Mentors & Protege

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.

Mentors & Protege

Introduction

The purpose of this document is to define scope and requirements of community portal for women - Mentors & Protege that will : a) aid development of knowledge inputs and e-services of relevance to women in India, b) encourage active involvement in Events and happenings in their areas, c) the broad aim is to involve participation with community, provide forum for them to earn.

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

Management Summary

The community is setup to leverage the experience of Volunteers in various fields that can be shared with the community at large. On this forum, volunteers welcome the activities of mentoring one or more women to enable them to become independent.

1. Register women as members into Mentors & Protege portal. They can access to the information on various Disease Categories of the Mentors & Protege initiative.

2. Engage mentors to pool in information on the various livelihood options that women can take up and get started.

3. The community members keep sharing their successes with various options they try.

4. Various events are posted for members to attend and participate in.

5. Members can post their classified advert for the services or products they begin to market as part of the mentoring programs.

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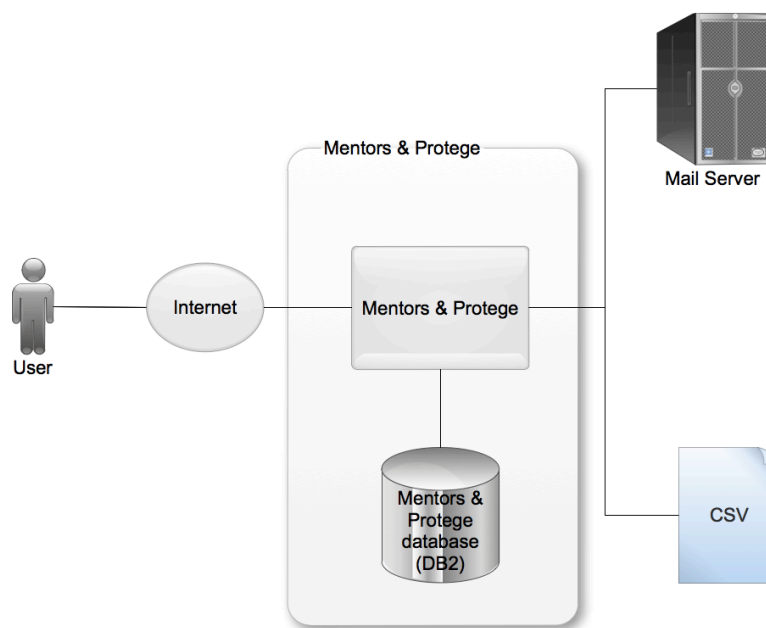
Key Assumptions

1. The system is ideally meant for women having mobiles for receiving SMS of the information they have subscribed to. For project scope, email id is being used for communicating.

2. The Project owner is familiar with community initiatives taken by various organizations. They are eye openers from the technology point of view.

High Level Architecture

Mentors & Protege high level architecture is illustrated through the context diagram shown below. It will have following categories of users:

1. Members (Protege)

2. Mentors

3. Volunteers (Providers of Mentors & Protege website)

4. Administrator

Mentors & Protege Context Diagram

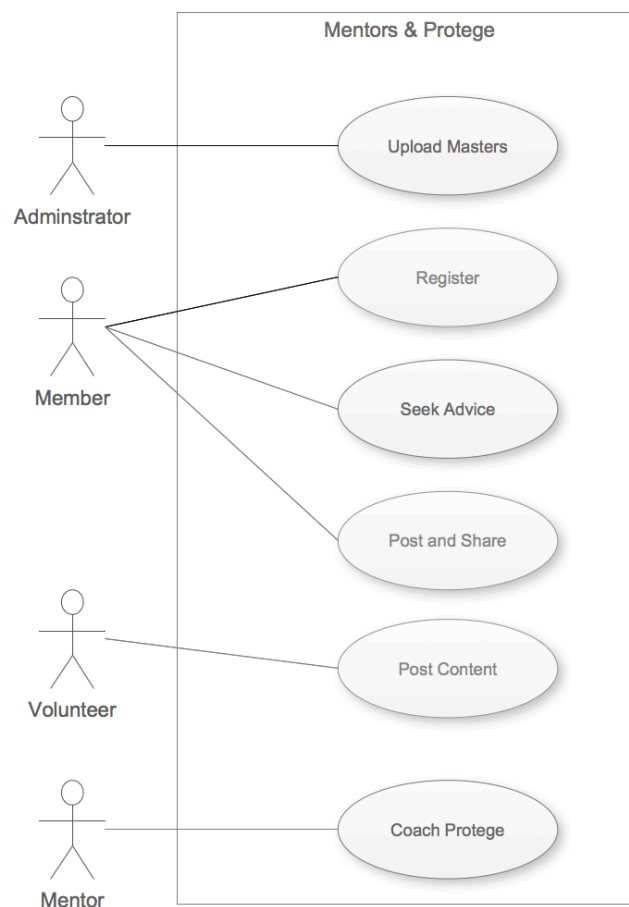|	
Mentors & Protege	Maintain Members profile by uploading their basic contact, geographical details along with any skill preference they may have. Volunteers cater to information needs for Members in their location. Maintain a list of Mentors for coaching the members assigned to the mentor.
Mentors & Protege Database	Designed to store Members profile, locations, Categories. Mentors, Volunteers, Content for posting on website, classifieds from the members
CSV	Masters like locations, Members, Categories, Mentors, Volunteers are uploaded via CSV
Mail Server	All notifications are routed through the Mail Server

Functional Requirements

The high level functional requirements for the Mentors & Protege are outlined in the Use Case diagram described in this section.

Mentors & Protege 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 Mentors & Protege. Internal users will be provided user id/password pair separately.

Once user logs in, s/he can view a web page with sections on 1) Member's success stories 2)Latest Events, 3) New Classified as per category. The discussions with mentors are displayed chronologically on the landing page of the website. The menu options shall primarily come from the use case of the role player.

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

Use Case Element	Description
Application	Masters in Mentors & Protege are uploaded using CSV Format State Master contains State id, State Name Location Master contains Location id, Location Name, State id Category Master contains Category id, Category Name (Employment based Skills are uploaded here) Mentors master contains mentor id, mentor name, location id, contact number, email id Volunteer master contains volunteer id, volunteer name, location, contact number and email id
Use Case Name	Upload Masters
Primary Actor	Administrator
Secondary Actor	None
Pre-condition	None
Trigger	User clicks on the **Upload Masters** menu item on the landing 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 tables; 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)

Register

Use Case Element	Description
Number	UC.02
Application	People, by word of mouth or reference in some articles covering the Mentors and Protege program visit the website to register as members
Use Case Name	Register
Primary Actor	Members
Secondary Actor	Volunteer
Pre-condition	User has not registered earlier in the system.
Trigger	User clicks on the **Register** link on the landing page

Use Case Element	Description
Basic Flow	• The system displays a form to take the following inputs • Name, House #, Street, Area, Location(City drop down), phone number, email id, user name(not more than 8 characters - Alpha numeric), Current employment status(Employed, Self-employed, Dependent), Choose category-display list of categories for mentoring. • Click on submit will generate a unique password for the member's record. • The system automatically assigns a mentor based on the location of the new member. • An email is sent to the registering user with details of the assigned mentor, as well as the login id and unique password.
Alternate Flow	None
Output	Email with id and password to member Email with protege information to Mentor.

Seek Advice

Use Case Element	Description
Number	UC.03
Application	Members visit the website to seek advice from mentors
Use Case Name	Seek Advice
Primary Actor	Members
Secondary Actor	None
Pre-condition	Authorized user is logged in to the system
Trigger	User clicks on the **Seek Advice** link on the landing page
Basic Flow	• The system displays member's information like name, location, mobile number etc. if the member has logged in to the system. The system also displays conversation on earlier topics. • The Advice asked by the Member is entered in the Seeking Advice box. • User selects the Category in which question falls. • The system displays success stories of other members under the selected category, who have followed mentors advice and have been happy with the results and share the same on this forum. • User may choose to click on 'Like' link place along side the success story. The like count of the story is incremented by 1. • Submit will post the question in the mentor's inbox for response.
Alternate Flow	None
Output	Notification to mentor for response awaited.

Coach Protege

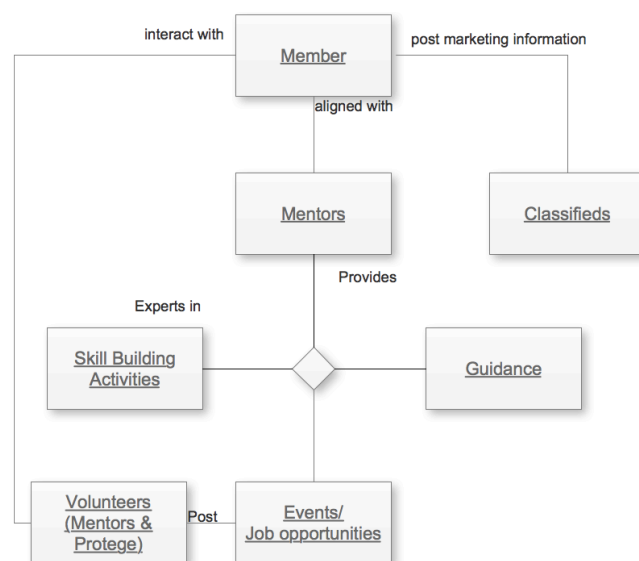| Use Case Element	Description
Number	UC.04
Application	Mentors respond to the question as a discussion thread
Use Case Name	Coach Protege
Primary Actor	Mentors
Secondary Actor	Volunteer
Pre-condition	None
Trigger	The user clicks on the **Coach Protege** link on the landing page
Basic Flow	• All questions appear with category as label with name of member and other details, date posted, success story liked. • The Mentor responds in the Advice box to be communicated to member • Mentor Clicks on Communicate. • An email for the same is sent form the system.
Alternate Flow	None
Output	Email to member for checking the response.

Post Content

Use Case Element	Description
Number	UC.05
Application	The volunteers take up the responsibility of posting content on the website that is of immediate use to its members. Earning opportunities/ Events around Skill building / Exhibitions, Community meets etc. are posted by the volunteers for the location they belong to.
Use Case Name	Post Content
Primary Actor	Volunteer
Secondary Actor	None
Pre-condition	User is an authorized role in the system
Trigger	The user clicks on the **Post Content** link on the landing page.
Basic Flow	The system displays list of Categories to select from.(Multi-select is possible) A form is displayed to capture the event details such as: Date, location, Content Type (Event/ Opportunity/ Buzz around) , Title, Details of content (Text box) The entries are saved as content for the Members to access on home page of the website in their respective locations.
Alternate Flow	Pressing Cancel abandons operation, no database gets affected
Output	None

Post and Share

Use Case Element	Description
Number	UC.06
Application	Members post success stories and are given the forum to post a classified for their job work or services they are able to provide.
Use Case Name	Post & Share
Primary Actor	Member
Secondary Actor	None
Pre-condition	None
Trigger	The user clicks on the **Post & Share** link on the landing page.
Basic Flow	• System displays a list view of the categories in which the user wishes to post or share • For Post entries, the system displays, the fields like Description, Cost per Unit, upload image of sample. • For Share entries, the system displays the fields such as Success Story-Title (5 words) Challenge faced (10 words), Mentor Name (Select from master), Story Text, Benefit, Picture upload if any. • The system saves information along with its Category and the member id.
Alternate Flow	Pressing Cancel abandons operation, no database gets affected
Output	None

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. Community Member is the customer for the Mentors & Protege System.

2. Members from various locations contact the Mentors & Protege Volunteers enquiring for assistance.

3. The Mentors & Protege system maintains list of mentors who take up the activity of counseling members on acquiring skills to earn money based on either selling products or services).

4. Volunteers posts information about events taking place or some skill building exercises in the areas of operation.

5. The members share their experiences on the site for others members to benefit.

6. Members are provided space on classified section to market their products or services.

7. Volunteers update the events, open employment opportunities, exhibitions in the areas of operation.

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
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/