SKIM - Skill Management System

SRS Document

Disclaimer

This Software Requirements Specification document is a guideline. The document details all the high level requirements. The document should be used as a guideline by the students to design the Solution Architecture for the project. The document also describes the broad scope of the project and high level logical object model. But while developing the solution if the developer has a valid point to add more details being within the scope specified then it can be accommodated after consultation.

Skill Management System - SKIM

Introduction

The purpose of this document is to define scope and requirements of a Skill Management System - SKIM for a leading business conglomerate, ABCL. They have a major challenge in managing skills of its large workforce through their existing spreadsheet driven manual system. The proposed system will automate the skill management process across the enterprise.

This document should be used by the development team to architect the solution the project.

Management Summary

ABCL, a leading business conglomerate, is experiencing challenge in managing skills of its large workforce. Currently they have spreadsheet driven manual system, where skills of thousands of employees are maintained manually. This lacks well defined mechanisms to identify employees' skill up gradation needs, determine suitable training program, update skill levels and search skills. In order to address the challenge, they needed a web based system to automate skills management across the enterprise. The proposed system will:

1. Allow easy online maintenance of employee wise skills according to their respective roles. The employee skills will be managed by the immediate supervisor.
2. Allow HR to assess and identify gaps in employee skills in consultation with her/his supervisor.
3. Allow HR to set up an quarterly training calendar based on the gap analysis and automatically send mail notifications to the employee needing the training.
4. Allow management to identify employees with required skills in situations like new projects and initiatives.

SKIM will be a web-based system. It will be designed & developed to run on IBM WebSphere Application Server and IBM DB2 Universal Database in a 2-tier architecture.

Key Assumptions

1. The planned release will maintain only skills and not their levels (e.g. expert, advance, etc.) to reduce the development time.

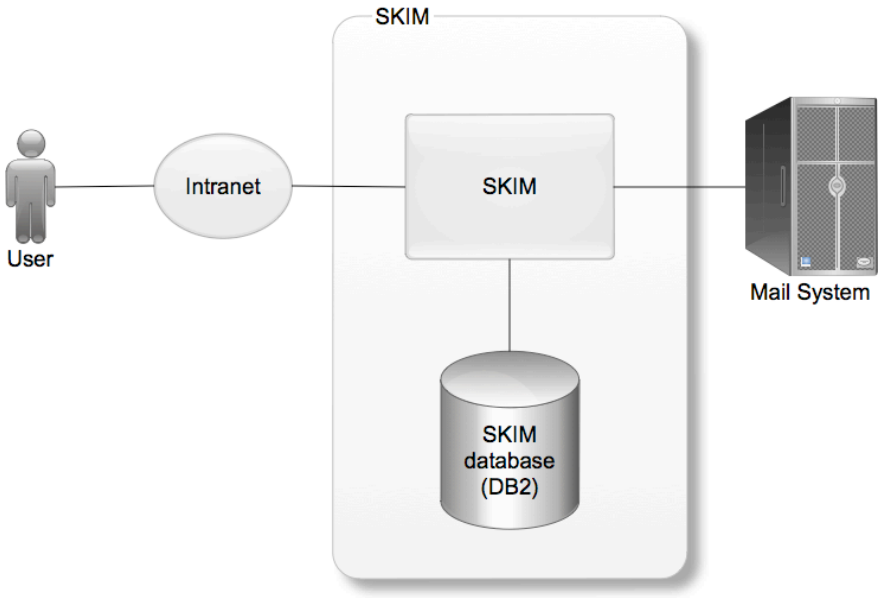
2. There is no approval workflow planned for this release. Examples of possible workflows are skill update by employee and approval by supervisor or approval driven training enrollment by the employee.

High Level Architecture

SKIM's high level architecture is illustrated through the context diagram shown in figure 1 below. It will have following categories of users:

1. Employees and their supervisors

2. HR (employee(s) with HR role)



SKIM Context Diagram

SKIM	The proposed web-based application, SKIM will be accessed by the ABCL employees to manage skills, gap analysis and scheduling trainings on an ongoing basis.
SKIM Database	This will hold all the SKIM data including roles, role wise skills, employee skills, trainings and skills developed by them, training schedule, and training history.
Mail Server	It is responsible for delivering notification e-mails to SKIM users.

Functional Requirements

The high level functional requirements for the SKIM system are outlined in the Use Case diagram described in this section.

SKIM will provide a secure user-id/password based secured login mechanism to access its services. The details of this are not outlined here. The development team is expected to create these keeping in mind the general practices followed by the web applications. Login will be a prerequisite to use SKIM.

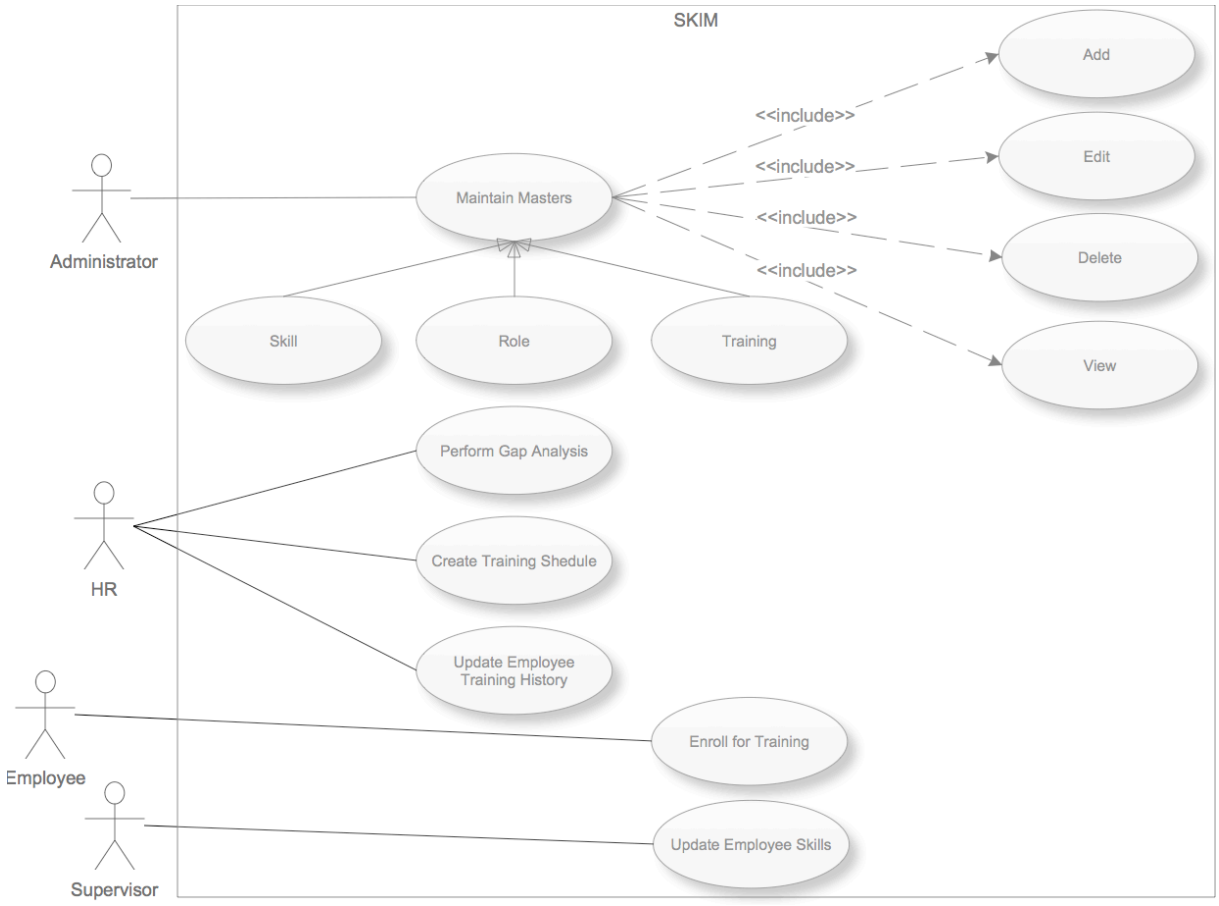
Once employee logs in, s/he can view his/her training & skills details on “My Page”. This will be the default landing page for every employee.

SKIM will provide a search mechanism to enable supervisors to search for employee’s possessing specific skills. The search will be carried out by entering one or more skills (separated by comma) in the search text box and clicking on search button. Search option will be available on all pages.

There are no use cases for creating/deleting employee records. This is out of scope from the current system. They are managed directly from the DB backend.

Use Case Diagrams

The following figure illustrates the Use Case diagram for the system.



Use Case Diagram

Use Cases

Maintain Masters

Use Case Element	Description
Number	UC.01
Application	Master maintenance in terms of basic operations viz. **add**, **edit**, **delete** and **view**. All master maintenance i.e. skill, role, and training are child use cases of this Use Case.
Use Case Name	Maintain Masters
Primary Actor	Administrator
Secondary Actor	None
Pre-condition	None
Trigger	Administrator clicks on the **Maintain Masters** menu item on the admin interface page
Basic Flow	• System presents a list of masters that can be maintained. Administrator selects the desired master. • System displays a list *view* and links for *add*, *edit* and *delete*. 1. In case add, a new master record data entry form is presented. The master record is saved on clicking the save button, provided form clears all the data validations (if any). The list view is updated accordingly. 2. In case of edit, from the list view user is prompted to select the desired record to edit, Selected record is opened for editing. The edited master record is updated on clicking the update button, provided form clears all the data validations (if any). 3. In case of delete, from the list view user selects the check box(s) against each record. Selected records are deleted up on clicking the delete button. However, user is presented a confirmation dialog before deleting the records.
Alternate Flow	• In event of any error, it is clearly displayed and user is asked to renter data or perform operation again.
Output	System displays the details of the successful operation.

Perform Gap Analysis

Use Case Element	Description
Number	UC.02
Application	To perform employee wise skills gap analysis to plan quarterly trainings.
Use Case Name	Perform Gap Analysis
Primary Actor	HR
Secondary Actor	None

Use Case Element	Description
Pre-condition	None
Trigger	User clicks on the **Perform Gap Analysis** menu item on the HR page
Basic Flow	• System displays the list of employees who have gaps in their skills against the role that they are performing. • User updates skill gaps for every employee. By default, system shows the difference of skills between role based skills and employee's current skill set. This may be edited/updated by HR optionally. • User clicks on save button. Employee's skill gaps are updated.
Alternate Flow	• In event of any error, HR is asked to make correction. • If HR clicks cancel button then the operation is cancelled and gaps are not updated.
Output	Error message, in event of error.

Create Training Schedule

Use Case Element	Description
Number	UC.03
Application	To create a training schedule so that employees can attend trainings to upgrade their skills.
Use Case Name	Create Training Schedule
Primary Actor	HR
Secondary Actor	None
Pre-condition	None
Trigger	User clicks on the **Create Training Schedule** menu item on the HR page
Basic Flow	• System displays the consolidated lists of skills from the skills gap of every employee. Against each skill, the count of employees needing that skill is also displayed. The list is sorted in descending order of this count. • User selects a skill from the list. System automatically suggests required training(s) from the training master based on the skill. If there are multiple trainings then a list is displayed with the list of skills (*only from the above set*) addressed by each training. The training that addresses the maximum skills is shown on the top. • User now selects a training and click schedule now button. This allows the user to define the start and end date (according to the training duration), venue and the instructor. Upon clicking save button the training is scheduled. The instructor and the employees needing that training (on the basis of their skill gap) are notified via e-mail. • The skill & training list is now updated. The process is repeated until all the skills have been addressed.
Alternate Flow	None

Use Case Element	Description
Output	E-mail notifications to employees and the instructors. Error messages, if any.

Update Employee Training History

Use Case Element	Description
Number	UC.04
Application	To update the training history of an employee
Use Case Name	Update Employee Training History
Primary Actor	Supervisor
Secondary Actor	None
Pre-condition	None
Trigger	User clicks on the **Update Employee Training History** menu item on the HR page
Basic Flow	• System displays the list of employees who have attended training(s). • User selects the required employee to update. The list of training the employee has attended from the last update is displayed. User may choose to update this list. Upon save, the new training history records for that employee are created.
Alternate Flow	None
Output	E-Mail to employee and his/her supervisor about the update.

Enroll for Training

Use Case Element	Description
Number	UC.05
Application	To enroll for a training for skill up-gradation.
Use Case Name	Enroll for Training
Primary Actor	Employee
Secondary Actor	None
Pre-condition	None
Trigger	Employee clicks on the **Enroll for Training** link on “My Page”

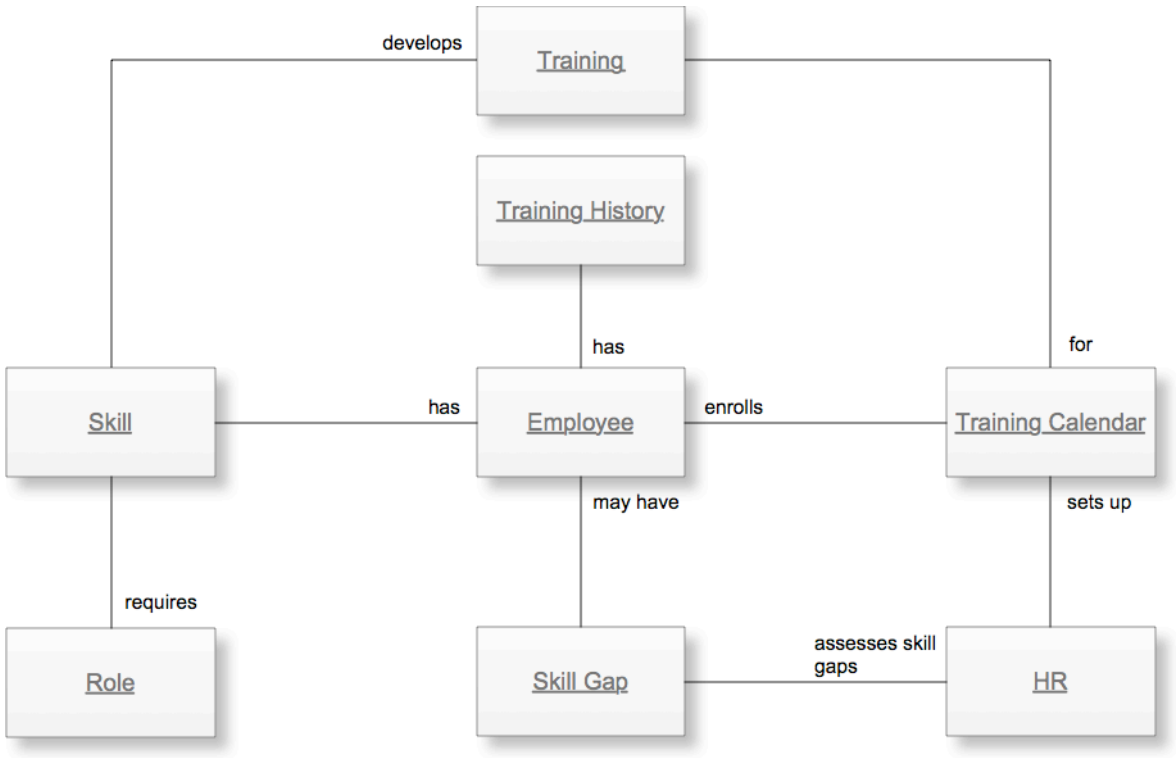
Use Case Element	Description
Basic Flow	• System displays the list of trainings on the basis of skill gaps. The training that addresses maximum skill gaps are shown on top. It also shows the balance trainings separately. • User can select a training to attend. Upon clicking enroll button, s/he is automatically enrolled for the selected training. The HR and instructor are notified on the enrollment via an e-mail. • User can repeat the above process any number of times. Currently it is not planned to carry out conflicting schedule in training enrollments. However, adventurous developers may attempt schedule conflict resolution.
Alternate Flow	None
Output	E-Mail to HR and the instructor about the enrollment.

Update Employee Skills

Use Case Element	Description
Number	UC.06
Application	To update employee's skill after his/her skill up-gradation.
Use Case Name	Update Employee Skills
Primary Actor	Supervisor
Secondary Actor	None
Pre-condition	None
Trigger	Employee clicks on the **Update Employee Skills** link on "My Page". This link is visible to the employee who have people reporting to them.
Basic Flow	• System displays the list of all the direct reports. • User selects the employee to update skills and updates the required skills along with the reason for update (e.g. training attended etc.). Upon clicking save button, skills are updated and skill gaps may accordingly be reduced.
Alternate Flow	None
Output	E-Mail to the employee confirming skill updates.

Logical Object Model

A high level logical object model of the system is shown below. During technical design it will be transformed into a physical model covering all system entities. Such a diagram will include their relationship and its cardinality.



Logical Object Model

1. Employee performs a role. Employee's role changes with time in the organization due to his/her movement between various department or role elevation i.e. promotion.
2. Employee has skills.
3. Employee reports to his/her supervisor (who is also an employee).
4. In order to perform a role, specific skills are required.
5. Employee may have skill gaps. Skill gap is defined as the gap between employee's current skills and the skills required to perform his/her role effectively. This gap is assessed by the HR.
6. Training develops skill and therefore reduce or eliminate skill gap.
7. HR also performs gap analysis to identify employee wise skill gaps. As the next step, HR sets up quarterly training calendar for various training on the basis prevailing skill gaps.
8. Employee enrolls for training based on her/his skill gaps. His/her training history builds up as s/he completes training(s). This is updated by the HR.
9. Employee's supervisor updates his/her skills periodically on the basis of training(s) attended by the employee and subsequent performance.

Database Design Guidelines

This involves the transformation of the use cases, state diagrams, and logical object model into detailed and optimized physical database table designs.

Typically persistent classes will map to table(s) with their attributes as columns of the table. In some cases a high level object may map in to a master-child table. Invoice is one such example where it maps in to "invoice_header" and "invoice_line_item" table.

Associations between two persistent objects are realized as foreign keys to the associated objects. A foreign key is a column in one table that contains the primary key value of the associated object.

Similarly, a standard technique in relational modeling is to use an intersection entity to represent many-to-many associations. Following is a broad checklist for physical database database design:

1. Database must be properly normalized except those instances where de-normalization help improves performance. This option must be used with special care.
2. All persistent classes that use the database for persistency must map to database structures.
3. Many-to-many relationships must have an intersecting table.
4. Primary keys should be defined for each table, unless there is a performance reason not to define a primary key.
5. Indexes should be defined to optimize access.
6. Data and referential integrity constraints should be defined.

Testing Approach

Quality of the software can be achieved with basic hygiene and consistency followed during design and development of User Interface(UI), Navigation, Validations as per the business process requirement.

To ensure the project delivers acceptable quality to the customer, its important to create a checklist of the conventions to be followed across. Common checks as below are for your reference during design and development:

Common Checks	Validation Type
Page Title is valid for the feature being provided on the page	UI
Order of the Data Entry Fields is logical as per the functionality being provided by the feature	UI
Order of the Display only Fields makes viewing and understanding easy for the user	UI
Spellings and Correctness of Label for the Data Entry and Display fields	UI
The labels are not wrapping onto another row thereby adding a blank row on the page	UI
The fields with drop down are displayed in single row instead of drop down coming on the next row	UI
Data Entry field basic validations are working i.e Text field /Numbers / Dates allow data for their type only	Functional
The dates are following a standard format dd/mmm/yyyy on all forms	UI
The color scheme of all forms i.e headers labels , alerts, entry fields are uniform throughout the application	UI
The action buttons for a New Data Entry Form are uniform for all forms that is allowing data entry	UI

Common Checks	Validation Type
The action buttons are performing the desired action e.g. "submit" is creating a new record if there are no errors and recording all the input fields, whereas 'cancel' is not creating a new record in the database	Functional
The links provided on the forms are opening correctly.	Functional
The data feed mechanism for Read and Write files is generating a log with count of entries.	Navigation

For testing purpose, dummy employee records can be created on the database directly from the DB backend. Similarly, employee role change etc can be simulated by updating the employee record directly from the DB backend.

Suggested Technical Reading

The project is aimed at making the student understand concepts of Design and Development using IBM Rational tools, Web Sphere Application Server and DB2 Database. The following reading reference is easy to understand and should be read to get a clear understanding of capabilities of the tools and how you would leverage them to execute a project.

Technical Reference	URL to access
RAD - Tackling challenges of software development with Rational Application Developer for WebSphere Software	http://www.ibm.com/developerworks/rational/library/08/0926_ackerman-mahate/index.html
IBM Education Assistant - Rational Application Developer 7.5	http://publib.boulder.ibm.com/infocenter/ieduasst/rtnv1r0/index.jsp?topic=/com.ibm.iea.rad_v7/rad/rad75.html
RSA-Overview of Rational Software Architect for WebSphere Software Version 7.5	http://www.ibm.com/developerworks/rational/library/08/0926_arnold/index.html
Using the new features of UML Modeler in IBM Rational Software Architect Version 7.5	http://www.ibm.com/developerworks/rational/library/08/0926_diu/index.html
Rational Technical Library	http://www.ibm.com/developerworks/rational/library/