System Requirements Specification Index

For

Tourist Guide Application (MS SQL)

Version 4.0

IIHT Pvt. Ltd.
IIHT Ltd, No: 15, 2nd Floor, Sri Lakshmi Complex, Off MG Road, Near SBI LHO,
Bangalore, Karnataka – 560001, India
fullstack@iiht.com

TOURIST GUIDE APPLICATION

System Requirements Specification

1. Project Abstract

1.1 Problem Statement:

The purpose of this application is to allow the visitors/Traveller to view various destinations to travel, search destination details by name and Id, and hire a tour guide to show them all explorations or many more functions.

1.2 Following is the requirement specifications:

	Tourist Guide Application
Tourist Controller	
1	View All Destinations by name and Id.
2	Find a destination by name and attraction.
3	Hire a tour guide.
4	Find place details
5	Find tour guide details.
6	Know about all India destinations.

2. Template Code Structure

2.1 Package:TouristGuide

Names	Resource	Remarks
Package Structure/Project		
controller	TouristController	School controller handle all action method for respective user interface view(cshtml file)
Properties	launchsettings.json	Contain all URL settings for API.
	appsettings.json	Contains connection string for database.

2.2 Package:TouristGuide.BusinessLayer

Names	Resource	Remarks
Package Structure		
Interfaces	Interface directory contain all interface for Services class	Inside this directory contains all business logic code and CURD Operation Logic method.
Services, Repository	ITourguideServices, TourguideServices, ITourguideRepository, TourguideRepository cs file for Method and business logic	Using this all cs file performed all CRUD operation.
ViewModels	Cs file for represent all view entities	All view entities setting class

2.3 Package:TouristGuide.DataLayer

Names	Resource	Remarks
Package Structure		
TouristguideDbContext	Contain all business logic for data set and Dbset setting	Using this cs file performed all Data related settings operations.

2.4 Package:TouristGuide.Entities

Names	Resource	Remarks
Package Structure		
Entities Class	About India, ContactUs, Destination, Place, TourGuide cs file	Contain all entities property for application

2.5 Package: TouristGuide.Test

Resources

Note: - Under the TouristGuide.Test contains All Test cases for code evaluation, please don't try to alter and edit it.

3. Rest Endpoints

Rest End-points to be exposed in the controller along with method details for the same to be created

3.1 TouristController

URL Exposed			Purpose
/get-all-destinations	Http Method	GET	Show or all destinations
	Parameter 1	placeId	
	Return	all destinations	
/find-destination	Http Method	GET	Get place by PlaceId and its details
	Parameter 1	name	
	Return	<Place>	
/hire-tour-guide	Http Method	POST	Hire a tour guide
	Parameter 1	TourGuide tourGuide	
	Return	HTTP Status Code	
/tour-guide-info	Http Method	GET	Get tourist guide info
	Parameter 1	tourId	
	Return	<IEnumerable<TourGuide>>	
/about-india	Http Method	GET	Know the all india visiting destination
	Parameter 1	-	
	Return	<IEnumerable<AboutIndia>>	
/place-details	Http Method	GET	Get place details
	Parameter 1	placeId	
	Return	<Place>	

4.Business Validations

4.1 Entities:

- All the value for Place details: must not be null and min of 4 chars
- All the About India value: must not be null and min of 3 chars
- All the Destination: must not be null and min of 3 chars.
- All the TourGuide: must not be null.

4.2 Common Constraints:

- For all receiving Url parameter, validation check must be done and must throw custom exception if data is invalid
- Must not go and touch the test resources, as they will be used for Auto-Evaluation.

5. Execution Steps to Follow

1. **All actions like build, compile, running application, running test cases will be through Command Terminal.**

2. **To open the command terminal the test takers need to go to the Application menu (Three horizontal lines at left top)   Terminal → New Terminal.**

3. **On command prompt, cd into your project folder (cd <Your-Project-folder>).**

4. **To connect SQL server from terminal:**
   (Project_directory/TouristGuide/sqlcmd -S localhost -U sa -P pass@word1)
   - **To create database from terminal -**
     1. **Create Database TouristGuide_Db**
     2. **Go**

5. **Steps to Apply Migration(Code first approach):**
   - **Press Ctrl+C to get back to command prompt**
   - **Run following command to apply migration-**
     (Project_directory/ TouristGuide /dotnet-ef database update)

6. **To check whether migrations are applied from terminal:**
   (Project_directory/ TouristGuide /sqlcmd -S localhost -U sa -P pass@word1)

```
1> Use TouristGuide_Db
2> Go
1> Select * From __EFMigrationsHistory
2> Go
```

7. To build your project use command:
   (Project_directory/TouristGuide /dotnet build)

8. To launch your application, Run the following command to run the application:
   (Project_directory/TouristGuide /dotnet run)

9. This editor Auto Saves the code.

10. To test any Restful application, the last option on the left panel of IDE, you can find ThunderClient, which is the lightweight equivalent of POSTMAN.

11. To test any UI based application the second last option on the left panel of IDE, you can find Browser Preview, where you can launch the application.

12. To run the test cases in CMD, Run the following command to test the application:

13. (You can run this command multiple times to identify the test case status, and refactor code to make maximum test cases passed before final submission)
    (Project_directory/TouristGuide.Test/dotnet test --logger "console;verbosity=detailed").

14. If you want to exit(logout) and continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use CTRL+Shift+B - command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository. Else the code will not be available in the next login.

15. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.

16. You need to use CTRL+Shift+B - command compulsorily on code IDE, before final submission as well. This will push or save the updated contents in the internal git/repository, and will be used to evaluate the code quality.