**Interview Tracker System Requirements Document**

**Purpose:** Interview Tracker is simple Spring boot application with MongoDb, where it allows any unregistered users (visitors) also to manage the user and interview like create, view, modify and delete.

Every user should have the details like userId, firstName, lastName, email, reportingTo, userType, userStatus, mobile and interview details like interviewId, interviewerName, interviewName, user (Many to many relationship with user), usersSkill, time, date, interviewStatus, technicalInterviewStatus, remarks etc.

.

Visitors can perform the follow actions:

1. Allows to add a user
2. Allows to delete an existing user
3. Allows to update the user status on go
4. Allows to display all the users
5. Allow to add interview
6. Allow to delete an existing interview
7. Allow to update interview status on go
8. Allows to search the interview on the basis of InterviewName and InterviewerName
9. Allows to search total no of count of an interview.
10. Allows to display all the interviews
11. Allow to add attendee
12. Allow to show interview

**Software Requirements:**Springboot, RESTful Web Services, MongoDb, Postman.

**Scope of the project**: Create a Server-side application.

**Assumptions, Dependencies, Risks / Constraints:**

1. While adding a user and interview , if user id and interview id is already exists, it should throw a custom exception.
2. While updating, it is assumed that id already exists and if new id is given add it.
3. While deleting the user and interview also, assumed that id already exists.
4. For all rest endpoints receiving @RequestBody, validation check must be done and must throw custom exception if data is invalid
5. Must not go and touch the test resources, as they will be used for Auto-Evaluation
6. All the business validations must be implemented on DTO object only.
7. All the database operations must be implemented on model / entity object.
8. The conversion from model to dto can be done in the service layer.
9. The methods required for the conversion is provided in the UserUtilities and InterviewUtilities class in utils package in the project.

**URL’s Exposed in the controller**

|  |  |
| --- | --- |
| **Url Exposed** | **Purpose** |
| /user/addUser | To add user |
| /user/viewAllUsers | To view all users |
| /user/updateUser | To update an existing user |
| /user/deleteUser | To delete an existing user |
| /interview/addInterview | To add a new interview info |
| /interview/addAttendee/{interviewId}/{userId} | To add attendee |
| /interview/updateInterview | To update an existing interview |
| /interview/deleteInterview | To delete an existing interview |
| /interview/viewAllInterviews | To view all interviews |
| /interview/showInterview/{interviewId} | To show interview |
| /interview/search/interviewName/{interviewName} | To search interview by interview name |
| /interview/search/interviewerName/{interviewerName} | To search interview by interviewer name |
| /interview/count | To see total no of interview count |
|  |  |

**Business Validations:**

1. user id and interview id must be not null and unique
2. firstName is not null , min 5 and max 30 characters.
3. lastName is not null , min 3 and max 25 characters.
4. email is not null , and it should be valid mail id.
5. reportingTo is not null , min 3 and max 25 characters.
6. userType is not null , min 3 and max 25 characters and it should be Trainee, Trainee Intern, Manager, Developer, Fresher Associate, Team Lead, Team Supervisor, Delivery Lead, Production Lead or Not Assigned.
7. userStatus is not null , min 3 and max 25 characters and it should be Locked, Bench, In-training or Released.
8. mobile is not null , min 10 and max 10 character.
9. interviewerName is not null , min 5 and max 30 characters.
10. interviewName is not null , min 3 and max 30 characters
11. usersSkill is not null , min 5 and max 30 character.
12. interviewStatus is not null , min 5 and max 100 character.
13. technicalInterviewStatus is not null , min 5 and max 100 character.
14. remarks is not null , min 5 and max 100 character.

**Resources Available in the platform**

|  |  |  |
| --- | --- | --- |
| Name | Resource | Remarks |
| Package Structure |  |  |
| Controller | UserController  InterviewController | Two Controller class to handle user and interview request seperatelly. |
| Model | User  Interview  Attendee | Entity/Document class |
| Dto | UserDTO,  InterviewDTO  AttendeeDTO | Both are the DTO class |
| Exception | InsertionFailed  ResourseNotFoundException | Custom Exception class |
| Repo | UserRepository  InterviewRepository | Repository interface |
| Service | UserService(Interface),  UserServiceImpl  (Implementation class)  InterviewService(Interface),  InterviewServiceImpl  (Implementation class) | Service Layer |
| Utils | UserUtilities  InterviewUtilities  AttendeeUtilities | Class containing resources for inter-conversion between model and dto |

**Note:**

1. **Need to implement only service methods in UserServiceImpl.java , InterviewServiceImpl class and UserController.java and InterviewController class**
2. **Do not add any method or change rest end point pattern.**