**Apex Hospital**

Apex hospital is one of the best hospitals in India, They need to create a service application. Help them to implement the needed services using ASP.NET CORE WEB API

Major functionalities to cover through this Service application

   1.  **Hospital**- Create hospital and perform filter functionality

   2.  **Doctor**- CRUD Operation on doctor info and related business logic

   3.  **Patient** - Create patient details and perform filter functionality

#### 1.  **MODEL**

The models are already provided in the code template. In the models below, properties and   data types such as Hospital, Doctor and Patient are mentioned.

##### **Hospital.cs**

|  |  |
| --- | --- |
| **Property** | **Data Type** |
| WardId | int |
| WardName | string |
| HeadDoctorName | string |
| AvailableBedsCount | int |

##### **Doctor.cs**

|  |  |
| --- | --- |
| **Property** | **Data Type** |
| Id | int |
| FirstName | string |
| LastName | string |
| Gender | string |
| Hospital | Hospital |
| Contact | long |

##### **Patient.cs**

|  |  |
| --- | --- |
| **Property** | **Data Type** |
| Id | int |
| FirstName | string |
| LastName | string |
| Gender | string |
| Hospital | Hospital |
| BedNumber | int |
| Contact | long |

#### 2.  **CONTROLLER**

Implement below controllers such as Hospital, Doctor and Patient controller

##### **HospitalController.cs**

IHospital should be injected in HospitalController

|  |  |  |  |
| --- | --- | --- | --- |
| **Service** | **Http Type &     Return Type** | **Functionality** | **Repository ( Check HospitalRepository Section )** |
| GetAllDetails | **GET**Method    ActionResult | This service is used to get all the ward details | Call **"GetAllDetails"** |
| GetWardDetailsById/{id} | **GET**Method    ActionResult | This service is used to return particular ward details  by Id  Parameter type is **'int'** | Call **"GetWardDetailsById"** |
| GetWardWithMaxBedCountAvailable | **GET**Method    ActionResult | This service is used to get details of the ward which has the maximum bed count | Call **"GetWardWithMaxBedCountAvailable"** |

##### **DoctorController.cs**

IDoctor should be injected in DoctorController

|  |  |  |  |
| --- | --- | --- | --- |
| **Service** | **Http Type &     Return Type** | **Functionality** | **Repository ( Check DoctorRepository Section )** |
| GetAllDoctors | **GET**Method    ActionResult | This service is used to get all the doctors details | Call **"GetAllDoctors"** |
| GetDoctorsByWardName/{wardName} | **GET**Method    ActionResult | This service is used to get all the doctors details based on the ward name  Parameter type is **'string’** | Call **"GetDoctorsByWardName"** |
| AddDoctor | **POST**  Method  IActionResult | This service is used to insert doctor into the doctors list  If the result is null or zero, it should return **status 500** | Call **"AddDoctor"** |
| UpdateDoctorContact/{id} | **PUT**Method   ActionResult | This service is used to update doctor's contact number using by Id  Parameter type Id - **int**  Contact- **long** (pass with in the body)  If it has  any exceptions it should return **status 500**OR return **OK** | Call**"UpdateDoctorContact”** |
| RemoveDoctor/{id} | **DELETE**  Method   ActionResult | This service is used to delete doctor by their Id  Parameter type is **'int'**  If it has  any exceptions it should return **status 500**OR return **OK** | Call **"RemoveDoctor"** |

##### **PatientController.cs**

IPatient should be injected in PatientController

|  |  |  |  |
| --- | --- | --- | --- |
| **Service** | **Http Type &     Return Type** | **Functionality** | **Repository ( Check DoctorRepository Section )** |
| GetAllPatients | **GET**Method    ActionResult | This service is used to get all the patients details | Call **"GetAllPatients"** |
| AddPatient | **POST**Method    ActionResult | This service is used to add a new patient details. Add the patient object into the list    If it has  any exceptions it should return **status 500**OR return **OK** | Call **"AddPatient"** |
| UpdatePatientWardName/{id} | **PUT**Method    IActionResult | This service is used to update the patient’s ward name based on the id.  Parameter type Id - **int**  wardName- **string** (pass with in the body)  If it has  any exceptions it should return **status 500**OR return **OK** | Call **"UpdatePatientWardName"** |

**NOTE: If you need to handle exceptions, use a try catch block.**

#### 3.  **REPOSITORY**

##### **StaticData.cs**

1)  This class contains three variables and its sample data. These datas should be used for the entire application(CRUD  Operations).

2)  No need to configure the models, the application will consume sample data from StaticData.cs file. Do not make any changes the models and static data files. Should not edit OR remove the sample data and the variable name.

Implement below repositorys such as Hospital, Doctor and Patient repository

##### **HospitalRepository.cs**

1)  Create variable for Hospital with the name of "lsWards".

**List<Hospital> lsWards = StaticData.lsWards;**

2) "IHospital" Interface already created and given in the skeleton, implement the interface in **HospitalRepository.cs**

3)  Implement below functionalities into **HospitalRepository.cs**

|  |  |
| --- | --- |
| **Method** | **Functionality** |
| public List<Hospital> GetAllDetails() | This method is used to get all wards list from  **'lsWards'** |
| public Hospital GetWardDetailsById(int id) | This method is used to get particular ward details by ward id |
| public Hospital GetWardWithMaxBedCountAvailable() | This method is used to get the details of the ward which has the maximum bed count |

##### **DoctorRepository.cs**

1)  Create variable for Doctor with the name of "lsDoctors".

**List<Doctor> lsDoctors = StaticData.lsDoctors;**

2) "IDoctor" Interface already created and given in the skeleton, implement the interface in **DoctorRepository.cs**

3)  Implement below functionalities into **DoctorRepository.cs**

|  |  |
| --- | --- |
| **Method** | **Functionality** |
| public List<Doctor> GetAllDoctors() | This method is used to get all doctors list from  **'lsDoctors'** |
| public List<Doctor> GetDoctorsByWardName(string wardName) | This method is used to get doctors details based on the ward name |
| public bool AddDoctor(Doctor doctor) | This method is used to Insert doctor details into  the **'lsDoctors'** list    If doctor added successfully, it should return true else return false |
| public bool UpdateDoctorContact(int id, long contact) | This service is used to update doctor contact by their Id    If doctor contact updated successfully, it should  return true else return false |
| public bool RemoveDoctor(int id) | This service is used to delete doctor by their Id |

##### **PatientRepository.cs**

1)  Create variable for Patient with the name of "lsPatients”.

**List<Patient> lsPatients = StaticData.lsPatients;**

2) "IPatient" Interface already created and given in the skeleton, implement the interface in **PatientRepository.cs**

3)  Implement below functionalities into **PatientRepository.cs**

|  |  |
| --- | --- |
| **Method** | **Functionality** |
| public List<Patient> GetAllPatients() | This method is used to get all patients list from  **'lsPatients'** |
| public bool AddPatient(Patient patient) | This method is used to insert data into **'lsPatients'**    If patient details added successfully, it should return true else return false |
| public bool UpdatePatientWardName(int id, string wardName) | This method is used to update patient’s ward name by patient Id    If patient details updated successfully, it should  return true else return false |

#### 4.  **BUSINESS RULES**

1)  Doctor Id should be unique, Should return false the "**AddDoctor**" method while existing Doctor Id

2)  "**UpdateDoctorContact**" method should return false while parameter Id or contact is equal to zero OR the specific id record could not be found.

3)  "**RemoveDoctor**" method should return false while invalid parameter Id. Which means the Id should not be zero OR Id not available in the local list variable.

4)  Patient Id should be unique, Should return false the "**AddPatient**" method while existing patient Id.

#### 5.  **SAMPLE OUTPUT**























