

# Queen Alia International Airport.

## Software Design Document

Travel Assist  
*Revision 7.0*

*Date: 27/01/2023*

## Revision History

Version	Name	Reason For Changes	Date
1.0	Kareen Ziadat	<i>Initial Revision</i>	22/11/2023
2.1	Kareen Ziadat	<i>Added system analysis, wireframes, and sitemap. Modified list of abbreviation, functional and non-functional requirements.</i>	27/11/2023
2.2	Kareen Ziadat	<i>Switched from mobile application to website. Modified requirements, system analysis, sitemap, and wireframes.</i>	4/12/2023
3.0	Kareen Ziadat	<i>Created mockups, ERD, and justification.</i>	7/12/2023
3.1	Kareen Ziadat	<i>Revision of whole SDD.</i>	9/12/2023
3.2	Kareen ziadat	<i>Tweaking changes on SDD including stakeholders and actors</i>	26/12/2023
4.0	Kareen Ziadat	<i>Overall development</i>	2/1/2024
5.0	Kareen Ziadat	<i>Testing</i>	7/1/2024
6.0	Kareen Ziadat	<i>Implementation and support.</i>	15/1/2024
7.0	Kareen Ziadat	<i>Final revision</i>	27/1/2024

## Approved By

*Approvals should be obtained for project manager, and all developers working on the project.*

Name	Signature	Department	Date
Kareen Ziadat		BP-IT-Development	

## Contents

Revision History .....	2
Approved By .....	2
1. Introduction .....	6
1.1 Overview .....	6
1.2 Purpose .....	6
1.3 Project Scope.....	7
1.4 Target Audience .....	7
1.5 List of Abbreviation .....	8
2. Overall Description.....	9
2.1 Problem Statement .....	9
2.2 Proposed Solution .....	10
2.2.1 Project Idea.....	10
2.2.2 Stakeholders.....	10
2.2.3 Actors.....	11
2.3 Areas of Risk.....	12
2.4 Product Overview.....	13
2.4.1 Functional requirement.....	13
2.4.2 Nonfunctional Requirement (Quality Attributes).....	16
1. Reliability.....	16
2. Availability .....	16
3. Security .....	16
4. Portability.....	16
5. System constraint .....	16
2.5 System Analysis .....	17
2.5.1 Class Diagram.....	17
2.5.2 Activity Diagram .....	19
3. Overall Design.....	20
3.1 High-Level Design (HLD) .....	20
3.1.1 Sitemap .....	20
3.1.2 Wireframes .....	21
3.1.3 Mockups .....	33
3.1.4 ERD .....	47
4. Overall Development.....	48
4.1 Development Overview.....	48
4.1.2 Development Techniques Comparison.....	49
4.1.3 Development Tools.....	50

4.1.4	The preferable Development Tool and Technique .....	51
4.2	Software Development Methodology .....	52
4.3	Overall Design and Development Justification.....	53
4.4	Development Plan Presentation .....	59
4.4.1	Peer review feed back.....	59
4.4.2	Interpret the peer review feedback. ....	59
4.4.3	Identify and evaluate new opportunities.....	59
4.5	Application Development .....	61
4.5.1	Database .....	61
	4.5.2 Recording of application .....	65
5.	Testing .....	66
5.1	Testing plan .....	66
5.2	Performance review.....	69
5.3	Overall Business Application review.....	70
5.3.1	Critical review for all application development phases.....	70
5.3.2	Application evaluation and future development.....	75
6.	Implementation and support .....	77
	References.....	80

## Table of Tables

Table 1: evaluation of functional requirements. ....	15
Table 2: Advantages and disadvantages of each development tool.....	50
Table 3: Test cases plan table .....	68

## Table of Figures

Figure 1: Class diagram for Travel Assist system.	17
Figure 2: Activity diagram for requesting a service functionality (#4).	19
Figure 3: Sitemap showing the hierarchy of Travel Assist application.	20
Figure 4: wireframe 1	21
Figure 5: Wireframe 2	21
Figure 6: wireframe 3	22
Figure 7: wireframe 4	22
Figure 8: wireframe 5	23
Figure 9: wireframe 6	23
Figure 10: wireframe 7	24
Figure 11: Contact us page after peer review.	25
Figure 12: wireframe 9	26
Figure 13: wireframe 10	26
Figure 14: wireframe 11	27
Figure 15: wireframe 12	28
Figure 16: wireframe 13	29
Figure 17: wireframe 14	30
Figure 18: service request form after peer review	31
Figure 19: wireframe 16	32
Figure 20: Mockup 1	33
Figure 21: Mockup 2	33
Figure 22: Mockup 3	34
Figure 23: Mockup 4	34
Figure 24: Mockup 5	35
Figure 25: Mockup 6	36
Figure 26: Mockup 7	37
Figure 27: Mockup 8	38
Figure 28: Contact page after peer review	39
Figure 29: Mockup 10	40
Figure 30: Mockup 11	40
Figure 31: Mockup 12	40
Figure 32: Mockup 13	41
Figure 33: Mockup 14	42
Figure 34: Mockup 15	43
Figure 35:Service request form after peer review	44
Figure 36: Mockup 17	45
Figure 38: Mockup 19 showing home screen on mobiles part 2.	46
Figure 37: Mockup 18 showing the home screen on mobiles.	46
Figure 39: ERD of travel assist	47
Figure 40: Users database	61
Figure 41: Structure of database for service appointments	62
Figure 42: Table to view animals	63
Figure 43: When view is pressed	63
Figure44 : Database for one-on-one appointments	64

## 1. Introduction

### 1.1 Overview

This software design document (SDD) is a formal structure that specifies a software system's design, structure, and behavior. It gives developers and stakeholders a clear understanding of how software is developed, including components, data structures, and communication. Moreover, it functions as a development and maintenance framework.

This document is prepared for Queen Alia International Airport. QAIA is Jordan's primary airport, located 30 kilometers south of the capital city of Amman. It is named after Queen Alia, who died in a helicopter crash in 1977. The airport underwent a large extension project, which was finished in 2013. It provides a diverse choice of internal, regional, and international flights run by several airlines, linking Jordan to places all over the world.

### 1.2 Purpose

This software design document describes the architecture and system design of Travel Assist and acts as an important communication tool, enabling clear and effective collaboration among the development team, stakeholders, and investors. It keeps all parties involved up to date on the project's progress, reducing the risk of miscommunication and associated costs such as time and resources. The document describes the system's design and structure in detail, including components, data flow diagrams, interfaces, UML diagrams specific to the project's specific requirements and nature. The SDD supports the formulation of an accurate timeline, giving sufficient time for each activity, and ensuring timely project delivery by specifying essential project milestones. Moreover, the SDD's early definition of the software's architecture and design enables the identification of potential flaws, allowing for the development of strong mitigation plans. The SDD also addresses compliance with industry standards and security needs as needed. The document distinguishes between functional and non-functional requirements, which improves client knowledge of the software's expected capabilities and simplifies the design and development process by providing the development team with a common knowledge of tasks and objectives. Importantly, the SDD simplifies system maintenance, especially when migrating from the development team to the maintenance team, by eliminating the need to analyze existing software. It also contains a history revision table to track changes in requirements over time.

The intended audience of this SDD are primarily members of the development team (software architects, designers, developers, and testers), project managers who use the SDD to ensure that the project is on track and that the development is in line with the project's goals and timetables. Moreover, it is prepared for Queen Alia International Airport as it explains the software's functional and non-functional needs, giving insight into what to expect from the final product, ensuring satisfaction and alignment with end user requirements.

This document is written to be understood by individuals with diverse backgrounds and varying levels of technical expertise, functioning as a comprehensive reference guide.

### 1.3 Project Scope

The current system involves reaching out to airline's offices, travel agents, or contacting airport personnel upon arrival for special needs assistance and depending on the case, the latter might not work as most services provided require a (24-72)-hour notice in advance. Therefore, this software aims to simplify and streamline the process by offering a dedicated website.

The primary purpose of the Travel Assist program is to potentially eliminate the need for multiple points of contact and provide passengers with special needs with a simple and efficient platform to request the special needs assistance services provided by the airport. These services include wheelchair assistance, the presence of dedicated staff to guide passengers seamlessly through their airport journey, communication support and assistance for passengers traveling with service animals.

The Travel Assist software's goal is to become the go-to platform for passengers with special needs, decreasing the confusion, time and effort required to request assistance and setting a new standard for accessibility and ease in air travel.

This software aligns with QAIA's corporate aims and mission to offering accessible and inclusive air travel experiences for all passengers, regardless of ability. The program intends to simplify and streamline the process of seeking special needs help by providing a specialized website decreasing the time and effort required for passengers to receive the assistance they require.

### 1.4 Target Audience

The rest of this document is divided into sections to thoroughly detail the project. The document starts with an Overall Description that encompasses the core of the project, beginning with the problem statement and continuing with the proposed solution, which investigates the project idea, identifies stakeholders, and outlines the many actors involved. This is followed with the areas of risk section which evaluates potential difficulties and their priority. The paper then identifies the functional and nonfunctional requirements, with a particular emphasis on reliability, availability, security, portability, and system constraints. The document also provides visual aids like class diagram and activity diagram to offer a complete examination of the system's structure and functionality. Lastly, the design of the system is specified through a sitemap, wireframes, mockups, and ERD.

#### **Intended audience:**

**Developers:** Write code to create and manage the application while assuring its functionality and performance. Begin by understanding the problem and the solution to get an overall idea. One of the most important sections is the functional and nonfunctional requirements section, which specifies the system's specific requirements. It gives you a clear picture of what must be coded to ensure system functionality and performance. After understanding the requirements, you will need to view the design section to meet the system's design (sitemap for the hierarchy of the pages, wireframes for the layout of the pages and mockups for the

colors and other visual components). Moreover, for a clearer idea on the data and the system structure see the class and activity diagrams.

**Project managers:** Oversee project planning, execution, and delivery, managing resources and stakeholders to meet goals on schedule and within budget. Begin by becoming acquainted with the problem statement and proposed solution sections to comprehend the project concept and its stakeholders. Then, analyze the areas of risk section to identify probable challenges and prioritize them based on cost and impact to form an effective mitigation plan. Knowing the functional and non-functional requirements will help you create a clear timeline ensuring timely project delivery.

**Marketing staff:** They promote the application using various marketing tactics to attract and keep clients. Begin with the proposed solution section, which provides a thorough grasp of the project concept then view additional visual aids like mockups to understand house style and brand's image to market more effectively.

**User:** The person who interacts with the application. View the proposed solution section for insight into what the system wants to achieve and how it will help you. You can also read the functional requirements to see if the system meets your need and expectations.

**Testers:** Evaluate the application's functionality and performance to find problems and ensure quality. The most relevant part would be the functional and nonfunctional requirements section, that provides you with a clear knowledge of what must be tested in terms of system functioning, reliability, security, and so on. You can also see the activity diagram which shows the flow of a functionality and test if it is met.

**Documentation writers:** responsible for writing materials that provide instructions, explanations, and directions for products, services, or procedures. Start with the problem statement, then move onto the proposed solution, and visual aids.

### 1.5 List of Abbreviation

QAIA: Queen Alia International Airport.

SDD: Software Design Document.

SRS: Software Requirements Specification.

ROI: Return on investment.

## 2. Overall Description

### 2.1 Problem Statement

Individuals with special needs often encounter significant challenges when seeking assistance during travel. The current method for obtaining special needs assistance when traveling is not only time-consuming but also complex, frequently requiring aid from many entities and requiring advanced notice, thereby posing the risk of miscommunication. This complexity stands in the way of attaining the overarching goals of inclusivity and accessibility in airports.

#### User requirements:

1. Create a profile with detailed information about their specific need. They will be able to also modify these details and delete their account.
2. Users can add their service animal through the website so that they are allowed to take their animal on board free of charge.
3. Book wheelchair assistance, guidance staff, and communication support.
4. Services that need to be booked a day to two days in advance will be specified clearly in the service description.
5. Users can send an email to ask about specific cases.
6. Or they can book a face-to-face consultation free of charge, through the website if they want to get more reassurance or information.
7. Users should be able to view frequently asked questions.

#### System Requirements:

1. Scalable to handle increasing numbers of users especially during peak travel times.
2. User data should be encrypted to maintain confidentiality.
3. Regular maintenance should be done to enhance features and address issues.
4. Minimal downtime (less than 24 hours).
5. Compliance with the airport's regulations.
6. The system should only allow users that have a valid upcoming ticket to request service.
7. System should only allow the user to register two animals.
8. System should make sure no appointment conflicts.

## 2.2 Proposed Solution

### 2.2.1 Project Idea

After doing some research by talking to a travel agent and checking out different airline websites, I realized that the present method is inefficient and has a lot of flaws. So, I came up with a couple solutions. Initially, I considered developing a mobile app or a website, but after better understanding the needs, I opted on a website. The system requires collecting a large amount of data, which most people prefer to do on a website rather than on a mobile phone. Furthermore, due to the time constraints, the portability of a mobile app isn't required.

Travel Assist is a website that allows users to enter their specific needs and their ticket number and request one of the many services that will make their journey in the airport as smooth as possible. These services include wheelchair assistance, the presence of guidance staff, communication support, and assistance for passengers traveling with service animals. The website has description for all these services but will also offers a place to send email inquiries through a simple form or book a face-to-face consultation through a simple process form (all in contact page).

### 2.2.2 Stakeholders

Stakeholders are individuals or groups with an interest in the Travel Assist application and its development. They may be influenced by the project's outcome or can influence it. Below is a list of the types of the stakeholders involved:

**Customers:** individuals or entities who requested the application.

1. **QAIA:** Jordan's main international airport. They are the customers who asked for the application. It is the physical site where the application will be used, and they will be responsible for providing the infrastructure and resources required to enable the app's operation. They impact the requirements, development and have input on the design of the software.

**Employees:** Individuals who work on developing and implementing the project.

1. **Application developers:** designers, developers, and maintainers of software programs or apps for multiple platforms. They will turn the vision and idea into a real product. They impact the software by deciding how to implement, what the overall design should be, based on their level of experience and understanding for the requirements.

**Investors:** Individuals or organizations who contribute capital sometimes in exchange for a ROI and will financially support the development of the application.

1. **Civil Aviation Regulatory Commission (CARC):** Regulatory authority in charge of regulating civil aviation activities, safety, and adherence to aviation legislation including those involving the transportation of individuals with special needs. They will contribute funds to the application's development, so they will impact scope, and other developmental decisions. (sponsor and donor)

**Governments:** will be enforcing rules and regulations on the application's development.

1. **Civil Aviation Regulatory Commission:** Guarantees that software complies (impact the application) with international norms and regulations, especially those governing the rights and requirements of disabled passengers.

**Suppliers:** organizations that supply necessary services ensuring the smooth operation of the website.

1. **Airline companies:** Corporations that run the flights that passengers will take. They need the information about the service animals, will offer in plane services and will provide the user with guidance personnel. They are affected by the application since they will get the appointments through it and will impose restrictions and details over the services (affect).
2. **Travel agents:** Professionals or organizations that help people plan and schedule their travel arrangements. They may be required to help travelers with disabilities in scheduling air travel and utilizing the application. They are affected by the application since they will no longer rely on the traditional method of requesting a service.
3. **Service providers:** They are staff working at the airport including customer service agents and luggage handlers who will be responsible for assisting passengers who use the program. They will be impacted by the application as they will have appointments booked through it.

**Communities:** Any local groups that may be affected by the website.

1. **Travelers:** People who are travelling. The community will have greater accessibility and inclusivity at airports to help the larger community. They are affected by the application.
2. **Passengers with disabilities:** Individuals who require help due to physical or cognitive limitations while flying. These are the people who will use the Travel Assist application to request assistance while traveling through the airport (are affected). They prefer a user-friendly application that meets their specific requirements (affect).
3. **Travel agents:** They can recommend the website to their clients with special needs and put in a good word improving publicity. (affect)

### 2.2.3 Actors

An actor is a user, system, or external entity that interacts with the system and produces or consumes data. They have an interface. Actors play an important

part in defining a system's functionalities and behaviors since they help identify who or what interacts with the system and what their duties are. There are two types the primary and secondary actors:

**Primary actors:** user who has a clear user goal that the system fulfills.

1. **End users (travelers with disabilities):** Individuals who connect with the Travel Assist website to request services, register their animal and ask questions.

**Supporting Actors:** external objects that provide a service (for example, information) to the system.

1. **External database system:** connect to QAIA database that contains all the flights and ticket numbers for the current and upcoming trips. This will be needed to ensure compliance with system constraint.

## 2.3 Areas of Risk

### Design phase risks:

1. **Inadequate understanding of user requirements:** The application may lack important features, or services, or forget to ask about important user details if the team fails to adequately comprehend the different demands of users with special needs. Conducting interviews with individuals with special needs may present challenges due to the comparatively smaller representation of this demographic. **HIGH**
2. **Security and privacy:** the application incorporates personal and sensitive information regarding their disability, there is a risk of breaches if effective security measures are not prioritized in the design. Failure to handle this issue might have serious repercussions for both users and the company, including legal sanctions and loss of trust. **HIGH**
3. **User Interface:** This should be designed carefully with the user's disability needs in mind. Failing to understand the limitations of the user will cause customers dissatisfaction and frustration. **LOW**

### Development phase risks:

1. **Failure to meet accessibility standards:** Failure to meet the parameters necessary for the application to meet accessibility standards for individuals with disabilities during development may result in legal consequences as well as the exclusion of a substantial user base. **HIGH**
2. **Scalability issue:** If the application's scalability isn't considered during the development phase, the website will be unable to manage the travelers' service request or encounters performance issues during peak times, it will cause less people to rely on the website and more on traditional channels (travel agents, airlines) and potential user loss. **MEDIUM**
3. **Technical Challenges:** Implementing seamless connection with the airport's database to check ticket validity necessitates solid technical skills.

If not developed properly, it will allow people without a valid upcoming ticket to book an appointment (which is not allowed). **HIGH**

4. **Time:** failure to understand user needs due to not having enough people to interview will cause the requirements gathering to be delayed causing delay in the development time. **HIGH**
5. **Hosting tool going out of business.** Since I am relying on hosting on pantheonsite.io, there is a risk that it goes out of business. **LOW**
6. **Any of used plugins is no longer maintained by the supplier.** This poses security, performance, and integration issues with the website. **LOW**

#### Testing phase risks:

1. **Inadequate user acceptance testing (UAT):** It is critical to guarantee that people with different special needs can use the application efficiently otherwise critical flaws may go unreported until the program is deployed. **HIGH**
2. **Not having enough users to test the application properly.** **HIGH**

## 2.4 Product Overview

### 2.4.1 Functional requirement

**Note: anything in bold is a non-optional field.**

1. Users can register and manage their profiles (edit details or delete account) by giving personal information such as their name, contact information, their disability, and their ticket number. Managing the account button will be an icon of a person at the top right icon of the navigation bar. This includes changing all the details specified in the registration form.

**Detailed Requirements:**

1. First collect **username, user's first, last name**.
2. Contact information: **email, phone number**.
3. About yourself: **age, gender, nationality**
4. Trip details: **Passport number, expiration date**, Ticket number
5. Disability Nature: **nature of special needs** (checkbox: mobility impairment, visual impairment, hearing impairment), Mobility requirements (dropdown box: wheelchair, cane, crutches) and medications used (as a text box).
6. Emergency contact: **name, relation, and phone number**.
7. **Password and repeated password. (minimum medium strength and minimum length of 5)**

The user can choose whether they want to have their credential sent to them by email.

2. Add a service animal: Provide a detailed form (all information should be collected on the same screen with description). **Service animal type (drop down list: guide dog, hearing dog, mobility assistance dog, psychiatric service dog), size and weight.** Optional: animal name, description, breed, age, specific tasks the service animal is trained for, emergency contact person and their contact information. Each animal will be added in a separate form. This form can be accessed using a button on the home page, a slide in the slider, and through the guidance service description. The user can only add a service animal if they are logged in otherwise, they are shown an error message.
3. Log into account using the username and password.
4. Service request: Users can request specific services such as wheelchair assistance, communication support, and guidance for visually impaired passengers. The interface should be simple, the website should show all services available with a brief description, a picture and bullet points so that the users can easily understand what is available.  
The user can only request a service if they are logged in otherwise, they are shown an error message.

**Each service detailed Requirements:**

1. Wheelchair assistance: **User Mobility Level (dropdown: user can shift to a seat or requires assistance during the travel), Availability**

- of Wheelchair (dropdown: doesn't need, has their own wheelchair, or requires one provided by the airport), date, time, and ticket.**
2. Communication: **communication method (dropdown: sign language, written communication), date, time, and ticket.** Optional: any additional communication needs (text box) e.g. language preferences or specific communication tools).
  3. Guidance for visually impaired: **level of visual impairment (text box: mild to severe), preferred assistance method (dropdown: verbal instructions, physical assistance), date, time, ticket.**
  5. Email notification System: send a confirmation email for the services. Users can view details of upcoming service details like name of service provider, date, time, details of location, and nature of service and cancel their appointment. This will be within the emails.
  6. Allow sending an email asking a question through an easily accessible form.
  7. User can see frequently asked questions within the contact us page as an accordion.
  8. Users can book a one-to-one consultation through a simple process. The users first choose the **consultation duration, from what date they are available, days they are available, and specific time slots.** Following that, a thorough list of available **time options** is displayed, which the user chooses from. Following that, users provide their personal information, including their **name, phone number, email address**, and any extra notes. While these details are automatically filled in from their account, individuals can modify them if necessary. After that, users will receive a confirmation message and an email verifying their booking.

Functionality	Priority	Benefit	Penalty	Cost	Risk
1	High	7	3	6	4
2	High	8	2	3	3
3	High	7	3	3	4
4	High	9	3	8	6
5	Medium	5	3	4	3
6	Medium	6	4	4	5
7	Low	4	1	2	1
8	Medium	5	3	4	5

*Table 1: evaluation of functional requirements.*

The table evaluates functional needs according to priority, benefit, penalty, cost, and risk (their potential impact on the system). Note that the greater the value, the more significant while the lower the value, the greater the negative impact or risk.

## 2.4.2 Nonfunctional Requirement (Quality Attributes)

### 1. Reliability

The system should perform its intended functions consistently and accurately, with no faults or disturbances. The system should maintain the integrity of user data by performing backups and doing data validation checks on ticket numbers, passport numbers, and so on. The system should make sure that once a user requests a service, the appointment is made. It should make sure that there is no incorrect information about the user, missed bookings or double bookings of a service.

### 2. Availability

The uptime should be 99.99% because the service can be requested at any moment by someone with disabilities and this is a new approach to requesting a service which means that the word of mouth is extremely important, and frustration will cause a hit to the airport's reputation. This is especially important during peak traffic seasons.

### 3. Security

Ensure that user data is securely saved, and that privacy is preserved throughout the program. Since the website is saving critical, confidential information about the passengers disabilities security is important.

- 1) Encryption should be used for user profiles and sensitive information.
- 2) Conduct frequent security audits and penetration testing.

### 4. Portability

Should be responsive on all desktop and mobile device sizes. Since the traveler might want to request a service / register their service animal or view details about the services provided on the go (through mobile phone) or if they are more comfortable inputting information on a desktop.

### 5. System constraint

1. Only allow registration of 2 animals for each user.
2. Only allow users that have a valid upcoming ticket and passport to request service.
3. Failed emails should be logged.
4. Only logged in users can request a service or add their service animal.

## 2.5 System Analysis

### 2.5.1 Class Diagram

A **class diagram** is a visual representation of an object-oriented system's structure and interactions between classes. It displays their attributes, methods, and relationships (associations, inheritance, aggregation, composition, and dependencies). They provide an overview of the system's design, which classes (attributes and methods) and comprehending class relationships. Class diagrams also help system analysts, developers, and stakeholders communicate more effectively by assuring a shared knowledge of the system's structure and behavior. Below is the class diagram of the Travel Assist system (website).

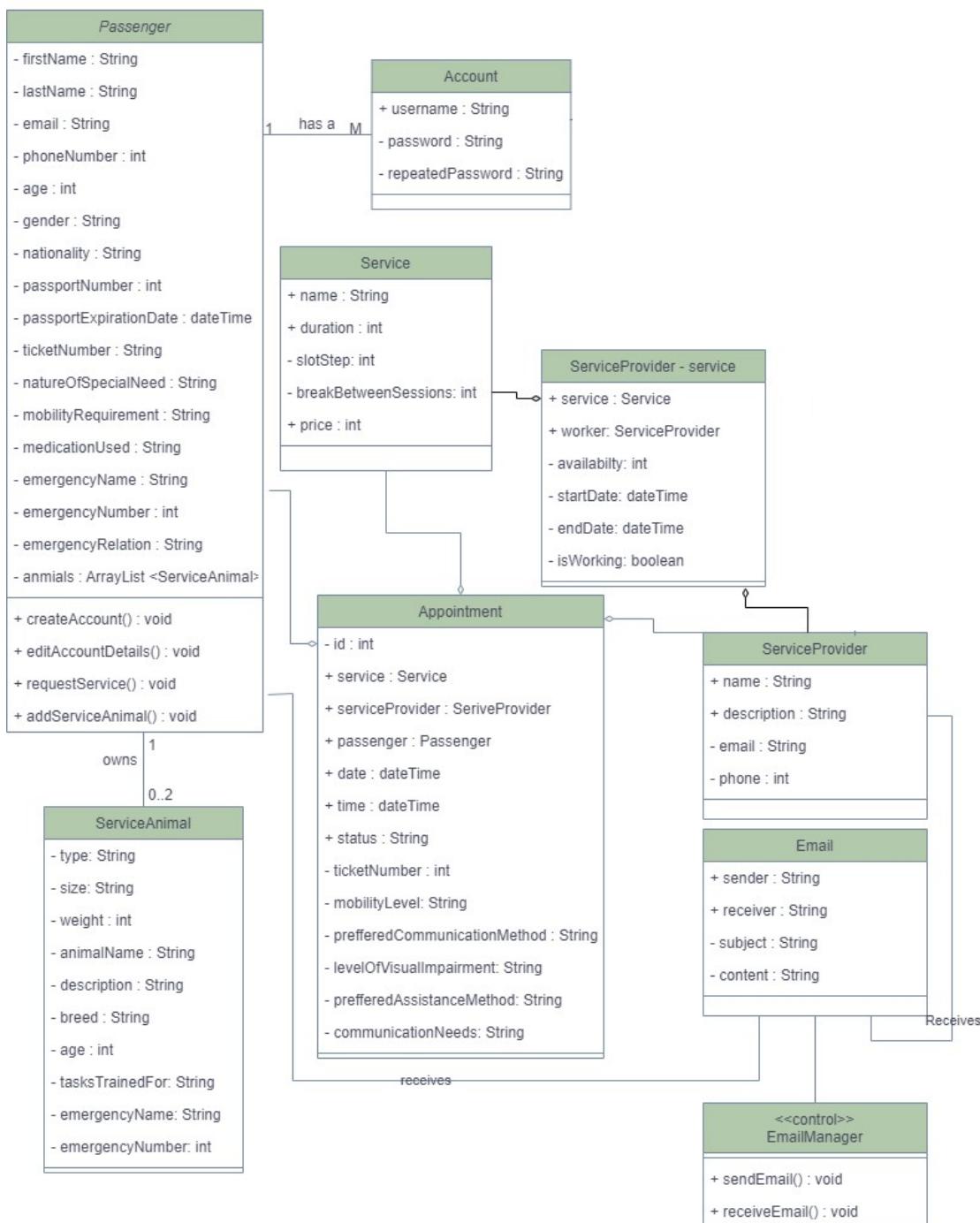


Figure 1: Class diagram for Travel Assist system.

The above class diagram shows the main entities of the Travel Assist website as a system.

1. **Account:**

- a. Passenger: who requests the services and registers their service animals.

- b. Future extension of actors can be done.

2. **Service animal:** multiplicity on this association side is 0..2 meaning that a passenger can have 0, 1 or maximum 2 animals as per nonfunctional requirements.

3. **Service provider:** this is the person who provides the service requests. The admin is responsible of registering the service providers to the system and can modify their details.

4. **Service:** which is a range of all the services offered through the website.

5. **Service-provider** is the mapping of the service provider to the services that they offer. It includes information about their availability which is required for availability.

6. **Email:** that has information of sender, receiver, subject, and content.

7. **Email manager:** the system that sends emails to the service provider notifying them of an upcoming service appointment. (name of passenger, ticket number, type of disability, date, time, and type of service) and the passenger of the pending, cancelled and confirmed appointments with all its details.

8. **Appointment:** has details of the appointment that will be displayed in the emails of the passenger, service provider, and the admin dashboard.

Another structural representation is the **component diagram** which is a visual representation of the structure of the system components or modular parts and the relationships between them. It shows how various components interact and communicate with one another to achieve system functionality. They facilitate design, development, and maintenance of the system by providing a clear picture of the system's architecture and component dependencies. It also ensures effective communication among stakeholders.

### 2.5.2 Activity Diagram

An activity diagram is a visual representation of the flow of activities within a system or process. It displays the chronological order of activities, decision points, and transitions between actions. They aid in visualizing the workflow, emphasizing operation order, and defining decision points and conditions. It can serve in identifying potential areas for improvement within a system and enhance communication between stakeholders, developers, and analysts.

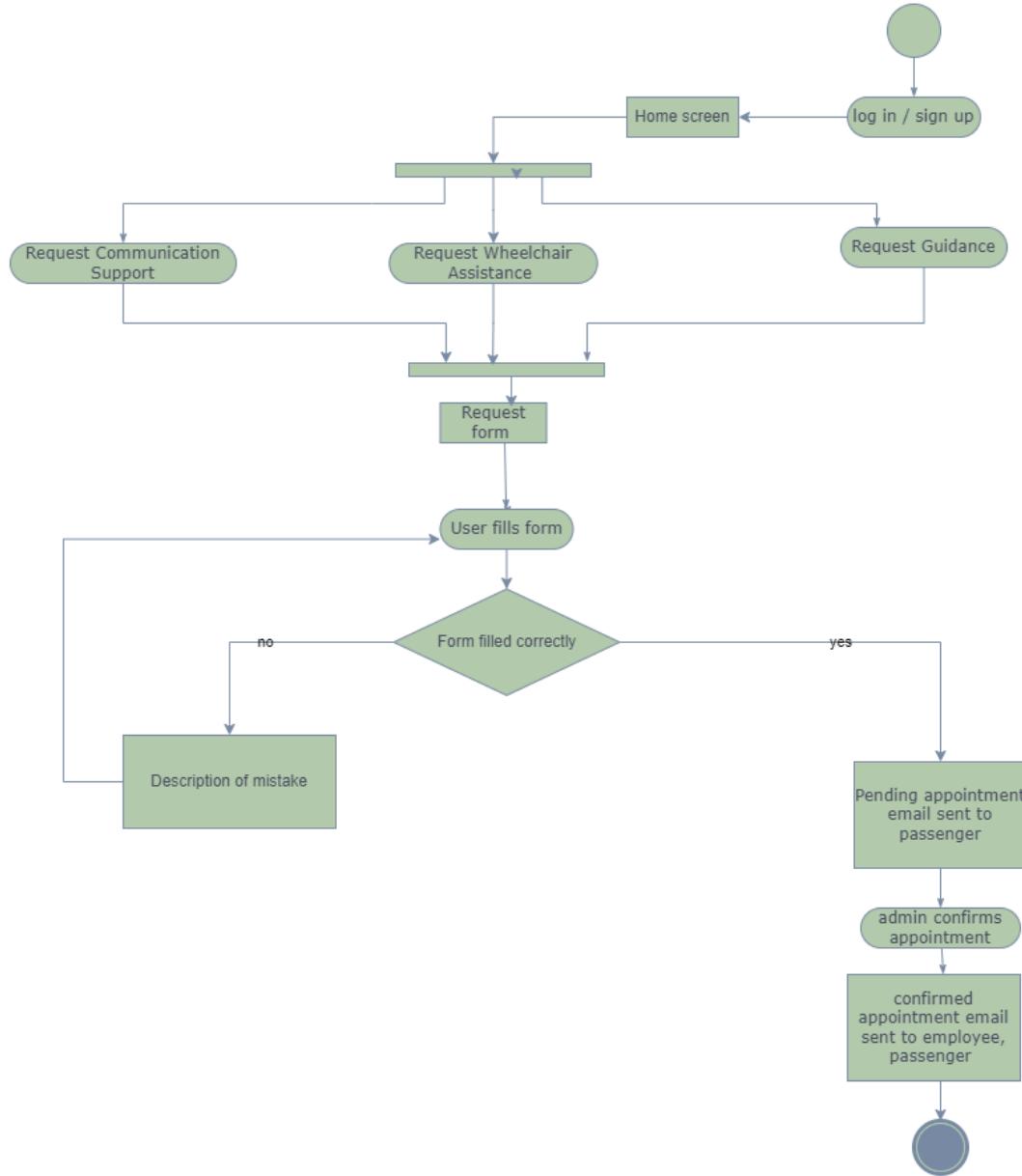


Figure 2: Activity diagram for requesting a service functionality (#4).

The above diagram shows the activity diagram of a user trying to request a service. The user must be logged in, go to home screen. The user has three services to choose from the home screen or the services page. The passenger will then choose the service provider or choose any if they have no preference. Then they will choose the date and time they want and the rest of the needed information and submit. If non optional information were left empty, they will get an error message and allowed to continue filling. If the form filling was done correctly, the service is requested and set as pending and once accepted, they will get a confirmation email.

Another behavioral representation is a use case diagram which is an effective visualization for clarifying the numerous ways in which users or other actors interact with a system. It organizes distinct functionalities into use cases that exemplify the system's intended behavior. Use case diagrams allow smooth communication with stakeholders by demonstrating the interactions between actors and the system, helping in requirements gathering and system design. It also acts as a blueprint, involving scenarios, system goals, and the scope, and representing actors, systems, and goals. Having a well-made use case diagram facilitates system implementation and maintenance.

### 3. Overall Design

#### 3.1 High-Level Design (HLD)

##### 3.1.1 Sitemap

A sitemap diagram is a graphical representation of the application's content structure and hierarchy. It displays the connections between the application's elements, enables developers, designers, and stakeholders to visualize the application's layout, primary navigational paths, and overall user flow. It can help with user experience (UX) design by identifying any missing pages or material that should be added.

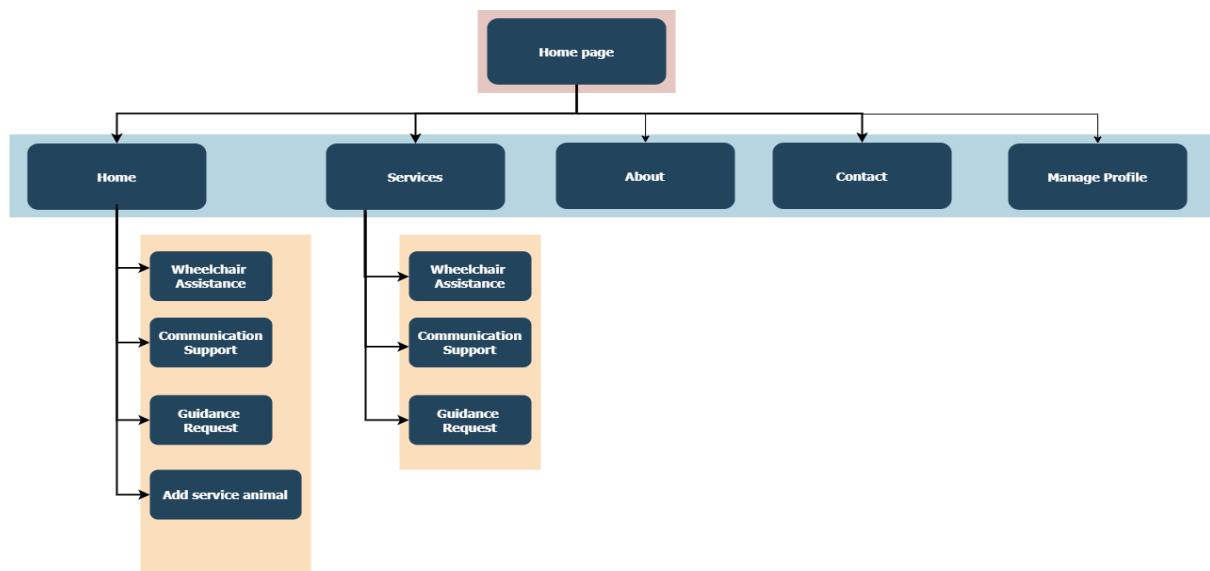


Figure 3: Sitemap showing the hierarchy of Travel Assist application.

### 3.1.2 Wireframes

A wireframe is a graphical representation of the structure of a software application, which focuses on content placement and user interactions. It is established early in the development phase and acts as a template for designers, developers, and stakeholders to collaborate on and change the structure of the application. Wireframes ensure that the interface is user-friendly by considering user needs. It is a simplified representation with no color to make it easy to change before proceeding to the visual design and content execution stages.

Home screen slider:

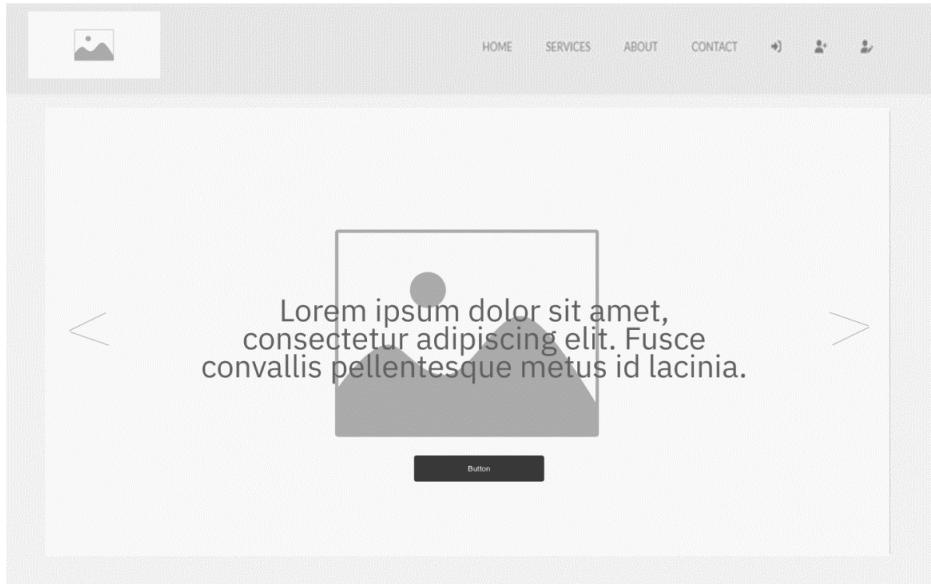


Figure 4: wireframe 1

Home screen part 2:

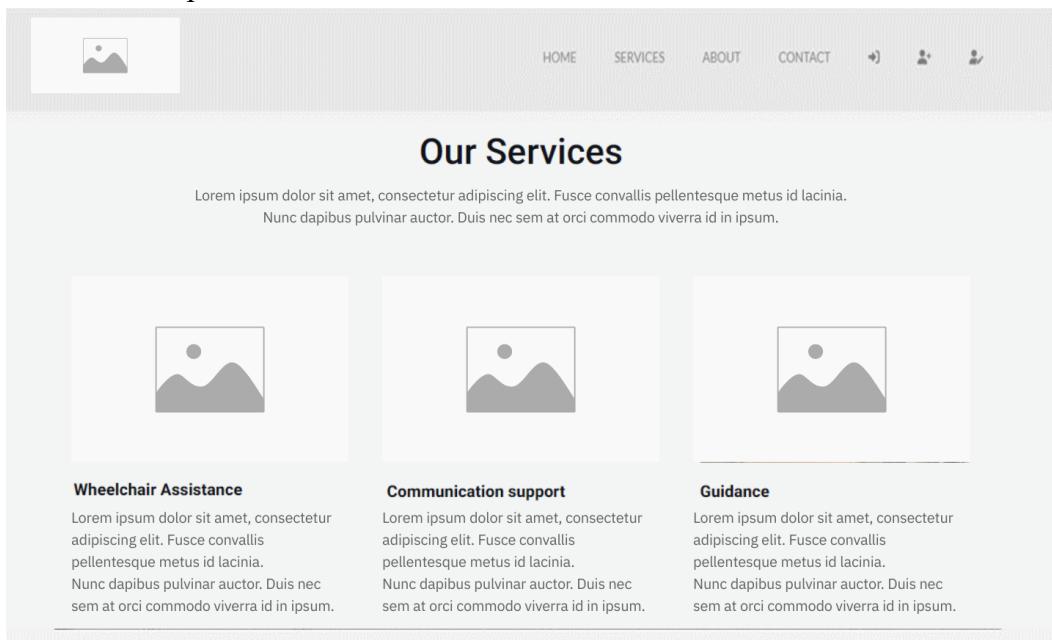


Figure 5: Wireframe 2

Home screen part 3:

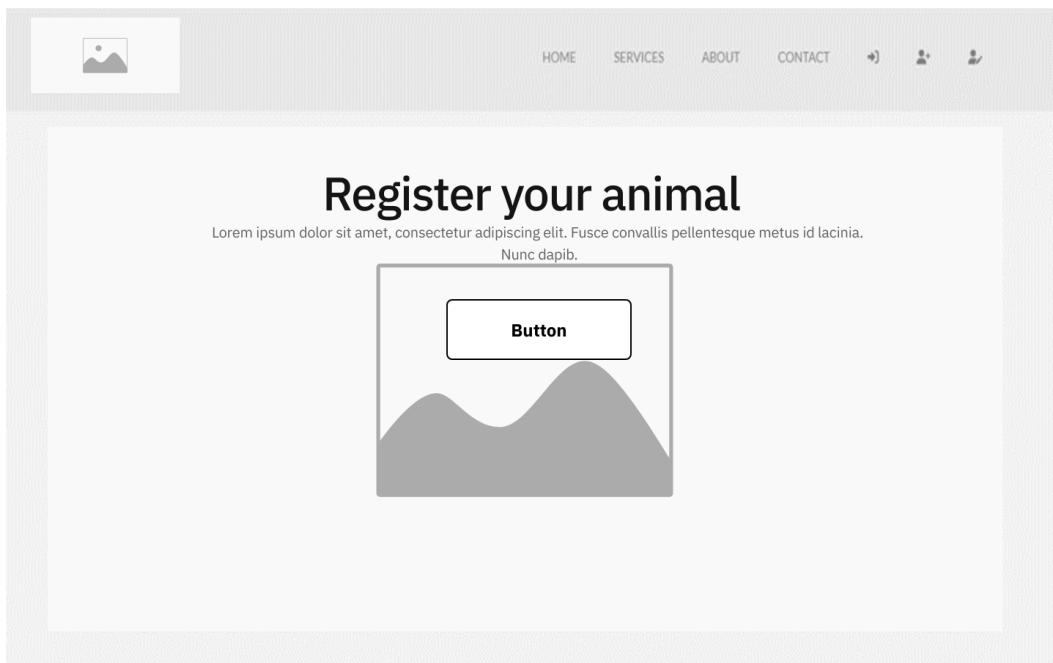


Figure 6: wireframe 3

Footer:

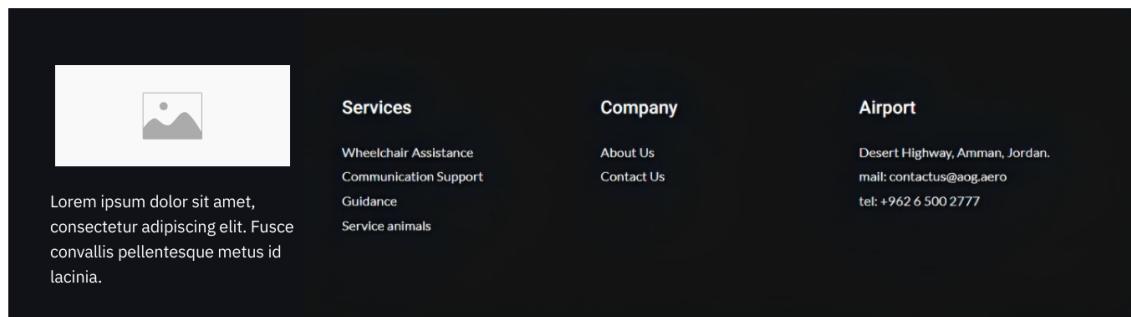


Figure 7: wireframe 4

## Log in pop up:

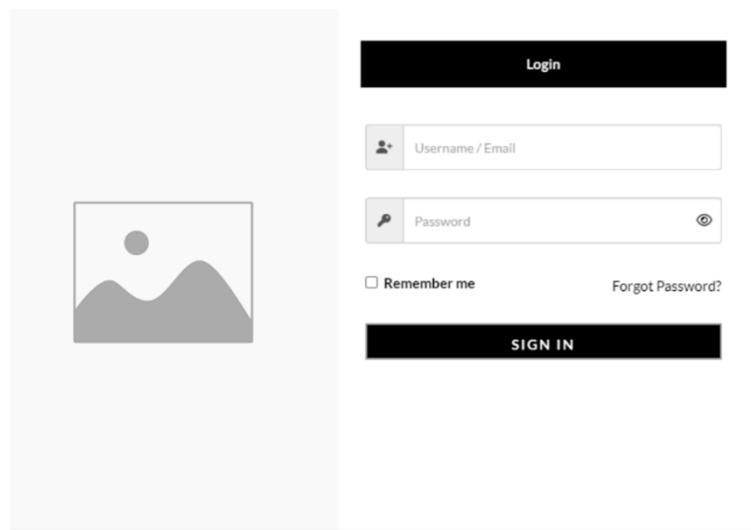


Figure 8: wireframe 5

## Services page:

**Services**

### What we do?

Placeholder text: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce convallis pellentesque metus id lacinia. Nunc dapibus pulvinar auctor. Duis nec sem at orci commodo viverra id in ipsum.



**Wheelchair assistance**

Placeholder text: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce convallis pellentesque metus id lacinia. Nunc dapibus pulvinar auctor. Duis nec sem at orci commodo viverra id in ipsum.



**Communication support**

Placeholder text: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce convallis pellentesque metus id lacinia. Nunc dapibus pulvinar auctor. Duis nec sem at orci commodo viverra id in ipsum.



**Guidance service**

Placeholder text: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce convallis pellentesque metus id lacinia. Nunc dapibus pulvinar auctor. Duis nec sem at orci commodo viverra id in ipsum.

Figure 9: wireframe 6

About us page:

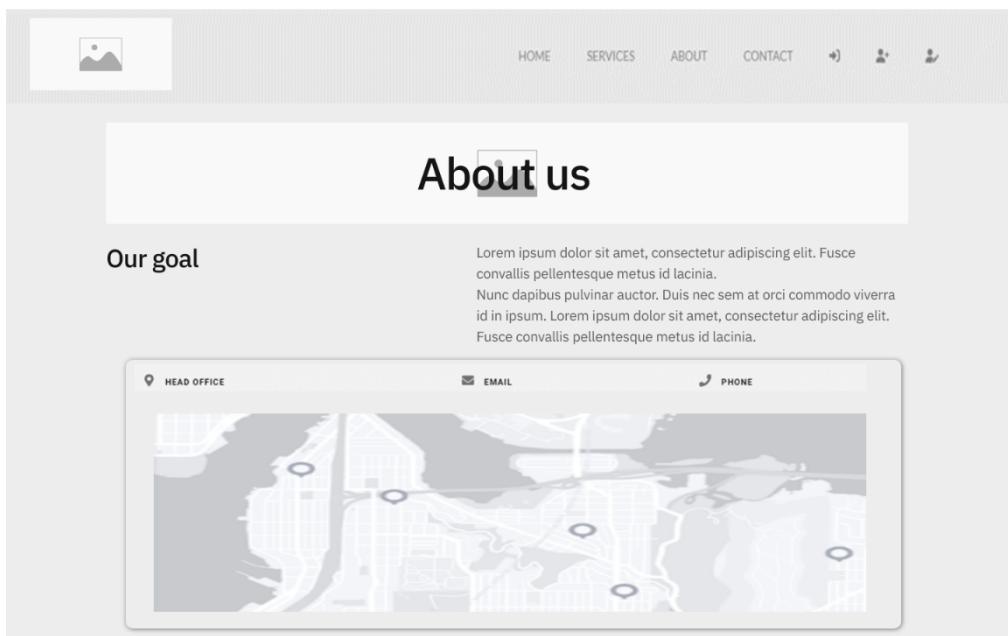
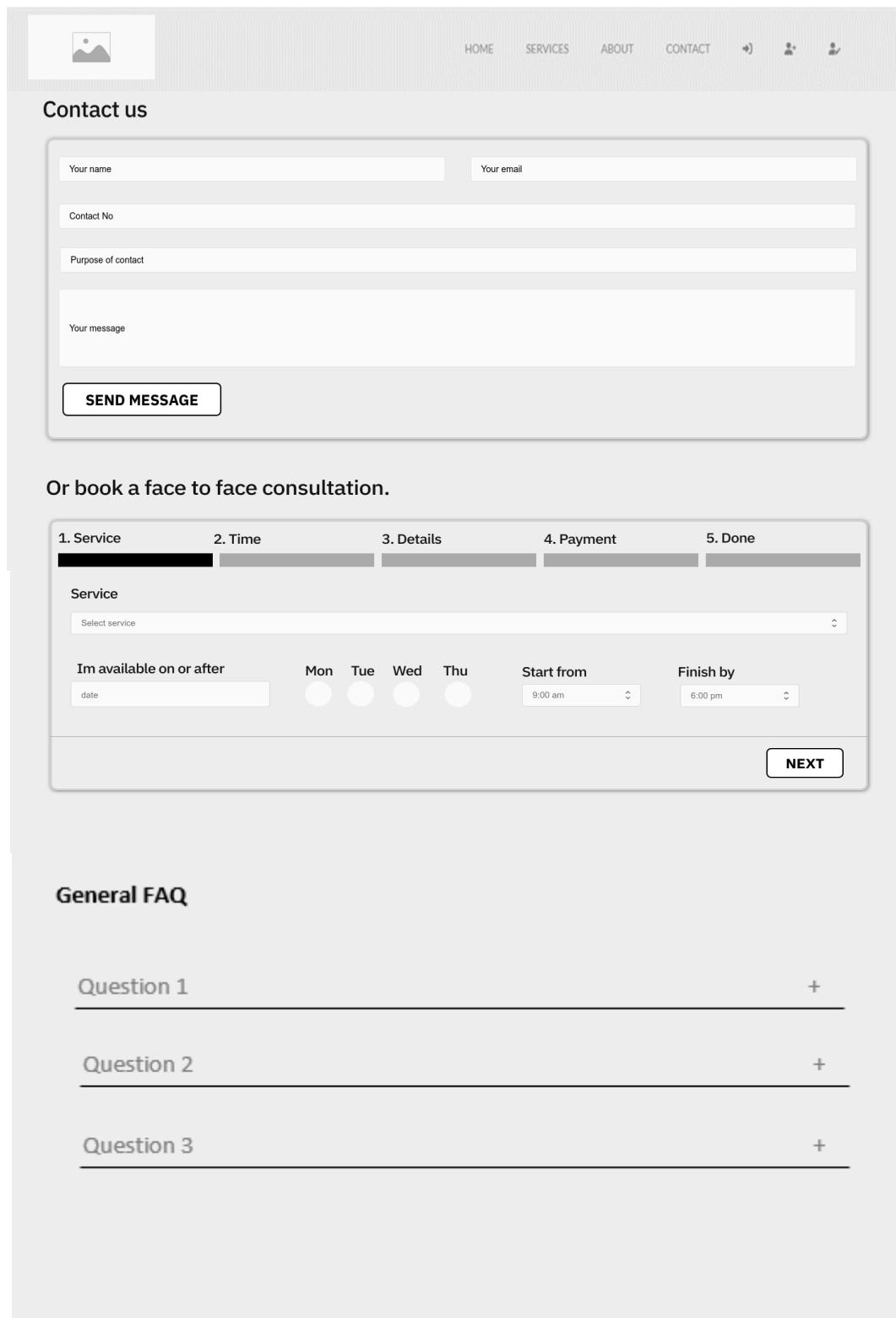


Figure 10: wireframe 7

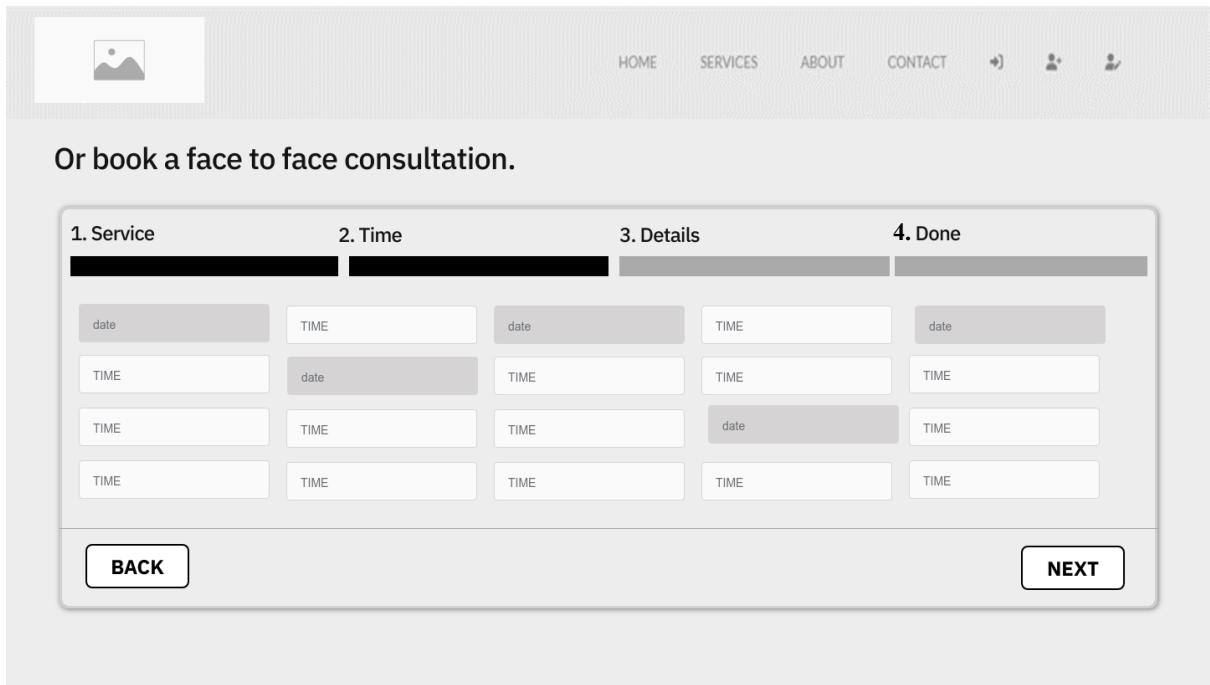
Contact us page:



The screenshot displays the 'Contact us' page of the Travel Assist website. At the top, there is a header with a logo, navigation links for HOME, SERVICES, ABOUT, and CONTACT, and social media icons. Below the header is a large contact form titled 'Contact us' with fields for 'Your name', 'Your email', 'Contact No', 'Purpose of contact', and 'Your message'. A 'SEND MESSAGE' button is located at the bottom of the form. Below the contact form, a section titled 'Or book a face to face consultation.' is shown, featuring a five-step process bar (1. Service, 2. Time, 3. Details, 4. Payment, 5. Done) and a service booking form. The service booking form includes a dropdown for 'Select service', a date input field, a day selector for 'I'm available on or after' (Mon, Tue, Wed, Thu), time inputs for 'Start from' (9:00 am) and 'Finish by' (6:00 pm), and a 'NEXT' button. At the bottom, there is a section titled 'General FAQ' with three expandable questions: 'Question 1', 'Question 2', and 'Question 3', each preceded by a '+' sign.

Figure 11: Contact us page after peer review.

## Book a consultation 2:



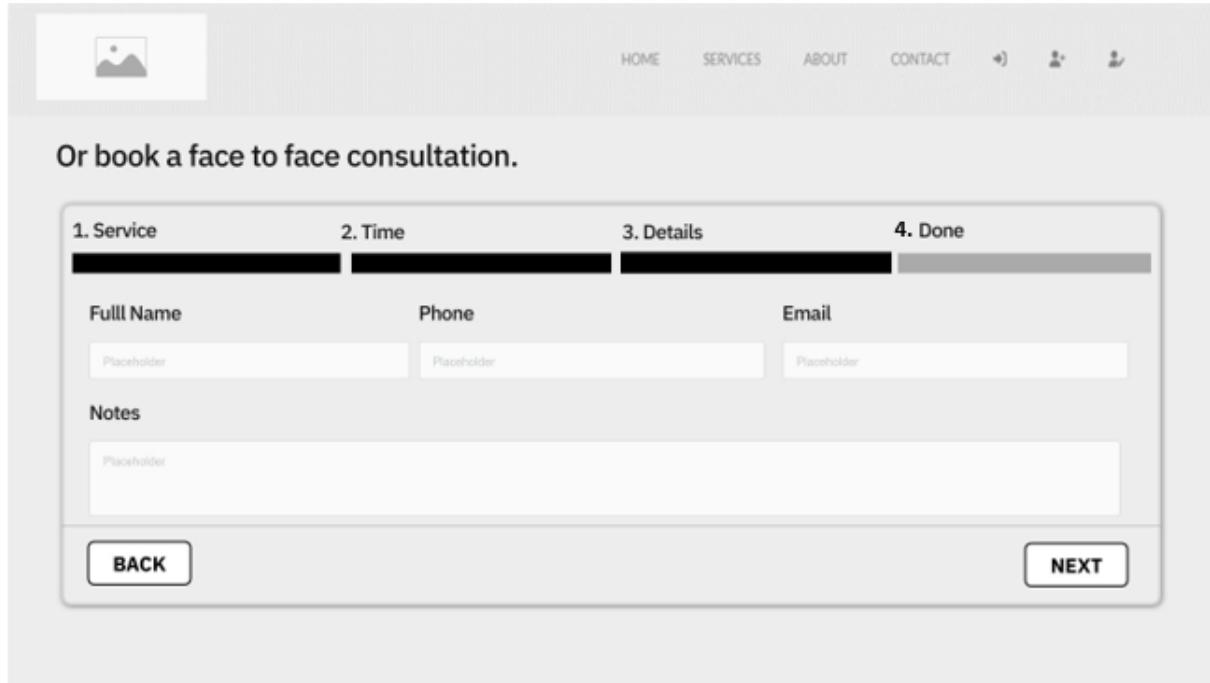
Or book a face to face consultation.

1. Service	2. Time	3. Details	4. Done
date	TIME	date	TIME
TIME	date	TIME	TIME
TIME	TIME	TIME	date
TIME	TIME	TIME	TIME

**BACK** **NEXT**

Figure 12: wireframe 9

## Book a consultation 3:



Or book a face to face consultation.

1. Service	2. Time	3. Details	4. Done
Full Name	Phone	Email	
Placeholder	Placeholder	Placeholder	
Notes			
Placeholder			

**BACK** **NEXT**

Figure 13: wireframe 10

Book a consultation 4:

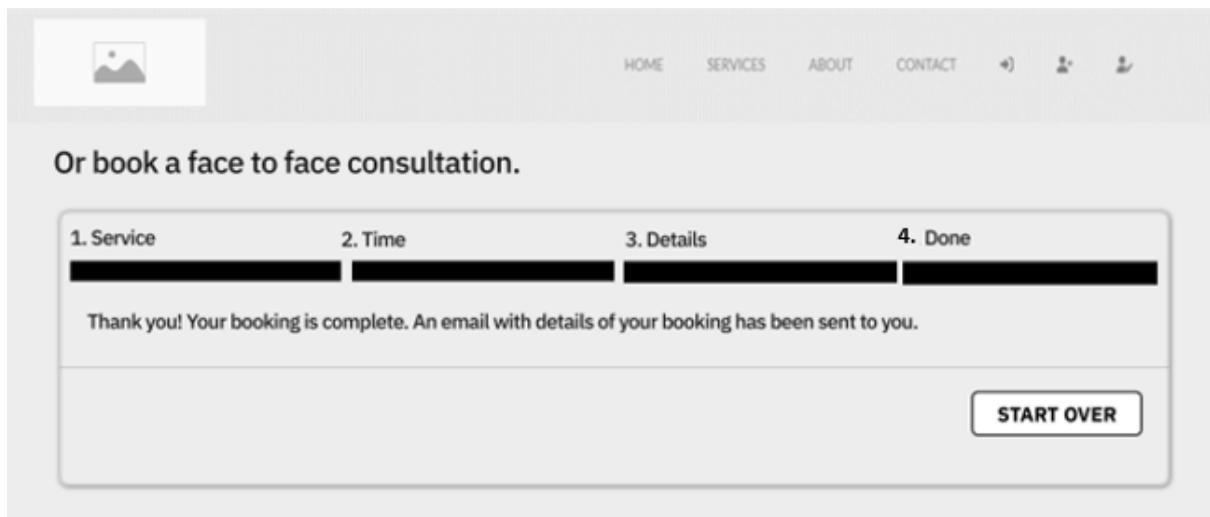


Figure 14: wireframe 11

## Registration page:


HOME
SERVICES
ABOUT
CONTACT




### Register

**Username \***

**First Name \***

**Last Name \***

### Contact info

**E-mail \***

**Phone \***

### About yourself

**Age \***

**Gender \***

**Nationality \***

### Trip Details

**Passport number\***

**Expiration Date \***

**Ticket Number**

### Disability nature

*This section will help us cater to your needs*

**Nature of special need \***  Mobility impairment  Visual impairment  Hearing impairment

**Mobility Requirement**  Wheelchair  Cane  Crutches

**Medication Used**

### Emergency Contact Info

**Name \***

**Relation \***

**Number \***

### Password

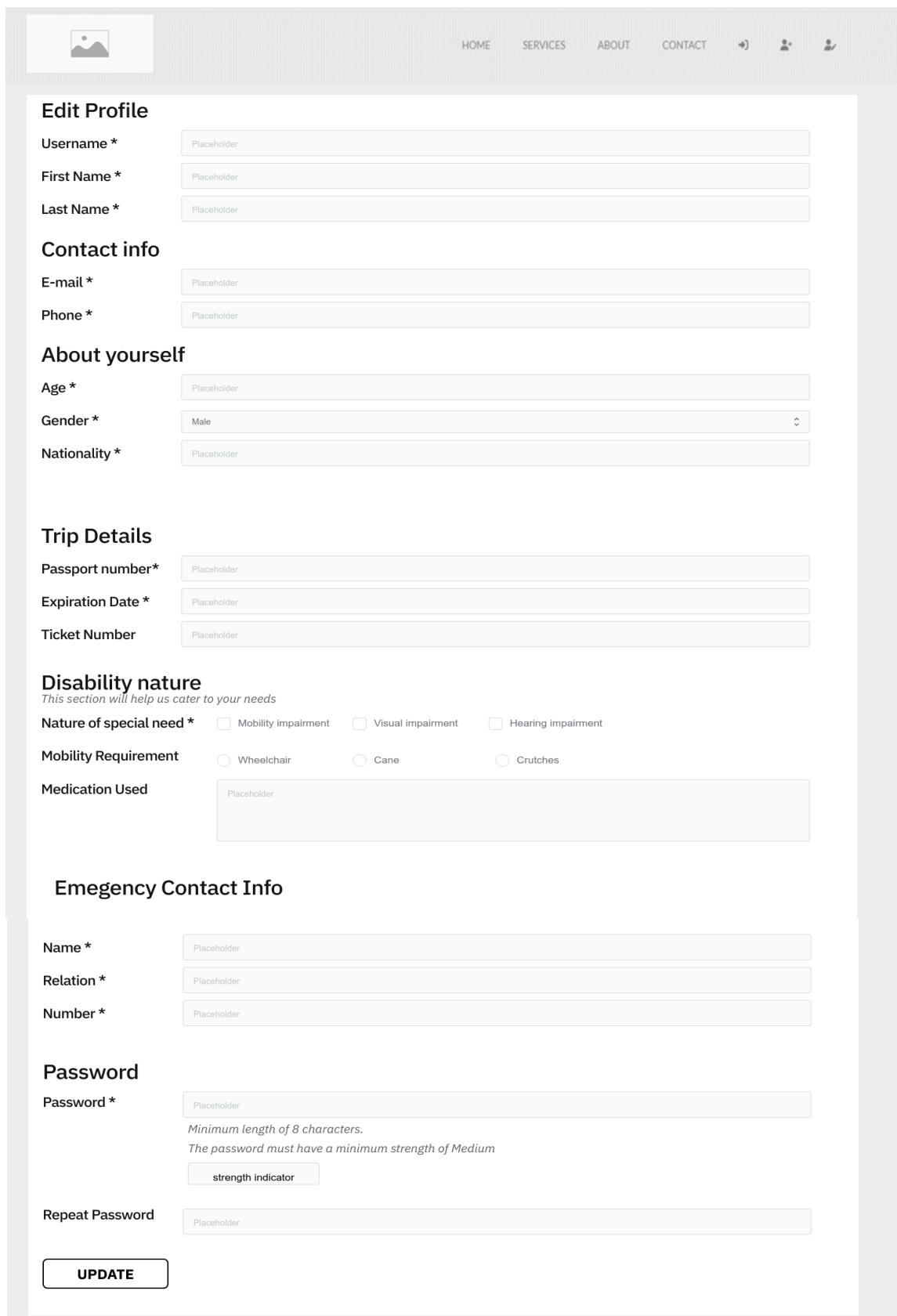
**Password \***   
Minimum length of 8 characters.  
The password must have a minimum strength of Medium

**Repeat Password \***

Send these credentials via email.

Figure 15: wireframe 12

## Manage profile page:



**Edit Profile**

**Username \*** Placeholder

**First Name \*** Placeholder

**Last Name \*** Placeholder

**Contact info**

**E-mail \*** Placeholder

**Phone \*** Placeholder

**About yourself**

**Age \*** Placeholder

**Gender \*** Male

**Nationality \*** Placeholder

**Trip Details**

**Passport number\*** Placeholder

**Expiration Date \*** Placeholder

**Ticket Number** Placeholder

**Disability nature**  
*This section will help us cater to your needs*

**Nature of special need \***  Mobility impairment  Visual impairment  Hearing impairment

**Mobility Requirement**  Wheelchair  Cane  Crutches

**Medication Used** Placeholder

**Emergency Contact Info**

**Name \*** Placeholder

**Relation \*** Placeholder

**Number \*** Placeholder

**Password**

**Password \*** Placeholder

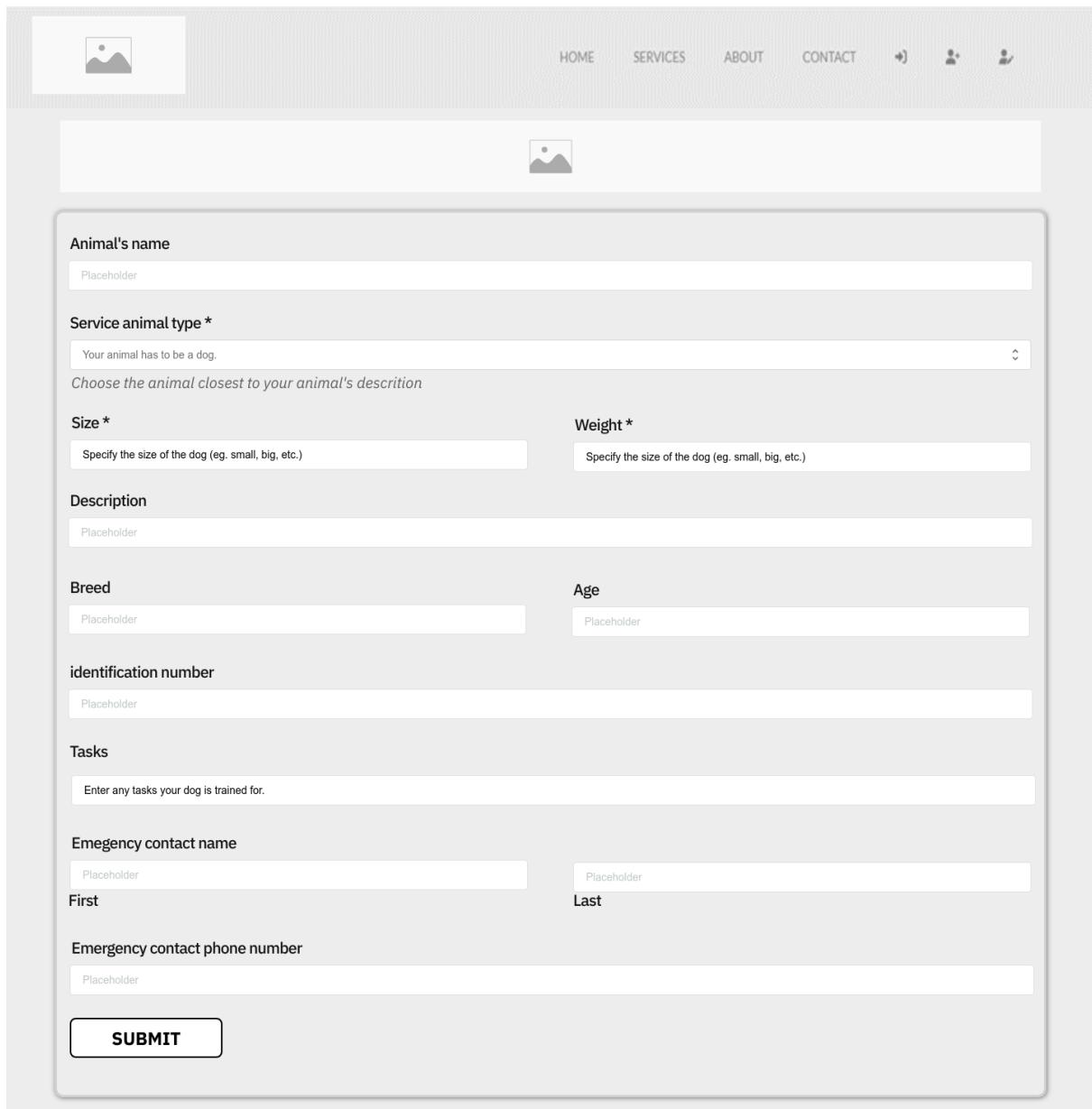
*Minimum length of 8 characters.  
The password must have a minimum strength of Medium*

**Repeat Password** Placeholder

**UPDATE**

Figure 16: wireframe 13

## Add service animal page:



The wireframe illustrates the 'Add service animal' form. At the top, there is a placeholder image for a profile picture. Below the header, there are two smaller placeholder images. The main form area contains the following fields:

- Animal's name:** Placeholder
- Service animal type \***: A dropdown menu with the placeholder "Your animal has to be a dog." and a note "Choose the animal closest to your animal's description".
- Size \***: Two input fields for "Size" and "Weight", both with the placeholder "Specify the size of the dog (eg. small, big, etc.)".
- Description:** Placeholder
- Breed:** Placeholder
- Age:** Placeholder
- identification number:** Placeholder
- Tasks:** Placeholder with the note "Enter any tasks your dog is trained for."
- Emergency contact name:** Two input fields for "First" and "Last", both with the placeholder "Placeholder".
- Emergency contact phone number:** Placeholder

A large "SUBMIT" button is located at the bottom left of the form.

Figure 17: wireframe 14

Request a service page:



- [HOME](#)
- [SERVICES](#)
- [ABOUT](#)
- [CONTACT](#)
- 
- 
- 

## Reserve you appointment!

**Service Provider**

DECEMBER

Sun	Mon	Tue	Wed	Thu	Fri	Sat
1	2	3	4	5		
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

### Personal information

**Name \***

**Phone \***

**Email \***

**Ticket Number\***

**mobility level \***

**Communication Method Preferred \***

**Communication needs**  
Specify any additional communication needs (e.g. Language preference or specific communication needs.)

**Level of visual impairment \***

**Preferred assistance method**

## Booking overview

Please check your appointment details below and confirm

<b>Location</b>	
<b>Service</b>	
<b>Worker</b>	
<b>Date &amp; Time</b>	

Figure 18: service request form after peer review

---

31 | Page

If user tries to request service or register their service animal, they will be redirected to this page and not allowed access:

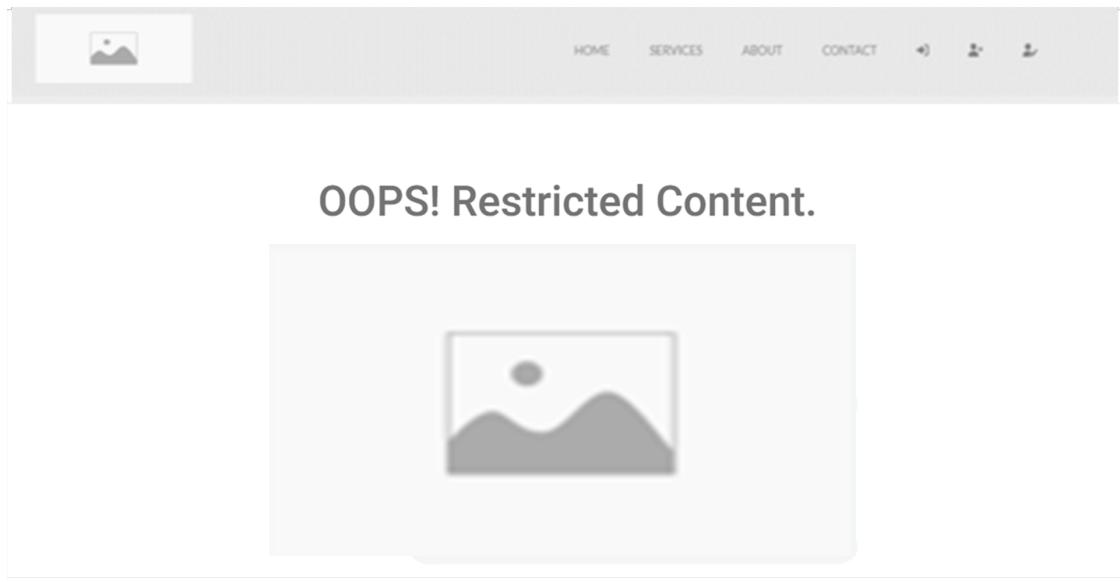


Figure 19: wireframe 16

### 3.1.3 Mockups

A mockup is a graphic representation of the user interface (UI) of an application. It is a static, non-functional depiction of the application's layout, design, and content. They are made early in the development process to act as a guide for designers, developers, and stakeholders and allow developers and designers to get input, and make necessary revisions early in the development process, lowering the likelihood of costly changes later.

Home screen slider:

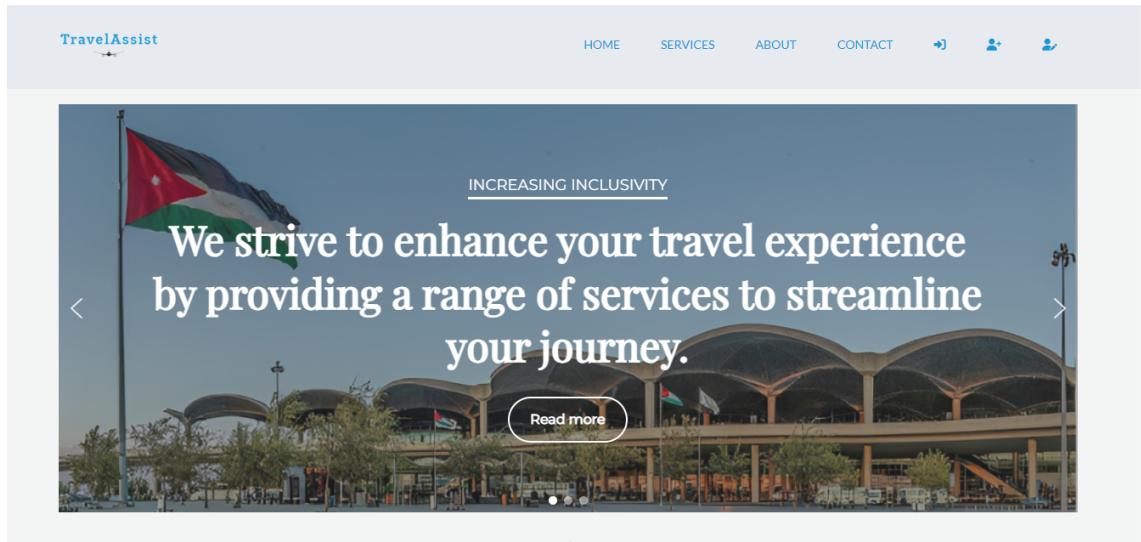
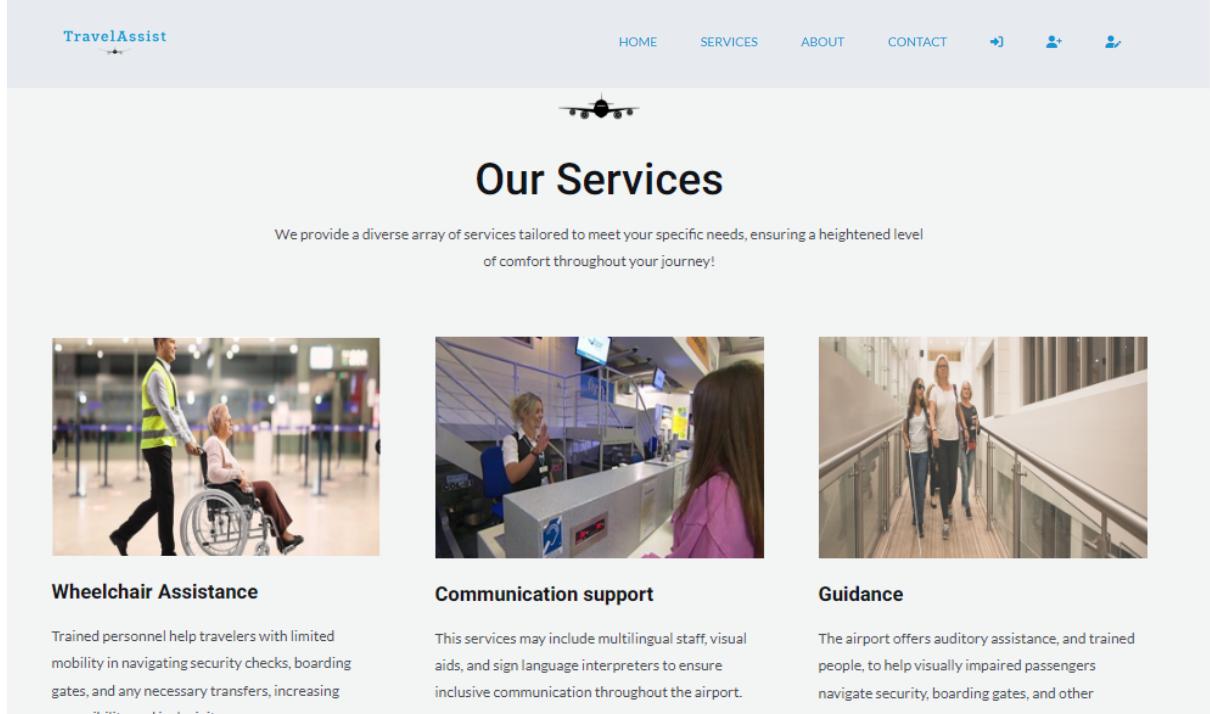


Figure 20: Mockup 1

Home screen part 2:



**Our Services**

We provide a diverse array of services tailored to meet your specific needs, ensuring a heightened level of comfort throughout your journey!



**Wheelchair Assistance**

Trained personnel help travelers with limited mobility in navigating security checks, boarding gates, and any necessary transfers, increasing



**Communication support**

This service may include multilingual staff, visual aids, and sign language interpreters to ensure inclusive communication throughout the airport.



**Guidance**

The airport offers auditory assistance, and trained people, to help visually impaired passengers navigate security, boarding gates, and other

Figure 21: Mockup 2

Home screen part 3:

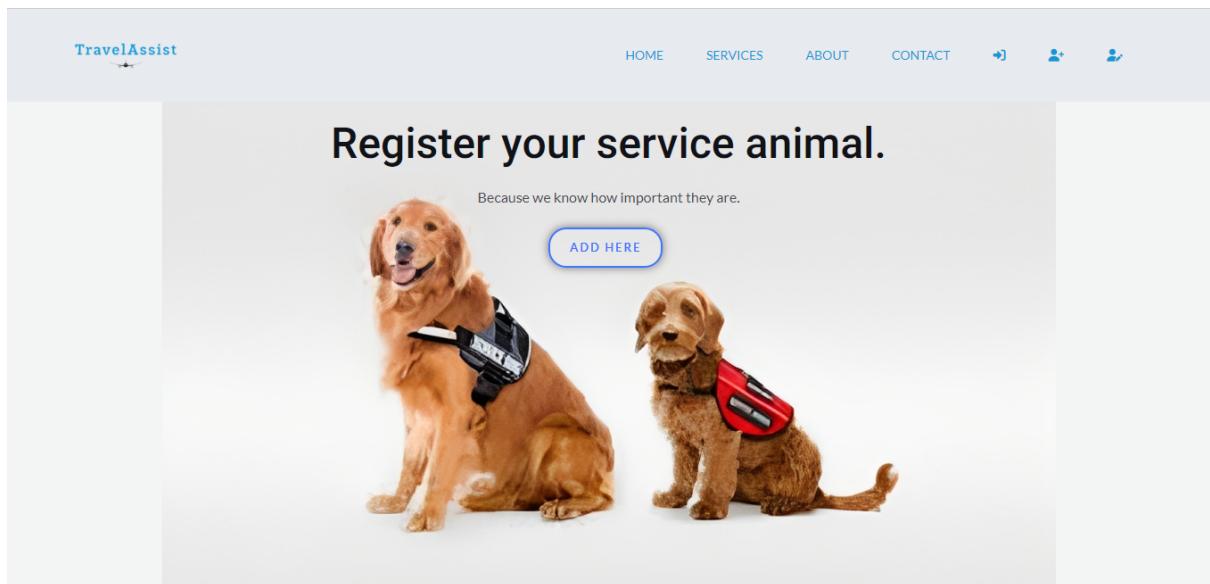


Figure 22: Mockup 3

Footer:

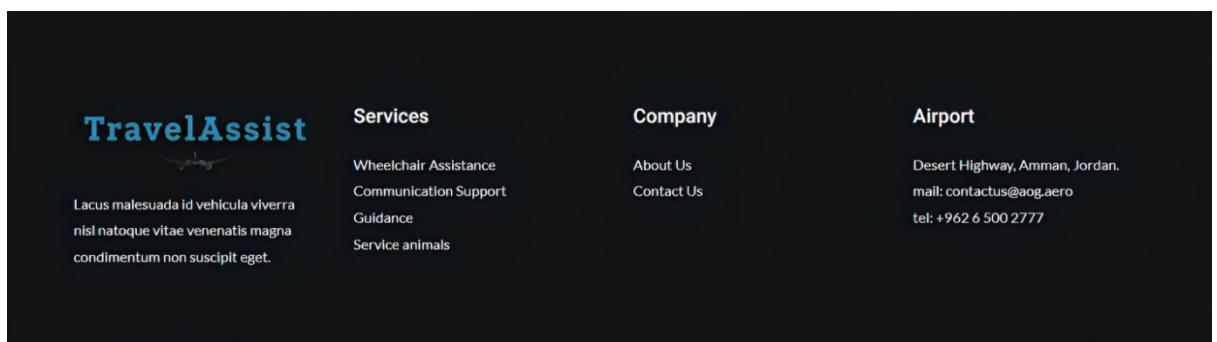


Figure 23: Mockup 4

Log in pop up:

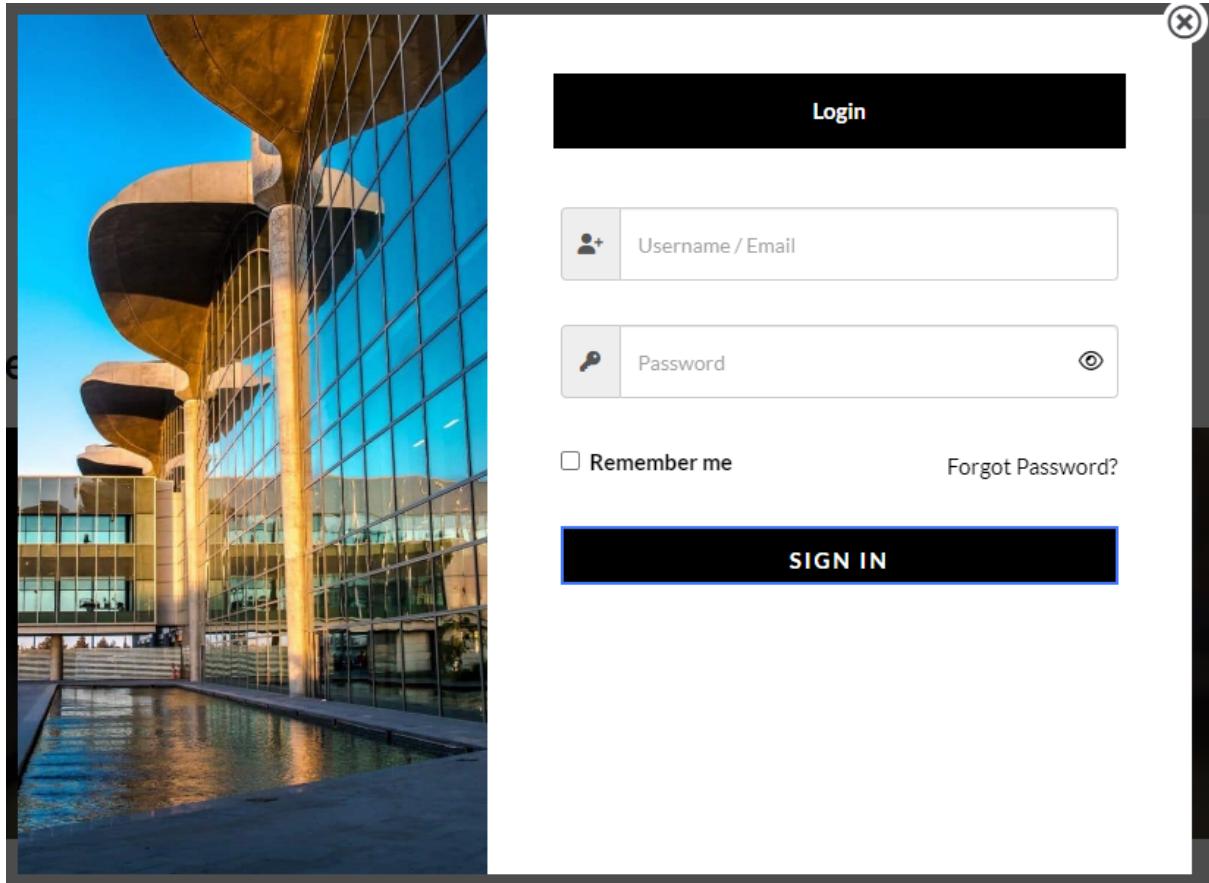


Figure 24: Mockup 5

## Services page:

**TravelAssist**

HOME SERVICES ABOUT CONTACT   



# Services

## What We Do

Travel Assist is committed to improving the travel experience for people with special needs. This website offers passengers a convenient and effective platform for requesting and receiving individualized help services. We seek to make air travel more accessible and inclusive by providing amenities such as wheelchair assistance, specialized staff, and communication help.



### Wheelchair Assistance

Providing essential mobility aid, wheelchair service ensures comfort, accessibility, and independence for users.

- » Diverse wheelchair options.
- » Request in advance for in-flight wheelchair.
- » Or bring your own wheelchair.
- » We will provide you with staff to help you.

---

### Communication Support

We provide visually and hearing-impaired customers by providing with dedicated staff to accompany you and ensure smooth travel.

- » Please contact us at least 48 hours before departure so that we can arrange for assistance and seat reservations.
- » On-site, ground staff provide check-in and gate assistance; cabin crew offers in-flight support.
- » Specialized assistance for luggage handling, safety, and aid to/from restrooms.



Figure 25: Mockup 6

Services page part 2:



**Guidance service.**

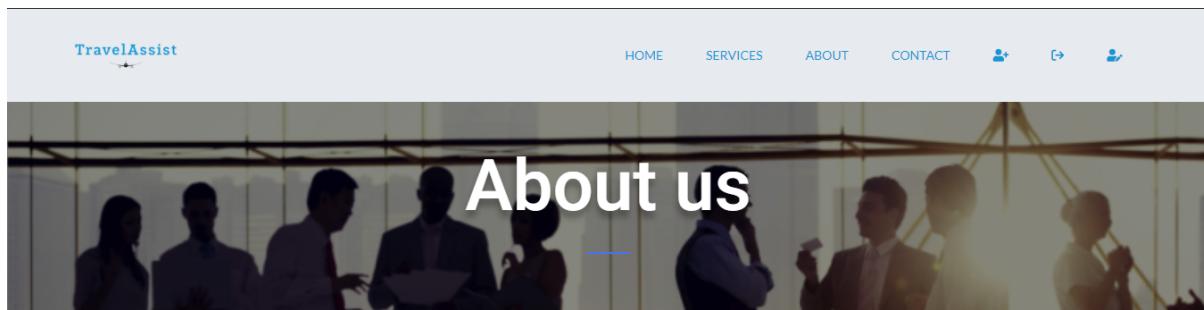
We provide you with dedicated staff to accompany you throughout your travel.

- » Expert support to ensure that security checks and baggage handling go well.
- » Stress-free airport navigation and timely support.
- » 48 hour notice in advance.
- » Have a service animal? Register them 48 hours in advance [here](#).

[BOOK](#)

Figure 26: Mockup 7

## About us page:



## Our goal

We aim to provide a seamless and inclusive air travel experience for individuals with special needs. By bridging the gap between airlines, airports, and passengers, we strive to ensure that every person with special needs feels supported, valued, and confident throughout their journey. Our vision is a world where air travel is accessible to all, where independence and dignity are upheld, and where individuals with special needs can freely explore the wonders of the world.

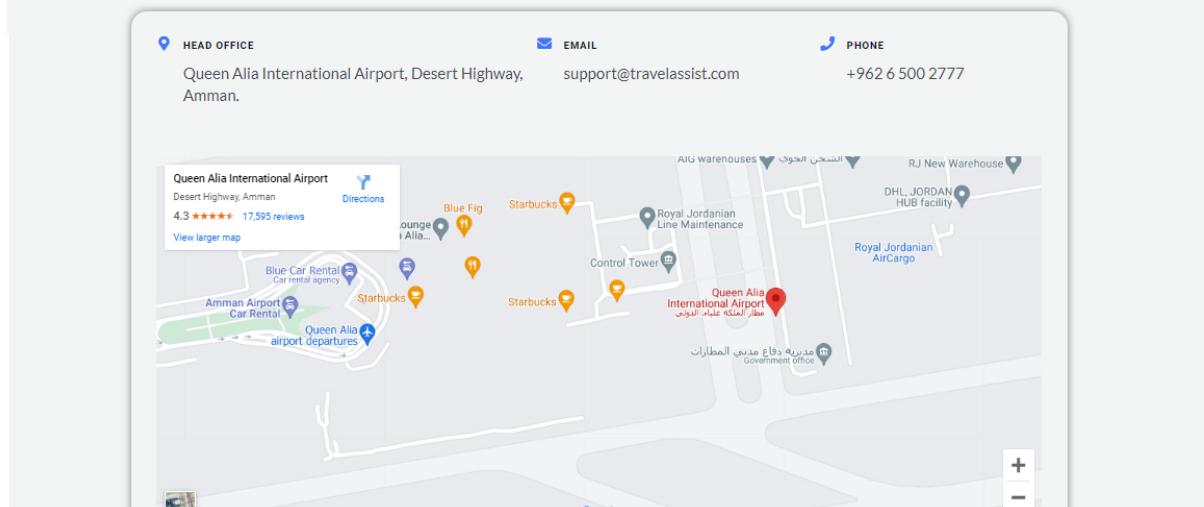
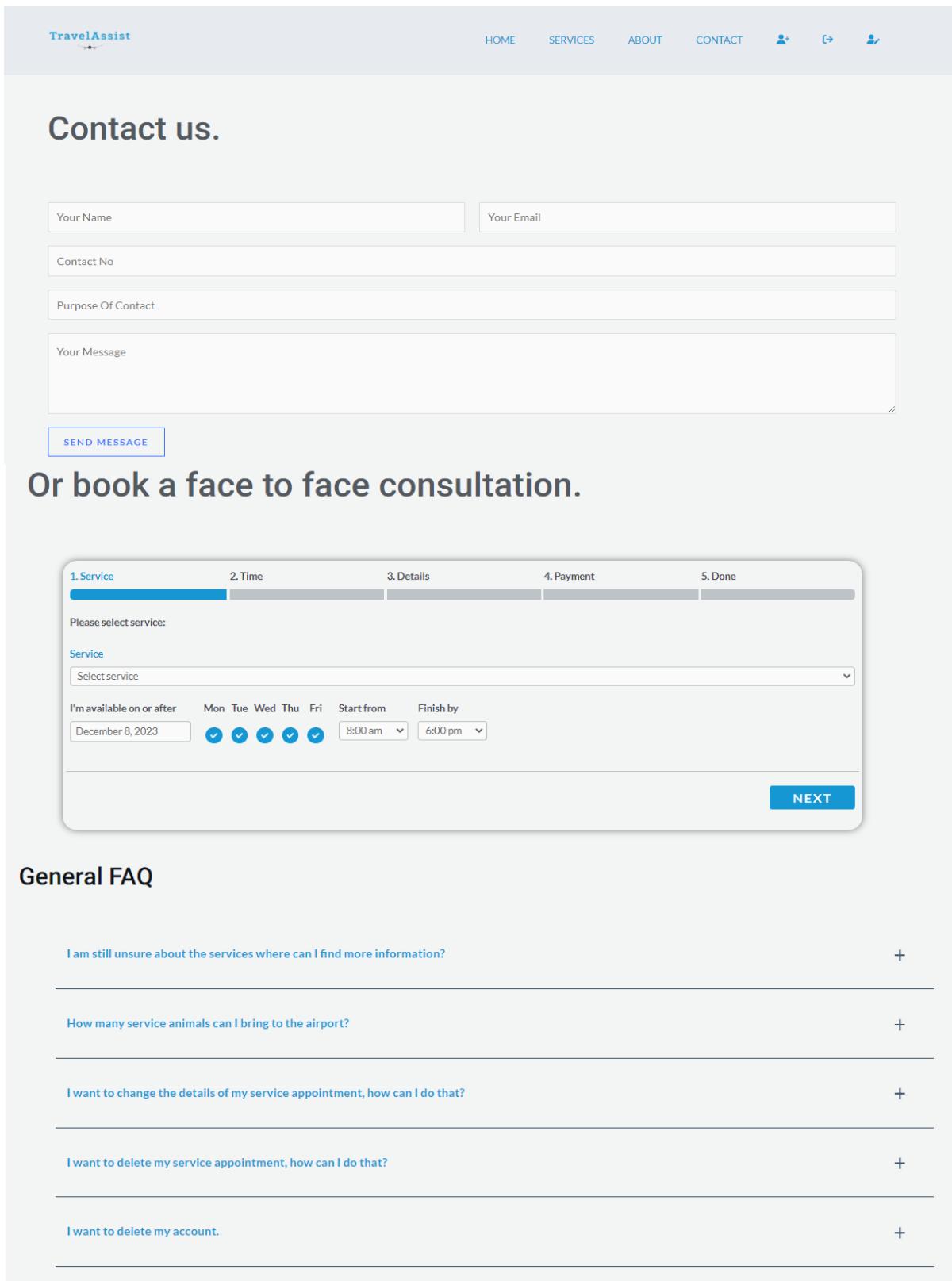


Figure 27: Mockup 8

Contact us page:



The screenshot shows the 'Contact us' section of the Travel Assist website. At the top, there's a navigation bar with links for HOME, SERVICES, ABOUT, CONTACT, and user icons. Below the navigation is a large 'Contact us.' heading. The form consists of several input fields: 'Your Name', 'Your Email', 'Contact No', 'Purpose Of Contact', and a large 'Your Message' area. A blue 'SEND MESSAGE' button is located below the message input. Below the form, a bold heading reads 'Or book a face to face consultation.' followed by a service booking interface. This interface includes a progress bar with steps 1. Service (selected), 2. Time, 3. Details, 4. Payment, and 5. Done. It asks 'Please select service:' with a dropdown menu labeled 'Select service'. Under 'I'm available on or after' is a date picker showing 'December 8, 2023' and a time range from '8:00 am' to '6:00 pm'. A blue 'NEXT' button is at the bottom right of the booking section. Finally, there's a 'General FAQ' section with five expandable questions, each preceded by a '+' sign.

Your Name

Your Email

Contact No

Purpose Of Contact

Your Message

SEND MESSAGE

## Contact us.

Or book a face to face consultation.

1. Service      2. Time      3. Details      4. Payment      5. Done

Please select service:

Service

Select service

I'm available on or after

Mon Tue Wed Thu Fri Start from Finish by

December 8, 2023 8:00 am 6:00 pm

NEXT

### General FAQ

I am still unsure about the services where can I find more information? +

How many service animals can I bring to the airport? +

I want to change the details of my service appointment, how can I do that? +

I want to delete my service appointment, how can I do that? +

I want to delete my account. +

Figure 28: Contact page after peer review

## book a consultation part 2:

1. Service      2. Time      3. Details      4. Done

Below you can find a list of available time slots for 1 hour consultation by Kareen Ziadat.  
Click on a time slot to proceed with booking.

Fri, Dec 08	<input type="radio"/> 10:00 am	<input type="radio"/> 9:00 am	<input type="radio"/> 5:00 pm	<input type="radio"/> 4:00 pm	<input type="radio"/> 3:00 pm	<input type="radio"/> 2:00 pm	<input type="radio"/> 1:00 pm	<input type="radio"/> 12:00 pm
	<input type="radio"/> 11:00 am	<input type="radio"/> 10:00 am	<input type="radio"/> 8:00 am	<input type="radio"/> 7:00 am	<input type="radio"/> 6:00 pm	<input type="radio"/> 5:00 pm	<input type="radio"/> 4:00 pm	<input type="radio"/> 3:00 pm
	<input type="radio"/> 12:00 pm	<input type="radio"/> 11:00 pm	<input type="radio"/> 9:00 am	<input type="radio"/> 8:00 pm	<input type="radio"/> 7:00 pm	<input type="radio"/> 6:00 pm	<input type="radio"/> 5:00 pm	<input type="radio"/> 4:00 pm
	<input type="radio"/> 1:00 pm	<input type="radio"/> 1:00 pm	<input type="radio"/> 12:00 pm	<input type="radio"/> 10:00 am	<input type="radio"/> 9:00 am	<input type="radio"/> 8:00 am	<input type="radio"/> 7:00 am	<input type="radio"/> 6:00 am
	<input type="radio"/> 2:00 pm	<input type="radio"/> 2:00 pm	<input type="radio"/> 1:00 pm	<input type="radio"/> 11:00 am	<input type="radio"/> 10:00 am	<input type="radio"/> 9:00 am	<input type="radio"/> 8:00 am	<input type="radio"/> 7:00 am
	<input type="radio"/> 3:00 pm	<input type="radio"/> 3:00 pm	<input type="radio"/> 2:00 pm	<input type="radio"/> 12:00 pm	<input type="radio"/> 11:00 am	<input type="radio"/> 10:00 am	<input type="radio"/> 9:00 am	<input type="radio"/> 8:00 am
	<input type="radio"/> 4:00 pm	<input type="radio"/> 4:00 pm	<input type="radio"/> 3:00 pm	<input type="radio"/> 1:00 pm	<input type="radio"/> 12:00 pm	<input type="radio"/> 11:00 am	<input type="radio"/> 10:00 am	<input type="radio"/> 9:00 am
Mon, Dec 11	<input type="radio"/> 5:00 pm	<input type="radio"/> 4:00 pm	<input type="radio"/> 2:00 pm	<input type="radio"/> 1:00 pm	<input type="radio"/> 12:00 pm	<input type="radio"/> 11:00 am	<input type="radio"/> 10:00 am	<input type="radio"/> 9:00 am
	<input type="radio"/> 8:00 am	<input type="radio"/> 5:00 pm	<input type="radio"/> 4:00 pm	<input type="radio"/> 3:00 pm	<input type="radio"/> 2:00 pm	<input type="radio"/> 1:00 pm	<input type="radio"/> 12:00 pm	<input type="radio"/> 11:00 am
	<input type="radio"/> 9:00 am	<input type="radio"/> 8:00 am	<input type="radio"/> 4:00 pm	<input type="radio"/> 3:00 pm	<input type="radio"/> 2:00 pm	<input type="radio"/> 1:00 pm	<input type="radio"/> 12:00 pm	<input type="radio"/> 11:00 am

**BACK** **>**

Figure 29: Mockup 10

## book a consultation part 3:

1. Service      2. Time      3. Details      4. Done

You selected a booking for 1 hour consultation by Kareen Ziadat at 9:00 am on December 21, 2023. The price for the service is \$0.00.  
Please provide your details in the form below to proceed with booking.

Full name	Phone	Email
<input type="text"/>	<input type="text"/> 	<input type="text"/>
Notes <input type="text"/>		

**BACK** **NEXT**

Figure 30: Mockup 11

Note full name, email will be automatically filled out from account information.

## Book a consultation part 4:

1. Service      2. Time      3. Details      4. Done

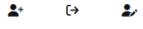
Thank you! Your booking is complete. An email with details of your booking has been sent to you.

**START OVER**

Figure 31: Mockup 12

Registration page:

**TravelAssist**

HOME SERVICES ABOUT CONTACT 

## Register

Username \*

First Name \*

Last Name \*

### Contact Info

E-mail \*

Phone \*

### About Yourself

Age \*

Gender \*  Male

Nationality \*

### Trip Details

Passport number \*

Expiration date \*

Ticket Number

### Disability nature

*This section will help us cater to your needs*

Nature of special need \*  Mobility Impairment  Visual Impairment  Hearing Impairment

Mobility Requirement  Wheelchair  Cane  Crutches

Medication Used

### Emergency Contact Info

Name \*

Relation \*   
*what is your relation to you emergency contact specified eg. (mother, father, etc.)*

Number \*

### Password

Password \*   
Minimum length of 8 characters.  
The password must have a minimum strength of Medium

 Weak

Repeat Password \*

Send these credentials via email.

**ADD USER**

Figure 32: Mockup 13

Manage profile page:

TravelAssist
HOME
SERVICES
ABOUT
CONTACT

### Edit Profile

Username *	KareenZiadat
First Name *	Kareen
Last Name *	Ziadat
<b>Contact Info</b>	
E-mail *	kareenbziadat@gmail.com
Phone *	
<b>About Yourself</b>	
Age *	
Gender *	Male
Nationality *	
<b>Trip Details</b>	
Passport number *	
Expiration date *	
Ticket Number	
<b>Disability nature</b>	
<i>This section will help us cater to your needs</i>	
Nature of special need *	<input type="checkbox"/> Mobility Impairment <input type="checkbox"/> Visual Impairment <input type="checkbox"/> Hearing Impairment
Mobility Requirement	<input type="radio"/> Wheelchair <input type="radio"/> Cane <input type="radio"/> Crutches
Medication Used	<input type="text"/>
<b>Emergency Contact Info</b>	
Name *	<input type="text"/>
Relation *	<input type="text"/>
<i>what is your relation to your emergency contact specified eg. (mother, father, etc.)</i>	
Number *	<input type="text"/>
<b>Password</b>	
Password	<input type="password"/> <small>Minimum length of 8 characters.</small> <small>The password must have a minimum strength of Medium</small>
<small>Strength indicator</small>	
Repeat Password	<input type="password"/>
<b>UPDATE</b>	

Figure 33: Mockup 14

## Add service animal page:



**TravelAssist**

HOME SERVICES ABOUT CONTACT   

---

Animal's name

Service Animal Type \*  
 You animal has to be a dog.  
 Choose the option closest to your animal's description

Size \*  
 Specify the size of the dog (eg. small, big, etc.)      Weight \*  
 Specify the weight of the dog in Kg.

Description

Breed  
 Age

Identification number

Tasks  
 Enter any specific tasks your dog is trained for.

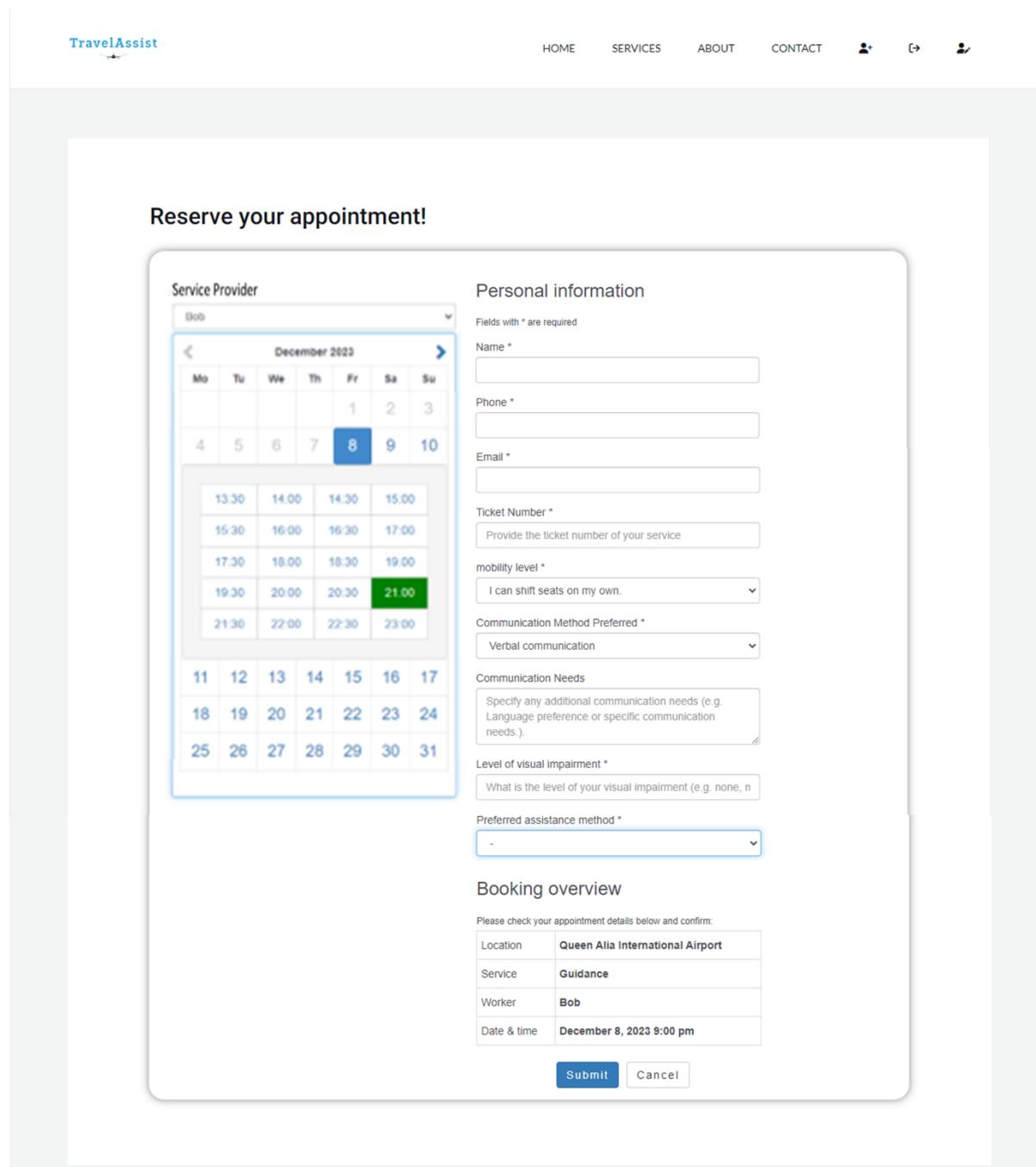
Emergency contact name  
 First  Last

Emergency contact phone number

**SUBMIT**

Figure 34: Mockup 15

Request a service page:



The screenshot shows a service request form for "Travel Assist". At the top, there's a navigation bar with links for HOME, SERVICES, ABOUT, CONTACT, and user icons. Below the navigation, a large button says "Reserve your appointment!". The main form area is titled "Service Provider" and contains a calendar for December 2023. A specific date, December 8, 2023, at 9:00 pm, is highlighted in green. To the right of the calendar, there are sections for "Personal information" (Name, Phone, Email, Ticket Number), "Communication Needs" (dropdown for mobility level, communication method, and additional needs), and "Level of visual impairment" (dropdown). Below these, a "Booking overview" section displays appointment details: Location (Queen Alia International Airport), Service (Guidance), Worker (Bob), and Date & time (December 8, 2023 9:00 pm). At the bottom are "Submit" and "Cancel" buttons.

Figure 35: Service request form after peer review.

If user tries to request service or register their service animal, they will be redirected to this page and not allowed access:

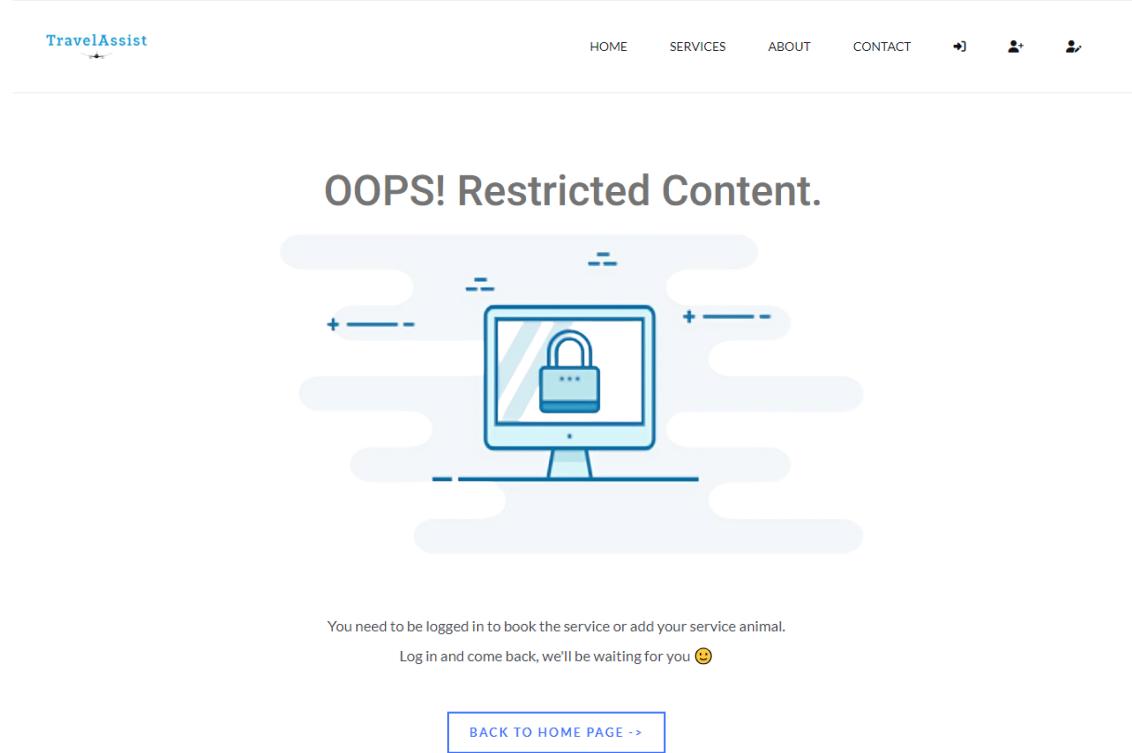
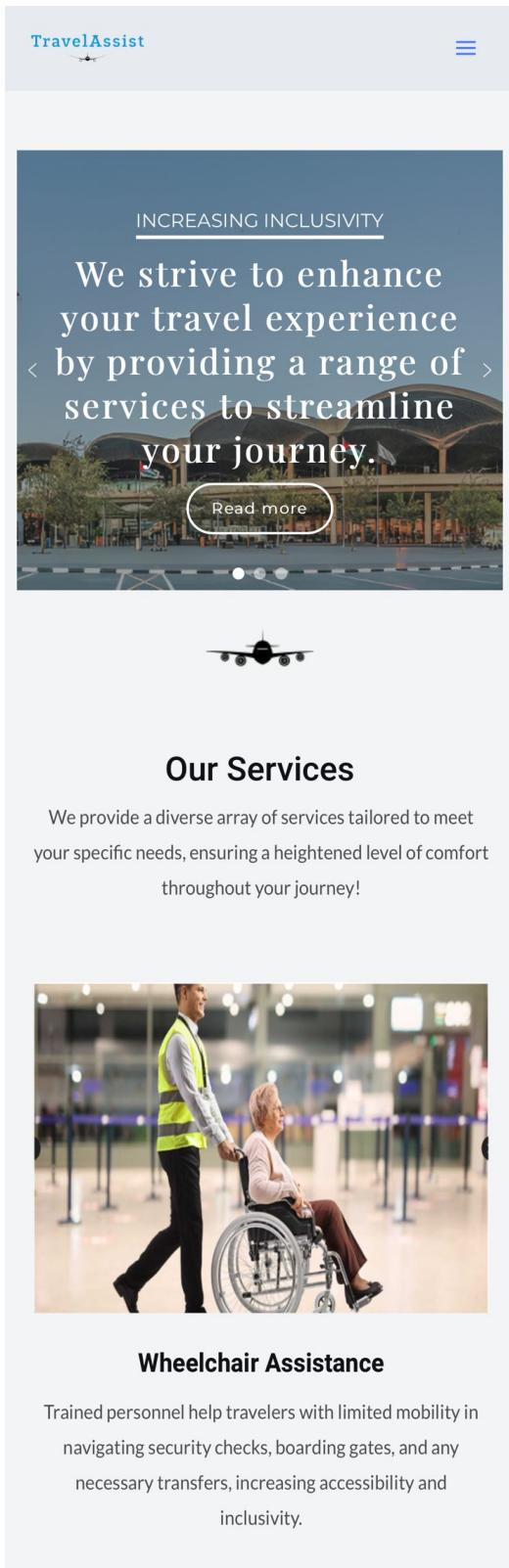


Figure 36: Mockup 17

Mobile interface of home page showing responsiveness:



**INCREASING INCLUSIVITY**

We strive to enhance your travel experience by providing a range of services to streamline your journey.

[Read more](#)



**Our Services**

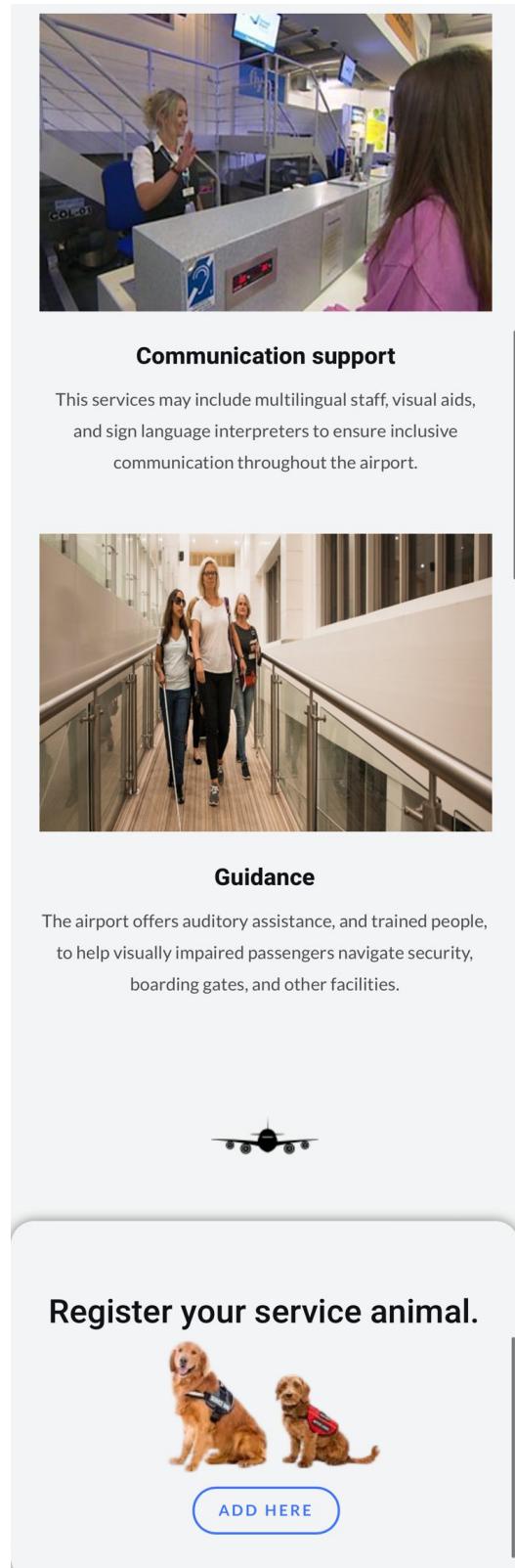
We provide a diverse array of services tailored to meet your specific needs, ensuring a heightened level of comfort throughout your journey!



**Wheelchair Assistance**

Trained personnel help travelers with limited mobility in navigating security checks, boarding gates, and any necessary transfers, increasing accessibility and inclusivity.

Figure 38: Mockup 18 showing the home screen on mobiles.



**Communication support**

This service may include multilingual staff, visual aids, and sign language interpreters to ensure inclusive communication throughout the airport.



**Guidance**

The airport offers auditory assistance, and trained people, to help visually impaired passengers navigate security, boarding gates, and other facilities.





**Register your service animal.**



[ADD HERE](#)

Figure 37: Mockup 19 showing home screen on mobiles part 2.

## Low Level Design (LLD)

## 3.1.4 ERD

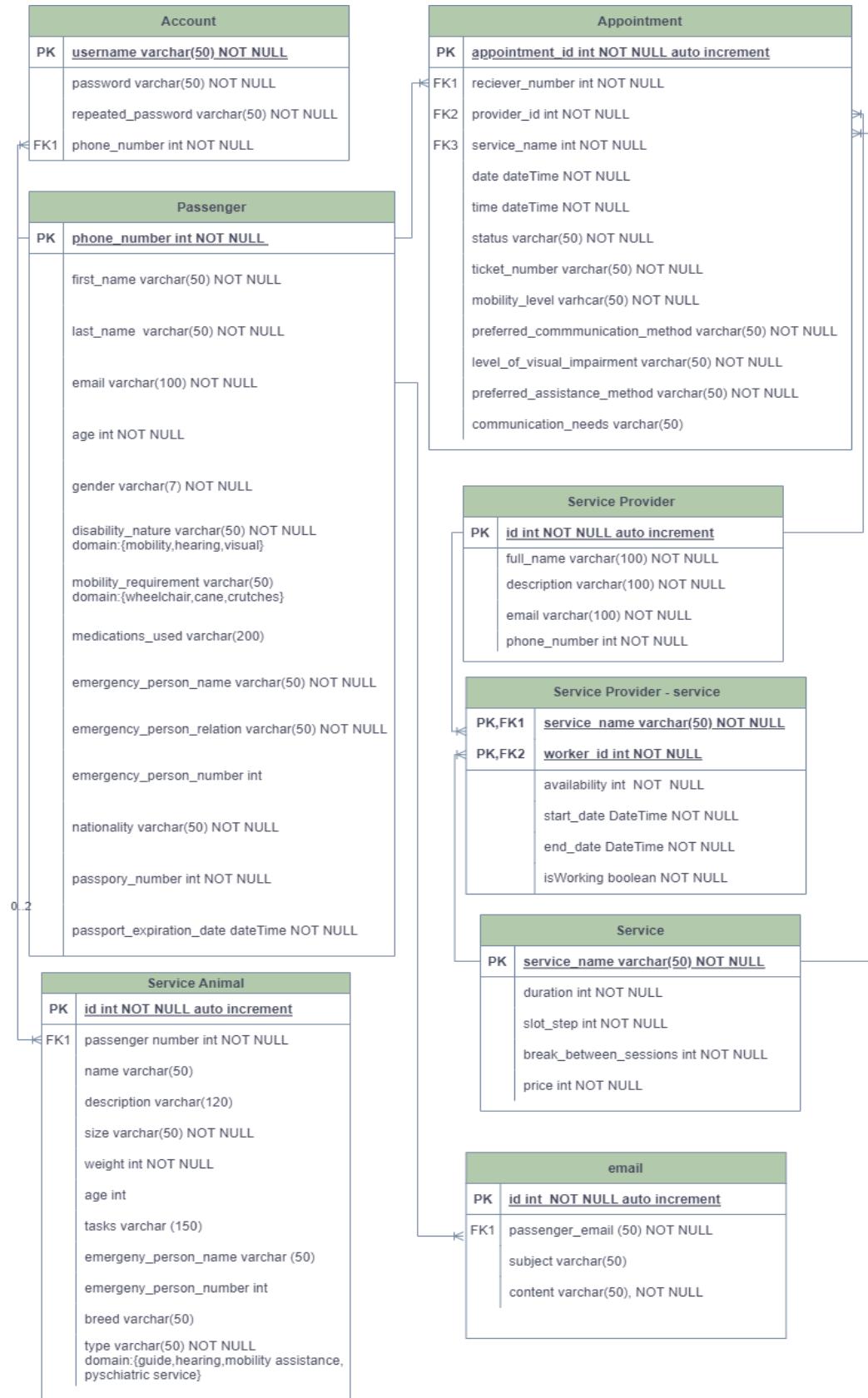


Figure 39: ERD of travel assist

## 4. Overall Development

### 4.1 Development Overview

An overview of the current software development techniques:

#### 1. RAD Platforms:

This approach prioritizes quick development and iteration with the goal of producing a working model as soon as possible. There are three types of RAD outlined below:

- **Code generation:** These are platforms that create code for application development, which automates the process of development and creates flexibility. They offer drag-and-drop functionality, places to enter prompt description, ability to upload wireframe pictures, and places for manual coding, allowing for more personalized functionalities. With the growth of AI, the number of code generation tools has expanded significantly. For example, cloud wizard, chatGPT.
- **Low-code:** Low-code platforms reduce the amount of coding and the expertise needed to develop an application by employing visual interfaces with drag-and-drop functionality while also allowing writing codes. For example, WordPress and OutSystems provide a low-code platform for rapidly developing enterprise-grade apps.
- **No-code:** They are used by non-developers and do not require any coding experience. They usually rely on templates and drag and drop functionalities to produce applications with intuitive interfaces. For example, Bubble, Adalo and Wix.

#### 2. Building applications from scratch by fully writing code:

This is the traditional method which requires coding databases, frontend (design), backend (functionality, workflow), responsiveness, and many other aspects. It provides complete control over the development process, but it can be time-consuming and costly (mostly affects complicated systems). For example, building a complicated e-commerce website employing technologies such as React for the frontend, Node.js for the backend, and MongoDB for the database.

#### 3. Customization of current product:

Modifying the program to satisfy specific user needs or to add new features. This approach is frequently chosen when an existing product's architecture and key features closely match the intended results, but changes are needed to better meet requirements or to integrate with existing system. For examples: customization of current products includes adapting an open-source CMS like Drupal.

#### 4.1.2 Development Techniques Comparison

Below is a comparison between the various development techniques:

**RAD - Code generation:** Can be used by both technical and non-technical users, however, non-technical people might find it hard to understand the generated code and it is usually more dependable and bug-free than code written from scratch. It also speeds up most of the development processes. However, in some cases, the code is difficult to understand, making it difficult to modify, affecting future maintenance. Furthermore, depending on the tool, it has less flexibility, complexity than totally hand-written code.

**RAD - Low code:** They are appropriate for both technical and non-technical users, allowing complex applications to be created with less coding work but with the ability to customize through code. However, the application complexity is still limited compared to fully coded which might not be suitable to meet the requirements of more complicated systems.

**RAD – no code:** It is mostly used by non-developers who want to develop an application that isn't complicated to support their business. The issue with no-code tools is that the functionality and complexity of the system produced is limited.

**From scratch:** It provides complete control and customization, as well as the ability to create complicated and one-of-a-kind applications. At the same time, it is slow, costly, and require good technical capabilities, particularly for bigger applications.

**Customization of current products:** This method does modification on current products which save time and resources. It is mostly used when the product's characteristics and basic architecture are close to intended result. On the other hand, it is not appropriate for apps that need to have special or cutting-edge functionality since it is limited by the product's current architecture and features.

Although code generation greatly accelerates development, complicated systems frequently require manual adjustments which might be complicated in some cases. Low-code and no-code platforms are becoming more and more used for their user interfaces, even non-technical individuals may create apps using graphical user interfaces with very little coding knowledge. These platforms can, however, have limitations regarding scalability, customization to business needs and flexibility (low code offers more customization options). Although it takes a lot of time and experience, developing from scratch gives developers the most flexibility and control and enables them to design highly personalized and optimized solutions. Finally, customization of current products provides a balance between development speed and customization, however it may be limited by the features and architecture of the original product. There are trade-offs between control, speed, usability, and personalization in each of these approaches so depending on the business needs, the approach will be chosen.

### 4.1.3 Development Tools

Tool	usage	advantage	disadvantage
WordPress (low-code)	It is mainly used for content driven websites (e.g. blogs, websites, e-commers), offers high flexibility through themes and plugins (If no plugin available one can program their own).	Provides plugins (extension of functionality) and themes. Extensive documentation. SEO friendly. Secure and scalable. Full ownership of website (can move to diff host or service provider)	Customization is complex. Requires updates and maintenance of plugins and themes (can be set automatic). Too many plugins used make the site unstable and slow. Some plugins and themes are paid.
Adalo (no-code)	To build mobile and webapp without the need for coding. It is used to create a degree of custom applications.	Drag and drop interface without coding. Supports integration with third parties. Provides templates to make it easier.	Doesn't offer enough customization for complex requirements. Doesn't offer scalability. Can have limitations in performance for data-intensive applications.
Outsystems (low-code)	It creates complex, enterprise-grade applications, usually used by businesses that need to develop robust, secure, and large-scale applications efficiently.	Highly scalable and secure. Allows integration with existing databases and systems.	Can be expensive. Steep learning curve. Some customization limitations.
Wix (no-code)	Wix makes it simple to personalize and quickly establish a small-medium scale website with its drag-and-drop interface and large selection of templates.	Design is intuitive. Provides templates. Updates are maintained by Wix.	Not scalable. SEO limitation. Website remain on Wix platform limiting data portability as migrating to other sites/platforms is hard.

Table 2: Advantages and disadvantages of each development tool

#### 4.1.4 The preferable Development Tool and Technique

I decided to solve this problem using a website since it can be accessed through mobile phones and pcs. The problem requires booking an appointment which relies heavily on data input from the user and most users prefer to do such tasks on the PC as it is more convenient. To add the website won't be accessed continuously since once the user books the appointment, they won't need to visit the website again. In past course, I've used WordPress to develop a website which gave me the opportunity to discover this tool, realize all the plugin options that can be used (the variety, the amount of documentation available to support learning), templates that can be employed and the ease of use in achieving such complex functionalities in a shorter amount of time.

Development technique: low-code rapid application development technique.

Justification: I decided to do rapid application development to enable quick development of my website which is essential since the solution that the website is providing is for a problem that needs to be solved as soon as possible. While speed can also be achieved through no-code approach, maintainability would be an issue since the functionalities of the business should be highly customizable. The low-code approach allowed me to mostly rely on drag and drop functionalities while also allowing customization in CSS and HTML code, making it an ideal choice for the application due to its complexity. This technique allows for quick improvements and changes which reduces the development time significantly and allows future maintenance. This is especially crucial for complying with airport security protocols and laws, which are always changing, as well as for adjusting to the changing needs of users with special needs. Furthermore, low-code platforms frequently come with integrated features that make it easier to create an application that meets business and user needs. Lastly, the scalability and ease of maintenance of these platforms aligns with my system requirements for supporting a growing user base and minimizing downtime.

Development tool: WordPress.

Justification: As mentioned before, I am familiar with this tool and I find it much easier to use compared to other low-code tools like OutSystems or Wix (no-code tool). WordPress has many plugins that align well with my booking functionality, various themes that fit the house style of the company and is already responsive which is especially important in a website that is tailored for use on mobile and website.

WordPress provides a large selection of highly customizable plugins to meet the unique requirements of the Travel Assist application. For example, requesting the services offered by the website correctly, registering service animals, and creating (and managing) user profiles. The booking functionality is complicated and has many requirements (calendar integration, multiple workers, multiple service providers, multiple connections, ensures no double booking (for worked, service, date, and time), all input fields. which can be extremely time-consuming and complicated to develop using tools like adalo and OutSystems. However, after researching, I found a wide variety of

appointment booking plugins each with their strengths or limitations and I found one that exactly meets the business requirements. Another functionality that aligns well with WordPress is the one-to-one face consultation booking appointment which needs to have a more streamlined process. After researching the other tools, I found the process, user interface and functionality altogether are most appropriate in WordPress and not found in other tools. As for the service animal registration functionality, I didn't need an extremely process and rather a simple form therefore any tool could've been used.

Moreover, the plugins can be used in the future to enforce the validation checks (ensure ticket number is valid, maximum 2 service animals per ticket) by paying for the basic plan. (Making WordPress extremely maintainable)

Additionally, WordPress (unlike adalo) can manage large amounts of traffic and user data, satisfying a critical system requirement: scalability during periods of high travel demand. Almost all the tools provide security to ensure the sensitive data of the travelers are maintained therefore this wasn't a limiting factor, however I decided to host the website on a reliable hosting service that provides an SSL certificate.

Lastly, WordPress has many integration features which are crucial for the application since they enable a smooth connection with external systems like flight and ticket information databases through a variety of plugins and APIs.

## 4.2 Software Development Methodology

### Software development methodologies available:

1. **Waterfall Model:** A methodical, step-by-step process in which every stage is based on the deliverables of the one before it.
2. **Rapid application development (RAD):** Emphasizes quick prototyping and iterative delivery, working closely with the customer to enable quick and adaptable adjustments.
3. **Agile:** It is an iterative, flexible strategy that breaks the project up into smaller, more manageable components. Each requirement is designed developed and tested until all requirements are met.
4. **Scrum:** It is an agile framework that divides software development into small teams that operate in sprints. Sprints are iterative cycles that include regular evaluation and modification.
5. **DevOps:** Combines IT operations (Ops) and software development (Dev) to reduce the duration of the development life cycle, produce high-quality software constantly with more frequent updates, and closely connect with business goals.

**Chosen methodology:** Agile.

One of the main deciding factors for not choosing the waterfall methodology was that in the context of this application, changes occur often since the requirements for the customer segment (travelers with special needs) is constantly being changed and enhanced, airport regulations change significantly. Waterfall requires that if any of the requirements is modified, the entire application must go through all the waterfall stages (analysis, requirement specification, design, development, testing and integration, and deployment) are repeated which takes a lot of time and in since the project is always changing, this will make it extremely hard to follow and a waste of effort. Moreover, I have never done a similar project and I am constantly trying to understand the customer and user needs through research which will cause me to modify requirements continuously. The budget is also not fixed.

Personally, my approach to development is always one feature at a time. I find that I move faster this way and in general it is my approach to problem solving (divide and concur). I also find that I get more motivated to continue working once a feature is done and I am closer to the goal. Working one feature at a time is consistent with agile methodology.

While the agile methodology is usually done in teams, in the case of this project, I am the whole team (testers, UX designer, technical writer, etc.)

Travel Assist needs flexibility, scalability, and frequent updates and the agile development methodologies provide flexible response to evolving requirements, multiple iterations based on user input, and ongoing enhancement, all of which are essential for the website which provides help to individuals with special needs in a constantly changing airport environment (e.g. different service, constraints, regulations, etc.). Moreover, Travel Assist was developed directly on WordPress rather than through prototyping. This strategy allows for fast testing and user feedback. Since the development team is small and manages several tasks, this model (no prototyping) promotes effective resource allocation by prioritizing functional software over prolonged prototyping. This aligns well with the agile methodology's emphasis on rapid deployment and iterative improvement. Not relying on prototyping, the scope, and not having a lot of time for development makes the RAD methodology not suitable.

Disadvantages include the need for constant stakeholder engagement to make sure that the requirements are as expected and there is always the risk of scope creep if it is not properly managed (I included enough time buffer and started early to avoid this). Benefits include the emphasis on tiny increments to enable testing and improvement of features like service animal registration, user profile building, and assistance service booking, guaranteeing that every component is successfully meeting the requirements of the target consumers and are given sufficient time.

#### 4.3 Overall Design and Development Justification

##### **Justification of solution to the presented problem:**

The Travel Assist system strives to solve the difficulties that passengers with special needs face when traveling through airports. The main issue is the absence of an organized, effective, and easily accessible platform for these passengers to request services. As a comprehensive website, Travel Assist directly solves this issue by giving a single point of access to all relevant services (Guidance support, communication support, wheelchair assistance, and registration of the service animal that they will bring), speeding the process, and improving the user experience. The website's features, such service animal registration, and service requests, are specifically developed to meet the different needs of disabled travelers, assuring inclusivity and convenience of use.

### **Overall design justification:**

Multiple design decisions were made with the user's needs and wants in mind. Below is the design justification for each main functionality of the system:

1. Information display: I used a slider in the main page to provide an overview on the main functionalities provided on the website, I relied on small paragraphs, bullet points and images to describe the services, Icons were used whenever possible to decrease words, accordion were used for Frequently asked questions to hide the answer unless necessary. Forms were also not crowded with words.
2. All services were described on one page but every service redirects to a form with a title clearly indicating which service they are requesting to help the customer not feel lost.
3. I didn't want to confuse the user thinking that we have a service to help with service animals so it was made in a separate page clearly showing that you can only add the service animal through the website. It is also a simple form for registration.
4. The face-to-face consultation booking form doesn't take the user to another page and instead they do the entire process within the contact page which is less frustrating and more convenient. To add, the design is extremely user friendly making the process smooth.
5. You can see that almost all forms are within a shaded, rounded container to make it easily identifiable.
6. The home page includes all the main functionalities of the website in an easy to understand and navigate way for the convenience of the user. (Most users exit the website if they don't find their needed functionality of the website which reduces customer retention.)

### **Design techniques, tools used:**

Multiple design techniques have been employed, each with a specific goal, to guarantee the result is both functionally effective and visually aesthetic. Sitemap, Wireframe, Mockup, and Prototype (which I didn't do) are examples of these methodologies. Let's go over each one and discuss which and why they were chosen for this project.

**Sitemap:** It is the first indication of the project which means it should be clearly understandable. The Sitemap shows how different pages are linked and categorized hierarchically and provides a high-level perspective of the website's structure. It is not overly difficult, and it improves the user experience by making navigation more intuitive. The Sitemap assists in keeping a clear structure by understanding the number of pages and indicating which pages need to be developed, whether as different pages or within the same page. It also ensures that no links are broken and allows for easy transitions between pages, all of which contribute to a better user experience. The Sitemap is crucial for Travel Assist

since it helped me arrange the sites inside the website's design, ensuring that all relevant pages are included and correctly ordered. This structural clarity results in decreased expenses during the wireframing and development stages. I've always used Draw.io as the design tool in creating sitemaps in all my projects since it has a template for site maps, is simple to use and visually appealing.

**Wireframe:** Wireframes are simple, structural designs that describe the positioning of elements and site features but do not include any design elements like colors or graphics. They are concerned with functionality and user experience. I used it to define the functional layout before knowing any design specifics. This stage helped me create a user-friendly design which is especially essential for those with impairments, by emphasizing simplicity of use, logical navigation, and clear information presentation. Moreover, most structural design decisions were made at this stage. The tool that I used was proto.io. It had all the features (unlimited) that I needed like dummy text, image placeholder, ability to paste from other sites, buttons, maps, all form elements, all which are needed to depict the layout of all the functionalities of my system. This tool also made it easier when exporting the images by exporting all pages separately into one folder which made it easy to access. Moreover, I didn't have to download a specific software and could work within the browser where all the tools were easily accessible in one window. I've previously tried many wireframing tools and faced the issue when the number of elements exceeded the allowed amount I couldn't add more (lucidchart), I exceeded my license on Balsamiq (also not that flexible to use since too many tabs), and the rest of the wireframing tools (figma and Justinmind work best for mobile wireframing).

**Mockup:** This is a step closer to the development phase which makes the idea clearer. Mockups are more thorough than wireframes and incorporate visual design components like colors, typography, and images. They provide a more accurate representation of the result but are still static. I used Mockups to provide stakeholders with a visual depiction of the site that has all the features and design decisions to ensure that the website's appearance and feel are consistent with Queen Alia International Airport's identity and that the interface is structured and designed well to the users. I used the same tool, Proto.io which made it easy to move from wireframe to mockup by using the colors from the right navigation bar and the importing of images from other websites (this feature isn't found in most tools). This was used to gain feedback during the peer review which helped me further improve the website to meet requirements and standards.

**Prototype:** They are interactive representations of Mockups. They imitate user interactions and provide a realistic sample of the website's functionality. Prototyping was skipped for Travel Assist since the development using WordPress is very quick compared to developing the website from scratch, so testing the site's usability and interactivity won't be postponed too much. Moreover, I've previously created a website for services on WordPress so the flow and interaction of the users is clear to me.

The process of going from a site map to wireframes and then to mockups guarantees a comprehensive and user-centered design process. Each stage facilitates feedback, which are essential to the user-centered design process.

#### Design tools from security standpoint:

1. Sitemaps provide a clear overview of a website's structure and the relationships between pages, which helps uncover potential security issues such as unprotected

pages. It allows security teams to guarantee that secure regions are isolated and that access control mechanisms are deployed consistently throughout the site, hence improving overall website security (ex. non-logged in users can't have access to the service pages within the home page and the services page).

2. Wireframing can help detect potential security flaws early in the design process since it outlines the application's structure identifying where sensitive data may be displayed or entered, allowing for the early deployment of security measures such as encryption or access controls.
3. Mockups offer more details which enable security teams to examine and plan for security features such as input validation, error handling, and secure data presentation (ex. hiding text in password fields), lowering the chance of security issues in the interface.)
4. Prototypes are often used for testing with data which can impose a security issue if the data is collected from the prototypes and the data used is real data. I didn't use any.

As mentioned, the design of wireframes and mockups is critical in application development, not only for user interface and experience, but also for security. At the same time, it might cause issues to security. One big risk is the disclosure of real sensitive information through wireframes or mockups which can result in unintentional breaches (use dummy or generic data which I did). Another concern is failing to appropriately describe essential security features like authentication and encryption which might be the reason they are not in the final product (should be clearly mentioned in the design phase). Moreover, forgetting to visually depict password strength, anonymity and double entry in the wireframes and mockups will cause the developers to forget to develop them causing security issues. Furthermore, ignoring input validation in designs can expose the application to risks such as SQL injection or cross-site scripting, hence it is critical to include input validation checks in wireframes. Finally, a common issue is that excessively focusing on accessibility and usability causes security to be overlooked (I've mitigated this by trying to find a balance between usability and security). These security implications affecting mockups and wireframes are carried on to prototype.

### **Development technique, tools, and methodology**

I used the RAD low-code technique on my website to prioritize performance without losing customizability, which is critical for solving a pressing problem (needs a quick solution) that my website is trying to solve. Low-code provides a good balance of drag-and-drop simplicity and CSS and HTML customization, which is ideal for my application's complexity. This technique greatly accelerates development and allows continuous maintenance, which is critical given the ever-changing airport security regulations and rules, as well as the changing needs of people with special needs. Furthermore, low-code platforms include built-in features that make it easier to create apps that meet the demands of both businesses and users. Scalability and simplicity of maintenance are particularly important for my application, ensuring that the system can serve an increasing user base while reducing downtime. While RAD stresses rapid development, low-code also allows for necessary coding, resulting in a unique solution that cannot be achieved with an entirely no-code approach.

I've used WordPress as the development tool. This decision was driven by the necessity for a platform that allows for easy access from a variety of devices, including PCs and mobile phones. Given that arranging an appointment requires significant user input, a platform that is

both convenient and user-friendly on a PC is essential. Another major reason why I decided to use WordPress is its admin dashboard. When the user logs in using the admin credentials, they will have access to a dashboard to manage appointments (view, edit, cancel, confirm, set to pending, set to abandoned), view user information, view service animals, add service providers, and edit the server provider details (this level of control is not available in other tools). Moreover, my previous experience with WordPress highlighted its benefits, including the large number of plugins, excellent documentation (from plugin developers or on YouTube), and a variety of templates that simplify difficult functionality. This experience distinguishes WordPress as a more user-friendly solution than other low-code platforms like OutSystems or no-code platforms like Wix. Furthermore, WordPress provides responsive themes and plugins that are ideal for mobile and desktop use, meeting the project's requirements for a responsive, mobile-friendly design. WordPress stands out for its customizable plugins, which are essential for the Travel Assist application's specialized needs such as service animal registration, user profile administration, and complex booking functionalities. These plugins offer features like calendar integration, managing and preventing double bookings, which are difficult to implement on platforms such as Adalo and OutSystems. WordPress excels at scalability and traffic management, allowing it to handle peak travel demand without sacrificing performance. While many platforms provide data security, I chose to host the site on a service that supports WordPress and has an SSL certificate to ensure even more safety. Furthermore, WordPress's integration features are critical for connecting to external flight and ticket information databases. Moreover, it is efficient in implementing needs, like limiting the number of animals registered or controlling service requests for users who are signed in. A variety of customizable plugins (I used content control) can be tailored to meet the specific requirements make this possible. In contrast, systems such as OutSystems can achieve this but is complex. Moreover, implementing system limits may be difficult for platforms like Wix and Adalo, which won't meet the requirements causing a distribution in a critical process within a critical sector.

As for my development methodologies I used agile. This decision was driven by key factors that are specific to the nature of this project and my own development strategy. First, Travel Assist must meet the ever-changing needs of its key users - travelers with special needs. These demographics have unique problems that might change quickly, typically affected by changing airport restrictions. Agile's inherent flexibility enables a dynamic reaction to growing needs. Unlike Waterfall, which requires revisiting all developmental phases for each change, Agile allows for rapid and efficient adjustments. (Adaptability is critical given the project's nature, as requirements are not static). Secondly, my personal approach was to develop one feature at a time, research all possible plugin options, select the one that best meets the business needs and provides the appropriate level of customizability, learn the plugin, integrate it into the website, test it, and then repeat the process for the next feature. This is very consistent with the Agile concepts. This technique strikes me as more efficient and effective. It not only adheres to the iterative nature of Agile, but it also keeps me motivated and focused because completing each feature marks a step toward the overall project goal. This technique is especially useful in projects like Travel Assist, where each new idea has a direct influence on the user experience and must be carefully considered and implemented quickly. Overall, the ideas and practices of this methodology are directly applicable to the project's unique challenges and requirements.

### **Development methodology from security standpoint:**

The emphasis on rapid development and continuous iteration in Agile methodologies can expose organizations to serious security concerns. As efforts are focused on meeting functional features, the faster pace frequently leads to insufficient security testing. Frequent modification on the application (and skipping SDLC phases) worsens this problem by potentially creating vulnerabilities that are not immediately recognized. Furthermore, Agile's reduced emphasis on extensive documentation may result in inconsistent execution of security measures. The situation gets worse if the team lacks strong security expertise, resulting in the accumulation of unsolved security issues. To mitigate these risks, by using WordPress, which provides security in every plugin and the entire website, and while building each functional requirement, the non-functional requirement was kept in mind (mainly security ex. Access control).

On the other hand, traditional software development approaches, such as Waterfall, have longer development cycles, which allow for more complete security implementation.

I've decided to employ responsive when app development which allows the website to be accessed from desktop and mobile interfaces increasing the reach to the customer segment. It ensured that the customer experience is maintained no matter the device the user accesses the website on. The SEO score is increased because of responsiveness making the website more discoverable and reached by more people.

#### 4.4 Development Plan Presentation

##### 4.4.1 Peer review feed back

One of my colleagues asked: Why do I have to select the service again from the form when I already specified that I want a particular service from the home page?

Another colleague suggested: since the users of the website will have a lot of questions and it will be hard to deal with all the email requests, you can add a frequently asked questions section at the end of the contact page.

##### 4.4.2 Interpret the peer review feedback.

The first question asked by my colleague has identified a potential problem with the website's user journey. This comment shows that I didn't consider the user experience well enough for this functionality and the website's flow may be redundant or disconnected. She highlighted a user experience issue in which the website may ask for the same information many times, which can be wasteful or unpleasant by users. This made me change the way I looked at this functionality and encouraged me to consider the user experience more.

The second colleague identified an issue that might threaten customer relations if the volume of email inquiry is high (You can't possibly keep up). She proposed a proactive solution of inserting a frequently asked questions (FAQ) section at the bottom of the contact page. By having the frequent questions and answers in a FAQ area, the website may provide customers with immediate support without them having to send an email.

The first comment drew my attention to an opportunity that I hadn't considered before and an important component of the user journey and security on my website which is limiting service booking to logged-in users only.

Moreover, to improve user navigation and accessibility, I conducted user testing and discovered a similar pattern: users intuitively clicked on the images next to the services listed on the service page.

##### 4.4.3 Identify and evaluate new opportunities.

According to the first feedback, I understood from the question that the user experience wasn't considered to the correct extent. A negative user experience on the website can have a substantial influence on its business and reputation. When consumers encounter a website that is not user-friendly, they are likely to feel frustrated and never return. This decrease in traffic leads to poorer engagement, lower conversions, and can keep us from accomplishing the goal of enhancing inclusion in air travel. Furthermore, negative experiences can damage a website's reputation since users are more likely to share negative experiences with others (ex. through word of mouth).

I implemented many modifications to the website's service request process to improve user experience and stick to the feedback. First, I made three separate form request pages, one for each service each with the short code filtering the service accordingly. This method provides clarity, focus, and simplifies the process by decreasing redundant input. Furthermore, I modified the page titles to clearly highlight the service being requested, helping consumers to ensure they are on the relevant page for their needs.

According to the second feedback, including a Frequently Asked Questions (FAQ) section on the website improves user experience and operational efficiency. It allows users to find answers to their questions quickly and accurately, eliminating the need to wait for email responses, and ensuring information consistency. The availability of a FAQ section 24/7 suits users in different time zones by offering support outside of traditional work hours. Furthermore, a well-maintained FAQ section can help create trust by demonstrating care for user needs. Due to these advantages and lowering the email management efforts, I decided to implement this feature. I found, activated, and customized a plugin that gives an accordion style full customizable (questions, answers, and style).

Limiting the service booking to logged-in users only will increase efficiency of the website by ensuring that we have the full required details of those who are requesting the service (filled during registration). Moreover, it allows customisation by pre-filling the form details decreasing the need for users to re-enter information, improving user experience. I believe that this needs to be a non-functional requirement for the website as it is extremely critical to the operations (I added this to the SDD) and so implementing it should be made a priority. Fortunately, the integration process was relatively simple. I discovered a plugin called Content Control that allows me to easily implement access control to the website. Using this tool, I've added a new restriction that restricts access to the request services and add service animal pages to logged-in users only. I have changed the content of these sites to redirect to an alternative existing page for visitors who are not logged in, offering a seamless user experience while keeping the essential security standards.

Moreover, based on the user testing results, I added hyperlinks to these images to the service request page to streamline the user experience. This improvement not only corresponds to natural user preferences, but also makes the procedure more intuitive and user-friendly.

## 4.5 Application Development

### 4.5.1 Database

Databases are used to efficiently store, retrieve, manage, and modify data, to handle massive volumes of information in a structured way. Databases are used to interact with essential information within the website like the appointment details (face to face or for the services) and the service animals detailed available.

One advantage of WordPress is the content management system integrated within the admin dashboard. Each plugin I used to provide functionality provided a MySQL (open-source, relational) database to manage all the data that is used in the functionality. Below are screenshots of the database structures used:

#### 1) User database:

Can edit, delete, view information about the users. I used a plugin in extension to the existing WordPress user database as it is limiting and doesn't meet the functional requirements.

Attributes: username (PK, not null), user's first (not null), last name (not null), email (not null, unique), phone number(not null, unique), age (not null), gender (not null), nationality (not null), Passport number (not null), expiration date (not null), Ticket number, nature of special needs (not null, available: mobility impairment, visual impairment, hearing impairment), Mobility requirements (available: wheelchair, cane, crutches), medications used, emergency contact name (not null), emergency contact relation (not null), emergency contact phone number(not null), Password (not null) and repeated password (not null).

All (3)   Administrator (1)   Subscriber (2)					Search Users
Bulk actions	Apply	Change role to...	Change	3 items	
<input type="checkbox"/> Username <span style="font-size: small;">▲</span>	Name	Email <span style="font-size: small;">▼</span>	Role	Posts	
<input type="checkbox"/>  admin	—	20210004@htu.edu.jo	Administrator	1	
<input type="checkbox"/>  newsmn	smn smn	newaccziadat@gmail.com	Subscriber	0	
<input type="checkbox"/>  TestUser Edit   Delete   View   Send password reset	John Doe	johndoe@example.com	Subscriber	0	
<input type="checkbox"/> Username <span style="font-size: small;">▲</span>	Name	Email <span style="font-size: small;">▼</span>	Role	Posts	

Figure 40: Users database

When the user is edited, the extra attribute can be viewed and modified.

## 2) Service appointments database:

Can be filtered by location, service, worker, from and to date, status and searched. Data can be edited, deleted, viewed, and cloned.

Attributes: id (PK, autoincrement), location of service (not null), service (not null), worker (not null), customer information (name (not null), phone (not null), email (not null), ticket number (not null), mobility level (not null, available: I can shift seats on my own, I require assistance in shifting seats), communication method preferred (not null, available options: sign language, written communication, verbal communication), communication needs, level of visual impairment (not null), preferred assistance method (not null)).

Appointments					
Location :	Queen Alia International	Service :	-	Worker :	-
Status :	-	From :	01/08/2024	To :	01/14/2024
Quick time filter:		Select period			
<a href="#">+ Add New Appointment</a>	<a href="#">Refresh</a>	Sort By:	Id	Order by:	desc
Id / Location / Service / Worker	Customer	Description	Date & time	Status / Price / Created	Action
#29 Queen Alia International Airport Wheelchair Assistance Bob	Kareen 07750595659 newaccziadat@gmail.com USILN I require assistance in shifting seats Written communication none Verbal assistance	none	January 11, 2024 - 05:00 January 11, 2024 - 06:00	pending 0.00 January 5, 2024 10:10	<a href="#">Edit</a> <a href="#">Delete</a> <a href="#">Clone</a>

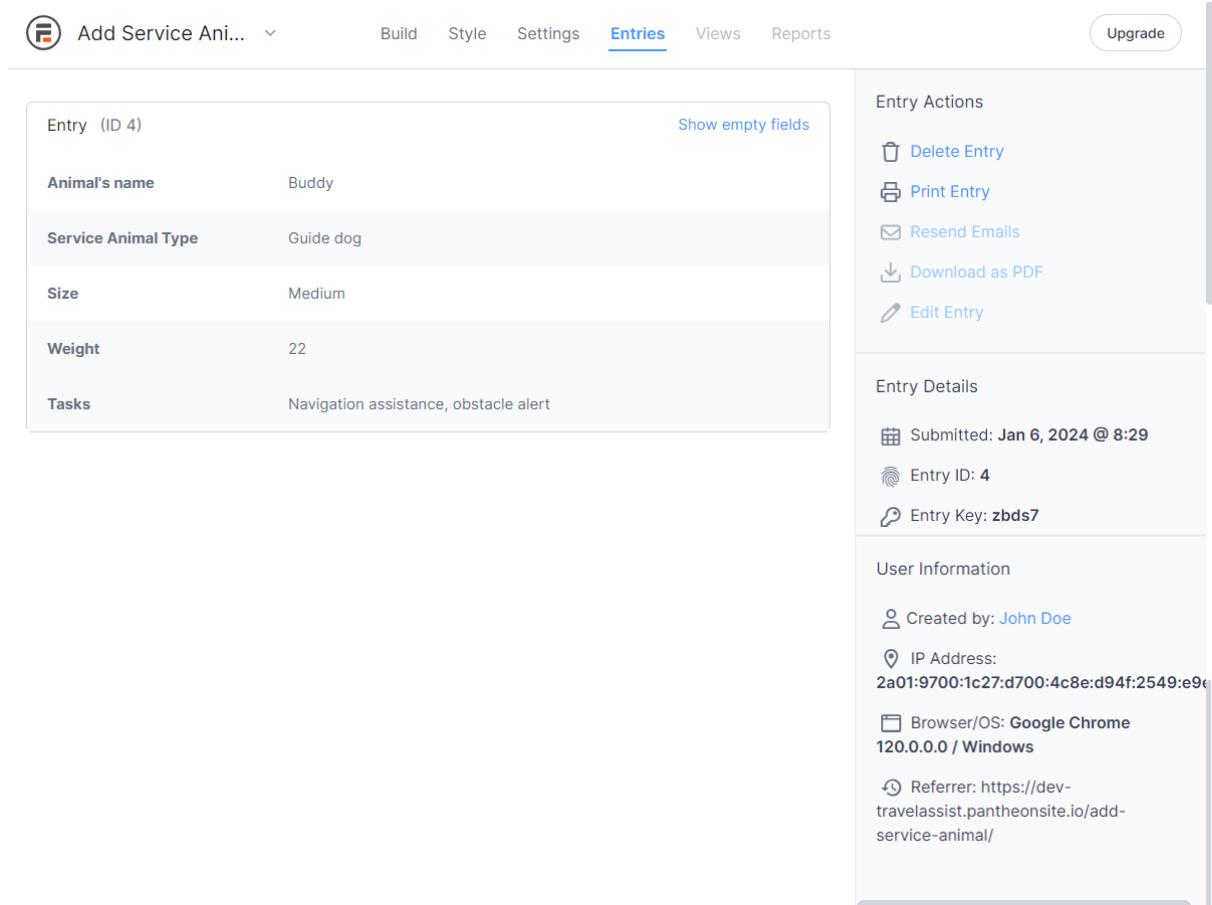
Figure 41: Structure of database for service appointments.

3) Add service animal database:

Attributes: id (PK), owner (username, FK, not null), Service animal type (not null, available: guide dog, hearing dog, mobility assistance dog, psychiatric service dog), size (not null) and weight (not null), animal name, description, breed, age, specific tasks the service animal is trained for, emergency contact person and their contact information

ID	Form	Entry Name	Created By	Entry Status	Entry creation date	Entry update date	IP
4	Add Service Animal	Buddy	TestUser	SUBMITTED	January 6, 2024 at 8:29 am	January 6, 2024 at 8:29 am	2a01:9700:1c27:d700:4c8e:d94f:2549:e9e5

Figure 42: Table to view animals.



The screenshot shows a web-based form for viewing a service animal entry. The main content area displays the following fields and their values:

Animal's name	Buddy
Service Animal Type	Guide dog
Size	Medium
Weight	22
Tasks	Navigation assistance, obstacle alert

On the right side, there is a sidebar with the following sections and details:

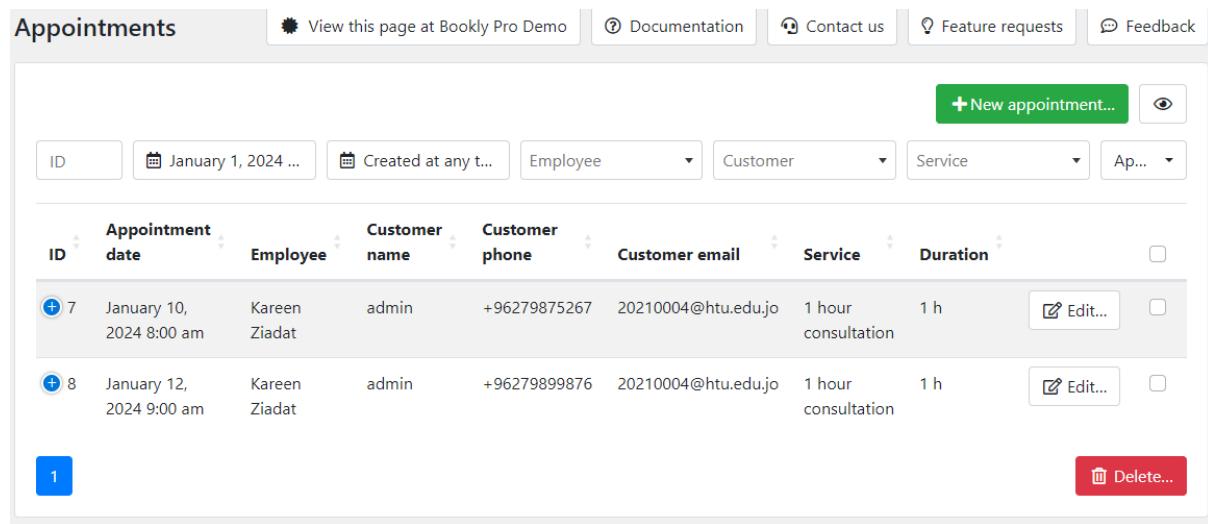
- Entry Actions:**
  - Delete Entry
  - Print Entry
  - Resend Emails
  - Download as PDF
  - Edit Entry
- Entry Details:**
  - Submitted: Jan 6, 2024 @ 8:29
  - Entry ID: 4
  - Entry Key: zbds7
- User Information:**
  - Created by: John Doe
  - IP Address: 2a01:9700:1c27:d700:4c8e:d94f:2549:e9e5
  - Browser/OS: Google Chrome 120.0.0.0 / Windows
  - Referrer: https://dev-travelassist.pantheonsite.io/add-service-animal/

Figure 43: When view is pressed.

The details of the user who added this service can be viewed by clicking the hyperlink in the created by.

## 4) One to one appointment

Attributes: consultation duration (not null, available 1 hour consultation, 2-hour consultation), date (not null), days (not null), time (not null), name (not null), phone number (not null), email address (not null), notes.



ID	Appointment date	Employee	Customer name	Customer phone	Customer email	Service	Duration	Actions
7	January 10, 2024 8:00 am	Kareen Ziadat	admin	+96279875267	20210004@htu.edu.jo	1 hour consultation	1 h	<button>Edit...</button> <input type="checkbox"/>
8	January 12, 2024 9:00 am	Kareen Ziadat	admin	+96279899876	20210004@htu.edu.jo	1 hour consultation	1 h	<button>Edit...</button> <input type="checkbox"/>

Figure 44: Database for one-on-one appointments

4.5.2 Recording of application

<https://www.youtube.com/watch?v=D5F9YsoH6Mg>

## 5. Testing

### 5.1 Testing plan

#### Test case 1: integration test (of two plugins)

- **Test ID:** 1, **Name:** TC\_AddServiceAnimal\_001
- **Module name:** Service Animal Registration
- **Test data:** Animal name: buddy, Type: Guide Dog, Size: Medium, Weight: 22 Kgs, Specific Tasks: Navigation assistance, obstacle alert
- **Assumptions or preconditions:** User has internet connection, is on home page, already logged in, and has all the required information about the dog.
- **Test priority:** Medium
- **Test scenarios:** User wants to add the service animal so that they can bring it with them to the airport.

#### Test case 2: functionality

- **Test ID:** 2, **Name:** TC\_WheelchairRequest\_001
- **Module name:** Service Request
- **Test data:** Service provider: Bob, Date: 31<sup>st</sup> January 2024, Time: 3:00, Name: Test, Phone: 1234567891, Email: [johndoe@example.com](mailto:johndoe@example.com), ticket number:TK12345, mobility level: I require assistance in shifting seats, Communication metho preferred: Verbal communication, Level of visual impairment: high, preferred assistance method: verbal assistance.
- **Assumptions or preconditions:** The user is already logged in, has chosen the wheelchair assistance service.
- **Test priority:** High
- **Test scenarios:** User wants to book an appointment to get the wheelchair assistance for their upcoming trip.

#### Test case 3: Performance

- **Test ID:** 3, **Name:** TC\_EmailNotification\_001
- **Module name:** Email Notification System
- **Test data:** Service provider: Bob, Date: 31<sup>st</sup> January 2024, Time: 3:00, Name: Test, Phone: 1234567891, Email: [johndoe@example.com](mailto:johndoe@example.com), ticket number:TK12345, mobility level: I require assistance in shifting seats, Communication metho preferred: Verbal communication, Level of visual impairment: high, preferred assistance method: verbal assistance.
- **Assumptions or preconditions:** The user is already logged in, has chosen the wheelchair assistance service, page is fully functional, and the email entered is valid.
- **Test priority:** High
- **Test scenarios:** User wants to book an appointment to get the wheelchair assistance for their upcoming trip.

## Test case 4: UI

- **Test ID:** 4, **Name:** TC\_FAQ\_001
- **Module name:** Contact us.
- **Test data:** none.
- **Assumptions or preconditions:** The user is on the website (in contact page), user is already logged in.
- **Test priority:** Medium
- **Test scenarios:** User is still confused about something.

## Test case 5: database

- **Test ID:** 5, **Name:** TC\_ConsultationBooking\_001
- **Module name:** Consultation Booking
- **Test data:** Duration: 1 hour, Availability: From 2024-01-10, Days: Tuesdays and Thursdays, Time Slot: 2:00 PM to 4:00 PM, User's Name: John Doe, Phone Number: 1234567890, Email: [johndoe@example.com](mailto:johndoe@example.com).
- **Assumptions or preconditions:** The user is on the website (in contact page) and is already logged in.
- **Test priority:** High
- **Test scenarios:** User is still confused about something but prefers to meet in real life.

Test id	Steps of Happy path	Steps of Alternative path	Steps of Exceptional path	Expected result	Actual result
1	Go to the service animal form by clicking the add here button at the end of the home page. Fill in all required and some optional fields (from the test case) correctly and submit the form.	User reaches the form through slide 2 of the slider in the home page and pressing the register now button.	Users forget to fill the non-optional fields; they will get an error message. Or they are not logged in which will redirect them to a page with an error message and not allowed access.	When the user submits, it creates a new entry in the service animal form plugin, when this entry is viewed, the user information created by should be correct and when the name (hyperlink) is clicked, I should be redirected to the user information page.	All paths worked as expected. Pass.
2	Select the wheelchair assistance service from the home screen from the three-card available, fill in all required fields (from the test case), submit the request.	Go to the services tab and choose the service from there by clicking the button or by clicking the image of the wheelchair assistance.	Attempt to access the request service page without being logged in, attempt to submit the request with missing required fields, enter invalid data, or try to book an unavailable appointment (same service, worker, date, and time)	Drop down lists are functional, Calander is functional, email input field checks invalidity, and submit button works shows a confirmation message and redirects to home page.	All paths worked as expected. Pass.
3	User fills the service form with the details, admin confirms appointment from dashboard.	User books an appointment for another service.	Enters an invalid email. Won't receive a message upon confirmation and won't be able to cancel appointment.	Service is confirmed, User receives email.	All paths worked as expected. Pass.
4	User goes to contact page, sees the FAQ section, presses on question, answer appear.	Logged out user.	User hovers on the question instead of clicking.	The FAQ should have consistent background, text font and color. It should expand when a question is chosen and shrink when another question is chosen, or the question is clicked again. It should be responsive on a mobile and website.	All paths worked as expected. Pass.
5	User selects consultation duration then chooses from the available dates and times, reviews and modifies their pre-filled personal information is auto filled (All from test case), sees the summary, submits the booking, and sees a confirmation message.	Logged out user so no pre-filled inputs.	User tries to book an unavailable date/time or forgets to/ enters invalid phone number (text) or email.	Successful booking of the consultation with a confirmation message, appointment should be added to the database of face-to-face consultation.	All paths worked as expected. Pass

Table 3: Test cases plan table

## 5.2 Performance review

The initial issue that the website is trying to address is that travellers with special needs often need to talk to many people and in advance to book the service that they need to have an easier travelling experience. This is time consuming, complicated, and often faces the issue of miscommunication between all the parties involved (sometimes causing double booked appointments or cancelled appointments). This current structure decreases inclusivity in air travel and causes frustration for travellers with special needs trying to get the most basic needs met.

The website developed eliminates this issue by creating a centralized website where the user can find all the information that they need about the services provided by the airport, and once they are logged in, they can book any of the services provided or register their service animal (airport policy states that service animals can only be on flight when they are preregistered). Moreover, the website offers multiple communication channels in which the traveller can communicate with staff in case they are still unsure about some details or want to discuss specific details about their case. Through the contact page, the traveller can send questions via email by submitting a simple form on the website, or book a one-to-one face-to-face consultation, or simply find the most frequently asked question in an accordion layout.

Travellers can view a description of all the services and any important notices like time constraints (having to book service 24 hr in advance to ensure validity) or other constraints set by the airline (ex. Only allowing two animals on board of the same flight). This reduces the need for multiple points of contact to understand airport policies and the services and offers a centralised place to achieve this directly solving the problem.

The functionalities that are achieved through the website and how they contribute to solving the problem:

1. The traveller can book communication support: Often hard of hearing individuals find it hard to navigate the airport. The person can book an appointment to get communication support with a competent service provider making their journey through the airport easier. The form is simple, can be filled within minutes, and includes all the information needed to provide the service efferently and aligns with airport regulations.
2. The traveller can book guidance support: Often visually impaired individuals face difficulties in navigating the airport on their own. They are often confused about who to contact (their travel agent, airline, or the airport) to get help. The website provides an easy to fill form that asks for all the necessary details to cater to the user's need and ensure simple travel. (again, reducing multiple points of contact and time)
3. The traveller can book wheelchair assistance: some passengers need help with navigating the airport on a wheelchair or require the airline to provide a wheelchair. The user might not know who to contact to get this service and the website solves this issue.
4. The traveller can register their service animal information: the airline requests that the service animal should be registered two days in advance with the necessary details. The traveller might not know this and might not know where to go. The website offers a simple form to register the animal

with all the details that the airline needs. (one step to get service unlike traditional method)

5. The traveller can ask questions through emails, book a face to face consultation, or simply access frequently asked questions. This will remove the confusion that the user faces directly solving the problem/

The user interface of the website is intuitive and easy to use. This is especially critical for the user demographic.

1. The slider has left and right arrow and dots signifying which slide the user is on. This helps achieve the slider's goal of displaying a summary of the goal of the website.
2. Consistency in colours (of text and images) and font through the whole website making it look professional.
3. Not full of text and mostly bullet points are used to get the most important information across without overwhelming the users.
4. It is responsive on mobile devices and pcs.
5. The sign up and edit form are kept as a simple form avoiding too much graphics.
6. The login is a user-friendly pop-up.
7. Booking a face-to-face consultation is shown in a user friendly with clear indication of the process and only asks for the essential information needed.
8. The accordion format of FAQ makes the page less cultured but still provides the valuable information needed.
9. The service request form provides a calendar that shades the dates that are unavailable making it easy for the user to understand.
10. The navigation bar doesn't include too much text and relies on standard icons to represent login, register, and edit profile.
11. non-logged users are not left wondering why they can't access the request service and add service animal pages but are redirected to a page with the documentation.

The systems functionalities were all tested to make sure no double booking occur, only logged in users can request service (which meets the non-functional requirement of the system), the database accurately reflect the addition of appointments, user information. It also accurately reflects the modification of user account details and removal of the account. Also, all this data is backed up daily.

The website also ensures emails are sent correctly to the admin and the traveller.

### 5.3 Overall Business Application review

#### 5.3.1 Critical review for all application development phases

##### **Design phase**

Moving on from the analysis phase, I had a clearer vision on what I wanted to achieve from the website to directly solve the problem statement identified, however I couldn't yet imagine how I can bring this idea to life. This is when the design phase comes into play. I started by creating a sitemap, the steps that I followed to do so was identifying the main functionalities that I want achieved through the website, deciding in which page each of these functionality will be positioned and then the hierarchy of these pages since it doesn't make sense to have all the pages in the top navigation bar (will make it crowded) decreasing the user experience and interface which should be carefully considered throughout this stage. Producing the sitemap helped me get the first idea on what pages I should include and the structuring of the website moreover, when implementing access control, I went back to this structure to understand where the pages are located and secure correctly. The next step of the design phase was to know the structure of each of the pages previously identified this was done through the wireframe that provided me with a stepping stone closer to the actual development. Wireframing allowed me to only focus on the user experience and the structure of pages without having to worry about design decisions like colours, typography, finding an appropriate image (all which take time). However, these are critical to the success of the website by ensuring a user-friendly interface. Moving on from wireframes, and focusing more on the themes, and more details now that I established the layout, I developed Mock-ups. After establishing the mock-ups many chose to develop prototype to clearly depict the user interaction with the system. The Travel Assist website doesn't include a lot of focus on interaction and can be explained to stakeholders as it only consists of buttons that take to other pages or display validation messages. Overall, I believe that there is better use for the time needed to develop a prototype that won't be used later to show something that can be easily explained (website not interaction based). To add, by this stage I knew that the development tool I will use (WordPress) which is a low-code rapid application development tool that doesn't take long to develop or apply changes based on feedback. If I decided to code from scratch (unrealistic given time, development team size (1), and experience) I would've chosen to do a prototype since it costs more to apply changes when developed than before after getting the feedback on the design stage.

My website is extremely reliant on data therefore Low-level design is equally as important as high-level design. I used ERD to depict this since it illustrates the entities (people, objects, or concepts), their attributes and how they interact with each other. The ERD was done based on the functional requirements and will direct me in choosing the most appropriate plugin. Another advantage of the ERD was that it helped me connect the functionalities with each other and to see the website as a cohesive system rather than multiple independent functionalities improving the performance of the website.

The tools that I chose to do the sitemap, wireframe, mock-up and ERD were all explained in previous section however the main reason is that I've used them in previous course to develop websites and know that they are appropriate.

The design phase clearly shows security measuring in the wireframes and mock-ups. This was achieved by showing the password strength indicator under the password field, hiding the data in the password field, and showing

the page that will be shown when the user tries to access pages when they aren't controlled. The clear description of security measures made it clearer for the development phase that this should be taken into consideration improving the performance of the website. The UI/UX was kept in mind throughout the entire design process making sure that the users have a smooth experience on the website improving the performance of the website. The decision of using sliders might cause issues to the website's performance making it slower as it loads. The clear structure of the ERD ensures all features are integrated well within the website and the performance is maintained across the website's data. The choice of using FAQ instead of relying on emails decreases traffic to the email service improving performance.

## Development phase

After having an extremely clear idea of how every page was going to look like, I started implementing this idea. There were many decisions like the development methodology and tool. Deciding on the development methodology (Agile) relied on factors like my personal approach of developing one feature at a time, the short time, the non-fixed budget, and the need for quick iterations and testing. It was easy to know which tool to use since I've previously done a similar website using WordPress, so I knew the functional requirements were feasible using plugins. The next step was to browse all the plugins available within WordPress, compare them with my ERD, Wireframes and Mock-ups and chose the most appropriate one for each functionality (process consisted of iteratively discovering plugin, learning about it, understanding customization limitations, and evaluating if it meets the requirements until I found the perfect one). Since I was limited to using the plugins, the mock-ups and wireframes had to be modified to match the developed website. Looking back, I think it would've been better to create the initial wireframe and mock-ups on paper to reduce time since using plugins within WordPress limits customization of the website.

The reliance on plugins ensured that the features were thoroughly tested and the reviews on the plugin showed that the owners of the sites didn't face any issues with the performance of this plugin, I was also aware that I shouldn't use too many plugins as it might break the site or cause performance problems. I was also aware that the use of Agile might cause security problems affecting the website's performance so from the beginning of the development, I made sure to use a reliable hosting site that provides an SSL certificate (paid) to encrypt incoming and outgoing , only used reliable plugins, used an access control plugin, used an account management account that ensures account passwords are minimum medium strength and a certain length, the forms only asked for necessary information, and I ensured that the admin account that has access to the data is protected with a strong password. To enhance performance, I ensure responsiveness during development to allow cross-platform use. I knew that scalability is very important as it affects performance of the website and was a key factor in choosing the hosting service and WordPress as they are known for scalability.

## Testing phase

The design phase allowed me to communicate with stakeholders and get feedback ensuring that my developed solution aligns well with the functional and system requirements. After the application was developed, I had to verify that the website performs as intended. The first step was to remember the functionalities that are achieved through the website, then I wrote 5 test cases each testing a different aspect (functionality, integration, performance, UI, and database ensuring all aspects are performing their job) that directly affects the websites performance if any of them fails. I incorporated testing a lot during the development phase in line with the agile methodology where I used to test the feature right after it was developed. I simply cannot move onto another feature without ensure that the feature is tested since a lot of them are connected and failure in one can affect the performance of the entire website. Testing helped me ensure that all the features work as intended and if not, I'd go back and fix them to ensure performance.

Now that the application is complete, it is important to see how the risks that were anticipated have affected or were mitigated:

### Design phase risks:

- **Inadequate understanding of user requirements:** during the analysis and design phase I made sure to understand the current services that are applied by the airport. The idea is that these services are created based on the user requirements which have been around for a long time. Further user research should be done to validate the needs; however, this wasn't possible due to the short time available to complete the project.
- **Security and privacy:** As mentioned before this was mitigated by carefully considering security features and adding them to the wireframes, mockups preventing them from being forgotten in the development phase. Moreover, I was aware of the security implications and benefits of using sitemaps, wireframes and mockups which helped me use them to my advantage from a security standpoint and follow best practices to avoid the security implications.
- **User Interface:** I've taken a course in UI/UX before going into this project and I made sure to implement as many principles as possible.

**Development phase risks:**

- **Failure to meet accessibility standards:** This wasn't considered extensively during the development stage and looking back I should've implemented it better.
- **Scalability issue:** All development tools used (WordPress and the hosting website) are extremely scalable.
- **Technical Challenges:** This is technically possible but needs a basic plan. I've reduced this risk by setting the appointment to pending initially and manually set to confirmed once the ticket is verified. While it isn't the most efficient way to approach this, it can be improved and is achieving the system requirement.
- **Time:** the risk in the design phase was avoided so this didn't cause an issue in the development phase.
- **Hosting tool going out of business:** I learnt that this is highly unlikely, and it is doable to switch host.
- **Any of used plugins is no longer maintained by the supplier:** This is also unlikely if WordPress is supported, and it is easy to switch to other plugins that do (they might be more expensive).

**Testing phase risks:**

- **Inadequate user acceptance testing (UAT):** testing wasn't done by people with disabilities due to the short time which can cause some issues to go undetected.

### 5.3.2 Application evaluation and future development

#### Strengths:

1. **Centralized access:** the website consolidates services for passengers with special needs into a single platform, streamlining the process of requesting assistance, making it more efficient and user-friendly.
2. **Comprehensiveness:** The platform offers comprehensive services such as wheelchair assistance, instruction for visually impaired passengers, and registration for service animals. This inclusivity improves the airport's accessibility and user experience.
3. **Interactive information presentation:** Utilizing sliders, short descriptions, bullet points, and an FAQ accordion makes the website informative and easy to navigate. This interactive approach enhances user engagement and comprehension.
4. **Streamlined service request process:** The user-friendly process for requesting services, which includes clicking on service images or buttons and filling out efficient forms, contributes to a positive user experience.
5. **Data integrity, and backup:** Reliance on WordPress, known for its reliability in ensuring data integrity. Regular backups are done through a plugin that automates the process by backing up the site, plugins and the database daily further enhancing data protection and minimum downtime.
6. **Hosting:** The website currently hosted on pantheon hosting. This SAAS provides an SSL certificate for custom domains which need to be bought, so in the future it is wise to buy a customizable domain to ensure encryption and data security. The database and entire website is backed up daily ensuring no critical information is being lost and no appointments are being deleted affecting the travellers experience which affects the entire goal of the application.
7. **Effective account management:** Account creation, management (modify details and delete account) were tested fully and the databases fully reflect every change. Services request and addition of service animals are only allowed to be done by logged in users which ensure the non-functional constraint. The login page is a simple user-friendly pop-up. This same plugin used for the login has a password recover option that needs payment so for the future improvement, it is recommended to have that feature.
8. **Restricted access to services:** Limiting service requests and animal addition to logged-in users supports security and operational efficiency.
9. **Reliable email communication:** The functionality of the email system, which includes confirmation messages and the ability to cancel appointments, adds to the platform's efficiency.
10. **High availability:** Hosting on a reliable platform like Pantheon ensures constant website availability, which is critical for a service-oriented platform.
11. **Clear communication tools:** Passengers and service providers can communicate effectively with an email notification system, FAQ area, and one-on-one consulting options.

**12. Responsive design:** The platform's flexibility with both desktop and mobile devices means that consumers can access it from any device they like.

### **Weaknesses:**

- 1. Dependency on paid plugin features:** (not coding the website) there are certain limitations. Specific validation checks, such as restricting users to two service animals and confirming ticket numbers, are currently only available in paid plans of the plugins. As a result, these checks are not enforced automatically. Instead, travellers get a message after submitting the service animal form that they are only permitted to have two service animals. To deal with ticket validation, appointments are originally set as pending. The admin checks the ticket number manually and if it is valid, he/she sets the appointment status as confirmed. For the future, it is recommended that these paid plans are brought to ensure automation, reduce the possibility of error, and reduce manual labour.
- 2. Dependency on third-party plugins:** The dependency on WordPress and its plugins restricts flexibility and scalability affecting long-term viability and adaptability.
- 3. Potential accessibility issues:** Despite being user-focused, the design lacks accessibility elements for individuals with impairments, which is important given the target population.

### **New opportunities for future development:**

To enhance the travel experience for individuals with special needs, two significant enhancements are proposed for the Travel Assist website. The first is an expansion of the website's capabilities to help these travelers organize their entire trip after arriving to QAIA. This includes features for organizing ground transportation from the airport, booking disability-friendly accommodations, and finding accessible restaurants. The second development involves also creating a mobile application to provide further assistance when in the airport. This will include augmented reality (AR) guidance for hard-of-hearing passengers as well as a feature that allows the user to send their position in the event of an emergency or when aid is required.

Expanding the website's functionality to include thorough trip planning is an important step toward inclusive tourism in Jordan. The website will help solve a fundamental barrier for tourists with special needs by making it easier to arrange ground transportation, lodging, and accessible food options. This comprehensive planning tool guarantees that their unique needs are evaluated and met from the moment they arrive at the airport to the time they stay and explore Jordan. This upgrade is about more than just convenience; it is about making Jordan's rich cultural and historical experiences accessible to everybody, removing barriers that may have discouraged disabled people from visiting.

Transforming the website into a mobile application adds another level of accessibility and convenience. The use of augmented reality (AR) for hard-of-hearing tourists represents a novel application of technology to aid in navigation and information distribution. AR may overlay visual guidance and data in real time, providing a straightforward way to navigate the airport. For people who struggle with audio-based instruction, this can considerably improve their trip experience removing the need for another person to help them (some people don't prefer that). Furthermore, the ability to share one's location with a button press is an essential safety feature. It provides peace of mind not just to the traveler, but also to their families and service providers. This feature ensures that help is provided quickly and effectively to the traveler's location in the event of an emergency or other scenario requiring assistance.

## 6. Implementation and support

Implementing the Travel Assist system and incorporating it into the business operations of QAIA necessitates a comprehensive plan that addresses everything from implementation to support and maintenance.

**Implementation plan** (This phase lasts until the system meets user requirements and is ready for production use.)

#### Pre-Implementation plan:

1. **User notification:** Contact the Higher Council for the Rights of Persons with disabilities (HCDI) to help us get numbers of as many people with disabilities as possible (there are approximately 1.25 million people with disabilities in Jordan alone and the website doesn't only target Jordanians rather all people with special needs arriving at the airport (statistic hard to obtain)). After obtaining this number send a text message from a credible source (ex. The airport, or government entities like (ministry of Health: MOH)) to them announcing the deployment of the website and a small description with the link.
2. **Staff notification:** Through an email, notify airport staff and airline companies about the deployment of the website, the benefit of the website and how it will affect each of them.
3. **Stakeholder engagement:** If the email wasn't enough (it likely isn't), it is recommended that a session is conducted with the stakeholders (a representative from each airline, airport staff, service provider (or a representative of the team)) to give extensive details on the process and align the Travel Assist system with the airport infrastructure.

#### Implementation plan:

1. **User training:** Conduct several comprehensive training sessions for the airport and airlines staff on how to use the travel assist admin dashboard and how to trouble shoot common issues with passengers. The number of training sessions will vary depending on the technical competency of the individuals so some might require more than others.
2. **Installation:** Access the website URL on all PCs of the dedicated staff that will be involved in the process (Airport help desks personal, airlines, anyone that manages the service providers), log them in using the admin credential to get access to the admin dashboard (save login credentials) and bookmark the website so it is easily accessible.
3. **Integration into daily work process:** Integrate Travel Assist into the airport's daily operational workflow by clearly identifying how and when and how the system will be accessed and how to use the information provided by the website.
  - a. **Suggested plan for airport personnel:** view the appointments two days in advance, status is confirmed, filter by worker, and provide the worker with screenshot of their appointments through emails. (any appointments booked less than two days prior to the flight won't be taken into consideration as per airport regulation).

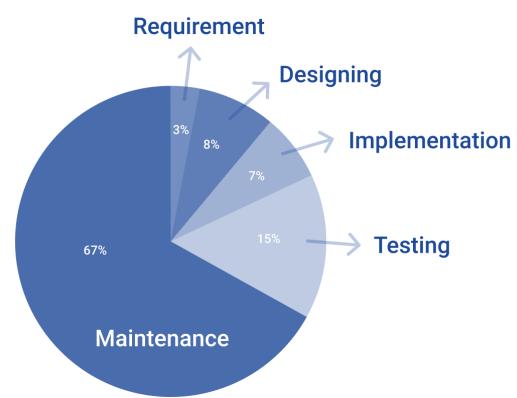
#### Support and Maintenance plan

#### Post launch support:

1. **Enable an effective feedback mechanism:** allow both staff and travelers to give feedback and address user feedback and suggestions to better meet user needs, improving performance, or meeting additional user requirements. This can be done by providing the travelers who benefited from the service with a survey near the end of their flight. It can include ratings and open-ended questions. (Flight attendants should be available to help them fill in the form). To get feedback from staff, they will be sent a survey through email.
2. **Continuous bug fixes:** have a team in place, provide them with the documents needed (ex SDD), and always notify them of any bugs that are reported by the staff or users.
3. **Regular updates:** Constantly update the plugins on the website.
4. **Regular audits:** to verify the integrity and performance of website.
5. **Future advancements:** always have a plan on implementing the future advancements provided in this document to catch up with the new technological trends or provide a more well-rounded solution.
6. **Security and regulatory compliance:** Conduct frequent security audits to ensure data protection and privacy is maintained and stay up to date on regulations.

Consider implementing DevOps methodology after application deployment. It promotes continuous improvement by providing frequent and consistent updates, leveraging approaches such as continuous integration and continuous delivery (CI/CD). This method improves collaboration between development and operations teams, resulting in faster responses to user feedback and better operational efficiency. After deployment, DevOps enables automatic deployments, effective issue management, and real-time application monitoring, ensuring the application's high availability and reliability. Furthermore, it provides strategies for performance tuning and scalability, which are critical for maintaining and improving the program as user interactions increase.

Keep in mind that the maintenance stage of SDLC costs the most.



Having a plan ensures that the Travel Assist system is implemented carefully, with a strong focus on user needs, particularly for disabled passengers. Moreover, continuous support and iterative enhancements will be required to keep the system relevant and efficient in improving the travel experience at QAIA.

## References

- Parker, K. (2021). *What Is a Software Design Document (SDD)?* [online] Technical Writing is Easy. Available at: <https://medium.com/technical-writing-is-easy/what-is-a-software-design-document-sdd-6a3ea3a953b2>.
- www.igi-global.com. (n.d.). *What is Actors or Stakeholders | IGI Global.* [online] Available at: <https://www.igi-global.com/dictionary/actors-or-stakeholders/108047>.
- Bears, D. (2022). *What Are The Highest Risks In The Mobile App Development Industry, And How To Avoid Them?* [online] Blogs. Available at: <https://darkbears.com/blog/what-are-the-highest-risks-in-the-mobile-app-development-industry-and-how-to-avoid-them/>.
- Visual-paradigm.com. (2019). *What is Component Diagram?* [online] Available at: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-component-diagram/>.
- Visual Paradigm (2019). *What is Class Diagram?* [online] Visual-paradigm.com. Available at: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/>.
- Visual Paradigm (2019). *What is Use Case Diagram?* [online] Visual-paradigm.com. Available at: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/>.
- www.mindmanager.com. (n.d.). *What is an Activity Diagram? | MindManager.* [online] Available at: <https://www.mindmanager.com/en/features/activity-diagram/#:~:text=An%20activity%20diagram%20is%20a>.
- www.lucidchart.com. (2013). *Benefits of Creating a Visual Sitemap | Lucidchart Blog.* [online] Available at: <https://www.lucidchart.com/blog/7-unexpected-ways-to-use-sitemaps#:~:text=A%20visual%20sitemap%20can%20help%20you%20quickly%20draw%20out%20the>
- Meir, D. (2019). *What is a Wireframe and How to Create Wireframes.* [online] Wix Blog. Available at: [https://www.wix.com/blog/what-is-a-wireframe-and-how-to-create-one?psafe\\_param=1&utm\\_source=google&utm\\_medium=cpc&utm\\_campaign=16242205830](https://www.wix.com/blog/what-is-a-wireframe-and-how-to-create-one?psafe_param=1&utm_source=google&utm_medium=cpc&utm_campaign=16242205830)
- OS-System. (2020). *Top 4 Software Development Methodologies: Comparison, Differences, Pros and Cons.* [online] Available at: <https://os-system.com/blog/top-software-development-methodologies-comparison-differences-pros-and-cons/>.
- www.softwaresecured.com. (n.d.). 7 Agile Software Development Habits that Produce Security Concerns. [online] Available at: <https://www.softwaresecured.com/post/7-agile-software-development-habits-that-produce-security-concerns>.
- en.royanews.tv. (n.d.). 1.25 million people with disabilities in Jordan. [online] Available at: <https://en.royanews.tv/news/48288/2024-01-11#:~:text=The%20Secretary%2DGeneral%20of%20the> [Accessed 24 Jan. 2024].