
Use Cases

for

CSE598 Web App Capstone

Version 1.0 approved

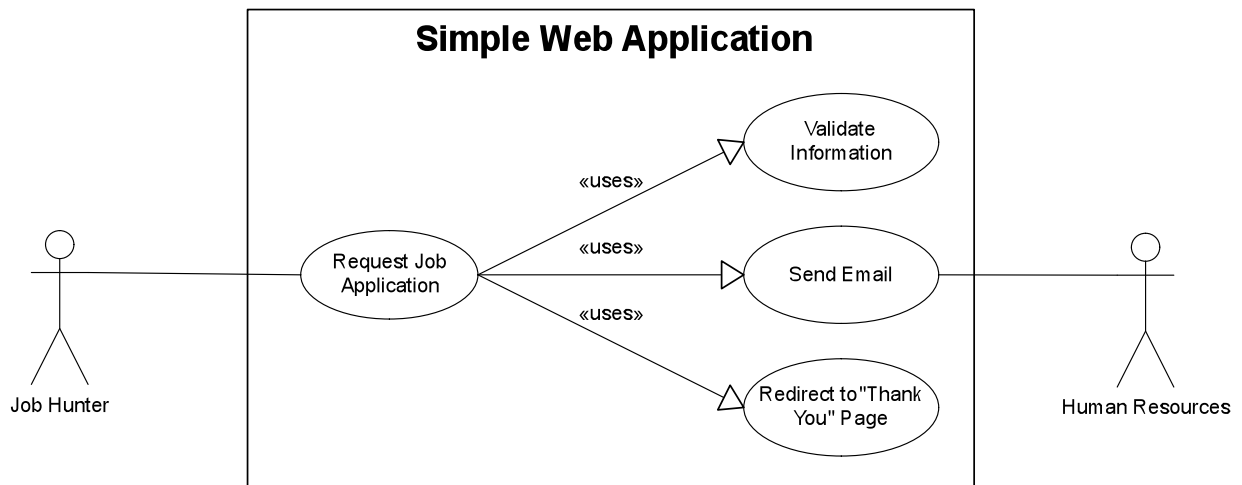
Prepared by Anthony Park, Edward Buskoski, and Todd Ellermann

Created on: 02/08/2007

Revision History

Name	Date	Reason For Changes	Version

Profile 1: Simple Web Application



Use Case 1-A: Request Job Application

Use Case ID:	1-A		
Use Case Name:	Request Job Application		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Job Applicant, Human Resources Department	
Description:	Customers visiting a static website can fill out a form to request a job application. The form will submit through an HTTP POST request to an interactive server side application (web application platform). An email containing the submitted information will be sent to the human resources department who can take additional actions on the information, such as contact the customer directly.	
Trigger:	None	
Preconditions:	1. User is able to use a web browser and navigate to the website.	
Postconditions:	1. Email has been sent to the human resources department. 2. Customer has been redirected to a “thank you” page.	
Normal Flow:	User Activity System Response	
	1. Visit the “Request Job Application” page	• The system returns the web page, which contains a form to collect the user’s information.
	2. Fill out required information	
	3. Click “Submit”	• Validate required information • Send email to Human Resources containing the form field labels and corresponding data • Return “Thank You” page to the user.
	4. View “Thank You” page	
Alternative Flows:	User Activity System Response	
	3. Click “Submit”	• Validate required information • Required information not valid • Return “Request Job Application” page to the user, containing originally submitted values, and an appropriate error message.
	4. Correct required information	
Business Rules:	Required fields:	
	<ul style="list-style-type: none"> • Name – Text, 50 Characters • Email – Text, 50 Characters • Phone Number – Text, 10 Numeric Characters Optional fields: <ul style="list-style-type: none"> • Comments – Text, 255 Characters 	
Special Requirements:		
Assumptions:		

Notes and Issues:	
-------------------	--

Profile 2: Database-Driven, Multi-tier Web Application



Use Case 2-A: Request Job Application

Use Case ID:	2-A		
Use Case Name:	Request Job Application		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Identical to Use Case 1-A from profile 1.

Use Case 2-B: View Active Jobs

Use Case ID:	2-B		
Use Case Name:	View Active Jobs		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Job Applicant									
Description:	Customers visiting a company website can view active jobs for that company.									
Trigger:	None									
Preconditions:	1. User is able to use a web browser and navigate to the website.									
Postconditions:	1. The user is shown a web page containing active jobs.									
Normal Flow:	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">User Activity</th> <th style="text-align: left;">System Response</th> </tr> </thead> <tbody> <tr> <td style="vertical-align: top;">1. Visit the “View Active Jobs” page</td> <td style="vertical-align: top;"> <ul style="list-style-type: none"> The system returns the web page, which contains a list of active jobs. The data will be read real-time from a database. Pagination of greater than 10 jobs is an optional feature. The job listings will display in a list format, and show the following fields: Job Title, Department, Salary, Start Date, and Description. Each job listing will contain a link titled “Request Job Application”, which will take the user to the “Request Job Application” page. </td> </tr> <tr> <td style="vertical-align: top;">2. View Job Listings page</td> <td></td> </tr> <tr> <td style="vertical-align: top;">3. Close web browser</td> <td></td> </tr> </tbody> </table>		User Activity	System Response	1. Visit the “View Active Jobs” page	<ul style="list-style-type: none"> The system returns the web page, which contains a list of active jobs. The data will be read real-time from a database. Pagination of greater than 10 jobs is an optional feature. The job listings will display in a list format, and show the following fields: Job Title, Department, Salary, Start Date, and Description. Each job listing will contain a link titled “Request Job Application”, which will take the user to the “Request Job Application” page. 	2. View Job Listings page		3. Close web browser	
	User Activity	System Response								
	1. Visit the “View Active Jobs” page	<ul style="list-style-type: none"> The system returns the web page, which contains a list of active jobs. The data will be read real-time from a database. Pagination of greater than 10 jobs is an optional feature. The job listings will display in a list format, and show the following fields: Job Title, Department, Salary, Start Date, and Description. Each job listing will contain a link titled “Request Job Application”, which will take the user to the “Request Job Application” page. 								
	2. View Job Listings page									
3. Close web browser										
Alternative Flows:										
Business Rules:										
Special Requirements:										
Assumptions:										
Notes and Issues:										

Use Case 2-C: Create an Account

Use Case ID:	2-C		
Use Case Name:	Create an Account		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Employee														
Description:	Employees of the company need the ability to add/update/delete jobs on the job board. In order to accomplish this, they will need to create a login to the employee section of the website.														
Trigger:	None														
Preconditions:	1. Employee is able to use a web browser and navigate to the website.														
Postconditions:	1. User account is created in the database. 2. Email has been sent to the employee containing the login credentials.														
Normal Flow:	<table border="1"> <thead> <tr> <th>User Activity</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1. Visit the "Employee Login" page</td> <td></td> </tr> <tr> <td>2. Click the "New Account" link</td> <td> <ul style="list-style-type: none"> The system returns the web page, which contains a form to collect the user's information. </td> </tr> <tr> <td>3. Fill out required information</td> <td></td> </tr> <tr> <td>4. Click "Submit"</td> <td> <ul style="list-style-type: none"> Validate required information Create a user account in the database. Send email to Employee's email address containing the login credentials (username and password) Return "Thank You" page to the user containing the Employee's name and login id. </td> </tr> <tr> <td>5. View "Thank You" page</td> <td></td> </tr> <tr> <td>6. Close web browser</td> <td></td> </tr> </tbody> </table>	User Activity	System Response	1. Visit the "Employee Login" page		2. Click the "New Account" link	<ul style="list-style-type: none"> The system returns the web page, which contains a form to collect the user's information. 	3. Fill out required information		4. Click "Submit"	<ul style="list-style-type: none"> Validate required information Create a user account in the database. Send email to Employee's email address containing the login credentials (username and password) Return "Thank You" page to the user containing the Employee's name and login id. 	5. View "Thank You" page		6. Close web browser	
User Activity	System Response														
1. Visit the "Employee Login" page															
2. Click the "New Account" link	<ul style="list-style-type: none"> The system returns the web page, which contains a form to collect the user's information. 														
3. Fill out required information															
4. Click "Submit"	<ul style="list-style-type: none"> Validate required information Create a user account in the database. Send email to Employee's email address containing the login credentials (username and password) Return "Thank You" page to the user containing the Employee's name and login id. 														
5. View "Thank You" page															
6. Close web browser															
Alternative Flow 1:	<table border="1"> <thead> <tr> <th>User Activity</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>4. Click "Submit"</td> <td> <ul style="list-style-type: none"> Validate required information Required information not valid Return "New Account" page to the user, containing originally submitted values, and an appropriate error message. </td> </tr> <tr> <td>5. Correct required information</td> <td></td> </tr> <tr> <td>6. Click "Submit"</td> <td></td> </tr> </tbody> </table>	User Activity	System Response	4. Click "Submit"	<ul style="list-style-type: none"> Validate required information Required information not valid Return "New Account" page to the user, containing originally submitted values, and an appropriate error message. 	5. Correct required information		6. Click "Submit"							
User Activity	System Response														
4. Click "Submit"	<ul style="list-style-type: none"> Validate required information Required information not valid Return "New Account" page to the user, containing originally submitted values, and an appropriate error message. 														
5. Correct required information															
6. Click "Submit"															
Business Rules:	<p>Required fields:</p> <ul style="list-style-type: none"> Name – Text, 50 Characters Username – Text, 50 Characters Email – Text, 50 Characters Password – Text, 6-12 Characters, masked, reasonably secure Confirm Password – Text, 6-12 Characters, masked, must match Password 														

Special Requirements:	<ul style="list-style-type: none">• A username can only exist once in the database. If a username is entered that already exists, an error should be returned to the user indicating that the username is already in use.• Any input into the web-based form should be sanitized for SQL Injection or Cross-Site Scripting attacks.• The user's password must be stored in a hashed format in the database, such that decryption is not possible.
Assumptions:	
Notes and Issues:	

Use Case 2-D: Log In

Use Case ID:	2-D		
Use Case Name:	Log In		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Employee	
Description:	Employees of the company need the ability to add/update/delete jobs on the job board. In order to accomplish this, they will need to log in to the employee section of the website.	
Trigger:	None	
Preconditions:	<ol style="list-style-type: none"> Employee is able to use a web browser and navigate to the website. Employee has performed the “Create an Account” use case and has a valid username and password. 	
Postconditions:	<ol style="list-style-type: none"> User’s session is marked as authenticated and is granted entry to the employee section of the website. 	
Normal Flow:	User Activity	System Response
	1. Visit the “Employee Login” page	<ul style="list-style-type: none"> The system returns the web page, which contains a form to collect the user’s login information.
	2. Fill out username and password	
	3. Click “Submit”	<ul style="list-style-type: none"> Validate username and password against a database table. Return “Job Manager” page to the user containing the Employee’s current job listings.
Alternative Flow 1:	User Activity	System Response
	3. Click “Submit”	<ul style="list-style-type: none"> Validate required information Username and password not valid Return “Employee Login” page to the user, containing originally submitted username, blank password, and a generic error message (ex: “Invalid username or password”).
	4. Correct username or password	
	5. Click “Submit”	
Business Rules:	Required fields: <ul style="list-style-type: none"> Username – Text, 50 Characters Password – Text, 6-12 Characters, masked 	
Special Requirements:	<ul style="list-style-type: none"> Any input into the web-based form should be sanitized for SQL Injection or Cross-Site Scripting attacks. 	
Assumptions:		
Notes and Issues:		

Use Case 2-E: Create Job Listing

Use Case ID:	2-E		
Use Case Name:	Create Job Listing		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Employee	
Description:	Employees who are logged into the employee section of the website can create a job listing by filling out required fields. The completion of the job listing will make it viewable by users of the website.	
Trigger:	None	
Preconditions:	1. Employee has performed the “Log In” use case, and is currently logged into the website.	
Postconditions:	1. A new job listing is stored in the database. 2. Employee is redirected to the “Job Manager” page.	
Normal Flow:	User Activity	System Response
	1. Click the “Create Job Listing” link on the “Job Manager” page	<ul style="list-style-type: none"> The system returns the web page, which contains a form to collect the job listing information.
	2. Fill out required information	
Alternative Flow 1:	User Activity	System Response
	3. Click “Submit”	<ul style="list-style-type: none"> Validate required information Required information not valid Return “Create Job Listing” page to the user, containing originally submitted values, and an appropriate error message.
	4. Correct required information	
Alternative Flow 2:	User Activity	System Response
	3. Click “Cancel”	<ul style="list-style-type: none"> Return the “Job Manager” page to the user without saving any values.
Business Rules:	<p>Required fields:</p> <ul style="list-style-type: none"> Job Title – Text, 50 Characters Department – Text, 50 Characters Annual Salary – Money, 2 decimal places Job Description – Text, 100 Characters Expected Start Date – Date (mm/dd/yyyy) <p>Optional fields:</p> <ul style="list-style-type: none"> None 	

Special Requirements:	• Any input into the web-based form should be sanitized for SQL Injection or Cross-Site Scripting attacks.
Assumptions:	
Notes and Issues:	

Use Case 2-F: Update Job Listing

Use Case ID:	2-F		
Use Case Name:	Update Job Listing		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Employee	
Description:	Employees who are logged into the employee section of the website can edit an existing job listing. The completion of the modification will make the updated information viewable by users of the website.	
Trigger:	None	
Preconditions:	<ol style="list-style-type: none"> 1. Employee has performed the “Log In” use case, and is currently logged into the website. 2. A job listing exists in the database, and is visible in the “Job Manager” page. 	
Postconditions:	<ol style="list-style-type: none"> 1. Modifications to the job listing are stored in the database. 2. Employee is redirected to the “Job Manager” page. 	
Normal Flow:	User Activity System Response	
	1. Click the “Edit Job Listing” link on the “Job Manager” page	<ul style="list-style-type: none"> • The system returns the web page, which contains a form to edit the job listing information. The form is pre-populated with the current job listing information.
	2. Update job listing information	
Alternative Flow 1:	User Activity System Response	
	3. Click “Submit”	<ul style="list-style-type: none"> • Validate required information • Required information not valid • Return “Edit Job Listing” page to the user, containing originally submitted values, and an appropriate error message.
	4. Correct required information	
Alternative Flow 2:	User Activity System Response	
	3. Click “Cancel”	<ul style="list-style-type: none"> • Return the “Job Manager” page to the user without saving any values.
Business Rules:	<p>Required fields:</p> <ul style="list-style-type: none"> • Job Title – Text, 50 Characters • Department – Text, 50 Characters • Annual Salary – Money, 2 decimal places • Job Description – Text, 100 Characters • Expected Start Date – Date (mm/dd/yyyy) <p>Optional fields:</p>	

	<ul style="list-style-type: none">• None
Special Requirements:	<ul style="list-style-type: none">• Any input into the web-based form should be sanitized for SQL Injection or Cross-Site Scripting attacks.
Assumptions:	
Notes and Issues:	

Use Case 2-G: Delete Job Listing

Use Case ID:	2-G		
Use Case Name:	Delete Job Listing		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

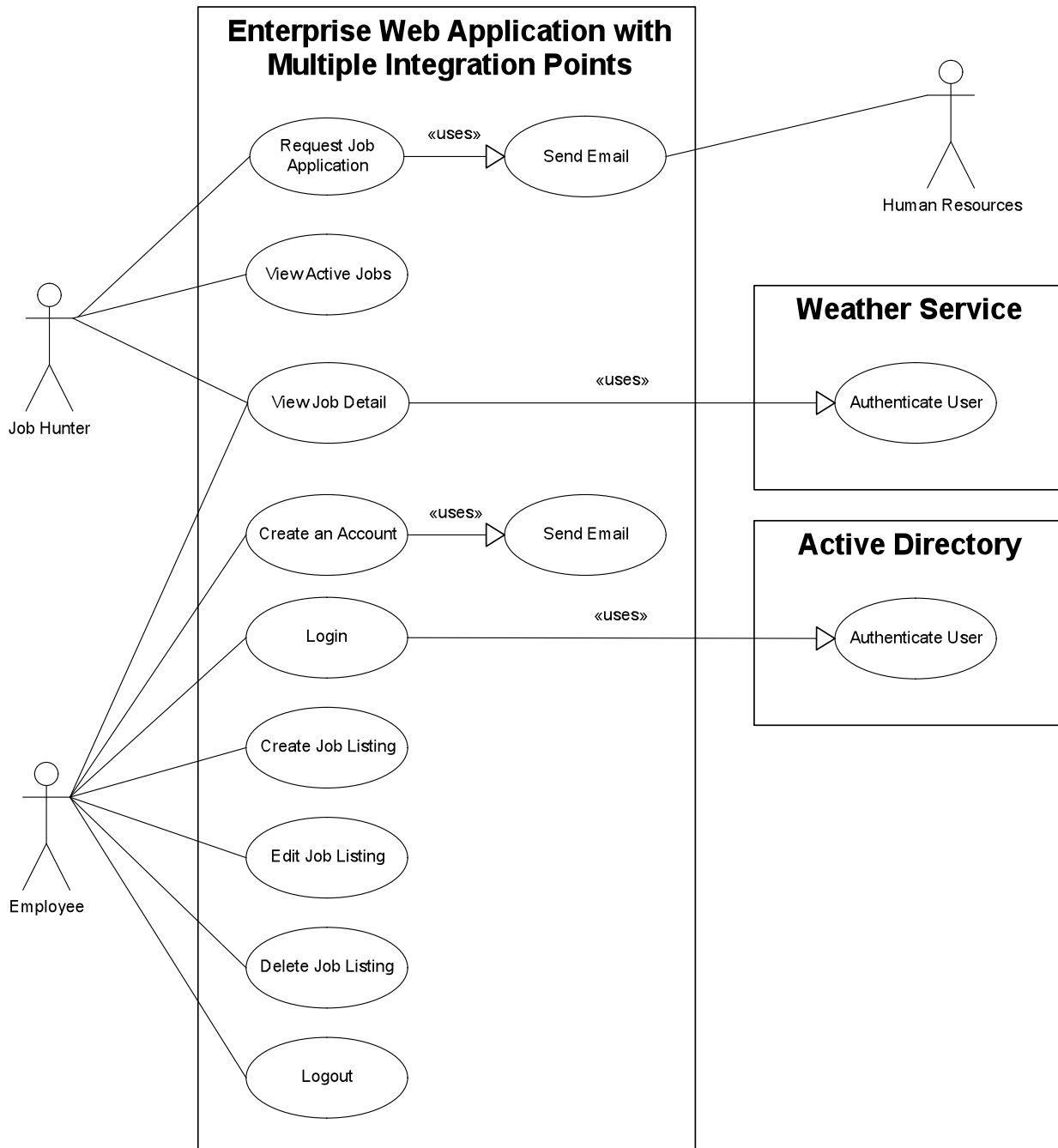
Actors:	Employee	
Description:	Employees who are logged into the employee section of the website can delete a job listing. The deletion of the job listing will remove it from the database and the customer area of the website.	
Trigger:	None	
Preconditions:	<ol style="list-style-type: none"> 1. Employee has performed the “Log In” use case, and is currently logged into the website. 2. A job listing exists in the database, and is visible in the “Job Manager” page. 	
Postconditions:	<ol style="list-style-type: none"> 1. The job listing is removed from the database. 2. Employee is redirected to the “Job Manager” page. 	
Normal Flow:	User Activity System Response	
	<ol style="list-style-type: none"> 1. Click the “Delete Job Listing” link on the “Job Manager” page 2. Click “Confirm” 	<ul style="list-style-type: none"> • The system returns the “Delete Job Listing Confirmation” web page, which contains the option to Confirm or Cancel the deletion process. • Delete the job listing from the database. • Return the “Job Manager” page to the user, which does not contain the deleted job listing.
Alternative Flow 1:	User Activity System Response	
	<ol style="list-style-type: none"> 2. Click “Cancel” 	<ul style="list-style-type: none"> • Return the “Job Manager” page to the user without deleting the job listing.
Business Rules:		
Special Requirements:		
Assumptions:		
Notes and Issues:		

Use Case 2-H: Logout

Use Case ID:	2-H		
Use Case Name:	Logout		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Employee					
Description:	Employees who are logged into the employee section of the website can logout. The logout of the employee will invalidate their session and require the employee to login again to access the employee area of the website.					
Trigger:	None					
Preconditions:	1. Employee has performed the “Log In” use case, and is currently logged into the website.					
Postconditions:	1. Employee’s session is invalidated. 2. Employee is redirected to the “Log In” page.					
Normal Flow:	<table border="1"> <thead> <tr> <th>User Activity</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1. Click the “Logout” link on the “Job Manager” page</td> <td> <ul style="list-style-type: none"> The system invalidates the employee’s session. The system redirects the employee to the “Log In” web page. </td> </tr> </tbody> </table>		User Activity	System Response	1. Click the “Logout” link on the “Job Manager” page	<ul style="list-style-type: none"> The system invalidates the employee’s session. The system redirects the employee to the “Log In” web page.
User Activity	System Response					
1. Click the “Logout” link on the “Job Manager” page	<ul style="list-style-type: none"> The system invalidates the employee’s session. The system redirects the employee to the “Log In” web page. 					
Alternative Flows:						
Business Rules:						
Special Requirements:						
Assumptions:						
Notes and Issues:						

Profile 3: Enterprise Web Application with Multiple Integration Points



Use Case 3-A: Request Job Application

Use Case ID:	3-A		
Use Case Name:	Request Job Application		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Identical to Use Case 1-A from profile 1.

Use Case 3-B: View Active Jobs

Use Case ID:	3-B		
Use Case Name:	View Active Jobs		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Job Applicant	
Description:	Customers visiting a company website can view active jobs for that company.	
Trigger:	None	
Preconditions:	1. User is able to use a web browser and navigate to the website.	
Postconditions:	1. The user is shown a web page containing active jobs.	
Normal Flow:	User Activity	System Response
	1. Visit the “View Active Jobs” page	<ul style="list-style-type: none"> The system returns the web page, which contains a list of active jobs. The data will be read real-time from a database. Pagination of greater than 10 jobs is an optional feature. The job listings will display in a list format, and show the following fields: Job Title.
	2. View Job Listings page	
	3. Close web browser	
Alternative Flows:		
Business Rules:		
Special Requirements:		
Assumptions:		
Notes and Issues:		

Use Case 3-C: View Job Detail

Use Case ID:	3-C		
Use Case Name:	View Job Detail		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Job Applicant or Employee							
Description:	Job Hunters visiting a company website can view detail of a selected job.							
Trigger:	None							
Preconditions:	<ol style="list-style-type: none"> 1. User is able to use a web browser and navigate to the website. 2. At least one job exists on the View Active Jobs page, and the user has selected a job to view. 							
Postconditions:	<ol style="list-style-type: none"> 1. The user is shown a web page containing the information about the selected job. 							
Normal Flow:	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr style="background-color: #4F81BD; color: white;"> <th style="width: 50%;">User Activity</th> <th style="width: 50%;">System Response</th> </tr> </thead> <tbody> <tr> <td style="vertical-align: top;"> <ol style="list-style-type: none"> 1. Selects the “View Job Detail” link from the “View Active Jobs” page </td> <td style="vertical-align: top;"> <ul style="list-style-type: none"> • The system returns the web page, which contains the job information for the selected job. The data will be read real-time from a database. • Page displayed to the user will be read-only, and will contain the following fields: Job Title, Department, Location, Annual Salary, Job Description, Expected Start Date • Page displayed will include weather information retrieved from a weather web service for the Location associated with the job listing. </td> </tr> <tr> <td colspan="2" style="vertical-align: top;"> <ol style="list-style-type: none"> 2. View Job Detail page </td> </tr> </tbody> </table>		User Activity	System Response	<ol style="list-style-type: none"> 1. Selects the “View Job Detail” link from the “View Active Jobs” page 	<ul style="list-style-type: none"> • The system returns the web page, which contains the job information for the selected job. The data will be read real-time from a database. • Page displayed to the user will be read-only, and will contain the following fields: Job Title, Department, Location, Annual Salary, Job Description, Expected Start Date • Page displayed will include weather information retrieved from a weather web service for the Location associated with the job listing. 	<ol style="list-style-type: none"> 2. View Job Detail page 	
User Activity	System Response							
<ol style="list-style-type: none"> 1. Selects the “View Job Detail” link from the “View Active Jobs” page 	<ul style="list-style-type: none"> • The system returns the web page, which contains the job information for the selected job. The data will be read real-time from a database. • Page displayed to the user will be read-only, and will contain the following fields: Job Title, Department, Location, Annual Salary, Job Description, Expected Start Date • Page displayed will include weather information retrieved from a weather web service for the Location associated with the job listing. 							
<ol style="list-style-type: none"> 2. View Job Detail page 								
Alternative Flows:								
Business Rules:								
Special Requirements:	<ul style="list-style-type: none"> • If the weather web service is unavailable or fails in any fashion, the job detail page must still display properly, and contain a reasonable error message in the weather section. 							
Assumptions:								
Notes and Issues:								

	<p>4. Click “Submit”</p> <ul style="list-style-type: none"> • Validate required information • Required information not valid • Return “New Account” page to the user, containing originally submitted values, and an appropriate error message. <p>5. Correct required information</p> <p>6. Click “Submit”</p>
<p>Business Rules:</p>	<p>Required fields:</p> <ul style="list-style-type: none"> • Name – Text, 50 Characters • Email – Text, 50 Characters • Active Directory Username – Text, 50 Characters • Active Directory Password – Text, 6-12 Characters, masked, reasonably secure
<p>Special Requirements:</p>	<ul style="list-style-type: none"> • A username can only exist once in the database. If a username is entered that already exists, an error should be returned to the user indicating that the username is already in use. • Any input into the web-based form should be sanitized for SQL Injection or Cross-Site Scripting attacks. • The user’s password must not be stored in the database.
<p>Assumptions:</p>	
<p>Notes and Issues:</p>	

Use Case 3-E: Log In

Use Case ID:	3-E		
Use Case Name:	Log In		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Employee	
Description:	Employees of the company need the ability to add/update/delete jobs on the job board. In order to accomplish this, they will need to log in to the employee section of the website.	
Trigger:	None	
Preconditions:	<ol style="list-style-type: none"> 1. Employee is able to use a web browser and navigate to the website. 2. Employee has performed the “Create an Account” use case and has a valid username and password. 	
Postconditions:	<ol style="list-style-type: none"> 1. User’s session is marked as authenticated and is granted entry to the employee section of the website. 	
Normal Flow:	User Activity	System Response
	1. Visit the “Employee Login” page	<ul style="list-style-type: none"> • The system returns the web page, which contains a form to collect the user’s login information.
	2. Fill out username and password	
	3. Click “Submit”	<ul style="list-style-type: none"> • Validate username and password against Active Directory. • Return “Job Manager” page to the user containing the Employee’s current job listings.
Alternative Flow 1:	User Activity	System Response
	3. Click “Submit”	<ul style="list-style-type: none"> • Validate required information • Username and password not valid • Return “Employee Login” page to the user, containing originally submitted username, blank password, and a generic error message (ex: “Invalid username or password”).
	4. Correct username or password	
	5. Click “Submit”	
Business Rules:	Required fields: <ul style="list-style-type: none"> • Username – Text, 50 Characters • Password – Text, 6-12 Characters, masked 	
Special Requirements:	<ul style="list-style-type: none"> • Any input into the web-based form should be sanitized for SQL Injection or Cross-Site Scripting attacks. 	
Assumptions:		
Notes and Issues:		

Use Case 3-F: Create Job Listing

Use Case ID:	3-F		
Use Case Name:	Create Job Listing		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Employee	
Description:	Employees who are logged into the employee section of the website can create a job listing by filling out required fields. The completion of the job listing will make it viewable by users of the website.	
Trigger:	None	
Preconditions:	1. Employee has performed the “Log In” use case, and is currently logged into the website.	
Postconditions:	1. A new job listing is stored in the database. 2. Employee is redirected to the “Job Manager” page.	
Normal Flow:	User Activity	
	System Response	
	1. Click the “Create Job Listing” link on the “Job Manager” page	<ul style="list-style-type: none"> The system returns the web page, which contains a form to collect the job listing information. Contained in the form is a Department combo box which is pre-populated with department information from the Department table. Contained in the form is a Location combo box which is empty
	2. Fill out required information	
	3. Select Department from combo box	<ul style="list-style-type: none"> Using AJAX, the system populates the Location combo box with location information for the selected Department.
4. Click “Submit”	<ul style="list-style-type: none"> Validate required information Insert the job listing information into the database. Return the “Job Manager” page to the user, which contains the newly added job listing. 	
Alternative Flow 1:	User Activity	
	System Response	
	4. Click “Submit”	<ul style="list-style-type: none"> Validate required information Required information not valid Return “Create Job Listing” page to the user, containing originally submitted values, and an appropriate error message.
5. Correct required information		
6. Click “Submit”		
Alternative Flow 2:	User Activity	System Response

	<p>4. Click “Cancel”</p> <ul style="list-style-type: none"> • Return the “Job Manager” page to the user without saving any values.
Business Rules:	<p>Required fields:</p> <ul style="list-style-type: none"> • Job Title – Text, 50 Characters • Department – Combo box • Location – Combo box • Annual Salary – Money, 2 decimal places • Job Description – Text, 100 Characters • Expected Start Date – Date (mm/dd/yyyy) <p>Optional fields:</p> <ul style="list-style-type: none"> • None
Special Requirements:	<ul style="list-style-type: none"> • Any input into the web-based form should be sanitized for SQL Injection or Cross-Site Scripting attacks.
Assumptions:	
Notes and Issues:	

	6. Click “Submit”				
Alternative Flow 2:	<table border="1"> <thead> <tr> <th>User Activity</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>4. Click “Cancel”</td> <td> <ul style="list-style-type: none"> Return the “Job Manager” page to the user without saving any values. </td> </tr> </tbody> </table>	User Activity	System Response	4. Click “Cancel”	<ul style="list-style-type: none"> Return the “Job Manager” page to the user without saving any values.
	User Activity	System Response			
4. Click “Cancel”	<ul style="list-style-type: none"> Return the “Job Manager” page to the user without saving any values. 				
Business Rules:	<p>Required fields:</p> <ul style="list-style-type: none"> Job Title – Text, 50 Characters Department – Combo box Location – Combo box Annual Salary – Money, 2 decimal places Job Description – Text, 100 Characters Expected Start Date – Date (mm/dd/yyyy) <p>Optional fields:</p> <ul style="list-style-type: none"> None 				
Special Requirements:	<ul style="list-style-type: none"> Any input into the web-based form should be sanitized for SQL Injection or Cross-Site Scripting attacks. 				
Assumptions:					
Notes and Issues:					

Use Case 3-H: Delete Job Listing

Use Case ID:	3-H		
Use Case Name:	Delete Job Listing		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Employee	
Description:	Employees who are logged into the employee section of the website can delete a job listing. The deletion of the job listing will remove it from the database and the customer area of the website.	
Trigger:	None	
Preconditions:	<ol style="list-style-type: none"> 1. Employee has performed the “Log In” use case, and is currently logged into the website. 2. A job listing exists in the database, and is visible in the “Job Manager” page. 	
Postconditions:	<ol style="list-style-type: none"> 1. The job listing is removed from the database. 2. Employee is redirected to the “Job Manager” page. 	
Normal Flow:	User Activity	System Response
	<ol style="list-style-type: none"> 1. Click the “Delete Job Listing” link on the “Job Manager” page 2. Click “Confirm” 	<ul style="list-style-type: none"> • The system returns the “Delete Job Listing Confirmation” web page, which contains the option to Confirm or Cancel the deletion process. • Delete the job listing from the database. • Return the “Job Manager” page to the user, which does not contain the deleted job listing.
Alternative Flow 1:	User Activity	System Response
	<ol style="list-style-type: none"> 2. Click “Cancel” 	<ul style="list-style-type: none"> • Return the “Job Manager” page to the user without deleting the job listing.
Business Rules:		
Special Requirements:		
Assumptions:		
Notes and Issues:		

Use Case 3-I: Logout

Use Case ID:	3-I		
Use Case Name:	Logout		
Created By:	Anthony Park	Last Updated By:	Anthony Park
Date Created:	02/08/2007	Date Last Updated:	02/08/2007

Actors:	Employee					
Description:	Employees who are logged into the employee section of the website can logout. The logout of the employee will invalidate their session and require the employee to login again to access the employee area of the website.					
Trigger:	None					
Preconditions:	1. Employee has performed the “Log In” use case, and is currently logged into the website.					
Postconditions:	1. Employee’s session is invalidated. 2. Employee is redirected to the “Log In” page.					
Normal Flow:	<table border="1"> <thead> <tr> <th>User Activity</th> <th>System Response</th> </tr> </thead> <tbody> <tr> <td>1. Click the “Logout” link on the “Job Manager” page</td> <td> <ul style="list-style-type: none"> The system invalidates the employee’s session. The system redirects the employee to the “Log In” web page. </td> </tr> </tbody> </table>		User Activity	System Response	1. Click the “Logout” link on the “Job Manager” page	<ul style="list-style-type: none"> The system invalidates the employee’s session. The system redirects the employee to the “Log In” web page.
User Activity	System Response					
1. Click the “Logout” link on the “Job Manager” page	<ul style="list-style-type: none"> The system invalidates the employee’s session. The system redirects the employee to the “Log In” web page. 					
Alternative Flows:						
Business Rules:						
Special Requirements:						
Assumptions:						
Notes and Issues:						