International Journal of Innovation and Scientific Research
ISSN 2351-8014 Vol. 76 No. 1 Dec. 2024, pp. 23-38
© 2024 Innovative Space of Scientific Research Journals
http://www.ijisr.issr-journals.org/

Design and implementation of a notification and geo-localisation system for the mobility of students and staff at Kimpa Vita-Uíge University

Kundakama Mpanda[1], Belarmino Garcia António[2], Bilolo António Vemba[3], Nzakiese Mbongo[4], Mavambu Diankatu[5], Bazangika Mfumuanene Victoire[6], and Nsuadi Kiese Glodi[7]

[1]Department of Computer Engineering, Polytechnic Institute of the Kimpa Vita University, Angola

[2]Department of Computer Engineering, Polytechnic Institute of the Kimpa Vita University, Angola

[3]Department of Computer Engineering, Polytechnic Institute of the Kimpa Vita University, Angola

[4]Department of Computer Engineering, University of the Luanda, Angola

[5]Department of Computer Engineering, Polytechnic Institute of the Kimpa Vita University, Angola

[6]Department of Electrotechnical, Institute of Applied Techniques in Gombe Matadi, RD Congo

[7]Department of Mathematics -Informatics, Pedagogical Institute of Mbanza-Ngungu, RD Congo

ABSTRACT: The objective of this work was to analyse the importance of the feasibility of using a geolocation alert system. With the evolution of technology and its growing importance in people's lives, existing vehicle tracking systems work with the aim of providing vehicle monitoring along a route. To meet the needs presented by the students, an electronic system was developed that will be installed on the bus, which will have the ability to send its geographical coordinates, based on a GPS system, to the GÊNESIS App and store them, thus enabling enrolled students and staff at Kimpa Vita University to locate a particular bus. The system presented in this research is designed to be a simple and easy-to-use tool, with a sufficient degree of precision to ensure that the results meet the specifications of the requirements.

KEYWORDS: smartphones, Arduino, RUP method, global positioning system.

1 INTRODUCTION

In recent decades, there have been major technological advances in the areas of computing and automation. With their diverse functionalities when networked, these technologies have created more and more integration in everyday activities, arousing the desire to use these resources as a way of simplifying operations and enabling autonomous and totally remote control [27].

With technology following a path in which systems are increasingly automatic and independent, it is necessary to utilise these technologies in functions that are considered simple, but which can have a significant impact on everyday activities.

Localisation systems are an example of technologies that have aroused great interest in both the academic and industrial fields. They have various applications, such as locating vehicles, fleets and people, and use self-localisation resources such as GPS [23].

As far as the geolocation system proposed in this research is concerned, it is intended for use in an automotive workshop, more precisely in locating vehicles. This location will be stored in a database, which will allow students to know the location of the buses.

Therefore, the creation of a geolocation alert system aims to provide students and staff with information on the positioning of buses moving along the UNIKIVI Campus - City Centre route on working days via SMS.

This solution will significantly improve the efficiency of the transport service provided by Kimpa Vita University (UNIKIVI) in Uíge. We hope that it will also be a mechanism to help drivers fulfil their routes and reduce student stress.

In addition, the geolocation alert system can also provide valuable data for the University's transport management, allowing it to monitor the most used routes and make more assertive decisions regarding route planning and resource allocation. With this, we believe that the implementation of this system could result in a more comfortable and efficient transport experience for the entire UNIKIVI academic community.

2 Literature Review

2.1 Brief History

Kimpa Vita University, abbreviated UNIKIVI, is a public institution with legal personality and classified as a public establishment. Its mission is to train higher education professionals in various areas of knowledge, as well as to carry out research and provide services to the community. UNIKIVI enjoys scientific, pedagogical, cultural, administrative, property and financial autonomy, as provided for by law.

The provincial government of Uíge has made an area of 1,500 hectares available to UNIKIV, located between the village of Condo Benza and Casseche, in the municipality of Uíge. However, this location is in a suburban area 7 km from the city centre of Uíge and is being built in phases.

As far as the transport system for students and staff is concerned, UNIKIVI has no system for alerting and geolocating buses to help students, thus causing students to be unaware of how the buses work and where they are. As a result, they spend a lot of time at bus stops in the sun and sometimes getting wet in the rain.

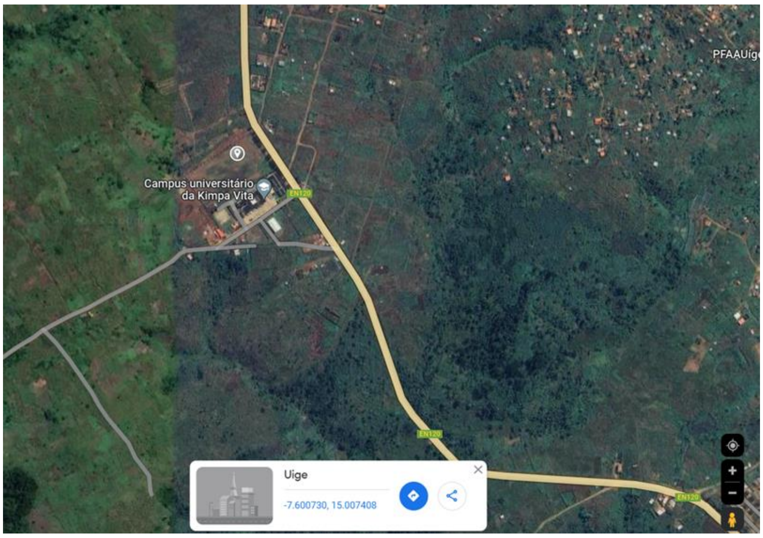


Fig. 1. Location of the Kimpa Vita university campus

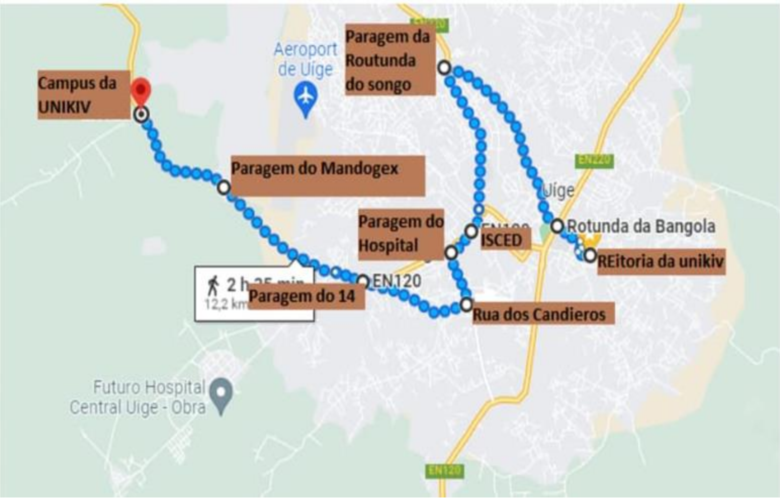


Fig. 2. Bus routes of Kimpa Vita University - Uíge City

UNIKIVI's rectory is located in the Popular neighbourhood, at Rua Henriques Freitas nº 1, in the same city. Due to the remoteness of the campus, the academic community faces considerable challenges related to the public transport that serves this area, specifically with regard to the buses provided by the rectory to cover these routes.

3 Methods And Tools

3.1 System Development

The RUP (Rational Unified Process) is a software development process that provides a set of guidelines and best practices for creating software systems. It was initially developed by the Rational Software Corporation (now part of IBM) in the 1990s and is based on software engineering principles.

The main objective of RUP is to meet the needs of users by guaranteeing high-quality software production that fulfils a predictable schedule and budget. Thus, RUP shows how the system will be built in the implementation phase, generating the project model and, optionally, the analysis model which is used to guarantee robustness.

The RUP perfectly defines who is responsible for each area, how the different phases should be done and when they should be carried out, describing all the development goals specifically so that they can be achieved.

The process is made up of four phases: conception, elaboration, construction and transition and each is related to a progress [14].

3.2 Tools

- The modelling language used is:

- The UML

- Diagram construction is

- Astah Community

- The main functions of the app genesis system:

- Logging in
- Registering a stop
- Register students
- Register Driver
- Register Buses
- List Students

- List Stops
- List Drivers
- List Buses
- Consult Students
- Consult Stops
- Consult Drivers;
- Consult Buses;
- Update (Driver, Student, Buses, Stops);
- Delete (Driver, Student, Buses, Stops);
- Locate Bus

- The programming language used to run the system is:

- JAVA programming language and INVETOR app

- Interfaces are programmed in

- JAVA Programming Language

- The database is located in

- MySQL database management system

- The materials that will be used to build the app genesis system are:

- Arduino board
- SIM808 GPS module,
- GSM/GPRS module
- Bluetooth module

3.3 Design Phase

In the initial system design, a discussion is held about the problem, defining the scope of the project, estimating the resources needed to carry out the project, among other things. It is during this phase that the project plan, initial use case and project glossary, among others, are presented [26].

3.4 Estimated Project Time

Table 1. Below shows the data needed to calculate the average project execution time

Id	Activity	Duration of the activity (in days)		
		Expected time	**Minimum time**	**Maximum time**
(1)	Conception	20	10	40
(2)	Elaboration	15	20	30
(3)	Construction	50	30	60
(4)	Transition	15	10	30
Total		100	70	160

The average project execution time is determined using the following formula:

d (i) ≃ (Expected time +Maximum time+ 4 * Minimum time) / 6

Substituting the values, we get:

d (1) ≃ (20 + 40 + 4 * 10) / 6= 100 / 6 ≃ 17
d (2) ≃ (15 + 30 + 4 * 20) / 6= 125 / 6 ≃ 21
d (3) ≃ (50 + 60 + 4 * 30) / 6= 230 / 6 ≃ 38
d (4) ≃ (15 + 30 + 4 * 10) / 6= 85 / 6 ≃ 14

So let’s calculate the total duration of the project using the formula:

$$DT = \sum Average\ times = 17 + 21 + 38 + 14 = 90$$

The project will therefore last 90 days, corresponding to 2 months and 28 days.

3.5 System Analysis And Specification

The crucial phase of the project is analysing requirements and documenting the system, as this is where the specification and essential elements of the project are presented. In the context of developing the "App GÊNESIS" application, various requirements were identified. Initially, the functional requirements that were indispensable for designing the proposed solution were identified.

3.6 Functional Requirements

Functional requirements describe what the system should do, they refer to the system’s functionalities. The main functionalities are:

Table 2. Functional Requirements

Nº	References	Description
1	RF01	Login
2	RF02	Register Stop
3	RF03	Register Students
4	RF05	Register Driver
5	RF05	Register Buses
6	RF06	List Students
7	RF07	List Stops
8	RF08	List Drivers
9	RF09	List Buses
10	RF10	Consult Students
11	RF11	Consult Stops
12	RF12	Consult Drivers;
13	RF13	Consult Buses;
14	RF14	Update (Driver, Student, Buses, Stops);
15	RF15	Delete (Driver, Student, Buses, Stops);
16	RF16	Locate Bus

3.7 Non-Functional Requirements

Non-functional requirements include performance requirements and other product quality attributes.

Table 3. Non-functional requirements

Nº	References	Description
1	RNF01	Computer with Windows 8,10 operating system Hardware Features: RAM 4GB or higher, Processor: Intel (R) Core (TM) i3-3110M CPU, 2.40GHZ or higher, HD 500GB
2	RNF02	NetBeans Development Platform with Java Language and App Inventor
3	RNF03	Mysql Database Manager

3.7.1 Use Case Diagram

The Use Case Diagram is a graphical tool used in software engineering to model and describe the interaction between users (or actors) and a system. It is generally used at the beginning of the software development process to help identify the functional requirements of the system and the interactions that occur between its components [28].

The main purpose of the use case diagram is to provide an overview of the system and help understand how it interacts with the external environment. This can help ensure that system requirements are identified and documented clearly and completely.

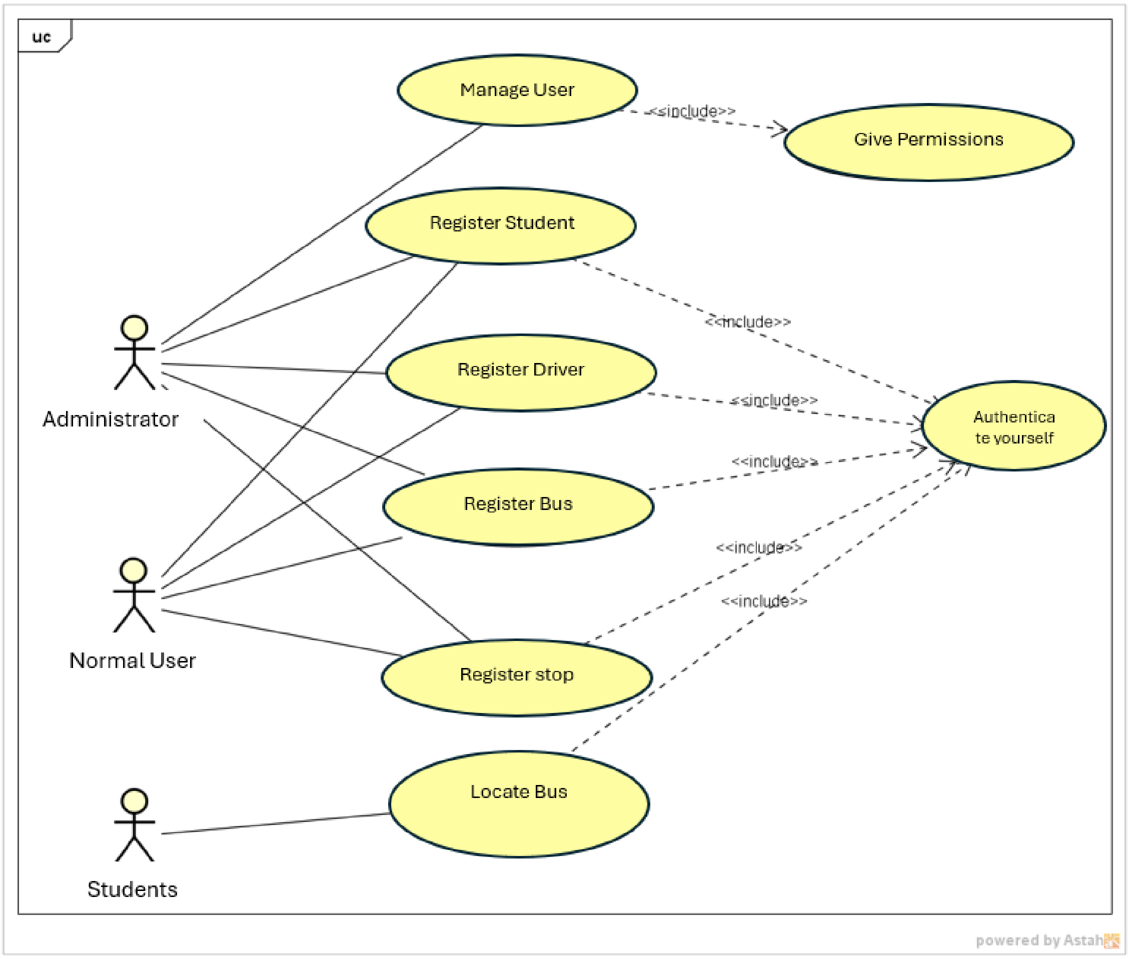


Fig. 3. Use case diagram: "give bus positioning"

3.7.2 Project Preparation Phase

Technological (referring to the capacity of the tools available), skills (of the project members) and political. This is the most critical phase of all, because at the end of this phase the engineering is considered complete and the costs of modifying the system increase as the project progresses.

3.7.3 Class Diagram

The main purpose of the class diagram is to visualise the classes that will make up the system, their attributes, methods and how these classes relate to each other [28].

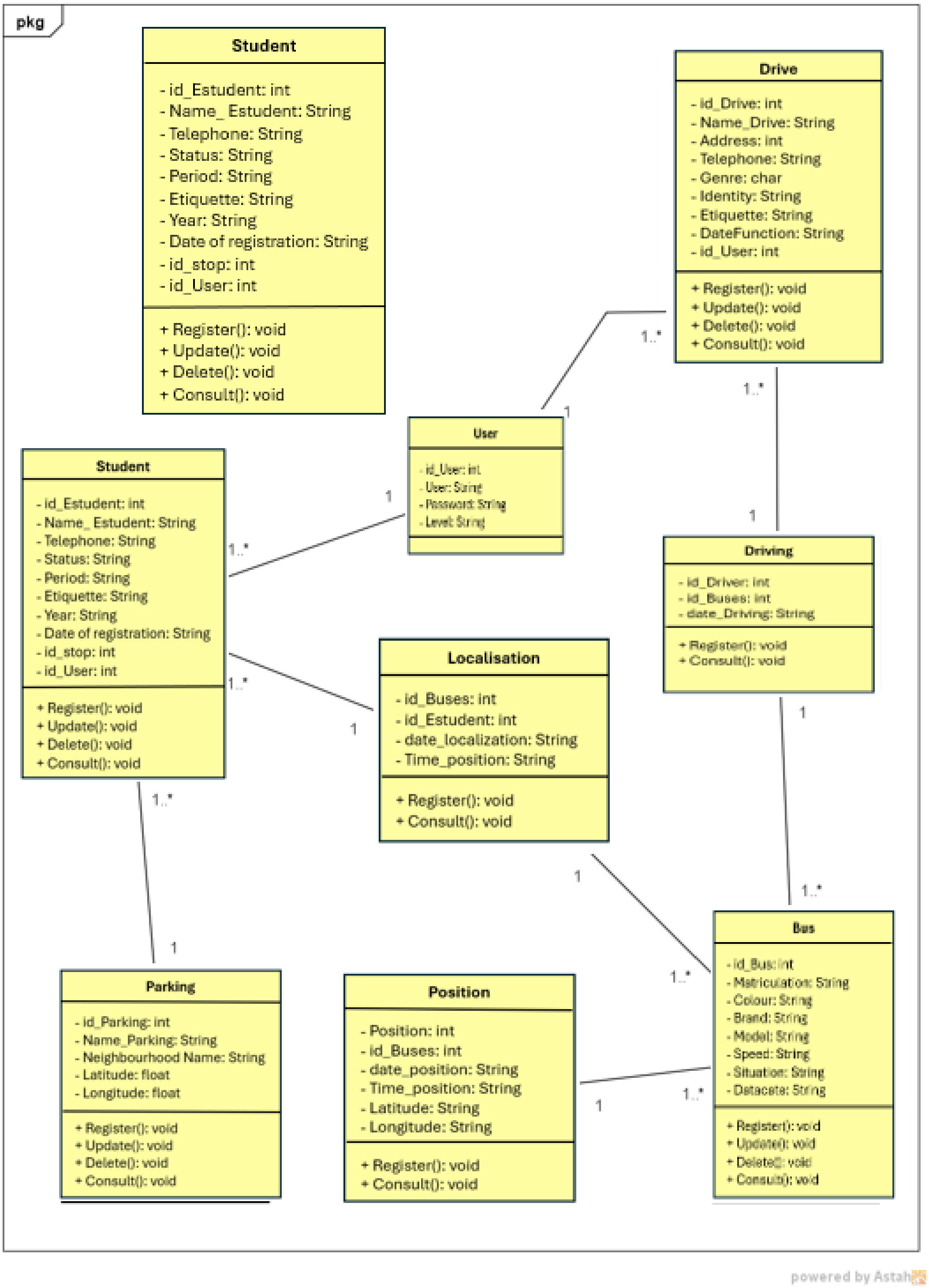


Fig. 4. Class diagram

3.7.4 Sequence Diagram

This type of diagram shows an interaction made up of a set of objects and their relationships, including the messages that can be sent between them. The following figure shows the sequence diagram designed.

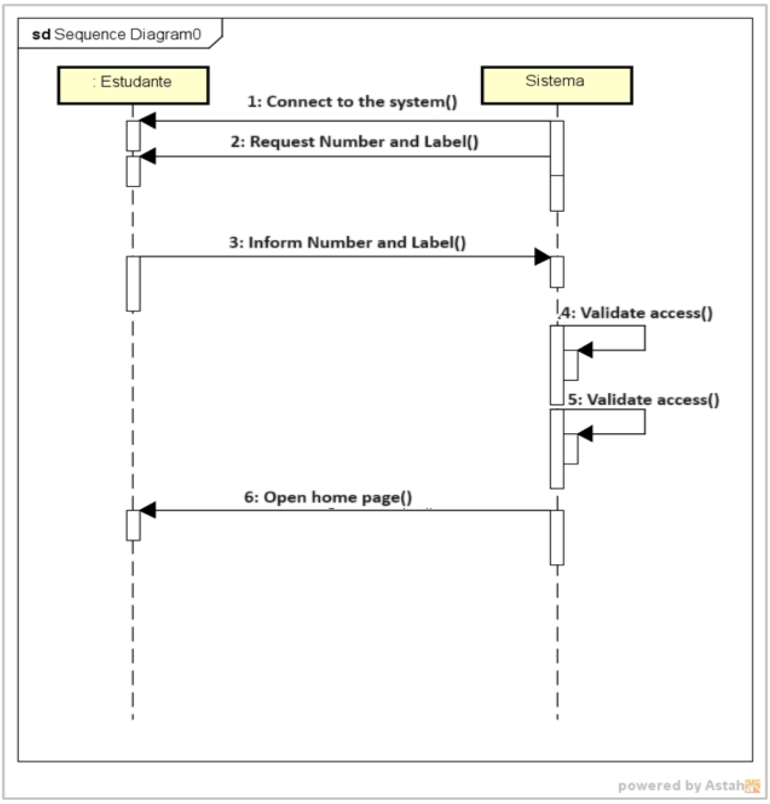
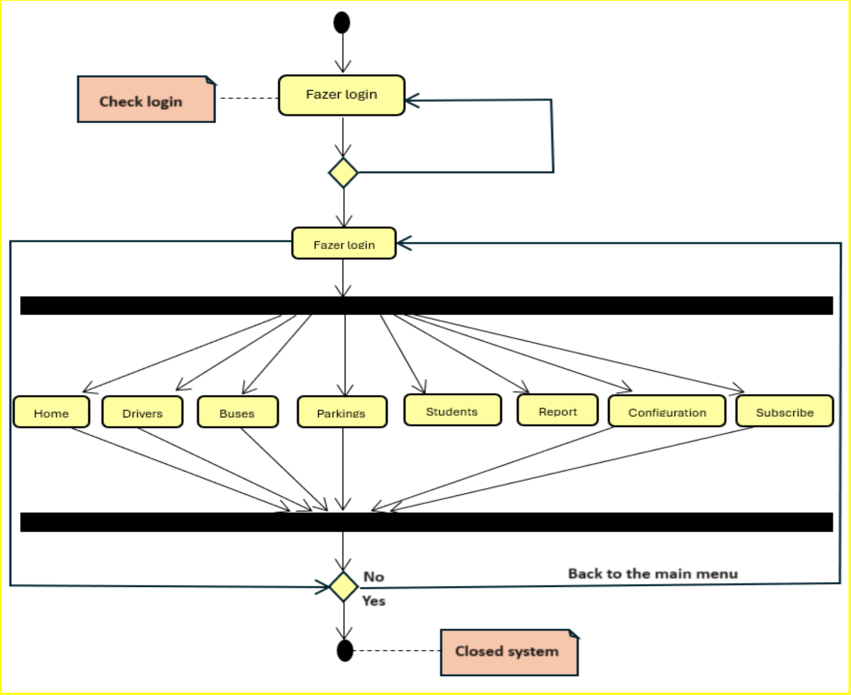


Fig. 5. Sequence diagram "Give bus positioning"

3.7.5 Activity Diagrams

The activity diagram is a simple modelling element that is most effective for describing workflows in an organisation or for detailing the operations of a class, including behaviours that involve parallel processing [22].



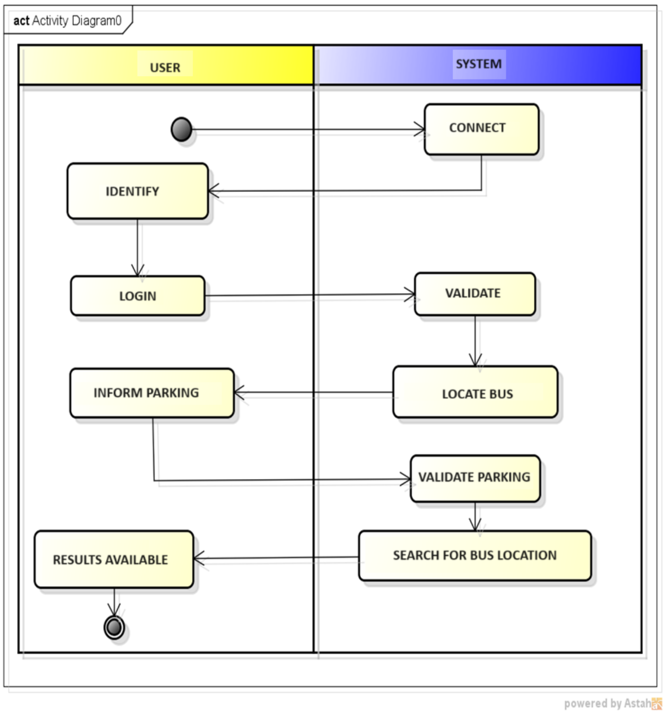


Fig. 6. Activity diagram

3.7.6 CONSTRUCTION PHASE

This phase includes the modelling phase and the development phase itself, where the system is actually programmed. The modelling phase should use some notation defined by the Unified Modelling Language (UML). This is the phase in which the application is created, including the creation of the interfaces, the codes and the respective tests [1].

3.7.7 DATA DICTIONARY

The data dictionary describes the structure of the data used in the system. As an illustration, we have the following tables:

Table 4. Data dictionaries for the students table

Field	Description	Data Type	Note
id_ Student	Student's unique identifier	Int	Not Null
Student Name	Student's name	Text (40)	Not Null
Phone number	Student telephone number	Text (40)	Not Null
Status	Student Status	Text (10)	Not Null
Period	Period the student is attending	Text (15)	Not Null
Etiquette	Student identifier when sending SMS to the system	Text (7)	Not Null
Year	Year the student is attending	Text (10)	Not Null
Registration date	Student registration date	Text (40)	Not Null
stop_id	Unique stop identifier	Int	Not Null

Table 5. Data dictionary of the drivers table

Field	Description	Data Type	Note
id_Driver	Driver's unique identifier	Int	Not Null
name_Driver	Driver's name	Text (40)	Not Null
Address	Driver's address	Text (40)	Not Null
Phone number	Driver's telephone number	Text (10)	Not Null
Genre	Driver's gender	Text (15)	Not Null
Identity	Driver's BI	Text (7)	Not Null
Etiquette	Driver's password in the day's signature	Text (10)	Not Null
Job date	Driver registration date	Text (40)	Not Null

3.7.8 TRANSITION PHASE

This is the stage in which the genesis system was tested and validated with the client, in this case the transport department of the UNIKIVI Rectory, in accordance with the requirements set for the system. the aim of this stage is to make the system available, accessible and understandable to the end user. the activities of this stage include training the end users and also carrying out tests on the beta version of the system, in order to ensure that it has the appropriate level of quality.

3.7.9 TEST PROCEDURE

Listed below are the procedures for testing whether GÊNESIS complies with the established objectives, taking into account the previously defined functional and non-functional requirements.

- ❖ Test with the first procedure

The first test procedure consists of the following parameters:

- A computer with Windows operating system;
- Checking the xampp server and the database management system, in this case MySQL;
- Checking that GÊNESIS is installed on the system;
- Verification of the installation of GÊNESIS on mobile devices;
- Verification of communication between the GÊNESIS desktop application and Arduino;
- Verification of communication between the GÊNESIS Android application and the server.

Table 6. Results of the first test procedure

Nº	Requirements	Result	
		Positive	Negative
01	Computer	✓	
02	Screen resolution	✓	
03	Xampp	✓	
04	Genesis latest version	✓	
Expected results	Check that the computer is ready to run the tests for the established use cases.		

- ❖ Test with the second procedure

The second procedure consists of:

- Checking the established use cases;
- Checking the integrity of the data in the database;
- Checking the performance of GÊNESIS and Arduíno.

Table 7. Results of the second test procedure

Nº	Functional Requirements	Result	
		Positive	Negative
01	User authentication	✓	
02	Register students	✓	
03	Driver registration	✓	
04	Print reports	✓	
05	Find Buses	✓	
Expected results	Check that the computer is ready to run the tests for the established use cases.		

4 Results

GÊNESIS was developed with the aim of facilitating transport management services. To this end, it was designed to be simple and easy to use. The final test results for the functional and non-functional requirements are positive (Table 8).

Table 8. Final test results

Nº	Procedure	Result	
		Positive	Negative
01	Non-functional requirements	✓	
02	Functional requirements	✓	

4.1 Login Screen

To gain partial or full access to GÊNESIS, you need to go through the login screen to authenticate (Fig. (a)). To access GÊNESIS (Android App) for students, you need to go through the login (authentication) screen, as shown in figure 6 (b).

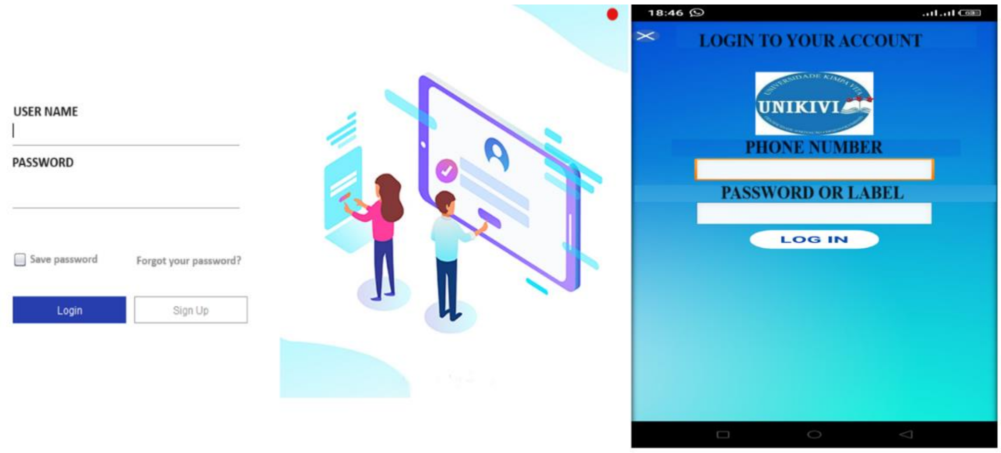


Fig. 7. Login screen

4.2 Main Menu

On the system's main screen, the user will have an overview of the data processed by the system, as well as access to the general menu which allows access to all the other system screens

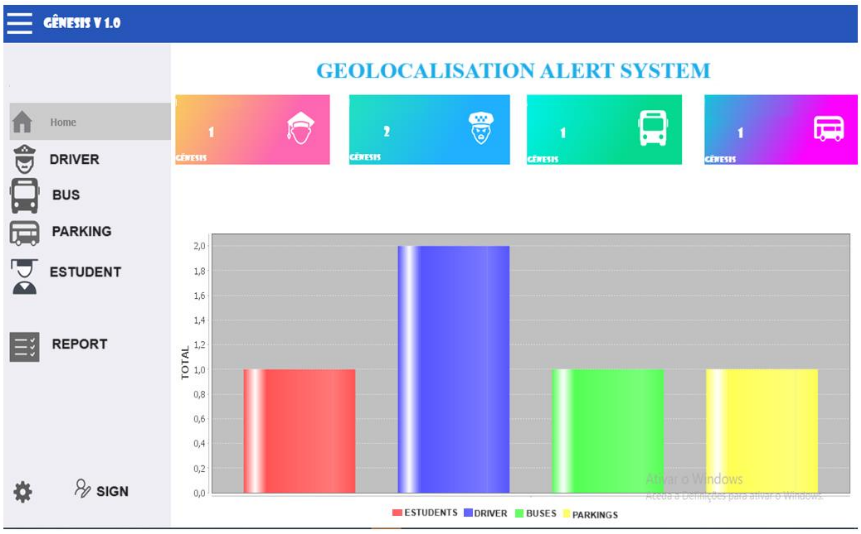
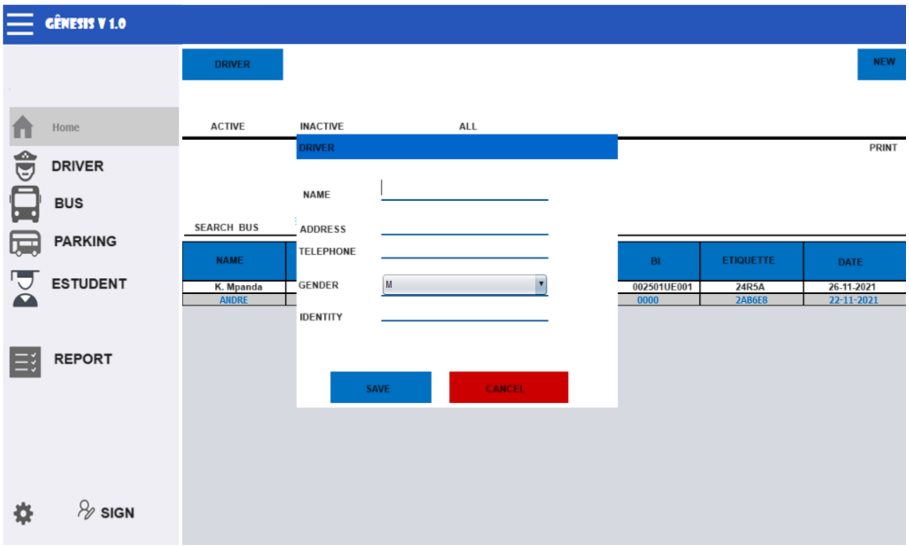


Fig. 8. Main menu screen

4.3 Driver Registration

This interface is where operations relating to drivers are carried out, from registering them to updating other relevant information.



DRIVER

NAME

ADDRESS

TELEPHONE

GENDER M

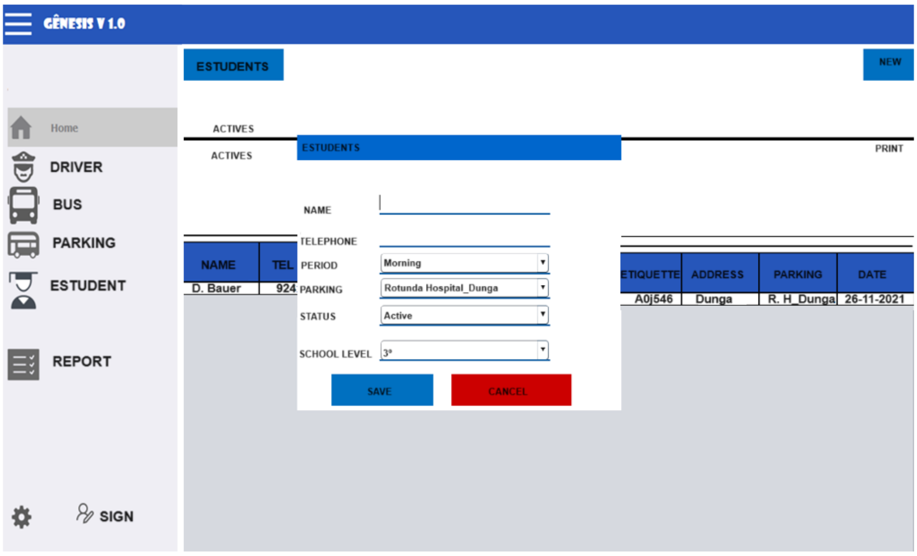
IDENTITY

NAME	BI	ETIQUETTE	DATE
K. Mpanda	002501UE001	24R5A	26-11-2021
ANDRE	0000	2AB6E8	22-11-2021

Fig. 9. Driver registration

4.3.1 Student Registration

Registration screen: screen responsible for registering the students who will have the privilege of receiving SMS alerts and locating the bus.



ESTUDENTS

NAME

TELEPHONE

PERIOD Morning

PARKING Rotunda Hospital_Dunga

STATUS Active

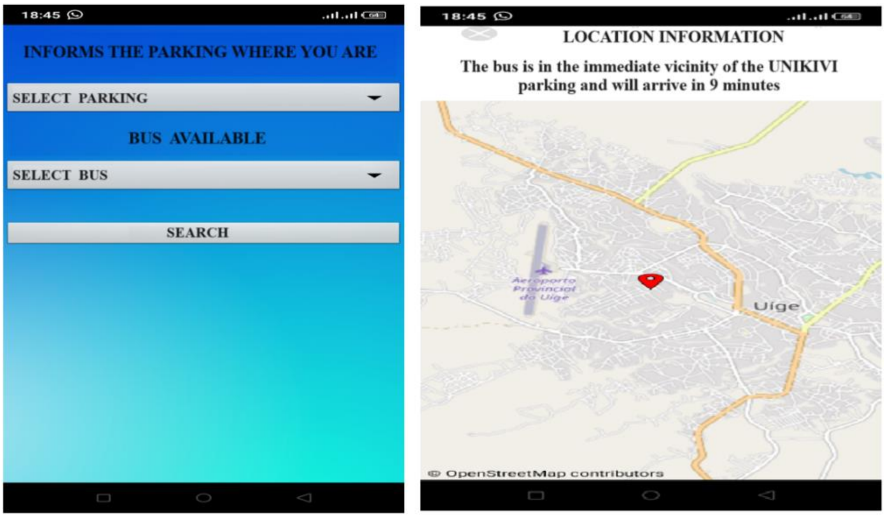
SCHOOL LEVEL 3º

NAME	TEL	ETIQUETTE	ADDRESS	PARKING	DATE
D. Bauer	924	A0j546	Dunga	R. H_Dunga	26-11-2021

Fig. 10. Student registration

4.3.2 Bus Consultation, Visualisation And Location Screen

This screen allows students to enter the stops they are at and choose the bus they want to locate in order to obtain its position on Google Maps (a). This screen allows the position of the bus to be visualised using the Google Maps API (b).


18:45
INFORMS THE PARKING WHERE YOU ARE
SELECT PARKING
BUS AVAILABLE
SELECT BUS
SEARCH
18:45
Aeroporto Provincial do Uíge
Uíge
© OpenStreetMap contributors

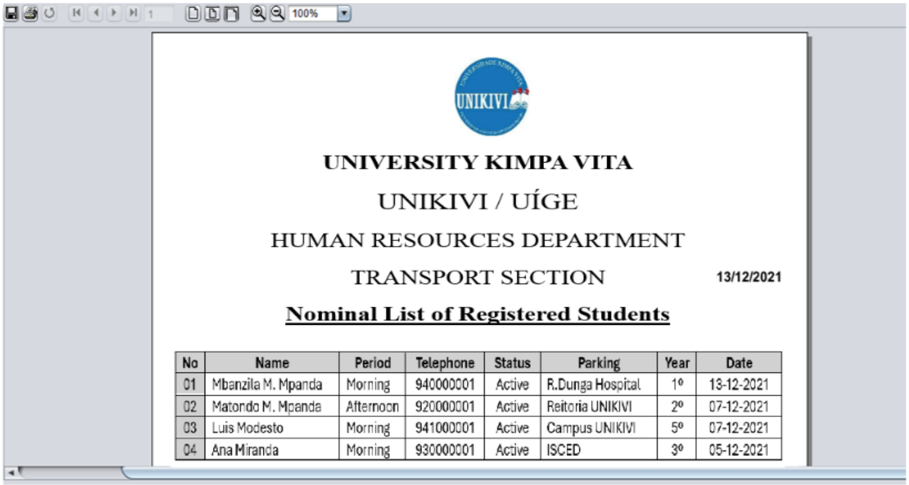
LOCATION INFORMATION

The bus is in the immediate vicinity of the UNIKIVI parking and will arrive in 9 minutes

Fig. 11. Bus location information

4.3.3 Report

A report is a document drawn up to report on a given situation. However, more than just a report, it contains information in an organised and summarised form, so that the recipients can understand the context and analyse it clearly.



UNIVERSITY KIMPA VITA

UNIKIVI / UÍGE

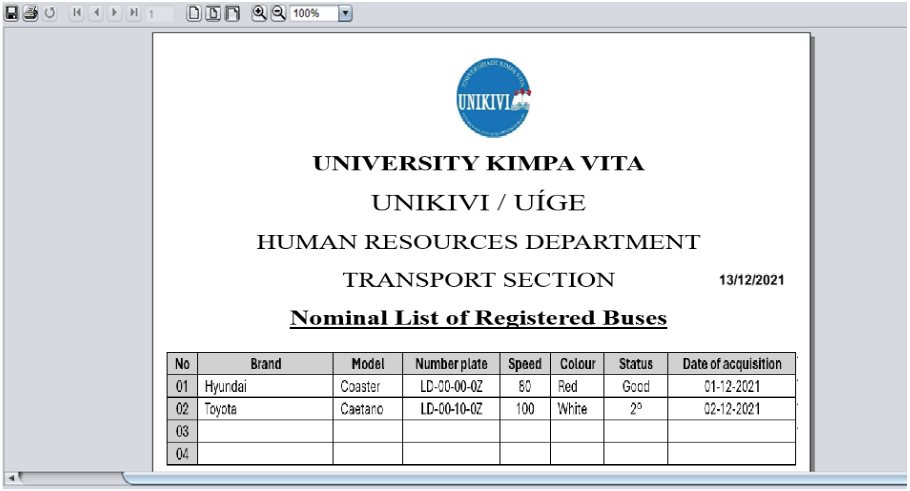
HUMAN RESOURCES DEPARTMENT

TRANSPORT SECTION 13/12/2021

Nominal List of Registered Students

No	Name	Period	Telephone	Status	Parking	Year	Date
01	Mbanzila M. Mpanda	Morning	940000001	Active	R.Dunga Hospital	1º	13-12-2021
02	Matondo M. Mpanda	Afternoon	920000001	Active	Reitoria UNIKIVI	2º	07-12-2021
03	Luis Modesto	Morning	941000001	Active	Campus UNIKIVI	5º	07-12-2021
04	Ana Miranda	Morning	930000001	Active	ISCED	3º	05-12-2021

Fig. 12. Registered Students Report



UNIVERSITY KIMPA VITA

UNIKIVI / UÍGE

HUMAN RESOURCES DEPARTMENT

TRANSPORT SECTION 13/12/2021

Nominal List of Registered Buses

No	Brand	Model	Number plate	Speed	Colour	Status	Date of acquisition
01	Hyundai	Coaster	LD-00-00-0Z	80	Red	Good	01-12-2021
02	Toyota	Caetano	LD-00-10-0Z	100	White	2º	02-12-2021
03							
04							

Fig. 13. Report of registered buses

4.4 Interpreting The System

After building the system, the first stage consists of coding the SIM808 GSM, GPRS Quad Band, GPS and Bluetooth module in the Arduino IDE. To do this, it is necessary to include the <DFRobot_sim808.h> library, which allows the Arduino board to communicate with the satellite, providing information such as latitude, altitude and longitude and carrying out serial communication between the Arduino and the module.

4.4.1 System Electronic Circuit

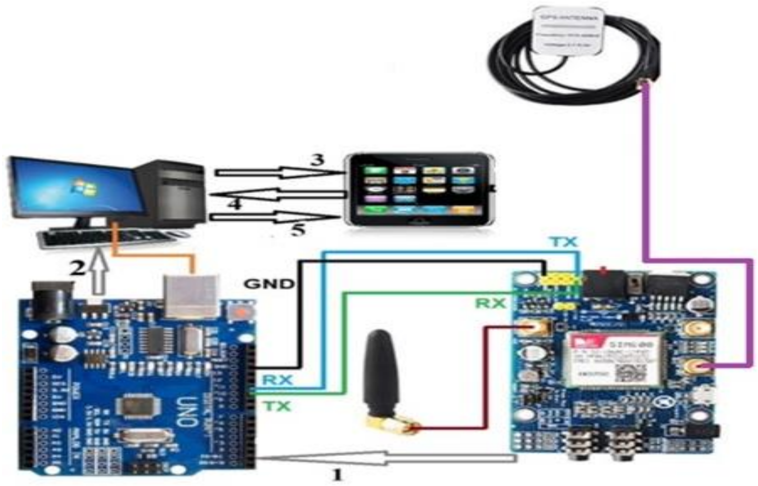


Fig. 14. System electronic circuit

4.4.2 Operation Of The System's Electronic Circuitry

Table 9. Electronic circuit operation

Sequence	Interpretation
1	SIM808 sends the coordinates to the Arduino
2	Arduino in turn sends the coordinates to the system where they will be stored
3	The SIM808 module sends SMS alerts when the bus is running.
4	The student tracks the positioning of the bus
5	The system, in turn, sends the coordinates to the students, allowing them to visualise the positioning of the bus using Google Maps.

5 Conclusion

This article presents the development of the GÊNESIS system, which is an alert and geolocation system to notify by SMS about the operation of the Kimpa Vita University buses on working days and locate the buses via an Android App. To meet the needs, interview, observation and survey techniques were used, as well as analytical and historical methods.

With the support of the RUP methodology and technological tools such as MySql, NetBeans (java language), Astah Comunity and Arduino, it was possible to realise the system. The specific objectives were achieved, since the system is viable and easy to use, allowing students and staff to receive alerts about the buses' operation and locate their position via the Android app.

References

[1] Arsanjani, A., Johnston, S., & Smith, J, The RUP Update for ServiceOriented Architecture. Rational Software White Paper, 2005.

[2] Banzi, M., Getting Started with Arduino, O'Reilly Media Inc, 2010.

[3] Beck, K. and Fowler, M., 1ª edição Planning Extreme Programming, Addison Wesley, 2000.

[4] Blaha, M., Rumbaugh, J., 2. ed., Modelagem e Projectos Baseados em Objetos com UML 2. Tradução Daniel Vieira. Rio de Janeiro: Elsevier.2006.

[5] Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., Stal, M. (1996). Pattern-Oriented Software Architecture, John Wiley and Sons, New York, NY.

[6] Cockburn, A. (2005). Escrevendo Casos de Uso Eficazes, Porto Alegre: Bookman.

[7] Crocker, P. More than maps: the evolution of location-based applications.

Disponível em: http://www.decarta.com/files/gigaom_more_than_ maps. 2014.

[8] Dall´ogio, P., 2ª ed PHP Programando Com Orientação a Objetos, Novatec. 2009.

[9] Deitel, H. M., Deitel, P. J. 6. ed., Java Como Programar; tradução Edson Furmankiewicz; revisão técnica. São Paulo: Pearson. 2005.

[10] Flávio, B. (n.d). Aprenda Arduíno do zero.
Disponível em: https://flaviobabos.com.br/wp-content/uploads/2020/11/o-guia-definitivo-para-aprender-arduino-na-pratica-1.pdf.

[11] Fonseca, E. G. P. (2010) – BEPPU, Mathyan Motta – Arduino – UFFCT. Glovacki Graneman de Melo, J. L., & Baraniuk, J., (2012). Minicurso de Arduino. Curitiba.

[12] Google Maps. (2013). API do Google Maps - Criar aplicativos com base no local. Acesso em.
Disponível em: https://developers.google.com/maps/locationbased-apps. fev, 14,2023.

[13] Guedes, G. T. A. 2a edição UML – 2. Uma Abordagem Prática. São Paulo: Novatec Editora., P., (2000). The Rational Unified Process – An Introduction, AddisonWesley. 2009.

[14] Kruchen, P. Introdução ao RUP – Rational Unified Process, Rio de Janeiro: Editora Ciência Moderna Ltda. 2003.

[15] Larman, C. (2007). Utilizando UML e Padrões: uma Introdução a Analise e ao Projecto Orientados a Objetos, Porto Alegre: Bookman.

[16] Lawrence, S. P. (2004). Engenharia de Software. (D. Franklin, Trad.) São Paulo: ABDR.

[17] Lecheta, R. R. (2010). Google Android: aprenda a criar aplicações para dispositivos móveis com o Android SDK. São Paulo: Novatec.

[18] Mamede, H. S. Segurança Infromática nas organizações. Lisboa: FCA. 2006.

[19] Meike, B. et al. Desenvolvimento de aplicações android. São Paulo: Novatec. 2009.

[20] Menjoge, Z. Software Development using the Knowledge Insight Approach. North Carolina State. Thesis [Master of Science in Computer Science] North Carolina State University. 2003.

[21] Michael, M., 2ª ed, Iniciando Arduíno (Tecnologia em Ação). 2010.

[22] Nunes, M. & O'Neill, E., Fundamento de UML. Rio de Janeiro. Ponta Grossa. 2015.

[23] Oliveira, J. A., Sistema de gestão de uma frota de veículos baseados em GPS.Lisboa- Portugal: Mariana Vasques. 2014.

[24] Paulo, C. S. (2019). Projecto de Software Usando UML. Acesso em 07/02/2023.
Disponível em: https://www.escavador.com/sobre/6981859/paulo-cezar-stadzisz.

[25] Peterson, L. L. & Davie, B.S. (2007). Computer Networks – A System Approach. 4ª Ed., Editora Elsevier.

[26] Pollice, G. (2001). Using the IBM Rational Unified Process for Small Projects: Expanding Upon eXtreme Programming. Rational Software White Paper.
Disponível: <http://www3.software.ibm.com/ibmdl/pub/software/rational/web/whitepapers/2003/tp183.pdf>.

[27] Ribeiro, M. A. Automação Industrial, 4ª ed., Salvador: Tek Treinamento & Consultoria Ltda. 2001.

[28] Runeson, P. and Greberg, P. Extreme Programming and Rational Unified Process – Contrasts or Synonyms? Lund University, Sweden. 2004.

[29] Santos, N. P. Arduino, Introdução e Recursos Avançados. 2009.

[30] Woodill, G. The Mobile Learning Edge: 1st ed, Tools and Technologies for Developing Your Teams, McGraw-Hill, Ed., Available: http://www.mobilelearningedge.com. 2010.