Desktop Allocation System

TcsDesktopAllocationSystem needs an application for allocating desktops to trainees with the following modules:

1. **Add new trainee**
2. **Get desktop allocation details of each trainee**

The description of different classes is given below:

Class Descriptions

1. Model classes   (com.tcs.dto)

      TraineeDTO
            - id: Integer
            - name: String
            - phoneNo: String
            - desktop: Desktop

      DesktopDTO
            - machineName: String
            - make: String

2. SpringConfig class   (com.tcs.configuration)

This is the configuration class of application. Annotate it proper annotations do declare it as Spring configuration class and auto scanning of beans present in com.tcs.service and com.tcs.repository package.

3. Validator class   (com.tcs.validator)

This class is used for validating the input values. Implement this class according to class diagram and description given below:

Note : The methods of this class are static.

Method description

- validate(TraineeDTO trainee)
  - This method validates the trainee details by invoking respective validator methods. If validation fails, throw TcsDASystemException with message as shown in the table below:

Violation For	Message
traineeId	Validator.INVALID_DETAILS

- validateTraineePhoneNo(String phoneNo)
  - This method validates the received phoneNo.
  - The length of the phoneNo should be 10 digits.
  - The first number in the contactNo should be either 7,8 or 9.
  - If the above conditions satisfy, return true else return false.

4. TcsDASystemServiceImpl   (com.tcs.service)

This is the service class of application. Implement this class according to class diagram and description given below:

    TcsDASystemServiceImpl
        - tcsDASystemRepository: TcsDASystemRepository

- Annotate it with proper stereotype annotation with value "tcsDASystemService":
- Inject the tcsDASystemRepository using autowiring.

Method description

- addNewTrainee(TraineeDTO trainee)
  - It invokes validate() of Validator class to validate the trainee details.
  - If all the details are valid, invoke addNewTrainee() of TcsDASystemRepositoryImpl, which in turn returns an integer.
  - It returns the integer received in previous step.

- ○ It logs TcsDASystemException thrown from service and validator class and re-throw them.
- getAllocationDetails(Integer traineeId)
    - ○ This method returns the desktop allocation details for the given traineeId.
    - ○ It invokes the getAllocationDetails() method of TcsDASystemRepositoryImpl which returns the trainee details.
    - ○ If trainee details are not found it throws an TcsDASystemException with message "Service.NO_DETAILS_FOUND", else returns the trainee details found in previous step.
    - ○ It logs TcsDASystemException thrown from service class and re-throw them.

5. TcsDASystemRepositoryImpl   (com.tcs.repository)

This is the Repository class of this application and its methods are already implemented. You have to annotate it with proper stereotype annotation with value "tcsDASystemRepository":

Method description

- addNewTrainee(TraineeDTO trainee)
    - ○ This method is used to save the trainee details in database.
- getAllocationDetails(Integer traineeId)
    - ○ This method returns desktop allocation details of a trainee.

6. UserInterface class   (com.tcs.ui)

UserInterface class is used as an interface between the application and the user.

UserInterface
+ LOGGER: Log

You can modify the inputs to check output for various scenarios. Some valid and invalid input values and expected output is given below:

addNewTrainee()

- For valid trainee details:

Variable	Value
id	1008
name	Rohit
phoneNo	9887766554

- The output should be logged in console as: Trainee added successfully with id: 1008
- For invalid trainee details

Variable	Value
id	1008
name	Rohit
phoneNo	988754

The output should be logged in console as :Invalid trainee details

getAllocationDetails()

- For valid traineeId :1001, output should be logged in console as:

TraineeDTO [id=1001, name=Aman, phoneNo=9785746325, desktop=DesktopDTO [machineName=TCSTR111111D, make=Acer]]

- For invalid traineeId = 1002, output should be logged in console as: No trainee exists for the specified id.