# **DEDICATION**

**THIS WORK IS DEDICATED TO MY MOTHER**

# **ACKNOWLEDGMENTS**

Drafting this document would have not been possible without the contribution of some people who took upon themselves to see this work being accomplished. Our gratitude goes to the following people:

* The Resident Representative of IAI-Cameroon, **Mr. ABANDA ARMAND CLAUDE**, for his support, words of encouragement and the different advices on how to approach situations.
* The chief executive officer of Welldone planet, **Mr. BOM GABRIEL** and his staffs for trusting us by given us the opportunity to serve as interns in their prestigious enterprise.
* Our professional supervisor, **MR. TIOMELA JOU DANIEL** for his moral support, advice and professional expertise he shared with us.
* Our academic supervisor **Mrs. ONGUENE EBENYE** for her moral support, effort and advices throughout the academic year.
* To our academic teachers **Mr. MESSIO, Mr. AGBOR ANDERSON AND Mr. NDENGE** for their advices and assistance in realizing this document.
* To all my teachers of **AICS CAMEROON** for the knowledge they impacted me with for me to be where I am today.
* For my brothers and sister for their love and support.
* The countless contributors of open-source programming community, for their great help in learning basic skills and detecting and solving bugs.
* To all my classmates for their collaborative work throughout the academic year.

Table des matières

[**DEDICATION** 1](#_Toc174455610)

[**ACKNOWLEDGMENTS** 2](#_Toc174455611)

[**GLOSSARY** 7](#_Toc174455612)

[**ABSTRACT** 8](#_Toc174455613)

[**RESUME** 9](#_Toc174455614)

[**GENERAL INTRODUCTION** 10](#_Toc174455615)

[**PART I: INSERTION PHASE** 11](file:///D:\E-books\Cours%20Ecole\BA2\Stage\Youth%20employment%20app\Rapport%20de%20stage\My%20Report\Internship%20report.docx#_Toc174455616)

[**INTRODUCTION** 13](#_Toc174455617)

[I. WELCOME AND INTERGRATION 13](#_Toc174455618)

[1. Welcome 13](#_Toc174455619)

[2. Integration 13](#_Toc174455620)

[II. GENERAL PRESENTATION OF WELLDONE PLANET 13](#_Toc174455621)

[1. History and Missions 13](#_Toc174455622)

[2. Presentation of the structure 14](#_Toc174455623)

[b. Material Safety Data Sheet 14](#_Toc174455624)

[C. Resources 15](#_Toc174455625)

[3. Visions, Objectives of Welldone Planet 17](#_Toc174455626)

[b. Goals 18](#_Toc174455627)

[c. Attributions 20](#_Toc174455628)

[**CONCLUSION** 21](#_Toc174455629)

[**PART TWO: EXISTING SYSTEM** 22](file:///D:\E-books\Cours%20Ecole\BA2\Stage\Youth%20employment%20app\Rapport%20de%20stage\My%20Report\Internship%20report.docx#_Toc174455630)

[**INTRODUCTION** 24](#_Toc174455631)

[I. THEME PRESENTATION 25](#_Toc174455632)

[II. EXISTING SYSTEM STUDY 25](#_Toc174455633)

[III. CRITISISM OF THE EXISTING SYSTEM 25](#_Toc174455634)

[IV. PROBLEMATICS 25](#_Toc174455635)

[V. PROPOSED SOLUTION 25](#_Toc174455636)

[CONCLUSION 26](#_Toc174455637)

[**PART THREE: SPECIFICATION BOOK** 27](file:///D:\E-books\Cours%20Ecole\BA2\Stage\Youth%20employment%20app\Rapport%20de%20stage\My%20Report\Internship%20report.docx#_Toc174455638)

[**INTRODUCTION** 29](#_Toc174455639)

[I. CONTEXT AND JUSTIFICATION 30](#_Toc174455640)

[II. OBJECTIVES OF THE PROJECT 30](#_Toc174455641)

[A. General Objective 30](#_Toc174455642)

[B. Specific Objectives 30](#_Toc174455643)

[III. BENEFICIARIES AND TARGETS 30](#_Toc174455644)

[i. Beneficiaries: 30](#_Toc174455645)

[ii. Targets: 30](#_Toc174455646)

[IV. EXPRESSION OF NEEDS 30](#_Toc174455647)

[A. Functional needs 30](#_Toc174455648)

[B. Non-functional Needs 30](#_Toc174455649)

[V. PLANNING THE PROJECT 32](#_Toc174455650)

[A. Chronogram of activities 32](#_Toc174455651)

[B. Gantt Chart 34](#_Toc174455652)

[A. List of participants 34](#_Toc174455653)

[VI. ESTIMATED COST OF THE PROJECT 35](#_Toc174455654)

[A. Hardware Resources 35](#_Toc174455655)

[B. Software Resources 35](#_Toc174455656)

[C. Human Resources 37](#_Toc174455657)

[D. Total Project Estimated Cost 37](#_Toc174455658)

[VII. LIST OF PARTICIPANTS AND DELIVERABLES 38](#_Toc174455659)

[VIII. DeliverableS 38](#_Toc174455660)

[**CONCLUSION** 39](#_Toc174455661)

[**PART FOUR: ANALYSIS PHASE** 40](file:///D:\E-books\Cours%20Ecole\BA2\Stage\Youth%20employment%20app\Rapport%20de%20stage\My%20Report\Internship%20report.docx#_Toc174455662)

[**INTRODUCTION** 42](#_Toc174455663)

[I. PRESENTATION OF THE ANALYSIS 42](#_Toc174455664)

[1. COMPARATIVE STUDY OF UML AND MERISE 42](#_Toc174455665)

[MODELING WITH UML 2.5 43](#_Toc174455666)

[a) Structural Diagrams : 43](#_Toc174455667)

[b) Behavioral Diagrams : 44](#_Toc174455668)

[c) UML 2.5 diagrams overviews 45](#_Toc174455669)

[2. COMPARATIVE STUDY OF UNIFIED PROCESS 45](#_Toc174455670)

[THE PROCESS OF DEVELOPMENT OF A SOFTWARE 45](#_Toc174455671)

[a) A Unified Process 46](#_Toc174455672)

[b) The Two Track Unified Process (2TUP) 46](#_Toc174455673)

[a) The left branch (functional branch) 47](#_Toc174455674)

[b) The right branch (Technical branch) 47](#_Toc174455675)

[c) The middle branch (Realization or implementation branch) 47](#_Toc174455676)

[JUSTIFICATION OF THE ANALYSIS 48](#_Toc174455677)

[II. MODELING OF THE SYSTEM 49](#_Toc174455678)

[A. CAPTURE OF FUNCTIONAL NEEDS 49](#_Toc174455679)

[1. *Use Case Diagram* 49](#_Toc174455680)

[2. Communication Diagram 53](#_Toc174455681)

[3. Sequence Diagram: 54](#_Toc174455682)

[4. ACTIVITY DIAGRAM 57](#_Toc174455683)

[5. State Machine Diagram 61](#_Toc174455684)

[**CONCLUSION** 65](#_Toc174455685)

[**PART FIVE: CONCEPTION PHASE** 66](file:///D:\E-books\Cours%20Ecole\BA2\Stage\Youth%20employment%20app\Rapport%20de%20stage\My%20Report\Internship%20report.docx#_Toc174455686)

[**INTRODUCTION** 68](#_Toc174455687)

[I. TECHNICAL BRANCH 68](#_Toc174455688)

[A. GENERIC DESIGN 68](#_Toc174455689)

[a) Hardware diagram of the system 68](#_Toc174455690)

[d) High Level Architectural Diagram of the Software 68](#_Toc174455691)

[II. CAPTURES TECHNICAL NEEDS 68](#_Toc174455692)

[a) Physical architecture 68](#_Toc174455693)

[iii. Logical architecture 69](#_Toc174455694)

[III. RELATED UML DIAGRAMS 70](#_Toc174455695)

[Class Diagram: 70](#_Toc174455696)

[71](#_Toc174455697)

[b) State Machine Diagram 72](#_Toc174455698)

[c) Package Diagram 73](#_Toc174455699)

[**CONCLUSION** 76](#_Toc174455700)

[**PART SIX: REALISATION PHASE** 77](file:///D:\E-books\Cours%20Ecole\BA2\Stage\Youth%20employment%20app\Rapport%20de%20stage\My%20Report\Internship%20report.docx#_Toc174455701)

[**INTRODUCTION** 79](#_Toc174455702)

[I. ARCHITECTURE OF THE APPLICATION 80](#_Toc174455703)

[A. Logical architecture 80](#_Toc174455704)

[B. Physical architecture 80](#_Toc174455705)

[II. TECHNOLOGICAL CHOICE 80](#_Toc174455706)

[C. Choice of programming language and framework 80](#_Toc174455707)

[D. Choice of tools used 80](#_Toc174455708)

[**CONCLUSION** 97](#_Toc174455709)

[**PART SEVEN: FUCTIONALITY TESTING** 98](file:///D:\E-books\Cours%20Ecole\BA2\Stage\Youth%20employment%20app\Rapport%20de%20stage\My%20Report\Internship%20report.docx#_Toc174455710)

[**INTRODUCTION** 100](#_Toc174455711)

[**PART EIGHT: INSTALLATION AND USER GUIDE** 101](file:///D:\E-books\Cours%20Ecole\BA2\Stage\Youth%20employment%20app\Rapport%20de%20stage\My%20Report\Internship%20report.docx#_Toc174455712)

[**INTRODUCTION** 104](#_Toc174455713)

[I. INSTALLATION 104](#_Toc174455714)

[II. USER GUIDE 104](#_Toc174455715)

[**Conclusion** 105](#_Toc174455716)

[**GENERAL CONCLUSION** 106](#_Toc174455717)

[**ANNEXE** 107](#_Toc174455718)

[**BIBLIOGRAHY** 108](#_Toc174455719)

[**WEBOGRAPHY** 109](#_Toc174455720)

[**TABLE OF CONTENT** 110](#_Toc174455721)

# **GLOSSARY**

* **2TUP:** Two Track Unified Process
* **AICS:** African Institute of Computer Sciences
* **API**: Application Programming Interface
* **CSS**: Cascading Style Sheet
* **DBMS:** Database Management System
* **ER**: Entity Relationship
* **ERD**: Entity Relationship Diagram
* **JSON**: JavaScript Object Notation
* **GUI**: Graphical User Interface
* **HTML**: Hypertext Markup Language
* **HTTP**: Hypertext Transfer Protocol
* **IDE**: Integrated Development environment
* **SQL:** Structured Query Language
* **PDM:** Physical Data Model

# **ABSTRACT**

This project addresses the critical issue of youth unemployment in Cameroon by developing a digital platform designed to connect young job seekers with entry-level positions, internships, and freelance opportunities. Our **YOUTH EMPLOYMENT AND CAREER DEVELOPMENT PLATFORM** aims to facilitate the transition of young people into the workforce by providing a centralized hub for job search, career resources, and mentorship. By leveraging technology, the platform seeks to create a more efficient and accessible job market for young Cameroonians,

By combining these elements, the platform strives to create a dynamic ecosystem that fosters youth employability, entrepreneurship, and professional growth. Ultimately, the project envisions a Cameroon where young talent is empowered to contribute meaningfully to the nation

The project employs tools such as UML, React Native, Python, Django and SQLITE3 for efficient implementation. Through comprehensive testing and iterative development, the project aims to deliver a robust and user-friendly youth employment platform providing efficiency and convenience in job opportunity for camerounian youths.

Key words :

* Youth
* Intership
* Freelance
* Employment
* Job
* Workforce

# **RESUME**

Ce projet s’attaque au problème crucial du chômage des jeunes au Cameroun en développant une plateforme numérique destinée à mettre en relation les jeunes demandeurs d’emploi avec des postes d’entrée de carrière, des stages et des opportunités de freelance. Notre **PLATEFORME POUR L’EMPLOI ET LE DÉVELOPPEMENT DE CARRIÈRE DES JEUNES** vise à faciliter la transition des jeunes vers le monde du travail en offrant une plateforme centralisée pour la recherche d’emploi, les ressources de carrière et le mentorat. En tirant parti de la technologie, la plateforme cherche à créer un marché du travail plus efficace et accessible pour les jeunes Camerounais.

En combinant ces éléments, la plateforme vise à créer un écosystème dynamique qui favorise l’employabilité des jeunes, l’entrepreneuriat et le développement professionnel. En fin de compte, le projet envisage un Cameroun où les jeunes talents sont habilités à contribuer de manière significative à la nation.

Le projet utilise des outils tels qu’UML, React Native, Python, Django et SQLITE3 pour une implémentation efficace. Grâce à des tests complets et un développement itératif, le projet vise à fournir une plateforme d’emploi jeune robuste et conviviale offrant efficacité et commodité dans la recherche d’opportunités d’emploi pour les jeunes Camerounais.

Mots-clés :

* Jeunesse
* Stage
* Freelance
* Emploi
* Travail
* Main-d’œuvre

# **GENERAL INTRODUCTION**

The burgeoning youth population in Cameroon presents a significant opportunity for economic growth and development. However, the challenge of transitioning from academia to the professional world remains a formidable obstacle for many young graduates. This project addresses this critical issue by developing a comprehensive digital platform designed to facilitate the integration of young Cameroonians into the workforce.

The platform serves as a centralized hub for job seekers, employers, and mentors, offering a dynamic ecosystem that fosters career development and professional growth. By leveraging technology, the platform aims to revolutionize the traditional job search process, providing a more efficient, accessible, and inclusive platform for young people.

A key component of the platform is its focus on mentorship. Recognizing the invaluable role of experienced professionals in guiding young talent, the platform facilitates connections between mentors and mentees, creating opportunities for knowledge sharing and career guidance. Additionally, the platform incorporates a freelance marketplace, enabling young individuals to explore entrepreneurial ventures and gain practical work experience.

Through a user-centric design approach and the integration of cutting-edge technologies, the platform seeks to empower young Cameroonians by providing them with the tools and resources necessary to achieve their career aspirations. By addressing the challenges of youth unemployment and underemployment, the platform contributes to the overall social and economic development of the country.To obtain these objectives, we used 6 main sections or phases which are ;

1. **Insertion Document:** In this book, we shall present the company in which we spent our internship period and the way we were welcomed in the company. At the end of this section, we will elaborate on our theme.
2. **Existing System:** Here, we shall present the already present system in place, that is the one used for consultation and follow-up purposes.
3. **Specification Book:** In this book, we specify the needs of the user taking into considerations the time and cost of the project.
4. **Analysis Document:** Here, we shall present the analysis method chosen together with the presentation of all the diagrams used for the analysis of this project.
5. **Conception phase:** This presents the generic and detailed conception of the project to bring out real world constituents.
6. **The Realization phase:** This phase will permit us to visualize the implementation process of the solution.
7. **Test of functionalities:** In this phase, we shall present to you the different functionalities or modules of our application and how they work.
8. **The User Guide:** This elaborates on all conditions necessary to use the application and how to use it.

# **PART I: INSERTION PHASE**

Preamble

This section of our report will cover details of how we were welcomed in the host company, the company presentation and organization, and a brief introduction to our project.

Content overview

**INTRODUCTION**

1. **WELCOME AND INTERGRATION**
2. **GENERAL PRESENTATION OF WELLDONE PLANET**
3. **ORGANISATION OF WELLDONE PLANET**
4. **HARDWARE AND SOFTWARE RESOURCES**

**CONCLUSION**

# 

# **INTRODUCTION**

Professional integration is a process which allows an individual or a group of individuals to enter the labor market under conditions favorable to obtaining a job. As part of the academic internship, we were received as an intern within the premises of Welldone Planet for a period of three (03) months and the integration phase lasted 02 weeks. This part of the internship report presents the structure or internship was carried out, its operation, the reception of the student and the research theme during this period.

# I. WELCOME AND INTERGRATION

## Welcome

Within the Welldone Planet structure, we were welcomed on Monday July 1, 2024 by its Technical Director, Mr. TIOMELA JOU Daniel

## Integration

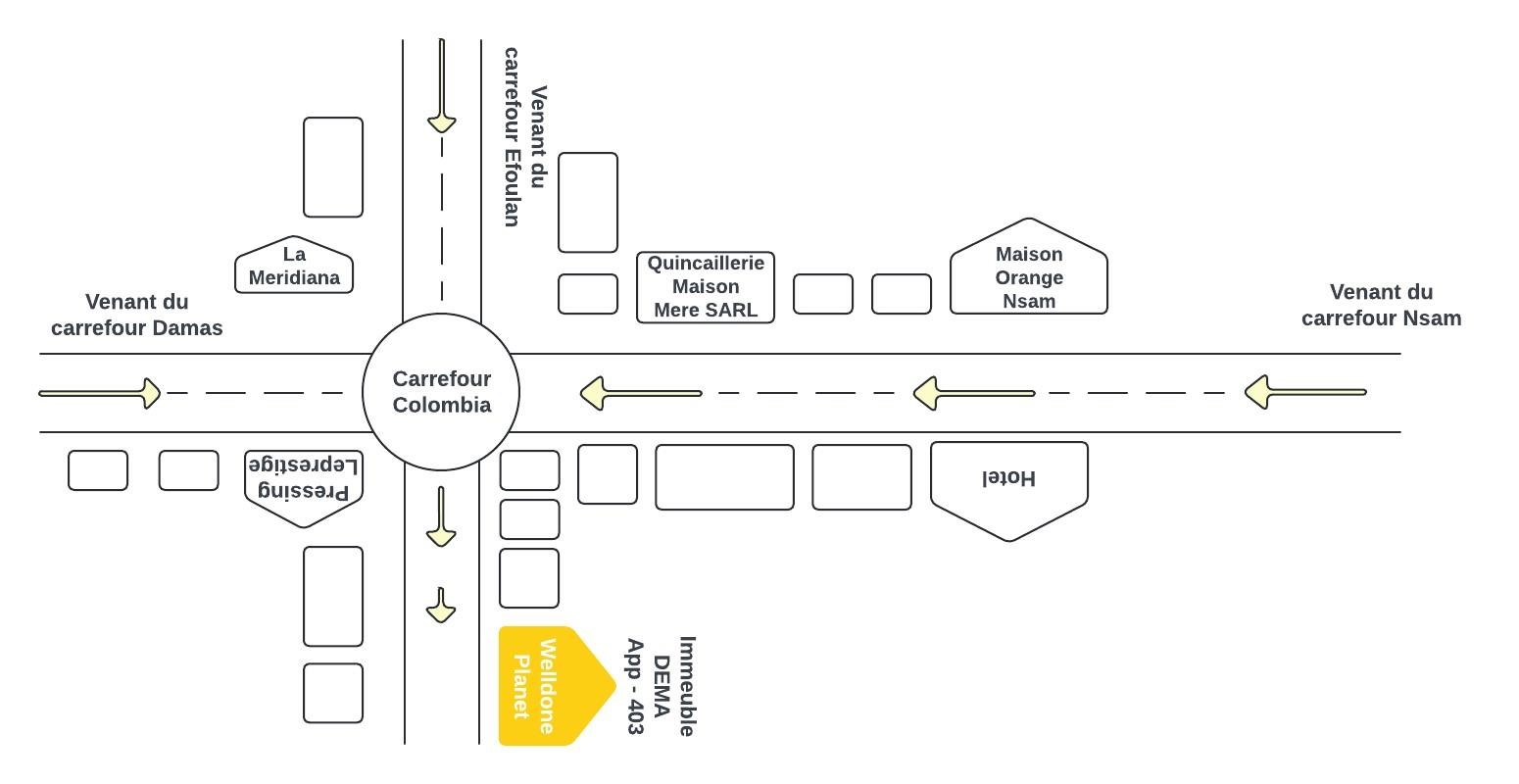
Mr. TIOMELA JOU Daniel organized a meeting during which he reminded us of the internal regulations of the establishment, its operation, its vision, its objectives, its services, its requirements and the development of the weekly schedule. Emphasis was placed on the conduct to be followed and compliance with the regulations in force. Several pieces of advice and recommendations were also given to us during various online sessions with the Technical Director. Then we argued about our theme.

# II. GENERAL PRESENTATION OF WELLDONE PLANET

## History and Missions

Welldone Planet was born from an ambitious project designed by a group of engineers graduated from the African Institute of Computer sciences (AIC) in Cameroon. Created in June 2023 and officially inaugurated on June 1, 2024 by the resident representative of AICS-CAMEROON, this company aims to become a leader in the provision of digital services. By advocating the emergence of the new information and communication technologies sector in Africa, and more particularly in Cameroon, Welldone Planet's mission is to inspire excellence and achieve innovation. Their slogan, "We believe that thanks to digital technology all dreams can come true. Let's Dream Big!" perfectly reflects their ambition to transform dreams into reality thanks to the power of digital technology. The company motto, “Inspire Excellence, Achieve Innovation, Dream Big,” embodies their commitment to creating a bright, technological future for the continent.

## Presentation of the structure



There

~~Efoulan~~

Coming from



DO

Welldone

Ann.

op

-

403

~~Building~~

a. Structure location plan

figur

e 1: Location plan

## b. Material Safety Data Sheet

table 1: Welldone Planet Material Safety Datan Sheet

|  |  |
| --- | --- |
| QUALITY | DESCRIPTION |
| The head office | Yaoundé (Nsam – Carrefour Colombia) |
| Legal form Type | SARL |
| of business | Private |
| Services | IOT/AI BIG DATA  Software Development  IT maintenance Web and Mobile development  Infographics  Cybersecurity  Digital Marketing IT training |
| Phone | (237) 677 223 241 |
| Website | www.welldoneplanet.com |
| E-mail | welldoneplanet@gmail.com |
| Languages used | French, English |

## C. Resources

i. Material resources

The company has numerous equipment listed in the following table.

table: Welldone Planet's material resources

|  |  |  |
| --- | --- | --- |
| DESIGNATION PICTURE | | |
| 3 HP desktops | |  |
| 3 Wifi Modem | |  |
| 3 plasma screens | |  |
| 1 fridge | |  |
| 1 canon printer | | Canoe |
| 1 HP laptop | |  |
| Ultiamte Kit for Rasberry | | OsYoo |
| Super starter kit | |  |
|  | Smart robots | SMART R |

. Software resources

table 2: Welldone Planet software resources

|  |  |
| --- | --- |
| DESIGNATION | PICTURE |
| Windows 10 operating system |  |
| Windows 11 operating system | ~~Home~~    ~~Windows11~~ |
| Adobe Creative Cloud Suite 2024 | ~~COLLECTION 0~~ |

# 3. Visions, Objectives of Welldone Planet

*a. Visions*

Welldone Planet's vision is to become a major catalyst in the field of information and communication technologies in Africa and Cameroon in particular, with a focus on innovation and excellence. Firmly believing that digital can turn dreams into reality, Welldone Planet is committed to creating forward-thinking technology solutions that meet the growing needs of society. By inspiring excellence and achieving innovation, the company aspires to propel Cameroon, and by extension Africa, towards a future where new technologies are at the heart of economic and social development.

## b. Goals

Welldone Planet sets ambitious goals to have a significant impact on the economic and social development in Africa. Based on a holistic approach and inclusive, the company aims to:

* Reduce poverty: By providing innovative technological solutions, Welldone Planet intends to create economic opportunities and improve living conditions of local populations, thus contributing to the reduction of poverty..
* Facilitate the development of human capacities: Through programs of quality training in computer science and information technology, the company aspires to strengthen the skills of individuals and professionals, allowing them to remain competitive in the job market.
* Holistic approach, including young people\*\*: Welldone Planet adopts a global which includes young people, whether they are educated or have dropped out of school, in order to fully integrate them into technological development and economically, providing them with opportunities for learning and growth.
* Support businesses in their growth\*: By using techniques based on optimization, bottleneck management and cost-effectiveness sustainable, Welldone Planet is committed to supporting businesses in their expansion, by improving their operational efficiency and economic performance

**4. Activities, Organization Chart and Responsibilities of**

**Welldone Planet**

a. Activities

Welldone Planet offers a full range of technology services to meet to the diverse needs of its customers. The company's main mission is to solve IT problems that businesses encounter on a daily basis in the areas following:

* IOT/IA BIG DATA
* Software Development
* IT maintenance
* Web and Mobile Development
* Infographics
* Cybersecurity
* Digital Marketing
* IT training

b. Organizational chart

The organization chart is above all a communication tool intended to

Facilitate understanding of existing relationships and links within society. To this end, it allows fora global view of the company in terms of services, divisions and more. **Welldone Planet** has six departments within it and we have been assigned to technical direction. This is the division colored in orange in the organization chart below.

***figure 2 : Welldone Planet organization chart***

## c. Attributions

The Welldone Planet Company includes several executive-led divisions main competent persons:

* **The Human Resources Division (HRD)**  headed by a director of human resources which is responsible for personnel management, training and skill developpement
* **The Sales and Marketing Division (DCM**) headed by a director sales and marketing who is responsible for sales, marketing and development commercial.
* **The Communication Division (DCOM)** headed by a director of communication of the company which is responsible for sales, marketing and Business Development**.**
* **The Research and Development Division (DR&D)** headed by a director of research and development who is responsible for innovation and development of new products
* **The Financial Division (DRMF)** headed by a director of financialresources who is responsible for financial management, budgets and accounting.
*  **The Technical Division (DT)** headed by a technical director who is responsible for the technical management, production and maintenance**.**.
* **Secretariat** for invoicing, photocopying, customer registration

# **CONCLUSION**

The insertion phase was a very educative one, we got to know more about the history of the company, how it all started till the level it is right now, how it is then we discovered that it’s a very serious and goal oriented company. We got to Know the staff and other interns, in fact we felt at home. Finally, we decided to work on the theme **“CONCEPTION AND REALISATION ”.** Now that we have chosen the theme we will work on; we can move on to the specification book of our report where we will identify the objectives and requirements for our project.

# **PART TWO: EXISTING SYSTEM**

Preamble

The purpose of the user guide is to provide users of our platform with step-by-step instructions on how to install and use the system.

Content overview

**INTRODUCTION**

1. **THEME PRESENTATION**
2. **EXISTING SYSTEM STUDY**
3. **CRITISISM OF THE EXISTING SYSTEM**
4. **PROBLEMATICS**
5. **PROPOSED SOLUTION**

**CONCLUSION**

# 

# **INTRODUCTION**

In this section of our report, we will delve into the various research conducted for the realization of this project. We will explore the existing study, critique the current system, identify the problem statement, and propose a solution. Through thorough analysis and investigation, we aim to provide a comprehensive understanding of the research undertaken to address the challenges and develop an effective solution. By examining the existing landscape and presenting our proposed approach, we strive to contribute towards the advancement and improvement of the project's objectives.

# THEME PRESENTATION

Our theme is « **CONCEPTION AND IMPLEMENTATION OF A YOUTH EMPLOYMENT AND CAREER DEVELOPMENT PLATFORM** ». Youth unemployment in Cameroon is a pressing issue characterized by a mismatch between the skills of young graduates and the demands of the job market. Traditional job search methods are often inefficient and inaccessible to many young people. Moreover, a lack of mentorship and career guidance exacerbates the challenges faced by young job seekers. This project addresses this issue by developing a comprehensive digital platform designed to connect young job seekers with potential employers, mentors, and freelance opportunities. By leveraging technology, the platform aims to create a dynamic ecosystem that fosters youth employability, entrepreneurship, and professional growth. By providing a centralized hub for entry-level positions, internships, and freelance projects, connecting young professionals with experienced mentors to offer guidance and support and creating a marketplace for freelance services, empowering young people to generate income and build portfolios all while utilizing user-friendly interfaces, advanced search functionalities, and personalized recommendations.

# EXISTING SYSTEM STUDY

**As existing system, LinkedIn** has established itself as a dominant force in the professional networking sphere. Its core features revolve around profile building, job searching, and professional relationship management. Users can create detailed profiles showcasing their experiences, skills, and endorsements. The platform offers a vast job board with various filters to refine searches. LinkedIn also facilitates networking through connections, groups, and messaging. Furthermore, it provides features like company pages, job recommendations, and insights into industry trends.

# CRITISISM OF THE EXISTING SYSTEM

Critique of the existing systems

|  |  |  |  |
| --- | --- | --- | --- |
| **LIMITATION** | **CONSEQUENCE** | | **PROPOSED SOLUTION** |
| Focus on professional networking | | Difficulty for young job seekers to find entry-level positions. | Offer specialized search filters and job categories tailored for early-career individuals and specific industries. |
| Predominantly used by experienced professionals | | Reduced visibility for young job seekers | Implement features to highlight early-career talent, such as portfolio showcases and skill assessments. |
| Limited focus on mentorship | | Insufficient guidance for young professionals. | Establish a dedicated mentorship program and structured mentorship frameworks. |
| Limited focus on specific regions or industries | | Reduced relevance for job seekers in certain areas or industries. | Offer localized job boards and industry-specific filters. |
| Limited freelance opportunities | | Reduced income potential for young people | Integrated freelance marketplace with project management tools |
| Paiment for some core features | | Financial barrier for job seekers | Offer core features for free, with premium options for additional benefits (e.g., advanced job alerts, career coaching). |

# PROBLEMATICS

Cameroon, like many developing countries, faces a significant challenge in youth unemployment. A large portion of the young population struggles to transition from academia to the professional world, often due to a mismatch between skills and available opportunities. Traditional job search methods are inefficient, and access to mentorship and professional networks is limited. This situation leads to a loss of potential for both individuals and the nation's economy.

# PROPOSED SOLUTION

After our study and criticism of the existing situation, we propose a platform that will be able to provide:

1. **Centralized Job Board:** A platform that aggregates job opportunities from various sectors, providing a one-stop shop for young job seekers. This enhances visibility and accessibility to potential employment.
2. **Mentorship and Career Guidance:** Recognizing the importance of mentorship in professional development, the platform facilitates connections between experienced professionals and young job seekers. By providing a structured mentorship program, it offers invaluable guidance, support, and industry insights, enabling young people to navigate their career paths with confidence.
3. **Freelance Opportunities:** The platform expands employment possibilities by incorporating a freelance marketplace. This allows young individuals to showcase their skills and earn income while gaining practical work experience. By fostering entrepreneurship and self-employment, the platform contributes to the growth of the gig economy in Cameroon.
4. **User-Centric Design:** The platform is designed with the needs of young job seekers in mind. Its intuitive interface, mobile compatibility, and focus on accessibility ensure that it is user-friendly and inclusive. By prioritizing the user experience, the platform enhances job seeker satisfaction and engagement.
5. **Data-Driven Approach:** The platform utilizes data analytics to understand user behavior, identify trends, and optimize its services. By leveraging data insights, the platform can refine its job matching algorithms, improve user experience, and measure its impact on youth employment.
6. **Community Building:** The platform fosters a sense of community among young job seekers by providing networking opportunities and online forums for discussion. By creating a supportive environment, the platform helps young people connect with peers, share experiences, and build professional relationships.By combining these features, our platform offers a holistic approach to youth employment, empowering young people to take control of their career paths and contribute to the socio-economic development of Cameroon.

# CONCLUSION

In this section, we conducted research to support the realization of our youth employment platform.We analyzed the existing system, identified its limitations, and proposed a solution. Our findings contribute to improving the project's objectives. While existing job boards and career platforms offer some solutions, they often fall short in addressing the specific needs of young job seekers. Our platform differentiates itself by combining a centralized job board with comprehensive career development tools, mentorship opportunities, and a freelance marketplace. By creating a holistic ecosystem, we empower young Cameroonians to navigate the job market successfully, contributing to the overall economic and social progress of the country.

# **PART THREE: SPECIFICATION BOOK**

Preamble

This section of our report will cover details on the product to be delivered with the agreement of the client(s) and the solution provider. In this section, we are going to present the context in which we are to but in place a platform, what the system should do and how the system should do it

Content overview

**INTRODUCTION**

1. **CONTEXT AND JUSTIFICATION**
2. **OBJECTIVES OF THE PROJECT**
3. **BENEFICIARIES AND TARGETS**
4. **EXPRESSION OF NEEDS**
5. **PLANNING OF THE PROJECT**
6. **ESTIMATED COST OF THE PROJECT**
7. **CONSTRAINTS**
8. **DELIVERABLES**

**CONCLUSION**

# **INTRODUCTION**

The specification book gives us the different directives on the product to be delivered with the agreement of the client(s) and the solution provider. In this section, we are going to present the context in which we are to but in place a platform, what the system should do and how the system should do it. These specifications are to avoid the production of inadequate results. We will also include project detail such as the team involve, constraints, the budget, deadlines, constrains and the deliverable.

# CONTEXT AND JUSTIFICATION

# OBJECTIVES OF THE PROJECT

## General Objective

## Specific Objectives

# BENEFICIARIES AND TARGETS

## Beneficiaries:

The beneficiaries of our project will be the Camerounian citizens

## Targets:

For this project our target is the cameroun government

# EXPRESSION OF NEEDS

## Functional needs

## Non-functional Needs

* **Performance:**
  1. Responsiveness: The application should respond quickly to user interactions and provide a smooth user experience.
  2. Efficiency: The application should be optimized to perform tasks efficiently, minimizing resource usage and response times.
  3. Scalability: The application should be capable of handling increasing user loads and data volumes without significant performance degradation.
  4. Reliability: The application should be reliable and available, minimizing downtime and disruptions.
* **Usability:**
  1. Intuitive Interface: The application should have a user-friendly and intuitive interface that is easy to navigate and understand.
  2. Accessibility: The application should be accessible to users with disabilities, complying with accessibility standards and guidelines.
  3. Multilingual Support: The application should support multiple languages to cater to a diverse user base.
  4. Consistency: The application should maintain consistent design elements and interactions throughout different screens and modules.
* **Security:**
  1. User Data Protection: The application should securely store and handle user data, following best practices for encryption and data privacy.
  2. Authentication and Authorization: The application should provide secure user authentication mechanisms and access controls to protect user accounts and data.
  3. Secure Transactions: The application should ensure secure transmission and handling of sensitive information, such as payment details.
  4. Data Backup and Recovery: The application should have robust backup and recovery mechanisms to protect against data loss and enable quick recovery.
* **Compatibility:**
  1. Cross-Platform Compatibility: The application should be compatible with different operating systems and devices, including mobile and web platforms.
  2. Browser Compatibility: The web application should be compatible with major web browsers, ensuring consistent functionality and appearance.
  3. API Integration: The application should be able to integrate with external systems or APIs seamlessly.

|  |
| --- |
| **Performance Monitoring and Analysis:** |



* 1. Logging and Monitoring: The application should log events and errors for monitoring and troubleshooting purposes.
  2. Analytics and Insights: The application should provide analytics and insights on user behavior, system performance, and usage patterns.
  3. Performance Optimization: The application should continuously optimize its performance based on analytics and monitoring data.
* **Maintainability:**
  1. Modularity: The application should be designed with a modular structure, making it easier to maintain and enhance individual components.
  2. Code Quality: The application's code should follow best practices and coding standards, ensuring readability and maintainability.
  3. Documentation: The application should have comprehensive documentation, including installation instructions, user guides, and API documentation.

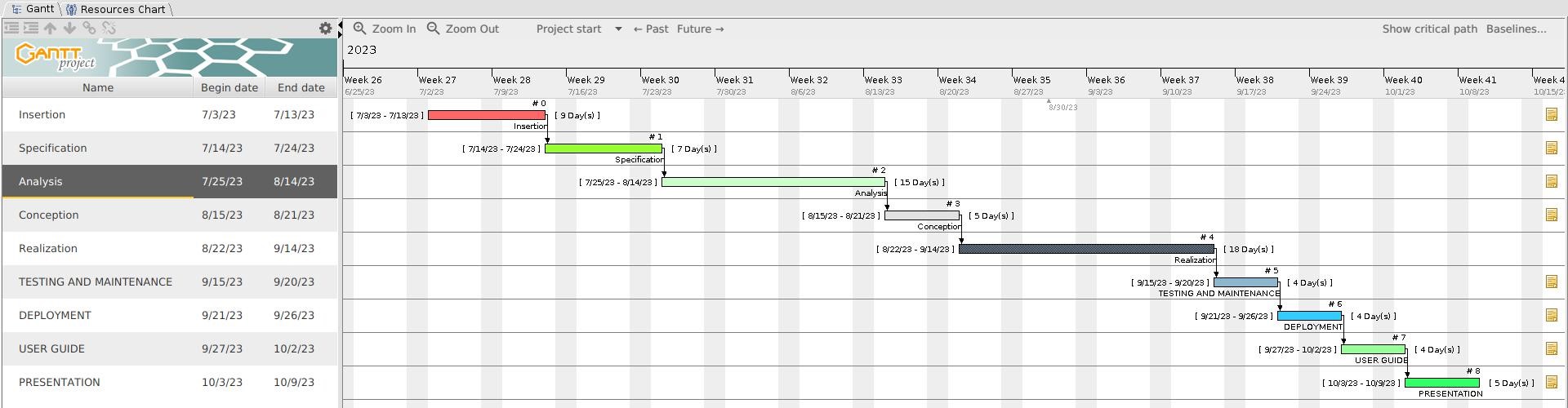
# PLANNING THE PROJECT

## Chronogram of activities

*Table 8:Chronogram of activities*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PHASE | OBJECTIVE | OUTPUT | DURATION | PERIOD |
| Insertion | Welcome and installation into the into the company, and the attribution of themes | Insertion report | 2weeks | 01st to 14th June 2022 |
| Specification | Assessment of  functional needs | Specification book | 1week 6days | 15th to 28th June 2022 |
| Analysis | Analysis of the system | Analysis book | 3weeks | 29th June to 19th July  2022 |
| Conception | Hardware and  software conception | Conception book | 2weeks | 20th July to 09th august  2022 |
| Realization | Realization of the system | Realization document | 4weeks 2days | 10th august to 09th  September |
| Deployment | Deployment of the system | Testing of the software and debugging | 1week 2days | 12th to 20th September |
| Writing user guide | Instructions on how to use the software and  hardware | User guide | 1week 3days | 21st to 30th September |

## Gantt Chart



## List of participants

|  |  |  |
| --- | --- | --- |
| NAME | TITLE ROLE | |
| Mrs. ONGUENE EBENYE | Lecturer at AICS-Cameroon | Academic supervisor |
| Mr. TIOMELA DANIEL | Computer science engineer at WELLDONE PLANET | Professional supervisor |
| TCHABAT TCHOUATE PETTANG VALIER | software engineering student at AICS-Cameroon | Analyst and developer |

# ESTIMATED COST OF THE PROJECT

## Hardware Resources

*Table 4:Hardware Resource (Source: https://www.scribd.com/document/561202966/mercuriale-2022)*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RESOURCE**  **S** | **HARDWARE** **USAGE** **QUANTITY** | | | **UNIT**  **COST(FCFA)** |
| COMPUTER | DELL  Windows 10 pro 64 bits,  Latitude E5440, 8GB RAM, 500GB hard disk; intel core i5. | main resource for the  accomplishment of this project; report writing, designs, analysis, coding | 1 | 368000 |
| Mobile phone | SAMSUNG A02  4 GB RAM, 64 GB hard disk | Testing of functionalities. | 2 | 80000 |
| Removable disk | 8GB USB key | For file transfer from one computer to another. | 1 | 6325 |
| CD | CD-ROM |  | 3 | 3000 |
| MODEM | 4g LTE-advanced **mobile**  **Wi-**  **Fi** hotspot **mq531** 150mps | Used to share internet access amongst devices | 1 | 23000 |
| **Total 1** |  |  |  | **559862** |

## Software Resources

*Table 5:Software Resources(https://www.g2.com/categories/pricing)*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RESOURCES | HARDWARE | USAGE | QUANTITY | UNIT COST(FCFA) |
| Development tool | Visual studio code | Code editing | 1 | Open source |
| Project planner | Gantt project | Project planning tool | 1 | Free software |
| Operating system | Microsoft windows 10 pro | Computer operating system | 1 | 103000 |
| Text Editor | Microsoft office word 2019 | For typing and formatting of  report | 1 | 161000 |
| Presentation | Microsoft office PowerPoint 2019 | For making  presentations | 1 | 84230 |
| Modelling tool | Visual-paradigm Modeler version | Modelling the system in uml | 1 | 3350 |
| Geo-location plan designing tool | Microsoft Visio 2007 | To draw  location plan | 1 | Free version |
| Web browser | Google chrome | For running and  testing application | 1 | Free version |
| Database management system | PostgreSQL | Communing  with the  database | 1 | Open source |
| **Total 2** |  |  |  | **351580** |

## Human Resources

*Table 6:Human Resource (Source: https://www.paylab.com/)*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RESOURCES | NUMBER | COST PER DAY | NUMBER OF DAYS | COST (FCFA) |
| **Project manager** | 1 | **250000** | **60** | **15000000** |
| **Analyst** | 1 | **150000** | **30** | **4500000** |
| **Designer** | 1 | **130000** | **25** | **3250000** |
| **Programmer** | 1 | **100000** | **32** | **3200000** |
| **TOTAL 3** | **4** |  |  | **25950000** |

## Total Project Estimated Cost

*Table 7:Total Project Estimate Cost*

|  |  |  |  |
| --- | --- | --- | --- |
| Total 1 (FCFA) Total 2 (FCFA) | | Total 3 (FCFA) | Overall total (FCFA) |
| **559862** | **351580** | **25950000** | **26861442** |
| **Margin of error = 10% of Overall total**  **=2686144.2 FCFA**  **Final Total = Overall Total + margin of Error**  **=29547586.2 FCFA** | |  |  |

# LIST OF PARTICIPANTS AND DELIVERABLES

# DeliverableS

At the end of the specified development timeline for our software, the deliverable will consist of a Report that will include:

In project management, any component materializing the result of a realization service is called a deliverable. In the case of our project, the deliverables are:

1) A report composed of the following document

1. The insertion document.
2. The Existing System
3. The specification book.
4. The analysis phase.
5. The conception phase.
6. The realization phase.
7. Test of functionalities.
8. The user guide.
9. A CD-ROM with:
10. The source code of the software
11. Database
12. A soft copy of the internship report
13. A corresponding PowerPoint
14. A prototype of our projec

# **CONCLUSION**

The specifications book contains detailed information about the project's objectives, requirements (both functional and non-functional), team members, project plan, cost, and constraints. This document serves as a guide to develop a solution that meets the project's needs while avoiding errors and compatibility issues. The specifications book acts as a foundation for the analysis phase of the project, where we carefully examine the provided information to determine the best approach and ensure a successful outcome.

# **PART FOUR: ANALYSIS PHASE**

Preamble

After specification book, we have the Analysis phase which permits us to represent a detailed analysis of the limitations identified in our context, and our solution, through a software development process and modelling language.

Content overview

**INTRODUCTION**

1. **PRESENTATION OF THE ANALYSIS METHOD**
2. **Comparative Study of UML and MERISE**
3. **Comparative Study of Unified Process**
4. **MODELING OF THE SYSTEM**
5. **Use Case Diagram**
6. **Communication Diagram**
7. **Sequence Diagram**
8. **Activity Diagram**

**CONCLUSION**

# **INTRODUCTION**

The analysis book permits us to examine in an explicit way the existing system, it’s limitations and how we can remedy them. We will also describe in details the modeling language known as UML (Unified Modeling Language) which is coupled with Two Tract Unified Process (2TUP) to form a method and its justification why we decided to use it in preference of another. then we will dive directly into the modeling of the proposed solution consisting of diagrams that meets the requirements of the functional needs.

# PRESENTATION OF THE ANALYSIS

## COMPARATIVE STUDY OF UML AND MERISE

We have studied in details some analysis methods by looking at its objectives, its structuring, its pros and cons, so as to choose an analysis method which is reliable and adaptive to our project, below are some analysis methods we studied and a brief explanation in order for you to understand why we made our choice;

* 1. **MERISE:** it is an information system design and development widely used in France. It was first early introduced in 1980s.
  2. **Agile:** this methodology is growing in popularity, thanks to highly competitive business environment and increased innovation. In general, agile methodologies prioritized shorter, interactive cycle and flexibility.
  3. **Scrum:** this is the most popular agile development framework because it is relatively simple to implement. It also solves so many problems that software developers struggle with in the past, convoluted development cycles, project plan, and shifting production schedules. This methodology allows for rapid development and testing, especially with small teams.
  4. **APF:** which stands for ADAPTIVE PROJECT FRAMEWORK, it grows from the difficulty in managing most IT projects using traditional project management methods due to uncertain and changing requirement. APF begins with a requirement breakdown structure (RBS) to define strategic goals based on productive requirements, functions, sub-function and features. The project proceeds in iterative stages, and at the end of each step, teams evaluate previous results to improve performance and practices.
  5. **XP:** which stands for Extreme Programming is a software development methodology that advocate frequent releases in short development cycles, which is intended to include checkpoints for the adoption of new customer requirements and improve productivity. This methodology takes its name from the idea that the traditional software engineering practices are taken to extreme levels.

## MODELING WITH UML 2.5

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.

The Unified Modeling Language (UML) was standardized in January 1997 by the Object Management Group (OMG) which is an American association created in 1989 and aims to promote and standardize the object model in all it forms. In 2005, UML was also published by the international organization for standardization (ISO) as an approved ISO standard. Uml since 2015 is in its version 2.5. This version consists of fourteen diagrams classified into structural and behavioral diagrams

### Structural Diagrams :

Structural diagrams represent the static components of a system; they emphasize on what should be in the system we are modeling. They include:

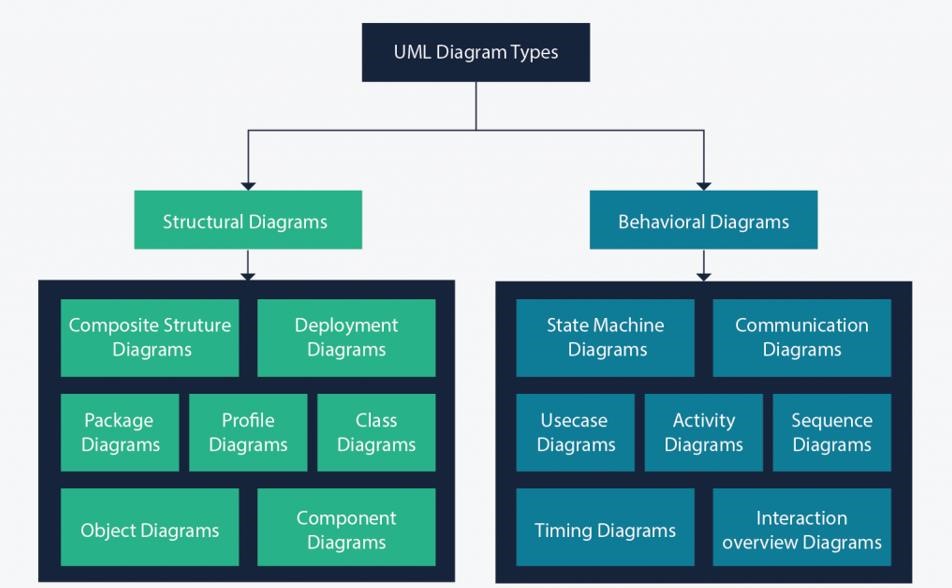
* Class diagram;
* Object diagram;
* Package diagram;
* Composite structural;
* Deployment diagram;
* Component diagram;
* Profile diagram.

### Behavioral Diagrams :

The behavioral diagrams capture the dynamic state of a system; they emphasize on what should happen in the system we are modeling. They are:

* + Use case diagram;
  + Activity diagram;
  + State machine diagram;
  + Sequence diagram;
  + Communication diagram;
  + Interaction overview diagram;
  + Timing diagram.

### UML 2.5 diagrams overviews



*Figure 4: UML 2.5 diagrams overview (source: https://creately.com/blog/diagrams/uml-diagram-types-examples/)*

It is important to note that uml is a modeling language and not a method or procedure. Hence, to give it an approach, we need to associate UML to a Unified process (UP) in order to give our conception a methodology to follow. A unified process is a generic method for developing software. This implies it is necessary to adapt the UP to the context of the project, team domain and or the organization. We will use the Two Tracks Unified Process (2TUP) throughout our project to implement our solution.

## COMPARATIVE STUDY OF UNIFIED PROCESS

## THE PROCESS OF DEVELOPMENT OF A SOFTWARE

A process can be defined as a partially sequence of steps that permits us to obtain software systems or evolution of an existing one. The main objective of software development is the production of quality software that response to the needs of the users during a particular time and at a particular cost.

### A Unified Process

A unified Process is a process of development of software constructed on UML; it is iterative, incremental, centered on architecture, driven by use cases and requirements.

**Iteration** is distinct sequence of activities with a basic plan and evaluation criterion that produces an internal or external output. Either the content of an iteration is improved or the evolution of the system is evaluated by users.

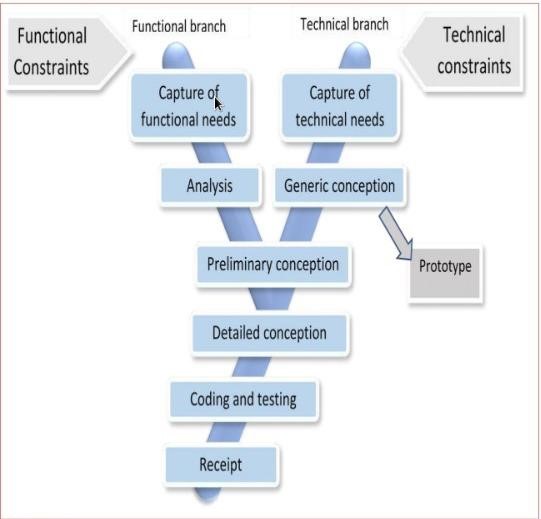
**An increment** is the difference between two released products at the end of two iterations. Each iteration that the group is capable of integrating the technical environment in order to develop a final product and give users the possibility of having tangible results.

**Centered on architecture** the different models derived during the establishment of system must be reliable and coherent.

**Driven by use case and requirements** enables the clear definition of a users’ needs and priorities respectively thereby minimizing the risk of project failure.

### The Two Track Unified Process (2TUP)

2TUP is a unified process which is belt on UML and has as objective to bring solution to constraints of functional and technical changes imposed on information systems by strengthening controls on development capacities. It proposes a Y-sharped development life cycle that separates the functional aspect from the technical aspects, and the merging of these two forms the implementation aspect. 2TUP distinguishes therefore two branches: the functional and technical branches, the combination of the result of these two branches forms the third: the realization branch – where we realize our system. The diagram bellow illustrates the branches of 2TUP.



### The left branch (functional branch)

It captures the functional needs of a system. This ensures the production of software that meets the needs/requirements of the user. The analysis here consists of studying precisely the functional specification in order to obtain an idea of what the system is going to realize and its result does not depend on any technology.

### The right branch (Technical branch)

The technical branch enumerates the technical needs and proposes a generic design validated by a prototype. The technical needs include constraints and choices related to the conception of the system, the tools and equipment as well as the integration constraint with the existing system condition. The different diagrams are shown in the table below.

### The middle branch (Realization or implementation branch)

In this branch, we study the preliminary conception, detailed conception, and documentation of the system. The realization branch supports the following:

**Preliminary conception:** This is the most sensitive step of 2TUP as it is the confluence of the functional and technical branch. It is completed when the deployment model, the operating model, the logical model, interphases and the software configuration model are defined.

**Detailed conception:** This is the detailed design of each feature of the system.

**Coding and testing:** This are the phase where we program the designed features and test the coded features.

**The recipe:** Also known as the deliverables is the validation phase of the functions of the developed system.

## JUSTIFICATION OF THE ANALYSIS

The reason why we chose UML modelling language and the software development process 2TUP instead of many others that exist, include:

UML is the current standard for programming in an object-oriented language. For this reason, it is widely understood and well known making it easy for a new programmer to join the project and be productive from the very first day.

UML diagrams allow teams to virtualize how a project is or will be working, and they can be used in any field, not just software engineering. The diagrams will allow teams to virtualize together how a system or a process will work or did work. It can provide new ideas for how teams have to collaborate to achieve the goal of the workflow process.

2TUP is centered around the creation and maintenance of a model, rather than the production of mountain documents.

2TUP is user oriented as it permits the development of software that responds to the needs of the users through the study of the user needs.

2TUP is iterative and incremental, hence it enables the project team to produce refined amelioration if necessary and easily integrate it in the already existing system. 2TUP by permitting the project team identify and test the key functionalities of the system limits the risk related to building the system.

# MODELING OF THE SYSTEM

## A. CAPTURE OF FUNCTIONAL NEEDS

The first step of the left (functional) branch of Two Track Unified Process (2TUP) is the capture of the functional needs. At this step, we capture the intended behavior of the system that maybe express as services, tasks or functions the system is required to perform.

## Use Case Diagram

#### Definition

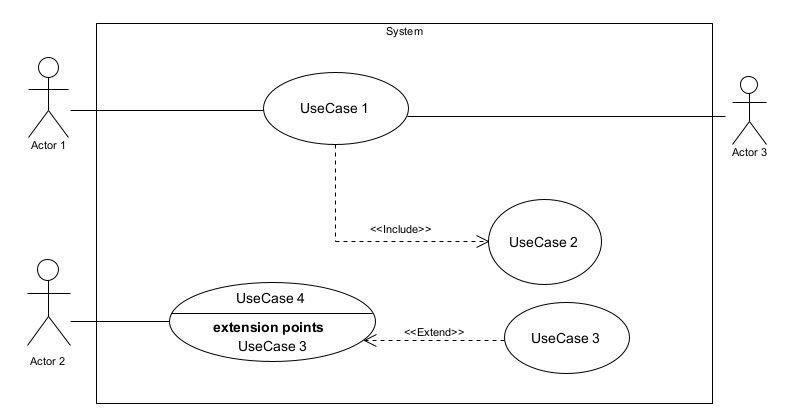
A use case diagram shows the functionalities of a system, their interdependencies and how they relate with actors of the system. A use case is a specification of behaviour.

The main objectives of the use case diagram are:

* Provide a high-level view of the system;
* Identify the functions of the system.

Use case diagrams are completed with a textual description of each use case that is intended to define the use case in greater details.

#### Formalism



*Figure 6: Use Case diagram formalism*

*Table 10:Use case diagram component*

|  |  |  |
| --- | --- | --- |
| Elements Notation Description | | |
| Actors |  | Represents an entity that directly interacts with the system. The actor is what performs the different possible actions of the system. |
| Use case |  | A use ase represents a |
|  | Use Case  1 | functionality of the system. It is an action that can be performed by an actor. |
| Association |  | it indicates that an actor takes part in a use Case. |
| Include |  | An inclusion denotes that an included action must be performed before the including action can be performed. |
| Extend |  | An extension denotes that an extending action may be performed while an extended action is being performed. |
| Generalization |  | This shows that an actor or a use case is a kind of another abstract or concrete actors can be defined and later specialized using generalization relationship. |
| System |  | It is a container of use cases which interact with external actors |

#### The Actor of our System

After the study of the current system in AICS-Cameroon, in other to implement our proposed solution, we identified the actors listed in the table below.

*Table 11:Actors of our system.*

|  |  |
| --- | --- |
| **Actor** | **Role** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **SMS API** |  |

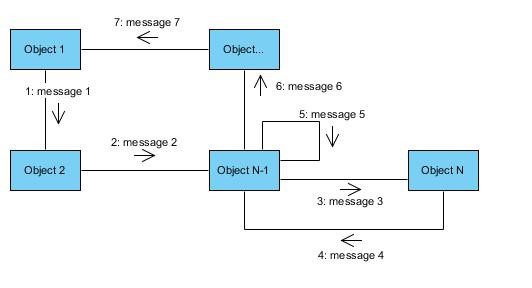
#### General Use Case Diagram

### Communication Diagram

1. Definitions:

It is a diagram which is used to show the relationship between the actors of a system, both the sequence and the communication diagrams represent the same information but differently. Instead of showing the flow of message. It depicts the architecture of the object residing in the system as it is based on object-oriented programming.

1. Formalism



*Figure 10:Communication Diagram Formalism*

|  |  |  |
| --- | --- | --- |
| **Element** | **Representation** | **Description** |
| **Object** |  | An actor represents an individual participant in the interaction conversation. |
| **link** |  | It initiates an association it connects two objects together for them to communicate. |
| **Actor** |  | A role play by an entity that interacts with the subjects. |
| **message** |  | Defines a particular communication between lifelines in an interaction. |

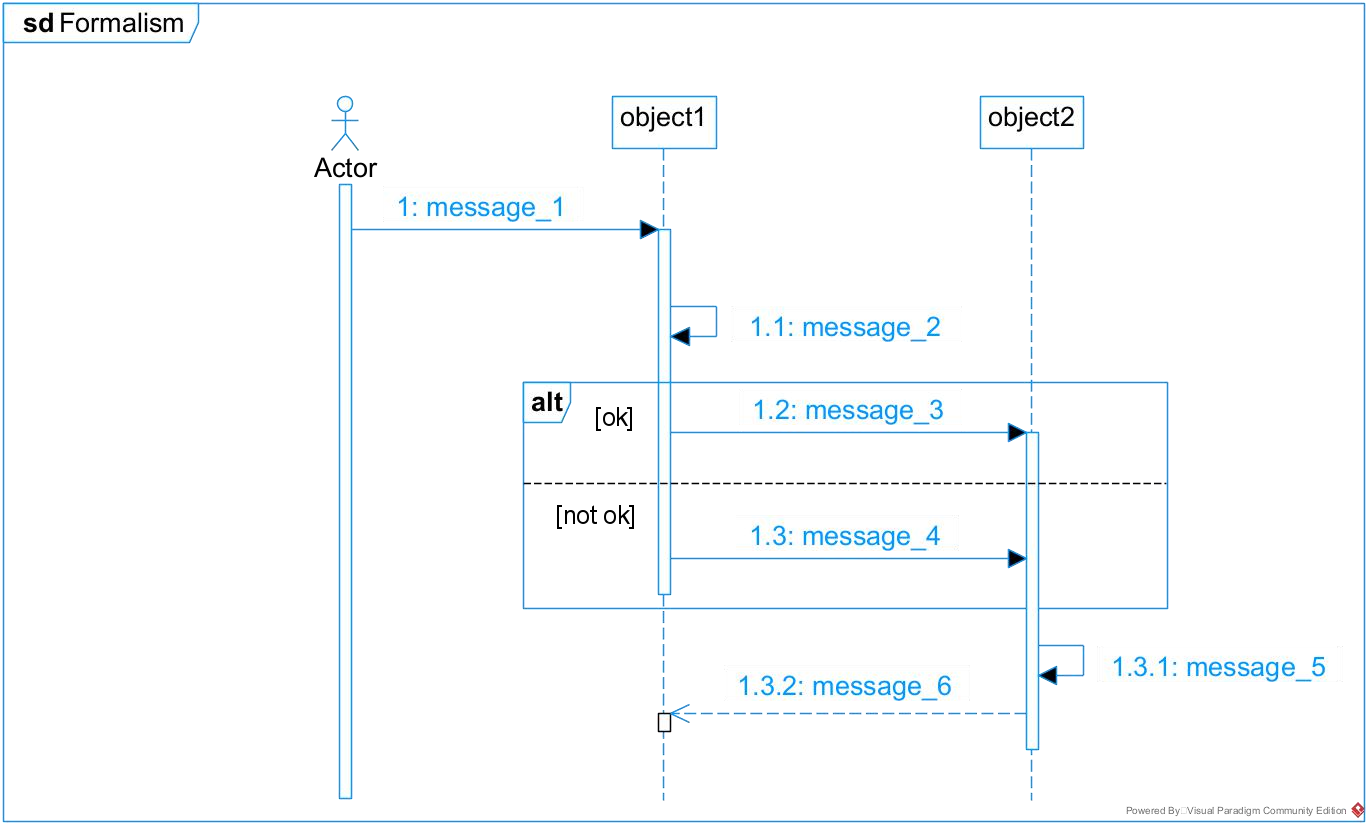
*Table 18:Communication Diagram Components*

DSEI Communication Diagram for Authentication

1. Sequence Diagram:
2. Definition:

A sequence diagram is an interaction diagram which represents the flow of message between elements in a system, it is termed as an event diagram. It portrays the communication between any two lifelines as a time-ordered sequence of events.

1. Formalism:

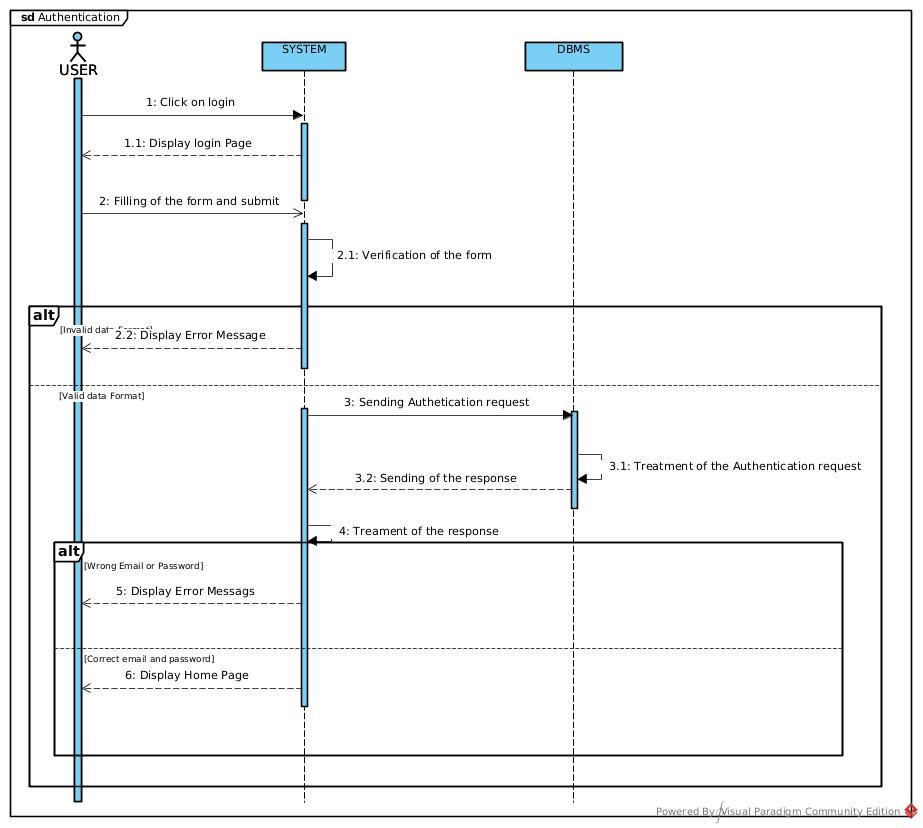


*Figure 14: Sequence Diagram formalism*

|  |  |  |
| --- | --- | --- |
| **Element** | **Representation** | **Description** |
| **Lifeline** |  | An individual participant in a sequence diagram, it is position at the top of the diagram. |
| **Combined fragment** |  | It represents a choice of behavior in which at most one operand will be chosen. |
| **Messages** |  | These are arrows which shows the direction of  message flow. We have the synchronous, the  asynchronous and the selfmessages. |
| **Activation** |  | It describes the time period in which an operation is performed by an element. |

*Table 19:Sequence Diagram Components*

DSEI Authentication Sequence Diagram

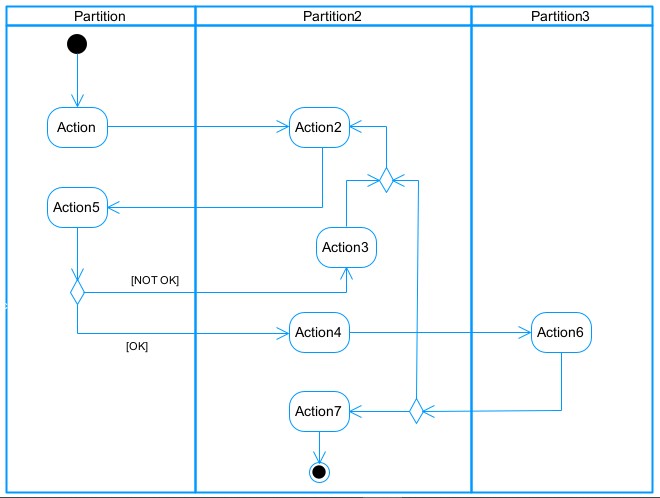


### ACTIVITY DIAGRAM

#### Definition

An activity diagram is a graphical representation of workflows that show the steps needed in the realization of a process; showing the details from a start point to an end point through all decisions and actions that can possible be performed. Activity diagrams are intended to model both the computational and organizational process. They flow can be sequential, branched or concurrent. Below is an activity diagram formalism.

#### Formalism:



*Figure 10:Activity Diagram Formalism*

*Table 14:components of an Activity Diagram*

|  |  |  |
| --- | --- | --- |
| **Elements Diagrammatic Representation** | | **Description** |
| **Activity** |  | Use to represent a set of actions. |
| **Action** |  | Represent a task to be performed. |
| **Activity edge** |  | A directed connection between two activity nodes through which tokens may flow. |
| **Initial node** |  | Shows the beginning of an activity or set of actions. |
| **Final node** |  | Stops all controls and object flows in an activity. |
| **Object node** |  | Represents an object connected to a series of object flows. |
| **Decision node** |  | Represents a test condition that slits an incoming activity edge into opposite outgoing activity edges. |
| **Merge node** |  | Reunite different decision paths created using a decision node. |
| **Fork node** |  | Slits behaviour into parallel or concurrent flows of activities  (or actions) |
| **Join node** |  | Unites a set of parallel or concurrent flows of activities or actions. |
| **Swimlane and partition** |  | A way of grouping activities performed by the same actor in an activity diagram or to group actions in the same thread. |

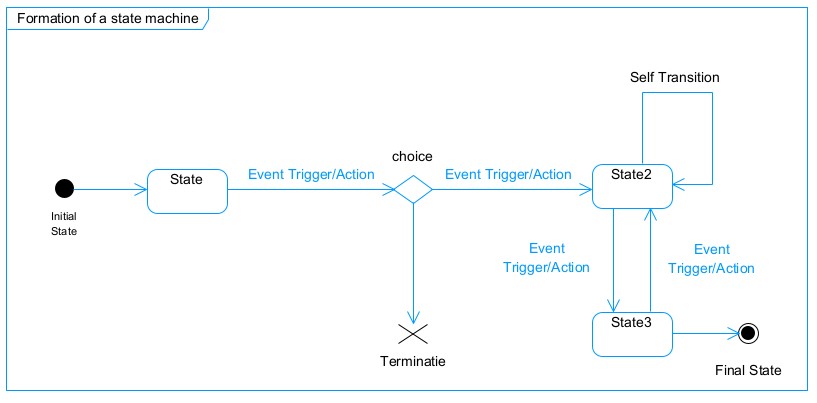
#### Activity Diagram

### State Machine Diagram

#### Definition

A state machine diagram describes the behaviour of a single object in response to a series of events in a system. Also known as the state machine diagram, it models the dynamic flow of control from the state of a particular object within a system.

#### Formalism



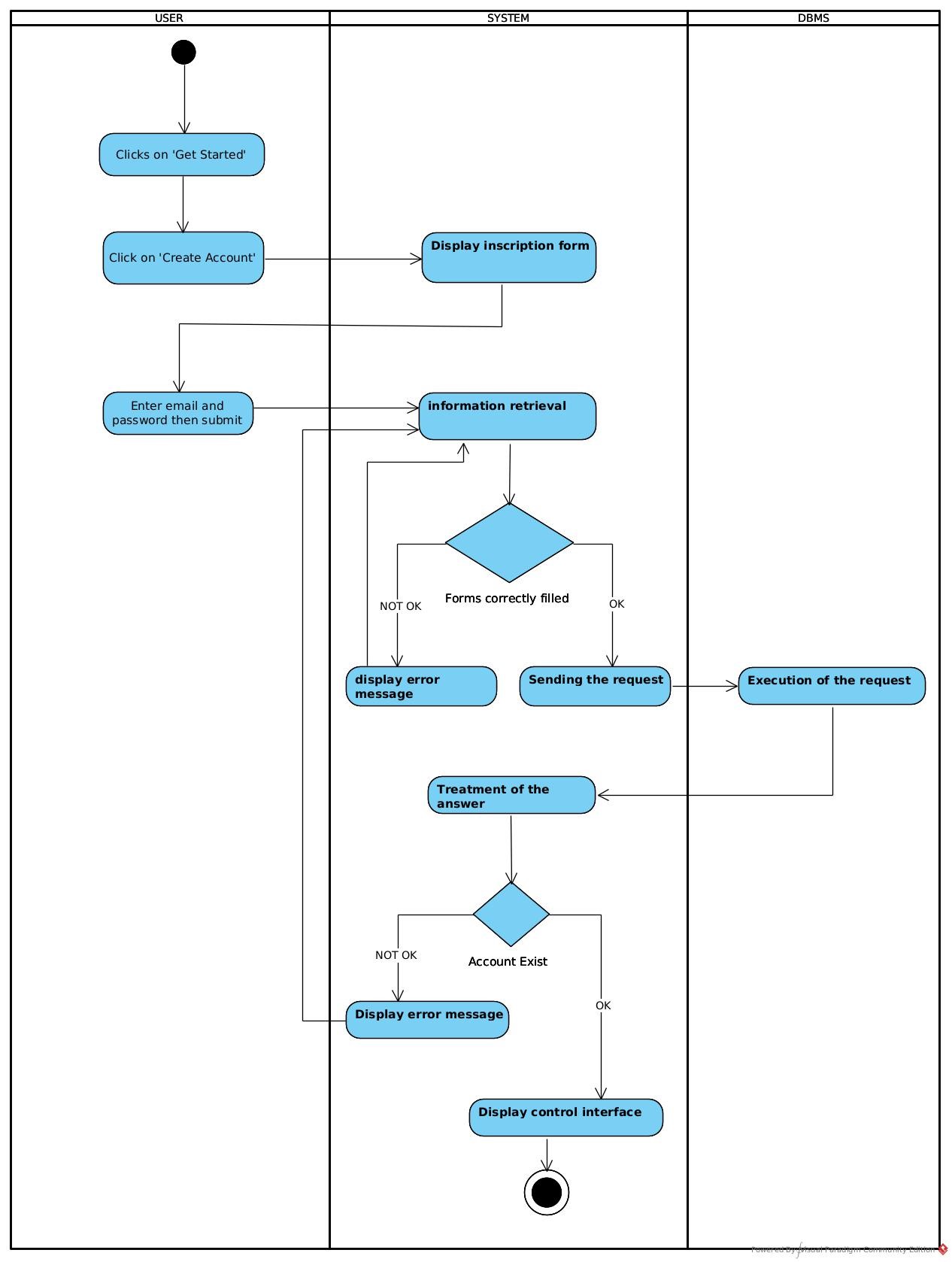
*Figure 13:Formalism of a state machine diagram*

*Table 16 State Machine Diagram Components:*

|  |  |  |
| --- | --- | --- |
| **Element** | **Diagram Relationship** | **Description** |
| **State** |  | Models a situation during which a certain invariant condition holds. |
| **First (Initial) State** |  | It represents a default vertex, that is, a source for a single transaction to the default or composite state. |
| **Final State** |  | A state specifying that the enclosing region is complete. |
| **Transition** |  | A direction relation between a source and a target vertex. |
| **Choice pseudo State** |  | A diamond symbol that indicates a dynamic condition with branched potential results |
| **Terminate** |  | Implies that the execution of a state by means of it context is terminated. |
| **Diagram Overview** |  | A placeholder for the linked states in a state machine diagram. |

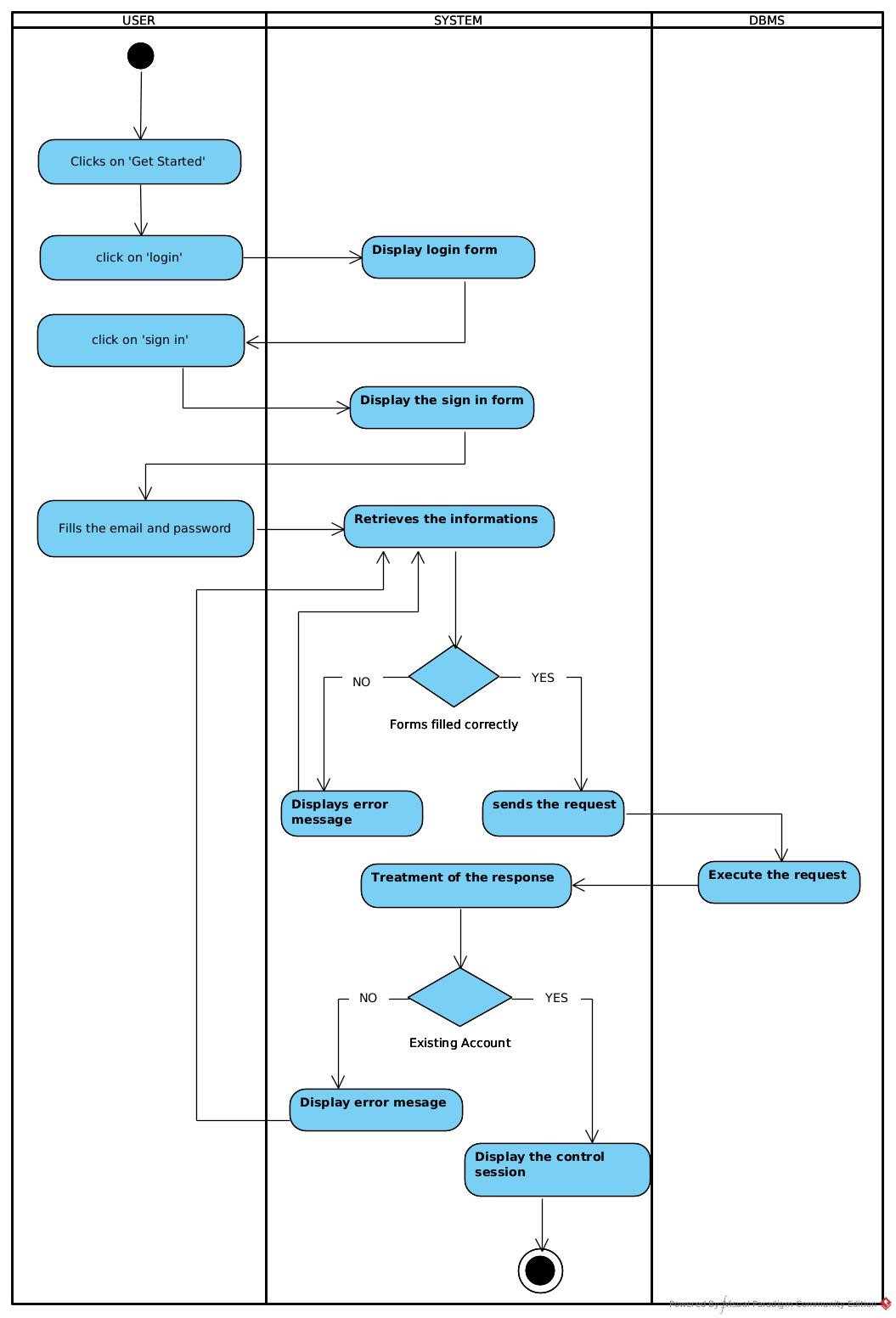
#### State Machine Diagram

1. to Create Account



*Figure 18: Activity Diagram of create Account*

1. to login



# **CONCLUSION**

In the analysis phase, we chose a software development process and modelling language, after which we presented the existing system, its limitation and our proposed solutions. We went forth explaining the functional need of our system, we saw the use case diagram which shows the relationship between the actors and use case (the action the actor can perform on the system), we saw the activity diagram which shows the workflow of our system, and lastly the State machine diagram which shows the behaviour of a single object in response to an event. We will now move to the conception phase in which we will present the other two branches of the 2TUP which are the Technical and the realization branch of our system.

# **PART FIVE: CONCEPTION PHASE**

SUMMARY

The conception phase will permit us to present in an orderly manner the components necessary for the good functioning of our software and also the architecture used for the proposed solution. It bridges the gap between the analysis phase and the realization phase.

Preview

**INTRODUCTION**

1. **TECHNICAL BRANCH**
2. **Generic Design**
3. **CAPTURE OF TECHNICAL NEEDS**
4. **RELATED UML DIAGRAMS**
5. **Class Diagram**
6. **State Transition Diagram**
7. **Package Diagram**

**CONCLUSION**

# **INTRODUCTION**

The conceptual phase will describe in details the necessary specifications, features and operations that will satisfy the functioning requirements of the proposed system as modelled in the analysis phase. This phase is meant to identify and consider essential components (hardware /or software), structure (network capabilities), processes and procedures for the system to accomplish it objectives. We will look at some diagrams such as the component diagram, package diagram, communication diagram, class diagram and the object diagram.

# TECHNICAL BRANCH

## GENERIC DESIGN

### Hardware diagram of the system

The hardware diagram simply shows how the system components of our system are deployed; it shows the positioning of each device into it right proportion.

### High Level Architectural Diagram of the Software

The high-level architecture diagram provides an overview of the entire system, identifying the main components that would be developed for the product and their interfaces. The high-level architecture diagram below illustrates this.

# CAPTURES TECHNICAL NEEDS

### Physical architecture

The design of the DBMS depends on its architecture. An n-tier architecture partitions on the whole system into related but separated n modules, which can be independently modified, altered, changed or replace. A large amount of data on web servers, personal computers (pc) and others are link with networks with the help of basic client or server architecture.

Within the scope of our project, we made use of the multi-tier architecture This architecture separate it tiers from each other based upon the user and the manipulated data in the database. It is important to note that with the multi-tier architecture, only neighboring layers can communicate. Each layer has a well-defined communication interface and the evolution of the layer is independent of the other. The multi-tier of our system is made up of:

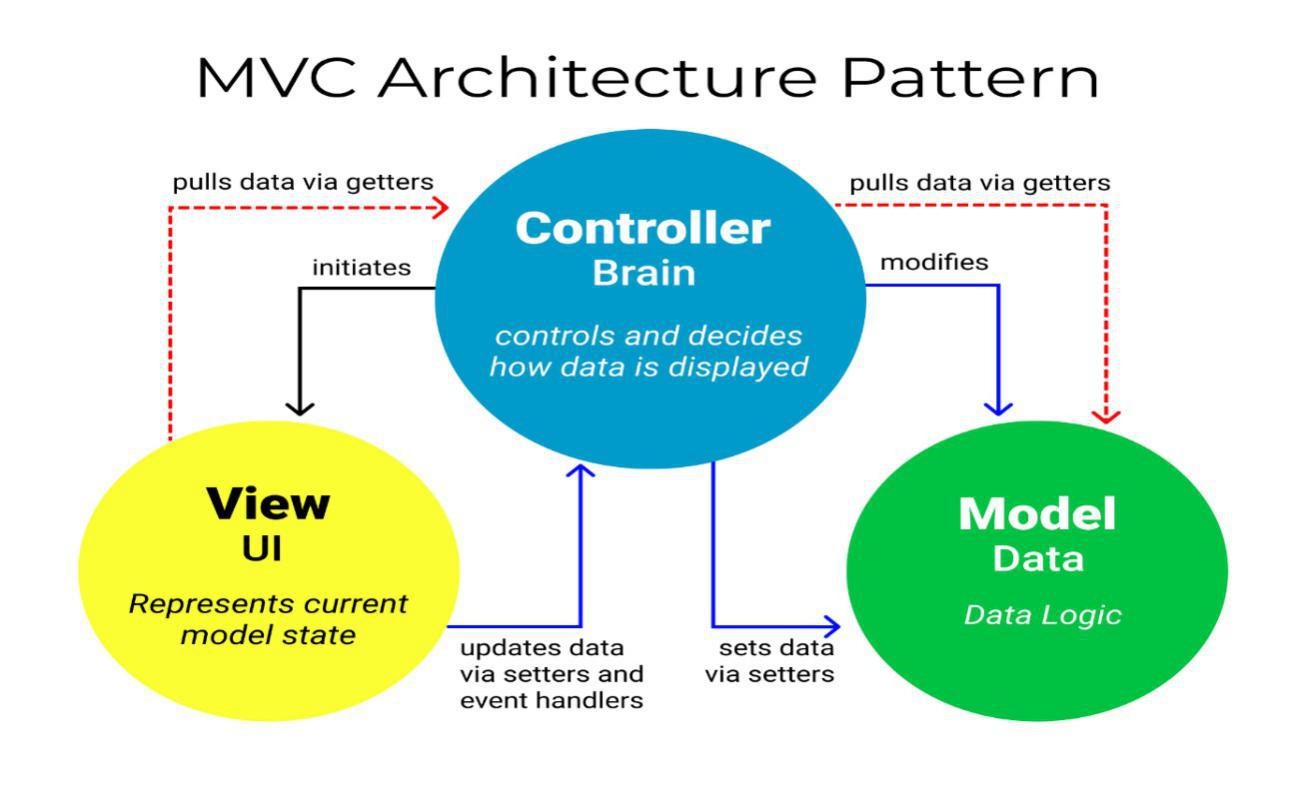
* The hardware tier, which represents our fingerprint sensor and it is connected directly to our mobile phone using a wired connection.
* The client tier, which is also known as our presentation interphase.
* Application Tier, which represents our webserver.
* The data tier, which represents our DBMS server

### Logical architecture

Model View controller or MVC as it is popularly called, is a software design patten for developing application. A model view controller patten is made up of the following three parts.

* Model: The lowest level of the patten which is responsible for maintaining data.
* View: This is responsible for displaying all or a portion of data to the user.
* Controller: It handles software codes that controls the interactions between the model and the view.

MVC is popular as it isolates the application logic from the user interface and supports separation of concerns. Here the controller receives all requests for the application then works with the model to prepare data needed by the view. The view then uses the data prepared by the controller to produce a final response. The MVC can be represented as follows.



# RELATED UML DIAGRAMS

## Class Diagram:

1. Definition:

A class diagram is a static diagram. It represents the static view of an application. class diagram is not only used for visualizing, describing and documenting different aspect of the system but also for constructing executable code of the software application. Class diagram describes the attribute and operation of a class and also constraints imposed on the system. It purpose is to model the static view of an application

* 1. Formalism:

## 

multiplicity

private

accessible

(

only

within

the

class

scop

e)

publi

c

accessible

out

of

the

class

scop

e)

0

..

0

.



class

name

Attribut

e

i



Classe\_

2

Class Diagram Elements Description

|  |  |  |
| --- | --- | --- |
| **Element** | **Representation** | **Description** |
| **class** |  | A class is an element that defines the attributes and behaviors that an object is able to generate |
| **Composition** |  | If a parent of a composite is deleted, usually, all of its parts are deleted with it. |
| **Aggregation** |  | If the parent of the aggregate is deleted, usually the children are not deleted. |
| **Dependency** |  | It existed between two classes, if one changes it may cause the change in the order, but the other way around. |
| **Generalization** |  | it a relationship between a whole thing (called superclass) and a more specific thing  (called subclass) |
| **Association** |  | It is a general type of relationship between elements, it may include cardinality, roles etc. |

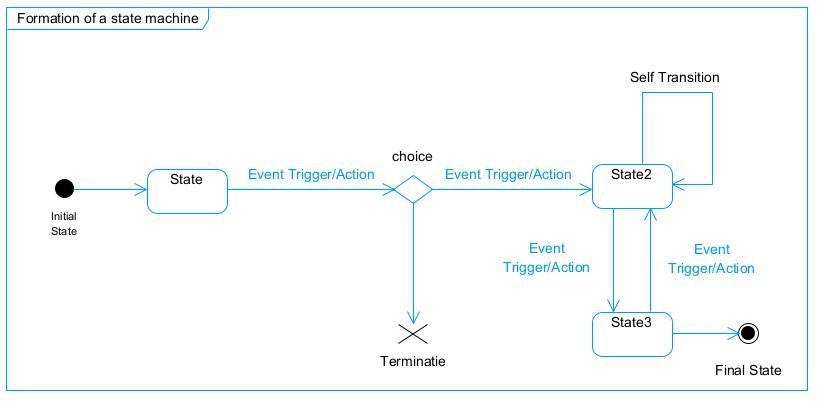
*Tabl*

## State Machine Diagram

1. Definition

A state machine diagram describes the behaviour of a single object in response to a series of events in a system. Also known as the state machine diagram, it models the dynamic flow of control from the state of a particular object within a system.

1. Formalism:



*Figure 30:Formalism of a state machine diagram*

1. State Machine Diagram Elements

|  |  |  |
| --- | --- | --- |
| **State** | tup | Models a situation during which a certain invariant condition holds. |
| **First (Initial) State** |  | It represents a default vertex, that is, a source for a single transaction to the default or composite state. |
| **Final State** |  | A state specifying that the enclosing region is complete. |
| **Transition** |  | A direction relation between a source and a target vertex. |
| **Choice pseudo State** | tup | A diamond symbol that indicates a dynamic condition with branched  potential results |
| **Terminate** |  | Implies that the execution of a state by means of it context is terminated. |
| **Diagram Overview** |  | A placeholder for the linked states in a state machine diagram. |

*Table 22:State Machine Diagram Components*

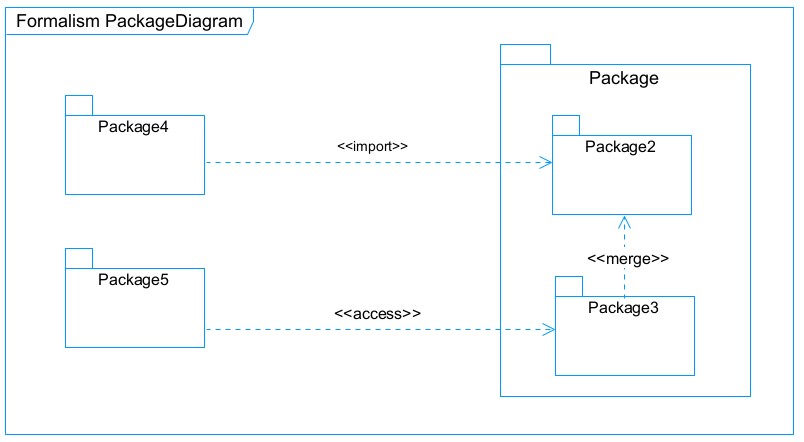
1. **DSEI start incubation state machine Diagram:**

## Package Diagram

#### Definition

This is a structural diagram used to show the organization and arrangement of various model elements in the form of packages. A package diagram is the grouping of related uml elements such as classes, diagrams or eve other packages.

#### Formalism



*Figure 21:Package Diagram Formalism*

*Figure 22:Package Diagram Components*

|  |  |  |
| --- | --- | --- |
| **NAME** | **Representation** | **Description** |
| **Package** |  | A package is a namespace use to group related elements; it is a mechanism used to group elements into a better structure in a system. |
| **Package import** |  | A relationship Indicate that, functionality has been imported from one package to another. |
| **Package access** |  | A relationship Indicates that one package requires assistance from the function of another package. |
| **Package merge** |  | It is a relationship which shows that, the functionality of two packages are combines to a single function. |

#### Mobile package Diagram

# **CONCLUSION**

In the conception phase, we set as objective to plane the different aspect of our system by showing how it will be structure and deployed within existing technical architectures. We began by taking into account the technical constraints for our system, after which we proceeded to identify the components of our system, how they are grouped together and how they should be deployed on appropriate deployment targets. We finished this phase by looking at interactions between the various aspects and actors of our system. The next phase of our report is the realization phase where we will look at aspects concerning the implementation of our system.

# **PART SIX: REALISATION PHASE**

Preamble

In this phase we will to straight forward in the implementation of our solution, we will base ourselves on the analysis and conception phases.

Content Overview

**INTRODUCTION**

1. **THE ENTITY REALTIONAL DIAGRAM**
2. **Logical architecture**
3. **Physical architecture**
4. **PRESENTATION OF DEVELOPMENT TOOLS**
5. **Choice of programming language and framework**
6. **Choice of tools used**

**CONCLUSION**

# **INTRODUCTION**

Here in the realization phase, we will concentrate on building or implementing our solution, based on the different analysis and conception that we had carried out, which will help to facilitate our work, this phase is as critical as the previous phases. We are going to look at the relationship that exist between the entities of the entity relational diagram. Furthermore, we will see the choices of technologies used for the implementation of our system.

# ARCHITECTURE OF THE APPLICATION

## Logical architecture

## Physical architecture

# TECHNOLOGICAL CHOICE

## Choice of programming language and framework

## Choice of tools used

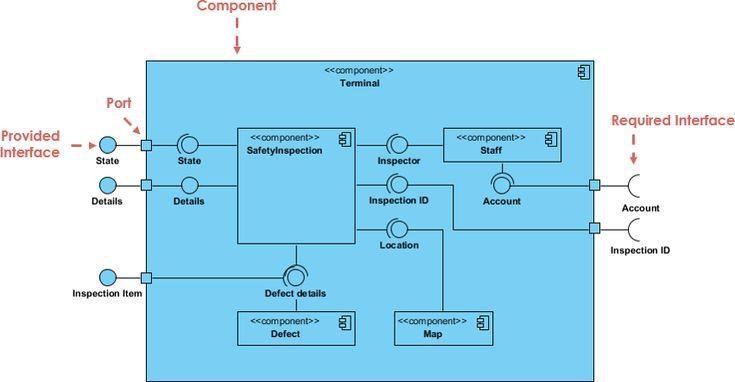
Here we will see the preliminary conception, detailed conception and documentation of the system.

C. PRELEMINARY DESIGN:

1. **Component Diagram:**
   1. Definition:

Component diagrams are used to model the physical aspect of a system. Now the question is what are this physical aspect? They are elements such as Executables, libraries, files, document etc. which resides in a node. The component diagram does notupt describe the functionality of the system but it describes the components used to make those functionalities.

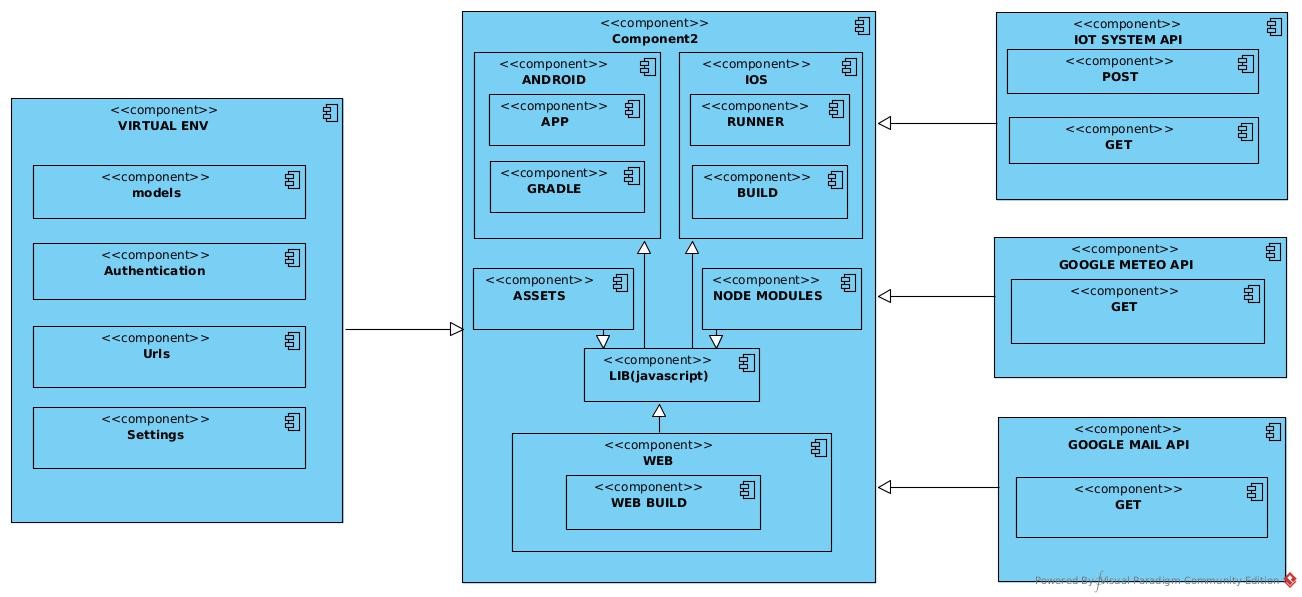
* 1. Formalism:



*Figure 34 :Component Diagram Formalism*

|  |  |  |
| --- | --- | --- |
| **NAME** | **REPRESENTATION** | **DESCRIPTION** |
| **A component** |  | A component is an abstract logical unit block of a system.it is represented as a rectangle with smaller rectangle in the upper right corner which saves as it icon for recognition. |
| **Dependency** |  | Dependency is a directed relationship which is used to show that some components are dependent on others for their correct functioning. |
| **interface** |  | An interface is a circle or a semicircle attached to a stick which looks like a lollipop. It describes groups of operations provided or required by components. |
| **port** | tup | A port (represented by a small square at the end of a required or provided interface) is used when the components delegate the interfaces to an internal class. |

*Table 24: Component Diagram Components*



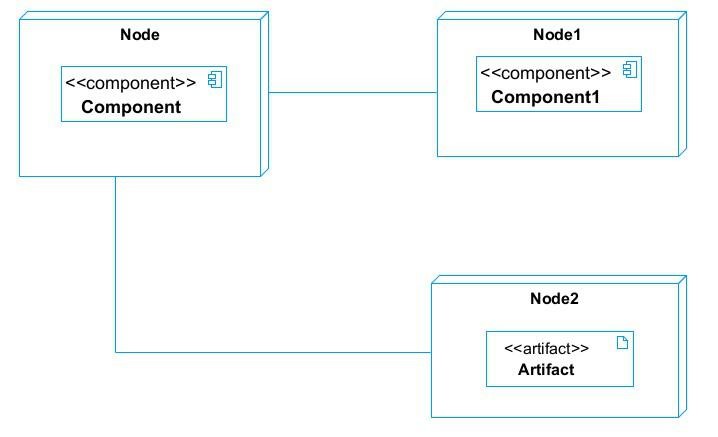
*Figure 35:System Component Diagram*

1. **Deployment diagram:**

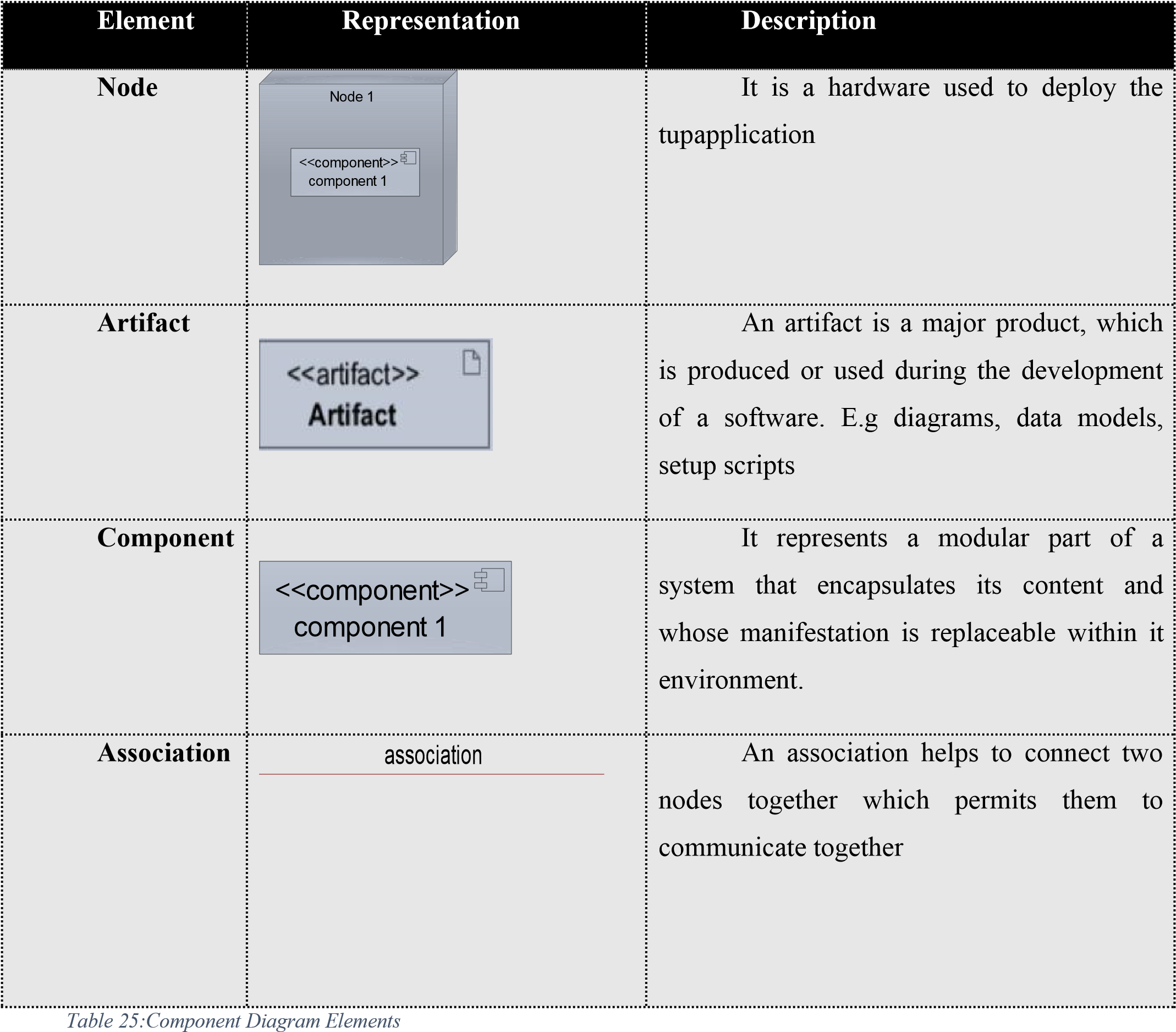
i. Definition:

consists of nodes. Nodes are nothing but physical hardware used to deploy the application Deployment diagram is a structural diagram used to visualize the topology of the physical components of a system, where the software is deployed. They consist of nodes and their relationship. It is related to the component diagram because the components are deployedtup using the deployment diagram. A deployment diagram

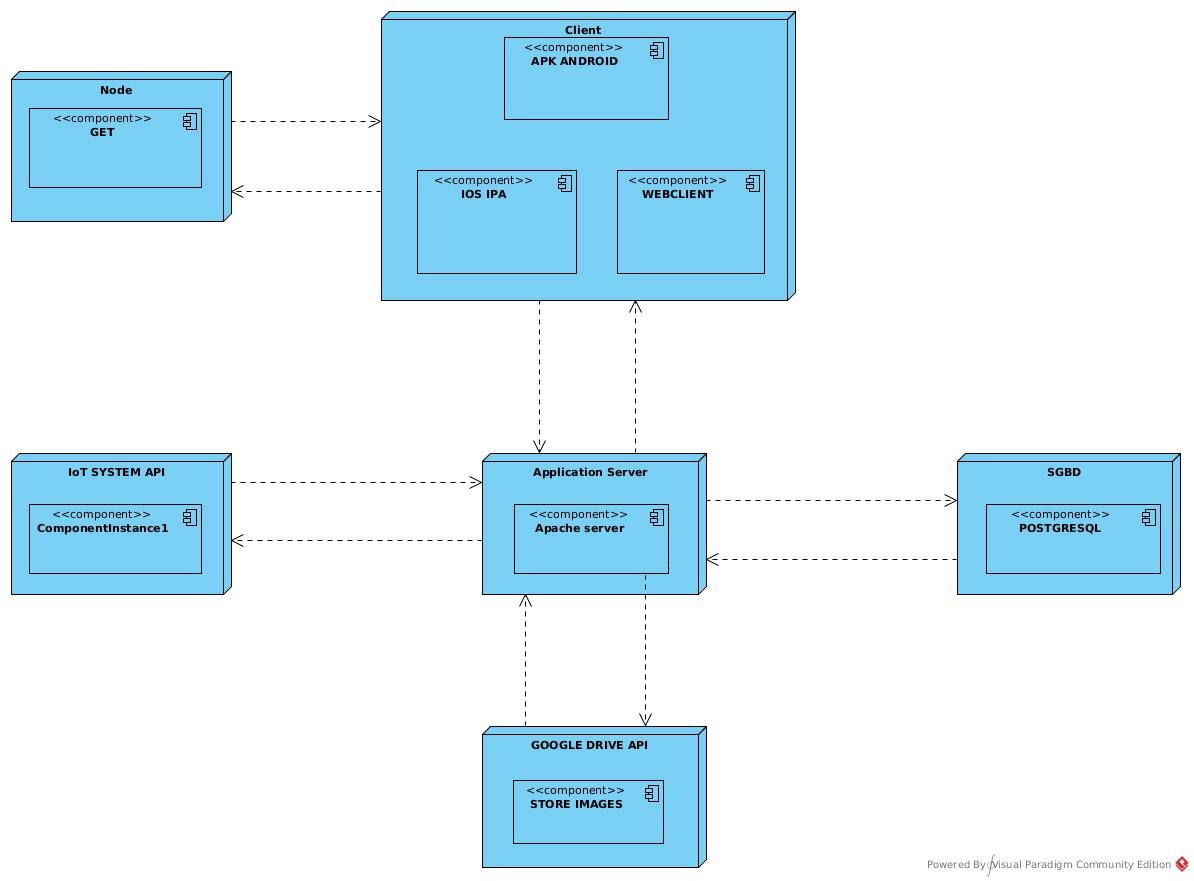
*ii.* Formalism



*Figure 36:Deployment Diagram Formalism*



b. System Deployment Diagram



*Figure 37:System Deployment Diagram*

Materials used for the development of the DSEI

|  |  |  |  |
| --- | --- | --- | --- |
| **Material** | **Quantity** | **Fonction** | **Image** |
| Servo Motor MG996r | 1 | For the rotation of the egg alveolus |  |
| Jumper Wires | ≈100 | Connector between the components of the system |  |
| LCD 2IC | 1 | To display the different information about the system to the user |  |
| ESP32 Cam | 1 | The device which is still a micro-controller will be used only for the image capturing purpose. |  |
| AM2320 | 1 | The temperature and humidity sensor of the system |  |
| Relay module | 2 | To switch on and off the fan and the bulb at a specific moment |  |
| Fans | 3 | We have 3 fans in our system, three that are on by default and the third one(most powerful) is only activated when the temperature in the system needs to be dropped down |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 12V power supply | 1 | To supply the system with power | |  |
| Voltage Transformer  and distributor | 1 |  | |  |
| 75V yellow bulb | 1 | To light the system and to create an increase in temperature and humidity | |  |
| ESP32 chip | 1 | A chip with an incorporated micro-controller to treat all the information received from the sensors and detectors according to the algorithm uploaded in it. | |  |
| Oraimo USB cable | 1 | For Uploading The code into | our |  |
|  |  | esp32 chip. And Exchange information and data. | of |  |
| White Plastic Cup | 2 | To keep the Keep water in  system to create humidity | the |  |
| Water Level Sensor | 2 | To measure the level of water in the cups | |  |
| MQ-135 Gas sensor | 1 | To detect if there any ammonia gas evolved from a rotten egg in the system | |  |

|  |  |  |  |
| --- | --- | --- | --- |
| PIR (Passive Infrared)  Sensor | 1 | To detect any movement in the system. This means if a chick hash.  He might do some movement. |  |
| Bread Board | 2 | This was used to make quick electrical connections between the different components |  |
| Dell LATITUDE  E7540 | 1 | The computer was used for the programming of the micro controller, the analysis, research, the design of the system architecture. This was the main component in this project. |  |
| HP computer | 1 | The computer was used for the development of the mobile  application and web application |  |
| iPhone 11 pro max | 1 | Testing the mobile app functionalities and debugging. |  |

Softwares used for the development of the DSEI

|  |  |  |  |
| --- | --- | --- | --- |
| Software | **Version** | **Role** | **Image** |
| OS  WINDOWS  10 | 18 362 | The operating system required to run the software is the one on which we have worked. |  |
| Sybase  PowerAMC | 16,5 | The software engineering workshop used for solution modeling. We utilized this tool for modeling the various diagrams of our system. |  |
| Visuel studio code | 1.70.0.8 | Éditeur de texte utilisé pour saisir les lignes de code qui seront interprétées par le navigateur. |  |
| Arduino IDE | 1.8.19 | The text editor used to input the lines of code that will be interpreted by the browser. |  |
| Postman | 9.4 | It is a platform that simplifies every step of the API lifecycle and  streamlines collaboration, making it easier and faster to create better APIs. |  |
| Expo go | 2.25.2 | It is a framework and platform for  React applications, designed to compile React mobile applications directly on the phone. |  |
|  |  |  |  |
| Github | 3.0.6 | It is a website and cloud service that helps developers store and manage their code. |  |
| Framework react native | 0.69.0 | React Native is an English framework built on JavaScript for creating Android and iOS applications. |  |
| EDRAW  MAX | 11.5.0 | the software you're referring to is a 2D technical and business diagram  creation tool that helps create  organizational charts, flowcharts, mind maps, network diagrams, floor plans, workflow diagrams, business charts, and engineering diagrams. |  |
| **PostgreSQL** |  | Database Management System |  |

*Table 26: Softwares used for the development of the DSEI*

#### Technologies used for the developpement of the

|  |  |  |
| --- | --- | --- |
| **Image** | **Name** | **Usage** |
|  | Django – Framework | Used for the development of the application core/kernel. |
|  | JavaScript | The programming language used for front-end processing is JavaScript. |
|  | C++ | Arduino programming |
|  | Framework React Native | Front-end development |
|  | Python3 | Backend development |

*Table 27:Technologies used for the development of the DSEI*

# **CONCLUSION**

In our realization phase, we implemented our application. In other to accomplish this phase, we made used of our analysis and conception phase. The Entity-Relational diagram. helped us to establish our database. We then presented the development tools we used for the development or realization of our system, we had material resource (visual studio code) and technology stack (React js, react-native js, Node js/express js). We will move to the user guide phase where we shall elaborate on how to install and use

# **PART SEVEN: FUCTIONALITY TESTING**

SUMMARY

The purpose of the user guide is to provide users of our platform with step-by-step instructions on how to install and use the system.

Preview

**INTRODUCTION**

**CONCLUSION**

# 

# **INTRODUCTION**

This is the final phase of our report. In this phase we will walk through the requirements for our system, the necessary installation process, accessing our system and it futures, all this in a step by step manner to facilitate the setting up of the platform for the first time users. The steps of different processes will be accompanied by images. After we will showcase our application by viewing the different screens and some brief explanation.

# **PART EIGHT: INSTALLATION AND USER GUIDE**

Preamble

The purpose of the user guide is to provide users of our platform with step-by-step instructions on how to install and use the system.

Content overview

**INTRODUCTION**

1. **INSTALLATION**
2. **USER GUIDE**

**CONCLUSION**

# **INTRODUCTION**

This is the final phase of our report. In this phase we will walk through the requirements for our system, the necessary installation process, accessing our system and it futures, all this in a step by step manner to facilitate the setting up of the platform for the first time users. The steps of different processes will be accompanied by images. After we will showcase our application by viewing the different screens and some brief explanation.

# INSTALLATION

# USER GUIDE

# **Conclusion**

# 

# **GENERAL CONCLUSION**

# 

# **ANNEXE**

# 

# **BIBLIOGRAHY**

# 

# **WEBOGRAPHY**

# 

# **TABLE OF CONTENT**

[**DEDICATION** 1](#_Toc173689683)

[**ACKNOWLEDGMENTS** 2](#_Toc173689684)

[**GLOSSARY** 5](#_Toc173689685)

[**ABSTRACT** 7](#_Toc173689686)

[**RESUME** 7](#_Toc173689687)

[**GENERAL INTRODUCTION** 7](#_Toc173689688)

[**PART I: INSERTION PHASE** 8](file:///I:\Sauvegarde%20Soutenance\ID%20app\Rapport%20de%20stage\Internship%20report.docx#_Toc173689689)

[**INTRODUCTION** 10](#_Toc173689690)

[I. WELCOME AND INTERGRATION 10](#_Toc173689691)

[1. Welcome 10](#_Toc173689692)

[2. Integration 10](#_Toc173689693)

[II. GENERAL PRESENTATION OF WELLDONE PLANET 11](#_Toc173689694)

[1. History and Missions 11](#_Toc173689695)

[2. Presentation of the structure 12](#_Toc173689696)

[b. Material Safety Data Sheet 12](#_Toc173689697)

[C. Resources 13](#_Toc173689698)

[3. Visions, Objectives of Welldone Planet 16](#_Toc173689699)

[b. Goals 17](#_Toc173689700)

[c. Attributions 19](#_Toc173689701)

[**CONCLUSION** 20](#_Toc173689702)

[**PART TWO: SPECIFICATION BOOK** 21](file:///I:\Sauvegarde%20Soutenance\ID%20app\Rapport%20de%20stage\Internship%20report.docx#_Toc173689703)

[**INTRODUCTION** 23](#_Toc173689704)

[I. CONTEXT AND JUSTIFICATION 24](#_Toc173689705)

[II. OBJECTIVES OF THE PROJECT 25](#_Toc173689706)

[A. General Objective 25](#_Toc173689707)

[B. Specific Objectives 25](#_Toc173689708)

[III. EXPRESSION OF NEEDS 25](#_Toc173689709)

[A. Functional needs 25](#_Toc173689710)

[B. Non-functional Needs 25](#_Toc173689711)

[IV. ESTIMATED COST OF THE PROJECT 26](#_Toc173689712)

[A. Hardware Resources 26](#_Toc173689713)

[B. Software Resources 27](#_Toc173689714)

[C. Human Resources 28](#_Toc173689715)

[D. Total Project Estimated Cost 29](#_Toc173689716)

[V. PROJECT PLANING 29](#_Toc173689717)

[A. Chronogram of activities 29](#_Toc173689718)

[B. Gantt Chart 30](#_Toc173689719)

[VI. LIST OF PARTICIPANTS AND DELIVERABLES 31](#_Toc173689720)

[A. List of participants 31](#_Toc173689721)

[B. Deliverable 31](#_Toc173689722)

[**CONCLUSION** 32](#_Toc173689723)

[**PART THREE: ANALYSIS PHASE** 33](file:///I:\Sauvegarde%20Soutenance\ID%20app\Rapport%20de%20stage\Internship%20report.docx#_Toc173689724)

[**INTRODUCTION** 36](#_Toc173689725)

[I. DESCRIPTION OF THE EXISTING SYSTEM 36](#_Toc173689726)

[II. PRESENTATION OF THE ANALYSIS 36](#_Toc173689727)

[A. MODELING WITH UML 2.5 38](#_Toc173689728)

[a) Structural Diagrams : 38](#_Toc173689729)

[b) Behavioral Diagrams : 39](#_Toc173689730)

[c) UML 2.5 diagrams overviews 40](#_Toc173689731)

[B. THE PROCESS OF DEVELOPMENT OF A SOFTWARE 41](#_Toc173689732)

[a) A Unified Process 41](#_Toc173689733)

[b) The Two Track Unified Process (2TUP) 41](#_Toc173689734)

[III. JUSTIFICATION OF THE ANALYSIS 42](#_Toc173689735)

[a) The left branch (functional branch) 43](#_Toc173689736)

[b) The right branch (Technical branch) 43](#_Toc173689737)

[c) The middle branch (Realization or implementation branch) 43](#_Toc173689738)

[IV. MODELING OF THE SYSTEM 44](#_Toc173689739)

[A. CAPTURE OF FUNCTIONAL NEEDS 44](#_Toc173689740)

[1. *Use Case Diagram* 44](#_Toc173689741)

[2. ACTIVITY DIAGRAM 48](#_Toc173689742)

[3. State Machine Diagram 51](#_Toc173689743)

[**CONCLUSION** 53](#_Toc173689744)

[**PART FOUR: CONCEPTION PHASE** 54](file:///I:\Sauvegarde%20Soutenance\ID%20app\Rapport%20de%20stage\Internship%20report.docx#_Toc173689745)

[**INTRODUCTION** 56](#_Toc173689746)

[I. TECHNICAL BRANCH 56](#_Toc173689747)

[A. GENERIC DESIGN 56](#_Toc173689748)

[a) Hardware diagram of the system 56](#_Toc173689749)

[High Level Architectural Diagram of the Software 56](#_Toc173689750)

[B. CAPTURES TECHNICAL NEEDS 57](#_Toc173689751)

[a) Physical architecture 57](#_Toc173689752)

[Logical architecture 57](#_Toc173689753)

[II. IMPLEMENTATION BRANCH 58](#_Toc173689754)

[A. PRELIMINARY DESIGN 58](#_Toc173689755)

[a) Component diagram 58](#_Toc173689756)

[Package Diagram 61](#_Toc173689757)

[Deployment Diagram 63](#_Toc173689758)

[B. DETAILED DESIGN 65](#_Toc173689759)

[**CONCLUSION** 66](#_Toc173689760)

[**PART FIVE: REALISATION PHASE** 67](file:///I:\Sauvegarde%20Soutenance\ID%20app\Rapport%20de%20stage\Internship%20report.docx#_Toc173689761)

[**INTRODUCTION** 69](#_Toc173689762)

[I. ENTITY RELATIONSHIP DIAGRAM 70](#_Toc173689763)

[II. PRESENTATION OF DEVELOPMENT TOOLS 70](#_Toc173689764)

[A. MATERIAL RESOURCE 70](#_Toc173689765)

[B. TECHNOLOGICAL STACK 70](#_Toc173689766)

[**CONCLUSION** 70](#_Toc173689767)

[**PART SIX: USER GUIDE** 71](file:///I:\Sauvegarde%20Soutenance\ID%20app\Rapport%20de%20stage\Internship%20report.docx#_Toc173689768)

[**INTRODUCTION** 73](#_Toc173689769)

[I. INSTALLATION OF THE APPLICATION 74](#_Toc173689770)

[II. SHOWCASES 74](#_Toc173689771)

[**GENERAL CONCLUSION** 74](#_Toc173689772)

[**ANNEXE** 74](#_Toc173689773)

[**BIBLIOGRAHY** 74](#_Toc173689774)

[**WEBOGRAPHY** 74](#_Toc173689775)

[**TABLE OF CONTENT** 74](#_Toc173689776)