

<b>Project Code:</b>	ITS-01
<b>Project Name:</b>	Interview Tracking System

## Revision History

Version (x.yy)	Date of Revision	Description of Change	Reason for Change	Affected Sections	Approved By
1.0	Apr-2013	Initial Draft			
2.10	Sept-2013	Revision	Mapping with CPC Tool		
2.20	Nov-2013	Revision	Aligning with UCF		
2.2.1	Jun-2014	Revision			

## Affected Groups

Development Engineering
Quality Assurance
XYZ Solutions

## List of Reference Documents

Name	Version No.
1.RS_ITS	2.20
2.FS_ITS	2.20
3.	
4.	

Prepared by/Date

Reviewed by/Date

Approved by/Date

## Table of Contents

<b>TABLE OF CONTENTS .....</b>	<b>2</b>
<b>1. INTRODUCTION .....</b>	<b>3</b>
1.1 Background.....	3
1.2 Purpose.....	3
1.3 Scope.....	3
<b>2. GLOBAL DATA STRUCTURES AND SHARED DATA FUNCTIONS.....</b>	<b>4</b>
<b>3. HIGH LEVEL DESIGN.....</b>	<b>5</b>
3.1 Use Case Diagrams.....	5
3.1.1 Use Case Diagram for Admin .....	5
3.1.2 Use Case Diagram for Tech Panel.....	6
3.1.3 Use Case Diagram for HR Panel.....	6
3.2 Use case Definition.....	6
3.3 Class Diagram .....	8
3.4 Sequence Diagram .....	8
TechPanel viewing final results declared by admin. ....	9
3.5 Packages / Classes / Interface .....	10
3.6 UI Templates.....	14
<b>4. CRITICAL FUNCTIONS AND FOCUS FOR TESTING .....</b>	<b>17</b>
<b>5. LIMITATIONS .....</b>	<b>17</b>
<b>6. APPENDIX.....</b>	<b>17</b>
1. Table: ITS_TBL_User_Credentials.....	17
2. Table: ITS_TBL_User_Profile.....	17
3. Table: ITS_TBL_Candidate .....	17
4. Table: ITS_TBL_Interview_Schedule .....	18
5. Table: ITS_TBL_TechPanel .....	18
6. Table: ITS_TBL_HRPanel .....	19
Database Sequences.....	19

## 1. Introduction

### 1.1 Background

XYZ Solutions provides automation of scheduling interviews for candidates.

### 1.2 Purpose

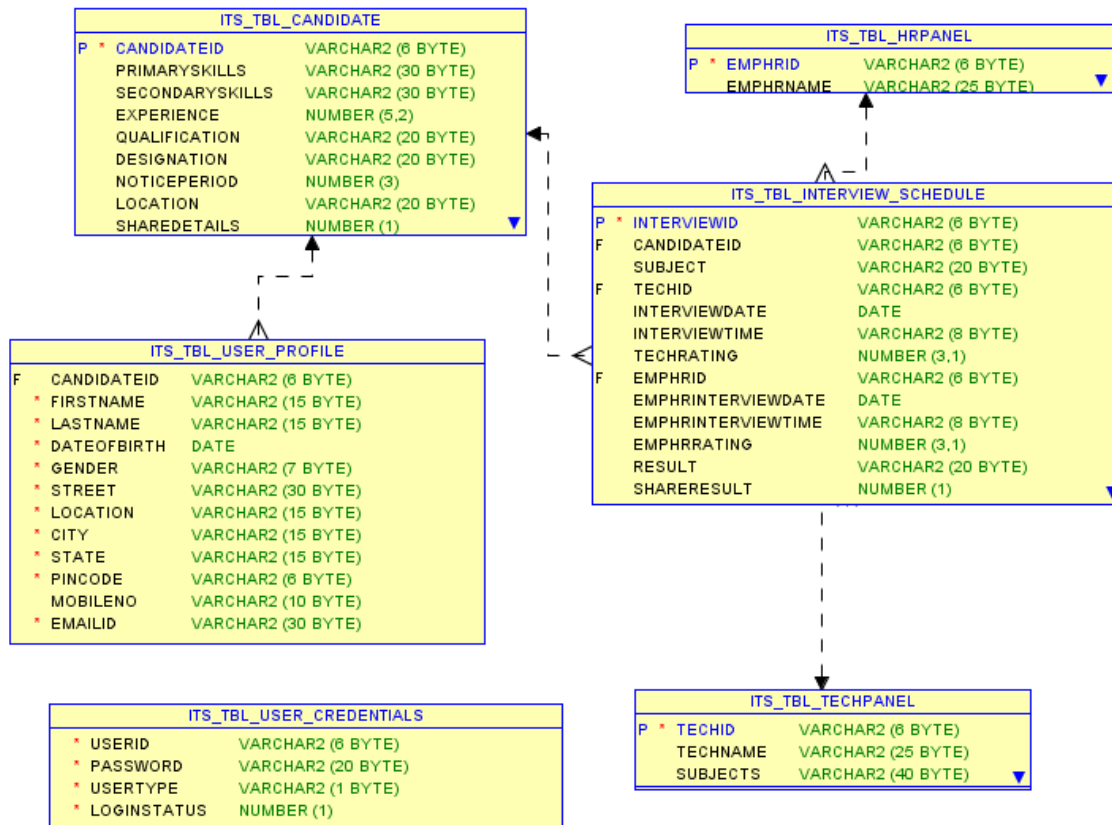
The interview and hiring process in the company is quite huge and involves lot of people. So a centralized system is required to keep track of the interview process. Hence, XYZ Solutions plans to develop “Interview Tracking System” - a standalone/Web application [Core Java Batches - Swing Application; J2EE Batches - Web Application] which will ensure that the interview process workflow is followed and the candidate's status of the interview is always available.

### 1.3 Scope

The scope of the Interview Tracking System (ITS) will be to provide the functionality as described below. The system will be developed on a Windows operating system using Java/J2EE.

## 2. Global Data Structures and Shared Data Functions

This section describes the structure of 9 tables to be used for the implementation of requirements as stated in the specification.



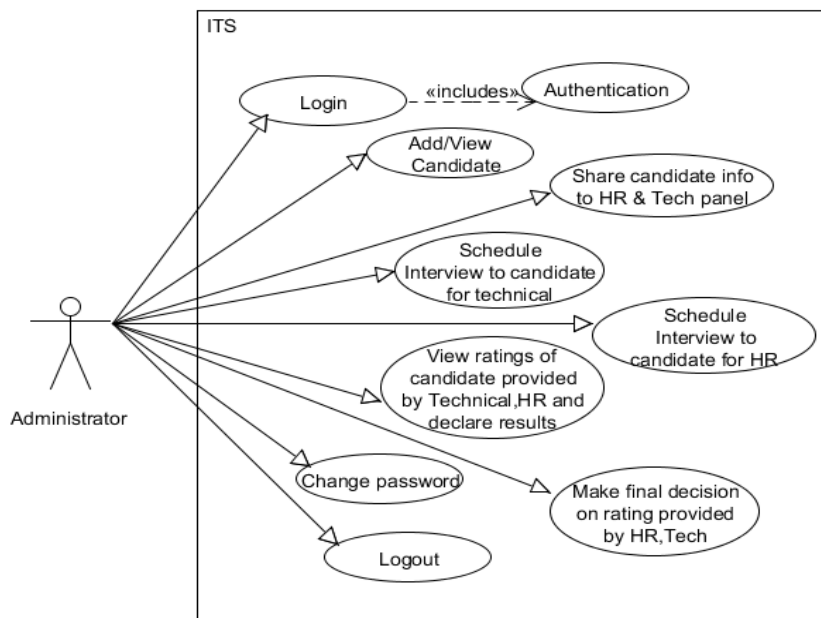
### 3. High Level Design

This section describes the high level design diagrams. Use case diagram with Use Case definition, Sequence Diagram and Class Diagram which provides a visual representation of the requirements, logical flow and their class representations.

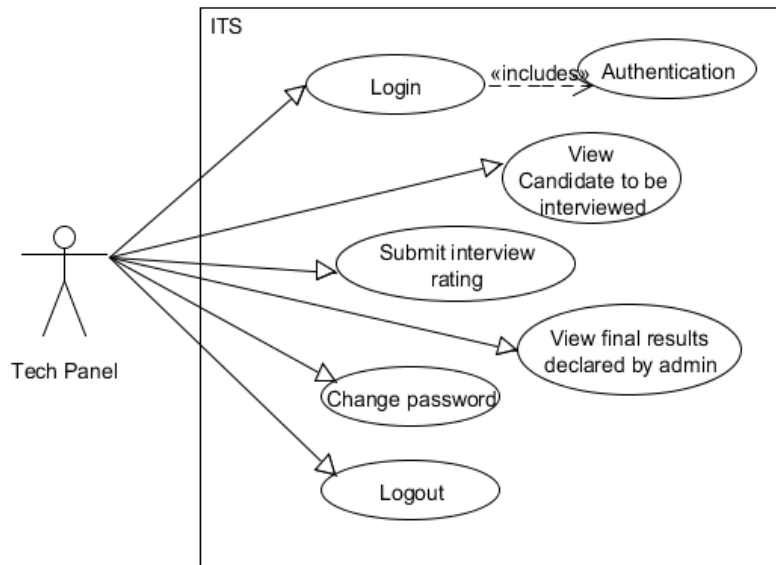
#### 3.1 Use Case Diagrams

The requirements of a system can be represented using a use case model in the Use Case Diagram. The use case diagram for the actors of this case study is given as below.

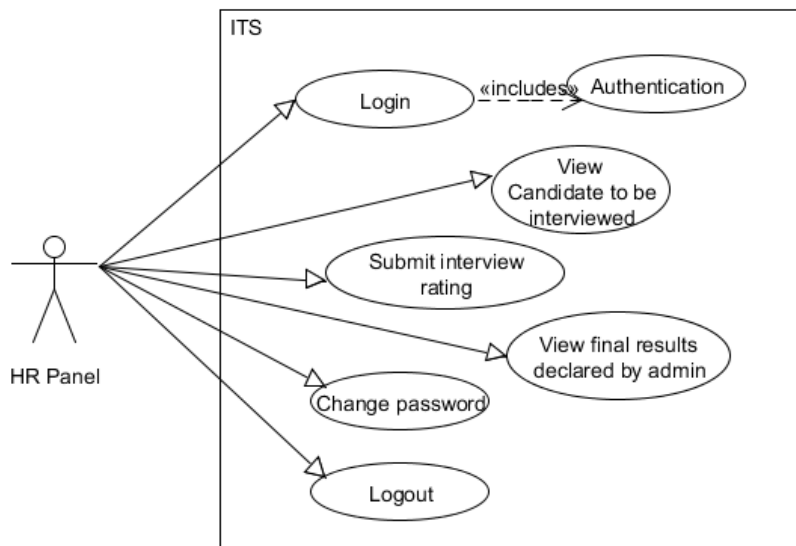
##### 3.1.1 Use Case Diagram for Admin



### 3.1.2 Use Case Diagram for Tech Panel



### 3.1.3 Use Case Diagram for HR Panel



## 3.2 Use case Definition

Generally, in a design document, Use case definitions should be written for all the *Requirements* of the system.

Note: Participants are expected to document use case definitions for all requirements. However, for few requirements documented below for reference.

Below table explains 'Use Case' definition for requirement "AA-001" - Login operation for all users.

### 3.2.1 Login

<b>USE CASE #</b>	AA-001 <i>Login</i>	
<b>Goal</b>	<i>All users logging into the system should be authenticated using a unique login-id and password (operations to be supported based on type of user)</i>	
<b>Preconditions</b>	If the user type is 'Admin', credential details should exist.	
<b>Success End Condition</b>	If the user type is 'Admin', then redirect to the Admin page. If the user type is 'Tech', then redirect to the Tech panel page. If the user type is 'HR', then redirect to the HR panel page.	
<b>Failed End Condition</b>	<i>The end user is redirected to an Error Page, and/or is asked to re-enter login credentials.</i>	
<b>Primary, Secondary Actors</b>	Admin, Tech, HR.	
<b>Trigger</b>	<i>Login button</i>	
<b>DESCRIPTION</b>	<b>Step</b>	<b>Action</b>
	1	<i>Enter Login credentials (id &amp; password)</i>
	2	<i>Click on Login button</i>
	3	<i>If id &amp; password is Success, then identify user type Display appropriate(Admin/Tech/HR) home page</i>
	<b>Step</b>	<b>Branching Action</b>
	1	<i>If 'id' is not existing then return with requesting for registration</i>
	2	<i>If password is not matching return with suitable error message say 'Re-enter id &amp; password'</i>
<b>Related Information/Use cases</b>	<i>Not Applicable</i>	
<b>Priority</b>	<i>P1</i>	
<b>Performance</b>	<i>5 seconds</i>	
<b>Frequency</b>	<i>10 / hour</i>	
<b>Assumptions</b>	Admin/Tech/HR login credentials are available in the Database and others are already registered with their credentials	

Below table explains 'Use Case' definition for requirement "AD-001" – ADD Candidate Details operation for Admin user only.

### 3.2.2 ADD Candidate Details

<b>USE CASE #</b>	AD-001 <i>Add Candidate details</i>	
<b>Goal</b>	<i>To enable Administrator to create and add new Candidate</i>	
<b>Preconditions</b>	Administrator must be logged in to be able to create a new Candidate.	
<b>Success End Condition</b>	<i>"Redirect to Admin home page"</i>	
<b>Failed End Condition</b>	<i>"Redirect to Error Page"</i>	
<b>Primary, Secondary Actors</b>	<i>Administrator</i>	
<b>Trigger</b>	<i>'Add Candidate button</i>	
<b>DESCRIPTION</b>	<b>Step</b>	<b>Action</b>
	1	<i>Provide appropriate Candidate details</i>
	2	<i>Click on Add Candidate button</i>
	<b>Step</b>	<b>Branching Action</b>
	1	<i>If failed to add Candidate details</i>
	2	<i>Display appropriate message to the admin</i>
<b>Related Information/Use cases</b>	<i>Not applicable</i>	
<b>Priority</b>	<i>P1</i>	
<b>Performance</b>	<i>Approx. 4 sec</i>	
<b>Frequency</b>	<i>10 / day</i>	
<b>Assumptions</b>	<i>Admin login credentials are available in the Database</i>	

### 3.3 Class Diagram

The class diagram is a very basic concept in object-oriented world. Class diagrams demonstrate a model, describing what attributes and behavior it has rather than describing the methods for accomplishing operations. Class diagrams are very useful in representing relationships between classes and interfaces.

<class diagram to be drawn by RLL participants>

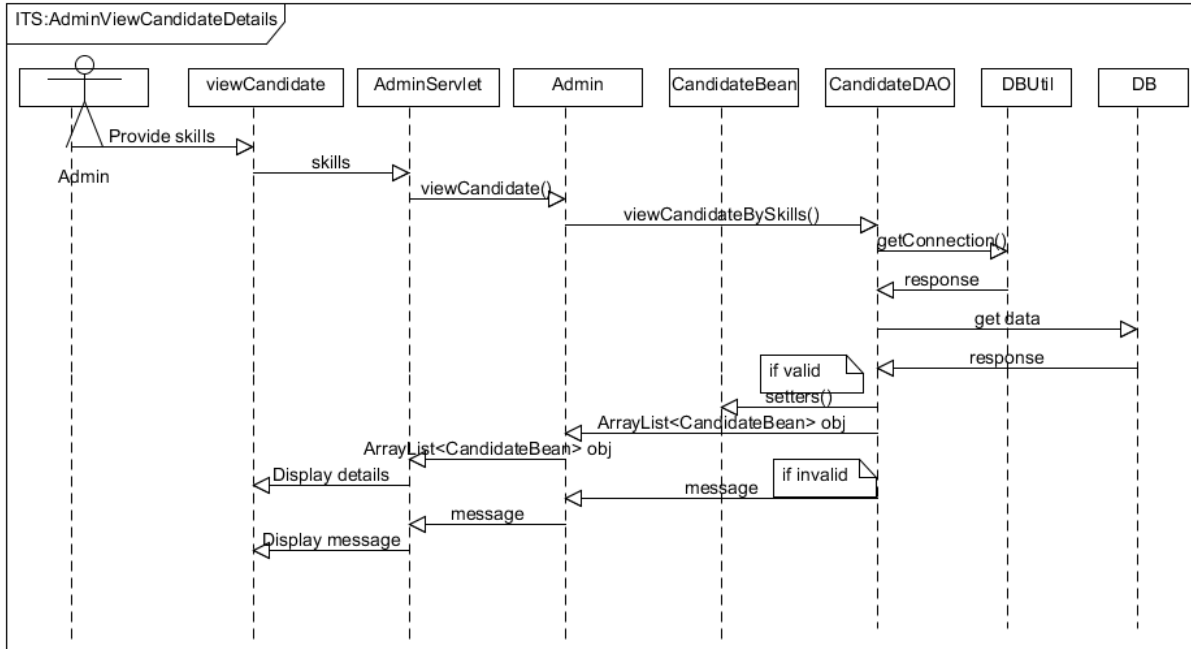
### 3.4 Sequence Diagram

A graphical representation of a module's function invoking functions of other modules in order to achieve a task (specific user requirement) is called a sequence diagram. A sequence diagram for the authentication process is given below for reference. The below example is for a Web Application using servlets/jsp.



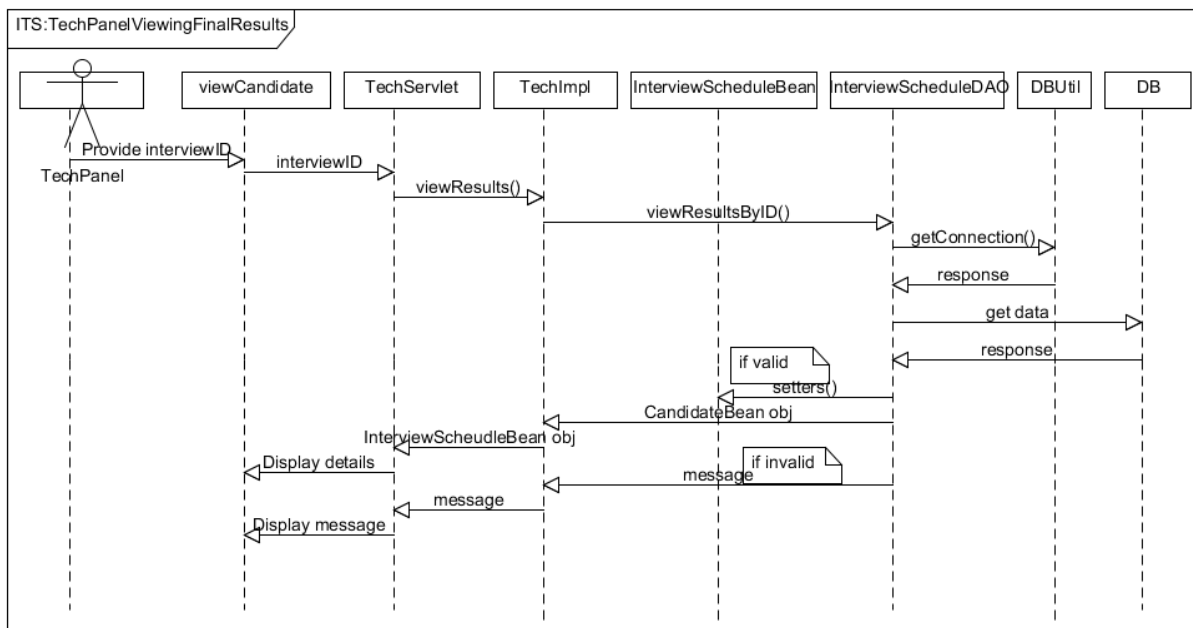
### 3.4.1 View candidate details:

Admin viewing candidate details.



### 3.4.2 View Final results:

TechPanel viewing final results declared by admin.



### 3.5 Packages / Classes / Interface

This section provides a brief outlook on the packaging hierarchy along with the respective classes to be used for the implementation.

The 4 packages mentioned below are for both GUI and Web Application.

Packages	
Package	Description
<b>com.wipro.its.service</b>	This package contains all the Service classes
<b>com.wipro.its.bean</b>	This package contains all the bean classes
<b>com.wipro.its.dao</b>	This package contains all the DAO functionality classes
<b>com.wipro.its.util</b>	This package contains all the generic functionality classes

This package is used only for a GUI application.

<b>com.wipro. its.ui</b>	This package contains all the UI related classes [ For Core Java ]
--------------------------	--

The package for the controller class should be used as below based on the type of application

<b>com.wipro.its.listener</b>	<b>listener</b> - core java
or	
<b>com.wipro.its.servlet</b>	<b>servlet</b> - Web Applications
or	
<b>com.wirpo.its.action</b>	<b>action</b> - Struts
or	
<b>com.wipro.its.controller</b>	<b>controller</b> - Spring

## Package com.wipro.its.bean

Class Name	Attributes	Data Type
<b>ProfileBean</b>	candidateID	String
	firstName	String
	lastName	String
	dateOfBirth	Date
	gender	String
	street	String
	location	String
	city	String
	state	String
	pincode	String
	mobileNo	String
	emailID	String
	password	String

<b>CredentialsBean</b>	userID	String
	password	String
	userType	String
	loginStatus	int

<b>CandidateBean</b>	candidateID	String
	primarySkills	String
	secondarySkills	String
	experience	float
	qualification	String
	designation	String
	noticePeriod	int
	location	String
	shareDetails	int

<b>InterviewScheduleBean</b>	interviewID	String
	candidateID	String
	Subject	String
	techID	String
	interviewDate	Date
	interviewTime	String
	techRating	float

	empHRID	String
	empHRInterviewDate	Date
	empHRInterviewTime	String
	empHRRating	float
	result	String
	shareResult	int

<b>TechPanelBean</b>	techID	String
	techName	String
	subjects	String

<b>HRPanelBean</b>	empHRID	String
	empHRName	String

## Package com.wipro.its.service

Interface Summary	
Interface	Description
<b>Administrator</b>	<p>Entity interface for Administrator dealing with the admin process functionalities</p> <div> <b>Method Summary</b>  String <b>addCandidate</b>(ProfileBean profileBean, CandidateBean candidateBean)  ArrayList&lt;CandidateBean&gt; <b>viewCandidate</b>(String skills, float experience, String qualification)  boolean <b>shareCandidateDetails</b>(String candidateID, String panel)  String <b>scheduleInterviewForTech</b>(InterviewScheduleBean interviewBean)  boolean <b>scheduleInterviewForHR</b>(InterviewScheduleBean interviewBean)  InterviewScheduleBean <b>viewRatings</b>(String interviewID)  boolean <b>finalizeResultDecision</b>(String interviewID, String result)  boolean <b>declareResults</b>(String interviewID, int share) </div>
<b>TechPanel</b>	<p>Entity interface of TechPanel for dealing with the his/her process functionalities</p> <div> <b>Method Summary</b>  ArrayList&lt;CandidateBean&gt; <b>viewCandidates</b>(String techID, Date interviewDate)  boolean <b>submitRatings</b>(String interviewID, String techID, float rating)  InterviewScheduleBean <b>viewFinalResults</b>(String interviewID) </div>

<b>HRPanel</b>	Entity interface of HRPanel for dealing with the HR process functionalities
<b>Method Summary</b>	
ArrayList<CandidateBean> <b>viewCandidates</b> (String empHRID, Date empHRInterviewDate)	
boolean <b>submitRatings</b> (String interviewID, String empHRID, float rating)	
InterviewScheduleBean <b>viewFinalResults</b> (String interviewID)	

### Package com.wipro.its.dao

Find below the suggestive approach for CRUD operations [method naming & signature] for the DAO Interface/classes. Create the necessary DAO classes.

Interface Name	Description
<b>xyzDAO</b>	DAO interface/class to deal with operations related to the specific table.
<b>Method Summary</b>	
String createXYZ(BeanoObject)	
int deleteXYZ(ArrayList<String> )	
boolean updateXYZ(BeanoObject)	
BeanoObject findById(String)	
ArrayList<BeanoObject> findAll()	

- If required, additional find methods can be created.

### Package com.wipro.its.util

Interface Summary	
Interface	Description
<b>Authentication</b>	This interface is responsible for performing the Authentication and Authorization process.
<b>Methods</b>	
boolean <b>authenticate</b> ( <b>CredentialsBean</b> credentialsBean)	
String <b>authorize</b> (String userID)	
boolean <b>changeLoginStatus</b> ( <b>CredentialsBean</b> credentialsBean, int loginStatus)	
<b>DBUtil</b>	This class is responsible for the Database connection establishment.
<b>Methods</b>	
static Connection <b>getDBConnection</b> (String driverType)	
<b>User</b>	Interface for handling different types of users
<b>Method Summary</b>	

	String <b>login(CredentialsBean credentialsBean)</b> Return value must be either: "A", "T", "H", "FAIL", "INVALID" A->Admin, T->TechPanel, H->HRPanel <b>Wrong username/password should return INVALID.</b>
	boolean <b>logout(String userId)</b>
	String <b>changePassword(CredentialsBean credentialsBean, String newPassword)</b> Return value must be either: "SUCCESS", "FAIL", "INVALID"

Note: **Wherever empty or NULL is the response in all such cases suitable message has to be displayed for user**

### 3.6 UI Templates

#### 3.6.1 UI Principle

The UI [Presentation Layer] should be designed with the below mentioned principles which helps easy interaction by the user to the application.

#### 3.6.2 UI controls and Usage Principle

UI Type	Controls	Description
Direct Entry	Text Box, Text Area	Any input that cannot be predicted and needs the user to key in. e.g Name, Address, contact no etc.
Static Selection	Option Button, Check Box, Drop Down	Should be used where the input can be predefined. e.g gender, month [ Jan – Dec ] etc. If number of items is more, drop down is preferred.
Dynamic Selection	Drop Down	The items for the drop down should be retrieved from a stored Data. e.g Displaying Districts in a drop down from places table.
Automation	Label Text Field [Read Only]	Data's that are calculative or an output of a function. e.g : Displaying system date, showing total amount etc.
Decision Control	Button	Operations like submit, save, clear should be executed only upon clicking respective buttons.

#### 3.6.3 UI Template

This section contains the design template for the website home page [Fig. 1] that will be displayed at the time of opening this web application and Actor specific home page [Fig. 2].

<logo>	< Project Title >
About Us    Contact Us	

<p>&lt; General Info &gt;</p>	<div data-bbox="915 306 1354 642"><p>Login</p><div><input type="text" value="Username"/></div><div><input type="password" value="Password"/></div><div><input type="checkbox"/> Remember me on this computer <input type="button" value="Login"/></div></div> <p>Forgot your password? <a href="#">Click here to reset it.</a></p> <p><a href="#">New User?</a> Register Here</p>
<p>Copyright @ 2014 Wipro Technologies. All rights reserved</p>	

**Fig. 1** - Main Page [ First Page to open ]

<logo>	< Project Title >	
< Logged in Name >	Home	Logout
<Navigation Links>	< Page based on the navigation link selected>	
<Navigation Links>		
<Navigation Links>		
<Navigation Links>		
<Navigation Links>		
<Navigation Links>		
Copyright @ 2014 Wipro Technologies. All rights reserved		

**Fig. 2** - Home Page for Actor

<logo>	< Project Title >						
< Logged in Name >	Home	Logout					
< Title for the View Screen >							
<Col Head>	<Col Head>	<Col Head>	<Col Head>	<Col Head>	<Col Head>		
						<a href="#">Edit</a>	<a href="#">Delete</a>
						<a href="#">Edit</a>	<a href="#">Delete</a>
						<a href="#">Edit</a>	<a href="#">Delete</a>
						<a href="#">Edit</a>	<a href="#">Delete</a>
						<a href="#">Edit</a>	<a href="#">Delete</a>
						<a href="#">Edit</a>	<a href="#">Delete</a>
						<a href="#">Edit</a>	<a href="#">Delete</a>
Copyright © 2014 Wipro Technologies. All rights reserved							

**Fig. 3** – View Screen with Edit and Delete Functionality



## 4. Critical Functions and Focus for Testing

login(), scheduleInterview(), addCandidate(), provideRating().

## 5. Limitations

- Adding Candidate details will be performed by admin, but not candidate.
- Candidate doesn't have login or to view any kind of information.

## 6. APPENDIX

### 1. Table: ITS\_TBL\_User\_Credentials

This table contains Authentication Information for Administrator, Tech Panel & HR Panel.

Field Name	Data Type	Description
UserId	VARCHAR2(6)	Primary Key
Password	VARCHAR2(20)	Not Null
Usertype	VARCHAR2(1)	Either ['A','T','H']
Loginstatus	NUMBER(1)	Either [1 ,0]

### 2. Table: ITS\_TBL\_User\_Profile

This table contains Candidate profile details.

Field Name	Data Type	Description
CandidateId*	VARCHAR2(6)	Foreign Key
Firstname	VARCHAR2(15)	Not Null
Lastname	VARCHAR2(15)	Not Null
Dateofbirth	DATE	Not Null
Gender	VARCHAR2(7)	Not Null
Street	VARCHAR2(30)	Not Null
Location	VARCHAR2(15)	Not Null
City	VARCHAR2(15)	Not Null
State	VARCHAR2(15)	Not Null
Pincode	VARCHAR2(6)	Not Null
MobileNo	VARCHAR(10)	Exact 10 digit only
EmailId	VARCHAR2(30)	

\* First 2 letters of First Name followed by 4 digits auto generated number

### 3. Table: ITS\_TBL\_Candidate

This table contains Candidate specific information.

Field Name	Data Type	Description
candidateId*	VARCHAR2(6)	Primary Key

PrimarySkills	VARCHAR2(30)	Not Null
SecondarySkills	VARCHAR2(30)	Not Null
Experience	NUMBER(2,2)	Not Null
Qualification	VARCHAR2(20)	Not Null
Designation	VARCHAR2(20)	
NoticePeriod	NUMBER(3)	
Location	VARCHAR2(20)	
ShareDetails	NUMBER(1)	0->None, 1->Tech, 2->HR, 3->All

\* First 2 letters of First Name followed by 4 digits auto generated number

#### 4. Table: ITS\_TBL\_Interview\_Schedule

This table contains Interview schedule details.

Field Name	Data Type	Description
InterviewID*	VARCHAR2(6)	Primary Key
CandidateID	VARCHAR2(6)	Foreign Key
Subject	VARCHAR2(20)	Not Null
TechID	VARCHAR2(6)	Foreign Key
InterviewDate	DATE	Not Null
InterviewTime	VARCHAR2(8)	Not Null
TechRating	NUMBER(1,1)	Rating->1.0 to 5.0 1.0->Min, 5.0->Max
empHRID	VARCHAR2(6)	Foreign Key
empHRInterviewDate	DATE	
empHRInterviewTime	VARCHAR2(8)	
empHRRating	NUMBER(1,1)	Rating->1.0 to 5.0 1.0->Min, 5.0->Max
Result	VARCHAR2(20)	
ShareResult	NUMBER(1)	

\* Id should be First 2 letters of Subject followed by 4 digits auto generated number

#### 5. Table: ITS\_TBL\_TechPanel

This table contains details about tech panel.

Field Name	Data Type	Description
techId*	VARCHAR2 (6)	Primary Key
techName	VARCHAR2(25)	Not Null
Subjects	VARCHAR2(40)	Not Null

**6. Table: ITS\_TBL\_HRPanel**

This table contains details about HR panel.

Field Name	Data Type	Description
empHRId	VARCHAR2 (6)	Primary Key
empHRName	VARCHAR2(25)	Not Null

**Database Sequences**

Sequence Name	Purpose	Start With
its_seq_candidateID	Candidate ID	1000
its_seq_interviewID	Interview ID	1000