

**International School**

**Capstone Project 1**

CMU-SE 450

**Architecture Design**

**Version 1.2**

**Date: 13/12/2020**

**viBOTour - Smart Chatbot for Tourist 4.0**

**Submitted by**

**Sang, Nguyen Thanh**

**Thuyen, Pham Van**

**Tuan, Doan Kim**

**Loc, Phung Hoang Phu**

**Approved by Nguyen Duc Man**

# Project Information

|  |  |  |  |
| --- | --- | --- | --- |
| **Project acronym** | viBOTour | | |
| **Project Title** | Smart Chatbot for Tourist 4.0 | | |
| **Start Date** | 24 Aug 2020 | **End Date** | 5 Dec 2020 |
| **Lead Institution** | International School, Duy Tan University | | |
| **Project Mentor** | Man, Nguyen Duc, Ph.D | | |
| **Scrum master / Project Leader & contact details** | Sang, Nguyen Thanh  Email: [thanhsang3111999.ntb@gmail.com](mailto:thanhsang3111999.ntb@gmail.com) Tel: 0344684069  ID: 23211212363 | | |
| **Partner Organization** |  | | |
| **Project Web URL** |  | | |
| **Team members** | Name | Email | Tel |
| 2321121321 | Tuan, Doan Kim | [doantuan21101999@gmail.com](mailto:doantuan21101999@gmail.com) | 0913622928 |
| 2321124963 | Loc, Phung Hoang Phu | [phunghoangphuloc@gmail.com](mailto:phunghoangphuloc@gmail.com) | 0912698173 |
| 2321125061 | Thuyen, Pham Van | [phamvanthuyen2802@gmail.com](mailto:phamvanthuyen2802@gmail.com) | 0795697332 |

# 

# Architecture Document

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Title** | Architecture Document |  |  |
| **Reporting Period** |  |  |  |
| **Author(s)** | Sang, Nguyen Thanh | |  |
| **Team Information** | **Name** | **Role** |  |
| Sang, Nguyen Thanh | Member |  |
| Tuan, Doan Kim | Member |  |
| Loc, Phung Hoang Phu | Member |  |
| Thuyen, Pham Van | Member |  |
| **Date** | 13/12/2020 | **Filename** | Architecture\_Design\_V1.1.docx |
| **Access** | Project and CMU Program | |  |

|  |  |  |
| --- | --- | --- |
| Document History | | |
| **Version** | **Date** | **Comments** |
| V1.0 | 04/09/2020 | Create Architecture Document |
| V1.1 | 25/09/2020 | Update 3 |
| V1.2 | 13/12/2020 | Update workflow for chatbot |

# 

# Document Approvals

The following signatures are required for approval of this document.

|  |  |  |
| --- | --- | --- |
| Msc Man, Nguyen Duc  Mentor |  | Date |
| Sang, Nguyen Thanh  Scrum master, DevTeam |  | Date |
| Tuan, Doan Kim  Product Owner, DevTeam |  | Date |
| Thuyen, Pham Van  DevTeam |  | Date |
| Loc, Phung Hoang Phu  DevTeam |  | Date |

**TABLE OF CONTENTS**

[Project Information 2](#_Toc58862498)

[Architecture Document 3](#_Toc58862499)

[Document Approvals 4](#_Toc58862500)

[1. Introduction 6](#_Toc58862501)

[1.1 Project overview 6](#_Toc58862502)

[1.2 Purpose 6](#_Toc58862503)

[2. Architecture driver 7](#_Toc58862504)

[2.1 Business constraints 7](#_Toc58862505)

[2.2 Technical constraints 7](#_Toc58862506)

[2.3 Functional requirement 7](#_Toc58862507)

[2.4 Quality attributes 7](#_Toc58862508)

[2.4.2.1 Security 8](#_Toc58862509)

[2.4.2.2 Performance 9](#_Toc58862510)

[2.4.2.3 Usability 10](#_Toc58862511)

[2.4.2.4 Correctness 12](#_Toc58862512)

[2.4.2.4 Modifiability 13](#_Toc58862513)

[3. Architecture overview 14](#_Toc58862514)

[3.1 System context 14](#_Toc58862515)

[3.2 Component and connector 16](#_Toc58862516)

[3.3 Sequence diagram 18](#_Toc58862517)

[3.4 Module view 24](#_Toc58862518)

[3.5 Allocation view 25](#_Toc58862519)

[4. ATAM 29](#_Toc58862520)

[4.1 Present the ATAM 29](#_Toc58862521)

[4.2 Present the business Drivers 29](#_Toc58862522)

[4.3 Present the Architecture 29](#_Toc58862523)

[4.4 Identify the Architecture approaches 29](#_Toc58862524)

[4.5 Create a Quality Attribute Tree 30](#_Toc58862525)

[4.6 Analyze the Architectural approaches 31](#_Toc58862526)

[4.7 Brainstorm and prioritize scenarios 31](#_Toc58862527)

[4.8 Re-analyze the architectural approaches 31](#_Toc58862528)

[4.9 Present the results 31](#_Toc58862529)

[5. References: 32](#_Toc58862530)

# Introduction

## Project overview

The viBOTour system is a travel system that integrates smart chat bot. This system helps tourists find suitable tours on the system. In addition, this system allows users to interact with the chat bot to receive useful information such as tour suggestions,..

## Purpose

This specification covers following:

* Brief specification of the project, high level requirement.
* Detail quality attribution.
* System context, sequence diagrams.
* Architecture presented by various view types: Component and Connect, Module view and Allocation view.
  1. **Business driver**

Business Problems:

* Tourists need assistance with details of a trip, the weather or the place.
* Users need an automated system to assist with billing, travel recommendations, weather, local food or hotel or room rental services.

Business Need:

* A chatbot can communication with tourists,
* Support for their tourists can book a tour, create a custom tour and get suggestions when they need it.
* Tourists can search, view tour details, book and pay for tours.
* Administrators can manage tour, user, promotion and tourist attractions.

# Architecture driver

## 2.1 Business constraints

* Sources: 4 people.
* Project was started on: 24/08/2020.
* Project will be ended on: 05/12/2020.
* Project will be finished in 104 days (980 hours).
* Cost: $2352.

## 2.2 Technical constraints

Technical to develop:

* Programming Language: JavaScript, Solidity.
* Frameworks / Libraries: ReactNative, ReactJS, NodeJS, Bootstrap 4.
* Database Management System: Postgresql.

Environment:

* Web browsers: Google Chrome, FireFox, Opera.
* App environment: IOS and Android
* Operation systems: Microsoft Windows 10, Ubuntu.

## 2.3 Functional requirement

References to Product Backlog specification of ProductBacklogV1.0.docx

## 2.4 Quality attributes

**2.4.1 Utility table**

There are following quality attributes that drive the design of architecture. Each quality attribute scenario is ranked with importance (I) defined by the Product Owner, and the estimated level difficulty (D). Both values are based on a scale of High (H) - Medium (M) - Low (L).

**2.4.2 Quality attributes**

#### 2.4.2.1 Security

|  |  |
| --- | --- |
| Scenario: When customers want to pay for the tour they choose, they are required to provide a confirmation code sent to them on the phone they provided, if within 1 minute they do not enter the correct code will ask for a new confirmation | |
| Type | Security |
| Stimulus | provide a confirmation code |
| Source of stimulus | Tourist |
| Environment | In runtime |
| Artifact stimulated | Application |
| Response | Confirmation code sent to phone |
| Response measure | If within 1 minute they do not enter the correct code will ask for a new confirmation |

|  |  |
| --- | --- |
| Scenario: When a tourist registers, their password will be encrypted through 3 layers of security before being saved to database | |
| Type | Security |
| Stimulus | Register |
| Source of stimulus | Tourist |
| Environment | In runtime |
| Artifact stimulated | System |
| Response | Encrypt password |
| Response measure | Encrypted through 3 layers of security |

#### 2.4.2.2 Performance

|  |  |
| --- | --- |
| Scenario: Tourists require a suggested tour in the chatbot while using the app. The system summarizes tour and response expected tours for them in less than 5 seconds. | |
| Type | Performance |
| Stimulus | Require a suggested tour |
| Source of stimulus | Tourists |
| Environment | In runtime |
| Artifact stimulated | Application |
| Response | Summarizes tour and response expected tour |
| Response measure | less than 5 seconds |

|  |  |
| --- | --- |
| Scenario: Tourists send a request payment for their tour while paying a tour. The system verifies this tour, makes a transaction and sends a result in less than 10 seconds. | |
| Type | Performance |
| Stimulus | Send a request payment |
| Source of stimulus | Tourists |
| Environment | In runtime |
| Artifact stimulated | Application |
| Response | Verify paid tour information, make transaction and response a result |
| Response measure | less than 10 seconds |

|  |  |
| --- | --- |
| Scenario: When a tourist chats to a chatbot, it will respond to search results by text or voice message from its existing data to give the closest answer to a traveler request within 10 seconds with stable network connection. | |
| Type | Performance |
| Stimulus | Respond the search result from tourist request |
| Source of stimulus | Tourists |
| Environment | In runtime and stable network connection |
| Artifact stimulated | Application |
| Response | A text or voice message |
| Response measure | within 5 seconds |

#### 2.4.2.3 Usability

|  |  |
| --- | --- |
| Scenario: Tourists access the system by signing up for a new account or logging in with their Google account with the same email. The application allows access with this email account in both ways. | |
| Type | Usability |
| Stimulus | Access the system by signing up for a new account or logging in with their Google account with the same email |
| Source of stimulus | Tourists |
| Environment | In runtime |
| Artifact stimulated | Application |
| Response | Allows access with this email account |
| Response measure | both ways |

|  |  |
| --- | --- |
| Scenario: Tourists book a tour in the application. When they have other devices they can also see the booked tour and pay for it instantly. | |
| Type | Usability |
| Stimulus | Book a tour and use other devices |
| Source of stimulus | Tourists |
| Environment | The system |
| Artifact stimulated | Mobile devices |
| Response | See the booked tour and pay it |
| Response measure | Instantly |

|  |  |
| --- | --- |
| Scenario: The staff will check the tourist ticket quickly via the QR code sent when payment is successful or the ticket information at the tourist’s email | |
| Type | Usability |
| Stimulus | Check ticket |
| Source of stimulus | The staff |
| Environment | In runtime |
| Artifact stimulated | The system |
| Response | Ticket information |
| Response measure | QR code or the ticket information at the tourist’s email |

#### 2.4.2.4 Correctness

|  |  |
| --- | --- |
| Scenario: Tourists tick the tour they want to pay in their mobile phone and checkout them. The system calculates the total amount and pays the correct quantity and selected tour. | |
| Type | Correctness |
| Stimulus | Tick the tour and checkout them |
| Source of stimulus | Tourists |
| Environment | The system |
| Artifact stimulated | Mobile phone |
| Response | Calculates the total amount and pays tour |
| Response measure | Correct quantity and selected tour |

|  |  |
| --- | --- |
| Quality attributes: When the tourist makes a successful payment, the system will ask the baking system to deduct the money in the account provided by the tourist equal to the amount paid by the tourist | |
| Type | Correctness |
| Stimulus | Pay successfully |
| Source of stimulus | Tourists |
| Environment | Runtime |
| Artifact stimulated | The System |
| Response | Deduct the money in the account provided |
| Response measure | Deduct the money in the account provided by the tourist equal to the amount paid by the tourist |

|  |  |
| --- | --- |
| Scenario: When the staff performs operations with a tour including adding, modifying, and deleting the tour, the corresponding tour information will be updated correctly on the tourist application. | |
| Type | Correctness |
| Stimulus | Performs operations with a tour |
| Source of stimulus | Staff |
| Environment | Runtime |
| Artifact stimulated | The System |
| Response | Change the information tour |
| Response measure | The corresponding tour information will be updated correctly on the tourist application. |

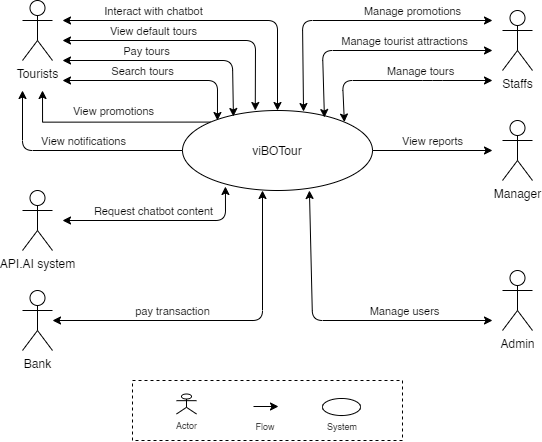
#### 2.4.2.4 Modifiability

|  |  |
| --- | --- |
| Scenario: A product manager wants to develop additional features and construct a travel social network in the next version. The system allows for an expansion within 4 person months of effort without affecting existing functions. | |
| Type | Modifiability |
| Stimulus | Develop additional features and construct a travel social network |
| Source of stimulus | A product manager |
| Environment | The next version |
| Artifact stimulated | The system |
| Response | Allows for an expansion |
| Response measure | 4 person months of effort without affecting existing functions |

# Architecture overview

This section shows the diagrams which bounds our target system and describes the architecture and interaction between components

## 3.1 System context



**Figure 1: System Context Overview**

**Tourist:**

* Request Login/Logout to the system, System check information and response
* Request to view tour and system response list tours
* Request for payment in the vibotour system, will the system respond successfully or not?
* Request to search the tour list, the system will respond to the list of required tours
* Request to display the tours added to the cart, the system will respond to the list of tours that the user has added.

**Admin:**

* Request Login/Logout to the system, System check information and response
* Enter sign up new account, System saves account.
* Checkout the tour, System saves that payment.
* Request to list view tour, System response view tour.
* Request to view tour detail, system response tour detail

**Bank:**

* Transfer money to the another account, the system
* Response table result.
* Receive notification from system.

**Manager :**

* Request Login/Logout to the system, System check information and response
* Enter sign up new account, System saves account.
* Checkout the tour, System saves that payment.
* Request to list view tour, System response view tour.
* Request to view tour detail, system response tour detail

**Staff :**

* Request Login/Logout to the system, System check information and response
* Enter sign up new account, System saves account.
* Checkout the tour, System saves that payment.
* Request to list view tour, System response view tour.
* Request to view tour detail, system response tour detail

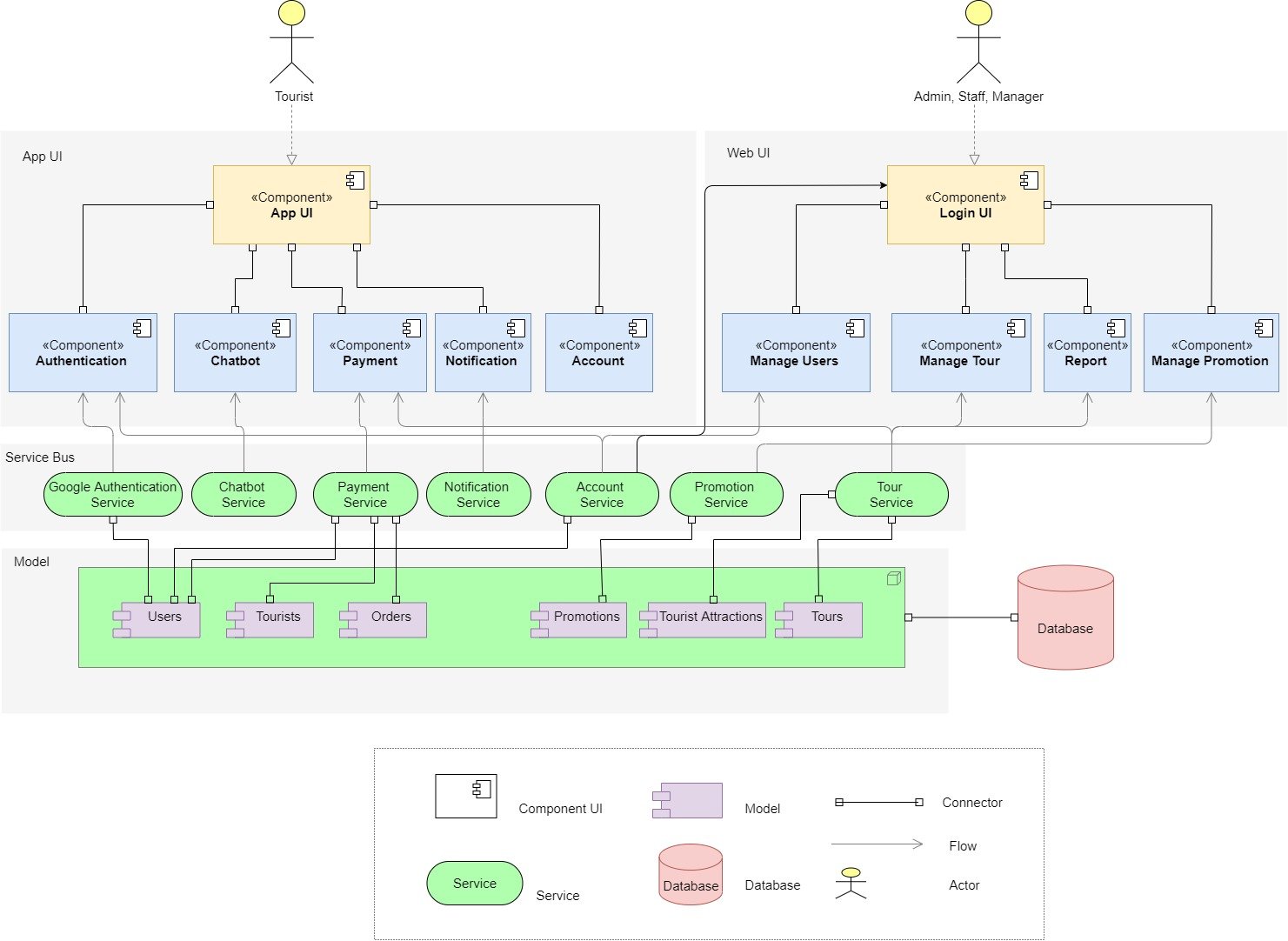
**API.AI System:**

* Request Login/Logout to the system, System check information and response
* Enter sign up new account, System saves account.
* Checkout the tour, System saves that payment.
* Request to list view tour, System response view tour.
* Request to view tour detail, system response tour detail

## 3.2 Component and connector

We mainly used a C&C view to argue and reason about architectural properties, quality attribute requirements, and functional requirements that the system must add here.

This view type partitions the system into components that have some runtime presence such as processes, objects, data stores, and connectors or that represent pathways of communication such as data flows and access to shared storage.



**Prose**

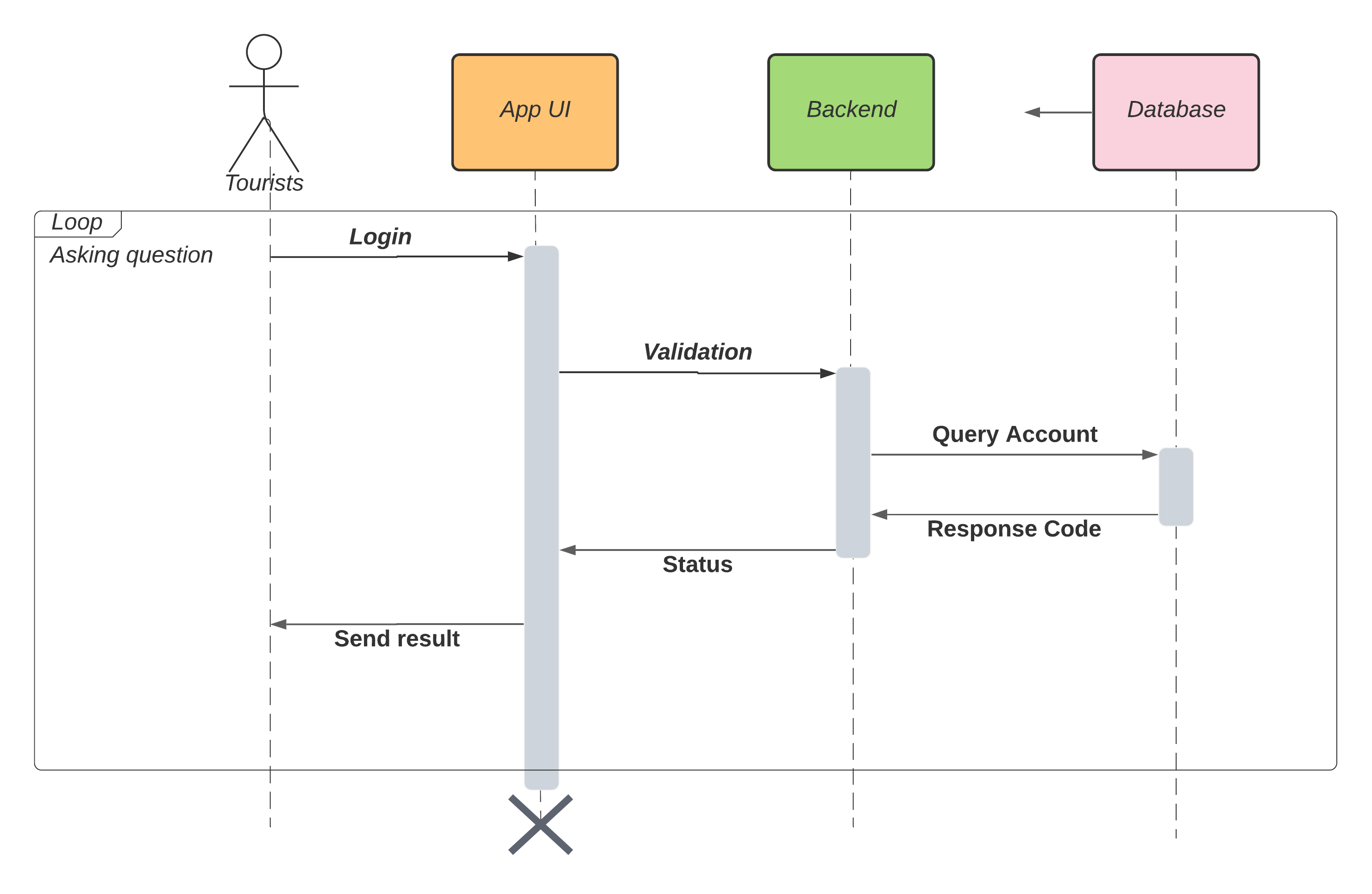
The web client sends and receives data from the server through the web service API. User operations will send requests to the server through the API, the server will process the data with the corresponding models and access the database to retrieve data, then respond to the information to the web client through the API to display it to the user

|  |  |
| --- | --- |
| **Element** | **Responsibilities** |
| Application | Web Client is a component that manages and implements interactive functions for users that are handled on the UI. Web Clients interact directly with users, receive requests and call APIs from Web Service to handle and return results. |
| Web service | Web Service is a component that manages and performs activities related to retrieval and storage of data such as get the students list, user authorization.. |
| Google authentication service | The service provides APIs for login with Google Account and phone authentication. |
| Chatbot service | The Dialog Flow service API is used to get chat content and train the chatbot. |
| Stripe service | Stripe is a platform that supports payment of international card transactions. |
| Account service | Service API in the backend to register a new account and login to the system. |
| Book tour service | Backend API service to choose and add a tour to the shopping card. |
| Management service | Service interacts with the database and provides APIs for managing the system. |
| Firebase service | The service provides image storage solutions and automatic SMS sending services |
| Database | Database is a component which contains information of users, tours, tourist attractions, payment histories. All data the system needs |
| Data Source | Data source is a source that provides data to chat bot systems. |

## 3.3 Sequence diagram

Sequence diagram is used to display the sequence of activities. Sequence diagrams show the workflow from a start point to the finish point detailing the many decision paths that exist in the progression of events contained in the activity.

**3.3.1 Login**

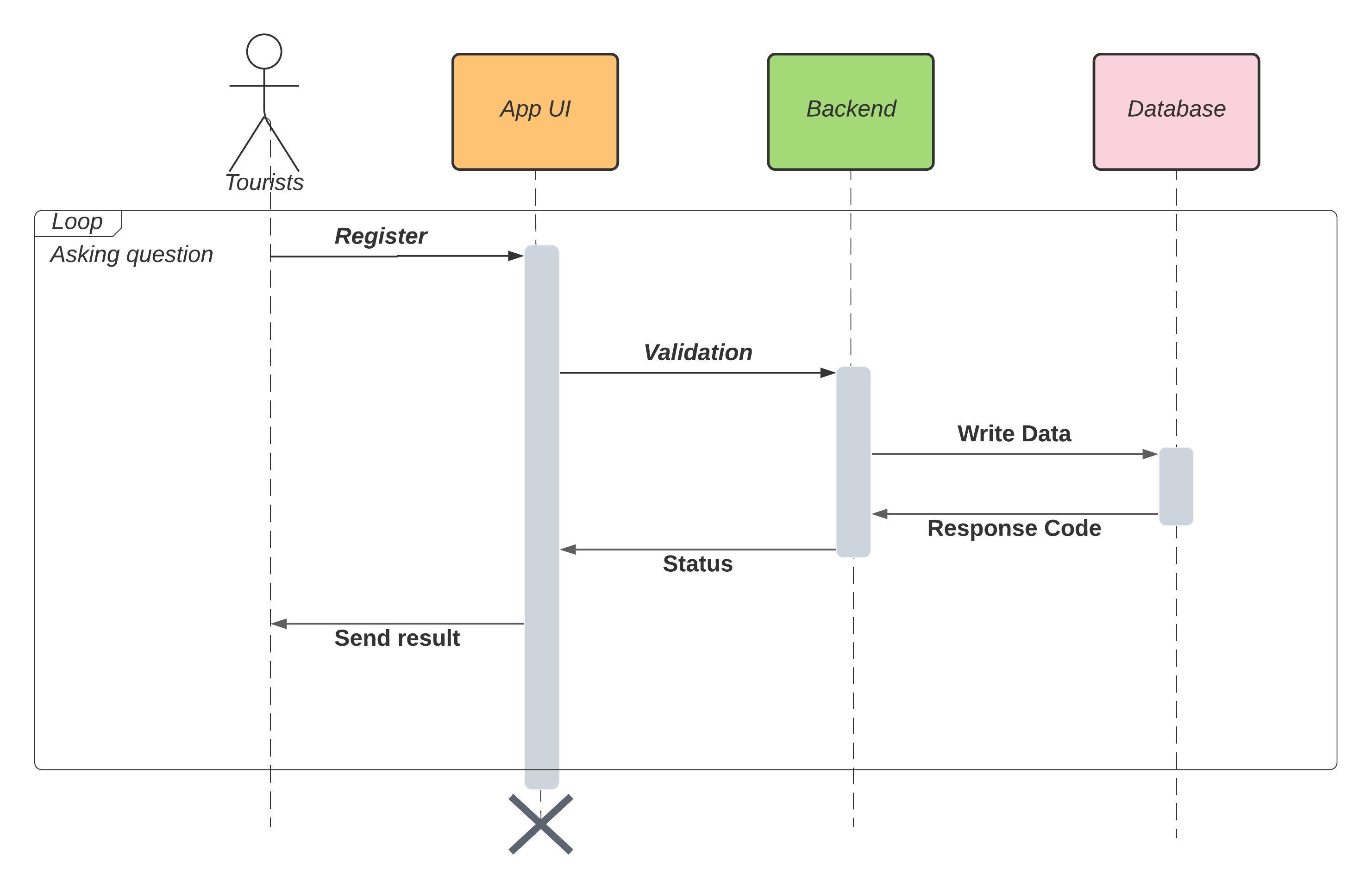


**Description:**

In order to use the viBOTour system, users have to login at first by using username and password. Our system receives requests, authenticates and returns login status.

If login successfully, users can use all functions of the system in their permission. Otherwise, their access will be denied.

**3.3.2 Sign up**

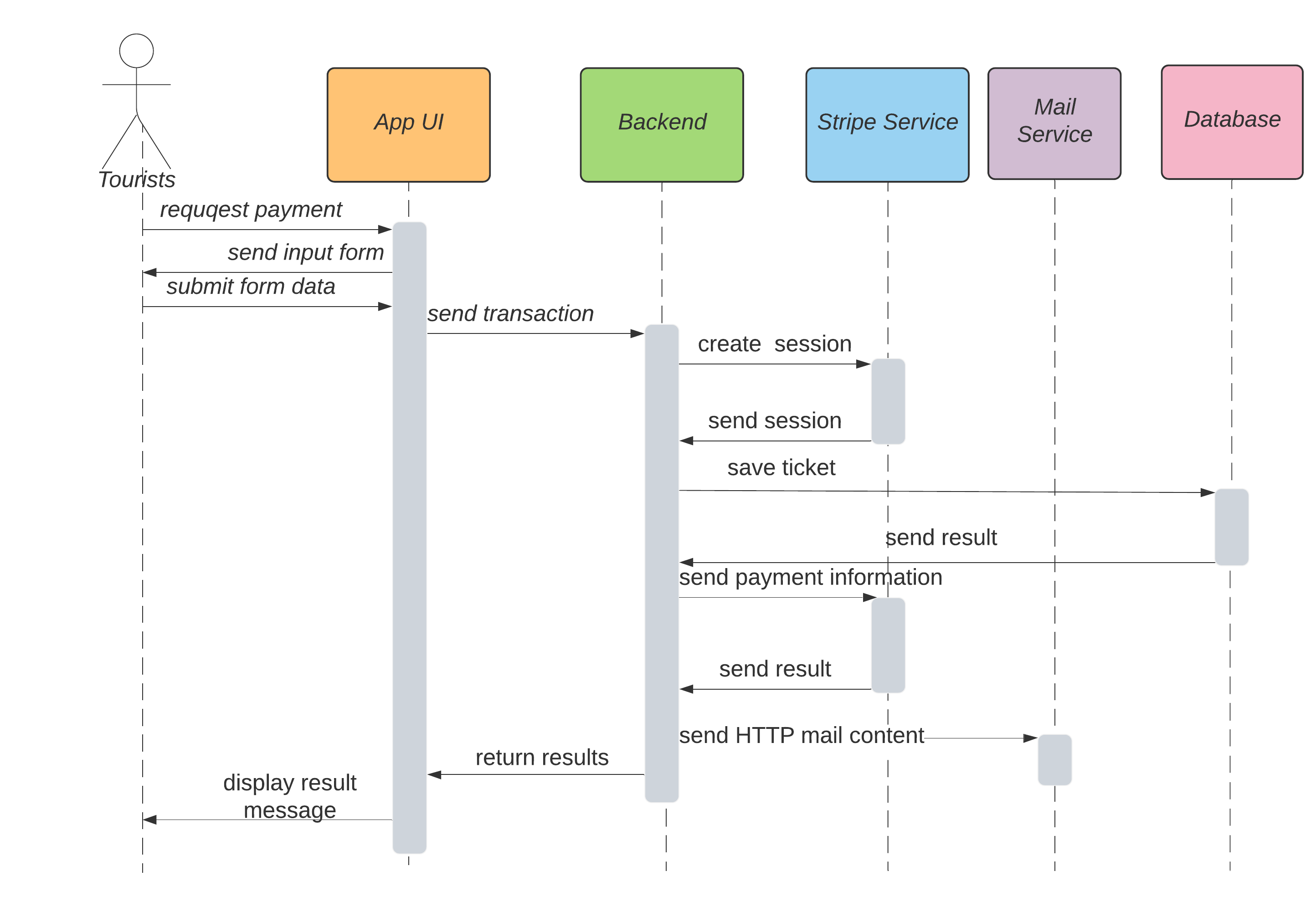


**Description:**

To login to the system, tourists have to register first.

Tourists are allowed to register a new account, to do that they create a new account with email, password and full name.

**3.3.3 Payment**

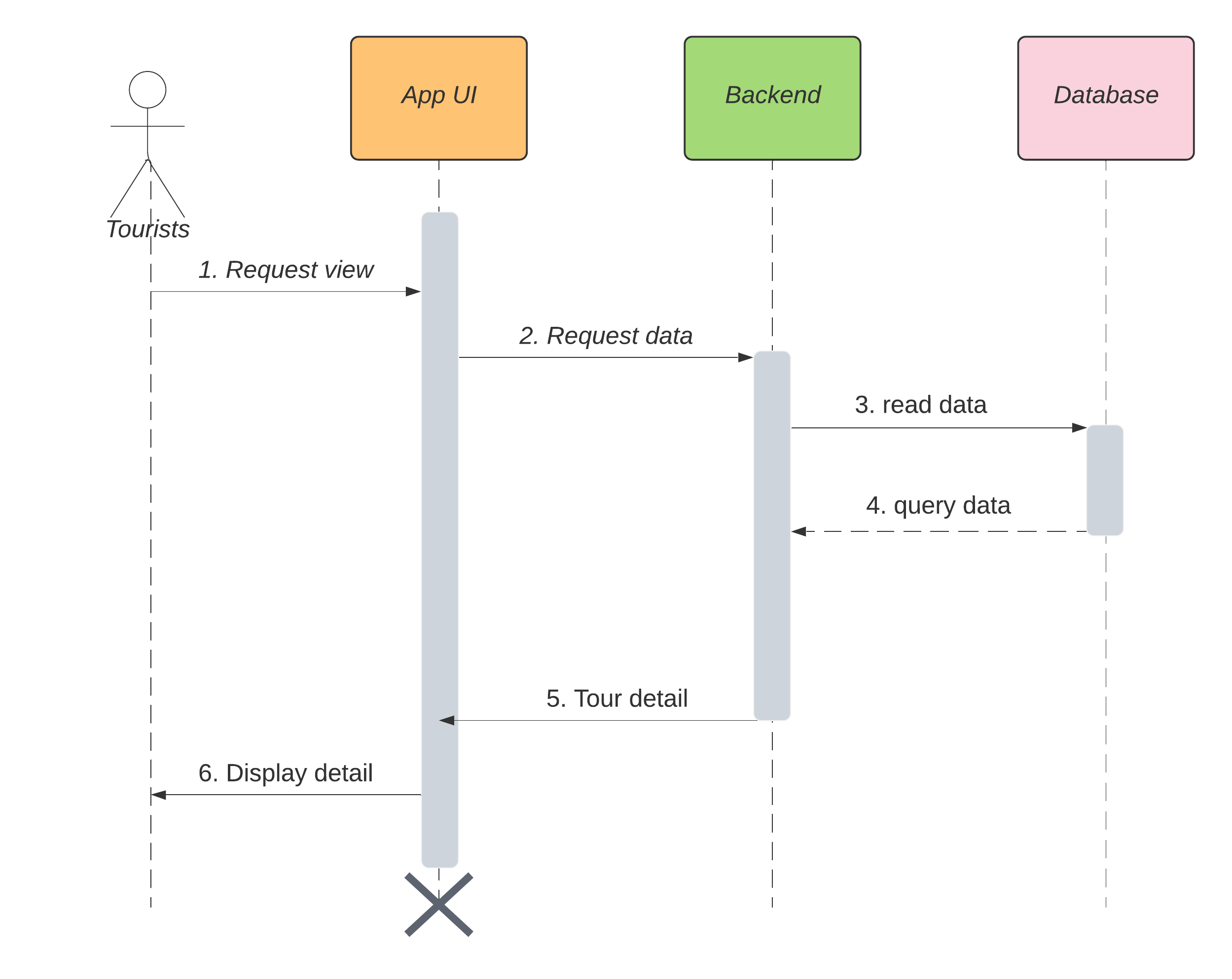


**Description:**

To pay, tourists have to login first.

To pay they need to choose for themselves at least one tour, fill in all contact information and confirm their phone number, and then make a payment with their bank account if successful they will receive a ticket which is sent via their email. Otherwise, they will receive an error message and the transaction is canceled

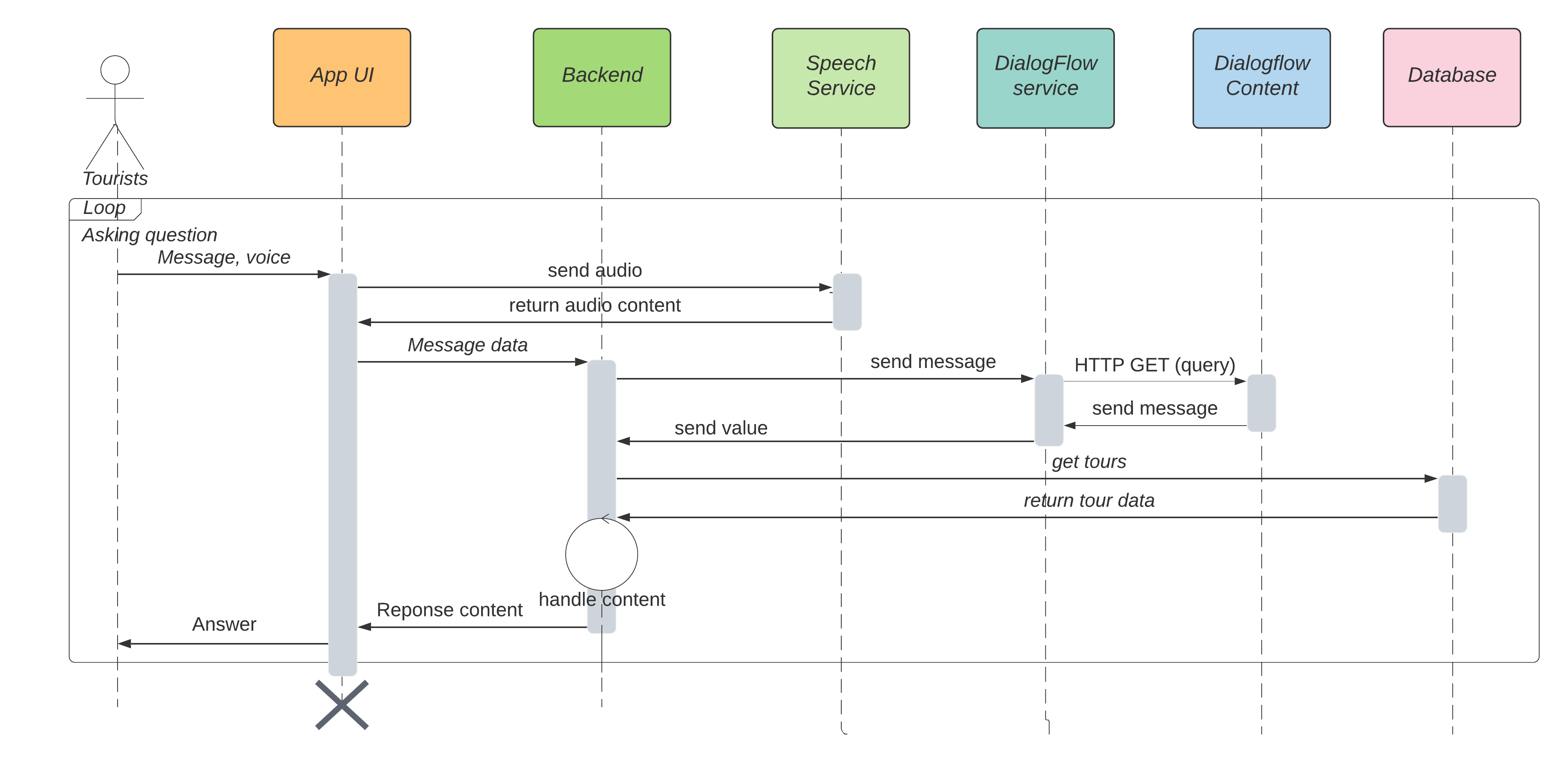
**3.3.4 View tour detail**



**Description:**

Tourists are allowed to view tour details in the app, to do that they have to let the system know which tour they want to view detailed information. We service receives requests, calls API to get data and return tour information.

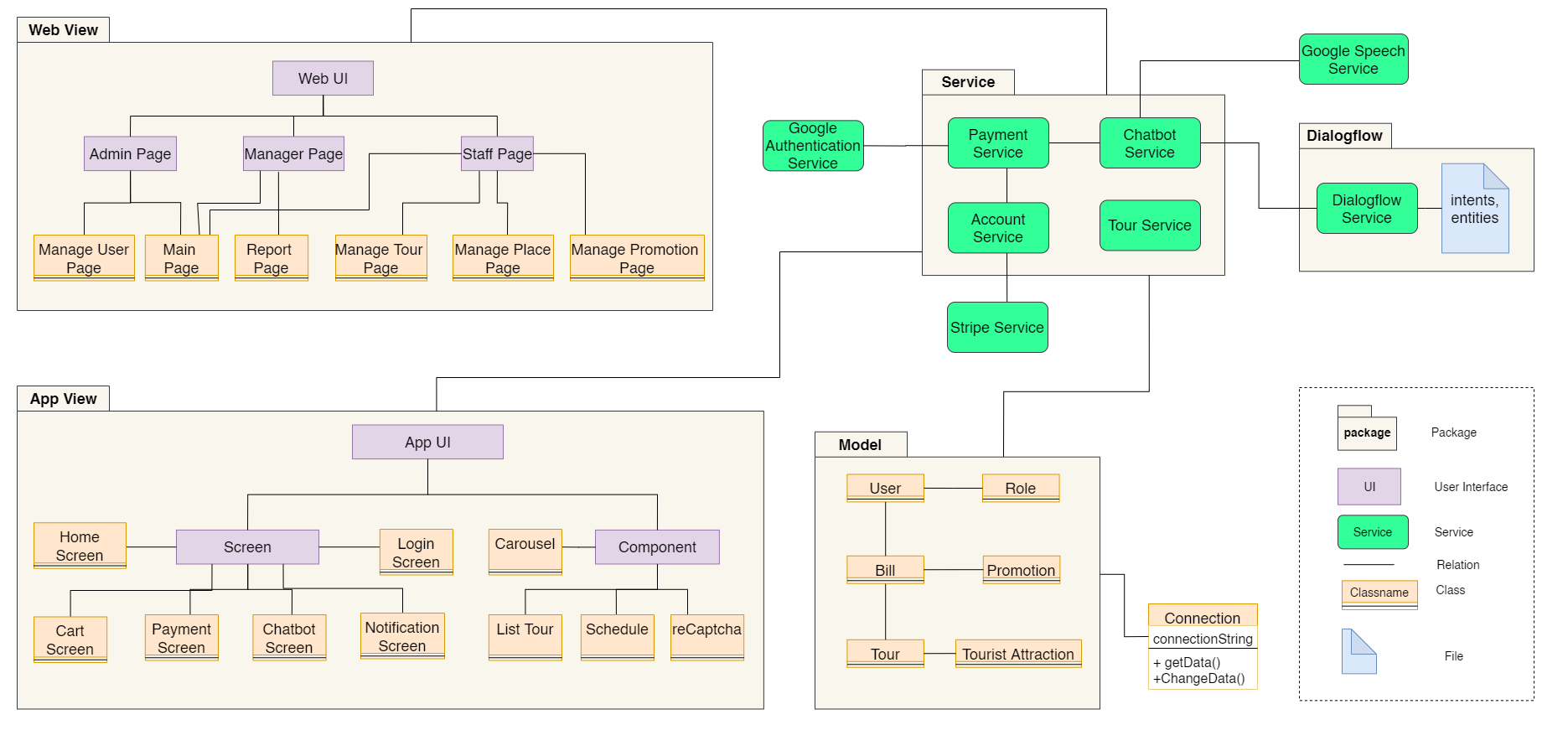
**3.3.5 Chat bot support**



**Description:**

Tourists are allowed to interact with chatbot to receive more traveling information, input is a chat message. Application reads data and calls APIs of Web Service. Then, it requests API of Dialog Flow Service and replies messages to tourists. Tourists can receive suggestions about tourist attractions, tour schedule and book their tour.

## 3.4 Module view



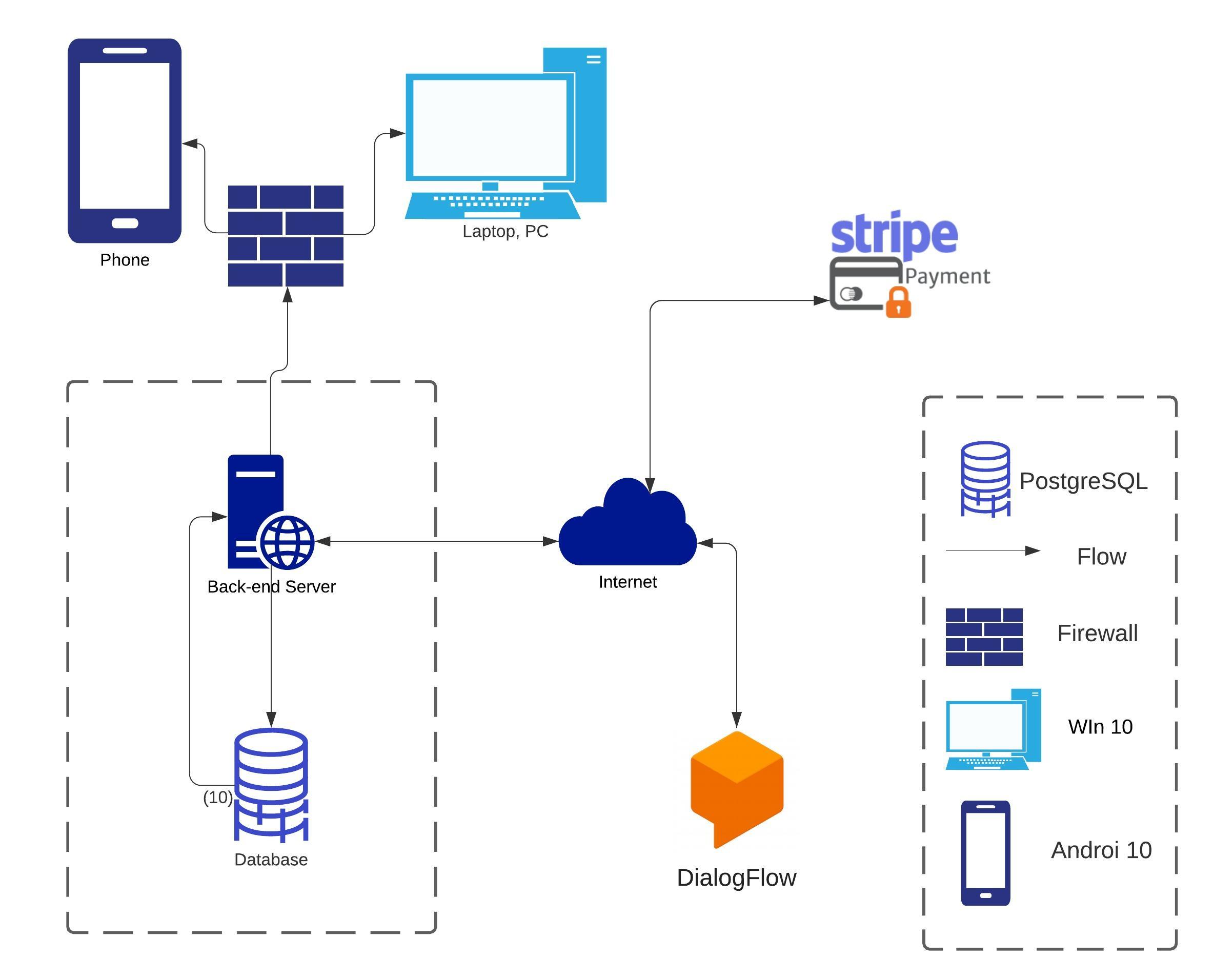
**Prose**

|  |  |
| --- | --- |
| **Element** | **Responsibilities** |
| Web View | The website package contains management modules for admin, staffs and manager. |
| App View | The application package contains interface modules for travelers to use the functionality of the app |
| Google Services | Authentication and speech support modules are used for login with google and voice chatbot. |
| Dialogflow Service | A natural language comprehension platform used to design and integrate chat user interface into bots |
| Stripe Service | The payment gateway provides a payment solution for in-app purchases |
| Models | Object data modules connect to tables in the database for transmission to the system interface |

**Description:** The system includes app view, web view, services, Dialogflow and models. After users interact with UI. Data will be requested to the corresponding service, then will direct to model or Dialogflow. Data be responded will be displayed on UI

## 3.5 Allocation view

The allocation view models the run-time architecture of a system. It shows the configuration of the hardware elements when the system is deployed.

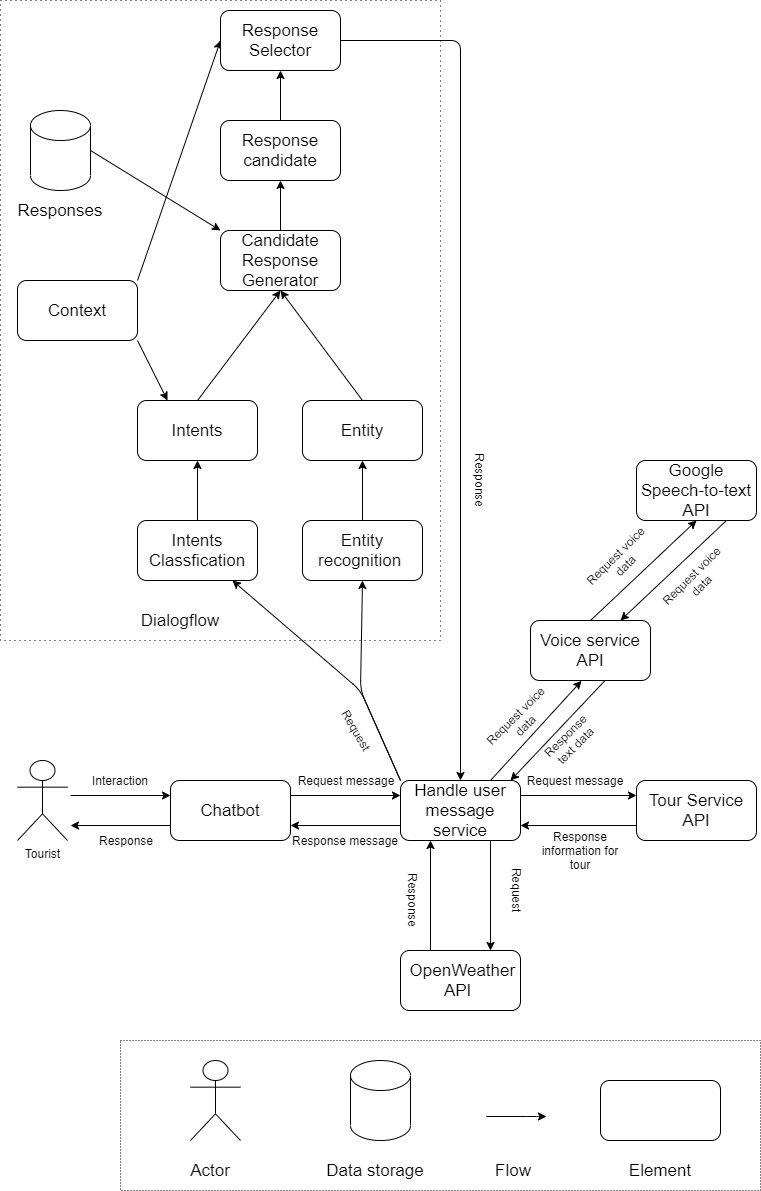


**Prose**

|  |  |
| --- | --- |
| **Element** | **Responsibilities** |
| Smart Phone | Tourists use this device to install and use vibotour app |
| Laptop or PC | Device running browser and helping Admin, Staff and Manager to use the functions of the website to manage. |
| Dialogflow | A natural language comprehension platform used to design and integrate chat user interface into bots |
| Stripe System | The payment gateway provides a payment solution for in-app purchases. |
| backend Server | Provide an API to support the interaction between the user interface and the server. where to install and run the backend API |
| PostgreSQL Database | The place contains all data about tours, user information,... It is organized in tabular form |

**Description**

The system is deployed on mobile environment (using React Native library) and web environment (using React JS library). They interact with the server through APIs to read and write data from the PostgreSQL database. In addition, the system interacts with Dialogflow to design conversations for chatbot and Stripe supports payments

* 1. **Workflow of chatbot**

**Description**

When tourist interacts with chatbot, the chatbot will analyze, process and send the request to the necessary service. The Services will process it and then send the response back to the chatbot to show it to tourist

# ATAM

## 4.1 Present the ATAM

* Overall evaluation of system architecture documents, system designs on 3 views: static view, dynamic view, and physical view based on ATAM 9 Steps method.
* Expect to achieve an accurate and objective evaluation of the architectural document. From there, the project team assesses the ability to complete the project and achieve the Architecture Drivers.

## 4.2 Present the business Drivers

* The content on the document presented about the following:
  + Who are the business drivers.
  + Business problems and goals for the system are presented by the Project decision makers.
  + System’s features.
  + System’s requirements.
  + Project constraints.
  + Project scope.

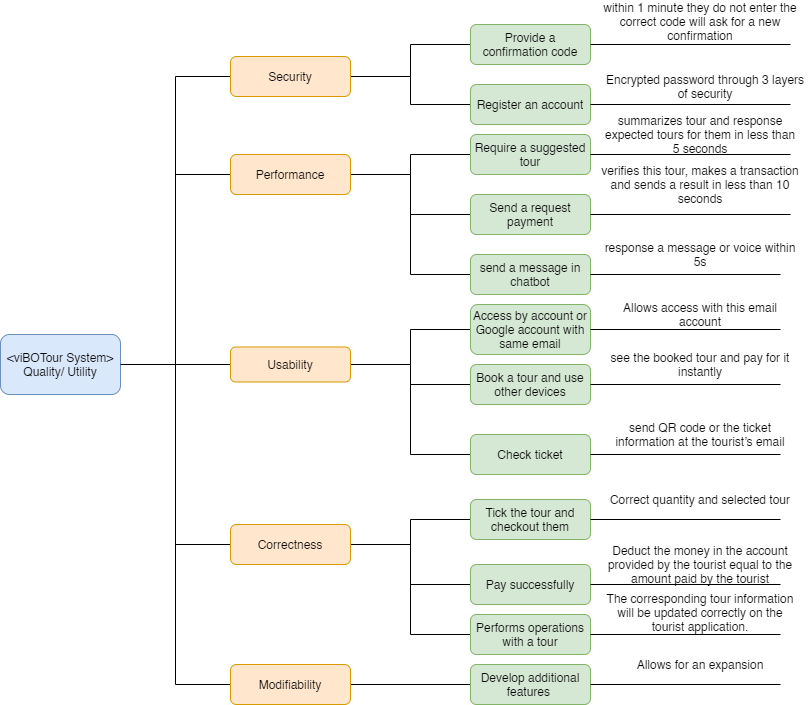
## 4.3 Present the Architecture

* Current Architecture state: The design is systematically overviewed on all 3 views: allocation view, module view and component and connector view.
* Expected Architecture state: The architecture is easy to understand, easy to read, full of content, clear and responsive to the constraints and Quality Attributes of the system.
* Impact of following project constraints in the architecture:
  + Time / Deadline: Project will be finished in 104 days (980 hours)
  + Cost / Available resources: 4 people with cost $2352
  + Complexity of the problem: high
  + Quality expectations: Meet the constraints and 5 Quality Attributes described above include: Security, Performance, Usability, Correctness and Modifiability.

## 4.4 Identify the Architecture approaches

* Architecture pattern: Service-oriented architecture(SOA)
* In SOA, services use protocols that describe how they pass and parse messages using description metadata. This metadata describes both the functional characteristics of the service and quality-of-service characteristics. Service-oriented architecture aims to allow users to combine large chunks of functionality to form applications which are built purely from existing services and combining them in an ad hoc manner. A service presents a simple interface to the requester that abstracts away the underlying complexity acting as a black box. Further users can also access these independent services without any knowledge of their internal implementation
* The architectural blueprints are broken down into sections and interact with the services.

## 4.5 Create a Quality Attribute Tree



## 4.6 Analyze the Architectural approaches

|  |  |
| --- | --- |
|  | Evaluate |
| Tradeoffs | + With a service bus system, the performance level is enhanced and the trade-off in system security decreases  + Ease of use pays off with system performance. Meeting good performance reduces ease of use and vice versa |
| Sensitivity points | + Dependence on the services of external systems  + Depends quite a lot on the network system and the data transmission speed of the services.  + No data backup solution yet |
| Risk and non-risk scenarios | + When security is threatened, hackers attack services, security can be affected.  + Network problem occurred.  + A service is dead. |

## 4.7 Brainstorm and prioritize scenarios

* Rank priority based on the constraints and attributes(descending):
  + Performance
  + Security
  + Correctness
  + Usability
  + Modifiability

## 4.8 Re-analyze the architectural approaches

* Validate with the system architect to discover and achieve with the system design.

## 4.9 Present the results

* Based on the above reviews:
  + The system can accommodate a number of Quality Attributes and constraints given.
  + However, some systemic risks will appear affecting the system and the Quality Attributes will be in order of priority.

# References:

|  |  |  |
| --- | --- | --- |
| **No.** | **References** | **Document Information** |
| 1 | Design standards,  Document standards | https://www.softwarearchitecturebook.com/svn/main/slides/ppt/26\_Standards.ppt |
| https://standards.ieee.org/standard/1471-2000.html |
| https://ieeexplore.ieee.org/document/917550 |
| 2 | Patterns | https://en.wikipedia.org/wiki/Architectural\_pattern |
| 3. | Evaluation standards | https://www.iso.org/obp/ui/#iso:std:iso-iec-ieee:42030:ed-1:v1:en |
| https://gabrielfs7.github.io/software-architecture/2019/10/18/atam-analyze-evaluate-architecture/ |