

# Software Specification: Tripedia

Yulin Zhang

7th Group

Software Engineering

Montreal, Canada

silveralex2023820@gmail.com

Yuhang Chen

7th Group

Software Engineering

Montreal, Canada

yuhang.chen@mail.concordia.ca

Jiaxi Yang

7th Group

Software Engineering

Montreal, Canada

yjxyang2@outlook.com

Boyang Wang

7th Group

Software Engineering

Montreal, Canada

wangboyang0626@outlook.com

## I. INTRODUCTION

### A. Problem Statement

Travel has gradually become an indispensable part of people's lives. According to relevant research, 83% of Canadians will travel to different cities to experience different customs during their holidays. Travel planning has become a problem that troubles most Canadians. At the same time, many people don't know which places are more suitable for them. Therefore, travel recommendations and planning have become essential issues affecting people's quality of life.

### B. Architecture Design

As shown in Figure1, the ICDE system architecture mainly contains two parts: client and server. Based on the limited server hardware equipment, we implement the data analysis logic as a single module in the server, which stands for ICDE 3rd Party Applications.

In the client part, we implement Tripedia's client logic based on the web. The web client service is mainly divided into two parts: the Data Capture Module and User Assistant Module. The Data Capture Module captures the user's behavior and personal information through questionnaires and behavior captures, and transmits the relevant information to the server. The User Assistant Module is responsible for displaying processed data sent by the server, such as recommendation results and data analysis results, which help users make decisions easily.

On the server side, we will develop three modules to implement the backbone of the ICDE system: Data-processing Module, Data-analyzing Module, and Data Access Service. The Data Access Service, as the median layer for database access, provides a unified database interface (deletion, modification, query, etc) for the Data-processing Module and the Data-analyzing module. The Data-processing Module is responsible for the basic logic of the server, such as registration, login, search, data preprocessing, and other interaction logic with the client. The Data-analyzing Module accesses the data in the database through the Data Access Service to perform data analysis and returns the analysis results to the Data-processing Module.

To sum up, Tripedia's architecture is a standard ICDE system. The Client part includes a Data Capture Service to collect user behavior and information and transmit relevant data to the server. The Data-processing Module in the Servers

section is responsible for implementing the interactive logic of the server; the Data Access Service provides a unified data access interface; the Data-analyzing Module serves as the core of the ICDE system to analyze user data and feedback the analysis results to the user through the server to provide User Assistance Service.

## II. OVERVIEW

### A. Project Goal

The goal of the project is to solve the travel planning difficulties faced by users. In response to travel planning issues, this project set out to develop search features to help users. Meanwhile, some users don't know where is more suitable for travel. This project provides clear charts through the background data analysis function to help users make better decisions.

### B. Key Functionalities

**Search** The search feature is responsible for obtaining search results based on the user's search keywords and collected user preferences. This function includes searching for tourist cities, searching for hotels, etc. The search function can provide users with travel route planning.

**Data Analysis** If the user doesn't know where is better for vacation, the website provides a data analysis function. Based on current user preferences and existing data in the system, the website provides data analysis charts of tourist cities suitable for users to help users make better decisions.

### C. User Stories

1. The traveler opens the travel website without logging in, the website will collect his location, so that the website can show local people's favorite place to visit which provides a reference for the user.

2. The traveler registers an account and login to the Tripedia website.

3. The traveler finishes a questionnaire so that the website can recommend appropriate information for users based on the questionnaire content.

4. The traveler has no idea about the where is the best places for him. He finds the data analysis diagram panel for more information to make a decision.

5. The traveler searches for his destination and then the website recommends related information (e.t. hotel, transport

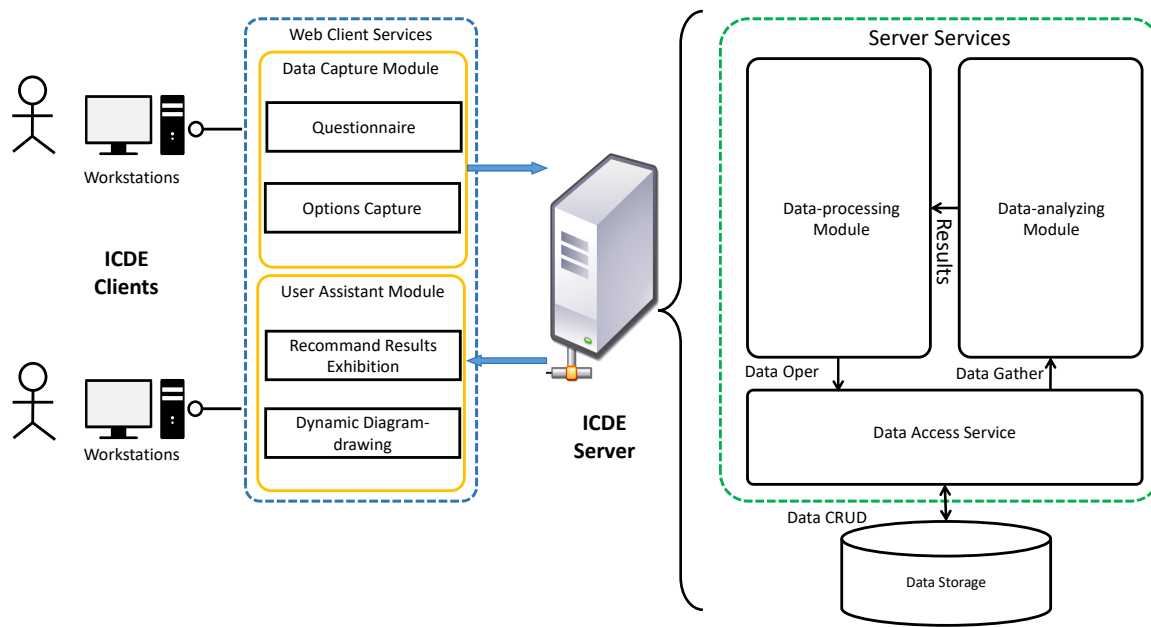


Fig. 1. The architecture graph of Tripedia.

function, Landmark Tickets, scenic spot review) so that he can make plans by reference.

6. The traveler checks suitable hotel timings so that he can book an appropriate hotel.

7. The traveler checks suitable flight or train timing so that he can book an appropriate transport function.

8. The traveler selects his favorite hotel and then clicks to jump to the booking screen so that he can make payment.

#### D. User Cases

##### 1) User Cases: Open Website:

**Use Case:** Open Website without Login.

**Actors:** Traveller, Client, Server, Analysis Module.

**Summary Description:** Allows any travelers to open Tripedia, and Tripedia shows some basic information about traveling according to the traveler's position.

**Status:** Medium level of details.

**Pre-condition:** Traveler has to register an account and log in to the Tripedia website to use its service.

**Post-condition:** The traveler has to have a personal computer and an available network environment.

##### **Basic-Path:**

1. The traveler enters the Tripedia website in a browser.
2. The Tripedia website catches the user's position and sends the basic information to the server.

3. The server recommends the travel information entry based on the user's position.

4. The website displays the available travel information sent by the server.

##### **Alternate Paths:**

None.

##### 2) User Cases: Register and Login:

**Use Case:** Register

**Actors:** Traveller, Client, Server, Data Store.

**Summary Description:** Allows any traveller register an account and login in the Tripedia website.

**Status:** Medium level of details.

**Pre-condition:** The traveler has to open the Tripedia website without login.

**Post-condition:** Traveler has to complete the questionnaire to use Tripedia's full service.

##### **Basic-Path:**

1. The traveler clicks the "register" button on the home page and enters the register page.
2. The traveler fills in the register information: user name, nationality, gender, and password. Then the traveler clicks the "submit" button.
3. The server validates the username is repetitive in the system, and saves the register information to Data Store.

4. The client displays the "register successfully" information and enters the login page.

5. The traveler fills in the user information: user name, and password. The the traveller clicks "login" button.

6. The server recommends the travel information entry based on the user's information and transfers the recommended information to the client.

7. The client enters the home page and displays the information entry sent by the server.

**Alternate Paths:**

3a. The user name is repetitive in the system.

5a. The user forgets username/password.

**3) User Cases: Questionnaire:**

**Use Case:** Questionnaire

**Actors:** Traveller, Client, Server, Data Store.

**Summary Description:** Allows any traveler within the login state to have opportunities to complete the questionnaire to get appropriate recommendations.

**Status:** Medium level of details.

**Pre-condition:** Traveler must be in login state.

**Post-condition:** None

**Basic-Path:**

1. The traveler logs in to the website firstly, the website pops up a dialog to ask the user to fill out a questionnaire.

2. The traveler clicks "yes", and the client enters the questionnaire page.

3. The website has several questions about preference. The traveler can select one answer for each question.

4. The traveler clicks the "submit" button, and the answers are sent to the server.

5. The server recommends the travel information entry based on the user's answers and transfers the recommended information to the client.

7. The client enters the home page and displays the information entry sent by the server.

**Alternate Paths:**

2a. The user abandons the chance to complete the questionnaire.

**4) User Cases: Data Analysis Diagram:**

**Use Case:** Questionnaire

**Actors:** Traveller, Client, Server.

**Summary Description:** Allows any traveler within the login state to check the data analysis diagram.

**Status:** Medium level of details.

**Pre-condition:** Travelers must be in login state and the website is on home page.

**Post-condition:** Travelers can search the recommended cities in the search area for more information.

**Basic-Path:**

1. The traveler clicks the "Analysis Panel" button.

2. The client enters the Data Analysis Page.

3. The server sends the analyzed data based on the user's information to the client.

4. The website displays the diagrams of recommended cities, such as the visitors' number, the visitors' nationalities proportion, etc.

**Alternate Paths:** None.

**5) User Cases: Recommend related information:**

**Use Case:** Recommendation

**Actors:** Traveller, Client, Server.

**Summary Description:** Allows any traveler to search for his destination, then the website recommends related information (e.t. hotel, transport function, Landmark Tickets, scenic spot review)

**Status:** Medium level of details.

**Pre-condition:** The traveler has to open the Tripedia website.

**Post-condition:** Website displays the relevant recommendation information.

**Basic-Path:**

1. The user enters the website.

2. The user enters the destination in the input box.

3. The user clicks the 'search' button

4. The website displays the relevant recommendation information in the form of a list.( Information including food, accommodation, transportation and shopping)

**Alternate Paths:**

1a. The user sign in the website.

4a. The website displays the relevant recommendation information in the form of a list.( Information related to the personal data of the user including food, accommodation, transportation and shopping)

**6) User Cases: Search hotels:**

**Use Case:** Hotels

**Actors:** Traveller, Client, Server.

**Summary Description:** Allows any traveler to search right hotels

**Status:** Medium level of details.

**Pre-condition:** The traveler has to open the Tripedia website.

**Post-condition:** Website displays suitable hotels.

**Basic-Path:**

1. The user enters the website.

2. The user enters the destination in the input box.

3. The user chooses the check-in date and check-out date from the date-form.

4. The user clicks the 'search' button

5. The website displays suitable hotels

**Alternate Paths:** None

**7) User Cases: Search suitable transportation tickets:**

**Use Case:** Transportation

**Actors:** Traveller, Client, Server.

**Summary Description:** Allows any traveler to search the right mode of transportation

**Status:** Medium level of details.

**Pre-condition:** The traveler has to open the Tripedia website.

**Post-condition:** Website displays a suitable mode of transportation.

**Basic-Path:**

1. The user enters the website.

2. The user enters the destination in the input box.

3. The user chooses departure time and return time from the date-form.
4. The user clicks the 'search' button.
5. The website displays suitable tickets. (include flights or trains)
6. The user can choose his favorite transportation function and then add it to his wish list.

**Alternate Paths:** None

8) *User Cases: Check favorite hotel:*

**Use Case:** Details of hotel

**Actors:** Traveller, Client, Server.

**Summary Description:** Allows any traveler to check the details of hotel.

**Status:** Medium level of details.

**Pre-condition:** The traveler has to open the Tripedia website.

**Post-condition:** Website jumps to new page of hotel booking details.

**Basic-Path:**

1. The user enters the website.
2. The user enters the destination in the input box.
3. The user chooses check in date and check out date from the date-form.
4. The user clicks the 'search' button.
5. The website displays suitable hotels.
6. The user clicks one of the hotels.
7. The website jump to a new page of hotel booking details.

**Alternate Paths:** None

### III. FUNCTIONAL REQUIREMENTS

#### A. Login

1) *Enter Username and password:* The users can enter their username and password in the input box. The user can click on the Show Password button in the password box. When the user clicks on this button, the password will be displayed in plaintext in the password box so that the user can check if the password is correct before logging in.

2) *Validate:* When the user clicks on the login button if the user enters an incorrect password or username, the message "Username or password error" will be displayed. If the user enters the correct username and password, the system will automatically jump to the home page, and the user's login status will be displayed on the home page.

#### B. Register

Users can create a new account by entering a username, name, password, email, and other relevant personal information.

1) *Enter Personal Information:* When the user clicks on the register button, the page will display several input fields to let the user enter their personal information. After finishing the input, the user can click the "Create" button to create a new account.

2) *Verify Email Address :* After the user inputs their email address, they can click the "Verify the email" button, and then the system will send an email to the address. The user should open the email and click the link in the email to finish the registration. If the user doesn't receive the email, they can click the "don't receive the email" button and the system will send another email.

#### C. Editing Personal Information

1) *Change Password :* When the user wants to change their password, they can enter their own account page, and click "Change the password" to edit it. The user needs to type the old password first to ensure safety. If the user inputs the wrong password, a system will display the message " wrong password. You can't change the password." If the password is right, the user can type the new password and they should input it for the second time to confirm the new password. After confirming the new password, the user clicks on the "Save" button to save the change.

2) *Update Email Address :* When the user wants to change their email address, they can enter their own account page, and click "Change the email address" to edit it. The user can input the new email address and click the "Verify the email" button, and then the system will send an email to the address. The user should open the email and click the link in the email to finish the update. If the user doesn't receive the email, they can click the "don't receive the email" button, and the system will send another email.

3) *Edit Other Information :* When the user wants to change other personal information, they can enter their own account page to edit it. After inputting the new information, click the "Save" button to save the edit.

#### D. Search

Users can enter keywords, locations, or other criteria to search for travel destinations. A list of related travel destinations is displayed and the user can click to view details.

1) *Search Method :* Users should be able to search for the information by using the search bar. When users are typing, the system will list the related entries under the search bar to help users to search. Moreover, Users should be able to use filters and restrictions to narrow down search results. The system should be able to save the user's search history so that the user can quickly find the required information the next time.

2) *Search Result Display :* Search results should be clear and sorted in order of relevance so that users can quickly find the information they need. When there are too many search results, pagination should be supported. Users can enter the details page by displaying the results, which introduces specific information about the scenic spot, and the user can enter the official page of the tourist destination through the link displayed on the website to learn more.

#### E. Intelligent Recommendations

Users can fill out a questionnaire describing their travel preferences. Based on the user's questionnaire responses, the

system will automatically recommend suitable travel destinations.

1) *Questionnaire* : After the user registers, the user is asked to choose to skip the location recommendation or make a location recommendation. If the user chooses to recommend a location, the system will jump to the questionnaire page, and then after completing a simple questionnaire, the system will send the questionnaire information into the ICDE system for processing, and return and save the results of the recommendation to the user's database.

2) *Recommendation* : When a user clicks on the questionnaire button after registration and completes the questionnaire, the system will analyze the results of the questionnaire and display results in the Recommendations module on the home page of the website.

If you are a guest or a registered user who skipped the survey, the currently popular destinations, highly-rated restaurants, and hotels will be displayed in the Recommendations module of the web page.

#### IV. NONFUNCTIONAL REQUIREMENTS

##### A. *Performance*

In order to cope with high concurrency scenarios, it ensures that users still get fast responses during high loads, while maximizing the utilization of server resources. The system should be able to support multi-threaded processing, making full use of the server's multi-core resources to improve response speed and processing efficiency.

##### B. *Usability*

The website's interface should be easy to use, clear, and appropriate for all age groups of users. The user interface should be simple and intuitive. Elements such as color, fonts, and layout should all contribute to the user experience. When the user enters incorrect information, there will be a clear error message to guide the user on how to correct the error.

##### C. *Security*

To ensure the security of user passwords, the system performs hash processing to encrypt the passwords instead of storing them in the database in plaintext.

##### D. *Compatibility*

1) *Browser compatibility* : The website is compatible with mainstream browsers such as Chrome, Firefox, Safari, etc.

2) *Operate System compatibility* : Users can access the server page through computer operating systems such as Linux, Windows, macOS, and mobile device operating systems such as iOS and Android.