Share-a-Slide

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.

Share-a-Slide Presentations Hub

Introduction

The purpose of this document is to define scope and requirements a Presentation Sharing System, that is being envisioned by a Infrastructure company operating across the country. The new system proposes to overcome the challenges of re-work on similar topics for such a spread out operations, where knowledge is of the essence.

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

Management Summary

In todays world, Information Overload, scatter bits of data could lead to

1. Inconsistent Communication to Customers / Partners
2. Increased Storage Costs
3. Productivity Loss
4. Efforts Disproportionate to Output
5. Risk of litigation

Share-a-Slide is focussed on lower cost of Storage, is self Indexing, Allows Searching, Viewing, Tagging, Marking as favorites, sharing of content with a specific user group. The system will have 3 focus areas mentioned below around which the application features will be built.

1. Access metrics - Tagging, Number of time downloaded, How many employees marked presentation as favorite.
2. Collaboration - Posting comments
3. Search - Tagging

The proposed solution will be designed & developed to run on IBM WebSphere Application Server and IBM DB2 Universal Database in a 2-tier architecture.

Key Assumptions

1. A typical Document Management System brings in a high degree of security element at each stage of its usage, this system is being proposed from the perspective of sharing knowledge on a like kind of topics with your peers. This presentation sharing system follows no rules for posting, accessing, re-using content for a particular knowledge domain within an organization.

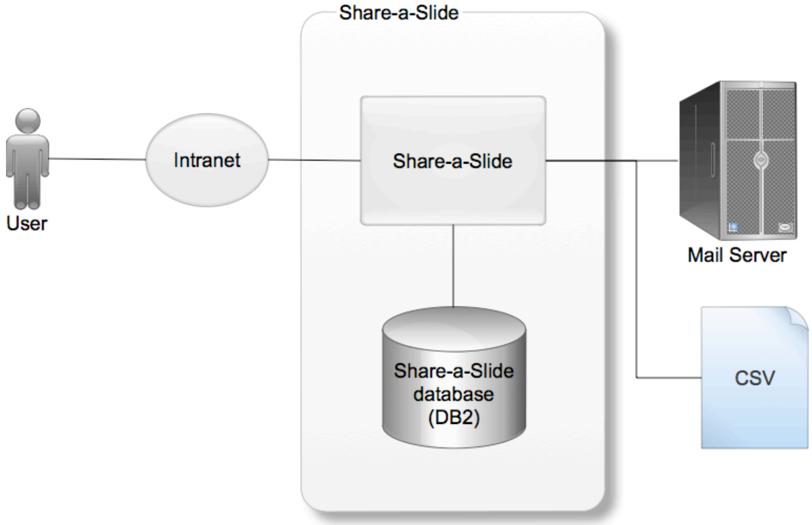
2. The employees for a Sales and Marketing across the country can access this system. The backend activity takes care of providing them with user id and password.

High Level Architecture

Share-a-Slide's high level architecture is illustrated through the context diagram shown below. It will have following categories of users:

1. Employee

2. Administrator



Share-a-Slide Context Diagram

Share-a-Slide	The system allows users to share their presentation on various topics. The users accessing content can post their comments, Tag the content, even follow employees to get informed as and when the content is posted by them
Share-a-Slide Database	The presentations, tags, followers are all stored in the database
CSV	Office locations, employee basic details with email id are uploaded using CSV
Mail Server	Notifications are sent using the mail server

Functional Requirements

The high level functional requirements for the Share-a-Slide are outlined in the Use Case diagram described in this section.

Share-a-Slide 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 Share-a-Slide. Internal users will be provided user id/password pair separately.

Once user logs in. the following sections can be accessible.

My Page

1. My Presentations (Created by User, Latest topics on top are displayed with statistics (Favorite Count, Download Count, Tags link)

2. My Favorites (The Presentations marked as favorite by the logged in employee)

3. My Followers (The Presentations of logged in user that are marked as favorite by other users)

Home Page

1. Latest Presentations uploaded (By all users of this application)

2. Top Favorites (Ranking of Favorites, Display top 5 and rest on click of more)

3. Most Downloaded (Ranking of Downloaded content, Display top 5 and rest on click of more)

4. Top 10 Tags (Ranking of Same Tags that have been given to Presentations by many users)

Search Page

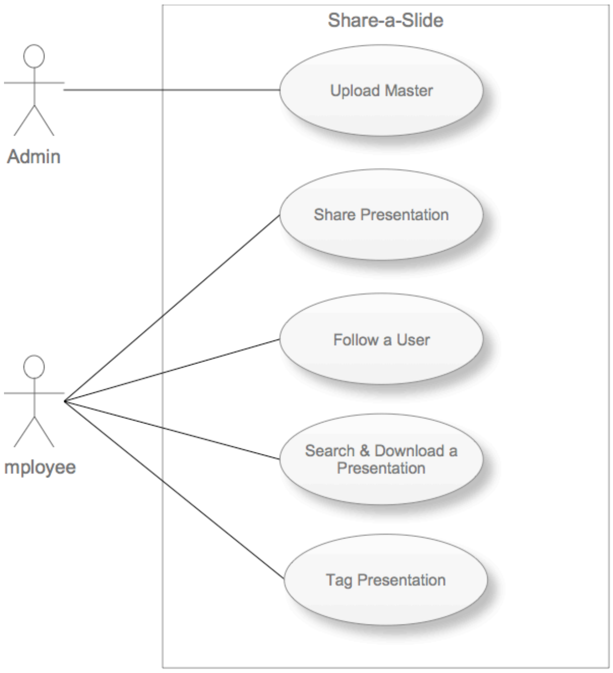
1. TAG Cloud is displayed, Click on any one to displays all Presentations for the selected Tag.

2. Location Search displays Presentations for the selected location

The views contain statistics of the Presentation, Number of Views, Favorite Count, Download Count, Tags link

Use Case Diagrams

The following figure illustrates the Use Case diagram for the system. The MiS use cases are not detailed here.



Use Case Diagram

Use Cases

Upload Masters

Use Case Element	Description
Number	UC.01
Application	Master are uploaded using CSV format. Company Locations Master contain Location id, Location Employees Master contains Employee Id, Employee Name, Location id, Email id
Use Case Name	Upload Masters
Primary Actor	Administrator
Secondary Actor	None
Pre-condition	None
Trigger	Administrator clicks on the **Upload Masters** menu item on the admin interface page
Basic Flow	System prompts for the file name to be uploaded. Standard file upload dialog is presented to select a file from the local system. • The selected file data is uploaded in the related masters; if an existing record is encountered, the old details are replaced with the new details.
Alternate Flow	• In event of incorrect CSV format, system gives an error and NO data is uploaded. • Operation is cancelled
Output	System displays the number of records uploaded. It also highlights the number of records updated (i.e. already existing ones being replaced)

Share Presentation

Use Case Element	Description
Number	UC.02
Application	The users share presentations that are already been presented during their work execution.
Use Case Name	Share Presentation
Primary Actor	Employee
Secondary Actor	None
Pre-condition	None
Trigger	User clicks on the **Share Presentation** menu item on the My Page of Share-a-Slide application

Use Case Element	Description
Basic Flow	- The system displays a form with following information to be entered - Topic [The display Name in views] - Type [Sales Pitch, Strategy, Branding, Products, CBA, Events, etc] - Purpose - Brief Description - Development Time : In hours - Presented to: Select from Internal Mgmt / Top Mgmt / External - Tag it - Attach file [File selection dialog box opens and user selects the presentation] The above information is saved along with the Presentation as its meta data.
Alternate Flow	- User abandons the operation, no database impact
Output	The users who follow the updates from this user are notified by email.

Follow User

Use Case Element	Description
Number	UC.03
Application	Users have the opportunity to track and follow the contributions by various users across the company by Follow up in this Share-a-Slide
Use Case Name	Follow a user
Primary Actor	Employee
Secondary Actor	None
Pre-condition	None
Trigger	User clicks on the **View users** menu item on the Home page
Basic Flow	- The system displays by default the most followed users. - The user can search by name or view all users - A link to Follow is given alongside the user name, click on the Follow link to begin following - A twitter like page is displayed with the logged in users profile, Following and Followed by counts are shown - The count of presentations shard on the system are also displayed - Click on a particular user entry displays the user’s contribution over a period of time.
Alternate Flow	- None
Output	The users followed in this activity are notified of being followed by logged in user

Search & Download Presentation

Use Case Element	Description
Number	UC.04
Application	The platform is of sharing and re-using IP on the knowledge domain of the users of this system. The users like to Search, view and download what is of interest or benefit
Use Case Name	Search
Primary Actor	Employee
Secondary Actor	None
Pre-condition	None
Trigger	User clicks on the **Search** menu item on the home page
Basic Flow	• The search is provided on Tags and Location • By default the TAG Cloud id displayed for top 50 Tags, click on More link to display the next 50 and so on • TAG cloud is a collection of Tags displayed in varying sizes as per their frequency of use while tagging the presentations e.g. if a Tag name 'Social' is being displayed like SOCIAL that would imply this tag has been used for various users to tag presentations, also signifying visually that there is lot of content under this tag. • The select tag is used as a filter to display presentations. On top of tags, the user can apply location filter as well. • The list of locations is displayed for selection. Click on search will filter Presentations for a selected TAG uploaded by a user in selected location. • The search results display the entries with Topic, Type, Employee name, Statistics and a link to download • The user clicks on download, will open the standard dialog box to save the file on local drive. • On download completion the system updates the download count by +1 • Viewed count by +1
Alternate Flow	• None
Output	None

TAG Presentation

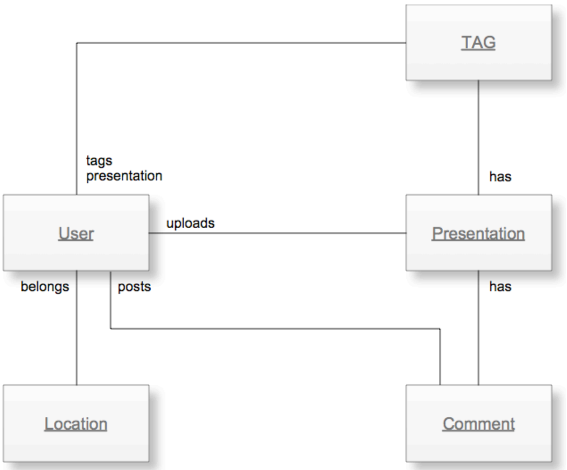
Use Case Element	Description
Number	UC.05
Application	The users like to categorize the presentation content in the context they can relate to or remember. This way they TAG content which they either search or come across on landing pages of the application

Use Case Element	Description
Use Case Name	Tag
Primary Actor	Employee
Secondary Actor	None
Pre-condition	None
Trigger	User clicks on **Tag it** link while viewing a meta data of a presentation in the application
Basic Flow	Assumption: The view of presentation is on screen and user is scanning through the presentations, meta data Any content that can be retrieved by keywords that a user can associate with in future are used to Tag content. Tag it will display a an entry box to add Tags, there could be more than 1 tag entry for a presentation. The Tags entered get added to the existing list. The system notifies the user whose presentation has been tagged.
Alternate Flow	- None
Output	The system notifies the user whose presentation has been tagged.

Marking a presentation as favorite also works the similar way. A link to mark as favorite is provided along with meta data, user can mark it as favorite, the entry will display on My Page section under My favorites.

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. Company has operations spread out at various locations

2. Users belong to a location. From each location users create presentations and upload them into Slide-a-share.

3. The presentations are Tagged by users from any location to make it contextually relevant and easy to pull out.

4. The users can have discussion in the form of posting comments on the presentations. The system maintains a thread chronologically.

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

Common Checks	Validation Type
The action buttons for a New Data Entry Form are uniform for all forms that is allowing data entry	UI
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

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/