**Use Case Scenarios**

Employee

|  |  |
| --- | --- |
| *Scenario name* | EmployeeLogin |
| *Participating actor*  *instances* | Bob:Employee |
| *Flow of events* | 1. Bob wants to login in the system. Bob navigates to the GoalManager Main Page, and presses the login button.      1. GoalManager opens a form that Bob can fill with his credentials. 2. Bob fills the form and presses the log in button. 3. Goal Manager takes submitted form and performs form validation. GoalManager attempts to validate Bob’s credentials by hashing the provided password, querying its database for an entry matching Bob’s username, and comparing the associated hash value with the hashed password submitted by Bob. GoalManager serves the default webpage for the Main Menu for validated Employees to Bob’s browser. 4. Bob is logged in and able to perform his tasks. |

|  |  |
| --- | --- |
| *Scenario name* | EmployeeInvalidLogin |
| *Participating actor*  *instances* | Bob:Employee |
| *Flow of events* | 1. Bob wants to login in the system. Bob navigates to the GoalManager Main Page, and presses the login button.      1. GoalManager opens a form that Bob can fill with his credentials. 2. Bob fills the form and presses the log in button. 3. Goal Manager takes submitted form and performs form validation. GoalManager attempts to validate Bob’s credentials by hashing the provided password, querying its database for an entry matching Bob’s username, and comparing the associated hash value with the hashed password submitted by Bob. GoalManager the prompts an error message for an invalid login. 4. Bob may attempt to log in again. |

|  |  |
| --- | --- |
| *Scenario name* | EmployeeViewGoal |
| *Participating actor*  *instances* | Bob:Employee |
| *Flow of events* | 1. Bob wants to check on his progress towards his quarterly and yearly goals. Bob then navigates to the GoalManager Main page that is authenticated for Employee’s. Bob presses the ViewGoal button      1. GoalManager queries a list of goals associated with Bob and prompts the list of goals associated with Bob. 2. Bob then selects a goal. 3. GoalManager then shows Bob the goal in an extended view page. 4. Bob is able to view the personal goals he set for himself or associated with his department. |

|  |  |
| --- | --- |
| *Scenario name* | EmployeeCreateGoal |
| *Participating actor*  *instances* | Bob:Employee |
| *Flow of events* | 1. Bob wants to set a new goal for himself. Bob then navigates to the GoalManager Main page that is authenticated for Employees. Bob presses the CreateGoal Button.      1. GoalManager prompts a form for Bob to fill out details about a Goal he would like to create. 2. Bob fills out the form and presses the submit button. 3. GoalManager performs form validation on entries. Bob’s submission is valid, and GoalManager writes new a new goal to the database. GoalManager returns Bob back to the Main Menu authenticated for Employees. |

|  |  |
| --- | --- |
| *Scenario name* | EmployeeCreateGoalFailure |
| *Participating actor*  *instances* | Bob:Employee |
| *Flow of events* | 1. Bob wants to set a new goal for himself. Bob then navigates to the GoalManager Main page that is authenticated for Employees. Bob presses the CreateGoal Button.      1. GoalManager prompts a form for Bob to fill out details about a goal he would like to create. 2. Bob fills out the form and presses the Submit button. 3. GoalManager performs form validation on entries. Bob’s submission is invalid, GoalManager then prompts an error message for an invalid validation form. 4. Bob may attempt to submit the form again. |

|  |  |
| --- | --- |
| *Scenario name* | EmployeeUpdateGoal |
| *Participating actor*  *instances* | Bob:Employee |
| *Flow of events* | 1. Bob wants to Update a goal on his progress. Bob then navigates to the GoalManager Main page that is authenticated for Employees. Bob presses the Update Goal button.      1. GoalManager queries a list of goals associated with Bob and prompts the list of goals associated with Bob. 2. Bob then selects a goal. 3. GoalManager then prompts Bob a form to fill out details about his update. 4. Bob fills out the form and presses the Update button. 5. Goal Manager adds the Update to the goal and updates the progress. Goal Manager prompts the Main Menu Authenticated for Employees. 6. Bob has Updated his goal. |

|  |  |
| --- | --- |
| *Scenario name* | EmployeeEditGoal |
| *Participating actor*  *instances* | bob:Employee |
| *Flow of events* | 1. Bob made a mistake in the wording of a personal goal and wants to edit the goal details. He logs into GoalManager with valid credentials and activates the “Goal Progress Report” page.      1. GoalManager queries goal data from database and presents it to Bob through the webpage. 2. Bob selects one of his personal goals to edit from the “Goal Progress Report” webpage. 3. GoalManager serves dynamic content on the “Goal Progress Report” webpage with a form populated with the goal’s current text. 4. Bob changes the goal’s content valid data into the “Goal Progress Form” dynamic webpage and presses “Submit”. 5. GoalManager performs form validation on Bob’s submitted form data. Passing that, GoalManager writes edited goal information for the selected goal to its database. GoalManager returns Bob back to the default Employee webpage. |

|  |  |
| --- | --- |
| *Scenario name* | EmployeeEditGoalFailure |
| *Participating actor*  *instances* | bob:Employee |
| *Flow of events* | 1. Bob made a mistake in the wording of a personal goal and wants to edit the goal details. He logs into GoalManager with valid credentials and activates the “Goal Progress Report” function.      1. GoalManager queries goal data from database and presents it to Bob through the webpage. 2. Bob selects one of his personal goals to edit from the “Goal Progress Report” webpage. 3. GoalManager serves dynamic content on the “Goal Progress Report” webpage with a form populated with the goal’s current text. 4. Bob changes the goal’s content valid data into the “Goal Progress Form” dynamic webpage and presses “Submit”. Unfortunately, Bob submits an invalid edit, such as emptying the entire goal description. 5. GoalManager performs form validation on Bob’s submitted form data. Failing that, GoalManager returns Bob back to the default Employee webpage with an error describing the edit failure. |

|  |  |
| --- | --- |
| *Scenario name* | EmployeeLogout |
| *Participating actor*  *instances* | Bob:Employee |
| *Flow of events* | 1. Bob wants to logout of the system. Bob is already logged in and navigates to the Goal Manager Main Menu authenticated for Employees. Bob presses the log out button.      1. GoalManager then logs out of the system and prompts the GoalManager Main Menu for unauthenticated users. 2. Bob is now logged out |

Supervisor Scenarios

|  |  |
| --- | --- |
| *Scenario name* | SupervisorLogin |
| *Participating actor*  *instances* | Steve:Supervisor |
| *Flow of events* | 1. Steve wants to login in the system. Steve navigates to the GoalManager Main Page, and presses the login button.      1. GoalManager opens a form that Steve can fill with his credentials. 2. Steve fills the form and presses the log in button. 3. Goal Manager takes submitted form and performs form validation. GoalManager attempts to validate Steeve’s credentials by hashing the provided password, querying its database for an entry matching Steve’s username, and comparing the associated hash value with the hashed password submitted by Bob. GoalManager serves the default webpage for the Main Menu for validated Employees to Steve’s browser. 4. Steve is logged in and able to perform his tasks. |

|  |  |
| --- | --- |
| *Scenario name* | SupervisorInvalidLogin |
| *Participating actor*  *instances* | Steve:Supervisor |
| *Flow of events* | 1. Steve wants to login in the system. Steve navigates to the GoalManager Main Page, and presses the login button.      1. GoalManager opens a form that Steve can fill with his credentials. 2. Steve fills the form and presses the log in button. 3. Goal Manager takes submitted form and performs form validation. GoalManager attempts to validate Steve’s credentials by hashing the provided password, querying its database for an entry matching Steve’s username, and comparing the associated hash value with the hashed password submitted by Steve. GoalManager the prompts an error message for an invalid login. 4. Steve may attempt to log in again. |

|  |  |
| --- | --- |
| *Scenario name* | SupervisorViewGoals |
| *Participating actor*  *instances* | steve:Supervisor |
| *Flow of events* | 1. Steve wants to check on his progress towards his quarterly and yearly goals. Steve then navigates to the GoalManager Main page that is authenticated for Employee’s. Steve presses the ViewGoal button      1. GoalManager queries a list of goals associated with Steve and prompts the list of goals associated with Steve. 2. Steve then selects a goal. 3. GoalManager then shows Steve the goal in an extended view page. 4. Steve is able to view the personal goals he set for himself or associated with his department. |

|  |  |
| --- | --- |
| *Scenario name* | SupervisorCreateGoal |
| *Participating actor*  *instances* | steve:Supervisor |
| *Flow of events* | 1. Steve wants to set a new goal for himself or his department. then navigates to the GoalManager Main page that is authenticated for Employees. Steve presses the CreateGoal Button.      1. GoalManager prompts a form for Steve to fill out details about a Goal he would like to create. 2. Steve fills out the form and presses the submit button. 3. GoalManager performs form validation on entries. Steve’s submission is valid, and GoalManager writes new a new goal to the database. GoalManager returns Steve back to the Main Menu authenticated for Supervisor. |

|  |  |
| --- | --- |
| *Scenario name* | SupervisorCreateGoalFailure |
| *Participating actor*  *instances* | Steve:Supervisor |
| *Flow of events* | 1. Steve wants to set a new goal for himself. Steve then navigates to the GoalManager Main page that is authenticated for Employees. Steve presses the CreateGoal Button.      1. GoalManager prompts a form for Steve to fill out details about a goal he would like to create. 2. Steve fills out the form and presses the Submit button. 3. GoalManager performs form validation on entries. Steve’s submission is invalid, GoalManager then prompts an error message for an invalid validation form. 4. Steve may attempt to submit the form again. |

|  |  |
| --- | --- |
| *Scenario name* | SupervisorUpdateGoal |
| *Participating actor*  *instances* | Steve:Supervisor |
| *Flow of events* | 1. Steve wants to Update a goal for himself or his department on his progress. Steve then navigates to the GoalManager Main page that is authenticated for Employee’s. Steve presses the Update Goal button.      1. GoalManager queries a list of goals associated with Steve and prompts the list of goals associated with Steve. 2. Steve then selects a goal. 3. GoalManager then prompts Steve a form to fill out details about his update. 4. Steve fills out the form and presses the Update button. 5. Goal Manager adds the Update to the goal and updates the progress. Goal Manager prompts the Main Menu Authenticated for Employees. 6. Steve has Updated his goal. |

|  |  |
| --- | --- |
| *Scenario name* | SupervisorViewReport |
| *Participating actor*  *instances* | steve:Supervisor |
| *Flow of events* | 1. Steve wants to view Employee records, Bob then navigates to the GoalManager Main Menu, and logs in with his credentials. Presses the View Report Button.      1. GoalManager the queries a list of Employees associated with Steve’s department to select from. 2. Steve then selects an Employee. 3. GoalManager then prompts a page detailing all goals, progress, and updates. 4. Steve presses the Done Button. 5. GoalManager then prompts the Main Menu authenticated for Supervisors. |

|  |  |
| --- | --- |
| *Scenario name* | SupervisorLogout |
| *Participating actor*  *instances* | Steve:Supervisor |
| *Flow of events* | 1. Steve wants to logout of the system. Steve is already logged in and navigates to the Goal Manager Main Menu authenticated for Employees. Steve presses the log out button.      1. GoalManager then logs out of the system and prompts the GoalManager Main Menu for unauthenticated users. 2. Steve is now logged out |

Administrator Scenarios

|  |  |
| --- | --- |
| *Scenario name* | AdministratorLogin |
| *Participating actor*  *Instances* | John:Administrator |
| *Flow of events* | 1. John wants to login in the system. John navigates to the GoalManager Main Page, and presses the login button.      1. GoalManager opens a form that John can fill with his credentials. 2. John fills the form and presses the log in button. 3. Goal Manager takes submitted form and performs form validation. GoalManager attempts to validate John’s credentials by hashing the provided password, querying its database for an entry matching John’s username, and comparing the associated hash value with the hashed password submitted by John. GoalManager serves the default webpage for the Main Menu for validated Administrator to John’s browser. 4. John is logged in and able to perform his tasks. |

|  |  |
| --- | --- |
| *Scenario name* | AdministratorInvalidLogin |
| *Participating actor*  *Instances* | John:Administrator |
| *Flow of events* | 1. John wants to login in the system. John navigates to the GoalManager Main Page, and presses the login button.      1. GoalManager opens a form that Bob can fill with his credentials. 2. John fills the form and presses the log in button. 3. Goal Manager takes submitted form and performs form validation. GoalManager attempts to validate John’s credentials by hashing the provided password, querying its database for an entry matching John’s username, and comparing the associated hash value with the hashed password submitted by John. GoalManager the prompts an error message for an invalid login. 4. John is able may attempt to log in again. |

|  |  |
| --- | --- |
| *Scenario name* | AdministratorViewGoals |
| *Participating actor*  *instances* | John:Adminstrator |
| *Flow of events* | 1. John wants to check on his progress towards his quarterly and yearly goals. John then navigates to the GoalManager Main page that is authenticated for Employee’s. John presses the ViewGoal button      1. GoalManager queries a list of goals associated with John and prompts the list of goals associated with John. 2. John then selects a goal. 3. GoalManager then shows John the goal in an extended view page. 4. John is able to view the personal goals he set for himself or associated with his department. |

|  |  |
| --- | --- |
| *Scenario name* | AdministratorCreateGoal |
| *Participating actor*  *instances* | John:Adminstrator |
| *Flow of events* | 1. John wants to set a new goal for an Employee or a Supervisor. Bob then navigates to the GoalManager Main page that is authenticated for Employees. Bob presses the CreateGoal Button.      1. GoalManager prompts a form for John to fill out details about a Goal he would like to create. 2. John fills out the form and presses the submit button. 3. GoalManager performs form validation on entries. John’s submission is valid, and GoalManager writes new a new goal to the database. GoalManager returns John back to the Main Menu authenticated for Adminstrator. |

|  |  |
| --- | --- |
| *Scenario name* | AdministratorSetGoalFailure |
| *Participating actor*  *instances* | John:Adminstrator |
| *Flow of events* | 1. John wants to set a new goal for himself. John then navigates to the GoalManager Main page that is authenticated for Employees. Steve presses the CreateGoal Button.      1. GoalManager prompts a form for John to fill out details about a goal he would like to create. 2. John fills out the form and presses the Submit button. 3. GoalManager performs form validation on entries. John’s submission is invalid, GoalManager then prompts an error message for an invalid validation form. 4. John may attempt to submit the form again. |

|  |  |
| --- | --- |
| *Scenario name* | AdminstratorUpdateGoal |
| *Participating actor*  *instances* | John:Administrator |
| *Flow of events* | 1. John wants to Update a goal an Employee or Supervisor’s progress. Steve then navigates to the GoalManager Main page that is authenticated for Employee’s. Steve presses the Update Goal button.      1. GoalManager queries a list of goals associated with the particular Employee or Supervisor and prompts the list of goals. 2. John then selects a goal. 3. GoalManager then prompts John a form to fill out details about his update. 4. John fills out the form and presses the Update button. 5. Goal Manager adds the Update to the goal and updates the progress. Goal Manager prompts the Main Menu Authenticated for Employees. 6. John has Updated his goal. |

|  |  |
| --- | --- |
| *Scenario name* | AdminstratorLogout |
| *Participating actor*  *instances* | John:Adminstrator |
| *Flow of events* | 1. John wants to logout of the system. John is already logged in and navigates to the Goal Manager Main Menu authenticated for Employees. John presses the log out button.      1. GoalManager then logs out of the system and prompts the GoalManager Main Menu for unauthenticated users. 2. John is now logged out |