horizontal line

Software Development Plan

**19/04/2023**

Version 1.2

**Presented To:**

**DR/Mohamed Ramadan**

**Submitted By:**

*Mohamed Osama*

*Yousef Ibrahim Mammon*

*Mohamed Gamal*

*Abdel Rahman Ahmed Mohamed*

*Ahmed Mohamed Elsaghier*

*Mahmoud Haggag Mahmoud*

# 

# 

[**REVISION HISTORY** 3](#_Toc135517374)

[**1.** **PRODUCT DESCRIPTION** 3](#_Toc135517375)

[**2. TEAM DESCRIPTION** 4](#_Toc135517376)

[**3. SOFTWARE PROCESS MODEL DESCRIPTION** 5](#_Toc135517377)

[**4. PRODUCT DEFINITION :Context Diagram** 6](#_Toc135517378)

[**Chatbot** 7](#_Toc135517379)

[**User Stories:** 8](#_Toc135517380)

[**Use Cases (click for individual descriptions)** 11](#_Toc135517381)

[Specify Requirements 21](#_Toc135517382)

[3.1 Functional Requirements 21](#_Toc135517383)

[3.1.1 Requirements of the Admin 21](#_Toc135517384)

[**Matrix of Responsibilities** 27](#_Toc135517385)

[**6. PERT Chart** 28](#_Toc135517386)

[**Test Strategy** 29](#_Toc135517387)

[**Wireframes:** 30](#_Toc135517388)

[**8. RISK ASSESSMENT** 31](#_Toc135517389)

[**Risk Identification** 31](#_Toc135517390)

[**Risk Prioritization** 32](#_Toc135517391)

[**Risk Mitigation** 32](#_Toc135517392)

[**9. CONFIGURATION AND VERSION CONTROL** 33](#_Toc135517393)

[**11. ARCHITECTURE** 33](#_Toc135517395)

# **REVISION HISTORY**

|  |  |  |  |
| --- | --- | --- | --- |
| **ate** | **Author** | **Distributed to** | **Version** |
| 19/04/2023 | All | All | 1.0 |
| 02/05/2023 | All | All | 1.1 |
| 18/05/2023 | All | All | 1.2 |

# **PRODUCT DESCRIPTION**

The tourist guide application is designed to assist users in exploring and navigating through a city or region by providing information about the places in Luxor city like temples, cafes, hospitals and places need to be visited. And if you need to know more information about the places so you can check whatever you need. And it is not just that you can choose the places you need to visit and it can help you make your own programme.

If you need to know more about the places desired, you can ask our tourist chat so it can help you to increase your information and if you find any think difficult using the application you can contact us using the barcodes in the application

# **2. TEAM DESCRIPTION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Team Members  Concepts | Mohamed Osama | Ahmed Mohamed | Mahmoud Haggage | Mohamed Gamal Youssef Abdelrahman  Ebrahem Ahmed |
| Front-End |  |  |  | X X |
| Back-End | X |  | X |  |
| Database Management |  |  |  | X X |
| Chatbot (AI) | X | X |  |  |
| Tester | X | X |  |  |
| Documentation |  |  | X | X X |
| Research | X | X | X | X X |
| User Management | X |  |  |  |
|  |  |  |  |  |

The skills needed for this project are:

* Time Management
* Good Communication Skills
* Positive Attitude
* Project Management and Collaboration
* Interpersonal Skills
* Respect each other’s ideas
* Programming Languages Experience
* Flexibility

As a whole, our team is diverse in many concepts needed for this project. Mohamed Osama, Mahmoud and Ahmed have knowledge about the programming language, Mohamed Gamal know about database and how to collect the data needed in the application. Abdelrahman know about data analyzing so he can help us with the application and Youssef know more things about network and this stuff so he can help us with it things.

We do not need a storage expert to find and store information because we are all working to find correct and accurate information to put it in the program, but we need permanent individuals to follow up on this information about the places that need to be put in the program and to make adjustments to it

# **3. SOFTWARE PROCESS MODEL DESCRIPTION**

**we choose the Agile methodology in this project for many reasons like**:

1-Agile emphasizes iterative development and incremental delivery of working software. It allows for flexibility and adaptability throughout the project.

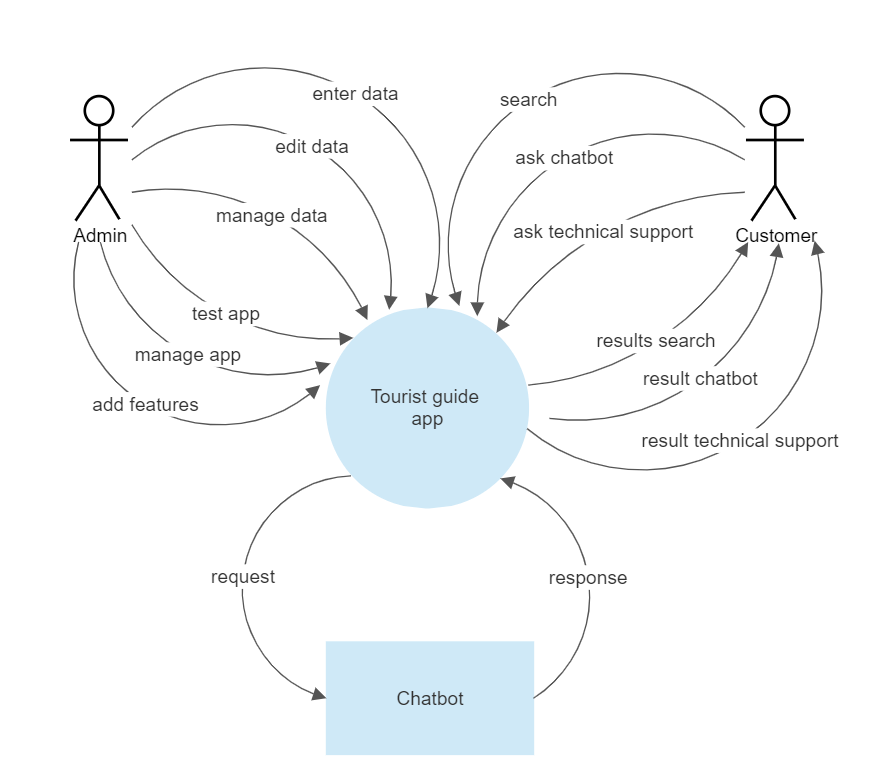
2-Agile methodology is a good fit for projects where requirements are subject to change or the project environment is dynamic.

3-The most important reason for choosing it is agile empowers cross-functional teams to make decisions, collaborate closely, and take ownership of their work.

4-Agile focus on the rapid delivery working software in short iteration

# 

# **4. PRODUCT DEFINITION: Context Diagram**

****

**Persons:**

**Customer**

The customer interacts with the Application using screens. It must be very easy for them to use the Application. They should be supported by the system in every possible way.

**Admin**

It should be easy to maintain the whole system. The Admin should be the only person that is allowed to connect a new application

## 

## **Chatbot**

It is an additional feature on the application that helps the customer to interact in simple way with the system as he can ask about more information on the place desired place or anything he need to know.

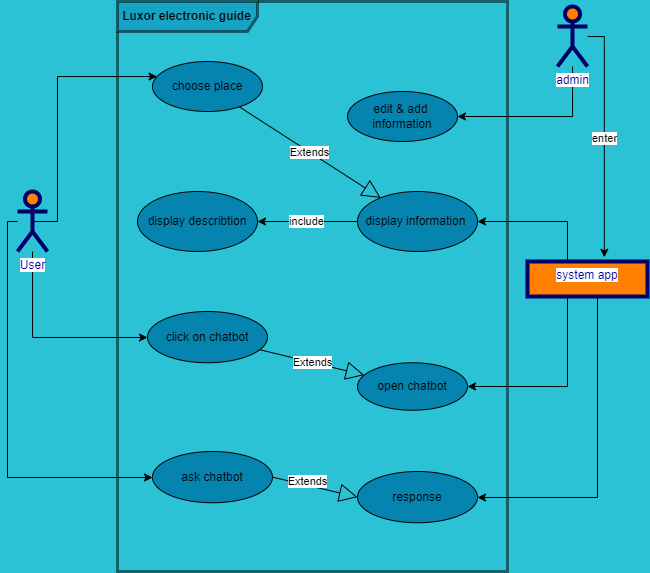
**User Stories:**

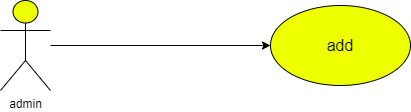
|  |  |  |
| --- | --- | --- |
| Title:  recommendations for places to visit. | Priority:  Medium | Estimate:  3days |
| User story:  As a tourist using the application, I want to be able to get recommendations for places to visit, so that I can discover new and interesting places in the city. The recommendations should be based on my preferences and past behavior. | | |
| Acceptance criteria:  -User can provide their preferences for categories of places to visit (e.g., historical sites, museums, parks, etc.).  -User can view a list of recommended places to visit based on their preferences and past behavior.  -Recommendations should be sorted by relevance and popularity.  -User can click on a recommendation to view more information about the place, including its name, address, description, and ratings. | | |

|  |  |  |
| --- | --- | --- |
| Title:  chat with a virtual assistant to get help | Priority:  High | Estimate:  9days |
| User story:  As a tourist using the application, I want to be able to chat with a virtual assistant to get help, so that I can easily find the information I need and get assistance with any issues I encounter. | | |
| Acceptance criteria:  -User can access the chatbot from the main menu of the application.  -Chatbot should be able to answer common questions about the city, such as information about tourist attractions, transportation, and restaurants.  -Chatbot should be able to understand and respond to natural language queries.  -If chatbot is unable to answer a question, it should escalate the query to a human agent.  -User can rate the chatbot's performance and provide feedback. | | |

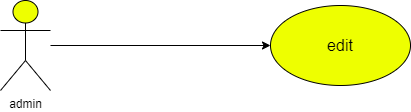
|  |  |  |
| --- | --- | --- |
| Title:  manage the content and settings of the application. | Priority:  High | Estimate:  6days |
| User story:  As an administrator using the application, I want to be able to manage the content and settings of the application, so that I can keep the information up to date and respond to user feedback. | | |
| Acceptance criteria:  -Administrator can access a web-based dashboard to manage the application.  -Administrator can add, edit, and delete points of interest and their associated data (e.g., name, description, images, ratings).  -Administrator can view user feedback and ratings for the application.  -Administrator can configure application settings, such as the language, currency, and categories of points of interest.  -Dashboard should include analytics and reports about the usage and performance of the application. | | |

## [**Use Cases**](https://docs.google.com/document/d/12XgfNrVNvVZUDHk7BYBtv-j7iTASqcnU_EvzfFDVBv4/edit) **(click for individual descriptions)**

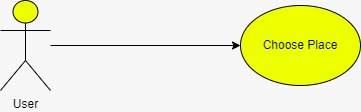




|  |  |  |
| --- | --- | --- |
| **Use case name** | Adding information | |
| **Unique ID** | 001 | |
| **Area** | Application | |
| **Actor(s)** | Admin | |
| **Description** | Admin adds data | |
| **Triggering Event** | Admin enters to database | |
| **Preconditions** | - The admin needs to login in to his account | |
| **Post conditions** | - Admin has successfully added data to application | |
| **Assumptions** | * Admin has application * A valid data | |
| **Steps Performed** | | **Information for Steps** |
| 1- Open application 2- Admin log in  3- admin enters database  4- Admin adds the new information in the system   1. Validation message | | Step 2: E-mail, Password |
| **Extensions (Alternative Flows)** | - If admin entered a non-valid data, a warning message should appear to him | |



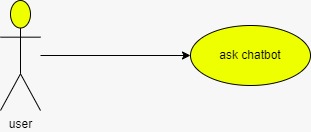
|  |  |  |
| --- | --- | --- |
| **Use case name** | Edit information | |
| **Unique ID** | 001 | |
| **Area** | Application | |
| **Actor(s)** | Admin | |
| **Description** | Admin edits data | |
| **Triggering Event** | Admin enters to database | |
| **Preconditions** | - The admin needs to login in to his account | |
| **Post conditions** | - Admin has successfully added data to application | |
| **Assumptions** | * Admin has application * A valid data | |
| **Steps Performed** | | **Information for Steps** |
| 1- Open application 2- Admin log in  3- admin enters database  4- Admin edits the information in the system   1. Validation message | | Step 2: E-mail, Password |
| **Extensions (Alternative Flows)** | - If admin entered a non-valid data, a warning message should appear to him | |



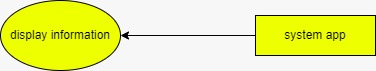
|  |  |  |
| --- | --- | --- |
| **Use case name** | * Choose place | |
| **Area** | * Application | |
| **Actor(s)** | * User | |
| **Description** | * The user select the desired place | |
| **Triggering Event** | * User selects the place on the screen | |
| **Preconditions** | - The user needs to be on the application | |
| **Post conditions** | - User has successfully entered the application | |
| **Assumptions** | * User has to use application | |
| **Steps Performed** | | **Information for Steps** |
| 1- Open application 2- User enters the application  3- User selects the icon | |  |
| **Extensions (Alternative Flows)** | - If user entered a non-valid data, a warning message should appear to him | |



|  |  |  |
| --- | --- | --- |
| **Use case name** | * Click on chat bot | |
| **Area** | * Application | |
| **Actor(s)** | * User | |
| **Description** | * The user select the Chat bot | |
| **Triggering Event** | * User selects the chat bot icon | |
| **Preconditions** | - The user needs to be on the application | |
| **Post conditions** | - User has successfully entered the application | |
| **Assumptions** | * User has to use application | |
| **Steps Performed** | | **Information for Steps** |
| 1- Open application 2- User enters the application  3- User selects the chat bot icon | |  |
| **Extensions (Alternative Flows)** | - If user entered a non-valid data, a warning message should appear to him | |

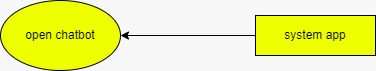


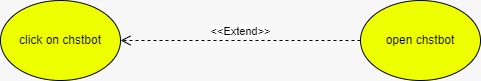
|  |  |  |
| --- | --- | --- |
| **Use case name** | * Ask chat bot | |
| **Area** | * Application | |
| **Actor(s)** | * User | |
| **Description** | * The user ask the Chat bot | |
| **Triggering Event** | * User selects the chat bot icon | |
| **Preconditions** | - The user needs to be on the application | |
| **Post conditions** | - User has successfully entered the application | |
| **Assumptions** | * User has to use application | |
| **Steps Performed** | | **Information for Steps** |
| 1- Open application 2- User enters the application  3- User selects the chat bot icon  4-User starts to ask it about what he needs to know | |  |
| **Extensions (Alternative Flows)** | - If user entered a non-valid data, a warning message should appear to him | |





|  |  |  |
| --- | --- | --- |
| **Use case name** | * Display information | |
| **Area** | * Application | |
| **Actor(s)** | * System | |
| **Description** | * The data user selects shown on the application | |
| **Triggering Event** | * Non | |
| **Preconditions** | - The system need to be used | |
| **Post conditions** | - User has successfully entered the application and the application is on | |
| **Assumptions** | * The system is on | |
| **Steps Performed** | | **Information for Steps** |
| 1- Open application 2- User enters the application  3- User selects the desired information | |  |
| **Extensions (Alternative Flows)** | - If there is non-valid data, a warning message should appear to him | |





|  |  |  |
| --- | --- | --- |
| **Use case name** | * Open chat bot | |
| **Area** | * Application | |
| **Actor(s)** | * System | |
| **Description** | * The system calls chat bot | |
| **Triggering Event** | * Non | |
| **Preconditions** | - The system need to be used | |
| **Post conditions** | - User has successfully entered the application and the application is on | |
| **Assumptions** | * The system is on | |
| **Steps Performed** | | **Information for Steps** |
| 1- Open application 2- User enters the application  3- User selects the chat bot | |  |
| **Extensions (Alternative Flows)** | - If there is non-valid data, a warning message should appear to him | |





|  |  |  |
| --- | --- | --- |
| **Use case name** | * Response | |
| **Area** | * Application | |
| **Actor(s)** | * System | |
| **Description** | * Application response to user | |
| **Triggering Event** | * Non | |
| **Preconditions** | - The system need to be used | |
| **Post conditions** | - User has successfully entered the application and the application is on | |
| **Assumptions** | * The system is on | |
| **Steps Performed** | | **Information for Steps** |
| 1- Open application 2- User enters the application  3- User askes chat bot  4- System response to him | | Step1- Ask chat bot |
| **Extensions (Alternative Flows)** | - If there is non-valid data, a warning message should appear to him | |

Specify Requirements

3.1 Functional Requirements

The functional requirements are organized in three sections: First requirements of the Admin, second requirements of the System and third one of the bot.

3.1.1 Requirements of the Admin

The requirements for the Luxor guide are organized in the following way: General requirements, requirements for authorization, requirements for a transaction.

Functional requirement 1

**Description**

The admin enters the data of the app (Restaurants, Temples, Hostels, …..)

**Input**

the data of the app

**Processing**

Save the data of the app

**Output**

Saves the data

Functional requirement 2

**Description**

The customer clicks icon.

**Input**

Touch icon.

**Processing**

Display icon clicked.

O**utput**

Data icon clicked.

Functional requirement 3

**Description**

The customer specify the data of destination desired.

**Input**

Touch icon clicked.

**Processing**

Process the data of location desired.

**Output**

Data information.

Functional requirement 4

**Description**

The customer destination of location desired.

**Input**

Touch icon clicked.

**Processing**

Process the data of location desired.

**Output**

Data information.

Functional requirement 5

**Description**

The customer clicks chat bot if the place not found.

**Input**

Click chat bot.

P**rocessing**

Enter Chat bot.

**Output**

Display chat bot.

Functional requirement 6

**Description**

Know another places.

**Input**

Click back.

**Processing**

Back home page.

**Output**

Display home page.

Functional Requirement 7

**Description:**

The application should include a chatbot that can answer user queries and provide recommendations for things to do and see in the city.

**Input:**

User queries in natural language format.

**Processing:**

The chatbot should be able to analyze the user query and identify the intent of the request.

It should then search through the available data (e.g. descriptions of tourist attractions, restaurant recommendations, etc.) to find the most relevant information.

The chatbot should then provide a response to the user in natural language format, using appropriate phrasing and tone.

**Output:**

Responses to user queries and recommendations for things to do and see in the city.

Functional Requirement 8

**Description:**

The chatbot should be able to personalize its responses based on the user's preferences and past interactions.

**Input:**

User preferences and past interactions.

**Processing:**

The chatbot should be able to remember previous conversations with the user and use that information to provide more relevant recommendations and advice.

It should also be able to analyze user preferences (e.g. dietary restrictions, preferred types of activities, etc.) to provide personalized recommendations.

The chatbot should be able to adapt its responses based on the user's tone and style of communication.

**Output:**

Personalized responses and recommendations based on the user's preferences and past interactions.

Functional Requirement 9

**Description:**

The chatbot should be able to handle errors and misunderstandings gracefully.

**Input:**

User queries that the chatbot cannot understand or process.

**Processing:**

The chatbot should be able to recognize when it doesn't understand a query and ask the user to rephrase or clarify their request.

It should also be able to recognize when the user is frustrated or confused and offer additional assistance.

The chatbot should be able to provide error messages that are clear and easy to understand.

**Output:**

Clear error messages and assistance for users who have trouble communicating with the chatbot.

**Nonfunctional requirement:**

1. **Performance: The app should respond quickly to user interactions and provide a smooth user experience, even when handling large amounts of data or complex operations.**
2. **Reliability: The app should be reliable and stable, minimizing crashes, errors, and data loss. It should also have appropriate error handling and recovery mechanisms.**
3. **Security: The app should ensure the security of user data and protect against unauthorized access. This includes implementing secure authentication, data encryption, and secure communication protocols.**
4. **Usability: The app should be intuitive and user-friendly, with a well-designed interface and clear navigation. It should accommodate users of varying technical expertise and provide helpful guidance or tooltips where necessary.**
5. **Maintainability: The app should be designed and developed in a modular and maintainable manner, enabling easy updates, bug fixes, and future enhancements. Code documentation and consistent coding practices can aid in maintainability.**
6. **Accessibility: The app should be accessible to users with disabilities, conforming to accessibility standards and guidelines. This may include providing alternative text for images, keyboard accessibility, and support for assistive technologies.**
7. **Backup and Recovery: The app should provide mechanisms for regular data backup and the ability to recover data in case of system failures or data corruption.**

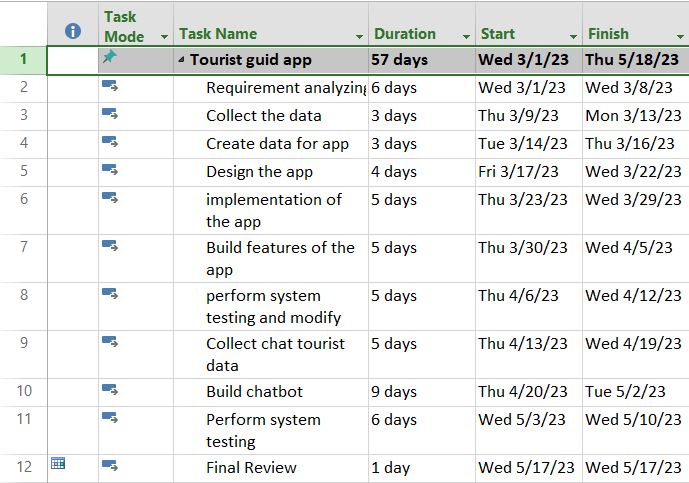
**5. PROJECT ORGANIZATION**

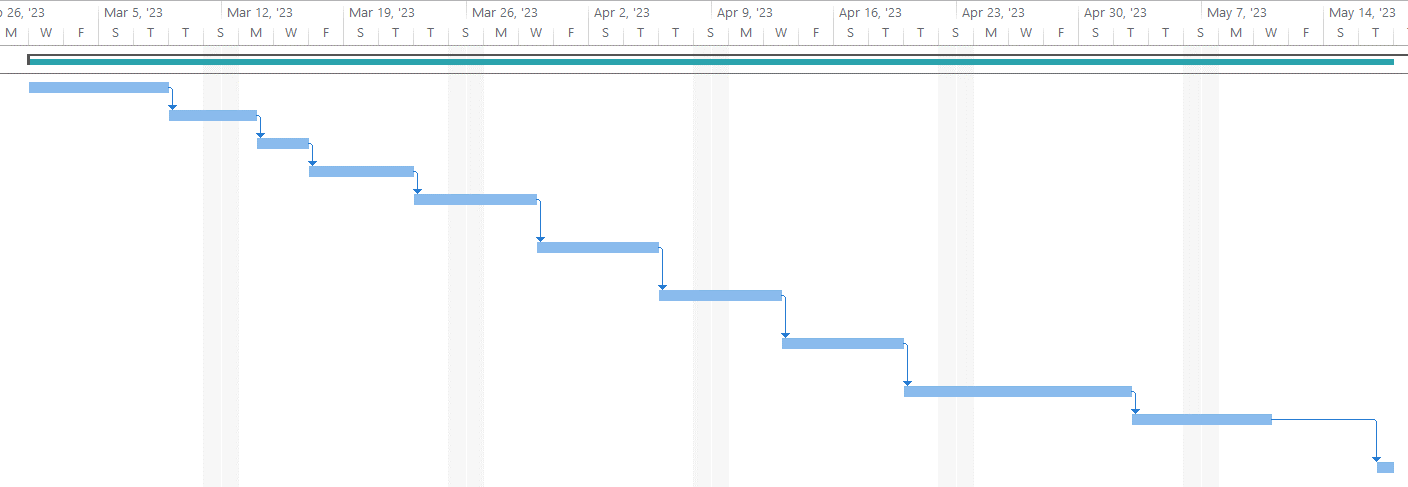
## **Matrix of Responsibilities**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Team Members  Concepts | Mohamed Osama | Ahmed Mohamed | Mahmoud Haggage | Mohamed Gamal Youssef Abdelrahman  Ebrahem Ahmed |
| Front-End |  |  |  | X X |
| Back-End | X |  | X |  |
| Database Management |  |  |  | X X |
| Chatbot (AI) | X | X |  |  |
| Tester | X | X |  |  |
| Documentation |  |  | X | X X |
| Research | X | X | X | X X |
| User Management | X |  |  |  |
|  |  |  |  |  |

\*\*\*Underlined X represent leader on subject

# **6.** [**PERT Chart**](https://drive.google.com/open?id=1rj1XyvQ63gfzoeDCI-cp3nwjir7VcBzN)





**7. VALIDATION PLAN**

## **Test Strategy**

The concept of "done" can be defined as fulfilling a set of requirements to ensure that the application is fully prepared for use by the project's deadline and satisfies the customer's need to find their desired record.

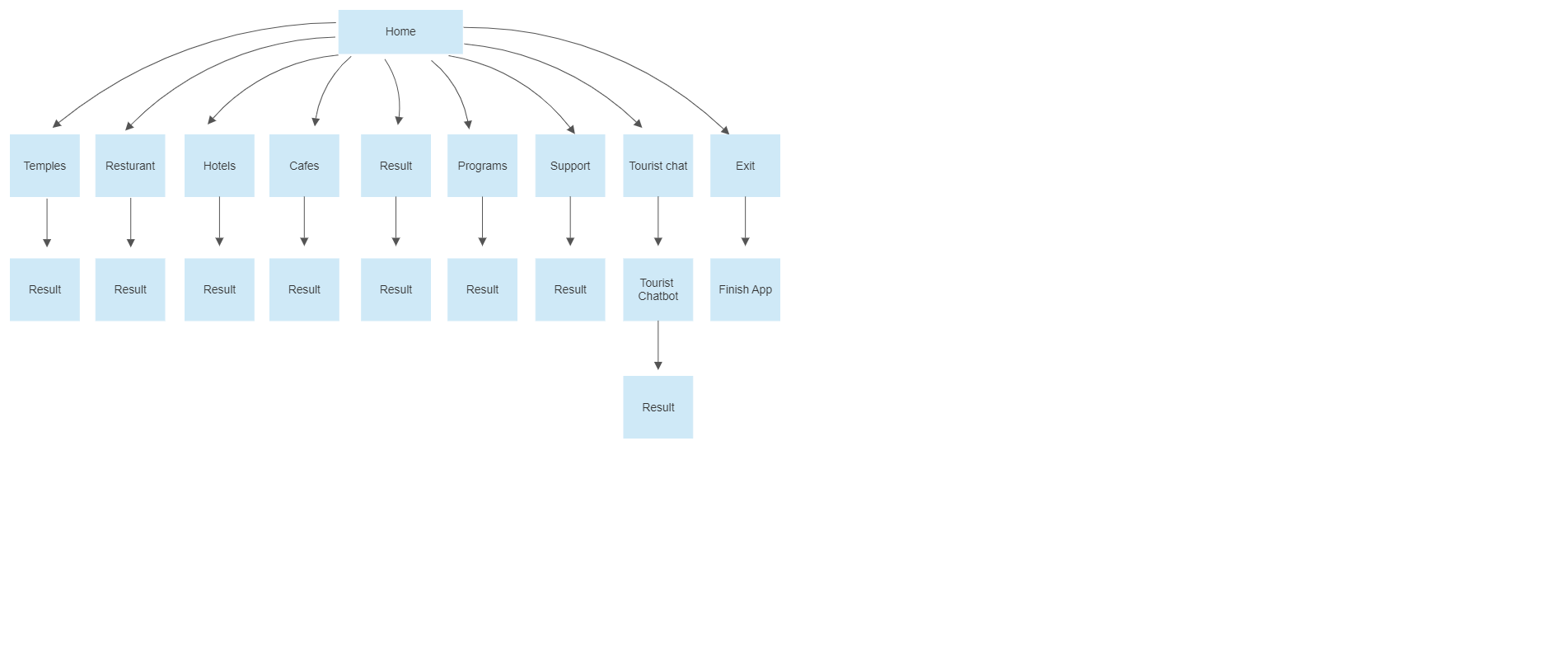
From our perspective, success entails the effective and convenient functioning of the app, allowing individuals to easily locate their desired places, gain in-depth information about them, and achieve widespread adoption of the system. Furthermore, it involves generating a favorable financial outcome that surpasses the associated expenses

Our main test plan will be as follows:

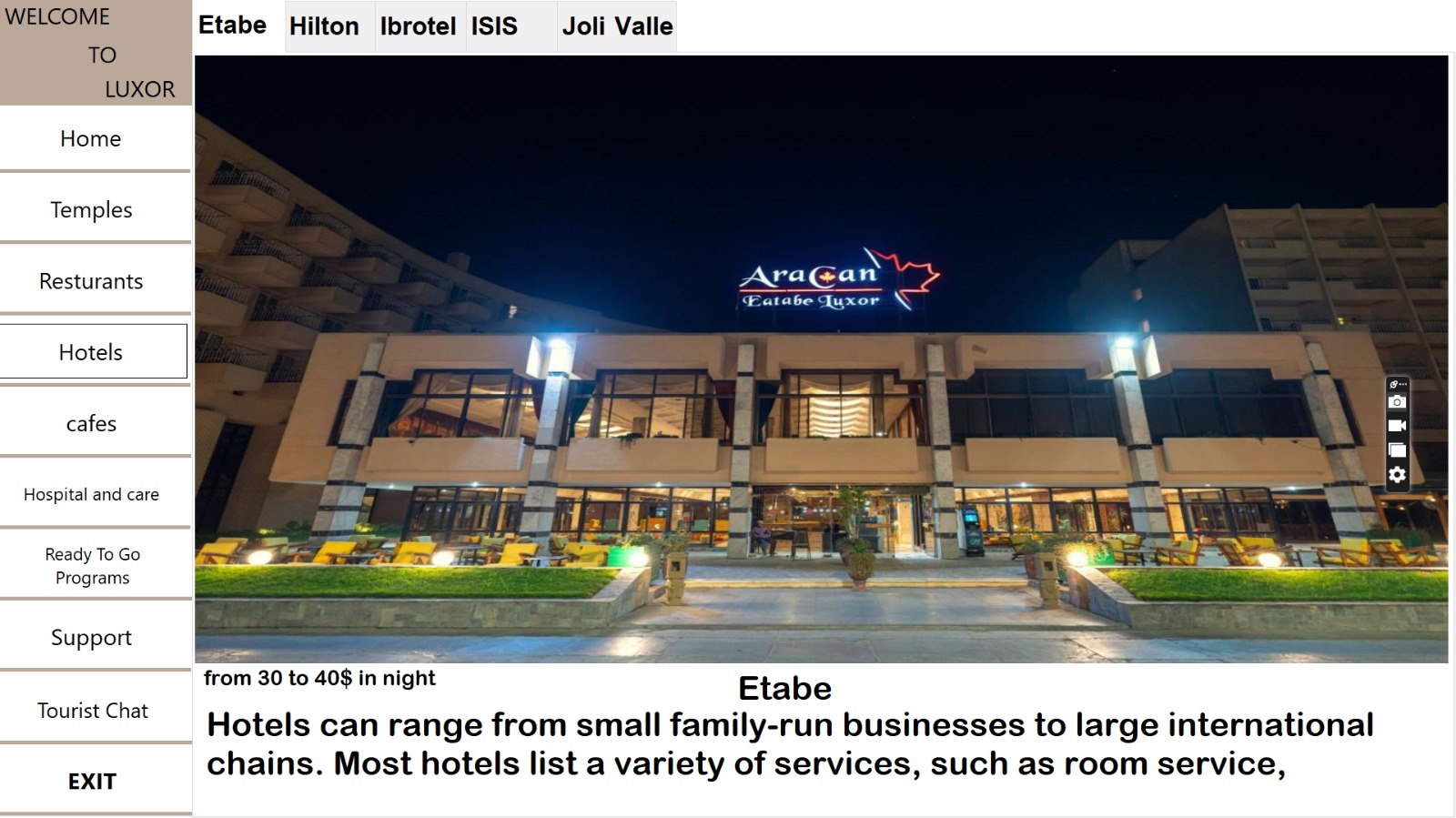
1. we must develop a set of test cases that cover the critical functionalities of the application.
2. Perform tests to validate the functionality of the application.
3. Verify the accuracy and completeness of the data stored in the application.
4. specifying the performance of the application under different conditions
5. Ensure that data such as names, addresses, contact details, and opening hours are correct.
6. Evaluate the user experience of the application by involving real testers.
7. Testing features like saving entities for later reference.

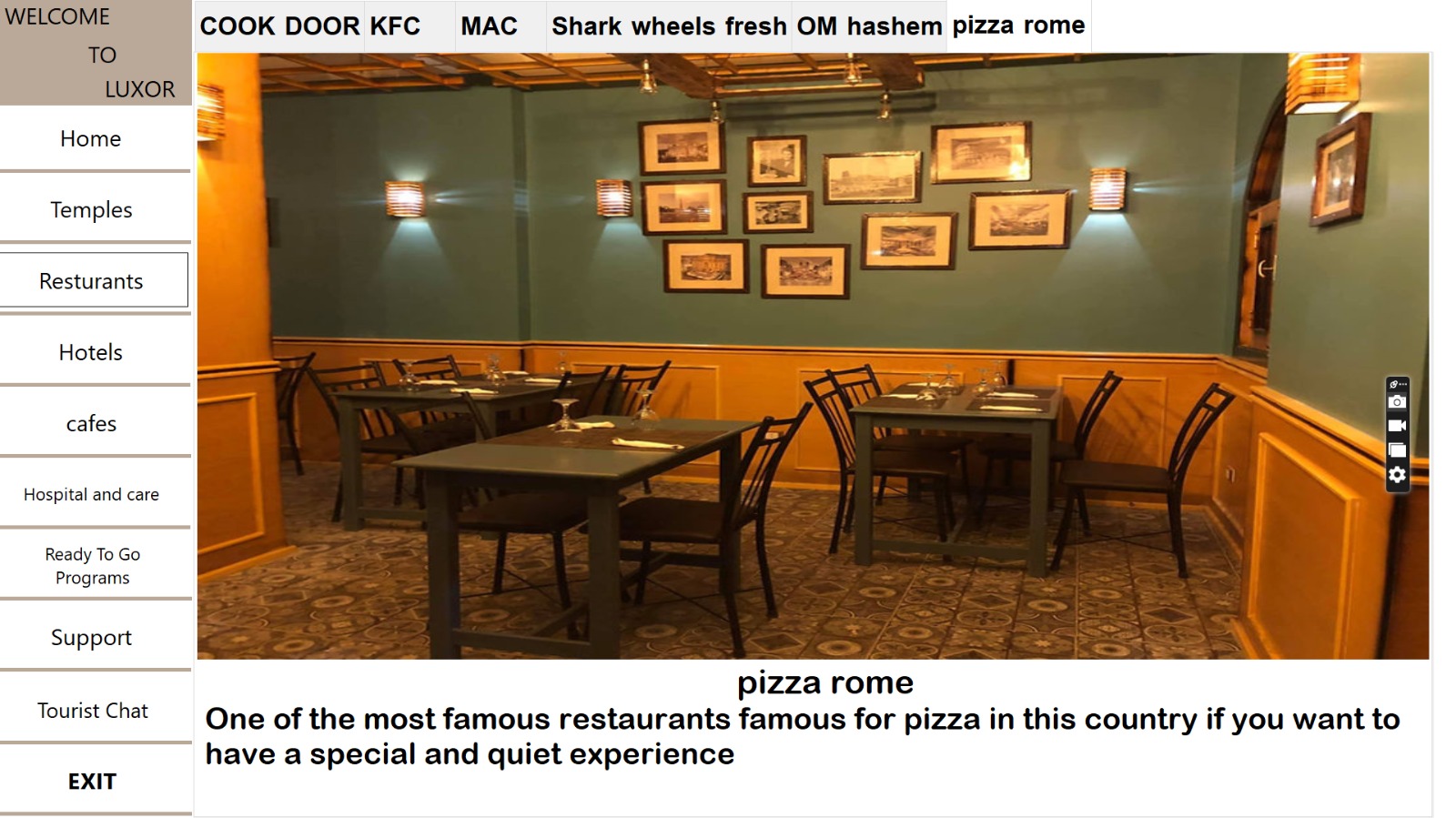
* Success will be if the application runs successfully without errors, the user is satisfied and covers its costs.

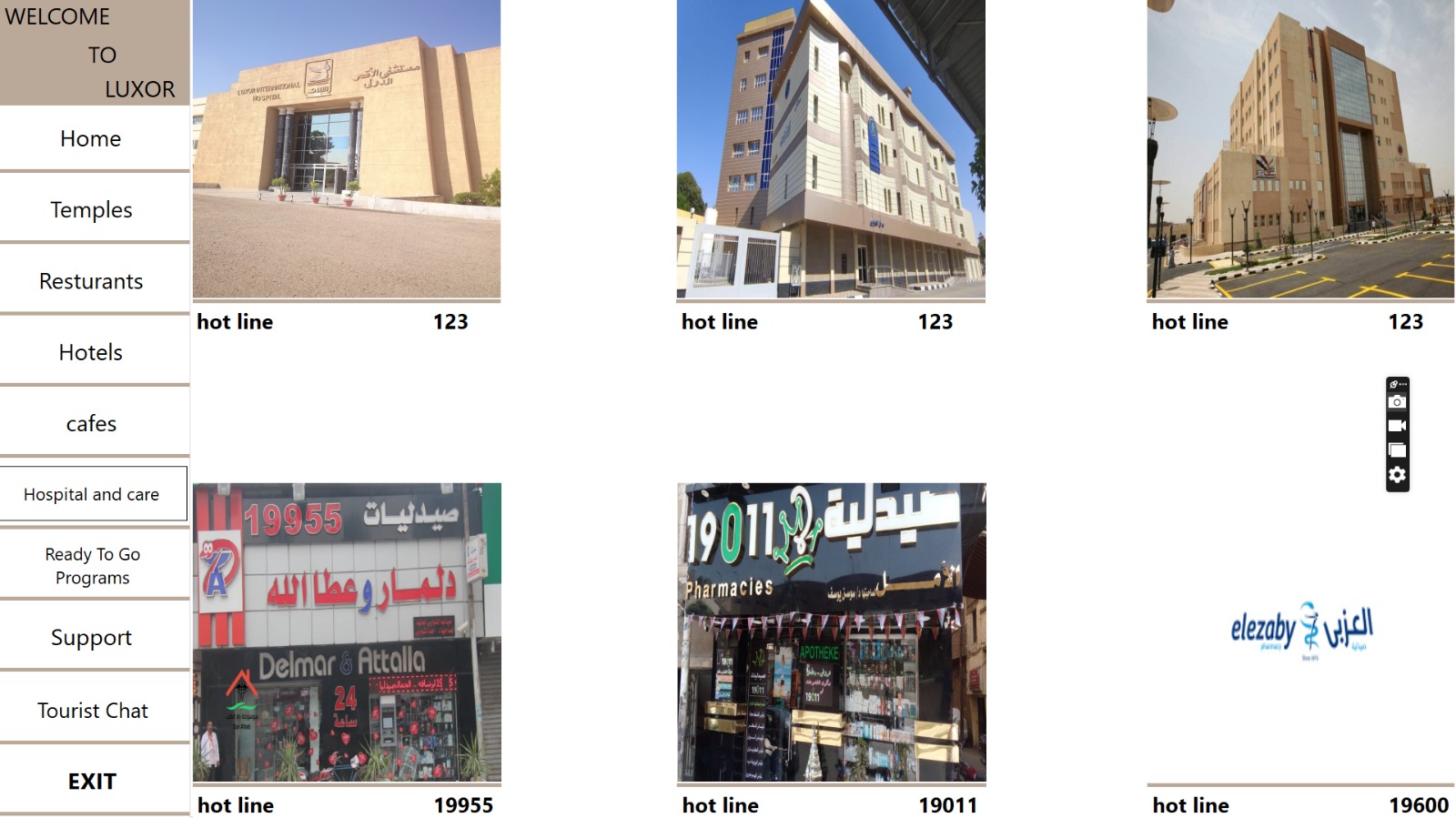
## **Wireframes and screen of the app:**

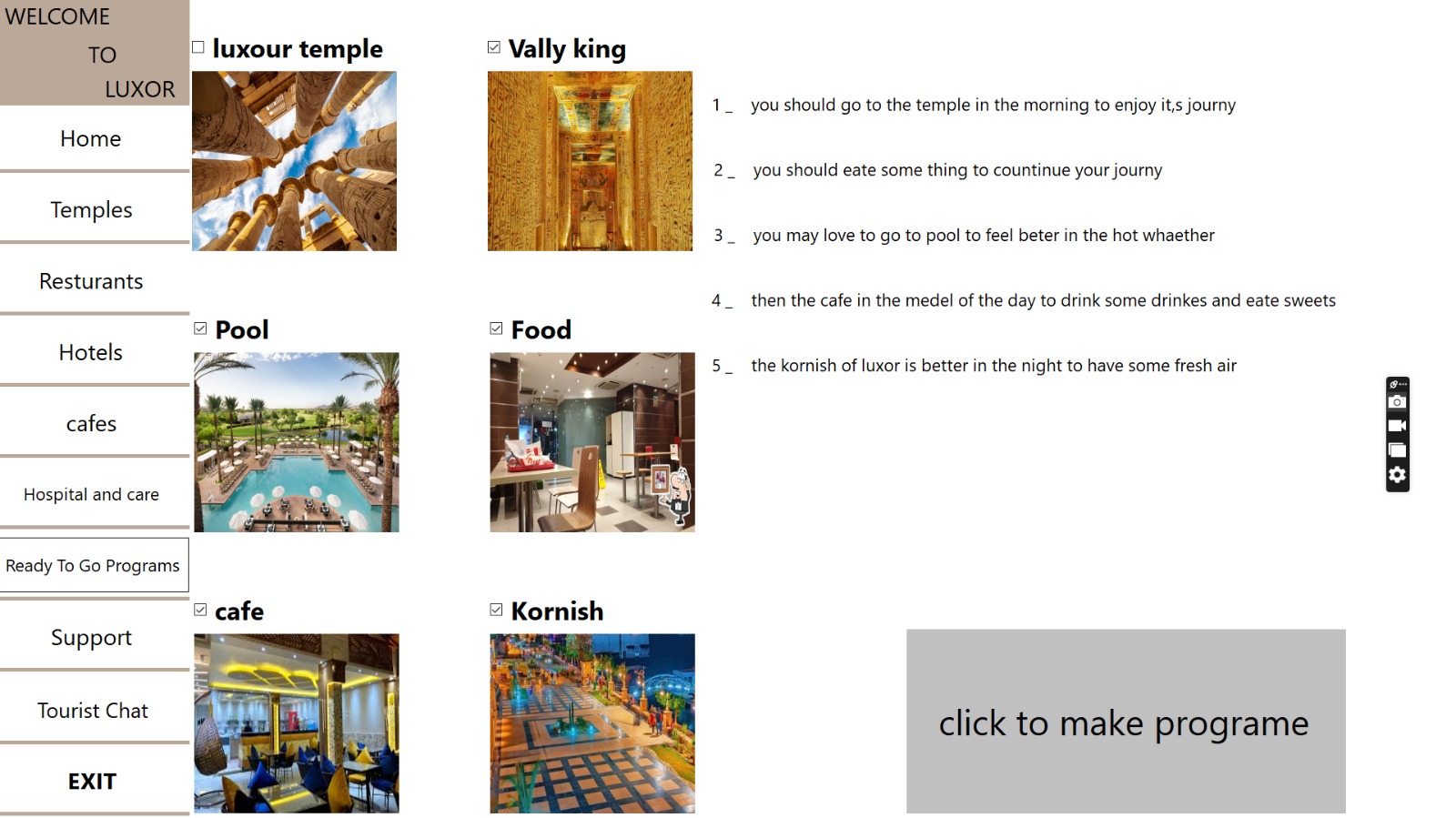
****

# 

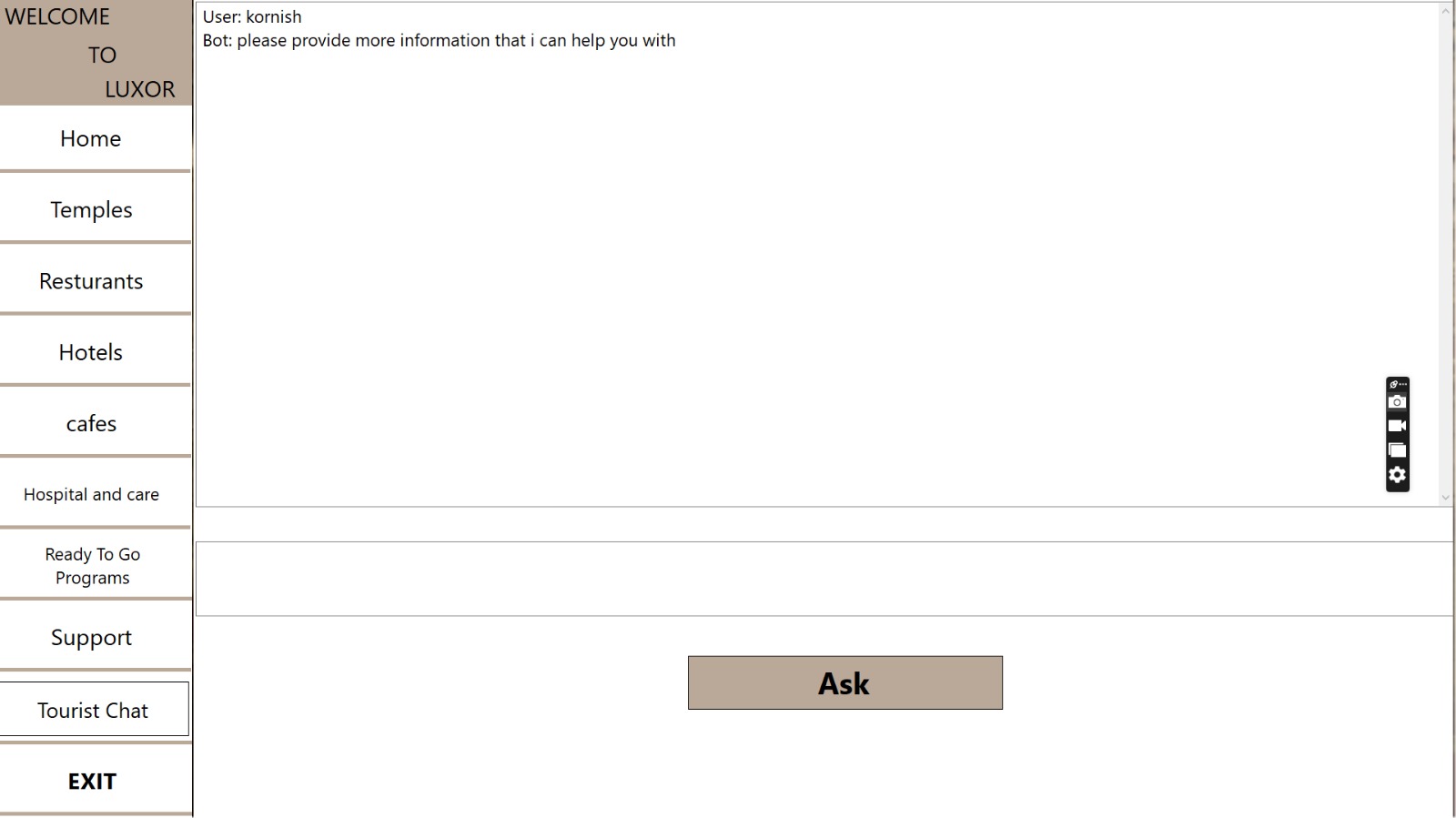












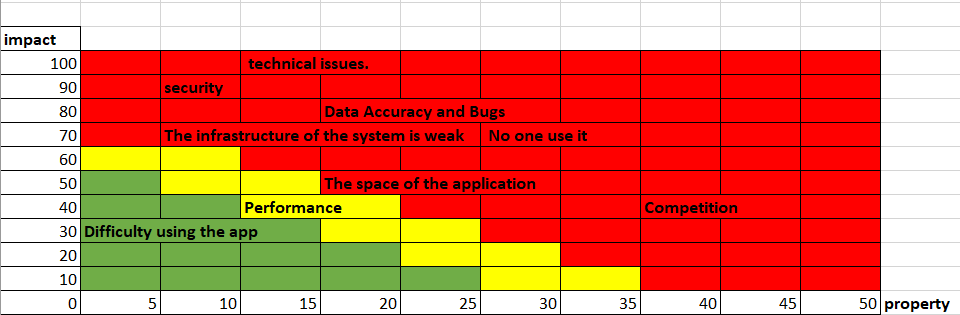
# **8. RISK ASSESSMENT**

## **Risk Identification**

* For us, we are the team entrusted with the work of this project. The most dangerous risk facing us is that the project does not cover the costs of hardware or infrastructure.
* Also, there is another part that faces us, which is that we need to determine the location of the devices in which the program will be placed, because if these places are not vital enough, then we have not achieved the maximum benefit from this project.
* If the program has a relatively large area, it negatively affects the work of the project, as it is difficult for users to use it and requires high device capabilities to run it, and therefore the cost will increase

## **Risk Prioritization**

1. The application is not noticed and no one would use it.
2. Incorrect data
3. Not covering project costs.
4. Technical issues and bugs.
5. The space of application.



## **Risk Mitigation**

* We need to implement a comprehensive marketing plan that includes online and offline channels such as social media, content marketing, targeted advertisements, and influencer collaborations.
* We need to implement a data verification process where collected data is reviewed and validated before being added to the application. Regularly update the data to ensure accuracy and completeness.
* Implement proper error handling and logging mechanisms to capture and track errors that occur in the production environment.
* We should monitor the application's performance in production, and proactively scale the infrastructure as needed to this.

# **9. CONFIGURATION AND VERSION CONTROL**

# In this project we have made three versions of the application V1.0, V1.1, V1.2, in the first version v1.0 the user cannot interact with the system and the system had larger space, in the second version V1.1 the user can interact with system and we can make it easier and in the last version v1.2 we could make it simpler and faster and we can add chat tourist so it may help the user interact with simple way.

# **TOOLS**

* + Visual studio
  + C#

# **11. ARCHITECTURE**

* Devices for testing
  + Desktop app
* Devices for execution
  + Touch screen
  + RAM (more than 2 GB)
  + HDD (more than 64 GB)
  + Compact Processor