



Peoples Democratic Republic of Algeria  
Ministry of Higher Education and Scientific Research



AKLI MOHAND OULHADJ UNIVERSITY of BOUIRA

Faculty of Sciences and Applied Sciences

Computer Science Department

# License memory

In Computer Science

*Specialty: ISIL / SI*

## Theme

---

Computer Architecture Interactive Course Website

---

Supervised by

- DR. TAHA ZERROUKI

Realized by

- GRINE Lyes
- LAMRI Farouk
- MIDOUN Mohamed

2021/2022

# *Acknowledgement*

First of all, let's thank Allah who has been giving us some mercies and blessing so that we have done this project without any troubles at all. Secondly, may prayers and peace be upon our Prophet Muhammad who has guided us from the darkness to the lightness in the world as well as in the next world.

Thank you our supervisor, Dr. TAHA ZERROUKI, for your patience, guidance, and support. We have benefited greatly from your wealth of knowledge and meticulous editing. We are grateful that you took us on as the team of this project and continued to have faith in us over the years.

Thank you our colleagues involved directly or indirectly with encouraging words and thoughtful, detailed feedback have been very important to us.

Thank you to everyone, who so generously took time out of their schedules to review our project and gave us constructive feedback.

Thank you to our parents, for your endless support. They have always stood behind us, and this was no exception. Thank you for all of your love and for always reminding us of the end goal.

# *Dedications*

First, I would like to dedicate this work to the most important people in my life, to the ones who supported me and guided me diligently throughout this work and throughout my whole life, to my mother and my father. I also dedicate this work to my brother and sisters, and of course to my team, Farouk and Mohammed, with whom I was able to accomplish this work.

*Grine Lyes*

# *Dedications*

I dedicate our work to my family and many friends. A special feeling of gratitude to my loving parents whose words of encouragement and push for tenacity ring in my ears. My Brothers SIDALI and ABD EL MONEIM have never left my side and were very special. I also dedicate this project to my many friends and university's family who have supported me throughout the process. I will always appreciate all they have done, especially AZZEDINE MOULAI for helping me develop my skills, NOUREDDINE and HITHEM for the many hours of study adventures, and IMAD KHIDER and OUSSAMA for sharing an amazing academic years. DR. TAHA ZERROUKI – I would like to thank you for your support and encouragement to pursue this work. Thank you for all the academic help you have provided me with throughout the last three years at UAMOB. I dedicate this work and give special thanks to my best friends LYES GRINE and MOHAMED MIDOUN for being there with me and sharing this entire Bachelor's degree program. Both of you have been my best companions.

*Lamri Farouk*

# *Dedications*

My dedications go particularly to my dear parents, as well as my family, And everyone who helped me to reach this level of success achieved, Docent Professor Taha Zerrouki our guide in this Pleasant work, and my Duo Ilyes Grine and Farouk Lamri, also every Colleague involved directly or indirectly. Not to forget everyone who encouraged me to go all the way long Along with a lot of respect to every employee at the University of AMO.

*Midoun Mohamed*

## ملخص

تساعد الدروس التفاعلية الطلاب على فهم دروسهم بشكل أفضل من خلال الاختبارات والأمثلة التفاعلية التي تختبر معرفتهم وتساعدهم على الفهم الأفضل للدروس .

وقد لاحظنا أن مادة بنية الآلة تفتقر إلى مثل هذه التطبيقات خاصة وأن أغلب الطلبة يمتلكون الوسائل التكنولوجية التي تسمح بذلك من هواتف نقالة وحواسيب لا سيما أنها أصبحت جزء لا يتجزأ من الحياة الاجتماعية اليوم .

ما ألهمنا إلى إيجاد وتجسيد فكرة المشروع .

## Abstract

Interactive courses help students better understand their lessons through interactive quizzes and examples that test their knowledge and help them better understand the lessons.

We have noticed that the subject of the computer architecture lacks such applications, especially since most students have the technological means that allow it, such as mobile phones and computers, especially since they do today an integral part of social life. That is what inspired us to find and realize the idea of the project.

## Résumé

Les leçons interactives aident les élèves à mieux comprendre leurs leçons grâce à des tests interactifs et des exemples qui testent leurs connaissances et les aident à mieux comprendre les leçons.

Nous avons remarqué que le sujet de la structure des machines manque de telles applications, d'autant plus que la plupart des étudiants possèdent les moyens technologiques qui le permettent, tels que les téléphones portables et les ordinateurs, d'autant plus qu'ils font aujourd'hui partie intégrante de la vie sociale. Ce qui nous a inspiré pour trouver et concrétiser l'idée du projet.

# Contents

<b>Contents</b>	<b>i</b>
<b>Table of figures</b>	<b>iv</b>
<b>List of tables</b>	<b>v</b>
<b>Abbreviations list</b>	<b>vi</b>
<b>General Introduction</b>	<b>1</b>
<b>1 Preliminary study</b>	<b>3</b>
1.1 Introduction . . . . .	3
1.2 General Definitions . . . . .	3
1.2.1 Computer Architecture . . . . .	3
1.2.2 Interactive Course . . . . .	4
1.2.3 Responsive Web Design . . . . .	4
1.2.4 Website . . . . .	4
1.3 Study of the existing . . . . .	5
1.3.1 Structure of the computer architecture course . . . . .	5
1.3.2 Computer architecture course . . . . .	6
1.4 Analysis of existing applications . . . . .	7
1.4.1 STRM Tests . . . . .	7
1.4.2 CS301 Computer Architecture . . . . .	8
1.4.3 BooleanTT . . . . .	8
1.4.4 DecToBin . . . . .	9

1.4.5	Summary . . . . .	9
1.5	Conclusion . . . . .	10
<b>2</b>	<b>Conceptual study</b>	<b>11</b>
2.1	introduction . . . . .	11
2.2	Graphic design . . . . .	11
2.2.1	Modeling . . . . .	11
2.2.2	Definition of Unified Modeling language . . . . .	12
2.3	Specification and needs analysis . . . . .	12
2.3.1	Basic functions of the application . . . . .	12
2.3.2	Non-functional needs . . . . .	12
2.4	Detailed design . . . . .	13
2.4.1	Use Case diagram . . . . .	13
2.4.2	Sequence diagrams . . . . .	14
2.4.3	COURSE . . . . .	15
2.4.4	QUIZ . . . . .	16
2.4.5	TEST . . . . .	18
2.4.6	Class Diagram . . . . .	20
2.4.7	The relational model and creation of the Database . . . . .	24
2.5	Conclusion . . . . .	25
<b>3</b>	<b>Implementation and hosting</b>	<b>26</b>
3.1	Introduction . . . . .	26
3.2	Work environement . . . . .	26
3.2.1	Hardware . . . . .	26
3.2.2	Software . . . . .	27
3.3	Used programming languages . . . . .	29
3.3.1	HTML: HyperText Markup Language . . . . .	29
3.3.2	CSS: Cascading Style Sheets . . . . .	29
3.3.3	JavaScript — Dynamic client-side scripting . . . . .	29
3.3.4	MySQL — Structured Query Language . . . . .	29
3.4	Presentation of graphical interfaces . . . . .	30
3.4.1	Home page interface . . . . .	30



3.4.2	Course page interface . . . . .	32
3.4.3	Try interface . . . . .	34
3.4.4	Quiz interface . . . . .	37
3.4.5	Test interface . . . . .	38
3.5	Hosting . . . . .	38
3.6	Conclusion . . . . .	38
<b>General conclusion</b>		<b>39</b>
<b>Bibliography</b>		<b>40</b>

# List of Figures

1.1	Structure of STRM . . . . .	5
2.1	Use Case diagram. . . . .	14
2.2	COURSE Sequence diagram. . . . .	16
2.3	QUIZ Sequence diagram. . . . .	18
2.4	TEST Sequence diagram. . . . .	20
2.5	Class Diagram. . . . .	21
2.6	Database's Class Diagram. . . . .	23
3.1	Home page . . . . .	30
3.2	Home page . . . . .	30
3.3	Home page . . . . .	31
3.4	Home page . . . . .	31
3.5	Course page . . . . .	32
3.6	Course page . . . . .	32
3.7	Course page . . . . .	33
3.8	Course page . . . . .	33
3.9	Home page . . . . .	34
3.10	Try option . . . . .	34
3.11	Try option . . . . .	35
3.12	Try option . . . . .	35
3.13	Try option . . . . .	36
3.14	Quiz page . . . . .	37
3.15	Test page coming soon . . . . .	38

# List of Tables

1.1	Analysis of existing summary table . . . . .	9
2.1	Text Description of Use Case "COURSE". . . . .	15
2.2	Text Description of Use Case "QUIZ". . . . .	17
2.3	Text Description of Use Case "TEST". . . . .	19
2.4	Description of Class Diagram. . . . .	22
2.5	Description of Database's Class Diagram. . . . .	24

# Abbreviations list

RWD	Responsive Web Design
IC	Interactive Course
MI	Math and Computer Science
AMO	AKLI MOHAND OULHADJ university
STRM	Computer Architecture or Structure Machine
UML	Unified Modeling Language

# General Introduction

As new students at the university, we noticed the difficulty of adapting to the new system of study and the linguistic discrepancy, in addition to the difficulty of communicating and interacting directly with professors.

In the machine structure lessons, we noticed the possibility of developing and embodying its lessons into applications that contain interactive lessons and examples in order to help students gain a deeper understanding of the lessons.

Computer architecture is a science or a set of rules stating how computer software and hardware are joined together and interact to make a computer work. It not only determines how the computer works but also of which technologies the computer is capable.[\[1\]](#)

In our case and as distant as our search goes there are no as of now existing solutions(neither apps nor websites), but there exists a few tools, websites and mobile apps that are to some degree related to our extend.

Our manuscript is structured in three chapters:

First, chapter will talk about the general dedefinitions of computer architecture, role, and problem in distributing them, and we will mention examples of the existing applications.

Second chapter, In this chapter we will present the conceptual solution for our application, and introduce a detailed demo of our app to explain its basic functions and structure. But before that, we will present Modelling tool used: UML.

Third chapter, we will talk about the environment (Work and Software) and about the will use programming languages and presentation of graphical interfaces.

# Preliminary study

## 1.1 Introduction

In this chapter we will go over some general definition, study of the existing and analysis of existing applications. All about the topic of computer architecture in the form of an interactive course.

## 1.2 General Definitions

### 1.2.1 Computer Architecture

#### **In general**

Computer architecture is a science or a set of rules stating how computer software and hardware are joined together and interact to make a computer work. It not only determines how the computer works but also of which technologies the computer is capable.[\[2\]](#)

#### **In our case**

Computer architecture or better known as "Structure Machine"(STRM) is a course for first-year students in the Math and Computer Science department (MI) of the university Bouira AMO. and also in all the MI departments in Algeria universities.

Computer architecture (STRM) has a coefficient of 3 and a credit of 5, which makes it one of the more fundamental courses.[\[3\]](#)

### 1.2.2 Interactive Course

The term "Interactive Course" (IC) typically describes material of an educational nature delivered in a format which allows the user to directly impact the materials content, and out-come. An example of such material would be a computer based presentation requiring a user to select the correct answer to a give question before proceeding to the next topic. These types of courses are almost always computer based and most likely to be delivered to the user thru the internet. Due to their convenient delivery, availability and almost endless subject matter, interactive courses have become a major tool for those seeking to provide as well as those seeking to obtain education, training or certification in a given area of study.[4]

With growing access and availability to computers and the internet, many schools, universities, businesses and government agencies are turning to IC's to train and educate their students and staff.[4]

### 1.2.3 Responsive Web Design

Responsive web design (RWD) is a web development approach that creates dynamic changes to the appearance of a website, depending on the screen size and orientation of the device being used to view it. RWD is one approach to the problem of designing for the multitude of devices available to customers, ranging from tiny phones to huge desktop monitors.[5]

### 1.2.4 Website

A website is a collection of publicly accessible, interlinked Web pages that share a single domain name. Websites can be created and maintained by an individual, group, business or organization to serve a variety of purposes.[6]



## 1.3 Study of the existing

We did go over the definition of computer architecture in a general sense and in our context of study(STRM), so in this part we will go more in-depth about the structure of the computer architecture(STRM) course (see Figure 1.1), and which resources we will be basing our own study of the course from.

### 1.3.1 Structure of the computer architecture course

The structure of the STRM course (see Figure 1.1) was fully based from our study of the STRM 1 Book[7], and the MI official program[3].

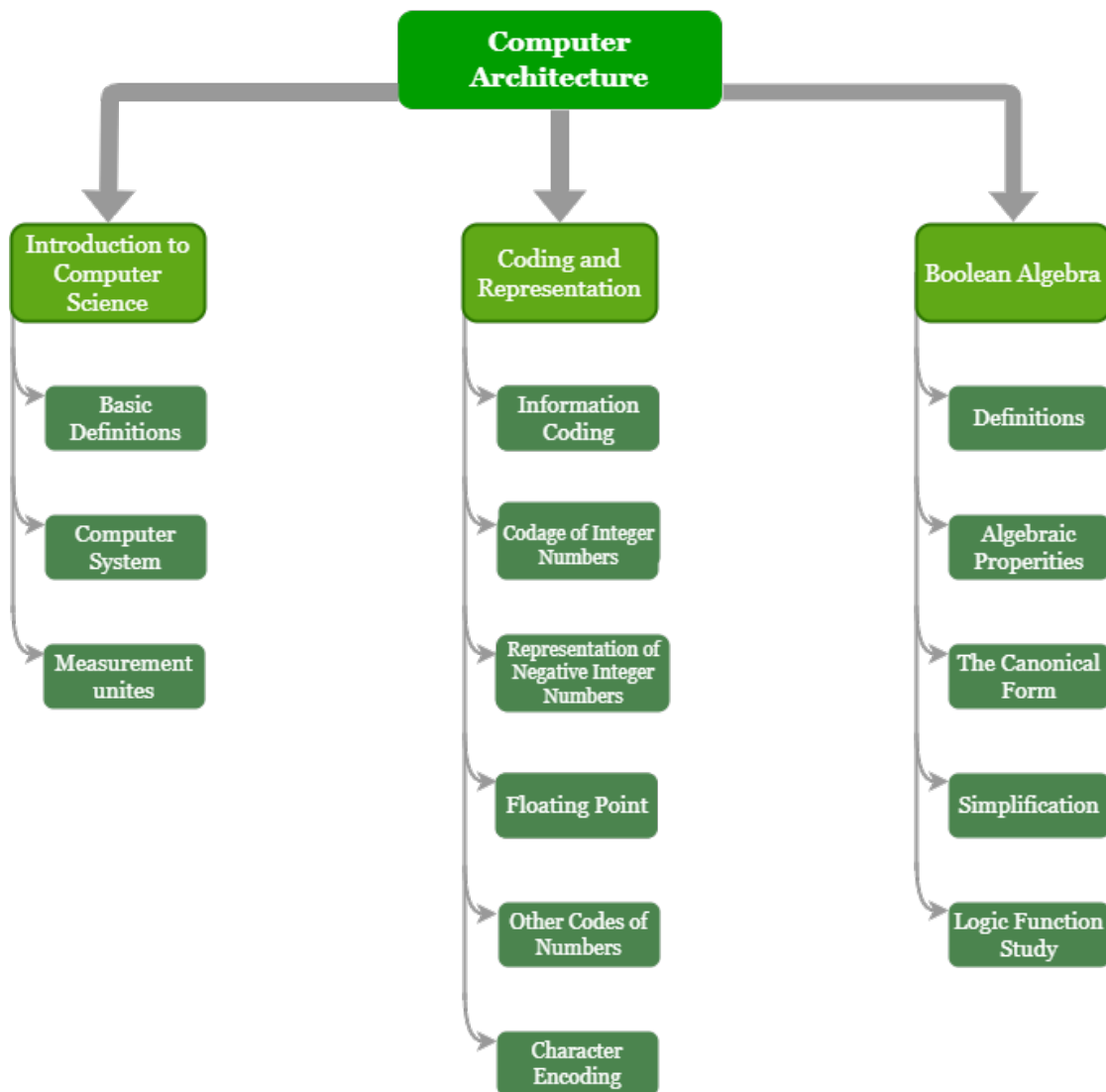


Figure 1.1: Structure of STRM

### 1.3.2 Computer architecture course

There exists some books that cover the course of computer architecture, and based on our experience as students of the MI department that has already studied this course we would like to mention the following :

#### Digital Design and Computer Architecture

This book by David Harris and Sarah Harris is designed for courses that combine digital logic design with computer organization/architecture or that teach these subjects as a two-course sequence. Digital Design and Computer Architecture begins with a modern approach by rigorously covering the fundamentals of digital logic design and then introducing Hardware Description Languages (HDLs). By the end of Digital Design and Computer Architecture, readers will be able to build their own microprocessor and will have a top-to-bottom understanding of how it works—even if they have no formal background in design or architecture beyond an introductory class. [8]

#### STRM 1 Book

This book by Dr. Taha Zerrouki is a lessons and solved exercises focus, intended for first-year students of mathematics and Computer Science department (MI) in Algerian universities.[7]

It contains in this part the lessons of the first semester:

- Interoduction to computer science.
- Coding and representation of information.
- Boolean Algebra.

The book contains a large number of exercises divided according to chapters, a large part of it is solved, as well as a special section for continuous assessment examinations with Correction, and another section for exams. This book comes as a result of the experience that Dr Taha gained in teaching at the University of Bouira for many years in the Department of Computer Science. The book is also bilingual, with lessons in French and translated into Arabic, in order to help new students who suffer from a language barrier in their college start. <sup>1</sup>

---

<sup>1</sup>We will be focusing our study of the course on the STRM 1 book [7]

## 1.4 Analysis of existing applications

A very important part in developing a new project in the domain of computer science is looking for and analyzing already existing solutions to the project you are working on, in our case the project is making an **Interactive Course For Computer Architecture**. if there is no already existing solutions you would look for solutions that are somehow related (similar) to the project at hand.

In our case and as far as our search goes there are no already existing solutions (neither apps nor websites), but there exists some tools, websites and mobile apps that are somewhat related to our project. like:

### 1.4.1 STRM Tests

This tool was build by Dr. Taha Zerrouki to generate and create random tests for computer architecture 1 (STRM 1) - first year MI in Algerian universities.[9] the features of the **STRM Tests** tool[9] are:

- Generate checks and questions with solutions.
- Supports the following classes:
  1. Enumeration systems.
  2. Representation of natural and real numbers and letters.
  3. Encoding information.
  4. Boolean algebra.
- Generate answers:
  1. Possibility to draw a logical function diagram.
  2. Generate Karnough tables with solution.
  3. Generating graphic solutions to Karnough's table.
- Generate duplicate forms of questions for easier typing.
- Random generation of questions.

### 1.4.2 CS301 Computer Architecture

Saylor Academy is a nonprofit initiative working since 2008 to offer free and open online courses to all who want to learn. They offer nearly 100 full-length courses at the college and professional levels, each of which is available right now . Modern computer technology requires an understanding of both hardware and software, since the interaction between the two offers a framework for mastering the fundamentals of computing. The purpose of this course is to cultivate an understanding of modern computing technology through an in-depth study of the interface between hardware and software. In the CS301 Computer Architecture course you will study the history of modern computing technology before learning about modern computer architecture and a number of its essential features, including :

- Instruction sets
- Processor arithmetic and control
- The Von Neumann architecture
- Pipelining, memory management, storage, input/output

The course will conclude with a look at the recent switch from sequential processing to parallel processing by looking at the parallel computing models and their programming implications.[\[1\]](#)

### 1.4.3 BooleanTT

This is a simple and easy-use app developed by Hashan Chamara Rajapaksa to :

- Simplify/Minimize Boolean expressions
- Solve Karnaugh maps
- Generate Truth-Tables of Boolean expressions
- Generate Min Terms and Max Terms from Truth-Table easily
- Check logic circuit

And also it will help you learn about basic logic gates theories.[\[10\]](#)

### 1.4.4 DecToBin

This application is developed by Kyssnar, it allows you to encode numbers in base 10 to different formats of binary coding:

- IEEE 32 and 64 bits
- One's and Two's Complement
- Binary Code Decimal
- Excess to N
- Excess to N - 1
- Signed Magnitude
- Simple and double precision

Also includes a binary calculator that can add, subtract, multiply and divide in binary and show results in binary and decimal.[\[11\]](#)

### 1.4.5 Summary

We summarized the previous study in this [Table 1.1](#).

	Courses			Tests		
	Intro CS	Code Reprsnt	Bool Algebr	Intro CS	Code Reprsnt	Bool Algebr
STRM Tests				X	X	X
CS301	X	X	X			
BooleanTT			X			X
DecToBin					X	

Table 1.1: Analysis of existing summary table

## **1.5 Conclusion**

We have done an analytical study and found a solution to the problem at hand, and we have studied the existing partial solutions to try and combine them for a better solution, and in order to be able to identify the functional and non-functional requirements of the application we seek to design, and we will present the adequate conception and modeling.

# Conceptual study

## 2.1 introduction

This chapter describes a very important part which is the conceptual study and an analytical look in addition to the base design.

This part define the basic steps for developing our system where specifying functional and non-functional requirements using the concepts of Unified Modeling Language ( Use Case diagram, Sequence diagrams, and Class diagram), and clarify the goal of this analysis and design all about the topic of computer architecture in the form of an interactive course.

## 2.2 Graphic design

### 2.2.1 Modeling

Modeling uses graphic notation to create visual models of software systems, the use of diagrams and illustrations makes something complex more understandable, that's why Unified Modeling language (UML) [\[12\]](#) is a standardized modeling language enabling developers to specify, visualize, construct and document artifacts of a software system.

Thus, UML makes these artifacts scalable, secure and robust in execution. It is an important aspect involved in object-oriented software development.

## 2.2.2 Definition of Unified Modeling language

The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system.[\[13\]](#)

The creation of UML was originally motivated by the desire to standardize the disparate notational systems and approaches to software design. It was developed at Rational Software in 1994–1995, with further development led by them through 1996.[\[13\]](#)

In 1997, UML was adopted as a standard by the Object Management Group (OMG), and has been managed by this organization ever since. In 2005, UML was also published by the International Organization for Standardization (ISO) as an approved ISO standard. Since then the standard has been periodically revised to cover the latest revision of UML. In software engineering, most practitioners do not use UML, but instead produce informal hand drawn diagrams; these diagrams, however, often include elements from UML.[\[13\]](#)

## 2.3 Specification and needs analysis

### 2.3.1 Basic functions of the application

The main requirements which are specifically related to the behavior and functions of the system can be summarized in the following points :

1. Interactive course with practical examples.
2. Fun Quiz and testing what the user has learned.
3. Practical models for machine structure tests.

### 2.3.2 Non-functional needs

These are the most important major requirements not specifically related to the behavior of the system but rather the identification of internal and external constraints of the system :



**Ergonomic interfaces**

The solution must present an ergonomic interface encompassing all the functionalities. The manipulation of the interface should not require advanced computer knowledge, it must be simple and clear in order to adapt to the computer knowledge of the users.

**Reliability and speed**

The system must guarantee the speed and reliability of the search for information, as well as optimal management of resources.

**The code**

Must be clear to allow future evolutions or improvements.[\[14\]](#)

## **2.4 Detailed design**

### **2.4.1 Use Case diagram**

Use Case Diagram describes functionality of a system in terms of actors, goals as use cases and dependencies among the use cases.[\[12\]](#)

**System**

Interactive Course For Structure Machine 1.

**Actors**

Specify a role played by a user or any other system interacting with the subject.[\[12\]](#)

**Use cases**

describes functionalities of the system .In this system :

1. COURSE : When user click on COURSE he can see the whole CHAPTERS of the course, when he click on try in each TITLE of CHAPTERS a question window will appear with a specific type of question depends on the chapter's TITLE.

2. QUIZ : When user click on QUIZ he can test what he learned in this course with a specific Options (choose chapters and number of questions), when clicking on start a question window will appear.
3. TEST : When user click on TEST he will find a practical models for machine structure tests including the whole course.

## Relations

1. INCLUDE : (QUIZ and Options),(Options and Question),(TEST and Question).
2. EXTENTION : the rest of use cases.

The figure 2.1 represents this system's Use Case Diagram.

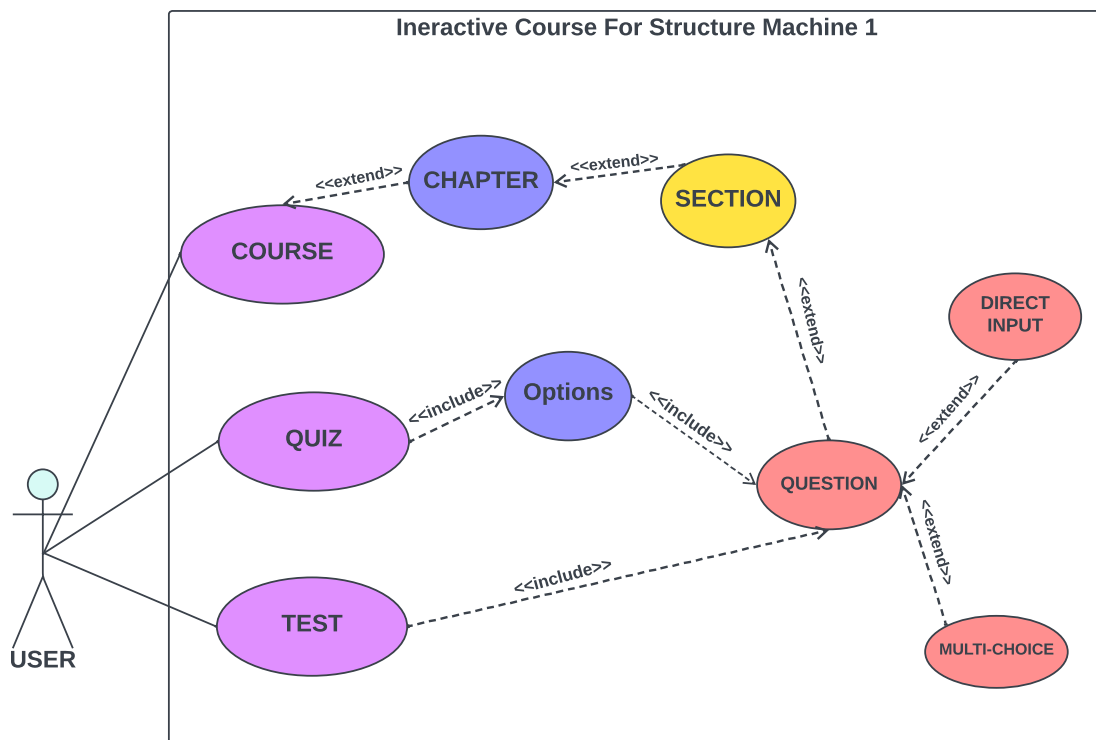


Figure 2.1: Use Case diagram.

### 2.4.2 Sequence diagrams

A sequence diagram, in the context of UML, represents object collaboration and is used to define event sequences between objects for a certain outcome, it is an essential component

used in processes related to analysis, design and documentation.[15]

A sequence diagram is also known as a timing diagram, event diagram and event scenario.[15]

### 2.4.3 COURSE

This first scenario shows how user can see the whole CHAPTERS of the course, and try a question with a specific type depends on the chapter's TITLE.

Identification summary	
Use Case Title	COURSE.
Abstract	User learn course.
Actor	User.
Identification summary	
Preconditions	Accessible system.
User exists	The purpose of this system is navigate and learn from the course.
Scenario	1- User clicks on Course button. 2- System loads all chapters. 3- System shows the chapters. 4- User chooses chapter. 5- System loads sections of chapter. 6- System shows the content of sections. 7- User clicks on Try. 8- System sends SQL request (Select section's question) to DataBase. 9- Database return data(section's question). 10- System loads results. 11- System shows results.
Object	User learn the course with live examples.

Table 2.1: Text Description of Use Case "COURSE".

The figure 2.2 represents COURSE Sequence diagram.

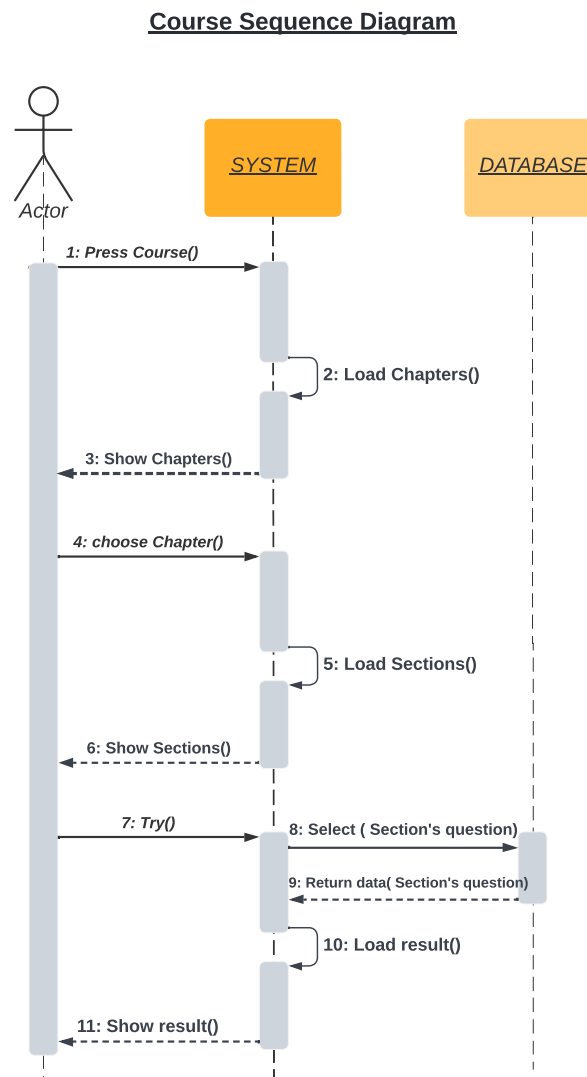


Figure 2.2: COURSE Sequence diagram.

#### 2.4.4 QUIZ

This second scenario shows how user can test what he learned in this course with a specific options (choose chapters and number of questions).

Identification summary	
Use Case Title	QUIZ.
Abstract	User takes a quiz.
Actor	User.
Identification summary	
Preconditions	Accessible system.
User exists	The purpose of this system is taking a quiz.
Scenario	1- User clicks on Quiz button. 2- System loads Options. 3- System shows Options. 4- User chooses options. 5- System shows Start button. 6- User clicks on Start button. 7- System sends SQL request (Select Questions) to DataBase. 8- Database return data(Questions). 9- System loads Quiz results. 10- System shows Quiz results.
Object	User can take a quiz with any options he wants.

Table 2.2: Text Description of Use Case "QUIZ".

The figure 2.3 represents QUIZ Sequence diagram.

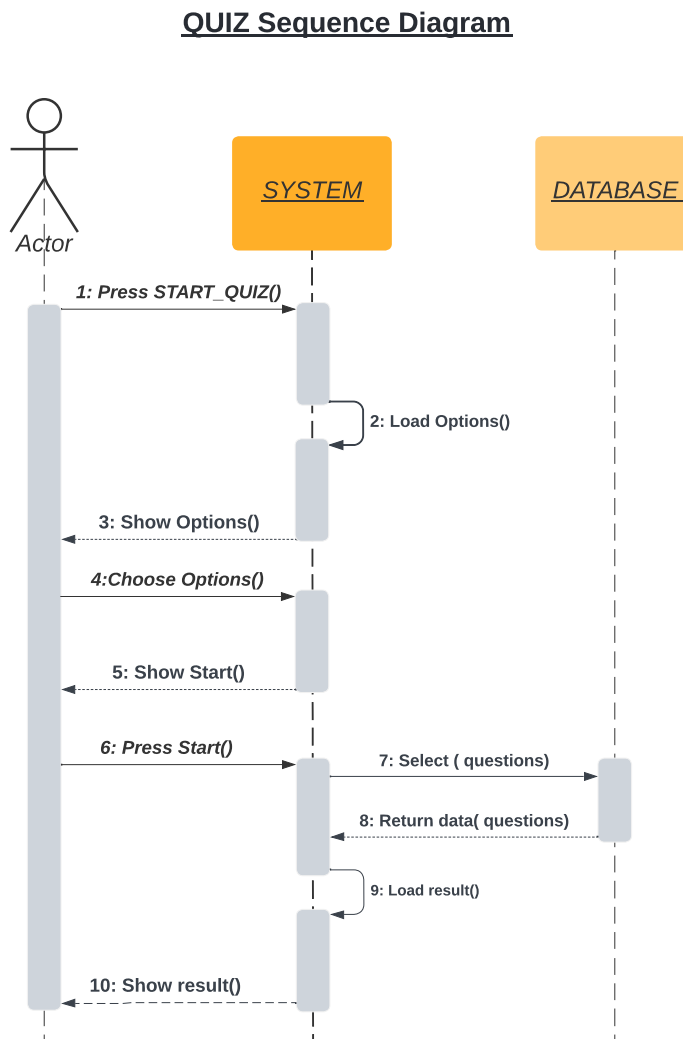


Figure 2.3: QUIZ Sequence diagram.

### 2.4.5 TEST

This third scenario shows how user can find a practical models for machine structure tests including the whole course.

Identification summary	
Use Case Title	Test.
Abstract	User takes a Test.
Actor	User.
Identification summary	
Preconditions	Accessible system.
User exists	The purpose of this system is taking a Test.
Scenario	1- User clicks on Test button. 2- System shows Start button. 3- User clicks on Start button. 4- System sends SQL request (Select Question from each chapter) to DataBase. 5- Database return data(Question from each chapter). 6- System loads Test results. 7- System shows Test results.
Object	User can find a practical models for machine structure tests including the hole course.

Table 2.3: Text Description of Use Case "TEST".

The figure 2.4 represents TEST Sequence diagram.

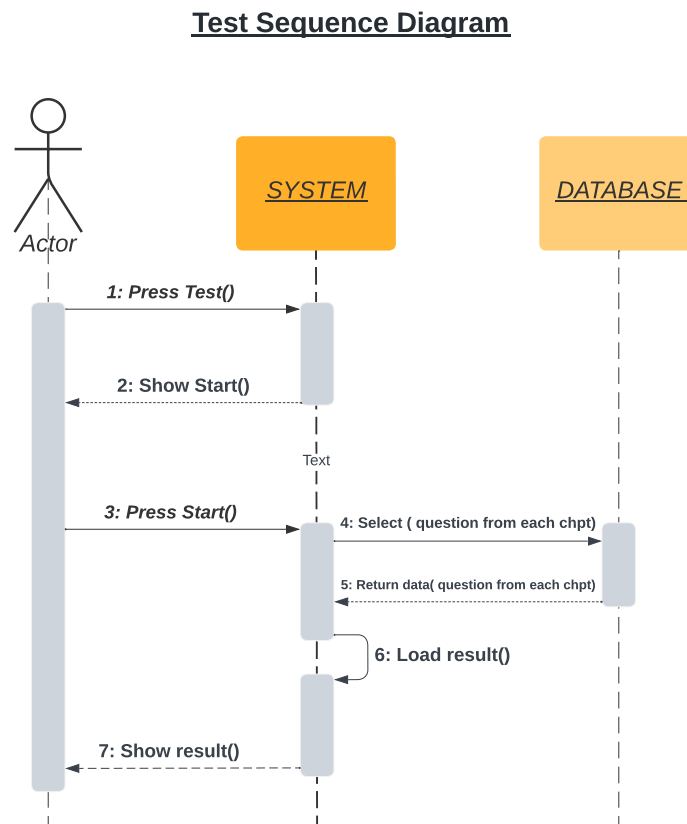


Figure 2.4: TEST Sequence diagram.

### 2.4.6 Class Diagram

A class diagram is a type of diagram and part of a unified modeling language (UML) that defines and provides the overview and structure of a system in terms of classes, attributes and methods, and the relationships between different classes. [16]

It is used to illustrate and create a functional diagram of the system classes and serves as a system development resource within the software development life cycle. [16]



## Why Class Diagram?

A class diagram is primarily designed for developers to provide the conceptual model and architecture of the system being developed. Typically, a class diagram consists of more than one class or all the created classes for a system. [16]

It is a type of structure diagram and looks similar to a flow chart having three main parts illustrated in rectangular boxes. The first or top part specifies the class name, the second or middle specifies attributes of that class and the third or bottom section lists the methods or operations that specific class can perform. [16]

The figure 2.5 represents this system's Class Diagram.

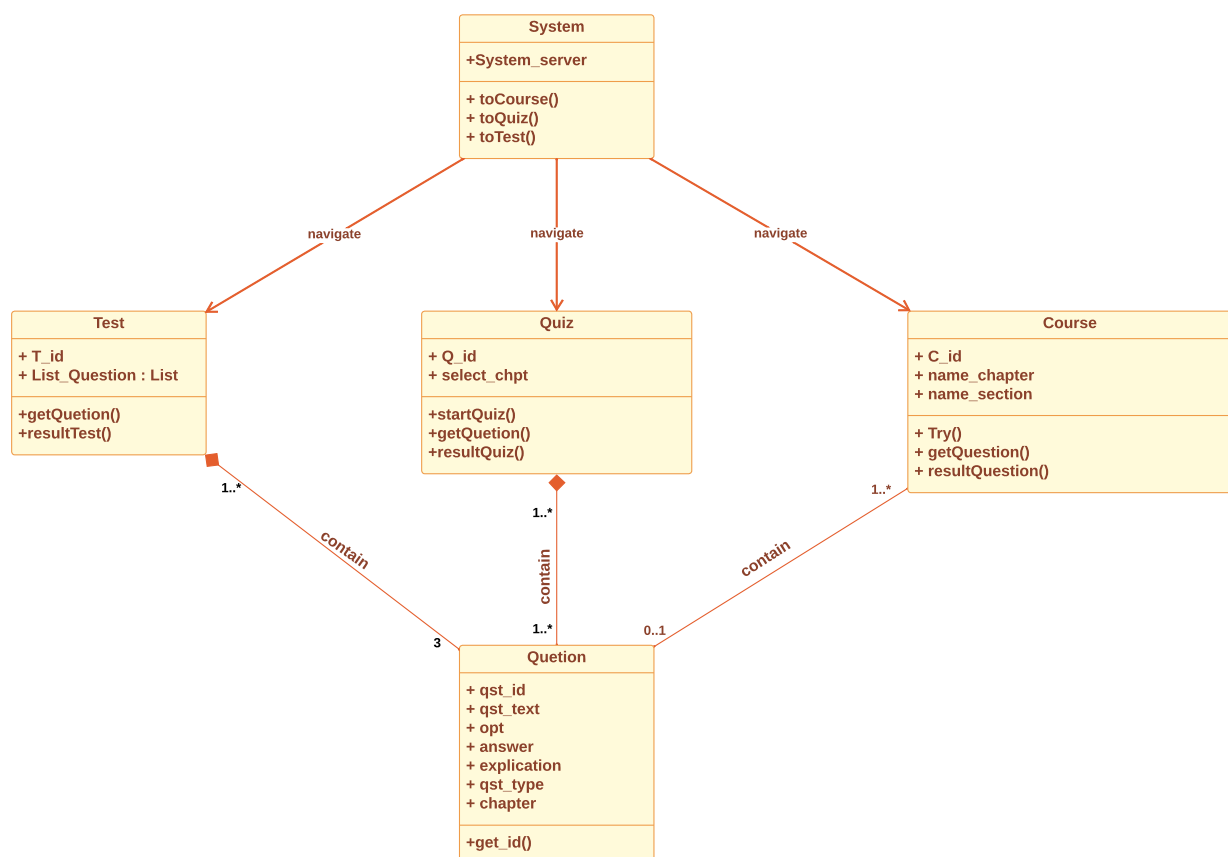


Figure 2.5: Class Diagram.

The table 2.4 represents classes and attributes as well as their types and description .

Class	Attributes	Description	Type
System	System-server	system's identifier.	Digital
Course	C-id	Course identifier.	Digital
	name-chapter	Chapter's name.	Alphabetic
	name-section	Section's name.	Alphabetic
Quiz	Q-id	Quiz identifier.	Digital
	select-chpt	Choosing chapter.	Digital
Test	T-id	Test identifier.	Digital
	List-Question	List of questions.	Alphabetic
Question	qst-id	Question identifier.	Digital
	qst-text	Questions text.	Alphabetic
	opt	Questions options.	Alphabetic
	answer	Questions answer.	Alphabetic
	explication	Answer explication.	Alphabetic
	qst-type	Questions type.	Alphabetic
	chapter	Chapter's number.	Digital

Table 2.4: Description of Class Diagram.

## Database's Class Diagram

The figure 2.6 represents the database in the form of a class diagram.

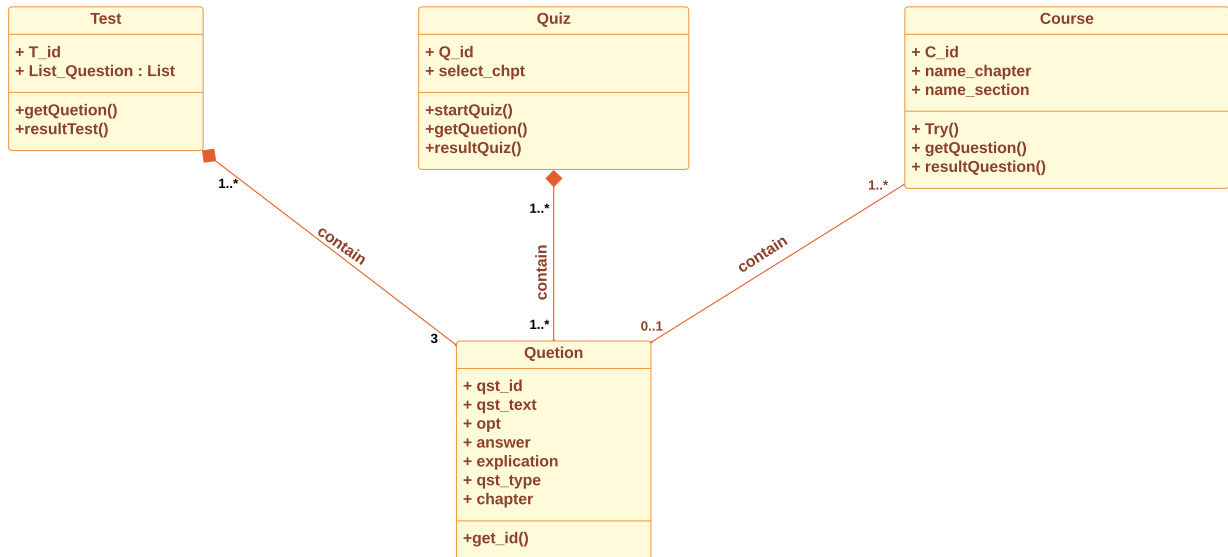


Figure 2.6: Database's Class Diagram.

The table 2.5 represents classes and attributes of database's class diagram as well as their types and description.

Class	attributes	Description	Type
Course	C-id	Course identifier.	Digital
	name-chapter	Chapter's name.	Alphabetic
	name-section	Section's name.	Alphabetic
Quiz	Q-id	Quiz identifier.	Digital
	select-chpt	Choosing chapter.	Digital
Test	T-id	Test identifier.	Digital
	List-Question	List of questions.	Alphabetic
Question	qst-id	Question identifier.	Digital
	qst-text	Questions text.	Alphabetic
	opt	Questions options.	Alphabetic
	answer	Questions answer.	Alphabetic
	explication	Answer explication.	Alphabetic
	qst-type	Questions type.	Alphabetic
	chapter	Chapter's number.	Digital

Table 2.5: Description of Database's Class Diagram.

### 2.4.7 The relational model and creation of the Database

A relational data model involves the use of data tables that collect groups of elements into relations. These models work based on the idea that each table setup will include a primary key or identifier. Other tables use that identifier to provide "relational" data links and results. Database administrators use something called Structured Query Language (SQL) to retrieve data elements from a relational database.<sup>[17]</sup>

Course(C-id,#name-chapter).

Quiz(Q-id,#select-chpt).

Test(T-id,List-Question).

Question(qst-id,qst-text,opt,answer,explication,qst-type,chapter).

## 2.5 Conclusion

This chapter defined the basic steps for developing the system where specified functional and non-functional requirements using the concepts of Unified Modeling Language ( Use Case diagram, Sequence diagrams, and Class diagram), and clarified the goal of the analysis and design. Next step is realizing the system by introducing the programming languages, tools, and development environment used on implementation.

# Chapter 3

## Implementation and hosting

### 3.1 Introduction

After completing the application design arrange, in this chapter we'll begin the implementation and assessment arrange by displaying the work environments and innovation utilized in creating the application, which made it conceivable to reach the ultimate result, and we'll also display the interfacing whereas actualizing the application capacities, and at the conclusion we provide an clarification of the foremost critical highlights of the application.

### 3.2 Work environnement

This project was built up on distinctive work environnement, agreeing to the available capabilities in arrange to reach the most excellent conceivable result. These environments are:

#### 3.2.1 Hardware

For programming we used a laptop, as for testing we used mobile phones, laptops, tabeltes and even a smart tv to monitor the performance of the website on them. The characteristics of these devices are:

- LENOVO ThinkPad T470s laptop(for programming):

- Processor : Intel (R) Core (TM) i7-7600U CPU @ 2.80 GHz(4 CPUs), 2.90 GHz.
- RAM : 16 GB.
- System : Windows 10 Pro - 64 Bit.
- HP ProBook 640 G1 laptop(for testing):
  - Processor : Intel (R) Core (TM) i7-4600M CPU @ 2.90 GHz(4 CPUs), 2.90 GHz.
  - RAM : 8 GB.
  - System : Windows 10 Pro - 64 Bit.
- LATITUDE 3330 laptop(for testing):
  - Processor : Intel (R) Core (TM) i5-3337U CPU @ 1.80 GHz, 1.80 GHz.
  - RAM : 12.0 GB.
  - System : Windows 10 Pro - 64 Bit.
- SamSung Galaxy J2 phone(for testing):
  - RAM : 1 GB.
  - System : Android 5.1.1.
- SamSung Galaxy M20 phone(for testing):
  - RAM : 3 GB.
  - System : Android 10.

### 3.2.2 Software

#### Visual Studio Code:

Visual Studio Code is not just another evolved Notepad with syntax colorization and automatic indentation. Instead, it is a very powerful code-focused development environment expressly designed to make it easier to write web, mobile, and cloud applications using languages that are available to different development platforms and to support the

application development life cycle with a built-in debugger and integrated support for the popular Git version control engine.[\[18\]](#)

### **Sublime Text:**

Sublime Text is a proprietary, cross-platform text editor designed for people who spend huge amounts of time shuffling code around. A programmer's editor, Sublime Text is a third option to the long-standing "Vi or Emacs" conundrum. Going beyond the basics of syntax highlighting and code folding, Sublime offers a litany of innovative and unique features. With version 3.0 just around the corner, I'm taking you on a tour of Sublime's most compelling features and add-on packages.[\[19\]](#)

### **XAMPP:**

XAMPP is a small and light Apache distribution containing the most common web development technologies in a single package. Its contents, small size, and portability make it the ideal tool for students developing and testing applications in PHP and MySQL. XAMPP is available as a free download in two specific packages: full and lite. While the full package download provides a wide array of development tools, this article will focus on using XAMPP Lite which contains the necessary technologies that meet the Ontario Skills Competition standards. As the name implies, the light version is a small package containing Apache HTTP Server, PHP, MySQL, phpMyAdmin, Openssl, and SQLite.[\[20\]](#)



## 3.3 Used programming languages

### 3.3.1 HTML: HyperText Markup Language

HTML (HyperText Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation (CSS) or functionality/behavior (JavaScript).[\[21\]](#)

### 3.3.2 CSS: Cascading Style Sheets

Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML or XML (including XML dialects such as SVG, MathML or XHTML). CSS describes how elements should be rendered on screen, on paper, in speech, or on other media.[\[22\]](#)

### 3.3.3 JavaScript — Dynamic client-side scripting

JavaScript is a programming language that allows you to implement complex things on web pages. Every time a web page does more than just sit there and display static information for you to look at—displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, or more—you can bet that JavaScript is probably involved.[\[23\]](#)

### 3.3.4 MySQL — Structured Query Language

MySQL is a very fast, robust, relational database management system. A database enables you to efficiently store, search, sort, and retrieve data. The MySQL server controls access to your data to ensure that only authorized users can obtain access. Hence, MySQL is a multi-user, multi-threaded server. It uses SQL the standard database query language worldwide.[\[24\]](#)

## 3.4 Presentation of graphical interfaces

### 3.4.1 Home page interface

When the client visit the website, the first thing he sees is the home page. fig 3.1

<sup>2</sup>

On Desktop:

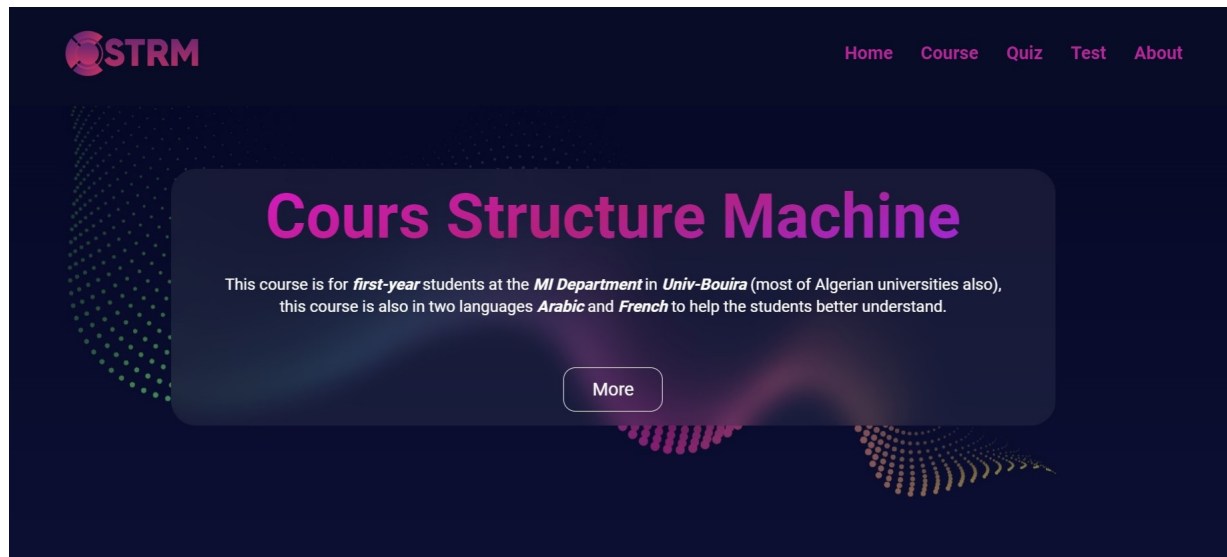


Figure 3.1: Home page

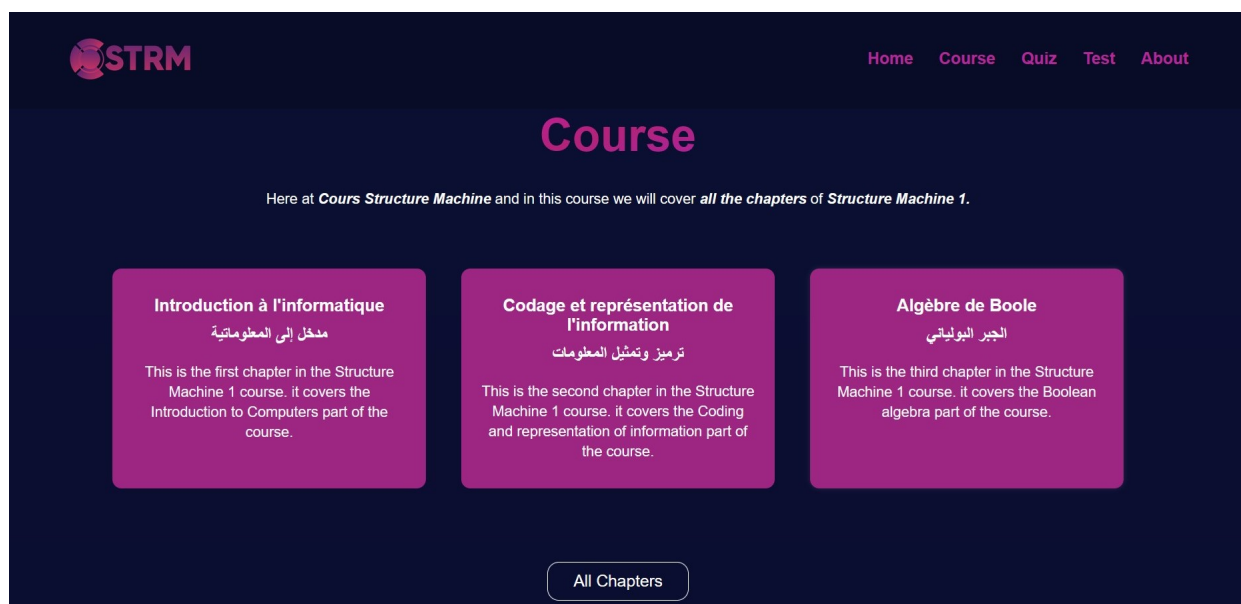


Figure 3.2: Home page

<sup>2</sup>The website is responsive so will show it in desktop and mobile[5]

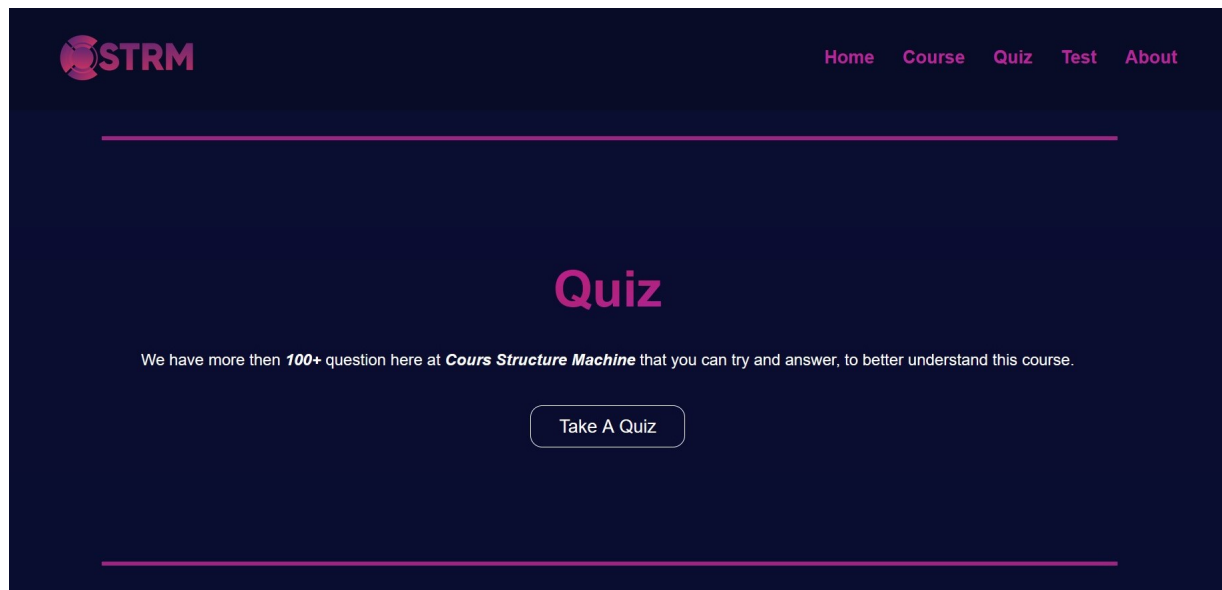


Figure 3.3: Home page

On Mobile:

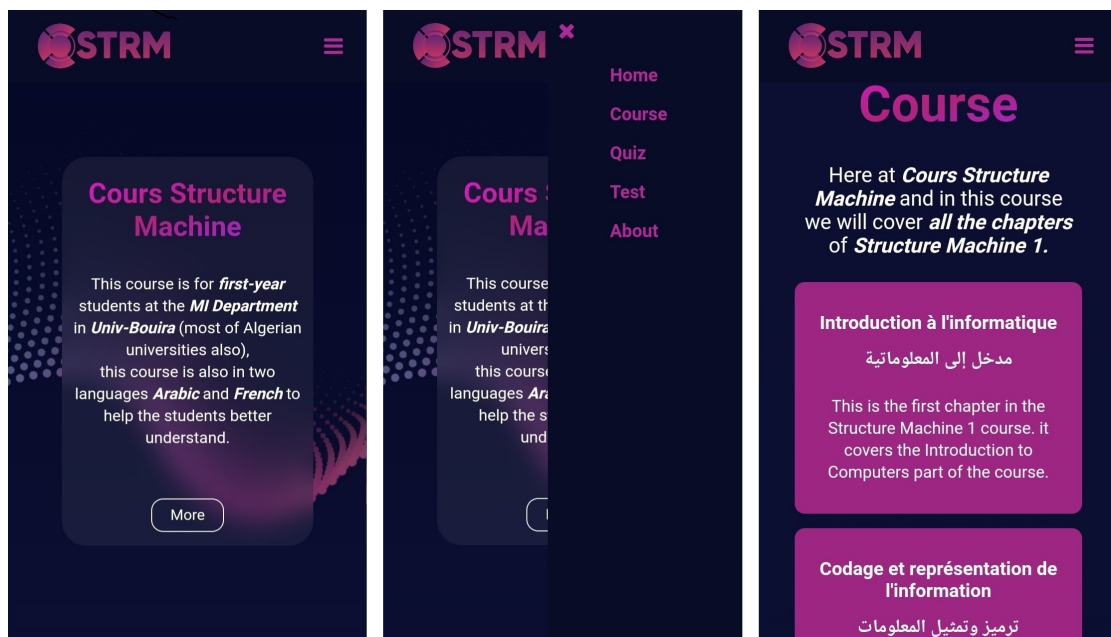


Figure 3.4: Home page

### 3.4.2 Course page interface

The user(client) will find the course in the course page, the course page have a local navigation (see figure 3.5), it is also in two languages (see figure 3.7)

On Desktop:



Figure 3.5: Course page



Figure 3.6: Course page



[Home](#)
[Course](#)
[Quiz](#)
[Test](#)
[About](#)

## 2. Système informatique

Un système informatique est composé de deux parties : Matériel (Hardware) et Logiciel (Software).

**Le Hardware :** Tout ce qui concerne les circuits électriques, électroniques ainsi que le mécanisme. L'architecture interne d'un ordinateur est composée de l'unité centrale et la mémoire et les périphériques.

**Le Software** (le logiciel) : Tout ce qui concerne les programmes nécessaires pour le bon démarrage et l'utilisation du micro-ordinateur. On peut citer les programmes, les systèmes d'exploitation qui sont développés en utilisant des langages de programmation.


## نظام المعلومات

تكون النظام المعلوماتي من قسمين العتاد والبرامج

العتاد : كل ما هو دارات كهربائية والإلكترونية وآليات ميكانيكية. يتكون الحاسوب من الوحدة المركزية والذاكرة والأجهزة الملحقة.

البرمجيات: كل ما يحتاج إليه الجهاز ليقطع ويعمل ويقدم خدمات للمستخدم عدا العتاد. نذكر منها البرامج والأنظمة التشغيل التي بنيت باستعمال لغات البرمجة.

Figure 3.7: Course page



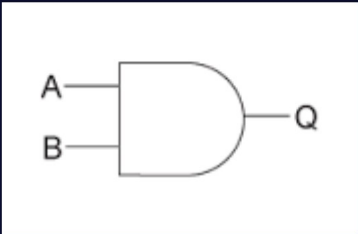
[Home](#)
[Course](#)
[Quiz](#)
[Test](#)
[About](#)

### 2.1. Conjonction

Elle est définie de la manière suivante : a ET b est VRAI si et seulement si a est VRAI et b est VRAI. Cette loi est aussi noté par un point «.»

### الوصل

نعرف الوصل بأن القضية 'أ و ب' صحيحة إذا وفقط إذا كان أ صحيحا و ب صحيحا. ونرمز له بالنقطة «.»



A	B	A et B
0	0	0
0	1	0
1	0	0
1	1	1

[Try](#)

Figure 3.8: Course page

## On Mobile:



Figure 3.9: Home page

## 3.4.3 Try interface

The objective of this project is to make an interactive course, so we gave the user the option to try and test his knowledge (see figure 3.10, 3.11, 3.12)

## On Desktop:

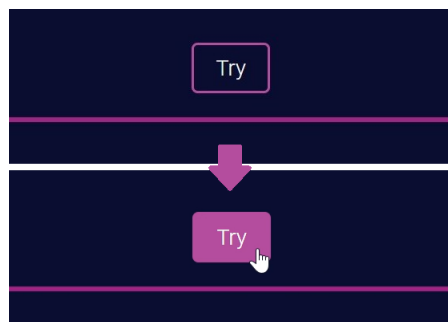


Figure 3.10: Try option

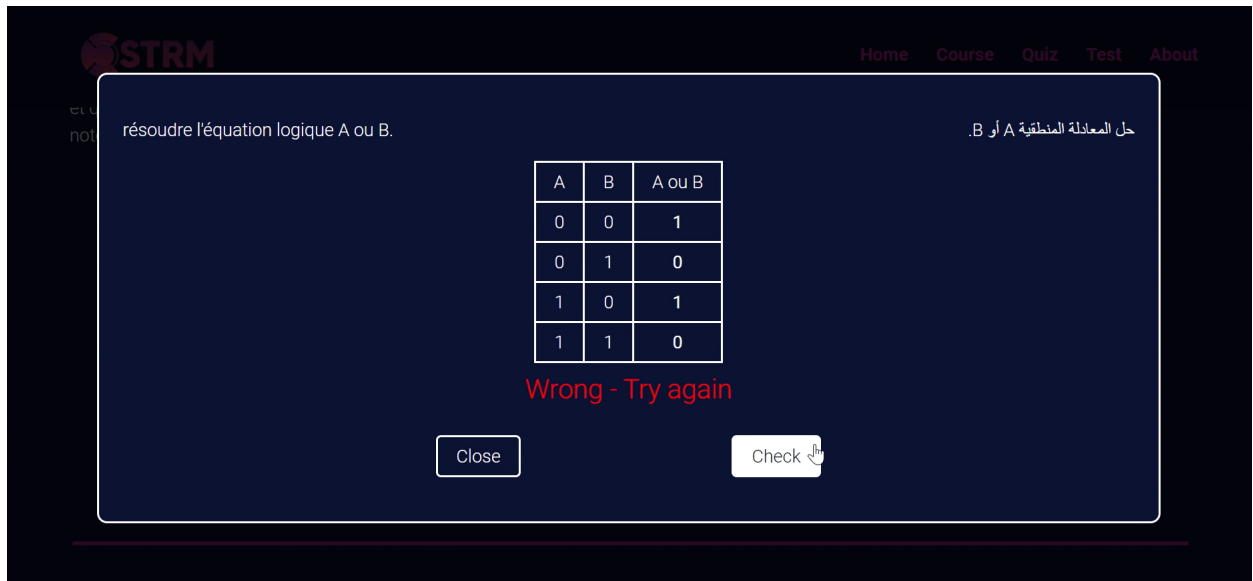


Figure 3.11: Try option

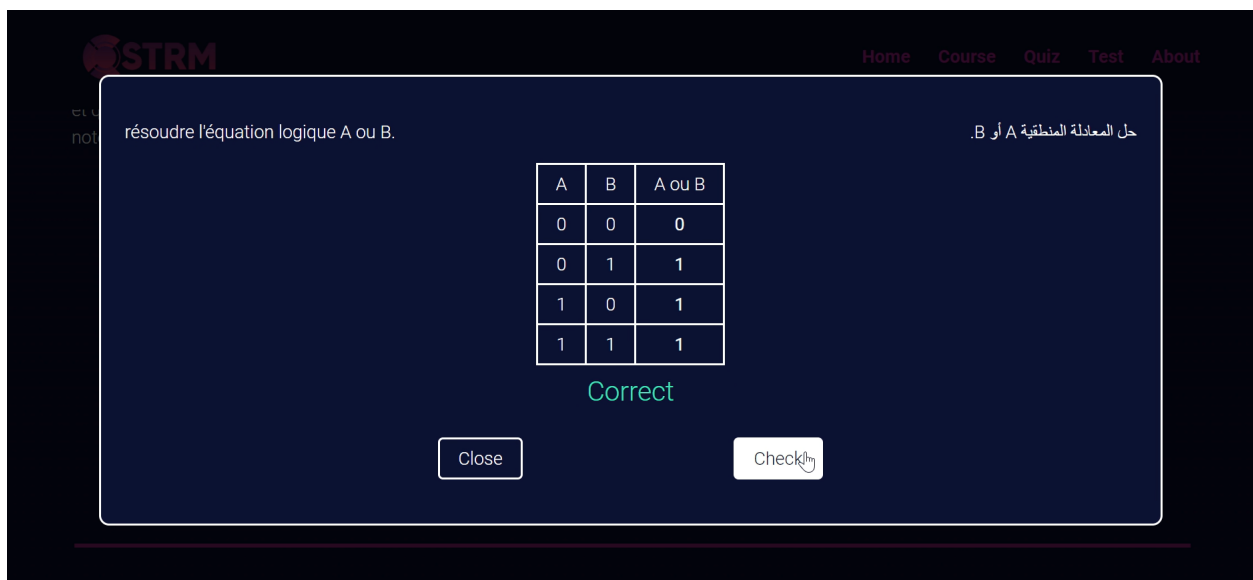


Figure 3.12: Try option

## On Mobile:



Figure 3.13: Try option



### 3.4.4 Quiz interface

The quiz is where the user can test his knowledge with random questions(see figure 3.14)

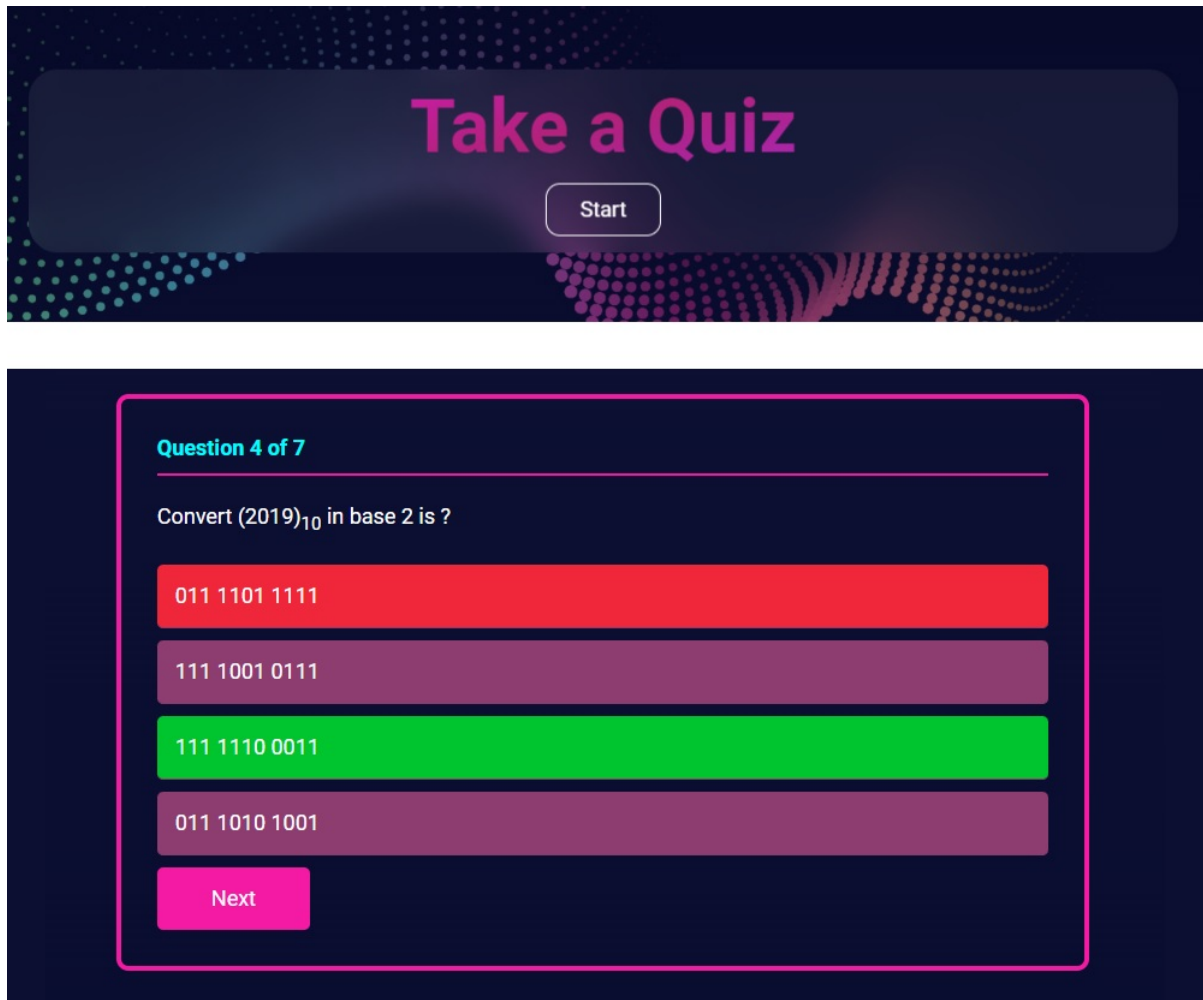


Figure 3.14: Quiz page

### 3.4.5 Test interface

The Test page is coming soon (see figure 3.15)

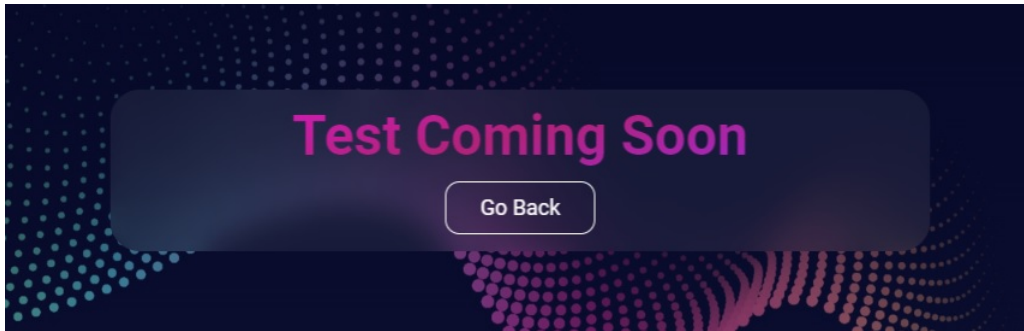


Figure 3.15: Test page coming soon

## 3.5 Hosting

We are hosting the website at github pages at : <https://lyes07.github.io/strm-website/index.html>  
and also at : <http://coursstrm.epizy.com/>

But hopefull in the future we will buy the domain name : [coursstrm.com](http://coursstrm.com) , so you can  
find the website at : <http://coursstrm.com/>

## 3.6 Conclusion

In this chapter, we have given a brief and clear clarification of the programming dialects utilized in creating of the website and the programs that were utilized without overlooking the devices utilized, and we too touched and showed the different interfaces, functions and given a clear clarification of the foremost imperative highlights of the website.

# General conclusion

The STRM interactive Course is a project that interactively provides the course of STRM to help the students of first-year MI, and various people to learn more about STRM, it is an open-source project (free to use).

We want this project to simplify the module of STRM and help the new bachelors in this next step in their life through this project, we faced a lot of problems as freshmen so we don't want new student to face the same problems especially in one of my favorite modules which is STRM.

# Bibliography

- [1] saylor academy. Computer architecture. <https://learn.saylor.org/course/CS301>. Accessed: 2022-3-7.
- [2] Pam. What is computer architecture? <https://www.computersciencedegreehub.com/faq/what-is-computer-architecture/>., October 2018. Accessed: 2022-3-6.
- [3] Ministry of Higher Education and Scientific Research. *officiel program - Maths and Informatiques*. Ministry of Higher Education and Scientific Research. Accessed: 2022-6-14.
- [4] Interactive courses. <https://en-academic.com/dic.nsf/enwiki/10720611>. Accessed: 2022-3-6.
- [5] Responsive web design (RWD) and user experience. <https://www.nngroup.com/articles/responsive-web-design-definition/>. Accessed: 2022-3-6.
- [6] Techopedia. Website. <https://www.techopedia.com/definition/5411/website>., November 2011. Accessed: 2022-3-6.
- [7] Dr.Taha Zerrouki. Strm 1 book. <https://archive.org/details/strm-1-book-taha-zerrouki-09-10-2021/STRM-1-Book-Taha-Zerrouki-25-12-2021/>, October 2021. Accessed: 2022-3-7.
- [8] David Harris and Sarah L Harris. *Digital design and computer architecture*. Morgan Kaufmann, 2010. Accessed: 2022-3-8.
- [9] Dr.Taha Zerrouki. Strm tests. <https://github.com/linuxscout/strm-tests/>. Accessed: 2022-3-7.

- 
- [10] Hashan Chamara Rajapaksa. BooleanTT. <https://play.google.com/store/apps/details?id=hashan.haze.booleanTT&hl=en&gl=US>, February 2022. Accessed: 2022-3-8.
- [11] Kyssnar. Dectobin. <https://play.google.com/store/apps/details?id=apps.kyssnar.android.dectobin&hl=en&gl=US>, September 2018. Accessed: 2022-3-8.
- [12] Techopedia. Uml. <https://www.techopedia.com/definition/3243/unified-modeling-language-uml#what-does-unified-modeling-language-uml-mean>, November 2012. Accessed: 2022-4-30.
- [13] YThi. Uml definition. <https://ythi.net/abbreviations/english/what-does-uml-mean-what-is-the-full-form-of-uml/>. Accessed: 2022-4-30.
- [14] Lyes Grine. strm website. <https://lyes07.github.io/strm-website/index.html>. Accessed: 2022-5-7.
- [15] Techopedia. Sequence diagram. <https://www.techopedia.com/definition/27490/sequence-diagram-uml>. Accessed: 2022-5-3.
- [16] Techopedia. Class diagram. <https://www.techopedia.com/definition/16466/class-diagram>. Accessed: 2022-5-6.
- [17] Techopedia. Relational data model. <https://www.techopedia.com/definition/21680/relational-data-model>. Accessed: 2022-5-7.
- [18] Alessandro Del Sole. Introducing visual studio code. In *Visual Studio Code Distilled*, pages 1–15. Apress, Berkeley, CA, 2021. Accessed: 2022-6-11.
- [19] Ken Kinder. Sublime text: one editor to rule them all? *Linux Journal*, 2013(232):2, 2013. Accessed: 2022-6-11.
- [20] Dalibor D Dvorski. Installing, configuring, and developing with xampp. *Skills Canada*, 2007. Accessed: 2022-6-11.
- [21] MDN Web Docs. Html: Hypertext markup language. *Mozilla Developers Network*, 2022-05-02. Accessed: 2022-6-11.

- [22] MDN Web Docs. Css: Cascading style sheets. *Mozilla Developers Network*, 2022-05-02. Accessed: 2022-6-11.
- [23] MDN Web Docs. Javascript — dynamic client-side scripting. *Mozilla Developers Network*, 2022-03-07. Accessed: 2022-6-11.
- [24] Luke Welling and Laura Thomson. *PHP and MySQL Web development*. Sams Publishing, 2003. Accessed: 2022-6-15.