Software Requirement Specification Document Augmented Reality for Automotive Maintenance Assistance

Judy Wagdy, Laila Hany, Rana Raafat, Steven Albert
Supervised by: Dr. Essam Eliwa, Eng. Nada Ayman

December 17, 2022

Table 1: Document version history

Version	Date	Reason for Change
1.0	17-Dec-2022	SRS First version's specifications are defined.

GitHub: https://github.com/judykhairalla/AR-Automotive-Assistance

Contents

1 Introduction **4**
- 1.1 Purpose of this document . . . 4
- 1.2 Scope of this document . . . 4
- 1.3 Business Context . . . 4

2 Similar Systems **4**
- 2.1 Academic . . . 4
- 2.2 Business Applications . . . 7
  - 2.2.1 WorxAR . . . 7
  - 2.2.2 Re'flekt One . . . 7

3 System Description **8**
- 3.1 Problem Statement . . . 8
- 3.2 System Overview . . . 8
  - 3.2.1 Scanning . . . 8
  - 3.2.2 Object detection and recognition . . . 8
  - 3.2.3 Instructions request . . . 9
  - 3.2.4 AR instructions overlay and object tracking . . . 9
- 3.3 System Scope . . . 9
- 3.4 System Context . . . 10
- 3.5 Objectives . . . 11
- 3.6 User Characteristics . . . 12

4 Functional Requirements **12**
- 4.1 System Functions . . . 12
- 4.2 Detailed Functional Specification . . . 15

5 Design Constraints **17**
- 5.1 Standards Compliance . . . 17
- 5.2 Hardware Limitations . . . 17
- 5.3 Other Constraints . . . 17

6 Non-functional Requirements **18**
- 6.1 Safety . . . 18
- 6.2 Usability . . . 18
- 6.3 Portability . . . 18
- 6.4 Scalability . . . 18
- 6.5 Maintainability . . . 18
- 6.6 Security . . . 18

7 Data Design **19**
- 7.1 Dataset . . . 19
- 7.2 Database . . . 20

8 Preliminary Object-Oriented Domain Analysis **21**

9 Operational Scenarios **22**

10 Project Plan **24**

11 Appendices **27**
11.1 Definitions, Acronyms, Abbreviations 27
11.2 Supportive Documents 27
11.2.1 Abou Ghaly Motors Collaboration 27
11.2.2 Wikitude Edu License 29

Abstract

In today's fast-paced society, fewer users tend to read user manuals. Instead, they are increasingly inclined to graphical, easy-to-navigate, and quick instructions. In the domain of automotive assistance, the lack of knowledge of the typical user, coupled with the seeming difficulty of straightforward procedures, encourages them to seek the help of a specialist. Augmented Reality (AR) replaces paper-based manuals by overlaying computer-generated graphical instructions in the real environment. Computer Vision (CV) techniques and Machine Learning (ML) models can be utilized for car component feature detection and repair instructions. The proposed system is an AR-based application that shall be accessible on mobile phones, tablets, and wearable devices. The application aims to guide average-skilled users through standard maintenance procedures and aid them in getting acquainted with their vehicle's features.

1 Introduction

1.1 Purpose of this document

The Software Requirements Specification (SRS) document aims to illustrate the proposed system's features, explain its objectives, and set guidelines for the developers while working on it. This document is intended for the developers and stakeholders of the application.

1.2 Scope of this document

This document covers the application's detailed description, functional and non-functional requirements, basic interface and data designs, and an analysis of the classes needed and the relations between them.

1.3 Business Context

In the United States alone, new car sales in 2022 are estimated to reach 15.47 million new vehicles, according to [1]. All these cars are expected to undergo routine maintenance and have their features illustrated by the manufacturer. In [2], it is highlighted that AR can reinvent how the industry communicates with the end buyer, resulting in a new marketing strategy that is very convincing. Instead of using paper-based manuals, companies may utilise augmented reality (AR) to help buyers understand the car's features, as illustrated in [3]. Not only will it benefit potential buyers, but existing customers who wish to save time and money and do their own basic maintenance tasks, will greatly benefit from such a system.

2 Similar Systems

2.1 Academic

Malta et al. [4] proposed building an AR-based system that recognizes mechanical car parts in real time, and gives directions in a work order for maintenance technicians to follow. The tasks in the technicians' field of view are also accompanied by audio messages. The recognition of the

automobile parts was based on learning a deep neural network; YOLOv5m network. The dataset created was a total of 900 images in various lighting conditions, of a 1998 Peugeot 206 car model interior. Eight constituent parts were captured:

1. oil dipstick
2. battery
3. engine oil reservoir
4. engine oil reservoir
5. air filter
6. brakes fluid reservoir
7. coolant reservoir
8. power steering reservoir

The model was trained on a Google Colab virtual machine with a total of 250 epochs, with the use of the PyTorch framework, and it showed results with precision of 0.985 and recall of 0.994. The model's classes labels were in Portuguese, and the objects recognized by the system were annotated using those labels as shown in Figure 1. The annotation was done using the Visual Object Tagging Tool (VoTT), which is an open source labeling and annotation tool for either video or image assets. To ensure the technicians are using the right tools and equipment, each equipment is saved by a unique code within the system. FFmpeg Platform was used to record, convert, and stream video and audio within the system. The system was also offered to the users both offline and online.

Figure 1: Augmented Reality Maintenance Assistant Using YOLOv5 [4]

Konstantinidis et al. [5] propose MARMA, an AR-based maintenance assistant. It is a Unity mobile application that has been implemented and tested by maintenance engineers in the automotive industry. This system identifies a component in the user's environment, and provides visualizations to its maintenance instructions depending on its condition. To build the system, the

Vuforia Object Scanner (including the FAST, and SURF algorithms) is utilized for the feature extraction and vector creation processes of the machine. In the runtime, the Vuforia natural feature tracking (NFT) method is integrated to identify and track the component of interest throughout the recording process after feature matching using the SURF algorithm. Employing the tiny-YOLOv3 detector, the system locates the area of interest within the camera's frame during the real-time process and crops the recorded image. The 3D CAD models are then overlaid on the asset of interest accompanied by a navigable guide of instructions, as shown in Figure 2 Comparing MARMA to paper-based and digital processes, it shortened the overall repair time of a car's AC compressor by 30%.

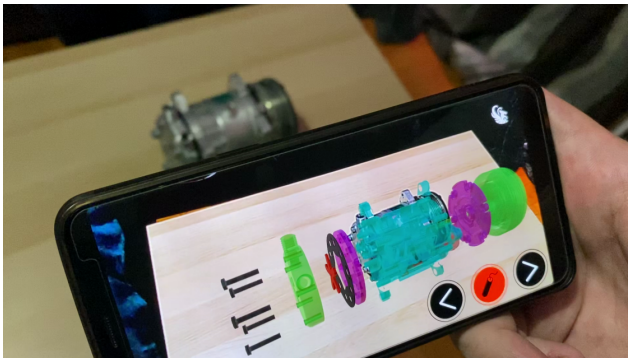


Figure 2: A Mobile Augmented reality system to assist auto mechanics [5]

Ana Cachada et al. [6] proposes a prototype for an AR maintenance application that serves an automotive production unit. The authors acknowledge how maintenance tasks are becoming increasingly complex, due the numerous steps required to be performed with various tools. Thus, Ana Cachada et al. designed the AR application shown in Figures 3 and 4 to make the training and execution processes easier. The application devises an Intelligent Personal Assistant (IPA) to help technicians in performing the maintenance tasks. Also, an Intelligent Decision Support System (IDSS) was utilized to further improve the decision making process during the execution of the maintenance tasks. Prior to the decision making step, Business Process Model and Notation (BPMN) standard is followed to describe the maintenance steps in a graphical notation, and then they are fed into Camunds platform to be converted into XML files. Then, the XML files are processed to be displayed by Unity 3D. Afterwards, Vuforia AR SDK is used to develop the AR marker-based application. The authors then used Microsoft HoloLens and Lenovo 10 Tablet to furnish the application. To validate the application, a System Usability Scale (SUS) Questionnaire was carried and resulted in a score of 84.4 which corresponds to excellent. Ultimately, the authors aim to make the holoLens more immersive, along with supporting voice media to provide more assistance to technicians.

Figure 3: AR for Industrial Maintenance [6]

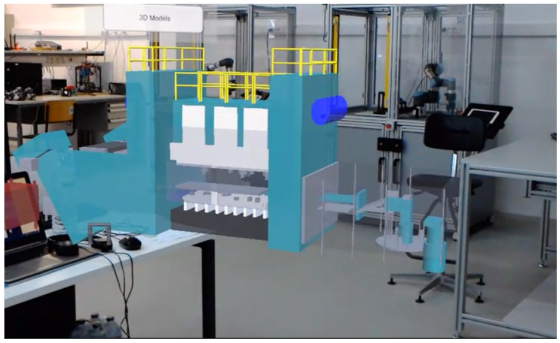


Figure 4: AR for Industrial Maintenance [6]

2.2 Business Applications

2.2.1 WorxAR

WorxAR [7] is an AR-based application, as shown in Figures 5 and 6, developed by an Australian company called Tradiebot. The application helps technicians by diagnosing any issues, locating and highlighting the parts, and guiding them throughout performing services and repairs. The latest digital schematic diagrams, data sheets, and step by step instructions are overlaid onto the user's field of view without concealing what they're working on. They're also notified about what cannot be modified without compromising the vehicle systems' safety and integrity. WorxAR also allows users to enter a remote video call with an expert and receive interactive guidance. Virtual quality control check sheets, including capturing images for quality records, are used so that the users can verify the condition and status of the job. It's made for Phones, Tablets, and Smart Glasses.

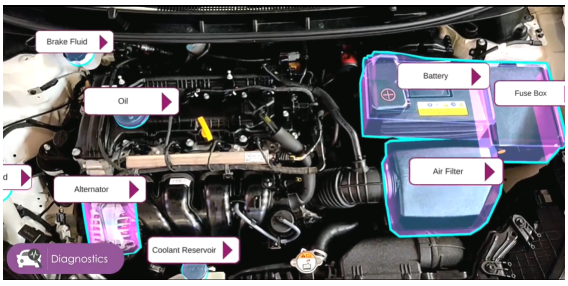


Figure 5: WorxAR [7]

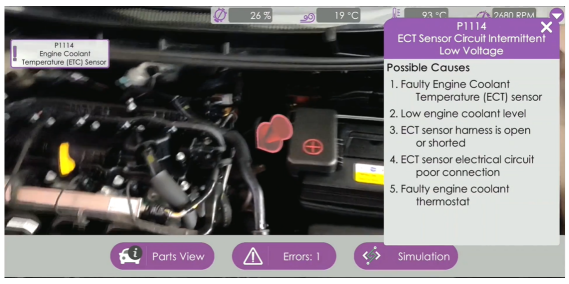


Figure 6: WorxAR [7]

2.2.2 Re'flekt One

Re'flekt One [8] is an AR content platform that enables innovative enterprises to create interactive manuals and documentation. It has provided solutions for many leading companies in the market, such as:

Bosch

The Bosch Common Augmented Reality Platform (CAP) [9], based on the Re'flekt One operating system, targets workshop trainees. It enables them to stay updated with the constantly changing

vehicle product lines developments. The system displays for them any new vehicle functionality or information on demand- as shown in Figures 7 and 8, as well as allows them to enlarge and rotate the components in relevance. It's made available on both Tablets and Smart Glasses (such as the Microsoft HoloLens).

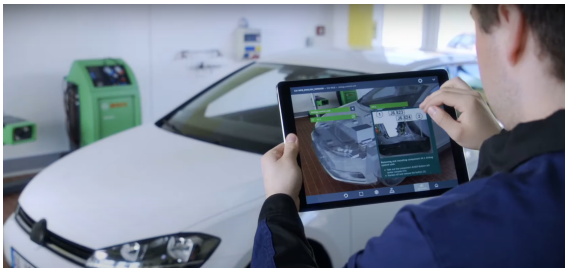


Figure 7: Bosch [8]

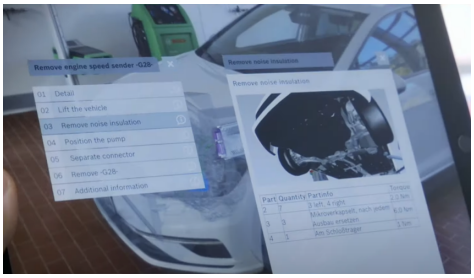


Figure 8: Bosch [8]

3 System Description

3.1 Problem Statement

In theory, owning a car provides autonomy and freedom as opposed to public transportation. However, car owners usually receive little to no initiation to their vehicle's standard maintenance procedures and features. Consequently, they show an inclination to pay a visit to a professional even for low-difficulty challenges they face, ranging from not understanding the signs on the dashboard to changing one of the car fuses. The traditional solution to this issue is to read the car manual provided by the manufacturer in a paper-based or pdf format. Nevertheless, most users show no interest in skimming through a large document to solve their problem. That is why, an AR application that superimposes information and maintenance procedure instructions graphically on the real-environment, constitutes a modern user-friendly alternative. This application would also encourage user independence and ease of access to information.

3.2 System Overview

As illustrated in figure 9, the system workflow can be modeled in 4 main steps.

3.2.1 Scanning

The user is prompted to scan their car from the inside or outside using their device's camera.

3.2.2 Object detection and recognition

An object detection and recognition model is utilized to identify the objects in real time. This model is trained via Wikitude Studio using a dataset composed of images of different car components.

3.2.3 Instructions request

Depending on the user's choice, the necessary instructions are requested from the cloud Firebase backend. These instructions include text and 3D models with their animation, they guide the user through a certain maintenance procedure or get familiar with a feature. The user has also the option to request remote assistance from a specialist for more advanced maintenance procedures. In this case, a live video call takes place where the professional technician can give instructions throughout the call using their voice or AR annotations through Microsoft's Dynamics 365 Remote Assist.

3.2.4 AR instructions overlay and object tracking

The retrieved instructions are overlaid onto the car components via the Wikitude SDK.

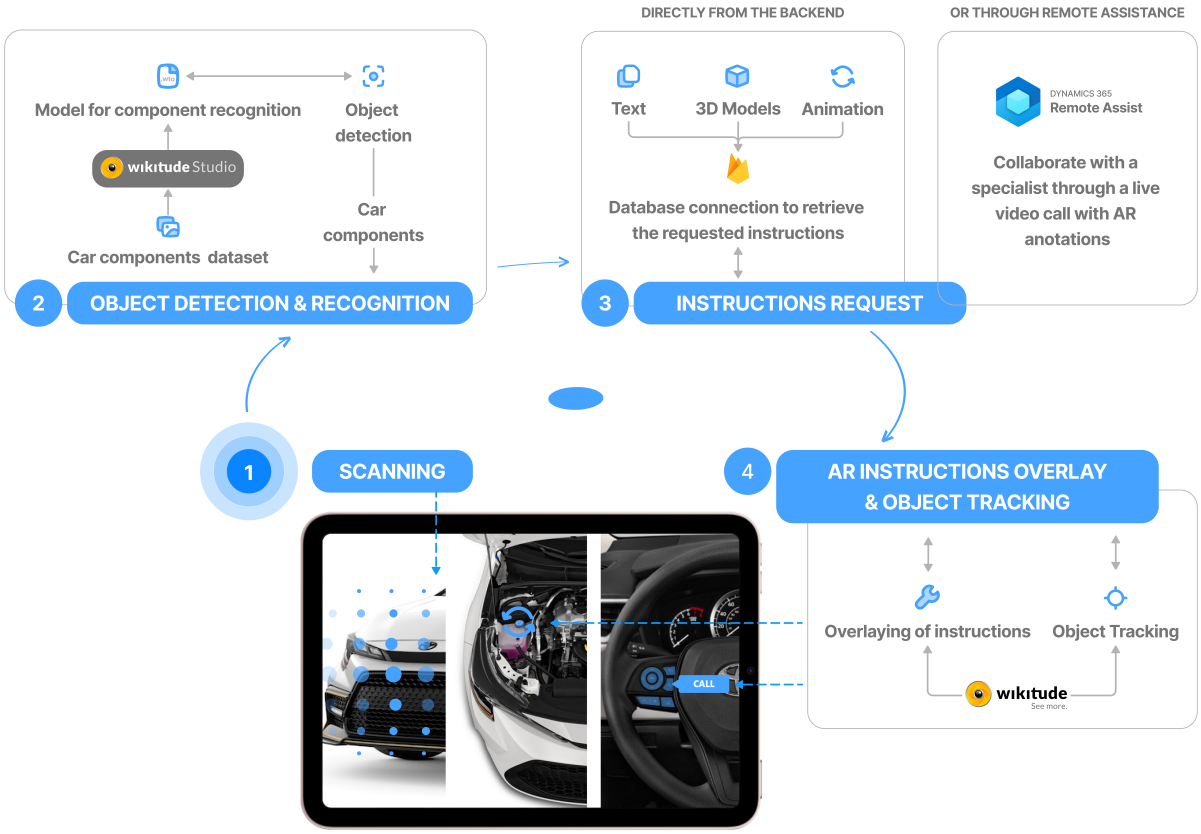


Figure 9: System Overview

3.3 System Scope

The system shall address the following:

- The system shall devise a markerless-based AR system to display detailed instructions for car assistance tasks.

- The system shall utilize machine learning models and computer vision techniques for car components detection and recognition.
- The system shall overlay the guiding instructions to assist car owners in maintaining their cars and getting acquainted with their cars' features.
- The system shall include procedures that ensures car owners' safety, that is the system will not incorporate hazardous procedures.
- The system shall be able to connect car owners to an expert at the headquarters for assistance.
- The system shall offer complementary services for car owners such as finding nearby maintenance centers.
- The system shall be compatible with different hand-held devices.

3.4 System Context

The system is built to interact with various external systems in order to achieve its functional and non-functional requirements. As illustrated in figure 10, the application is developed using the Flutter SDK. A connection with a Firebase backend is established for inserting and retrieving the data modeled in section 7.1. Moreover, the Wikitude Studio, SDK, and Encoder provide the needed environment to generate object recognition models, optimize 3D models for low-performance devices, and overlay and track objects onto the user's environment. Furthermore, Microsoft's Dynamics 365 Remote Assist shall be integrated to ensure live calls with technicians with the ability to overlay AR annotations.

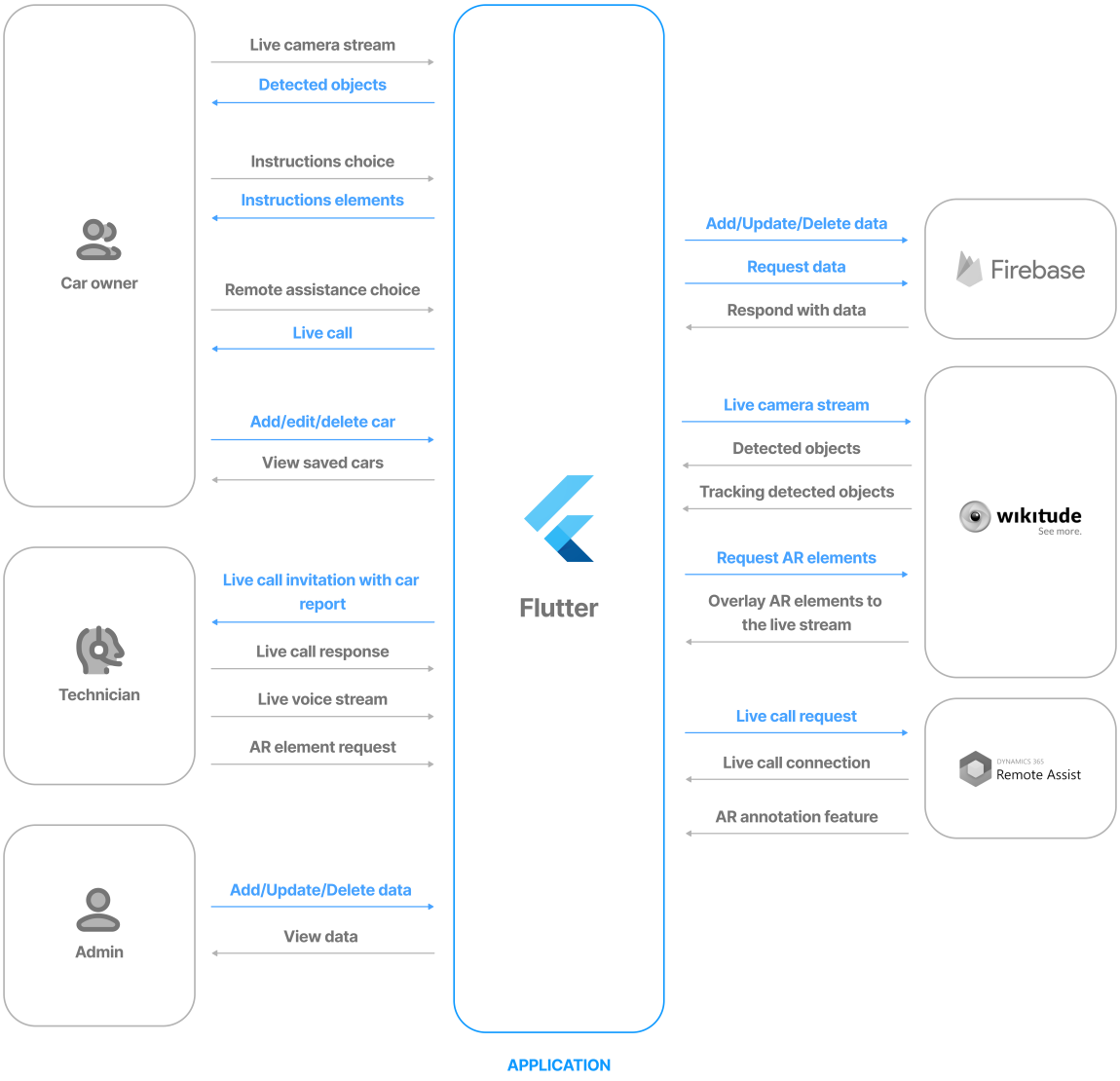


Figure 10: System Context

3.5 Objectives

- To develop a user-friendly application that is accessible through mobile devices and tablets to assist average-skilled car owners to easily follow the instructions for their vehicle maintenance.
- To allow the detection and recognition of the different car parts, whether under-hood components or car dashboard triggers.
- To overlay detailed instructions over the car parts, in the real scene, to perform a task. Instructions can be textual, pictorial, or animated 3D models.

- To display warnings to car owners, during a procedure, to prevent them from executing the procedure incorrectly.
- To offer a tele-assistance feature for the car owners to get assistance from a remote expert.
- To offer complementary car support features to car owners including list of nearby maintenance centers, reminders for car maintenance procedures, and maintenance checklist for road trips.
- To auto-generate detailed report after each car maintenance procedure.

3.6 User Characteristics

The system mainly targets car owners to assist them in maintaining their cars and getting acquainted with its features. Consequently, the system will also be used by admins and technicians to facilitate the maintenance and feature acquainting procedures onto the car owners. Also, the system is designed to be user-friendly.

Howevere, all users must have

- basic mobile handling skills e.g connecting to the internet, downloading apps, and signing into a software.
- a brief understanding of what AR is and how it works.

4 Functional Requirements

4.1 System Functions

The below use case diagram in figure 11 demonstrates the system functional requirements. The system is composed of three user types: car owner, admin, and technician.

Figure 11: Modules Use Case Diagram

1. General Requirements

   (a) The system shall detect and recognize the different car components scanned using machine learning models and computer vision techniques. (GR01)

   (b) The system shall label the different car components when scanning the car. (GR02)

   (c) The system shall overlay the guiding instructions in pictorial, textual, and 3d animated forms. (GR03)

(d) The system shall generate a full report of each car issue after each maintenance procedure. (GR04)

2. Car owner module

   (a) The car owner shall be able to add/edit their car's specifications into the system e.g car model. (COR01)

   (b) The car owner shall be able to choose a procedure from the available procedures, whether it is a repair procedure - e.g add car coolant, jump-start car, replace car fuses, change car tyres - or a feature acquainting procedure. (COR02)

   (c) The car owner shall be able to scan the car's under-hood components or dashboard triggers. (COR03)

   (d) The car owner should be able to preview all the instructions to be done ahead. (COR04)

   (e) The car owner shall be able to view overlaid instructions in animated 3D and textual forms. (COR05)

   (f) The car owner shall receive some warnings if a step was done incorrectly. (COR06)

   (g) The car owner shall be able to seek assistance from a remote mechanical assistant. (COR07)

   (h) The car owner shall be able to view historical reports for all procedures done. (COR08)

   (i) The car owner shall be notified when it's time to do maintenance to their car. (COR09)

   (j) The car owner shall be able to view the nearby maintenance centers depending on their car model. (COR10)

   (k) The car owner should be able to formulate a maintenance checklist to be viewed before traveling. (COR11)

3. Admin module

   (a) The admin shall be able to add/edit/delete a car model to the database. (AR01)

   (b) The admin shall be able to add/edit/delete the procedures available into the system. (AR02)

   (c) The admin shall be able to view the history of car owners' reports and history of the tele-communication between the car owners and the technicians. (AR03)

4. Technician module

   (a) The technician shall receive assistance requests from car owners. (TR01)

   (b) The technician shall be able to view a full detailed report of the user's car issues. (TR02)

   (c) The technician shall be able to view what is scanned by the car owner's camera, while in the live call. (TR03)

   (d) The technician shall be able to annotate on the screen to assist the car owner, while in the live call. (TR04)

4.2 Detailed Functional Specification

Table 2: selectProcedure Function Description

Name	selectProcedure
Code	COR02
priority	Extreme
Critical	This function is crucial to allow the car owner to proceed with a specific procedure.
Description	The car owner is presented with a list of procedures which they shall choose one to be guided through.
Input	carModelID, procedureID
Output	instructionIDs[]
Pre-condition	- The car owner must be logged in. - The car owner must have entered their car specs to be presented with the appropriate list of procedures.
Post- condition	The instructions will be overlaid onto the screen for the car owner.
Dependency	Depends on COR01
Risk	none

Table 3: overlayInstructions Function Description

Name	overlayInstructions
Code	GR03
priority	Extreme
Critical	This is one of the most crucial functionalities in the system as it encompasses the display of the procedure instructions to assist users in performing the maintenance tasks.
Description	After the car owner selects the procedure, this function displays the detailed steps by overlaying it using AR to accomplish the procedure. The textual instructions are accompanied with 3d animated models overlaid onto the screen to ensure the car owner will execute the steps correctly.
Input	instructionID[]
Output	True if the instructions are overlaid successfully.
Pre-condition	- User's car model must be available - A specific procedure is selected
Post- condition	A report is formulated if the task was for maintenance
Dependency	Depends on COR02
Risk	Instructions for this procedure are unavailable

Table 4: generateReport Function Description

Name	generateReport
Code	GR04
priority	Extreme
Critical	This function is crucial to facilitate the assistance process onto the technicians.
Description	After the car owner finishes a maintenance task, a detailed report of the issue is generated; the report indicates the car model, the issue happened, the procedure carried out, and the dates on which this issue happened.
Input	procedureID, userID, carModelID
Output	True if report was generated successfully.
Pre-condition	The car owner must have reported a car problem whether they chose a procedure or requested remote assistance.
Post- condition	The report is displayed to the user.
Dependency	Depends on COR02 and COR07
Risk	The car owner may enter an incorrect description of their car issue.

Table 5: addCarModel Function Description

Name	addCarModel
Code	AR01
priority	Extreme
Critical	Essential for the system to support more car models.
Description	This function allows the admin to add a new car model with its available procedures.
Input	carModelID, procedureID, instructionID
Output	True if new model was added successfully
Pre-condition	Admin needs to be logged in.
Post- condition	A pop-up indicates that the car model was added successfully.
Dependency	none
Risk	Admin may add incorrect information but this can be solved by the editing feature.

Table 6: annotate Function Description

Name	annotate
Code	TR04
priority	Extreme
Critical	This function is crucial to allow technicians to guide car owners through a technical procedure through communicating with a remote access and control API.
Description	After the technician receives an assistance request, they will be able to assist the car owner in a procedure through annotating on the screen e.g adding arrows to point to a specific component.
Input	modelID
Output	True if API preformed annotation successfully.
Pre-condition	- Technician must be logged. - Technician must accept the assistance request.
Post- condition	Car owner will view the annotation on their side.
Dependency	Depends on TR01
Risk	- Annotation is not visible to the car owner. - Connection error.

5 Design Constraints

5.1 Standards Compliance

- The smartphone must have access to an internet connection to use any online features.

5.2 Hardware Limitations

- The system must be installed on a device that supports Wikitude Flutter SDK Instant Object Tracking. The requirements that must be fulfilled can be found in Wikitude's documentation [10].

5.3 Other Constraints

- The system is compatible with the car models implemented only; which for the meantime are Toyota Corolla 2015, Hyundai Accent 2010, and Chevrolet Optra 2007.
- Only supported in English and Arabic at the time.
- Under-hood tasks promoted by the system should not have a safety hazard on the user.
- In order to be scanned correctly, components must be in clear view. Environment must not be dark or shaded, and device should be at sensible distance from car.

6 Non-functional Requirements

6.1 Safety

To ensure the correct application of instructions with optimum safety measures, the system shall display:

- Warning messages with the level of complexity of the requested instructions.
- Warning messages with the possible fault execution of instructions.
- Alarm messages when the user fails to follow instructions correctly.

6.2 Usability

The system shall provide an easy to understand UI (User Interface) and a satisfactory UX (User Experience). This shall be accomplished through:

- Clear overlaying of AR elements on the user's environment.
- A practical introduction to the use of AR.
- Easily navigable instructions.

6.3 Portability

The system shall be available on smartphones and tablets that support the Wikitude SDK as mentioned in section 5.2. The system shall be used on devices connected to the internet while giving the user the option of downloading a certain car model's instructions to be used offline and updated when available.

6.4 Scalability

The system shall be designed to make it simple to incorporate new car models and support new instructions.

6.5 Maintainability

The system shall respect the SOLID principles (Single Responsibility Principle (SRP), Open/Closed Principle, Liskov Substitution Principle (LSP), Interface Segregation Principle (ISP), and Dependency Inversion Principle (DIP)) to provide smooth long-term maintenance.

6.6 Security

All passwords shall be hashed, and admin privileges shall only be accessible from admin accounts.

7 Data Design

7.1 Dataset

An initial dataset, as shown in figure 12, has been constructed from a car owned by one of the team members. It consists of 461 images of car components distributed into the following classes: coolant, engine, oil reservoir, power, and windshield fluid.

Figure 12: Dataset

7.2 Database

The dataset images are utilized to create a point cloud on Wikitude Studio, from which a Wiktude Target Object (.wto) model is generated. This model shall be stored in the database and used for object detection and tracking. Moreover, FBX 3D models are converted into the WT3 format using the Wikitude Encoder and are inserted in the database in their new format. The Enhanced Entity Relationship Diagram (EERD) in figure 13 demonstrates the structure of the stored data.

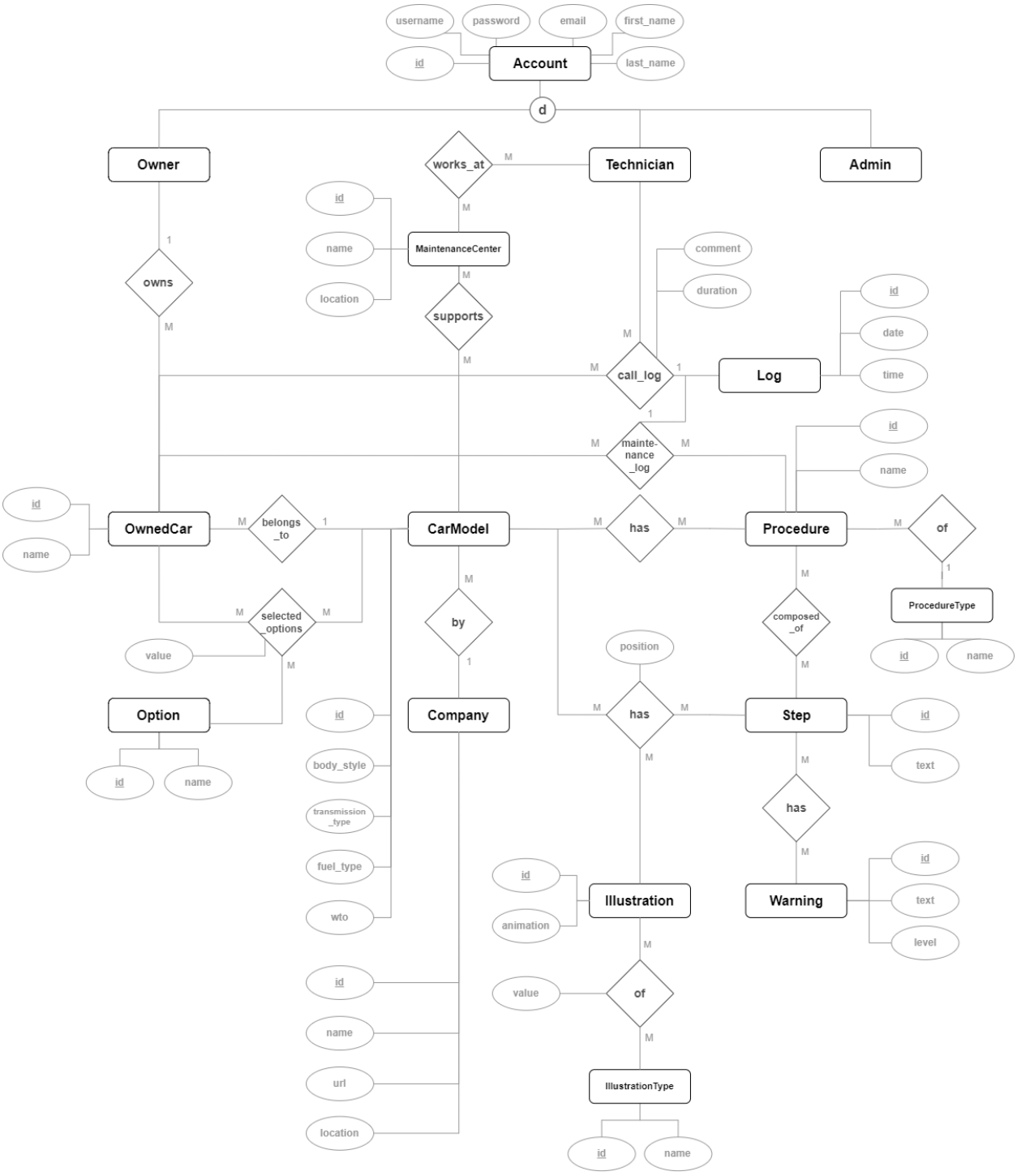


Figure 13: EERD

8 Preliminary Object-Oriented Domain Analysis

An initial Class Diagram of the system is found in Figure 14

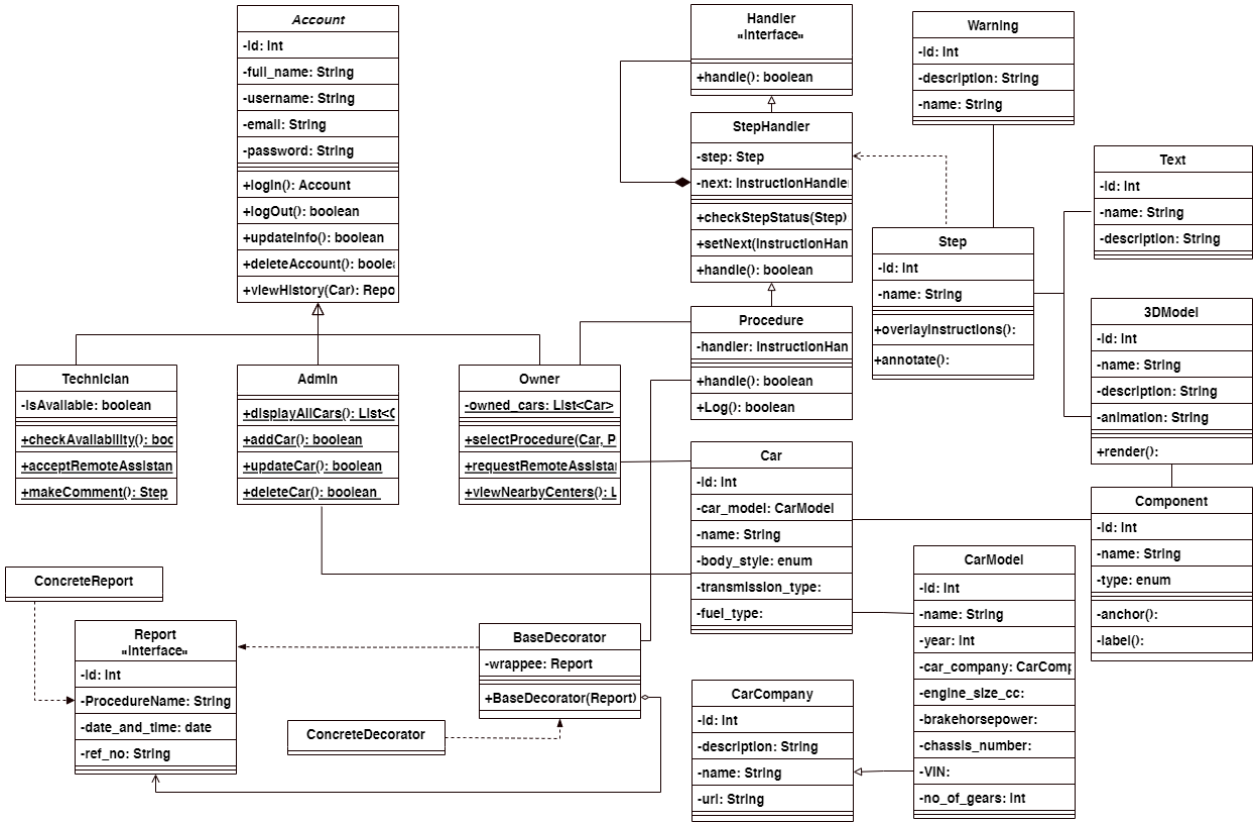


Figure 14: Initial UML Class Diagram

9 Operational Scenarios

Scenario 1: User registers their car

Initial assumption: The user has the app installed on their phone, with camera access permission granted. User also has access to their car's VIN.

Normal: The user taps on "Add your vehicles" from the home screen. The phone's camera launch and user is asked to scan their vehicle's VIN. The app will detect the car's manufacturer, model and category and show them to the user to confirm. If approved by the user, the user will be redirected to the home screen and can now access their car's instruction manual and maintenance operations.

What can go wrong:

1. VIN not identified due to difficult lighting conditions or being obstructed up by dust.
2. VIN identified but does not belong to a supported car.
3. Camera cannot launch due to lack of camera access permission by the user.

Scenario 2: User starting a maintenance procedure

Initial assumption: User has already registered their vehicle, and is currently in front of it.

Normal: The user opens the app, and is presented with their cars. The user chooses the car

they want to do the maintenance procedure on and then chooses the procedure they're looking for from a list of maintenance procedures. The camera launches, and asks the user to align the car outline drawn on their phone's screen to the car on front of them. Once the app detects the car's alignment, the maintenance instructions start.

What can go wrong:

1. Car alignment not detected due to harsh lighting conditions or damage to the exterior car body.

Scenario 3: A User discovering features on the dashboard

Initial assumption: User has already registered their vehicle, and is currently inside of it.

Normal: The user opens the app, and is presented with the cars they registered. The user chooses the car they want the user manual of. The camera launches and they are asked to align their steering wheel to the outline. Once aligned, the user will be able to touch a dashboard button on the phone's screen, and a description of the functionality of the button will be displayed.

What can go wrong:

1. Inability to recognize alignment due to interior modifications or accessories.
2. Car might be of a different category or different region, so buttons layout can be a different from the standardized user manual provided by the manufacturer.

Scenario 4: A User started a procedure and is following it

Initial assumption: User has already selected their car and initiated adding windshield fluid procedure.

Normal: The car owner is presented with steps that are illustrated in Augmented Reality. Steps such as the following:

1. Obtain a bottle of windshield wiper fluid. Make sure to use the correct type of fluid for your car. Most cars use a fluid that is a mix of water and a cleaning agent.
2. Open the hood of the car. Locate the hood release latch, which is usually located under the dashboard on the driver's side of the car. Pull the latch to release the hood.
3. Lift the hood and prop it open. There is usually a metal rod located at the front of the car that can be used to hold the hood open.
4. Find the windshield wiper fluid reservoir. It is usually a translucent plastic container with a fill line marked on it. The cap may have an icon that describes windshield wiper fluid.
5. Check the level of the fluid in the reservoir. If the fluid level is low, proceed to the next step. If the fluid level is at or above the "full" mark on the reservoir, you do not need to add more fluid at this time.
6. Remove the cap from the reservoir. The cap may be difficult to remove, so use caution when attempting to open it.
7. Pour the new fluid into the reservoir. Make sure not to overfill the reservoir. If the fluid level is too high, it may spill out when the car is in motion.
8. Replace the cap on the reservoir. Make sure the cap is securely fastened to prevent any spills or leaks.

9. Close the hood of the car. Make sure the hood is fully closed and latched before driving the car.

What can go wrong:

1. A fault at the execution of steps is detected.
2. Car components not correctly detected due to harsh lighting conditions.

Scenario 5: User seeking a technician live call

Initial assumption: The user already selected their vehicle and is in front of it.

Normal: The user clicks on the support icon, and clicks on "Connect to a live technician". The call is forwarded to a technician that is logged in the system and ready to connect to a car owner. Once a technician accepts the call, a connection is established and the car owner's camera starts and will be asked to point at the car. The technician can now see the car owner's view and can annotate it in AR to guide the car owner.

What can go wrong:

1. Live call is laggy or interrupted due to an unstable internet connection.
2. No live agents available at the time of the call.

10 Project Plan

A Gantt Chart of The Project Plan is sectioned below in Figures 15, 16, 17, 18, and 19

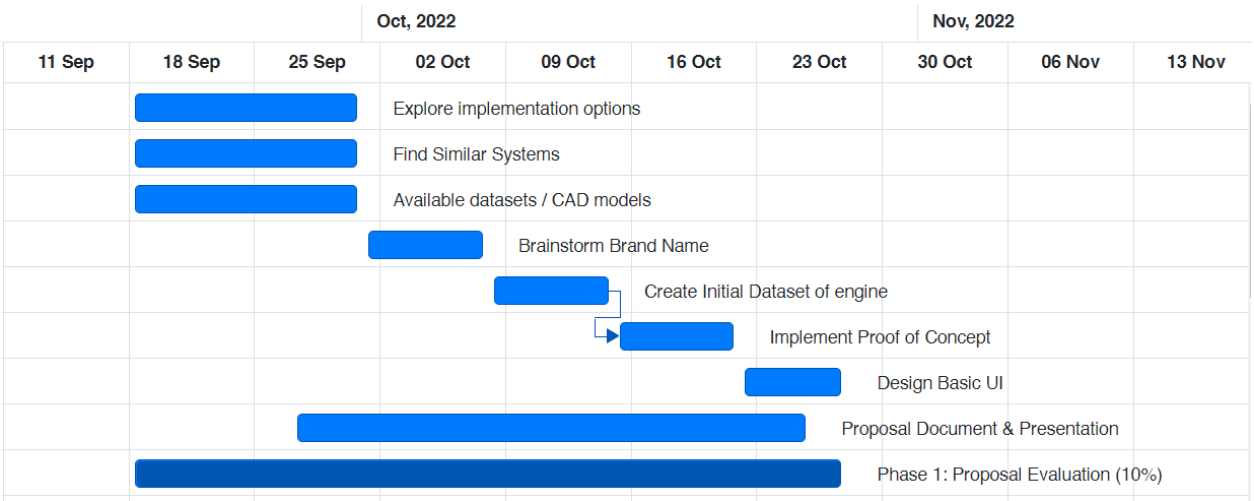


Figure 15: First Phase of Project Plan Gantt Chart

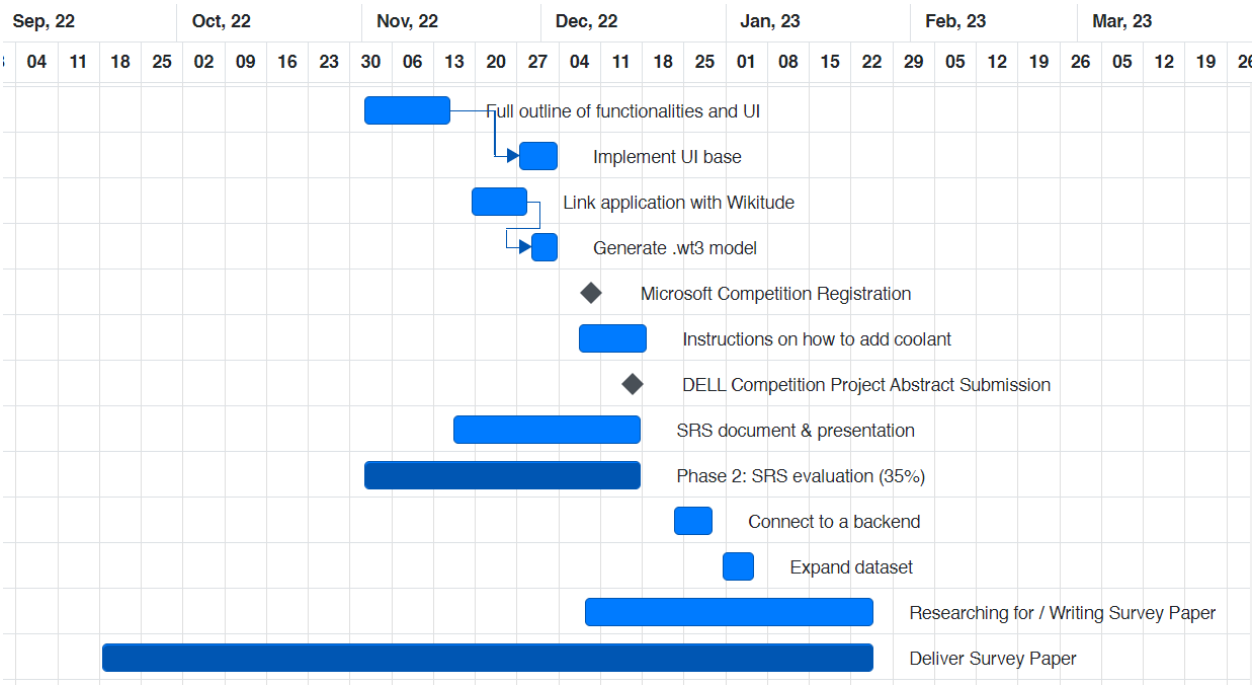


Figure 16: Second Phase of Project Plan Gantt Chart

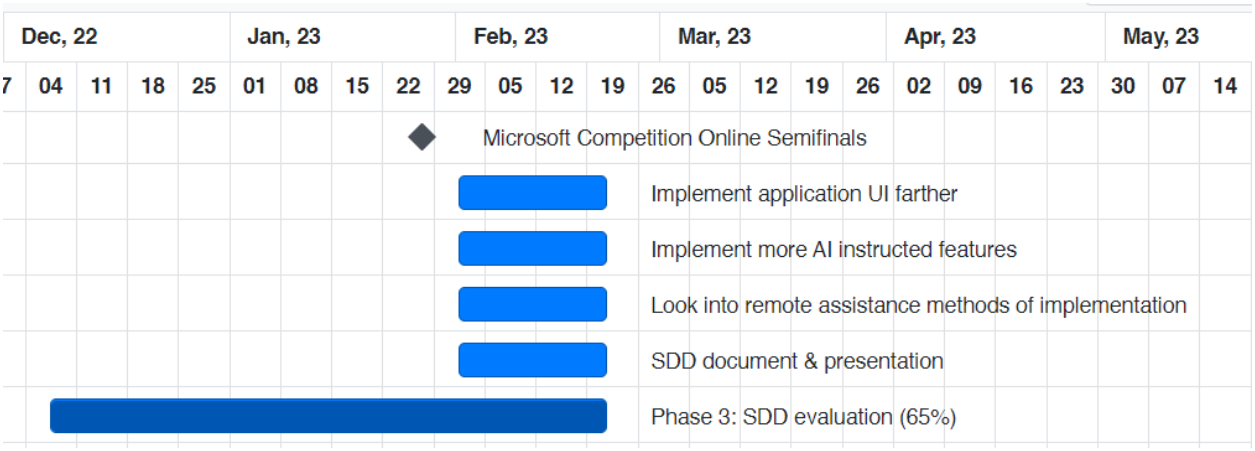


Figure 17: Third Part of Project Plan Gantt Chart

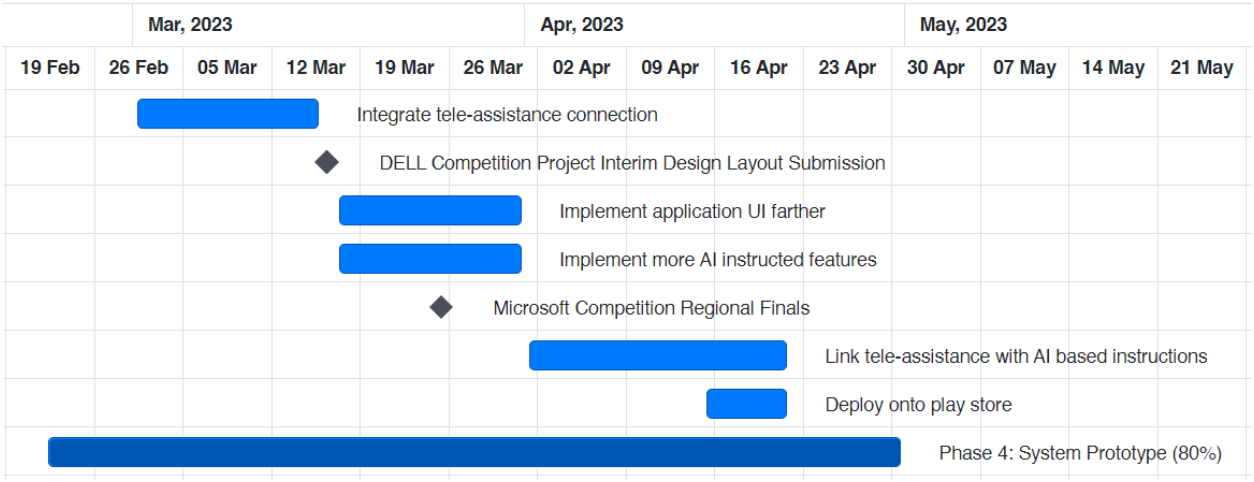


Figure 18: Fourth Part of Project Plan Gantt Chart

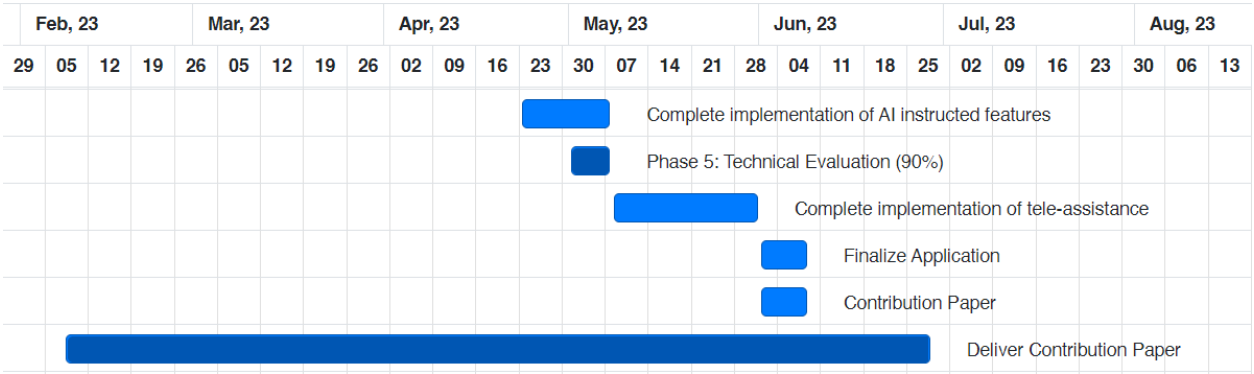


Figure 19: Fifth Part of Project Plan Gantt Chart

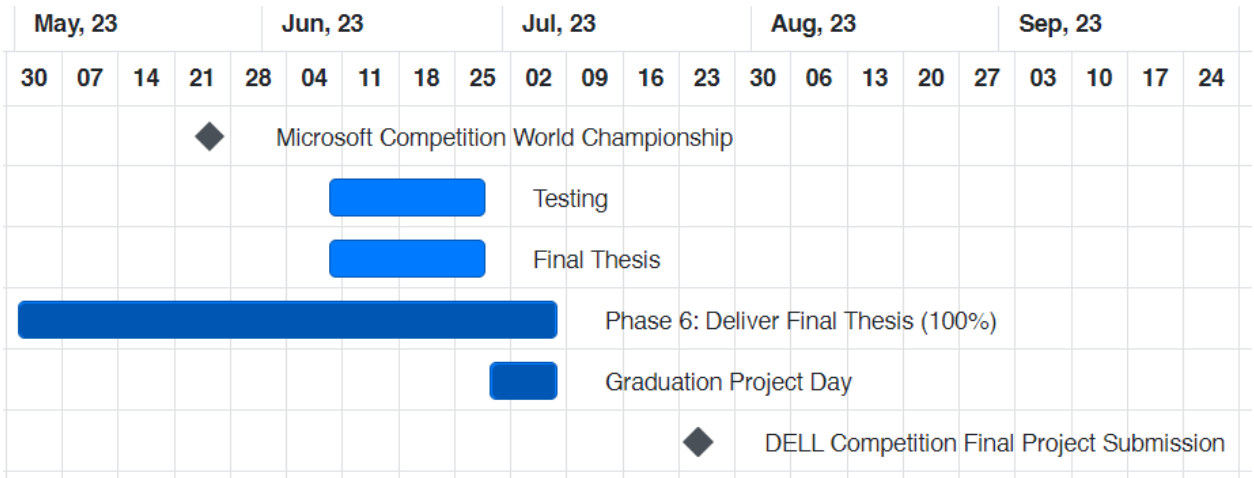


Figure 20: Sixth Part of Project Plan Gantt Chart

11 Appendices

11.1 Definitions, Acronyms, Abbreviations

Table 7: Appendices

AGM	Abou Ghaly Motors
API	Application Programming Interface
AR	Augmented Reality
CAP	Common Augmented Reality Platform
CV	Computer Vision
EERD	Enhanced Entity–Relationship Diagram
HMD	Head-Mounted Display
ML	Machine Learning
SRS	Software Requirement Specification
UI	User Interface
UX	User Experience
WTO	Wikitude Target Object recognition file format
WT3	Wikitude Target 3D Model format for fast loading on low-performance devices
3D	Three Dimensions

11.2 Supportive Documents

11.2.1 Abou Ghaly Motors Collaboration

Figures 21 and 22 represent a sample from the exchanges of the team with Abou Ghaly Motors (AGM)'s team.

Tamer Kotb <tamerkotb@aboughalymotors.com> Mon, Dec 5, 8:18 AM (5 days ago)

to me, AGM, Essam

Hi Judy

Thank you for your approach

My name is Tamer Kotb
I am the COO of AGM group

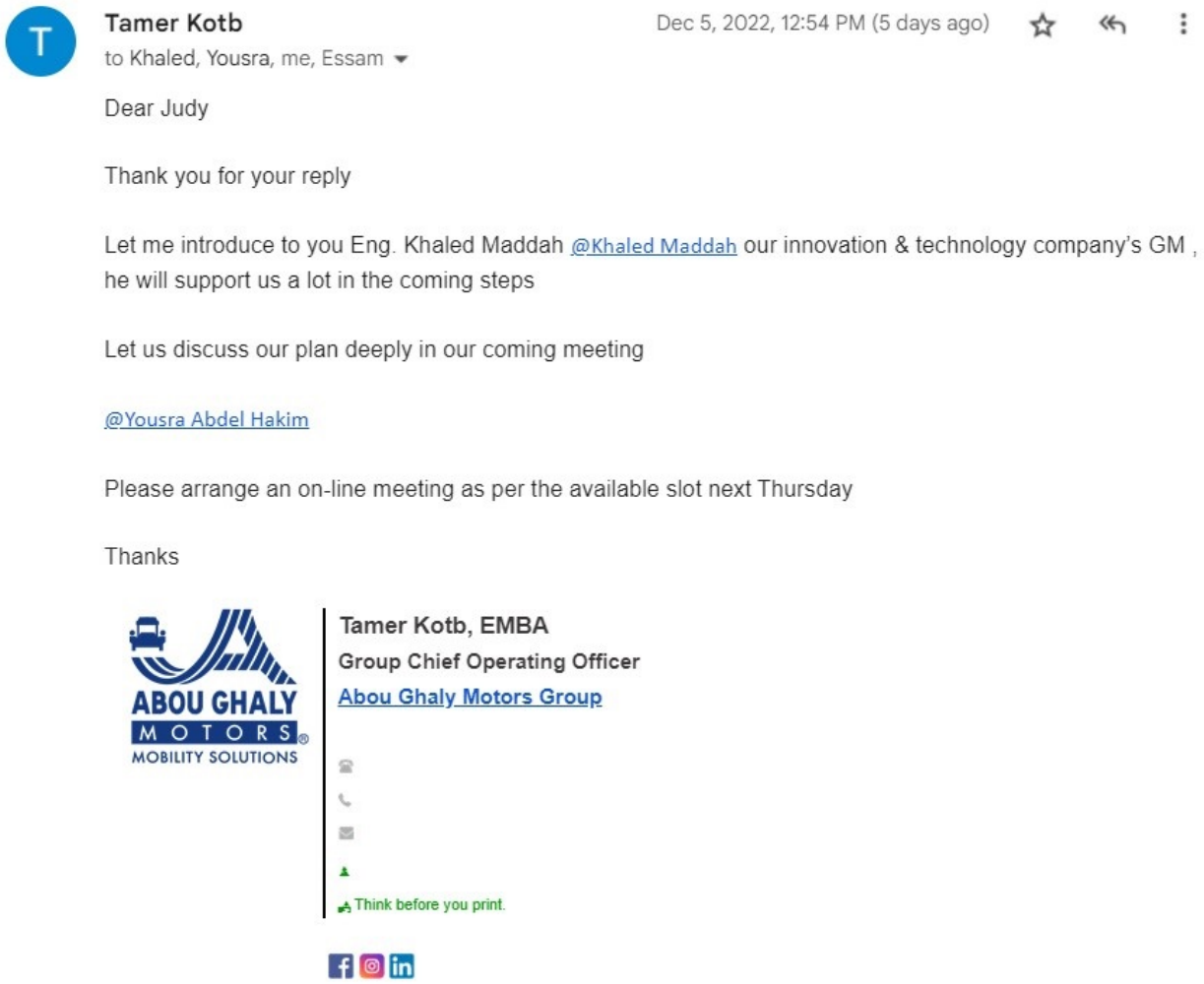
Please make a contact with me on my number today at 11:30 if it is OK with you to discuss your below brilliant idea and see how i can help

Wish to you the best of success

Thanks

Tamer Kotb

Figure 21: AGM email 1



Tamer Kotb Dec 5, 2022, 12:54 PM (5 days ago)

to Khaled, Yousra, me, Essam

Dear Judy

Thank you for your reply

Let me introduce to you Eng. Khaled Maddah @Khaled Maddah our innovation & technology company's GM , he will support us a lot in the coming steps

Let us discuss our plan deeply in our coming meeting

@Yousra Abdel Hakim

Please arrange an on-line meeting as per the available slot next Thursday

Thanks

ABOU GHALY MOTORS MOBILITY SOLUTIONS

Tamer Kotb, EMBA

Group Chief Operating Officer

Abou Ghaly Motors Group

Think before you print.

Figure 22: AGM email 2

11.2.2 Wikitude Edu License

The team members successfully applied and received the Wikitude Edu License valid for a year, as demonstrated in the email in figure 23.

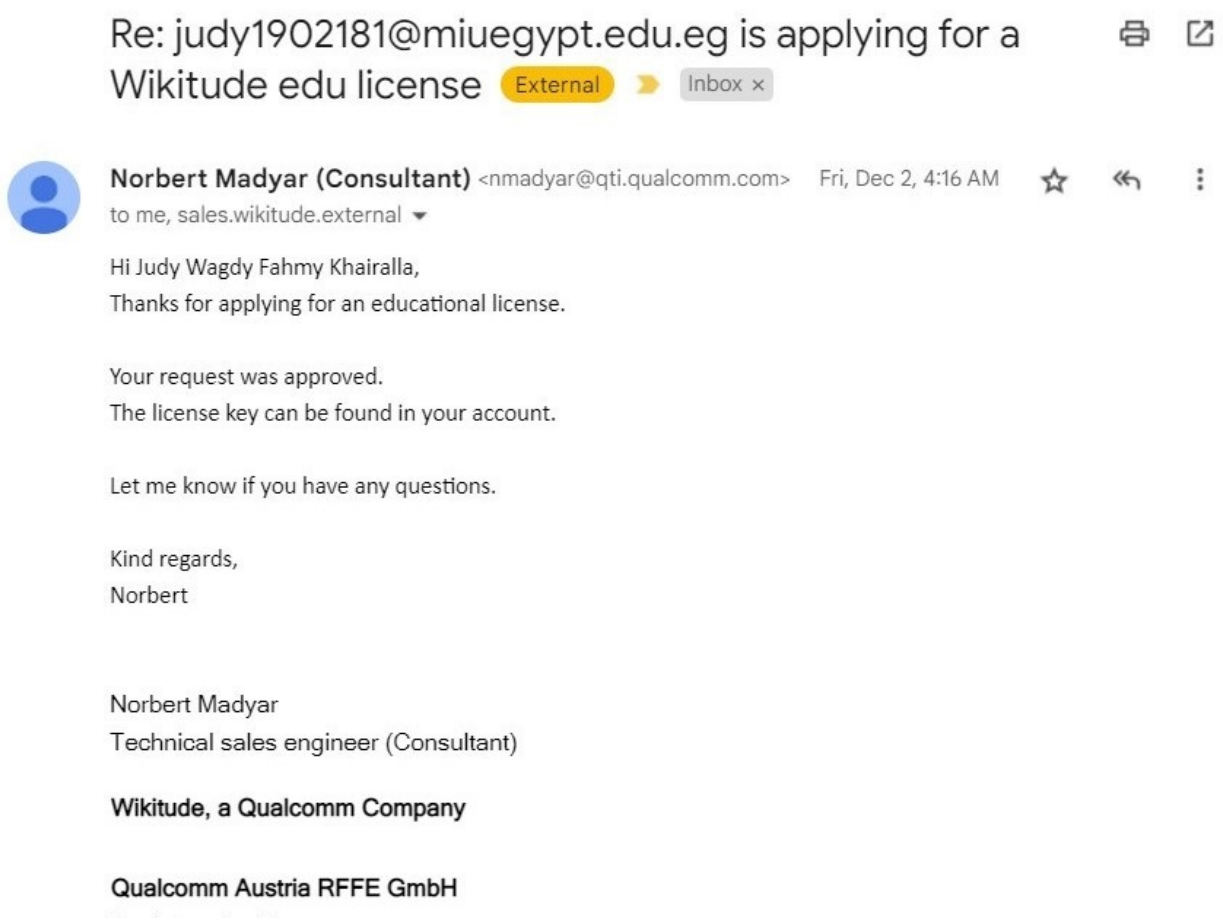


Figure 23: Wikitude Edu License Email

References

[1] *U.S. new car sales to rise in 2022*. URL: https://www.reuters.com/markets/europe/us-new-car-sales-rise-2022-pent-up-demand-edmunds-2021-12-16/#:~:text=Data%20firm%20IHS%20Markit%20on,and%2028.99%20million%20in%202024..

[2] In: *Augmented reality will become vital in car sales by 2021* (). URL: https://cardealermagazine.co.uk/publish/augmented-reality-will-become-vital-car-sales-2021/129175.

[3] Azreen Zuriathy Halim. "Applications of Augmented Reality For Inspection and Maintenance Process in Automotive Industry". In: Feb. 2018. DOI: 10.4314/jfas.v10i3s.35.

[4] Ana Malta, Mateus Mendes, and Torres Farinha. "Augmented Reality Maintenance Assistant Using YOLOv5". In: *Applied Sciences* 11.11 (2021). ISSN: 2076-3417. DOI: 10.3390/app11114758. URL: https://www.mdpi.com/2076-3417/11/11/4758.

[5] Fotios K. Konstantinidis, Ioannis Kansizoglou, Nicholas Santavas, et al. “MARMA: A Mobile Augmented Reality Maintenance Assistant for Fast-Track Repair Procedures in the Context of Industry 4.0”. In: *Machines* 8.4 (2020). ISSN: 2075-1702. DOI: 10.3390/machines8040088. URL: https://www.mdpi.com/2075-1702/8/4/88.

[6] Ana Cachada, David Costa, Hasmik Badikyan, et al. “Using AR Interfaces to Support Industrial Maintenance Procedures”. In: *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*. Vol. 1. 2019, pp. 3795–3800. DOI: 10.1109/IECON.2019.8927815.

[7] *Worxar - Tradiebot - immersive technologies*. Dec. 2021. URL: https://tradiebot.com/portfolio/worxar/.

[8] Re’flekt. *Augmented reality software for maintenance, operations and training*. URL: https://www.re-flekt.com/reflekt-one.

[9] Re’flekt. *Bosch common AR platform*. URL: https://www.re-flekt.com/portfolio-item/bosch-common-ar-platform?hsLang=en.

[10] *Documentation*. URL: https://www.wikitude.com/external/doc/documentation/9.10/flutter/supporteddevices.html#supported-devices.