accidents by communicating the details of the crash as quickly as possible with the concerned authorities

In order to achieve our plan, we decided to create two applications:

A mobile one dedicated for the signalization, in which Flutter was the development toolkit with its programming language “Dart”, in addition to other tools that have been explained previously in this project.

A web one dedicated for police and other protection services to receive the report that was sent by witnesses via the mobile application.

For the database, Google’s firebase was our choice.

Keywords: web, web application, mobile, mobile application, technology, website, database, call emergency, police, fire fighter, car accidents, road accidents.

Résumé

Dès lors que développement des applications est devenu un vrai métier, les applications mobiles sont de plus en plus présentes dans notre vie, ainsi que les accidents de la route.

Notre objectif était de créer une application simple et rapide qui vise à réduire le nombre d'accidents de la route en communiquant le plus rapidement possible les détails de l'accident aux autorités concernées.

Afin de réaliser notre plan, nous avons décidé de créer deux applications :

Une application mobile dédiée à la signalisation, dans lequel Flutter était la boîte à outils de développement avec son langage de programmation « Dart », en plus d'autres outils qui ont été expliqués précédemment dans ce projet.

Une application web dédiée à la police et aux autres services de protection pour recevoir le rapport envoyé par les témoins via l'application mobile.

Pour la base de données, firebase de Google a été notre choix.

Mots-clés : web, application web, mobile, application mobile, technologie, site web, base de données, appel d'urgence, police, pompier, accidents de voiture, accidents de la route.

ملخص

منذ أن أصبح تطوير أحد التطبيقات عملاً حقيقياً ، أصبحت تطبيقات الهاتف المحمول حاضرة أكثر فأكثر في حياتنا ، فضلاً عن حوادث الطرق .

كان هدفنا إنشاء تطبيق سهل وسريع يهدف إلى تقليل عدد حوادث الطرق من خلال توصيل تفاصيل الحادث في أسرع وقت ممكن مع الجهات المعنية من أجل تحقيق خطتنا، قررنا إنشاء تطبيق

تطبيق للهاتف المحمول مخصص للافتات ، حيث كان فلاتر هو مجموعة أدوات التطوير مع لغة البرمجة "دارت" ، بالإضافة إلى الأدوات الأخرى التي تم شرحها مسبقاً في هذا المشروع. تطبيق ويب مخصص للشرطة وخدمات الحماية الأخرى لاستلام التقرير المرسل من الشهود عبر تطبيق الهاتف المحمول

النارية هي خيارنا بالنسبة لقاعدة البيانات ، كانت قاعدة Google Firebase خيارنا.

الكلمات المفتاحية: الويب ، تطبيق الويب ، الهاتف المحمول ، تطبيقات الهاتف المحمول ، التكنولوجيا ، الموقع الإلكتروني ، قاعدة البيانات ، استدعاء الطوارئ ، الشرطة ، رجال الإطفاء ، حوادث السيارات ، حوادث الطرق.

LIST OF ABBREVIATIONS

API: *Application Programming Interface*

HTML: *Hypertext Markup Language*

SDK: *Software Development Kit*

XML: *Extensible Markup Language*

HTTP: *Hypertext Transfer Protocol*

HTTPS: *Hypertext Transfer Protocol Secure*

FTP: *File Transfer Protocol*

CSS: *Cascading Style Sheets*

JSON: *JavaScript Object Notation*

URL: *Uniform Resource Locator*

OS: *Operating System*

RAM: *Random-Access Memory*

WWW: *World Wide Web*

J2EE: *Java 2 Enterprise Edition*

PHP: *Hypertext Preprocessor*

CGI: *Computer-Generated Imagery*

JS: *Java Script*

UI: *User Interface*

iOS: *iPhone Operating System*

VSC: *Visual Studio Code*

SQL: *Structured Query Language*

LIST OF FIGURES

<i>Figure I.1</i>	<i>Logo of Android</i>
<i>Figure I.2</i>	<i>Logo of iOS</i>
<i>Figure I.3</i>	<i>Logo of Windows</i>
<i>Figure I.4</i>	<i>Mobile Operating System Market Share Algeria-May2021</i>
<i>Figure I.5</i>	<i>Example of HEADER of web page</i>
<i>Figure I.6</i>	<i>Example of LINKS of web page</i>
<i>Figure I.7</i>	<i>Example of FOOTER of web page</i>
<i>Figure I.8</i>	<i>A group of interlinked well-structured webpage</i>
<i>Figure I.9</i>	<i>Structure of static website</i>
<i>Figure I.10</i>	<i>Structure of dynamic website</i>
<i>Figure I.11</i>	<i>Structure of web application</i>
<i>Figure I.11</i>	<i>System architecture</i>
<i>Figure II.2</i>	<i>Connections between classes.</i>
<i>Figure II.3</i>	<i>Cardinality</i>
<i>Figure II.4</i>	<i>Cheft f triki class Diagram</i>
<i>Figure II.5</i>	<i>Use case System</i>
<i>Figure II.6</i>	<i>Use case</i>
<i>Figure II.7</i>	<i>Use case Actors</i>
<i>Figure II.8</i>	<i>Cheft f triki use case Diagram</i>
<i>Figure II.9</i>	<i>Sequence diagram Actor</i>

<i>Figure II.10</i>	<i>Sequence diagram Message</i>
<i>Figure II.11</i>	<i>Cheft f triki sequence Diagram Login for mobile application</i>
<i>Figure II.12</i>	<i>Cheft f triki sequence Diagram Login for web application</i>
<i>Figure II.13</i>	<i>Cheft f triki sequence Diagram Sign up for mobile application</i>
<i>Figure II.14</i>	<i>Cheft f triki sequence Diagram Sign up for web application</i>
<i>Figure II.15</i>	<i>Cheft f tri9y sequence Diagram report for mobile application</i>
<i>Figure III.1</i>	<i>Pillars of FLUTTER</i>
<i>Figure III.2</i>	<i>DART</i>
<i>Figure III.3</i>	<i>Characteristic of VSC</i>
<i>Figure III.4</i>	<i>Services of Google Firebase</i>
<i>Figure III.5</i>	<i>Hosting</i>
<i>Figure III.6</i>	<i>Firestore database</i>
<i>Figure III.7</i>	<i>Creation of new Firebase project</i>
<i>Figure III.8</i>	<i>Install Firebase CLI</i>
<i>Figure III.9</i>	<i>Initialize the project</i>
<i>Figure III.10</i>	<i>Deploy the project</i>
<i>Figure III.11</i>	<i>URL of web application</i>
<i>Figure III.12</i>	<i>Home page of web application</i>
<i>Figure III.13</i>	<i>Login page of web application</i>
<i>Figure III.14</i>	<i>Sign up page of web application</i>
<i>Figure III.15</i>	<i>Example of new account</i>
<i>Figure III.16</i>	<i>Storage of Data in Firebase</i>
<i>Figure III.17</i>	<i>Contact page of web application</i>

- Figure III.18 Home page for mobile application*
- Figure III.19 Language*
- Figure III.20 Sign up page for mobile application*
- Figure III.21 Storage of information in Firebase*
- Figure III.22 Login page for mobile application*
- Figure III.23 Code of exporting data to authenticate*
- Figure III.24 Report page for mobile application*
- Figure III.25 Report page choosing photos*
- Figure III.26 The report details upload on our firebase*
- Figure III.27 The code of exporting report data to firebase*
- Figure III.28 Profile edit page*
- Figure III.29 Future work*

TABLE OF CONTENT

ACKNOWLEDGMENTS	I
DEDICATION	II
ABSTRACT.....	III
LIST OF ABBREVIATIONS	IV
LIST OF FIGURES	V
INTRODUCTION	5
AIM	7
CHAPTER I. MOBILE AND WEB TECHNOLOGY	8
Introduction.....	9
I. MOBILE APPLICATION	9
Definition	9
1. Role of mobile application in business world.....	9
2. Type of mobile application.....	10
2.1 Hybrid application	10
2.2 Native application	10
3. Mobile operating system.....	11
3.1 Android (Google).....	11
3.2 iOS (Apple).....	12
3.3 Windows (Microsoft).....	12
4. Statistics.....	13
II. WEB APPLICATION	14
1. Web page	14
1.1 Elements	14
2. Website	16
2.1 Static website	16
2.2 Dynamic website.....	16
3. Web application	17
3.1 How does a web application work?	17
3.2 Function of web application.....	18
3.3 Architecture of web application.....	19
3.3.1 Components of web application architecture	19
3.3.2 Type of web application architecture	19
A. SINGLE PAGE APPLICATION SPA	20
B. MICROSERVICES.....	20
C. SERVERLESS ARCHITECTURES	20
3.3.3 Web server architecture.....	20
A. JAVA WEB APPLICATION ARCHITECTURE	20
B. NODE.JS WEB APPLICATION ARCHITECTURE	20
C. RUBY ON RAILS WEB APPLICATION DEVELOPMENT	21
3.3.4 Expertise of web application	21
3.3.5 Benefits of web application	21
4. Web application VS website.....	22

Conclusion	22
CHAPTER II. THE APPLICATION DESIGN	23
Introduction.....	24
1. System architecture.....	24
2. Class diagram	24
2.1 Definition	24
2.2 Class diagram structure.....	24
2.3 Cheft f tri9y class diagram	26
3. Use case diagram	27
3.1 Definition.....	27
3.2 Cheft f tri9y use case diagram	28
4. Sequence diagram.....	28
4.1 Definition.....	28
4.2 Cheft f tri9y sequence diagram.....	30
4.2.1 LOGIN.....	30
A. MOBILE APPLICATION	30
B. WEB APPLICATION	30
4.2.2 SIGN UP.....	31
C. MOBILE APPLICATION	31
D. WEB APPLICATION	31
4.2.3 REPORT.....	32
Conclusion	32
CHAPTER III. THE APPLICATION REALIZATION	33
Introduction.....	34
1. FLUTTER.....	34
1.1 Definition	34
1.2 Why FLUTTER?.....	34
2. Dart.....	35
2.1 Definition	35
2.2 Why Dart?.....	36
3. Visual Studio Code.....	36
3.1 Definition.....	36
3.2 Characteristics.....	37
3.3 Why VSC?.....	37
4. Firebase.....	37
4.3 Definition.....	37
4.4 Aim of firebase	38
4.5 Firebase services.....	38
I. WEB APPLICATION	40
1. Creation of firebase project.....	40
2. Creation of local project	41
2.1 Install firebase CLI	41
2.2 Initialize the project	42
2.3 Deploy the project.....	42
3. Presentation of graphical interfaces of web application	43
3.1 Home page.....	43
3.2 Login page	44
3.3 Sign up page	44
3.4 Contact page	46
II. MOBILE APPLICATION	46
1. User interfaces	46

1.1 Home page.....	47
1.2 Sign up page.....	48
1.3 Login page.....	49
1.4 Report page.....	50
1.5 Profile edit page.....	52
Conclusion	54
GENERAL CONCLUSION.....	55
BIBLIOGRAPHY	57

INTRODUCTION

INTRODUCTION

Since the appearance of cars, accidents on the road seemed like an inevitable risk that goes hand in hand with driving. Every year, around 1.35 million people die in road crashes. Sadly, this number has been on the rise since 1899.

Recently with pandemic lockdowns and stay-at-home orders which kept many drivers off the roads and highways, no one would think that the number of accidents would decline. However, the drivers who did venture out found open lanes that only invited reckless driving, leading to a sharp increase in traffic-crash deaths.

Car accidents causes are various, some of them are very common such as the spending, the weather, the night driving ...But some not everybody is aware of like the signalization.

The signalization of the accidents is one of the principal reasons, that not just cause another tragedy after the collision of the driver who was not aware of the accident but also the late intervention from the police can lead to a decreasing in the possibility of surviving.

While car accidents can often be out of one's control, there are things you could do to make the road safer and our application is definitely one of those things.

Aim of thesis

The aim of this thesis is to decrease the number of victims due to road accidents. The application gives the ability to people who have witnessed the accident to report about it and gives further details with documentation, therefore our app contacts all the important emergency services so that victims can receive immediate help.

Being involved in a car accident is a stressful and scary situation. You may be disoriented, confused and unable to fully process what happened, in the midst of this confusion you may not be able to recall the details of your accident that's why this app help to collect critical information's of the scene from witnesses

chapter I. MOBILE AND WEB TECHNOLOGY

Introduction

Mobile or Web technology are the trend, they go where the user goes, it's one of those things that make our life much easier. We are going to dedicate this chapter for applications specifically whether web or mobile applications.

I. Mobile technology

1 Mobile application

Definition

In today business world, it's no longer enough just to have a website. People today access the internet on mobiles much more than computers so, what is a mobile app?[1]

A mobile application or simply an app, is a computer program or software application designed to run on a mobile device such as a phone, tablet, or watch. Applications were originally intended for productivity assistance such as email, calendar, and contact databases, but the public demand for application caused rapid expansion into other areas such as mobile games, factory automation, GPS and location-based services, order-tracking, and ticket purchases, so that there are now millions of applications available.[1]

This industry has been active for over a decade now generating billions of dollars in revenue for Apple, Google and thousands of mobile application developers. The exploding popularity of smartphones and tablets have made mobile applications development an increasingly popular trend among business owners all over the world.[1]

1.1 Role of mobile application in business world

Mobile application have made everything so simple. They have simplified the lives of consumers; they have added pace and flexibility to it. But are these mobile applications making any difference in the business world?[2]

There are more than 2.7 billion smartphones users across the globe 90% of the time they spend on them is on the mobile application residing on them, and we can say that 39% of applications on Play Store are business application.[2]

Now mobile is a growing industry that attracts businesses from every marketplace, no wonder mobile application revenues are projected to reach almost 600\$ billion in 2020, for example Android and iOS

application revenue reached \$111 billion revenue in 2020, a 24% increase year-on-year. iOS was responsible for 65% of total application revenue in 2020.[2]

Revenue has climbed steadily from 2016 to 2019, but has seen remarkable growth over the past three quarters due to COVID-19 and global lockdowns which kept people indoors.

Millennial are ready to pay more for mobile applications, with $\frac{1}{3}$ purchasing more than one per month. Mobile applications are helping retail companies to acquire more users and generate higher sales in different ways.[2]

1.2 Type of mobile application

There are two basic types of mobile applications:

1.2.1 Hybrid application

Hybrid application is a software app that combines elements of both native applications and web applications. You install it like a native application, but it is actually a web application on the inside. Hybrid apps, like web apps, are built with JavaScript, HTML, and CSS and run in something called WebView, a simplified browser within your application. Hybrid applications are popular because they allow developers to write code for a mobile application once and still accommodate multiple platforms. Because a hybrid application adds an extra layer between the source code and the target platform, they may perform slightly slower than native or web versions of the same application.

The advantage with a hybrid application is that you only have one codebase to manage and it will operate on different platforms. Hybrid applications are faster in build times compared to native applications and cheaper to develop compared to building two versions of a native application for two different platforms it is easier to launch patches and updates and it can work online and offline.

1.2.2 Native application

A native application is a software application built in a specific programming language, for the specific device platform, either iOS or Android.

Native apps are applications written in languages that the platform they are being built for accepts. For example, native iOS application is written in Swift or Objective-C and native Android apps are written in Java.

Apple and Google offer app developers their own development tools. These tools allow any professional developer to develop a native mobile application relatively easily. Native app offers the fastest, most reliable and most responsive experience to users. They are easy to tap into device functionality like (camera, microphone, compass, accelerometer...). Furthermore, matching app UI to platform conventions is

really easy. But Native applications cost more to develop than hybrid applications and with native applications we have to work with different codebases for every platform we choose to build in, and that could be a big disadvantage, but in the last few years, several alternatives have become available to reap the benefits of cross-platform development without sacrificing the user experience or access to native APIs. There are several platforms that allow for cross-platform development including Xamarin, React Native, Titanium, and Google's Flutter.

1.3 Mobile operating system

A mobile operating system (OS) is the software that allows mobile devices like phones to run applications and other programs. Most mobile operating systems only work on specific hardware. In this part we are going to introduce some of the famous operating systems for mobile phones in Algeria; and showing which one of them is good for mobile applications. [3]

1.3.1 Android (Google)

Upon its release in 2008 it made the buzz and it was the dominant in the market. It is a free and opensource software and it is developed by a group of developers known as the Open Handset Alliance. At its simplest form, the Android platform is based on the Linux, among many mobile operating systems, Android is the most popular operating system. A developer survey conducted in 2017 found that 64.8% of mobile developers use Android as their preferred platform.[3]



Figure I.1 Logo of Android.

- ✓ Advantages
- Incredibly secure.
- Highly customizable.
- Fast development cycle.
- Cross-Platform potentiel.
- User Friendly.

1.3.2 iOS (Apple)

iOS is a mobile operating system developed by Apple. It has been around since the very first iPhone that was created in 2007 iOS uses a graphical user interface currently running on the iPhone, iPod touch, and iPad. It is designed to be simple and easy to use. A major evolution of the system happens every year and the update is free.[4]



Figure I.2 Logo of iOS.

- ✓ Advantages
- Easy navigation
- Perfectly optimized operation for all devices
- Apps come out first on the App Store and the updates are directly available for all iPhone models.

1.3.3 Windows (Microsoft)

For the past three decades, Windows has been the most popular operating system for personal computers. It is a desktop OS developed by Microsoft, beginning with first release Windows 1.0 in 1985 and Windows Mobile is a discontinued family of mobile OS for smartphones and personal digital assistants, its first initial release was in April 2000, 21 years ago. It is similar to the desktop versions of Windows and was aimed at business and enterprise consumers. Windows Mobile never equaled the success of newer mobile OS (iOS, Android) and it has been deprecated.[5]



Figure I.3 Logo of Windows.

✓ Advantages

- Ease of use and rearrange.
- Superb apps & cleaner options.
- An expandable storage.
- Support for new hardware.
- Great potential and integration.

1.4 Statistics

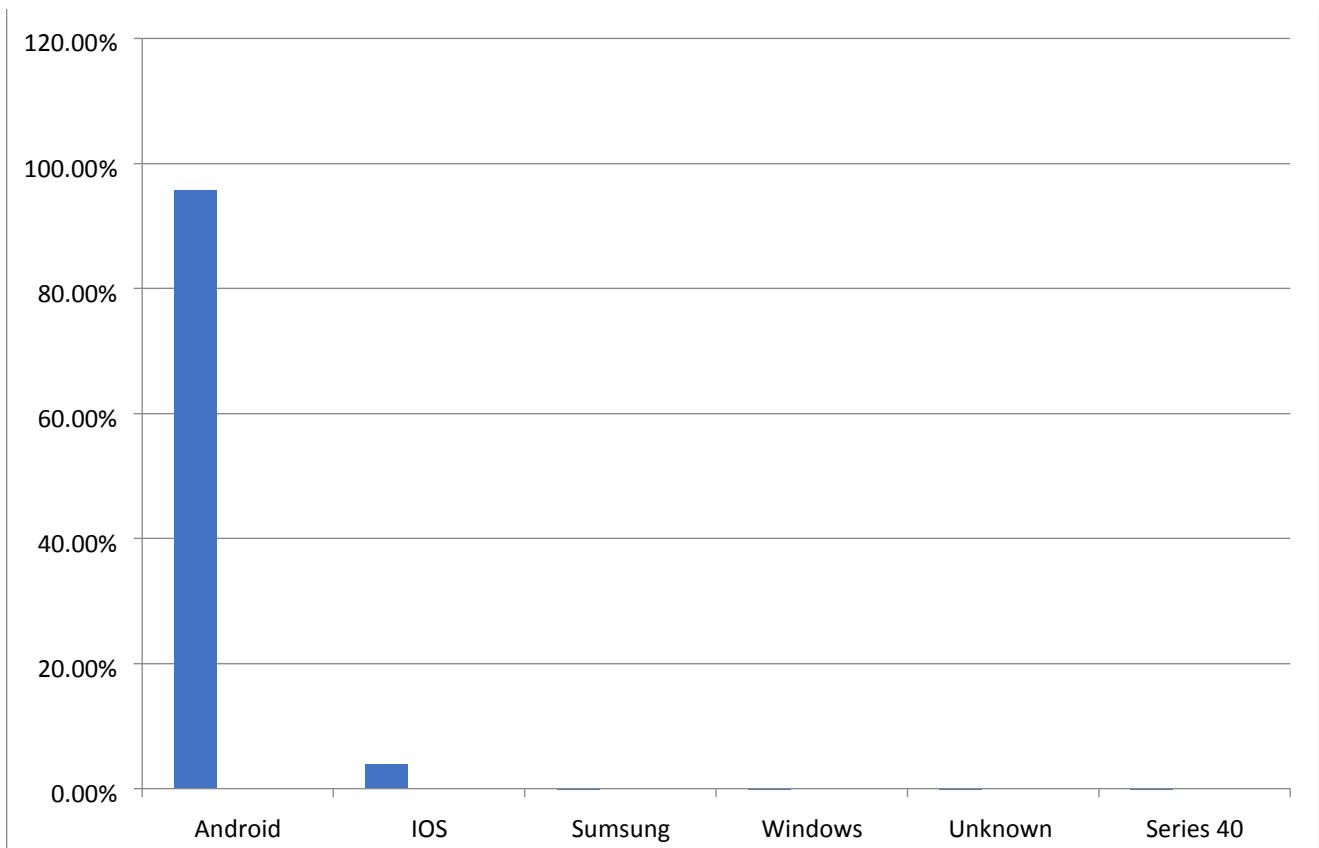


Figure I.4 Mobile Operating System Market Share Algeria-May2021.[6]

In these statistics we notice that almost all Algerians use android as an operating system. That's why we decided to create an application on android in the first place of course thanks to flutter which gives us the ability to create the same app on IOS.

II. Web technology

In this part we are going to see web technology in general including web page, website and web application.

1 Web page

Web page is a single document that is written using HTML, JAVASCRIPT code and use CSS code for presentation, and it's accessible via the web through the internet using a web browser, we access a web page by entering URL (uniform resource locator) address using a web browser. URL looks like:

[Http://www.example.com/index.html](http://www.example.com/index.html)

The URL address must end with a document extension such as: .html, .php... like we see in the example above. A web page may contain text, graphics and hyperlinks to another file. We can use it for searching information, to see new NEWS also we can use it for sell or buy something and many more activities. Multiple web pages make up a website. [7]

The first web page was created at CERN by Tim Berners-Lee on August 6, 1991. The difference between web page and website are that the website refers to a central location containing more than one web page.[7]

1.1 Elements

There is no two pages are similar. However, most pages contain some or all of the elements contained on web page, some of those elements are ('SECTION', 'ARTICLE', 'HEADING', 'FOOTER', 'NAVIGATION BAR', 'LINKS'), here is an example of web page:

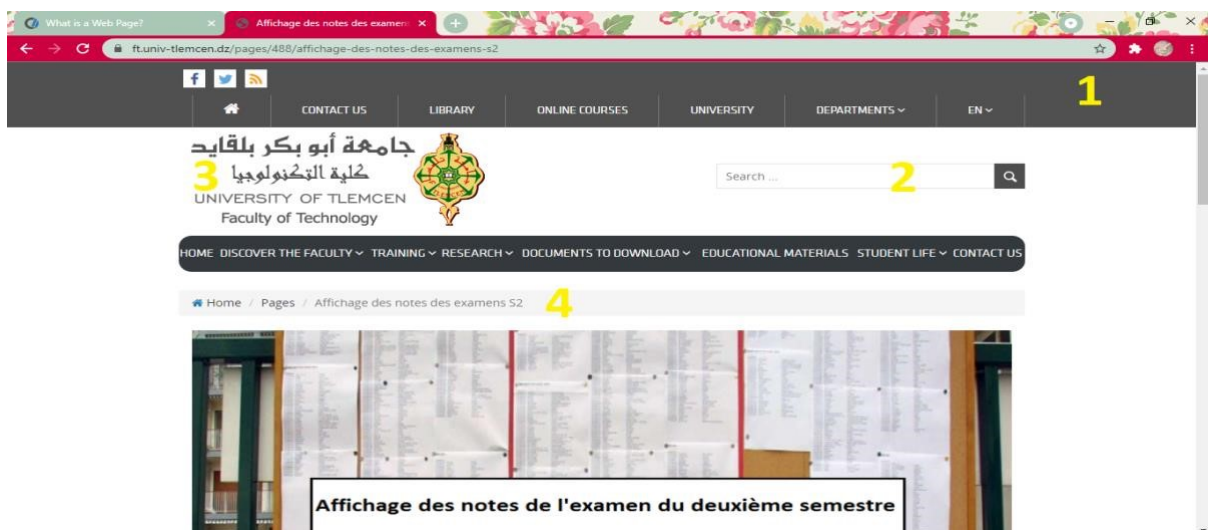


Figure I.5 Example of HEADER of web page.

1-The navigation bar, or menu in general found on header on the top of each web page it includes links to section of website.

2-The search allows a visitor to search a website for related information.[7]

3-Logo or name of website.

4- The breadcrumbs help the user know where they are on the website. [7]

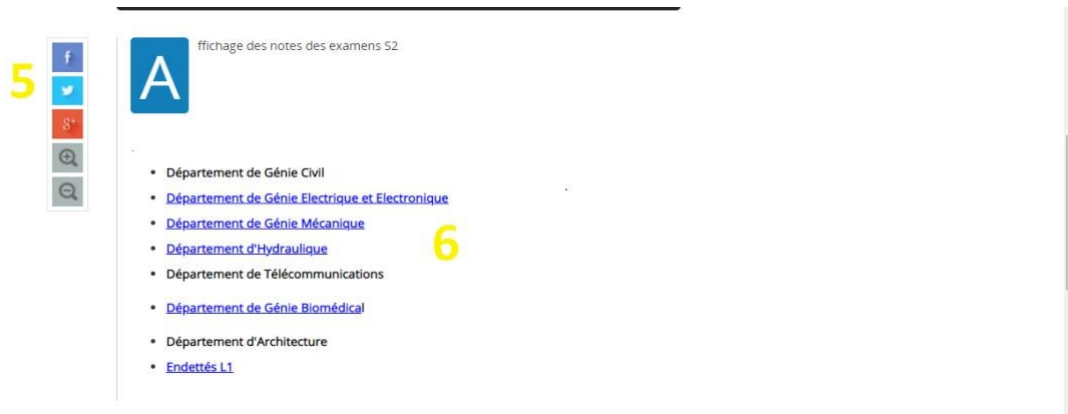


Figure I.6 Example of LINKS of web page.

5-Social share links, allow visitors to share the website with friends and family or anyone else.[7]

6-A list for links of the website.

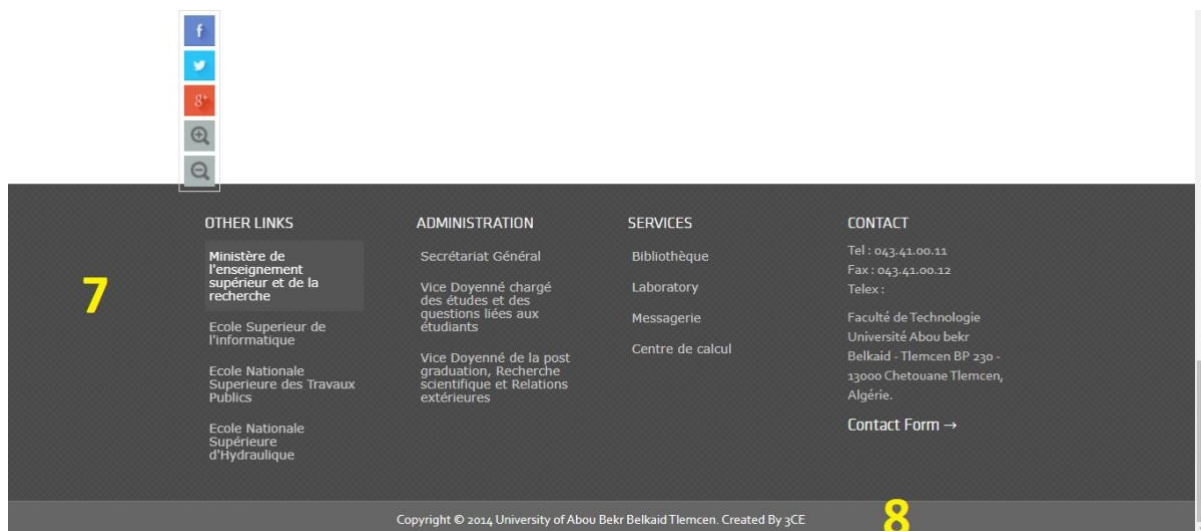


Figure I.7 Example of FOOTER of web page.

7-Footer include anything important related with the website to help visitor for their research.

8-Copyright should be on all the web page, we found it in the end of the web pages.

2 Website

The website is a group of interlinked well-structured webpage that exist on the same domain that is to say grouped together in one place.

a group of interlinked well-structured webpage he's look like this:

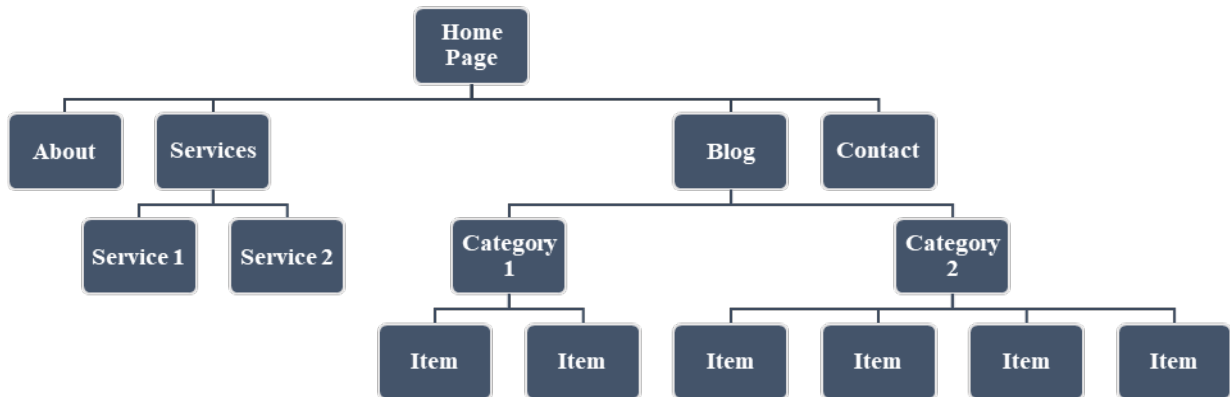


Figure I.8 A group of interlinked well-structured webpage.

We can distinguish between two types of website static website and dynamic website static website.

2.1 Static website

A static website contains only static web page on the server with the format that is sent to a client web browser, a static web page is a document that every time it is requested, display on the browser exactly as it is in the server, it doesn't change at unless if the Creator change it. In general, it coded in Hypertext Markup Language (HTML), Cascading Style Sheets (CSS) or JAVA SCRIPT. Audio or video might also be considered static content if it plays automatically or is generally non-interactive.



Figure I.9 Structure of static website.

2.2 Dynamic website

Dynamic website on the other hands contains dynamic web pages, namely web pages that are generated using scripting languages and programs Interactive with a database on the server side. Programs that add, restrict, edit or remove data without the creator intervention. This is important without the creator intervention the web page may change on the fly based on the user action or his computer information for example, the

current data and time at which you are viewing the web page. The website can display your profile information every time you login.

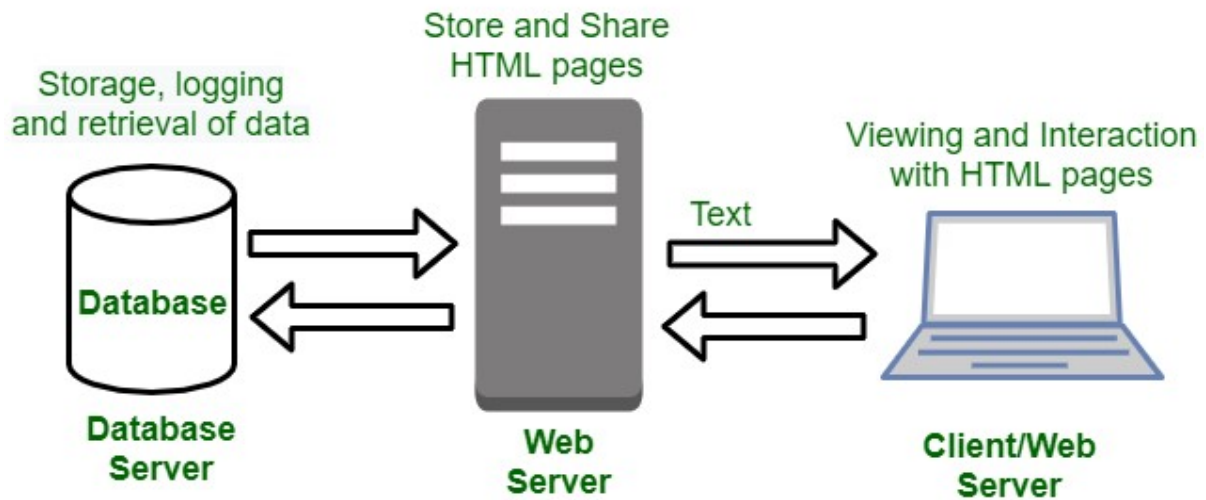


Figure I.10 Structure of dynamic website.[8]

3 Web application

Web application is software program that exists on the server and runs using a web browser through a web page with an active network connection. Unlike computer-based software programs that are run locally on the operating system (OS) of the device. These applications are programmed using a client-server. It means that it has a client-side and server-side. "client" here refers to the program the individual uses to run the application. It is part of the client-server environment, where many computers share information. For example, in the case of a database, the client is the program through which the user enters data. The server is the application that stores the information.[9]

Web application can be developed for many different reasons and used by companies or individuals. Also, employees can collaborate on projects and work on shared documents with web applications. They can create reports, files, and share information from anywhere and with any device.[10]

Web application is similar to desktop apps; it is created using a combination of programming languages and web application frameworks. It may use the RAM, allow user interactivity, and it is designed for many users. For example: Google mail, Facebook, YouTube.[10]

3.1 How does a web application work?

To connect to a web application, you need internet and one of the web browsers like Google Chrome. We have three elements for a web application that require to function:

- **Web server:** to handle request from client.
- **Application server:** to execute the tasks requested.
- **Database:** to stock up information.

There are some steps to show how application works:

- The user creates a request to web server through the application's user interface.
- Web server send request to web application server.
- Web application server executes the requested task after this generates the results of the required data.
- Web application server the results back to web server.
- Web server carries the requested information to the client.
- The requested information appears on the user's display.[10]

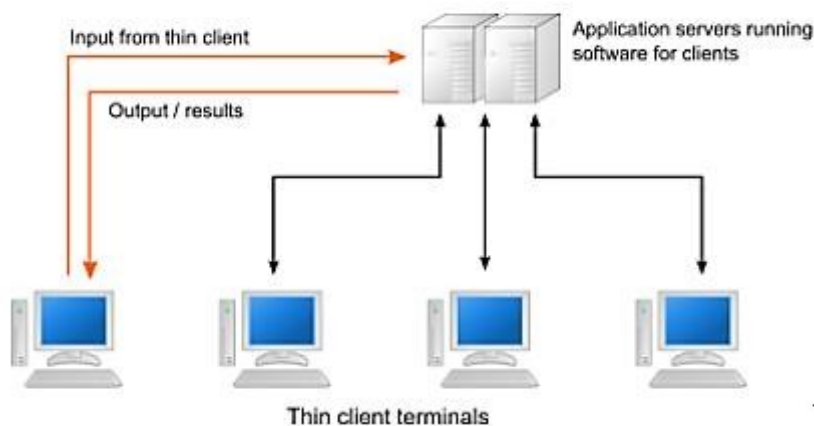


Figure I.11 Structure of web application.

3.2 Function of web application

Web application is a piece of software which can be accessed from a browser, a browser is an application that you use for browsing the internet like Google Chrome or Firefox technically we call a web application a web server although there is a difference in the sense that the web server can host multiple applications. The web server is actually Network application running on some machine, the browser is called web client in fact web client are not only browser but any application which can speak to a web server like URL or Telnet they are also our clients or users agents so the web client or the user region communicate with the web server to get its job done and actually for any communication both the parties need to use the same

set of rules or grammar which we call protocol in technical terms, protocol is when two parties are communicating they need to use the same language and the same set grammar rules so that they can understand each other similarly. In computer science when two application of communicating they need to use the same set of rules which we call protocol and the browser can speak some standard protocols like the HTTP protocol the FTP protocol the WebSocket protocol, and each of these Protocols are suitable for different kind of tasks for example the FTP protocol is used for transferring files, and among all these protocols the HTTP protocol is the merge protocol in the web client and the web server club that is means most of the time the web client and the web server speak to one another using the HTTP protocol.

3.3 Architecture of web application

Is a framework that is comprised of the relationships and interactions between application components, such as middleware systems, user interfaces, and database. Modern application architecture and its development are continuously improving in both of its frontend and backend capabilities. On the backend or server side, there are many application development architecture approaches that are emerging to cope with and solve current development needs such as microservices, serverless architecture and single page application.[11]

3.3.1 Components of web application architecture

There are many components can help web application build its digital makeup, and it is separated in two area the first one is user interface application and the other is structural components.

User interface application components displaying dashboards of web page, logo, notification, configuration setting, and the structural components are web browser or client, web application server and database server.

Web browser or client is interface rendition of a web application functionality which user interact with; this content goes to the client. Web application server might manage business data and can built using PHP, Python, Java, Ruby, Node.js. it's comprised of at least a centralized hub or control center to support multi-layer applications. Database server stock up data for the application, it may also supply the business logic and other information that is managed by the web application server.[12]

3.3.2 Type of web application architecture

We have three types of web application architecture which are Single Page Application, Microservices and Serverless architecture.

a. Single page application SPA

Modern application and efficient are designed to only request the most important elements and information, it interact with user in a more dynamic by providing updated content within the current page, rather than loading entirely new pages from the server with each action from user and all of this helps prevent interruptions in the user experience.[11]

b. Microservices

Microservices architecture framework offers a comprehensive set of advantages to help developers to be more speed up and more productive, they are small, lightweight services, the components in the application are not directly dependent on each other they do not need to be developed in the same coding language. [13]

c. Serverless architectures

Serverless architecture approach leverages third-party cloud infrastructure services, and development entity does not own or manage the backend servers, which makes it simpler when the development company does not want to manage or support the servers and the hardware, they run it.[14]

3.3.3 Web server architecture

Web server architecture is the logical layout or design of a web server, the purpose is to complete requests made by clients for a website. In general, the clients are browser and mobile application that make request with HTTP protocol. The structure is comprised of a physical capacity of the server, storage, memory, performance, operating systems, and network connectivity.[11]

a. Java web application architecture

Java web application architecture has traditionally been a favorite in the enterprise development environment due to its versatility.it is a simple, informative web application and can help achieve successful results. Also, can combine and work with a number of Java native tools to build a web application.[11]

b. Node.js web application architecture

Node.js is written using JavaScript and is the same technology as fronted components, which makes it easier for the same developers to program backend services and fronted user interfaces, and the architecture has become a strong candidate to develop web application. This architecture of Node.js provides coherence, code sharing and reusability, and provide speed and efficiency to the development environment, these benefits bring flexibility and efficiency when building a reliable web application.[14]

c. Ruby on Rails web application development

The Ruby on Rails web application development framework is easy open-source software that place top importance on the developer's happiness, it focuses heavily on productivity and rapid web development, to achieve this productive, Ruby on Rails relies on the (convention over configuration) concept, which are assumptions that are considered the top solution.[11]

This simplifies life for developers because conventions are used to deliberate and make decisions where necessary.[14]

3.3.4 Expertise of web application

Modern application architecture works better on multiple platforms and multiple devices, this is the results of an excellent developed architecture that is the foundation of a successful operability. A healthy web application architecture must be easy to use it and can solve problems, offer a fast response time also use the latest and most secure standards and avoid failures with self-healing capabilities.[14]

3.3.5 Benefits of web application

There is a lot of benefits of web application here is some of them:

- You can access web application anywhere with a web browser.
- You don't need to install it on the hard drive.
- All users can access to the same version because updates are always applied centrally.[8]
- Web application can run on differences platforms regardless of the operating system or device.
- It reduces costs for both the end-user and the business.
- It reduces software piracy in subscription-based web applications.[8]
- Web applications discharge the developer of the responsibility of building a client that is compatible with a specific type of computer or a particular operating system.
- Web application requires less support and maintenance from the business and lower technical requirements from the user's computer.

4 Web application VS website

Web application is based on user engagement, almost all their content is from the end user and also web applications are similar to desktop applications and there are more complicated and need more skills than websites do while dynamic website allow the user to interact and add content but the Creator content is the dominant and of course they are less complicated and need less skilled than web application do.

Conclusion

In this chapter we introduced mobile and web applications and some other concept that seemed important. We had to give some definitions to better understand what coming next in conception chapter.

chapter II. **THE APPLICATION DESIGN**

Introduction

This chapter represents the methodology used for the conception of the mobile and web application using UML diagrams in order to visually represent the system.

1 System architecture

A well-designed architecture is needed for all applications, both complex and simple. Using architecture for mobile application saves time, effort, and costs. A program with a reliable architecture is easier to tweak, test, and debug.

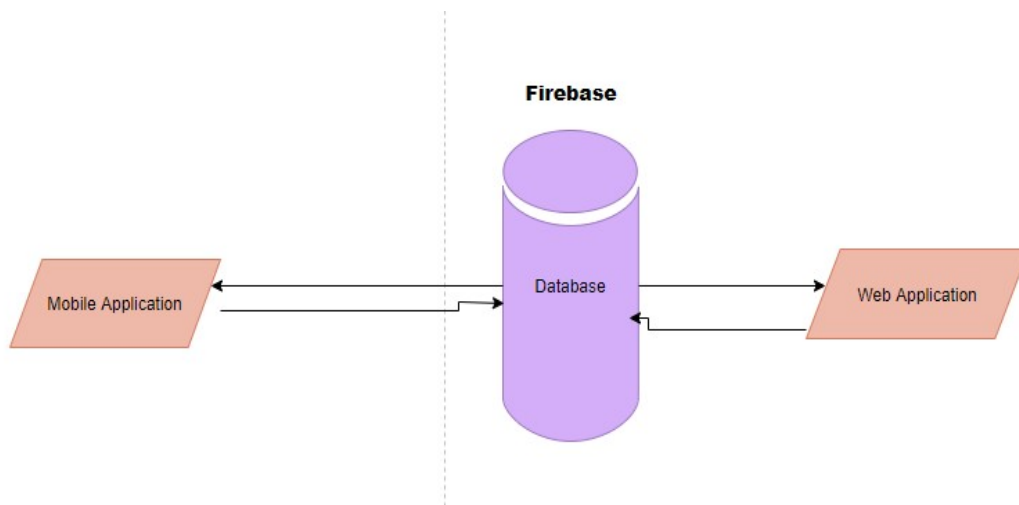


Figure II.1 System architecture

2 Class diagram

2.1 Definition

The UML class diagram is a static structure diagram that define the system architecture by describing and visualizing the multiple aspects of the system and constructing executable code of the application.[15]

The purpose of this diagram is to represent the static aspect of the system which it can be directly translate to the object-oriented language.

2.2 Class diagram structure

UML class diagram is composed of:

- ✓ **Classes:** class describe the behavior of an object, we cannot talk about classes without mentioning objects, we use classes to create objects this is what the whole oriented object is about.[16]

An object is a class instance each object contains the same properties and methods owned by his class.

- Class Notation:

1. Attributes:

- Written in the second partition.
- Attribute type is written after the attribute name.

2. Class Methods:

- Written in the third partition. They are services providing by the class.
- The methods type is written after the method name.

3. Visibility:

- Symbols written before the attributes and the methods to indicate visibility.
- + Public attribute.
- - Private attribute.
- # Protected attribute.

✓ **Relationships:** Relationships define the type of connections between classes.

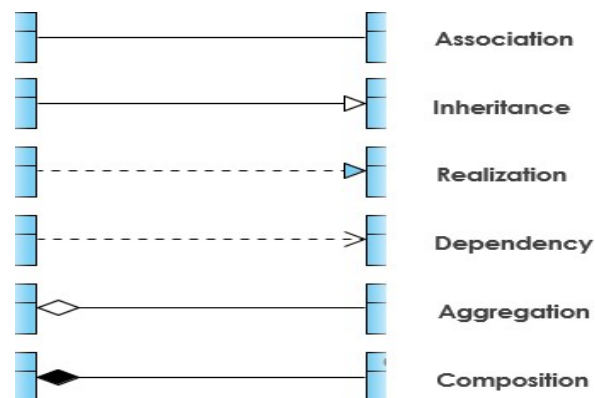


Figure II.2 Connections between classes.

- **Association:** represents the connections between classes.
- **Inheritance:** or as called “is a” relationship the child class is a type of the parent class. The child class inherit all the functionalities of the parent class.

- **Realization:** one entity (normally an interface) defines a set of functionalities as a contract and the other entity (normally a class) “realizes” the contract by implementing the functionality defined in the contract.
- **Composition:** composition represents multiple aggregation relationship. When a composite is deleted, the whole relationships is deleted.
- **Aggregation:** called as “has a” relationship. A class is a collection of another class.
 - ✓ **Dependency:** describe how elements depend on each other.
 - ✓ **Cardinality:**



Figure II.3 Cardinality

2.3 Cheft f tri9y class diagram

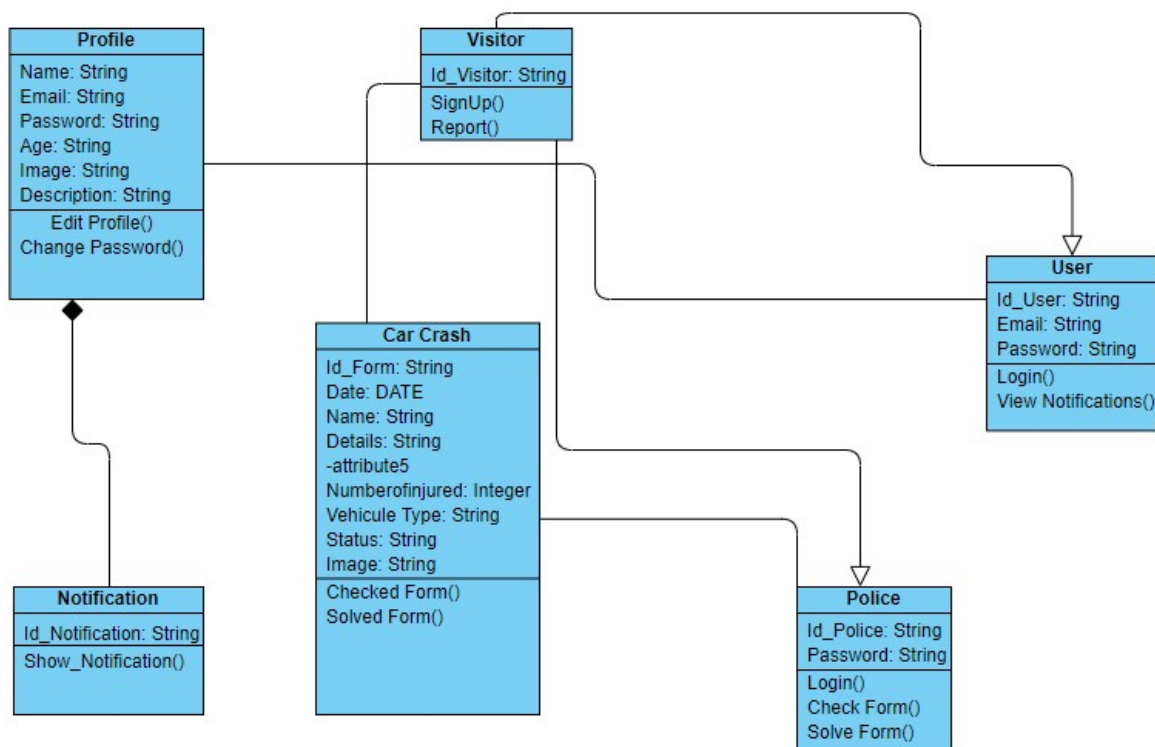


Figure II.4 Cheft f tri9y class Diagram

3 Use case diagram

3.1 Definition

Use case diagram represents all the system's functionalities and demonstrate the way the user interacts with the different parts of the system. It's the ideal tool to specify the system requirements and the multiple events.

✓ *Use case Diagram notation:*

- **System:** Represented with a rectangle that contain Use cases.



Figure II.5 Use case System

- **Use case:** An oval shape with the name of the use case represents à user functionality.



Figure II.6 Use case

- **Actors:** an actor represents a user. It's the element that trigger an interaction with the use case.



Figure II.7 Use case Actors

3.2 Cheft f tri9y use case diagram

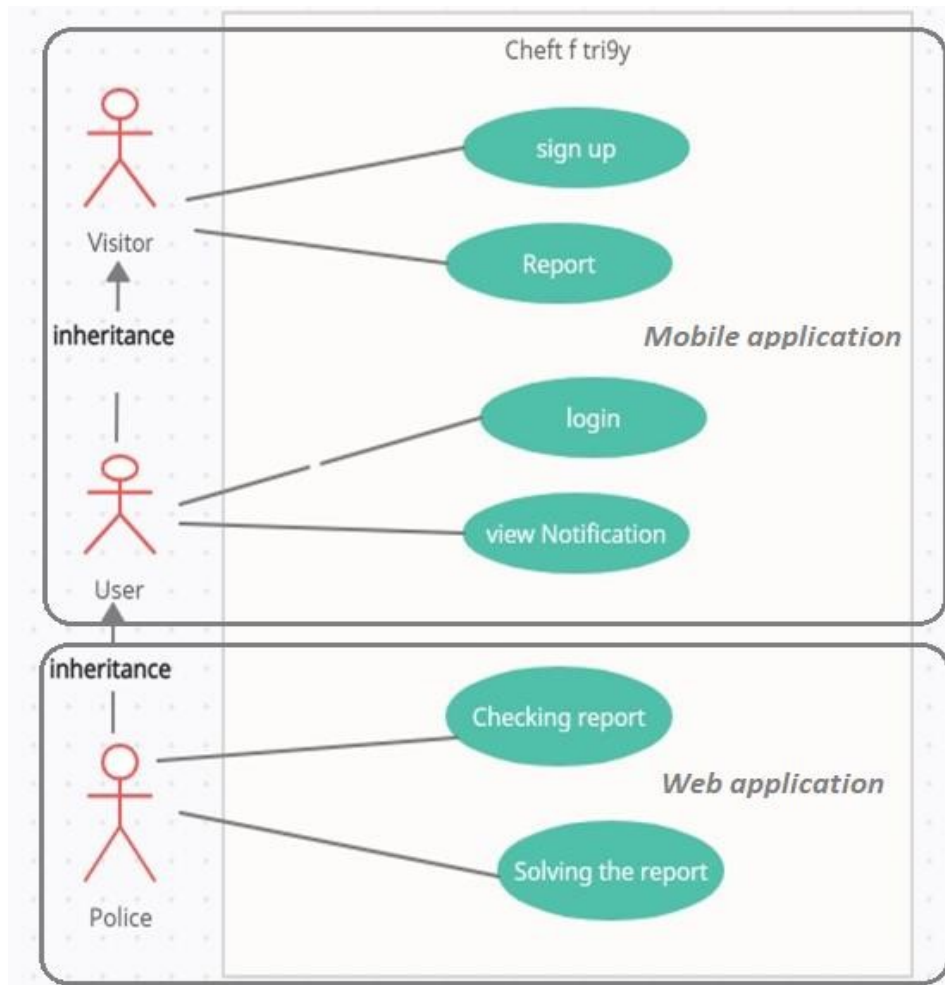


Figure II.8 Cheft f tri9y use case Diagram [17]

4 Sequence diagram

4.1 Definition

Sequence diagram is a behavioral diagram that detail the interactions and their order between different objects arranged in time sequence and defining multiples scenarios.[18]

✓ *Sequence diagram notation:*

- **Actor:** Define a type of role that interacts with the system.

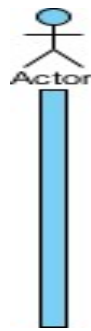


Figure II.9 Sequence diagram Actor

- **Messages:** shows the communication between objects they are represented by arrows; they appear in a sequence order.

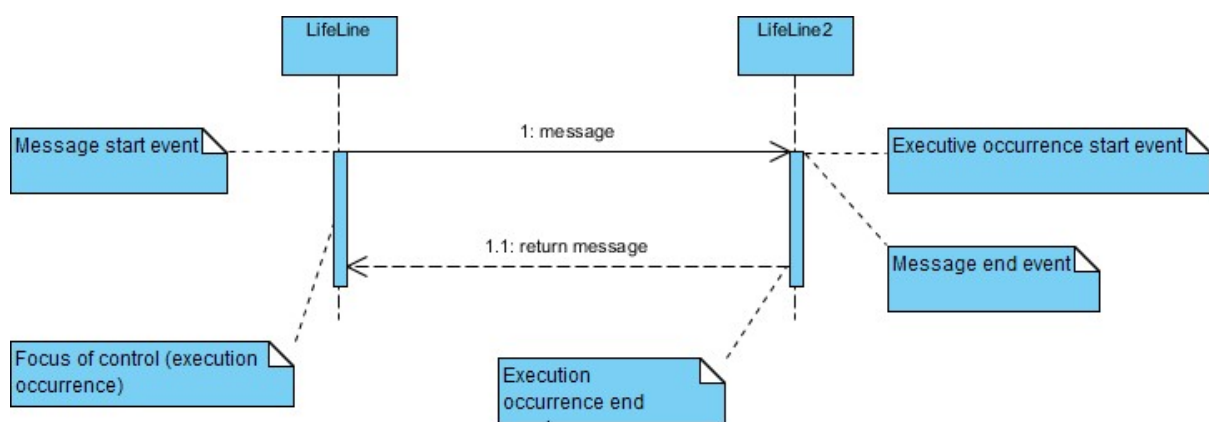


Figure II.10 Sequence diagram Message

4.2 Cheft f tri9y sequence diagram :

4.2.1 Login

a. Mobile application

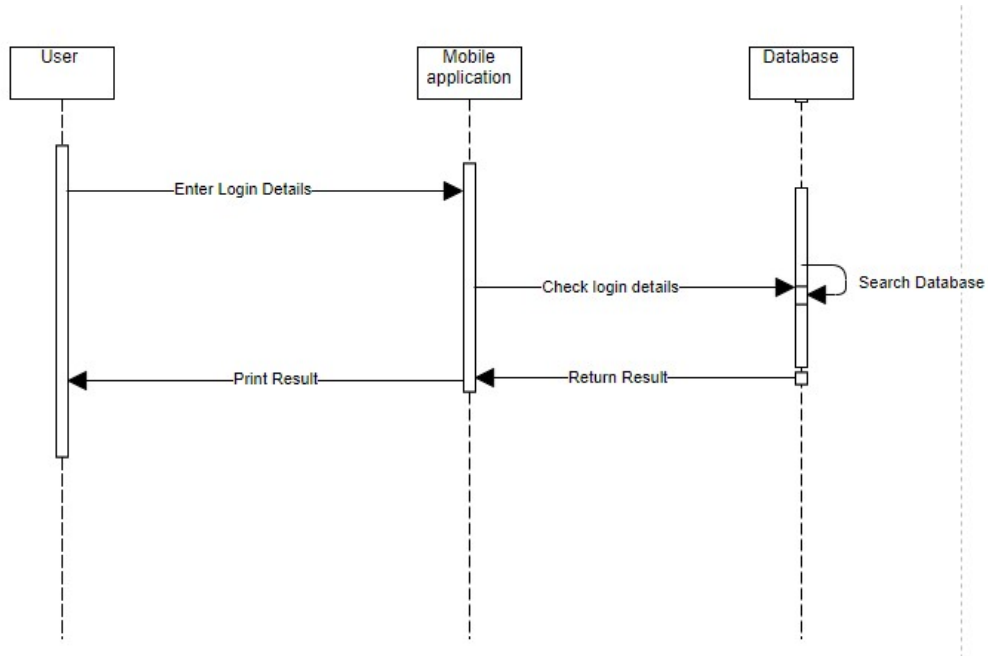


Figure II.11 Cheft f tri9y sequence Diagram Login for mobile application

b. Web application

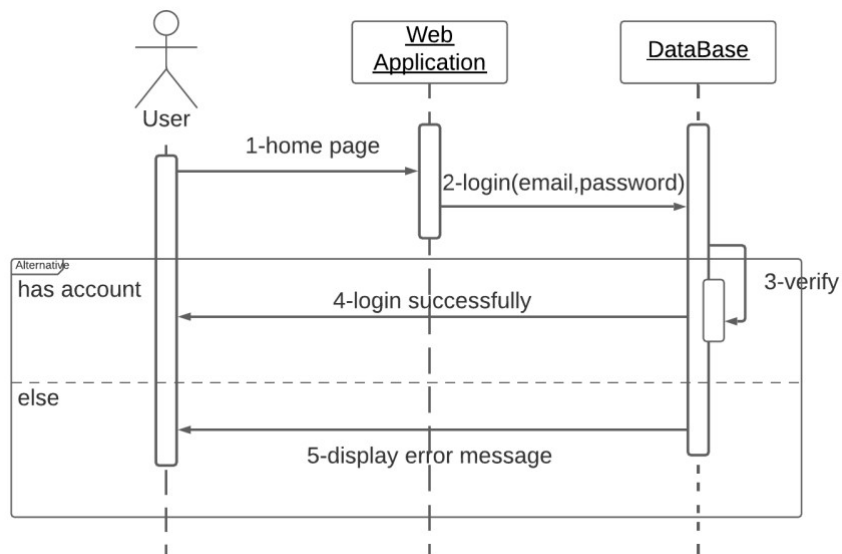


Figure II.12 Cheft f tri9y sequence Diagram Login for web application

4.2.2 Sign up

a. Mobile application

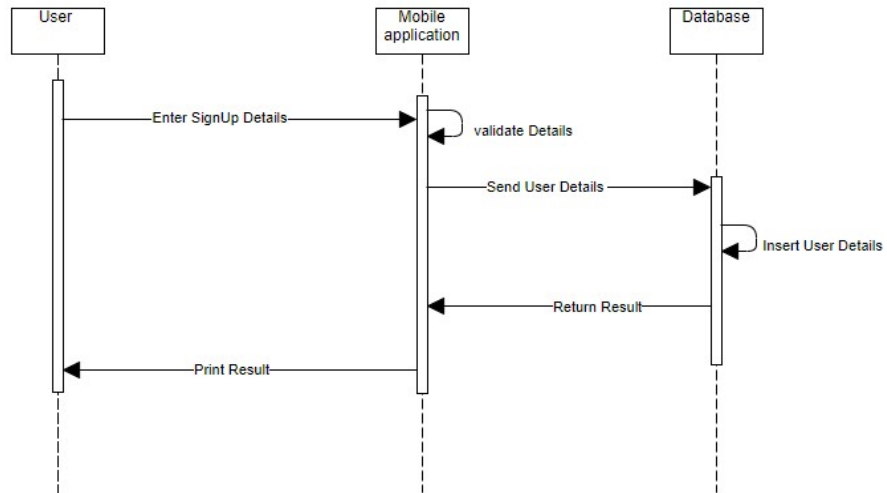


Figure II.13 Cheft f tri9y sequence Diagram Sign up for mobile application

b. Web application

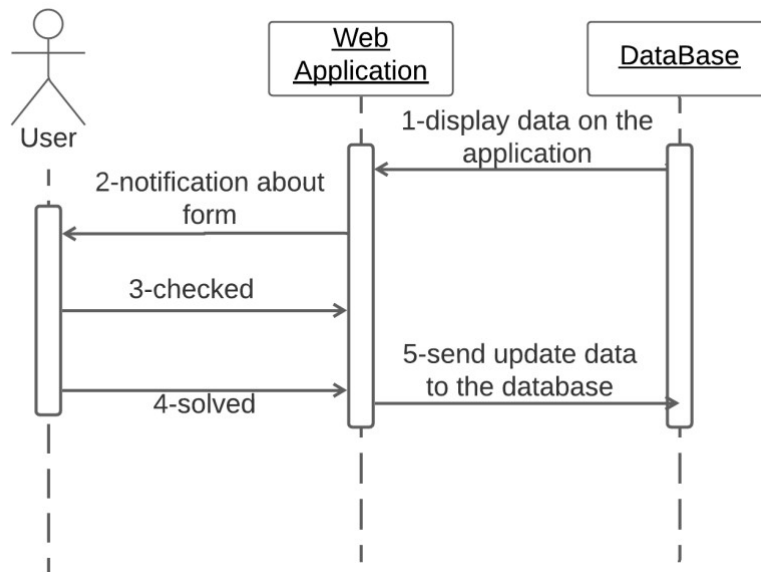


Figure II.14 Cheft f tri9y sequence Diagram Sign up for web application

4.2.3 Report

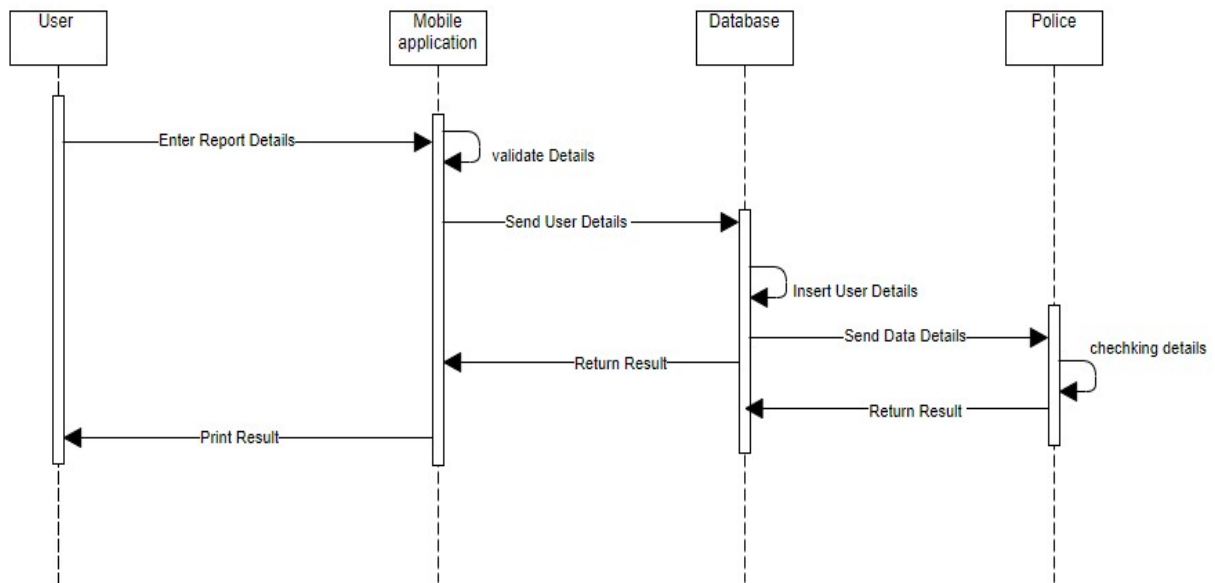


Figure II.15 Cheft f tri9y sequence Diagram report for mobile application

Conclusion

At the end of this phase, we aim to better understand the system functionalities, interactions and flaws by using those visual representations as the proverb says “a picture is worth a thousand words”.

chapter III. THE APPLICATION REALIZATION

Introduction

This chapter will focus on how we developed our mobile and web application. For that we are going to discuss the presentation of the development environment and the description of the applications interfaces.

Lastly, we dedicate the final part to the future work because, and due to timing, our project could not reach the final point that we planned for.

I. Development environment

1 FLUTTER

1.1 Definition

Back in the days of Objective C/Swift and Java/Kotlin as primary languages for mobile development, building apps was expensive. You had to build two separate apps, which obviously meant doing the work twice. But with Flutter the problem is solved Developers are making IOS and Android apps from a single codebase, it was ranked number three most loved toolkit, and in our project here, we have decided to work with some Google's tools which are (FLUTTER and FIREBASE).[19]

1.2 Why FLUTTER?

A lot of times when we talk about flutter, we do it with 4 pillars:

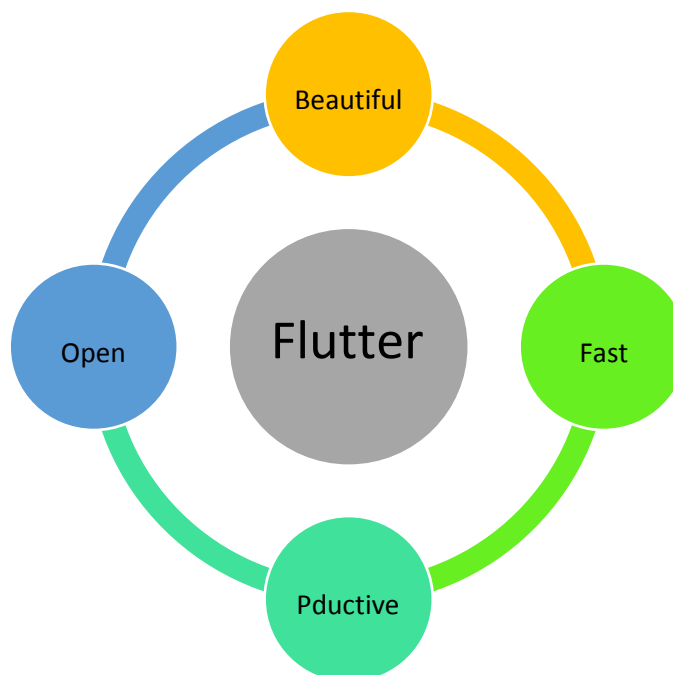


Figure III.1 Pillars of FLUTTER

So let dive into each one of these for little bit:

- 1- The reason Flutter is beautiful is because flutter allows us to paint every pixel to screen, everything in Flutter is widget (2) which is customizable which mean that me as a developer have full control over my application, they also have some widgets that give us, such as the material design in CUPERTINO widget so we make our app look to certain platforms but we can also customize them to make it very brand-driven app as we can see in our application.
- 2- Secondly, Flutter is Fast, Flutter use SKIA, it is an open source 2D graphics library which provides common APIs that work across a variety of hardware and software platforms. It serves as the graphics engine for Google Chrome and Chrome OS, Android, Flutter, Mozilla Firefox and Firefox OS, and many other products. Flutter app render over 60 frames per second, so we can create a highly powered animations that are junk free, we all agree that apps before were a little bit leggy.
- 3- Thirdly, Productive, all developers love is called stateful hot reload, which allows you to make changes to your code and literally in real time see them appear on the application, for example, if our app requires a user to log in, we can modify and hot reload a page several levels down in the navigation hierarchy, without re-entering our login credentials. State is kept, which is usually the desired behavior.
- 4- Finally, Flutter is Open and this is something that's really exciting to us which is free and open source, in fact a ton of community members have contributed a lot of code, packages and plugins to Flutter itself.

Every programming language has its advantages and disadvantages. But, in most cases, we can say that the advantages of Flutter, compared to similar technologies, clearly outweigh its disadvantages.

2 Dart

2.1 Definition

All the UI code In Flutter is written in DART, it is an open-source language, two Google developers took the liberty to create it in 2011, with the aim of allowing developers to use an object-oriented language with static type analysis. As for the syntax, Dart's is very similar to languages like JavaScript, Java and C ++, so learning Dart by knowing one of these languages is a matter of hours. Dart team and flutter team sit together and they work together to re-engineer dart to really work well with flutter. So, dart is a general-purpose client-optimized programming language for faster applications, and we can use it for almost any platform.[19]

2.2 Why Dart?

The Flutter team did not take choosing a programming language for the framework lightly. According to Eric Seidel (lead of the Flutter team in Google), JavaScript was the first choice and dozens of other programming languages were considered before the team settled on Dart.

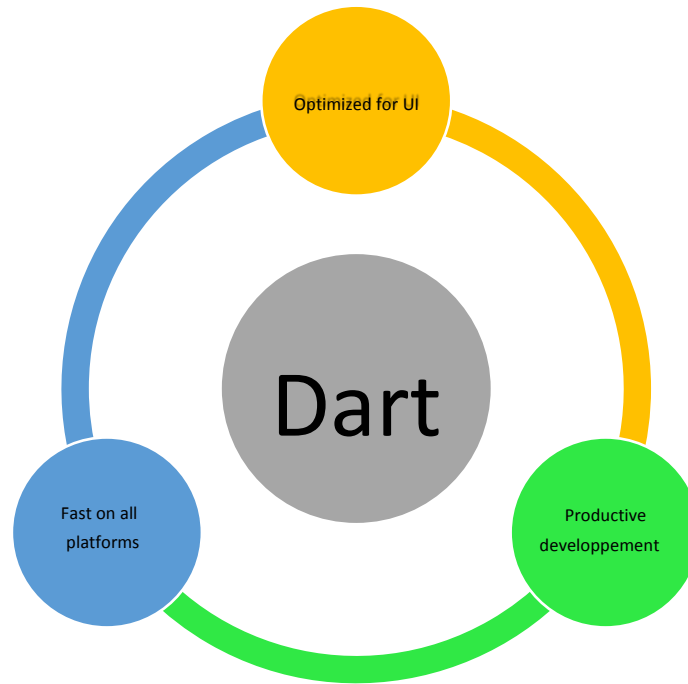


Figure III.2 DART

3 Visual Studio Code

3.1 Definition

Visual Studio Code is a lightweight but powerful source code editor which runs on our desktop, it is open-source and free, it was developed by Microsoft for cross-platform (Windows, MacOS and Linux). It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity). Its first version was in November 14, 2015.

VSC offers different elements that may be Interesting to developers at all levels, of course, compared to other text editors like Brackets the target level is rather advanced. In addition, VSC's advanced editing features can also be used in other areas such as formatting / cleaning up text files or raw data.

3.2 Characteristics

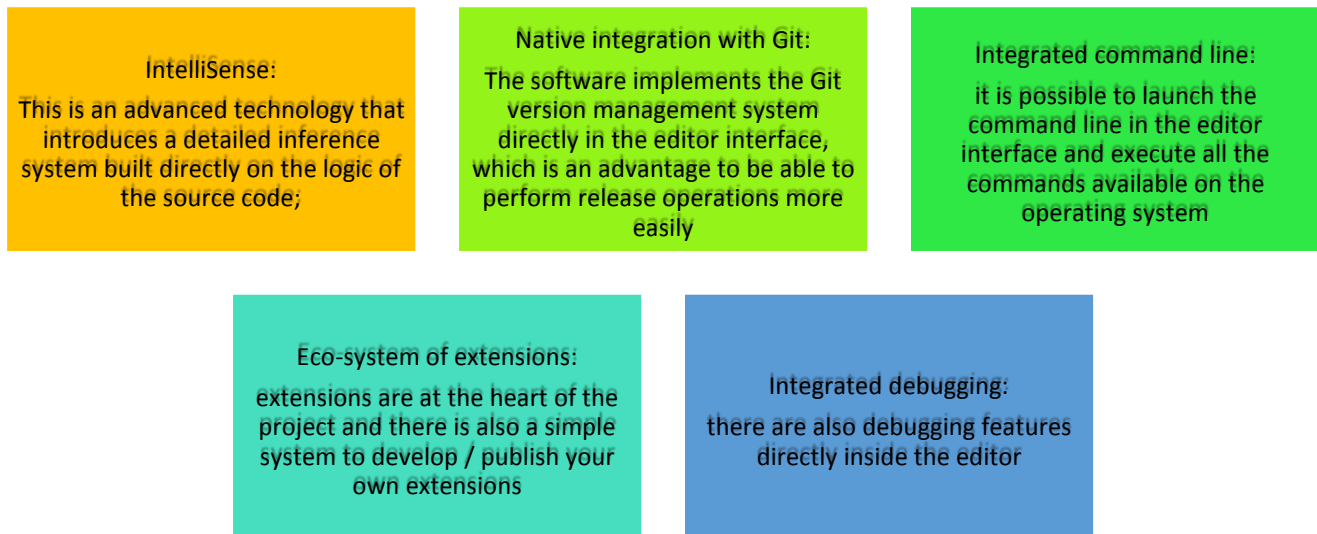


Figure III.3 Characteristic of VSC

3.3 Why VSC?

VSC can be a good starting point for beginners, it is free, open-source, and cross-platform. VSC can be easily installed on Windows, Mac and Linux. There is built-in support for almost all programming languages, including but not limited to HTML, CSS, TypeScript, Java, C++, PHP, Go, SQL, Objective-C, and Ruby. The download is less than 100MB and the space occupied on the disk is about 200MB, the software does not have any particular needs regarding the hardware, it is supposed to work on most machines.

It aims to provide only the tools a developer needs for a fast code-debugging cycle. Overall, Visual studio code is great for working on project like ours.

4 Firebase

4.1 Definition

Firebase created in 2011 by James Tamplin and Andrew Lee, it was a startup offer developers an API to integrate a chat functionality into online on their website. In 2012, Firebase became a company independent providing real-time functionality as a back-end as a service. Then it was acquired by Google in 2014. Firebase is the name of a Google platform that facilitates creation of a back-end that is both scalable and efficient. In another words, it is a platform that allow the rapid development of web and mobile application. Firebase is a Backend-as-a-Service (BaaS) app development platform that provides hosted backend services such as a Real-

time database, cloud storage, authentication, crash reporting, machine learning, remote configuration, and hosting for your static files. [20]

Google Firebase is a Google-backed application development software that enables developers to develop iOS, Android. Firebase provides tools for tracking analytics, reporting and fixing app crashes, creating marketing and product experiment.

4.2 Aim of firebase

The goal for the creation Firebase is to avoid professionals and individuals, to engage in a complex process of creation and maintenance of a server architecture.[20]

4.3 Firebase services

Firebase offers a number of services, including:

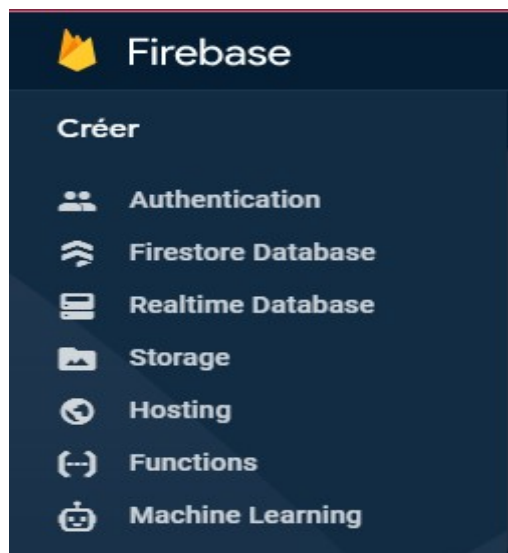


Figure III.4 Services of Google Firebase

- ✓ Authentication – Firebase Authentication makes it easy for developers to build secure authentication systems and enhances the sign-in. This feature offers a complete identity solution, supporting email and password accounts, phone authentication, as well as Google, Facebook, Twitter login and more.
- ✓ Cloud messaging – Firebase Cloud Messaging is a cross-platform messaging tool that lets companies reliably receive and deliver messages on iOS, Android and the web at no cost.
- ✓ Real-time database – the Firebase Real-time Database is a cloud-hosted NoSQL database that enables data to be stored and synced between users in real time. The data is synced across all clients in real time and is still available when an app goes offline.

- ✓ Crashlytics – Firebase Crashlytics is a real-time crash reporter that helps developers track, prioritize and fix stability issues that reduce the quality of their apps. With Crashlytics, developers spend less time organizing and troubleshooting crashes and more time building features for their apps.
- ✓ Performance – Firebase Performance Monitoring service gives developers insight into the performance characteristics of their iOS and Android apps to help them determine where and when the performance of their apps can be improved.
- ✓ Test lab – Firebase Test Lab is a cloud-based app-testing infrastructure. With one operation, developers can test their iOS or Android apps across a variety of devices and device configurations. They can see the results, including videos, screenshots and logs, in the Firebase console.
- ✓ Hosting – Firebase Hosting is production-grade web content hosting for developers. With a single command, you can quickly deploy web apps and serve both static and dynamic content to a global CDN (content delivery network).[21]

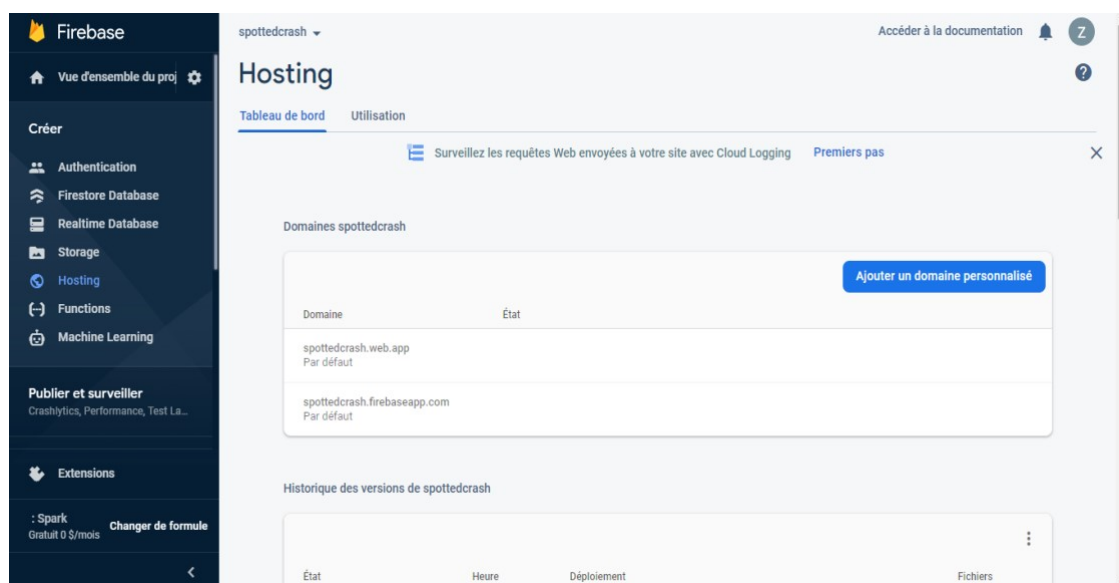


Figure III.5 Hosting

- ✓ Storage – Cloud Storage for Firebase is a powerful, simple, and cost-effective object storage service built for Google scale. The Firebase SDKs for Cloud Storage add Google security to file uploads and downloads for your Firebase apps, regardless of network quality.[21]
- ✓ Firestore database – Cloud Firestore is a flexible, scalable database for mobile, web and server development from Firebase and Google Cloud. Like Firebase Realtime Database, it keeps your data in sync across client apps through Realtime listeners and offers offline support for mobile

and web so you can build responsive apps that work regardless of network latency or Internet connectivity. Cloud Firestore also offers seamless integration with other Firebase and Google Cloud products, including Cloud Functions. [20]

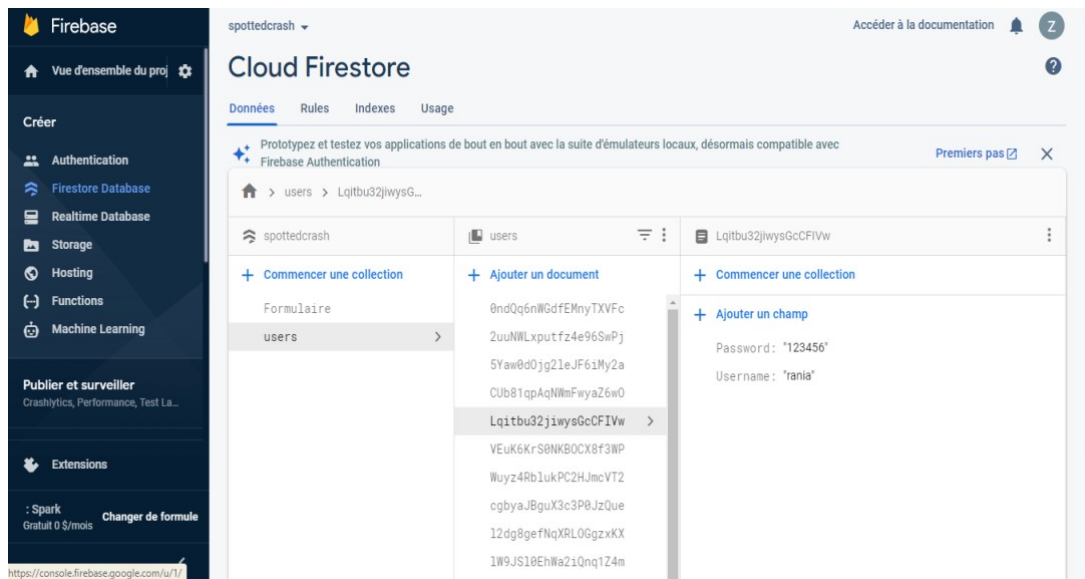


Figure III.6 Firestore database II.

Description of the application interfaces

1 Web application

In this part we are going to see how we created our web application, using firebase for database, and HTML for programming.

1.1 Creation of firebase project

First of all, to create project for web application using firebase we have two parts, the first one is to create firebase project and the second is to follow three steps which are install the Firebase CLI, initialize the project and deploy to our site.

Let's start with the first part which is creation of firebase project, to do this we need to access to www.firebase.google.com and we go to console, like in this figure:

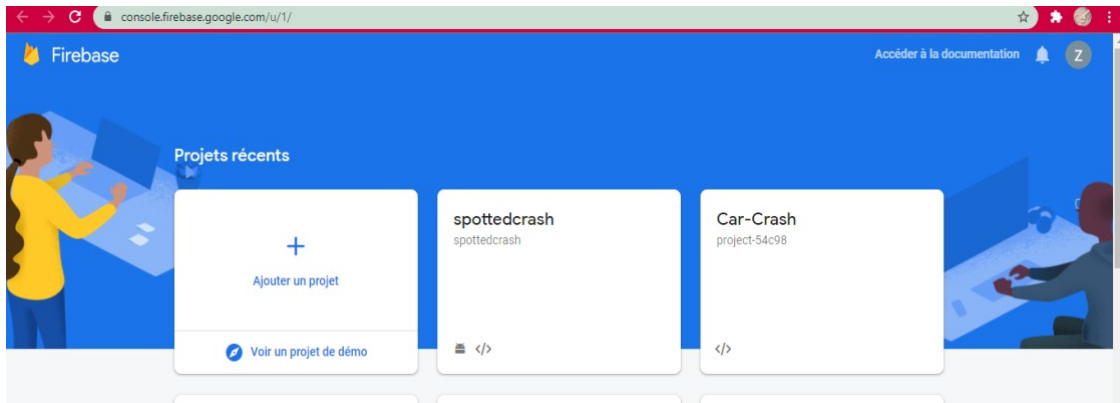


Figure III.7 Creation of new Firebase project

Here we can see all our firebase project, we can create new one, we can delete, also we can modify. In my case I will choose the project “*spottedcrash*” which is the project of our web and mobile application.

To complete the project, we have to go through all these steps which is authentication, Firestore Database, Realtime Database, Storage, Hosting.

1.2 Creation of local project

1.2.1 Install firebase CLI

Firebase CLI is method which firebase use it for initialize any project you want, for change account, for deploy the project and also for emulator to project firebase using command-line of a terminal.

So, for the installation is simple we search for Firebase CLI binary for windows and we download it after this we execute the file and a window like this will appear:

A terminal window titled 'C:\Users\DELL\Downloads\firebase-tools-instant-win.exe'. The terminal output shows a welcome message followed by a series of asterisks. It then prompts 'Let's make sure your Firebase CLI is ready...' and responds with '+ Looks like your CLI is set up!'. Below that, it says 'Already logged in as zouzouzen58@gmail.com' and '+ You can now use the 'firebase' or 'npm' commands!'. A link to the Firebase CLI documentation is provided: 'For more help see https://firebase.google.com/docs/cli/'. The terminal ends with a dashed line and a prompt character.

Figure III.8 Install Firebase CLI

Like you see in the figure above I already login with account of our project I use the command-line `$ firebase login` and I choose the account, and for update our Firebase CLI we need to run the command-line `$ npm install -g firebase-tools`.

1.2.2 Initialize the project

For the second step we need to run this command-line `$ firebase init` to initialize our project, when we run it, we will answer some question like if we are ready to proceed, which directory we want, which features of CLI firebase we want to install it and we choose everything else as default to complete initialization.

```
i Writing configuration info to firebase.json...
i Writing project information to .firebaserc...

+ Firebase initialization complete!

>
```

Figure III.9 Initialize the project

And to connect our local project file to our firebase project we run this command `$ firebase init hosting` from the root of our local project directory.

1.2.3 Deploy the project

In this last step, we have to run the command `$ firebase deploy --only hosting` from the root of our local project directory.

```
=== Deploying to 'spottedcrash'...

i deploying hosting
i hosting[spottedcrash]: beginning deploy...
i hosting[spottedcrash]: found 17 files in public
+ hosting[spottedcrash]: file upload complete
i hosting[spottedcrash]: finalizing version...
+ hosting[spottedcrash]: version finalized
i hosting[spottedcrash]: releasing new version...
+ hosting[spottedcrash]: release complete

+ Deploy complete!

Project Console: https://console.firebase.google.com/project/spottedcrash/overview
Hosting URL: https://spottedcrash.web.app

>
```

Figure III.10 Deploy the project

After we deploy the application, we can access to by URL which we found it in hosting firebase:

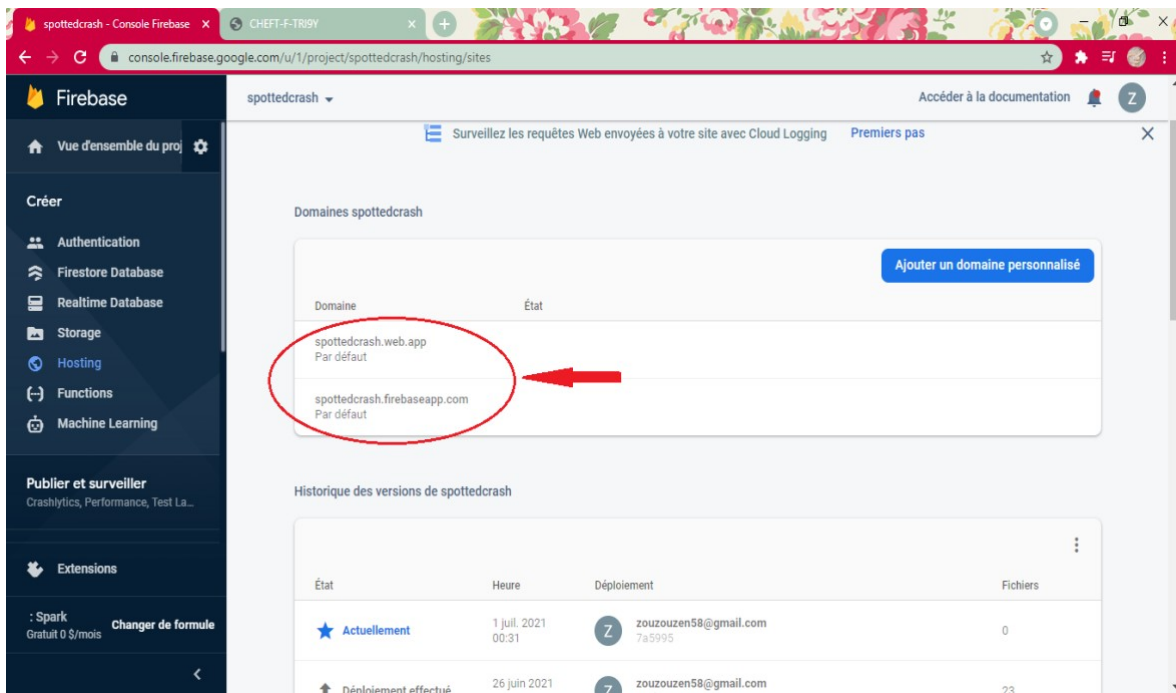


Figure III.11 URL of web application

1.3 Presentation of graphical interfaces of web application

The last step of this web application is to write the code with personal style, the code is written with extension .html and with style CSS.

1.3.1 Home page

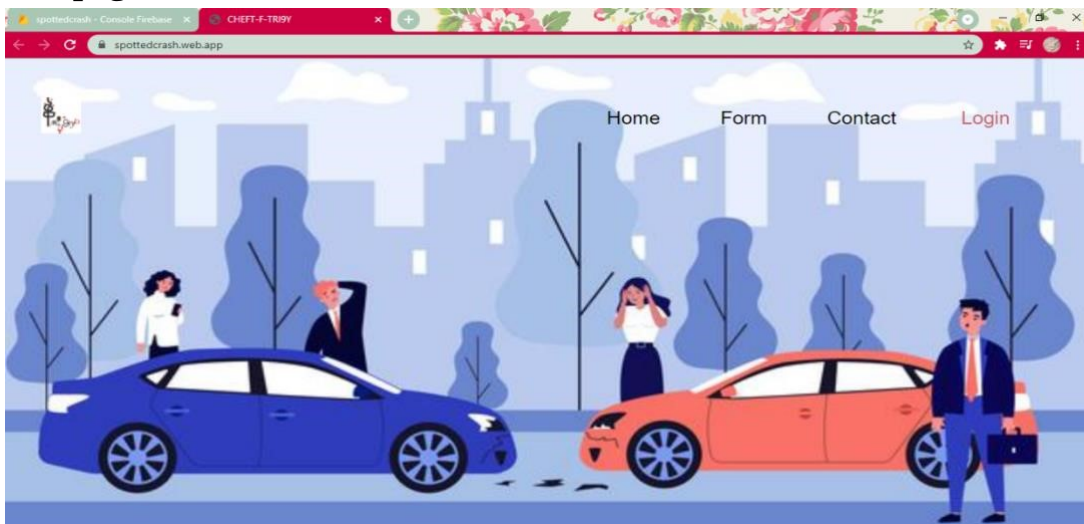


Figure III.12 Home page of web application

This interface shows us home page and it contain links of LOGIN, FORM, CONTACT-US. So, when we click in LOGIN another web page is displayed:

1.3.2 Login page

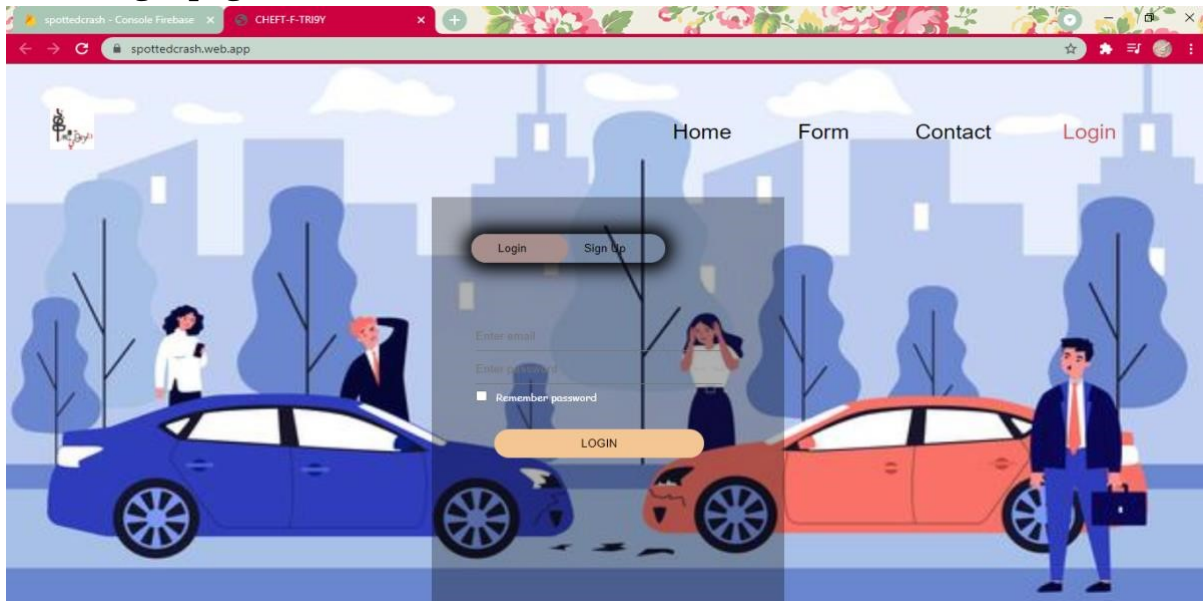


Figure III.13 Login page of web application

In the LOGIN page there is fields for “Email” and “Password” and a button to login.

1.3.3 Sign up page

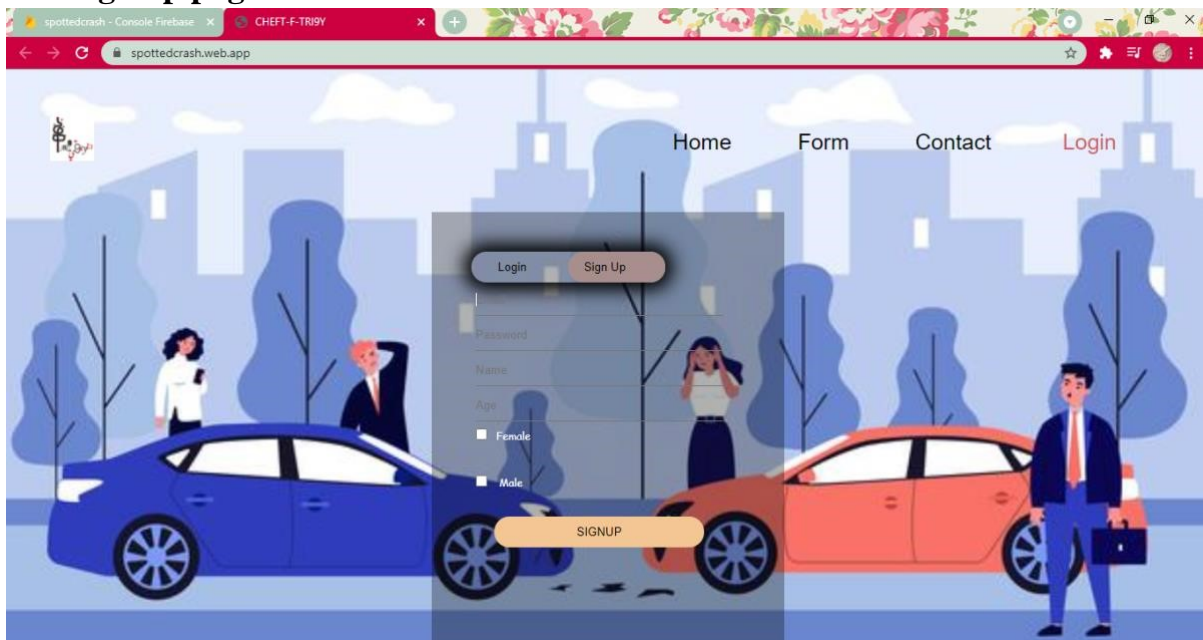


Figure III.14 Sign up page of web application

This is an example of how to create a new account, showing where data are stocked in Firebase.



Figure III.15 Example of new account

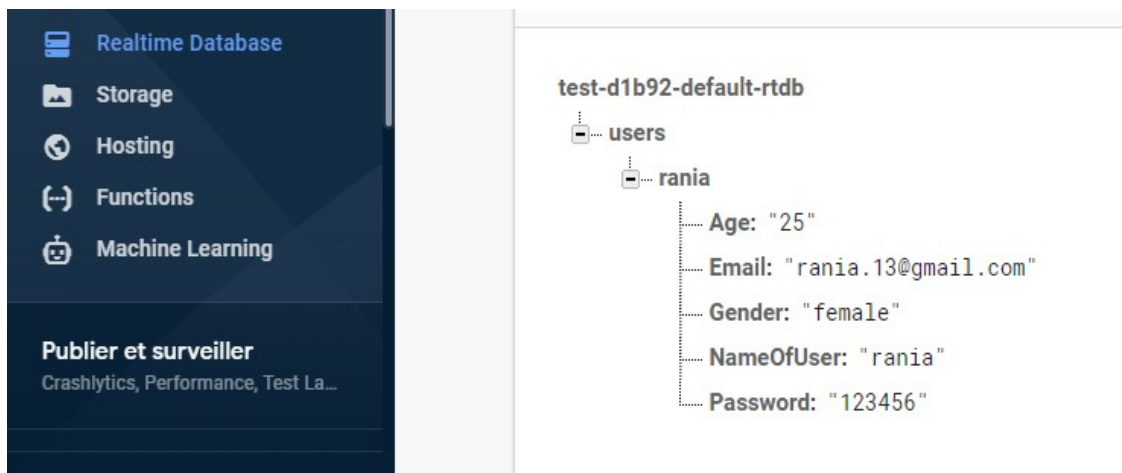


Figure III.16 Storage of Data in Firebase

1.3.4 Contact page

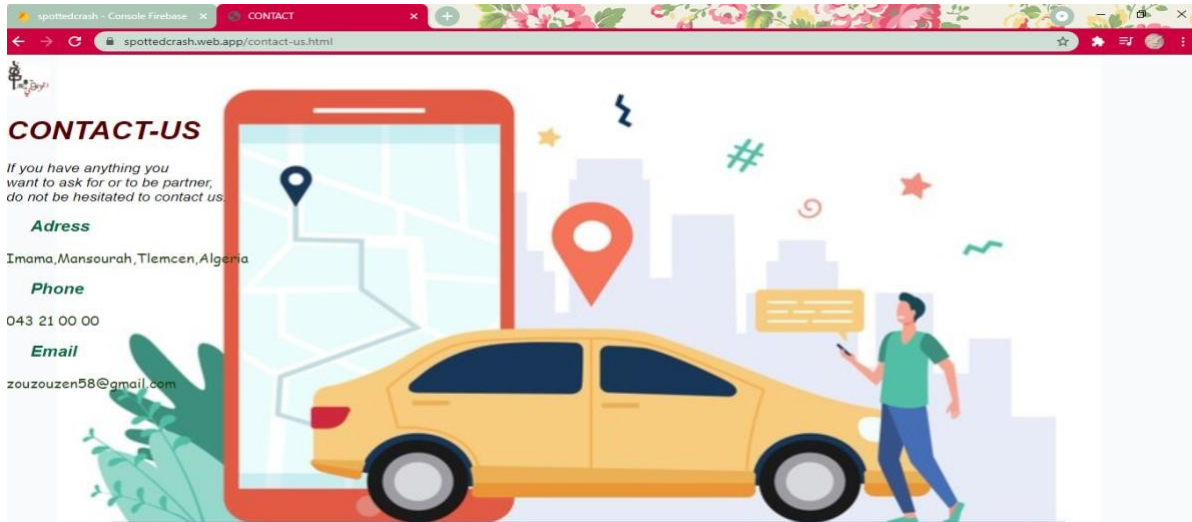


Figure III.17 Contact page of web application

2 Mobile application

2.1 User interfaces

It is important to create a good interface design in order to capture the user attention and facilities the browsing experience for him and creating a visual identity for our app. The application interface contains four main screens.

Each if this screen contains a navigation bar at the top of the screen in order to facility switching between screens.

2.1.1 Home page

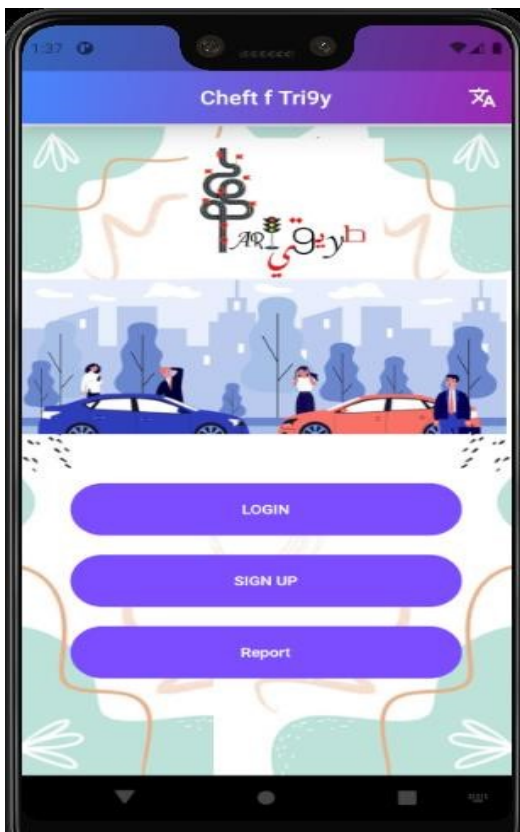


Figure III.18 Home page

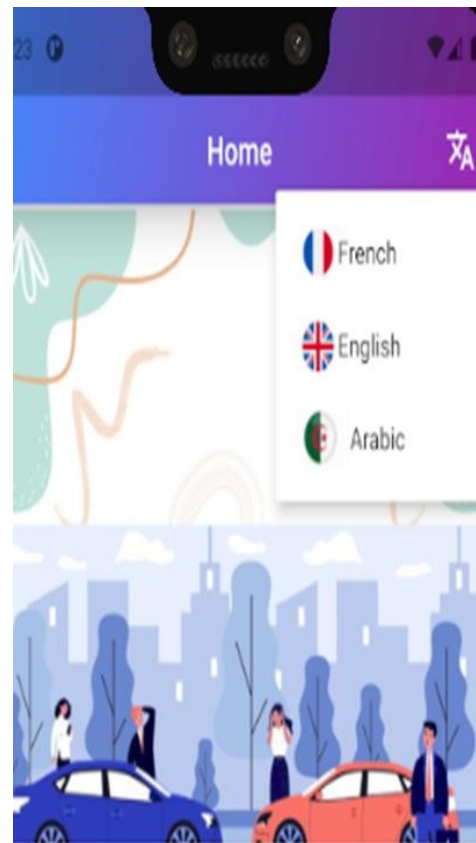


Figure III.19 Language

Home screen is the main screen the device will show, it contains a different buttons LANGUAGE LOGIN SIGNUP REPORT that led to their own screens

2.1.2 Sign up page

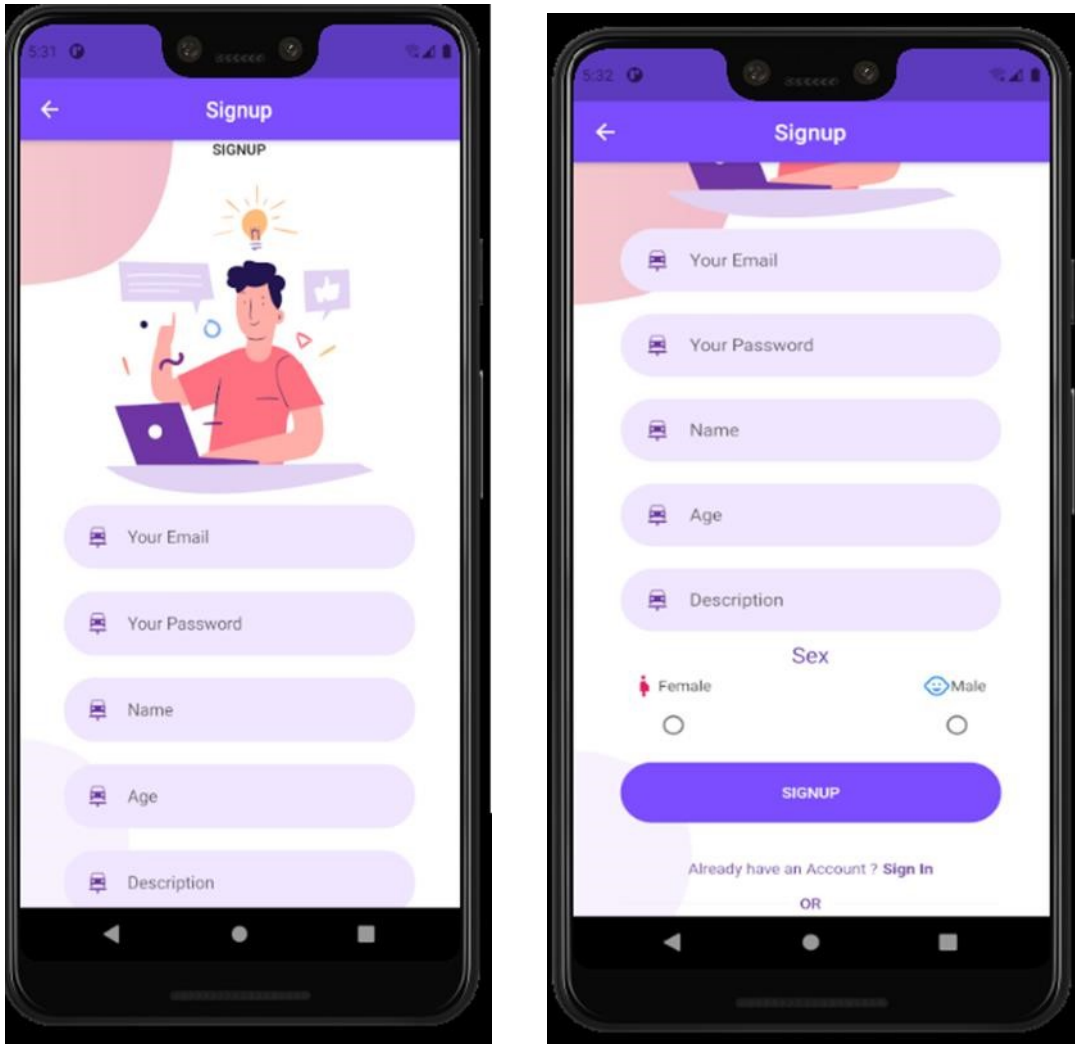


Figure III.20 Sign up page

It contains a list of fields that the user will input his personal data on it and will be stored on our firebase.

This figure shows the firebase part where inputs are stocked.

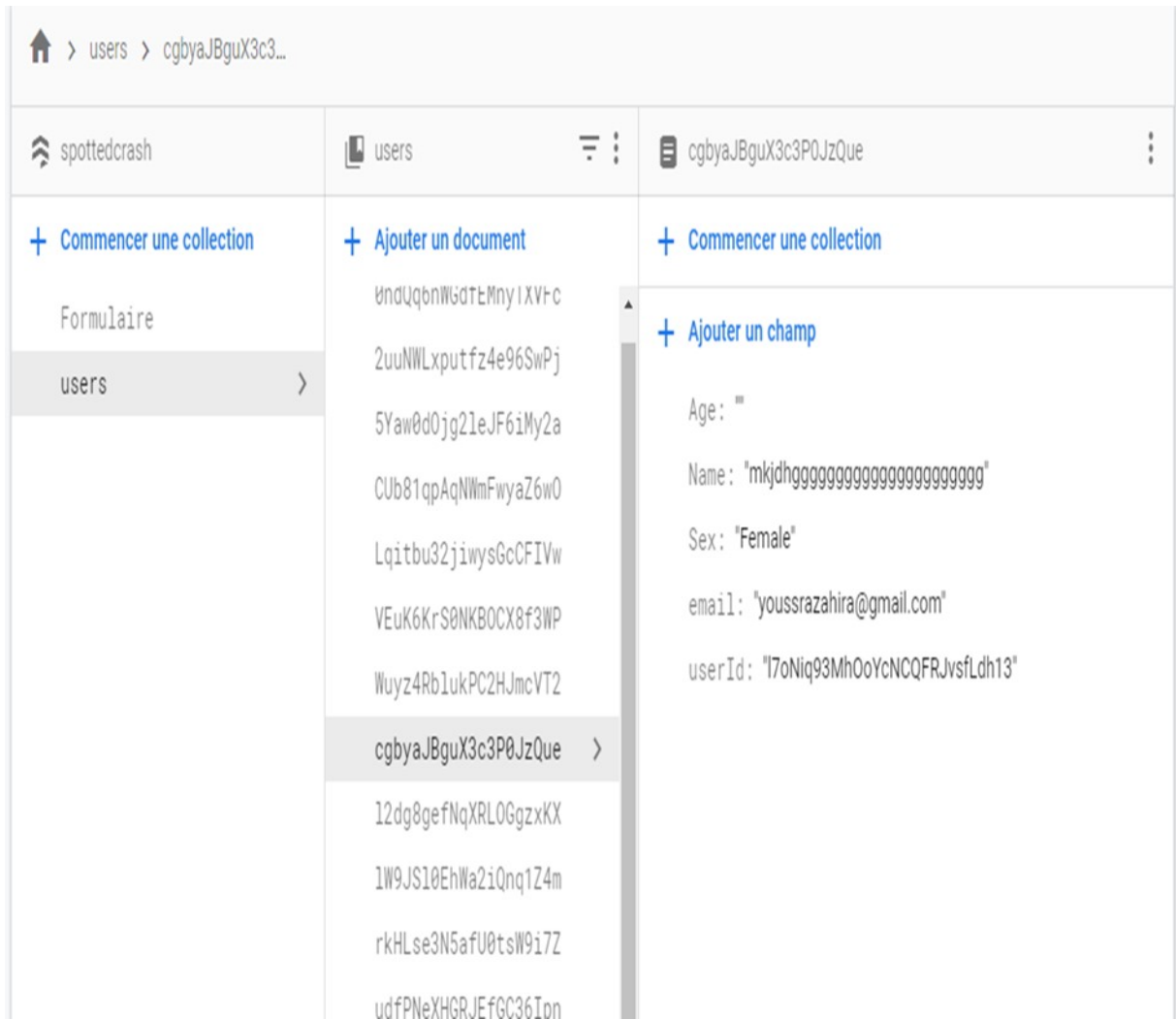


Figure III.21 Storage of information in Firebase

2.1.3 Login page

The user can login after he completes the registration. The Authentication process work by inputting the email & the password.

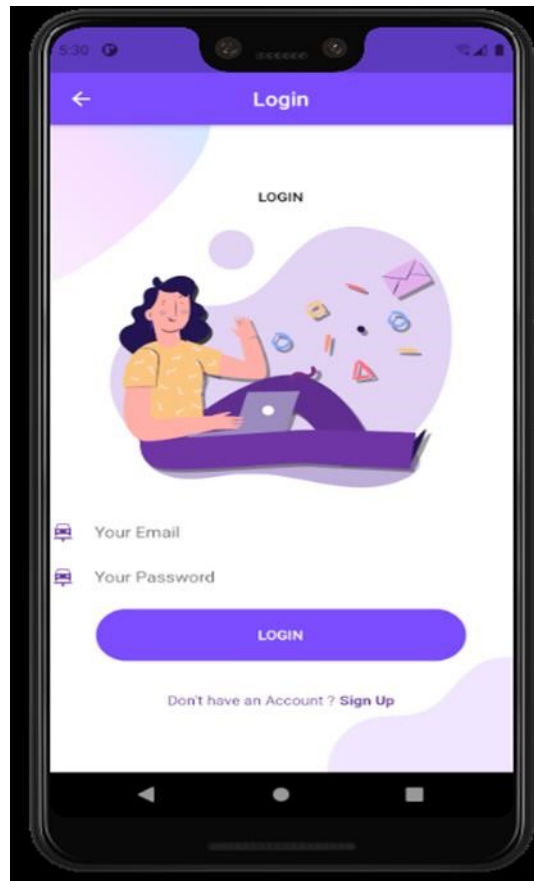


Figure III.22 Login page

This figure shows the code part of exporting data to authenticate.

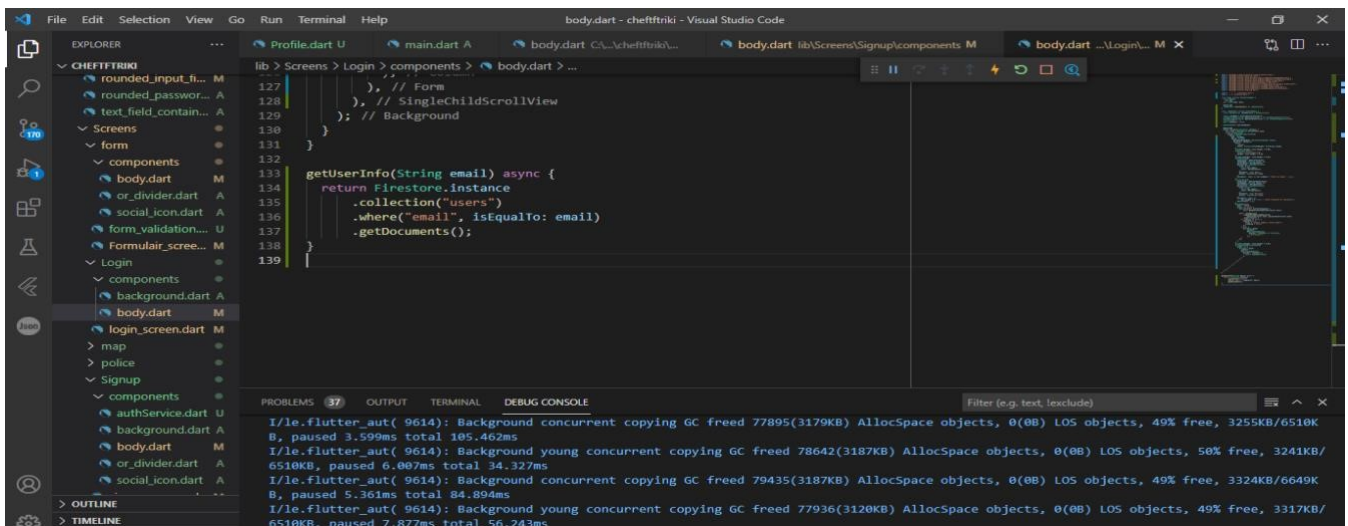


Figure III.23 Code of exporting data to authenticate

2.1.4 Report page

This page allows the visitor or the user of the app to report about the accident and give further details.

This form will be upload on the web police application.

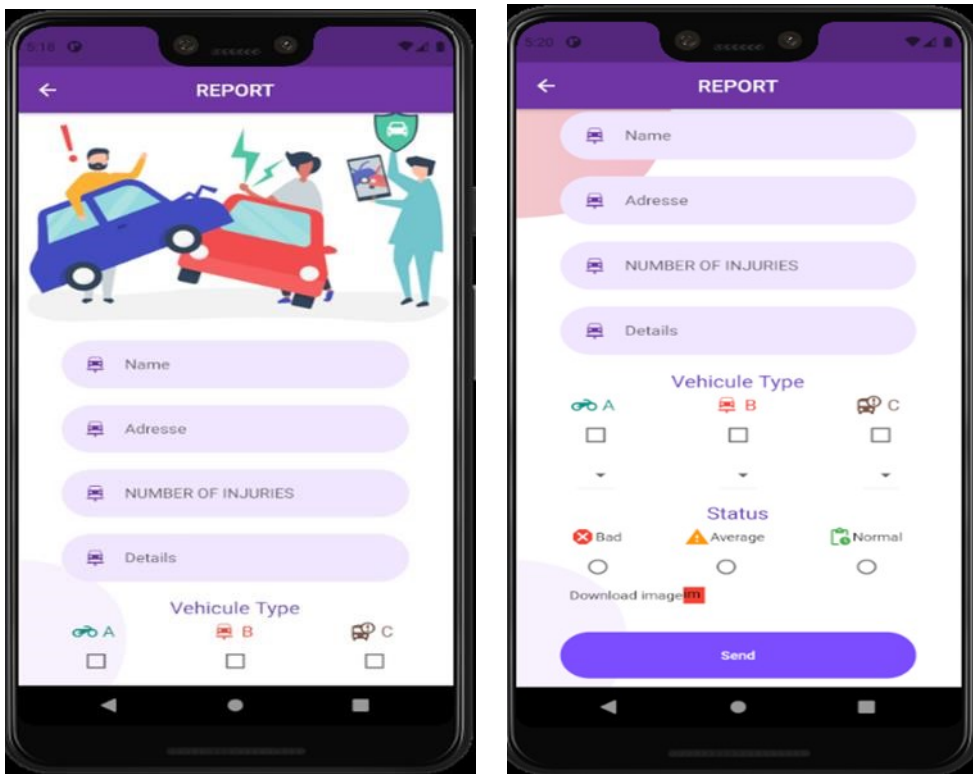


Figure III.24 Report page

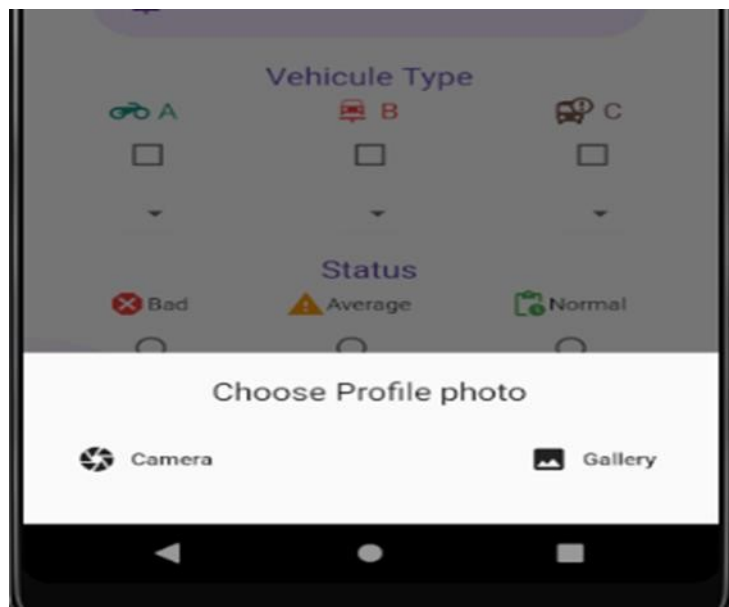


Figure III.25 Report page choosing photos

This figure shows the report details upload on our firebase.

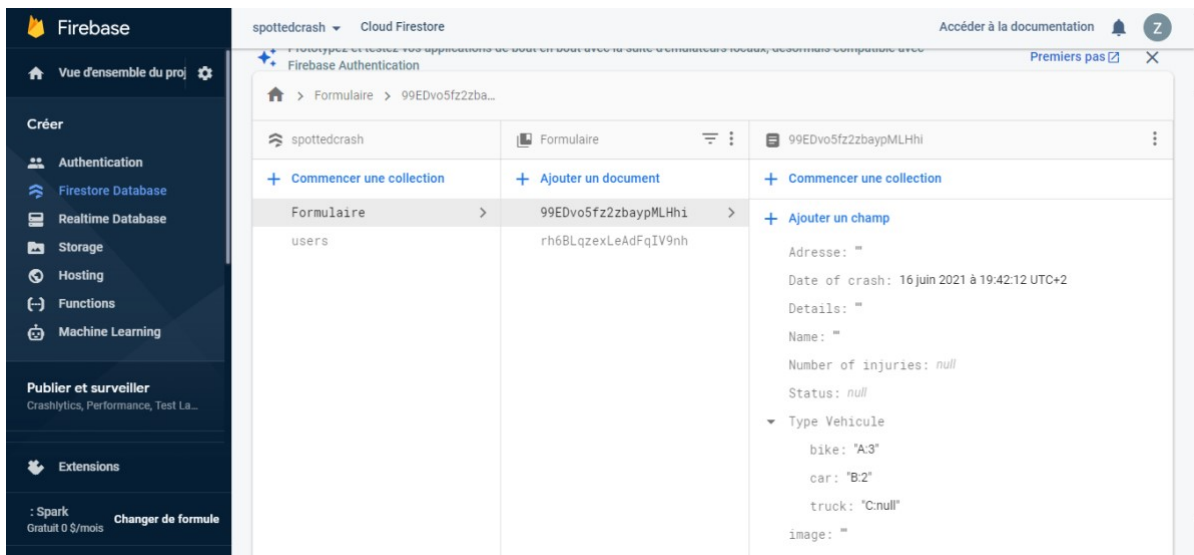


Figure III.26 The report details upload on our firebase.

The code part of exporting report data to firebase.

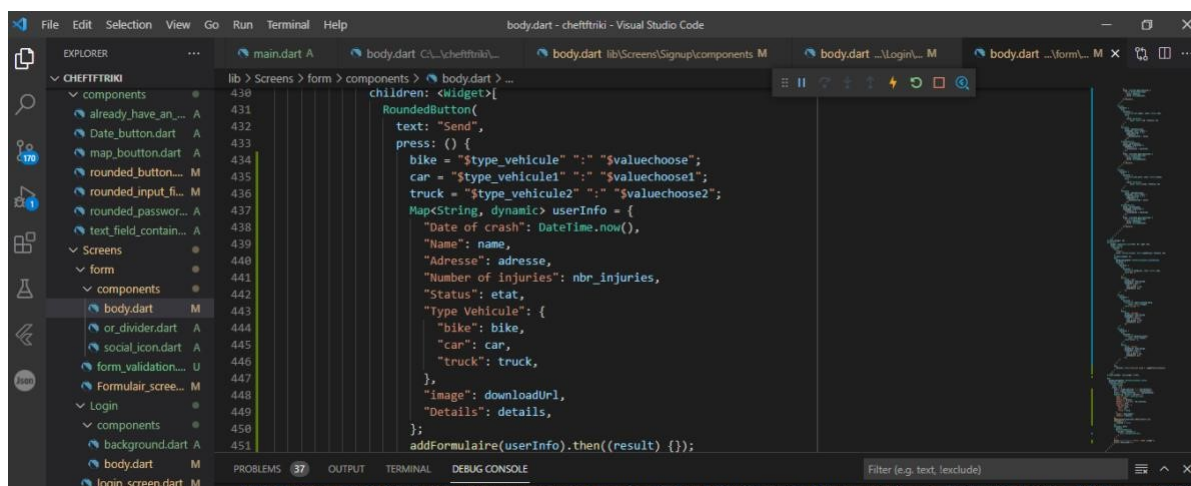


Figure III.27 The code of exporting report data to firebase

2.1.5 Profile edit page

This screen allows the user to edit his profile and show his personal details.

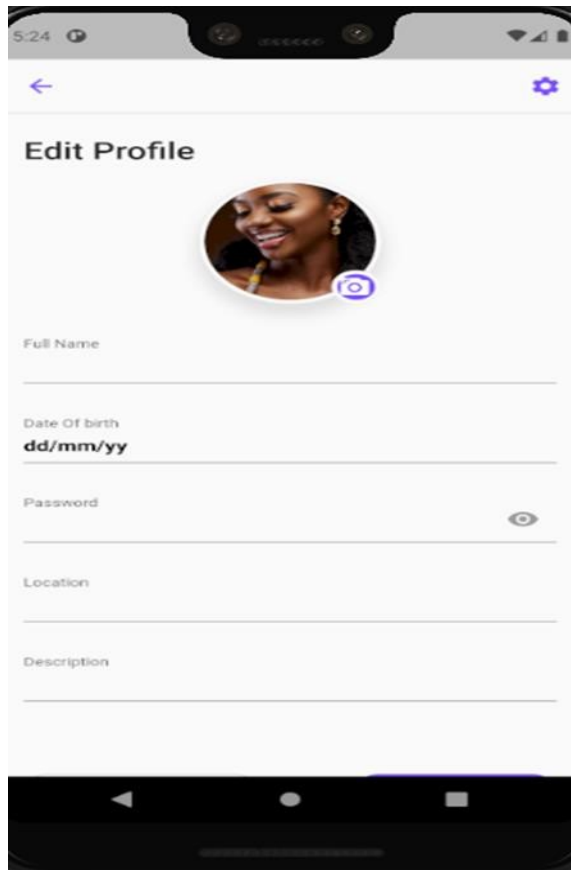


Figure III.28 Profile edit page

III. Future work

This was our target but because of timing we couldn't go that far but as a future work we are planning to add a map in which we can see our position and notifications of accidents if they may be happening around us.

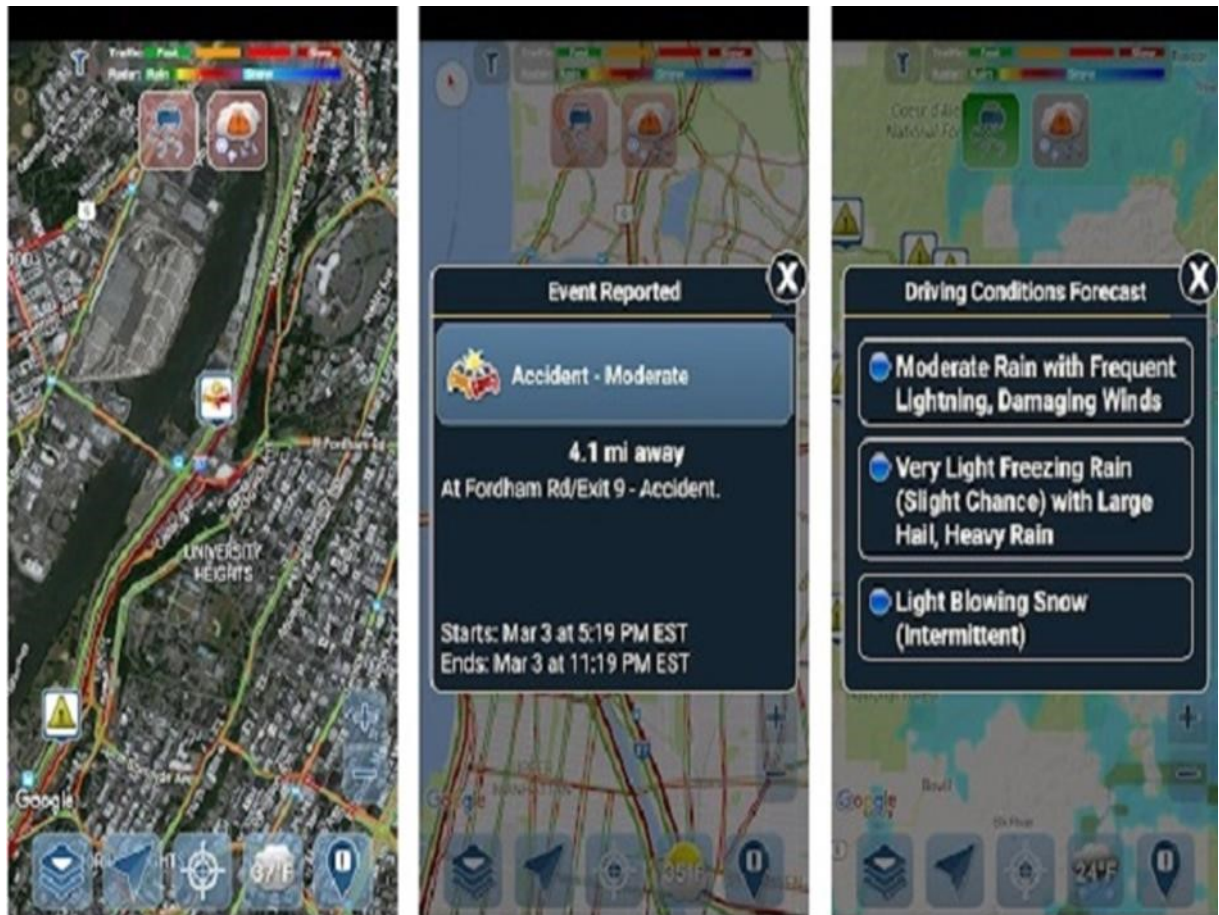


Figure III.29 Future work

Conclusion

Throughout this chapter we have presented our modest work and how we have been able to achieve it by showing some interface of our application and some code to better put you in the image, and we have also showed you some of our future work.

GENERAL CONCLUSION

It was a risk we were happily ready and willing to take, digging into this unknown new world of application development. Frankly, acquiring all the knowledge needed to complete this thesis in a limited amount of time has been such a challenge for us.

During the process of completing this thesis, we have been exposed to new things and information, starting with the flutter kit and the DART programming language. All the tools used to accomplish this project were a new discovery to us.

We have successfully created an application of immeasurable value to human society; an easy and fast application that operates on both platforms IOS and ANDROID.

Our journey in the application development world does not end here; our thirst for knowledge in this field has not been satisfied since we sadly were not able to achieve the initial goal that we have set for ourselves at the start. However, working on this thesis has started a flame of passion in us to learn more and acquire new skills in this domain, and realizing our commencing objective would be the first thing we will work on to do so.

BIBLIOGRAPHY

- [1] https://en.wikipedia.org/wiki/Mobile_app, accessed 22/06/2021
- [2] <https://www.businessofapps.com/data/app-revenues/>, accessed 17/06/2021
- [3] https://en.wikipedia.org/wiki/Android_operating_system, accessed 9/06/2021
- [4] <https://en.wikipedia.org/wiki/IOS> , accessed 5/06/2021
- [5] <https://www.quora.com/What-is-a-windows-operating-system-1>, accessed 2/06/2021
- [6] https://gs.statcounter.com/os-marketshare/mobile/algeria?fbclid=IwAR0fGbiDY1h_B5jYXhdfBIuiuxZXZyD8_ijpniBi1_KhjrB5_rQbuqjvG4nA, accessed 22/06/2021
- [7] <https://www.computerhope.com/jargon/w/webpage.htm>, accessed 15/06/2021
- [8] <https://www.geeksforgeeks.org/dynamic-websites/> , accessed 25/06/2021
- [9] <https://lenac1884.medium.com/know-all-about-web-applications-including-its-benefitsand-working-230f513641f2>, accessed 18/06/2021
- [10] <https://www.indeed.com/career-advice/career-development/what-is-web-application>, accessed 16/06/2021
- [11] <https://svitla.com/blog/web-application-architecture> , accessed 16/06/2021
- [12] <https://engineering.csiworld.in/web-application-architecture-part-1-guide-to-becomefull-stack-developer-cc9526a3519b> , accessed 17/06/2021
- [13] <https://svitla.com/blog/microservices-architecture>, accessed 16/06/2021
- [14] <https://datarob.com/what-is-web-application-architecture/>, accessed 15/06/2021
- [15] <https://www.smartdraw.com/uml-diagram/>, accessed 16/06/2021
- [16] <https://tallyfy.com/uml-diagram/>, accessed 16/06/2021
- [17] <https://www.uml-diagrams.org/use-case-diagrams-examples.html>, accessed 16/06/2021

[18]<https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/> , *accessed 16/06/2021*

[19]Julien TRILLARD,<Flutter Développez vos applications mobiles multiplateformes avec Dart>,edition ENI,june2020.

[20]<https://blog.stackpath.com/web-application/>, *accessed 20/06/2021*

[21] FERDI Soria, RAIS Soumia, The design and production of a mobile application for cloud-based patient medical record management. Memory of Master Telecommunications, Networks and Telecommunications. University of Tlemcen, August 2020.

[22]<https://firebase.google.com>, *accessed 26/06/2021*