**Synergy Distributed Meeting Scheduler System**

***Project Phase II***

**Deliverable**

**Team: HIGH-FLIERS**

**Team URL: http://www.utdallas.edu/~rxt058000/welcome.html**

|  |  |  |
| --- | --- | --- |
| **Name** | **UTD Email** | **SID** |
| Puneet Simha Team Leader | pxk068000@utdallas.edu | 11134514 |
| Deepthi Durgempudi | dxd072000@utdallas.edu | 11163460 |
| Mahitha Devalapalle | mxd065000@utdallas.edu | 11141826 |
| Sindhura Vallabhaneni | sxv069100@utdallas.edu | 11157441 |
| Ragavika Tarigopula | rxt058000@utdallas.edu | 11116375 |
| Nithya Bondalapati | nxb056000@utdallas.edu | 11107632 |
| Siddharth Dhall | sxd068000@utdallas.edu | 11141981 |

**Dr. Lawrence Chung**

**CS 6361 – Advanced Requirements Engineering**

**University of Texas at Dallas**

**Fall 2008**

**Table of Contents**

[1. The Process Specification 3](#_Toc216147981)

[1.1 A simple Process Model for Phase II 3](#_Toc216147982)

[1.2 Process SADT 4](#_Toc216147983)

[Roles Description: 9](#_Toc216147984)

[1.3 Relationship between Project Phase I and Phase II 10](#_Toc216147985)

[2. Issues 10](#_Toc216147986)

[2.1 Old Issues Revisited 11](#_Toc216147987)

[2.1.1 Ambiguous Issues 11](#_Toc216147988)

[2.1.2 Incomplete Issues 12](#_Toc216147989)

[2.2 New Issues Identified and Resolved 19](#_Toc216147990)

[2.2.1 Ambiguous Issues 19](#_Toc216147991)

[2.2.2 Incomplete Issues 20](#_Toc216147992)

[2.2.3 Miscellaneous Issues 20](#_Toc216147993)

[2.3 New Requirements 20](#_Toc216147994)

[3 The Product Requirements Models and Specification 21](#_Toc216147995)

[3.1 Develop business/enterprise/world/domain models, including a vision document, together with the goal models and agent models 21](#_Toc216147996)

[3.1.1 Develop a clarified definition of the SRS and specification models 21](#_Toc216147997)

[3.1.2 Establish Traceability between the two 34](#_Toc216147998)

[1.3.1 Forward Traceability 34](#_Toc216147999)

[3.1.3 Backward Traceability 36](#_Toc216148000)

[3.1.4 Features Implemented in the Prototype 38](#_Toc216148001)

[3.1.5 Traceability between System Requirements and Prototype Features 40](#_Toc216148002)

[3.1.6 Traceability between Different Scenarios and System Requirements 42](#_Toc216148003)

[4 Prototype Screen Shots 45](#_Toc216148004)

# The Process Specification

## A simple Process Model for Phase II

Refine

requirements of

Phase I

Existing

reports

Dependency

diagrams

Analyze the

process

Evaluate the

Prototype

Improvise the

Prototype

Depict the

product

specification

diagrams

Decide on the

process

specifications

Incorporate

changes in

Dependency

graph

Requirements

Engineer

Requirements

Engineer

End Users

,

Domain

Experts

,

Req Engineer

,

Software Engineer

Software Engineer

&

Requirements

Engineer

Requirements

Engineer

&

Software Architect

Requirements

Engineer

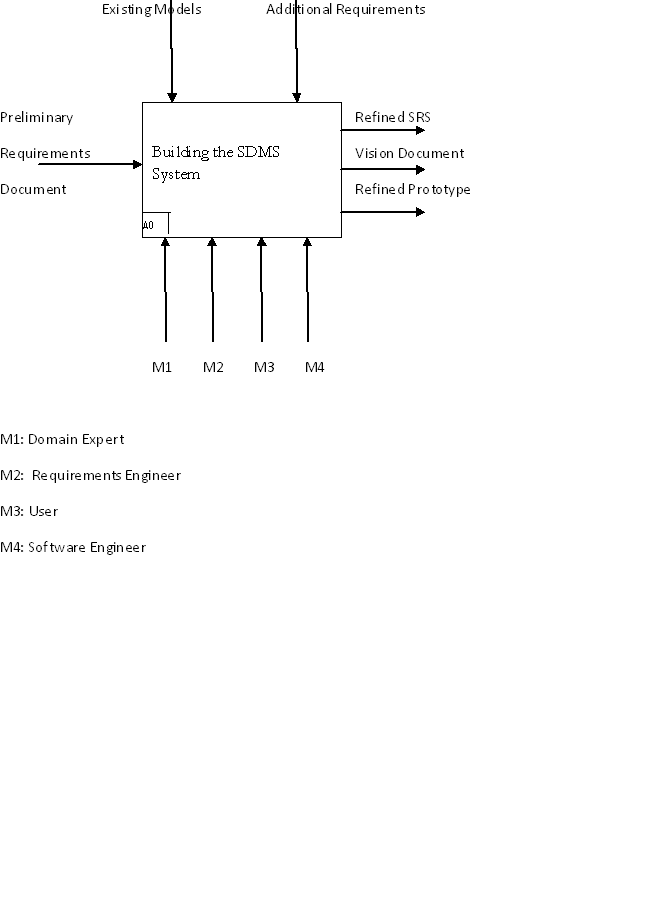
&

Software Architect

This diagram shows the complete process followed by our team through all the iterations. It involves requirements analysis, dependency graphs, process specification, product specification, Improvise the prototype, evaluate the prototype and improvise again if needed.

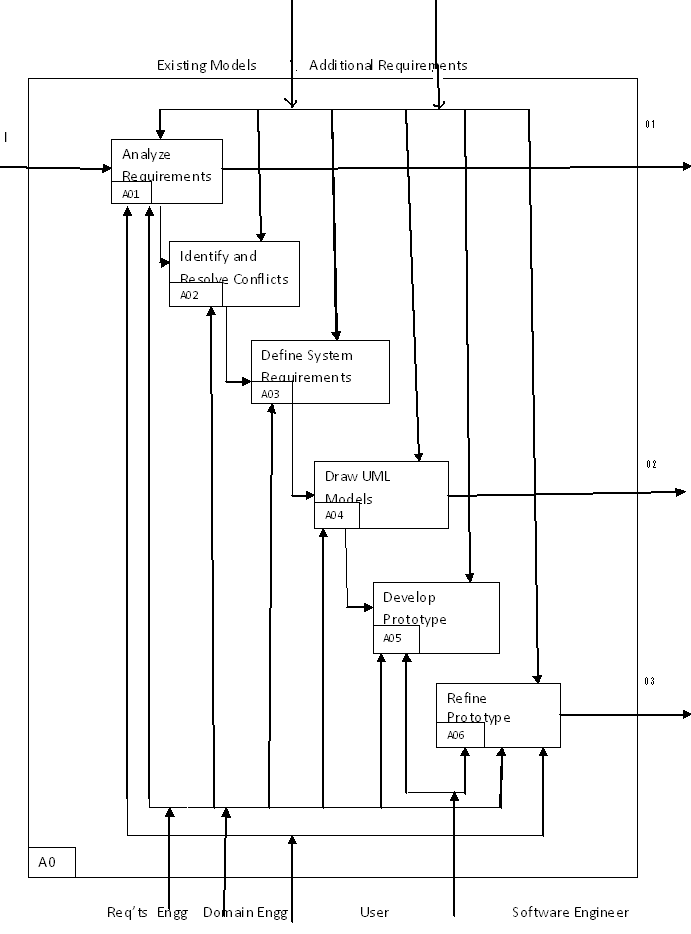
## Process SADT

#### Level 0



This diagram shows the Level 1 of the process SADT followed by our team. This level takes in the preliminary requirements document from phase I as input and gives out a refined SRS as ooutput along with a vision document and more refined prototype.

#### Level 1



I – Preliminary Requirements Document

O1 – Vision Document, O2 – Refined SRS, O3 – Refined Prototype

This is the Level 1 Process SADT followed by our team. It depicts the inner layers of the level 1 A0 process. They include A01, A02, A03, A04, A05 and A06.

#### Level 2

Existing Models Additional Requirements

Identify the Problem with Fishbone Analysis

A01

Draw Business Models and i\*

Understand Stakeholder Requests

Gather Stakeholder Requests

Preliminary

Requirements

A011

Document

A012

A013

Vision

A014

Document

User Requirements Engineer

Domain Engineer

This is the Level 1 Process SADT followed by our team. It shows the inner layers of the A01 process. Here it takes the preliminary requirements document as input and Vision document as output.

**Roles and Responsibilities:**

|  |  |  |
| --- | --- | --- |
| **Team Members** | **Roles Played** | **Primary Concern** |
| Nithya Bondalapati  Sindhura Vallabhaneni | *System & Subject World:*  End-user/Buyer  Team Leader | Requirement Analysis  Document preparation |
| Deepthi Durgempudi  Nithya Bondalapati  Sindhura Vallabhaneni  Ragavika Tarigopula | *Management World:*  Requirements Engineer  Team Leader | Process/Product Specification  Requirement Analysis  Dependency Diagrams  SADT  Goal Models and Agent Models |
| Mahitha Devalapalle  Puneeth Simha  Ragavika Tarigopula  Deepthi Durgempudi | *Developer World*  Designer, Prototype Developer | Prototype Development  UML Diagrams |
| Siddharth Dhall | *User World*  Requirements Engineer | Requirements Analysis  Deal with Requirements Issues |

## Roles Description:

* **Team Leader**: Team leader isresponsible for planning and controlling the engineering activities to meet project goals for cost, schedule and quality. She/he also responsible for co-coordinating and integrating activities across multiple functional lines.
* **Initiator:** Initiator is responsible for initiating the meeting, setting the date range , inviting participants, collecting data from the participants, feeding them into the Meeting Scheduler Software, and resolving conflicts
* **Participant:** Participant is responsible for giving the preference and exclusion date ranges, location and resource requirements and attending the meeting
* **Requirements Engineer:** Theperson responsible for eliciting, specifying and validating requirements and also developing the requirements specification document. Requirements Engineer is also the person responsible for talking with the client to accommodate any changing or new requirements of the project
* **Designer:** The person responsible for coming up with the design models such as the SADT diagrams, the UML diagrams and the cloud diagrams.
* **Prototype Developer:** The person responsible for making the design prototype operational

## Relationship between Project Phase I and Phase II

In phase I, we have analyzed the requirements given, and came up with dependency diagrams. We then identified the various issues in the requirements document and resolved some of those issues using the SIG diagram. We depicted these changes in the modified dependency diagram.

In the Phase II there was an additional requirement. It was ambiguous and we included the requirement after resolving them. Phase II of the project was an extension of the phase I. In Phase II we depicted the requirements which were analyzed in the phase I with the help of advanced notations like UML, SIG, i\* …etc. These consisted of more advanced notations and vocabulary than the dependency diagrams drawn in phase I.

We also did the fishbone analysis to identify the issues and worked on the vision document to study the environment and stakeholder requests for the SDMS. Along with the vision document, we came up with the supplementary specification and stakeholder request document.

Apart from the product specification, we also came up with the process specification. We used an SADT to show the process we have followed.

The basic prototype was built in Phase I. In phase II, after the requirements were analyzed further, we have modified the prototype to incorporate changes. The prototype in phase II was an enhancement of the prototype in the phase I.

# Issues

This section discusses the issues and the final resolutions of phase 1 and also identifies new issues and their possible solutions.

It discusses the various issues that have been identified from the Enterprise and System Functional and Non-functional requirements identified in the previous phase. The issues are categorized into incomplete, ambiguous, and miscellaneous.

Incomplete issues are derived from requirements where complete information is lacking.

Ambiguous issues are derived from requirements where multiple interpretations can be made from the same statement.

Miscellaneous issues in this document have been identified where requirements have been mentioned in the wrong category by Synergy Soft Inc.

## Old Issues Revisited

### Ambiguous Issues

***2.1.1.1 Meeting initiator will ask all potential meeting attendees for set of dates they cannot attend the meeting (exclusion sets) and the set of dates they would prefer the meeting to take place (preference sets).***

*The meaning of the term “potential attendees” is not specified.*

**Solution 1:** Potential attendees, active participants and important participants can be considered to be mutually exclusive sets of participants; where all the participants other than the active and important belong to the Potential Participants set.

**Solution 2:** Potential participants can be considered as the super set of active participants and important participants. All other participants who do not belong to either important or active category can be considered to be “optional” participants.

**Final Resolution:** The second solution is more appropriate according to the initial requirements presented by the Synergy Soft Inc. where it is specified that the meeting initiator will ask the *potential meeting attendees* to provide preferred and exclusion dates based on their agenda. Since any kind of participant will need to provide these details, we decided to opt for Solution 2.

***2.1.1.2******The exclusion and preference set should be contained in some time interval (date range) described by the initiator.***

*What does the term “some interval prescribed by the meeting initiator” mean?*

**Solution 1:** Assuming that the meeting initiator can prescribe time intervals in various ways, list out all the methods.

**Solution 2:** The only way that the initiator can prescribe time intervals is by specifying a date range. Therefore, the word “some” can be removed from this requirement.

**Final Resolution:** The exclusion and preference sets should be contained in ***the*** time interval prescribed by the meeting initiator.

***2.1.1.3******Each meeting shall also be held in the virtual place (e.g., through teleconferencing using laptop computers).***

*Does “each meeting” mean that every meeting should be accessible from a virtual location?*

**Solution 1:** “Each meeting” means that every meeting should be accessible from a virtual location.

**Solution 2:** The initiator shall decide which meeting should be accessible from a virtual location based on the location of the potential participants.

**Solution 3:** All the participants shall vote to decide whether the meeting should be accessible from a virtual location.

**Final Resolution:** Solution 1 may lead to unnecessary costs when all the participants attend the meeting in person. Solution 3 may lead to conflict in cases where half the participants vote for virtual meeting and half vote against it. Hence, solution 2 has been chosen as the optimal solution for this issue.

***2.1.1.6******SDMS shall manage concurrency when multiple requests overlap in time or space.***

*What does a meeting “overlap in time and space” refer to?*

**Solution 1:** Overlap in time and space refers to the cases where two or more meetings are contending for the same location at the same time.

**Solution 2:** Overlap in space refers to contention of location and overlap in time refers to a potential attendee being invited to two meetings scheduled during the same time period.

**Final Resolution:** The second solution has been chosen as the more appropriate solution because it closely represents the user environment.

### Incomplete Issues

***2.1.2.1******Conflicts can be resolved by:***

* ***The initiator extends the date range.***
* ***Some participants remove some dates from their exclusion set.***
* ***Some participants withdraw from meeting***
* ***Some participants add new dates to their preference set.***

*How to prioritize conflict resolution options?*

**Solution 1:** The options mentioned above should be considered in the following order irrespective of the type of the participant:

1. Some participants add some new dates to their preference set.
2. Some participants remove some dates from their exclusion set.
3. The initiator extends the date range.
4. Some participants withdraw from the meeting.

**Solution 2:** The options mentioned above should be considered in the following order irrespective of the type of the participant:

1. Some participants add some new dates to their preference set.
2. Some participants remove some dates from their exclusion set.
3. The initiator extends the date range.
4. Some participants withdraw from the meeting.

If the participant who withdraws from the meeting is an “Important” participant, then the SDMS shall cancel the meeting. If the participant is an “Active” participant, then the initiator shall be notified by the system, and he/she decide between adding a new active participant and extending the date range. If an “Optional” participant drops out, the meeting goes on normally.

**Final Resolution:** The second solution is more appropriate because the whole purpose of the meeting is lost if the intended audience (important participants) is absent from the meeting.

**Issue:**

**Important participant may not be able to attend** **the meeting which may lead to meeting cancellation**

**Participants might have already accepted the invitation for another meeting on the same date**

**The proposed meeting date may be there in the exclusion set of a participant**

**The proposed meeting date may not be present in the preference set of an active participant and he may want to withdraw**

Conflicts arising during scheduling a meeting

***Solution:***

1. **Some participants add dates to preference sets**
2. **Some remove dates from exclusion sets**
3. **Initiator extends date range**
4. **Some Participants withdraw from the meeting**

**The above options in order are for all kind of participants**

1. **Some participants add dates to preference sets**
2. **Some remove dates from exclusion sets**
3. **Initiator extends date range**

**The above three in order are for any kind of participant**

1. **Some Participants withdraw from the meeting**

**If Important participant withdraws meeting is cancelled by the system. If active one withdraws initiator shall be notified. If optional participant withdraws meeting shall go on normally.**

**Alt 2**

**Alt 1**

**\_ †**

**If Active participant withdraws care should be taken that another replaces him (i.e., initiator should be notified)**

**\_ †**

**Purpose of meeting may be lost (if Important participant i.e., intended audience withdraws)**

* + - ***Goal***
    - ***Soft Goal***
    - ***Task***
    - ***Dependency***

***2.1.2.2******Meeting room should be available at the selected meeting date.***

*How to determine availability of the meeting location on the selected date?*

**Solution 1:** The meeting Initiator should check ifa meeting room is available at the selected meeting date.

**Solution 2:** Active participants should check if a meeting room is available at the selected meeting date.

**Solution 3:** The system should have the capability to present the meeting locations based on the selected meeting date.

**Final Resolution:** Since the purpose of the meeting scheduler is to reduce the meeting initiator’s work, it is appropriate that the system should be able to resolve this issue. Hence, solution 3 has been chosen.

**Issue:**

**The room that is available on the meeting day may be smaller than required and may not be able to accommodate all the participants**

**The location of meeting room may not be convenient for the participants esp important participant**

**No room may be available on the meeting date**

Lack of availability of desired meeting room on the selected meeting date

**Solution:**

**The system should have the capability to present the meeting locations based on the selected meeting date**

**The Active Participants should check if the desired meeting room is available on the selected meeting date**

**The meeting Initiator should check if the desired meeting room is available on the selected meeting date**

**Alt 2**

**Alt 3**

**Alt 1**

**\_ \_ †**

**Efficient option**

**\_ † †**

**Reduce the burden on meeting initiator**

* + - ***Goal***
    - ***Soft Goal***
    - ***Task***
    - ***Dependency***

***2.1.2.3******Meeting room should meet the equipment requirements.***

*How to determine whether the meeting location meets the equipment requirements or not?*

**Solution 1:** The meeting Initiator should check ifa selected meeting room accommodates the required (requested) resources.

**Solution 2:** Active participants should check if a selected meeting room accommodates the required (requested) resources.

**Solution 3:** The system should have the capability to present the meeting locations based on the resources requested and the meeting date.

**Final Resolution:** Since the purpose of the meeting scheduler is to reduce the meeting initiator’s work, it is appropriate that the system should be able to resolve this issue. Hence, solution 3 has been chosen.

***2.1.2.4******SDMS shall re-plan a meeting to support the changing user constraints (preferred set, exclusion set, preferred location, etc...)***

*How to re-plan a meeting when some participants opt out of the meeting?*

Solution: Refer 2.1.2.1

***2.1.2.5******SDMS shall re-plan a meeting to support changing external constraints (need to accommodate a more important meeting)***

*How to re-plan a meeting when another more important meeting needs to utilize the same location at the same time?*

**Solution 1:** The meeting initiator shall be notified about the situation, and he/she shall decide whether to change the location or re-schedule the meeting.

**Solution 2:** The SDMS system automatically changes the location based on the dates and the resources required, and notifies the meeting initiator and the potential participants

**Final Resolution:** Though the second solution relieves the meeting initiator of extra work, the location chosen by the system might not be convenient to all the participants. Therefore, Solution 1 has been chosen to resolve this issue.

***2.1.2.7******SDMS shall make the participants aware of what is going on during the planning process.***

*How should the system keep the user aware of the planning process?*

**Solution 1:** Any changes in the meeting schedule, description, requirements, etc. shall be communicated to the users via e-mail.

**Solution 2:** Any changes in the meeting schedule, description, requirements, etc. shall be presented to the users when they access their SDMS accounts.

**Solution 3:** Any changes in the meeting schedule, description, requirements, etc. shall be communicated to the user via e-mail, and also be presented when they access their accounts.

**Final Resolution:** There is a chance that the user might not access one of his accounts (email and SDMS) regularly. Since it has to be ensured that the user should be kept notified, solution 3 is more fool-proof compared to the other two solutions. Therefore, it has been chosen as the optimal solution for this issue.

## New Issues Identified and Resolved

### Ambiguous Issues

***2.2.1.1******SDMS shall monitor meetings, especially when they are held in a distributed manner.***

*What does the term “monitor meetings” mean?*

**Solution 1**: Record the entire meeting in video format.

**Solution 2**: Record the entire meeting in audio format.

**Solution 3**: Keep a record of meeting minutes.

**Final Resolution**: The first two solutions are out of the control of SDMS. Moreover, the meeting initiator can request for recording equipment while initiating the meeting (or later) if required. Therefore, the third solution has been chosen as the optimal one.

***2.2.1.2******SDMS shall handle several meetings in parallel.***

*What does “meetings in parallel” refer to?*

**Solution 1:** The initiator should be able to log in to multiple sessions of SDMS and initiate a meeting from each session.

**Solution 2:** The system shall provide the initiator with the option of scheduling multiple meetings at the same time by allowing him/her to enter the details for all the meetings in a single form and submitting at once.

**Final resolution:** The second solution has been chosen as the optimal solution because such a functionality would be beneficial (time and effort) to users who have to initiate multiple meetings in a day.

### Incomplete Issues

***2.2.2.1******SDMS shall re-plan a meeting to support changing external constraints (need to accommodate a more important meeting)***

*How to determine if one meeting is important than the other?*

**Solution 1:** The system shall accept an “importance rating” along with other meeting details from the meeting initiator while scheduling a meeting. When two meetings contend for the same location at the same time, the system shall determine which is more important based on this “importance rating”.

**Solution 2:** When there is contention for meeting location, the system shall notify the meeting initiators of both the meetings, who can then negotiate external to the system to decide which meeting should be given preference.

**Final Resolution:** The first solution can be misleading, because the “importance rating” will be based on the perception of the meeting initiator and cannot be considered as a standard. Therefore, the second solution has been chosen as the optimal resolution for this issue.

### Miscellaneous Issues

***2.2.3.1******SDMS shall accommodate decentralized requests i.e., allow any authorized user to request a meeting independent of their location.***

**Solution:** The system shall allow the users to access their accounts irrespective of their location, as it shall be made available online.

***2.2.3.2******SDMS shall have a fixed lower bound between the time at which the meeting date is determined and the time at which the meeting is actually taking place.***

**Solution:** Since it is highly possible in the dynamic world that a meeting for the same day might need to be scheduled; this requirement is not a valid one. After presenting this view to the stakeholders from SynergySoft Inc., it has been decided that this requirement shall not be considered.

## 2.3 New Requirements

This section deals with the issues faced in accommodating the new requirements.

***2.3.1 Some meetings are organized and scheduled at the same time, as a chunk, where partial attendance can be allowed.***

*How to allow partial attendance?*

**Solution 1:** Participants can specify to the meeting initiator that they will be attending only part of the meeting by adding comments along with acceptance of a meeting invitation.

**Solution 2:** Do not allow partial attendance at all

**Final Resolution:** Since it is possible that a person wants to go to both the meetings scheduled at the same time, solution 1 has been chosen as the appropriate option.

# The Product Requirements Models and Specification

## 3.1 Develop business/enterprise/world/domain models, including a vision document, together with the goal models and agent models

The business models, the goal models and the agents models are included in the vision document which is provided as a separate supplement along with this document.

## Develop a clarified definition of the SRS and specification models

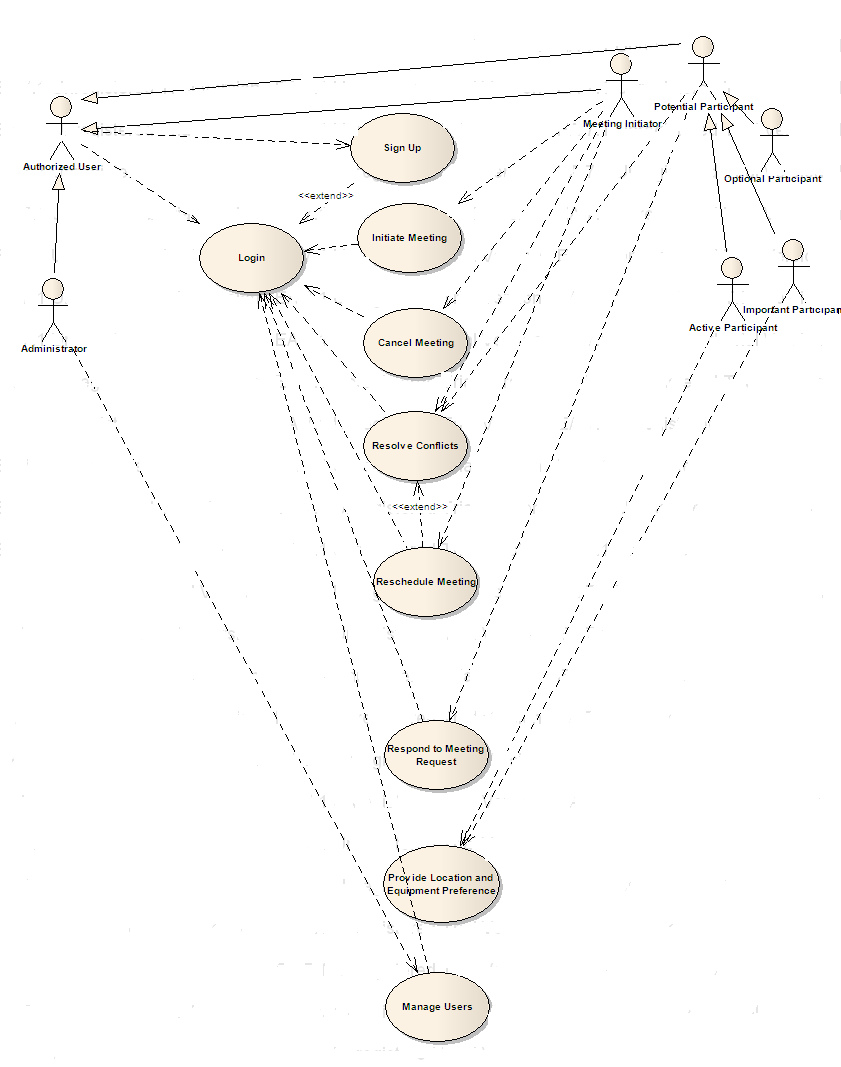
1. **Refined Software Requirements Specification**

The refined SRS is provided as a separate supplement along with this document.

1. **Specification Models**

The specification models included here are:

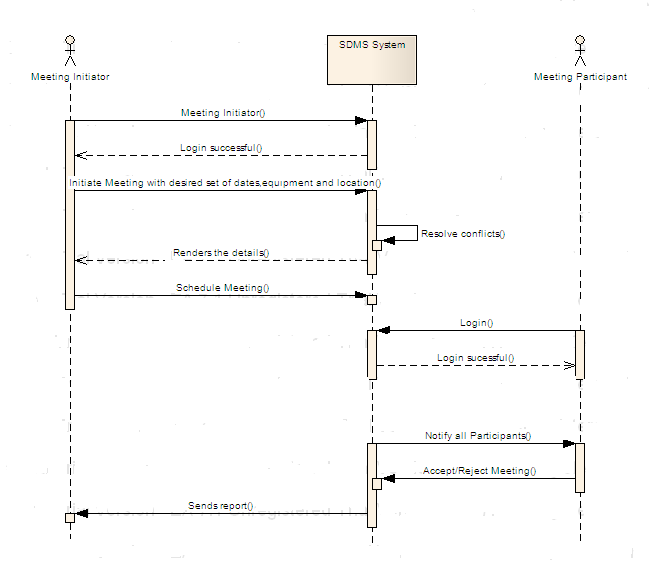
#### System Use Case Diagram



SDMS SYSTEM

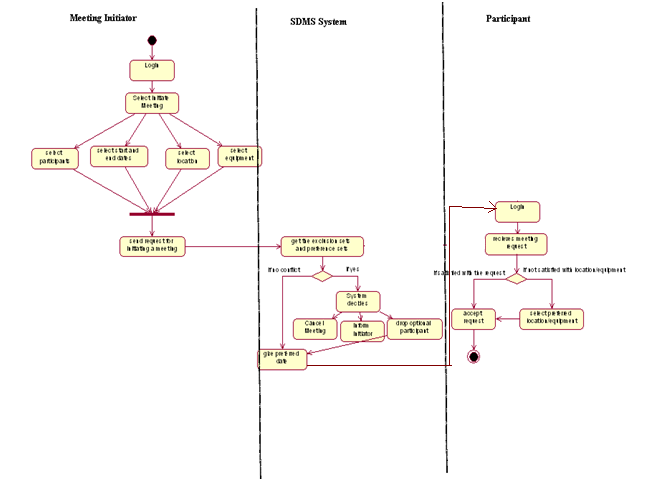
This diagram shows the use cases and the actors associated with the use cases.

#### System Sequence Diagram



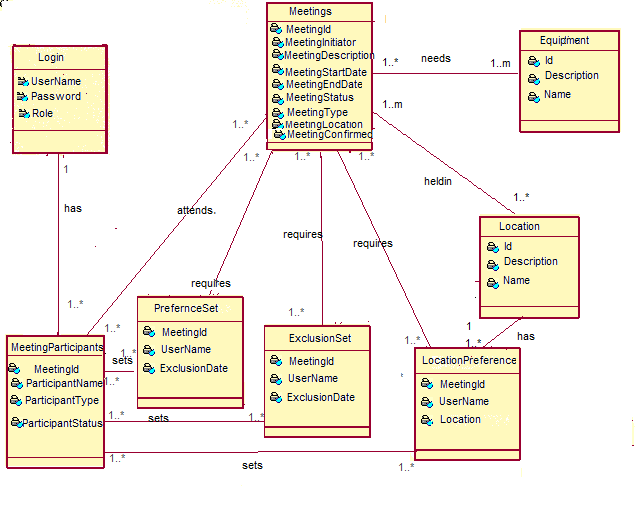
This diagram is the system sequence diagram which shows the complete flow of the system in a high level perspective.

#### System Activity Diagram



This diagram shows the system activity diagram, which shows the activities between the Meeting Initiator, Participant and the System.

#### System Class Diagram

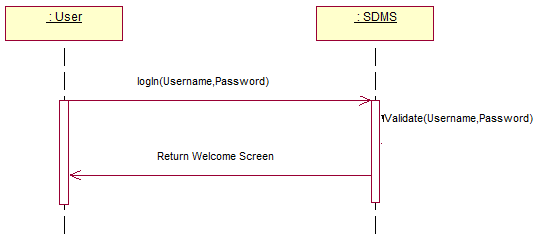


This diagram is the domain model which shows the identified classes and the relationship between the classes along with their multiplicity. This class diagram shows that :

* Each meeting is associated to one to many participants, preference sets, exclusion sets, Location preference, location and equipment.
* Each meeting participant has exactly one login and each login is has one to many participants.
* Each preference set, exclusion set, location preference, location and equipment are associated to one to many meetings.
* Each meeting participant has one to many preference sets, exclusion sets and location preference.
* Each location preference has exactly one location.

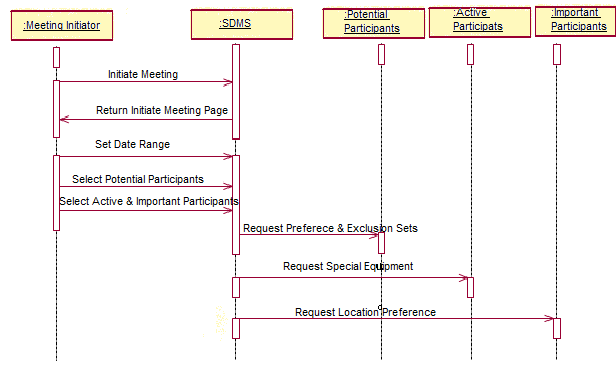
#### Sequence Diagrams

**Login:**



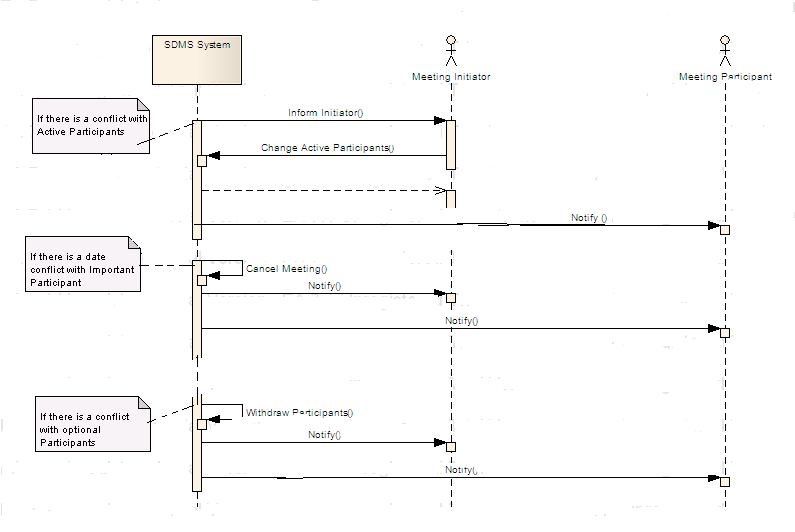
This sequence diagram shows the sequence of events in a Login Usecase. Here a valid user name and password is entered into the system. Then the SDMS system validates them. If they are validated successfully, the system shows the welcome screen to the user.

**Initiate Meeting:**



This diagram shows the sequence of events for the Initiate Meeting Use Case. Here When a meeting initiator select Initiate Meeting, an Initiate Meeting page is returned to him. Then he selects the date range, location, equipment, potential participants, assign roles to them and clicks submit. Then the system takes the exclusion set and preference sets from the participants calendars and calculates a desired meeting date.

**Resolve Conflict**

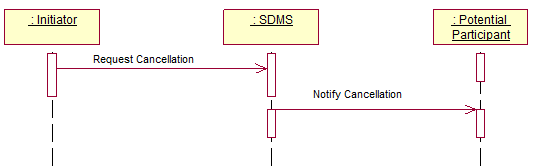


This diagram shows the three scenarios for conflict resolution in our meeting scheduler.

The three scenarios include:

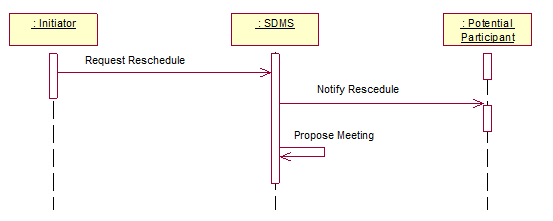
* Inform Initiator: When there is a conflict with an active participant, the system informs the initiator and he decides on changing the active participant.
* Cancel Meeting: When there is a conflict with the important participant, the system cancels the meeting.
* Withdraw Participants: When there is a conflict with optional participants, the system withdraws the participant.

**Cancel Meeting**:



This diagram shows the sequence of events needed to cancel a meeting. The initiator sends a request to the system to cancel the meeting. The system then notifies to all the potential participants.

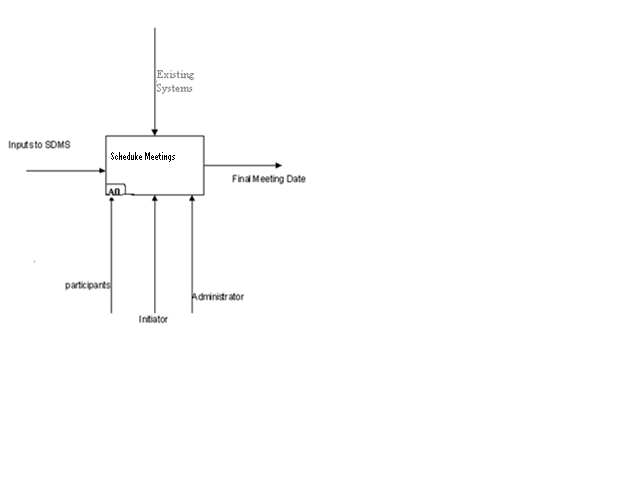
**Reschedule Meetings:**



#### This diagram shows the sequence of events required to reschedule a meeting. Whena request for rescheduling a meeting is sent to SDMS, the SDMS notifies the potential participants and tries to propose a new meeting date.

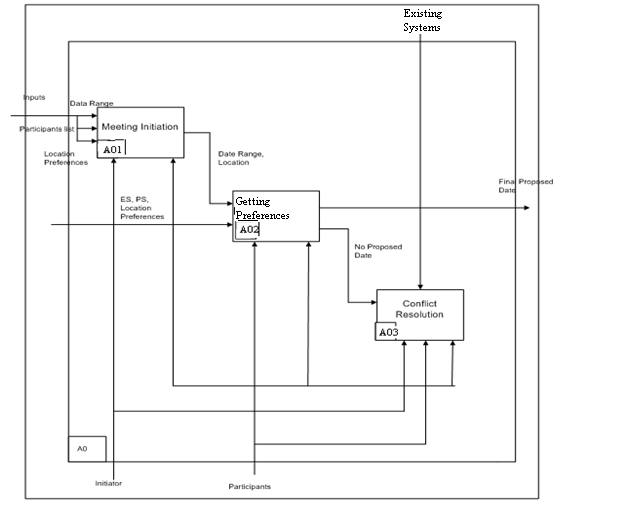
#### Product SADT

**Level 0**



This diagram shows the product SADT for the Level 1 – A0. The SDMS system takes some input and gives out the final meeting date.

**Level 1**

****

This diagram shows the product SADT of the inner layers of A0. The inner layers of A0 are Meeting Initiation, Preferences and Conflict Resolution.

## Establish Traceability between the two

### Forward Traceability

The table below is used to trace which product features satisfy the refined requirements that have been derived from the initial Enterprise and System requirements. The requirements IDs correspond to the identifiers of the requirements in the supplementary SRS supplied with this document. The product feature IDs are as per the supplementary Vision Document that accompanies this document.

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement ID | Requirement | Product Feature Ids | |
| 3.2.1.1 | SDMS shall allow the meeting initiator to initiate a meeting under constraints expressed by the participants and with the specified requirements | | 12,17,18,19 |
| 3.2.1.2 | SDMS shall allow the meeting initiator to select the participants for his meeting | | 6,24 |
| 3.2.1.3 | SDMS shall allow the meeting initiator to categorize the participants as important, active or optional | | 8 |
| 3.2.1.4 | SDMS shall allow the meeting initiator to select a date range within which the meeting will be scheduled | | 9 |
| 3.2.1.5 | SDMS shall allow the meeting initiator to select a meeting location from the list of available locations based on meeting date and resources required. | | 10,11,12 |
| 3.2.1.6 | SDMS shall allow the meeting initiator to request resources like projectors, DVD players...etc for the meeting. | | 11 |
| 3.2.1.7 | SDMS shall get the inclusion set and exclusion set of dates from the participants calendars | | 12 |
| 3.2.1.8 | After getting the preferred dates from all participants, the SDMS shall choose a preferred date which is convenient to all. | | 12 |
| 3.2.1.9 | SDMS shall send meeting invitations to all the participants after the meeting date is confirmed. | | 14 |
| 3.2.1.10 | SDMS shall re-plan a meeting to support changing external constraints | | 15,30,26,33 |
| 3.2.1.11 | SDMS shall allow the meeting initiator to schedule multiple meetings at the same time. | | 7 |
| 3.2.2.1 | SDMS shall allow the potential participant to respond to a meeting request from the initiator without comments | | 16,20 |
| 3.2.2.2 | SDMS shall allow a participant to add comments along with their acceptance to let the initiator know if they can only make a partial attendance to the meeting | | 17 |
| 3.2.2.3 | SDMS shall allow only important participants to suggest their location preferences while accepting a meeting request. | | 19 |
| 3.2.2.4 | SDMS shall allow only active participants to request additional resources while accepting a meeting request. | | 18 |
| 3.2.2.5 | SDMS shall allow a potential attendee to reject a meeting invite if he is not interested to attend the meeting | | 20 |
| 3.2.3.1 | SDMS shall send an update to all the participants whenever there is any change in the meeting request. | | 26 |
| 3.2.3.2 | SDMS shall allow the meeting initiator to change the location of the meeting based on the preference requests from the important participants | | 19,21 |
| 3.2.3.3 | SDMS shall allow the meeting initiator to reschedule a meeting in the event of irresolvable conflicts. | | 15,22,13 |
| 3.2.3.4 | SDMS shall allow the meeting initiator to add more details to the meeting description. | | 23 |
| 3.2.3.5 | SDMS shall allow the meeting initiator to add more participants at any time before the meeting. | | 24 |
| 3.2.3.6 | SDMS shall allow the meeting initiator to remove existing participants at any time before the meeting. | | 24 |
| 3.2.3.7 | SDMS shall allow the meeting initiator to add more equipment resources to the meeting, based on requests from the active participants. | | 18,25 |
| 3.2.3.8 | SDMS shall be capable of cancelling a meeting by itself (based on a pre-defined criteria) or by allowing the meeting initiator to cancel it. | | 31,32 |
| 3.2.4.1 | SDMS shall allow the administrator to accept a user’s request to sign up into the system. | | 34 |
| 3.2.4.2 | SDMS shall allow the administrator to unlock the user accounts which have been locked due to more than three attempts of incorrect password. | | 35 |
| 3.2.4.3 | SDMS shall allow the administrator to disable a user’s account which is temporarily not in use. | | 36 |
| 3.2.4.4 | SDMS shall allow the administrator to enable (activate) a user’s account when it is needed again. | | 37 |
| 3.2.4.5 | SDMS shall allow the administrator to delete the user accounts which are no longer in use. | | 38 |
| 3.2.5.1 | A user should be able to update his calendar with exclusion set and preference set. | | 30 |
| 3.2.6.1 | • If there is a date conflict with an important participant, the system cancels the meeting and asks the initiator to re-plan the meeting.  • If there is a date with an active participant, the system will inform the initiator about it. The initiator can decide on whether changing the important participant or re-schedule the meeting.  • If there is a date conflict with an optional participant, the system automatically withdraws him from the meeting. | | 31,32,33 |
| 3.2.7.1 | The system shall send meeting remainders to all the accepted meeting attendees 30 minutes before the meeting. | | 28 |
| 3.2.8.1 | SDMS shall allow the potential attendees of the meeting to upload a summary of the meeting or any concerning documents of the meeting. | | 27 |
| 3.3.2.9 | The system shall be available 24X7 to schedule meetings. | | 39 |
| 3.3.2.10 | The system shall be secure. | | 1,2,3,4,5 |

### Backward Traceability

The table below is used to trace which requirements are satisfied by the various features supported by the product – SDMS. The requirements IDs correspond to the identifiers of the requirements in the supplementary SRS supplied with this document. The product feature IDs are as per the supplementary Vision Document that accompanies this document.

|  |  |  |
| --- | --- | --- |
| Product Feature ID | Product Features | Requirement IDs |
| 1 | Login to the system | 3.3.2.10 |
| 2 | Sign Up for an account | 3.3.2.10 |
| 3 | Logout of the system | 3.3.2.10 |
| 4 | Change Password | 3.3.2.10 |
| 5 | Lock user account | 3.3.2.10 |
| 6 | Select participants for a meeting | 3.2.1.2 |
| 7 | Multiple meeting scheduling | 3.2.1.11 |
| 8 | Set participant category | 3.2.1.3 |
| 9 | Select “Date range” for meeting | 3.2.1.4 |
| 10 | Select “Location” for the meeting | 3.2.1.5 |
| 11 | Select “Resources” for the meeting | 3.2.1.5, 3.2.1.6 |
| 12 | Get Preferred Dates of all participants | 3.2.1.1, 3.2.1.5, 3.2.1.7, 3.2.1.8 |
| 13 | Confirm a specific meeting date from the “Date Range” | 3.2.3.3 |
| 14 | Send invites to participants | 3.2.1.9 |
| 15 | Re-plan a meeting | 3.2.1.10, 3.2.3.3 |
| 16 | Accept a meeting invite (without comments) | 3.2.2.1 |
| 17 | Accept a meeting invite (with comments) | 3.2.1.1, 3.2.2.2 |
| 18 | Request additional resources for the meeting | 3.2.1.1, 3.2.2.4, 3.2.3.7 |
| 19 | Suggest preferred location | 3.2.1.1, 3.2.2.3, 3.2.3.2 |
| 20 | Reject a meeting invite | 3.2.2.1, 3.2.2.5 |
| 21 | Change meeting Location | 3.2.3.2 |
| 22 | Change meeting Date | 3.2.3.3 |
| 23 | Add meeting Details (agenda) | 3.2.1.10, 3.2.3.4 |
| 24 | Add new participants or remove existing participants | 3.2.1.2, 3.2.3.5, 3.2.3.6 |
| 25 | Add resources to the meeting | 3.2.3.7 |
| 26 | Notify participants about any changes | 3.2.1.10, 3.2.3.1 |
| 27 | Upload meeting minutes | 3.2.8.1 |
| 28 | Email meeting reminders | 3.2.7.1 |
| 29 | Notify initiators about location contentions | 3.2.5.1 |
| 30 | Update Calendar | 3.2.1.10 |
| 31 | If there is a date conflict with an important participant, the system cancels the meeting and asks the initiator to re-plan the meeting. | 3.2.3.8, 3.2.6.1 |
| 32 | If there is a date with an active participant, the system will inform the initiator about it. The initiator can decide on whether changing the important participant or re-schedule the meeting. | 3.2.3.8, 3.2.6.1 |
| 33 | If there is a date conflict with an optional participant, the system automatically withdraws him from the meeting and notifies him. | 3.2.6.1 |
| 34 | Accept the request for a New User account | 3.2.4.1 |
| 35 | Unlock user accounts | 3.2.4.2 |
| 36 | Disable User Accounts | 3.2.4.3 |
| 37 | Enable User Accounts | 3.2.4.4 |
| 38 | Delete user account | 3.2.4.5 |
| 39 | Online Access | 3.3.2.9 |

### Features Implemented in the Prototype

|  |  |  |
| --- | --- | --- |
| Prototype ID | Product Features | Status |
| 1 | Login to the system | Completed |
| 2 | Sign Up for an account | Completed |
| 3 | Logout of the system | Completed |
| 4 | Change password | Completed |
| 5 | Lock user account | Completed |
| 6 | Select participants for a meeting | Completed |
| 7 | Multiple meeting scheduling | out of scope for current implementation |
| 8 | Set participant category | Completed |
| 9 | Select “Date range” for meeting | Completed |
| 10 | Select “Location” for the meeting | Completed - show all locations |
| 11 | Select “Resources” for the meeting | Completed- show all resources |
| 12 | Get Preferred Dates of all participants | At present being done manually |
| 13 | Confirm a specific meeting date from the “Date Range” | Completed |
| 14 | Send invites to participants | Completed |
| 15 | Re-plan a meeting | Partially completed |
| 16 | Accept a meeting invite (without comments) | Completed |
| 17 | Accept a meeting invite (with comments) | Completed |
| 18 | Request additional resources for the meeting | Completed |
| 19 | Suggest preferred location | Completed |
| 20 | Reject a meeting invite | Completed |
| 21 | Change meeting Location | Completed |
| 22 | Change meeting Date | out of scope for current implementation |
| 23 | Add meeting Details (agenda) | Completed |
| 24 | Add new participants or remove existing participants | out of scope for current implementation |
| 25 | Add resources to the meeting | Completed |
| 26 | Notify participants about any changes | Completed |
| 27 | Upload meeting minutes | Completed |
| 28 | Email meeting reminders | out of scope for current implementation |
| 29 | Notify initiators about location contentions | out of scope for current implementation |
| 30 | Update Calendar | Completed |
| 31 | If there is a date conflict with an important participant, the system cancels the meeting and asks the initiator to re-plan the meeting. | At present being done manually |
| 32 | If there is a date conflict with an active participant, the system will inform the initiator about it. The initiator can decide on whether changing the active participant or re-schedule the meeting. | At present being done manually |
| 33 | If there is a date conflict with an optional participant, the system automatically withdraws him from the meeting | At present being done manually |
| 34 | Accept the request for a New User account | Completed |
| 35 | Unlock user accounts | Completed |
| 36 | Disable User Accounts | Completed |
| 37 | Enable User Accounts | Completed |
| 38 | Delete user account | Completed |
| 39 | Online Access | out of scope for current implementation |

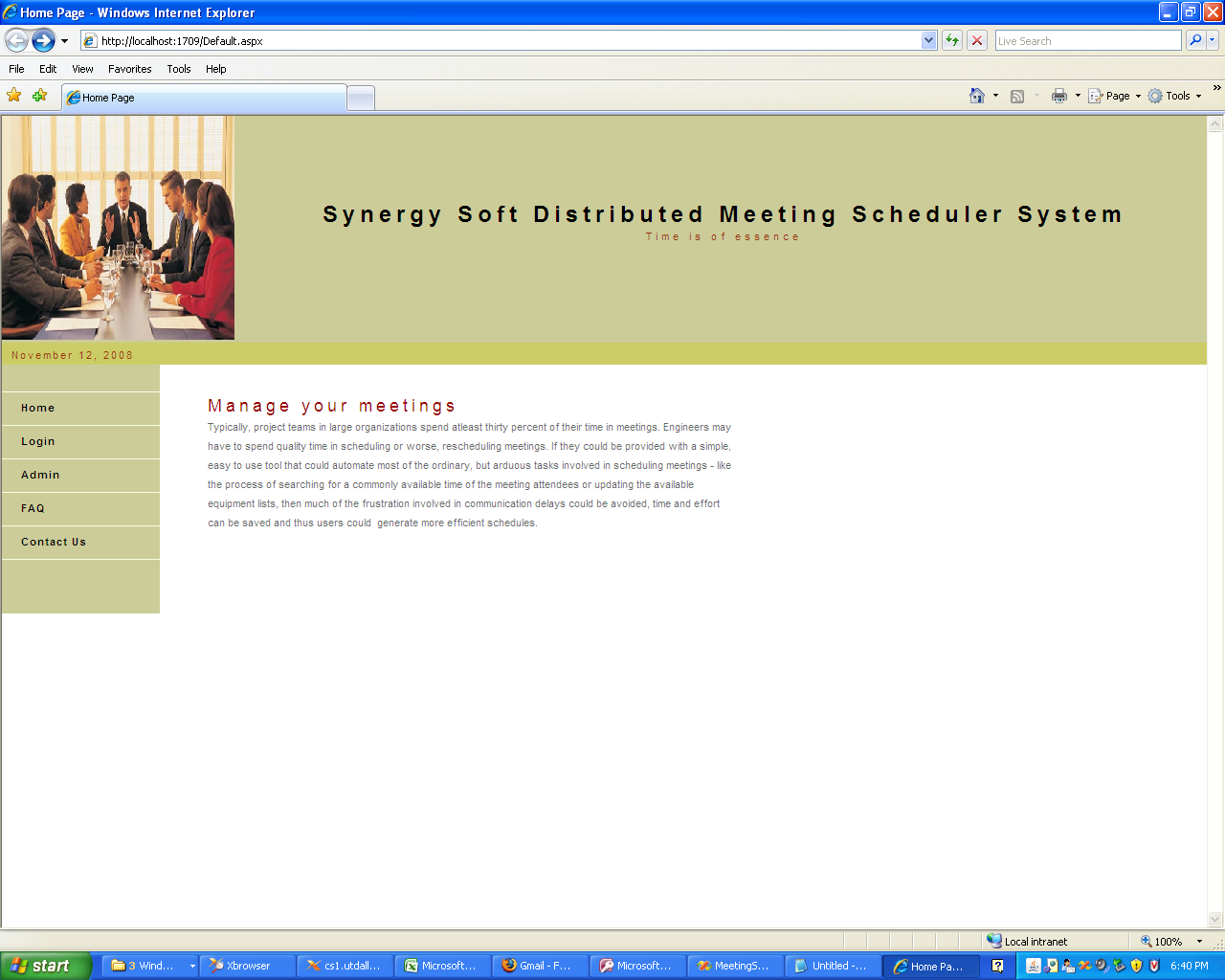
### Traceability between System Requirements and Prototype Features

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement ID | Requirement | Prototype ID | |
| 3.2.1.1 | SDMS shall allow the meeting initiator to initiate a meeting under constraints expressed by the participants and with the specified requirements | | 12,17,18,19 |
| 3.2.1.2 | SDMS shall allow the meeting initiator to select the participants for his meeting | | 6,24 |
| 3.2.1.3 | SDMS shall allow the meeting initiator to categorize the participants as important, active or optional | | 8 |
| 3.2.1.4 | SDMS shall allow the meeting initiator to select a date range within which the meeting will be scheduled | | 9 |
| 3.2.1.5 | SDMS shall allow the meeting initiator to select a meeting location from the list of available locations based on meeting date and resources required. | | 10,11,12 |
| 3.2.1.6 | SDMS shall allow the meeting initiator to request resources like projectors, DVD players...etc for the meeting. | | 11 |
| 3.2.1.7 | SDMS shall get the inclusion set and exclusion set of dates from the participants calendars | | 12 |
| 3.2.1.8 | After getting the preferred dates from all participants, the SDMS shall choose a preferred date which is convenient to all. | | 12 |
| 3.2.1.9 | SDMS shall send meeting invitations to all the participants after the meeting date is confirmed. | | 14 |
| 3.2.1.10 | SDMS shall re-plan a meeting to support changing external constraints | | 15,30,26,33 |
| 3.2.1.11 | SDMS shall allow the meeting initiator to schedule multiple meetings at the same time. | | 7 |
| 3.2.2.1 | SDMS shall allow the potential participant to respond to a meeting request from the initiator without comments | | 16,20 |
| 3.2.2.2 | SDMS shall allow a participant to add comments along with their acceptance to let the initiator know if they can only make a partial attendance to the meeting | | 17 |
| 3.2.2.3 | SDMS shall allow only important participants to suggest their location preferences while accepting a meeting request. | | 19 |
| 3.2.2.4 | SDMS shall allow only active participants to request additional resources while accepting a meeting request. | | 18 |
| 3.2.2.5 | SDMS shall allow a potential attendee to reject a meeting invite if he is not interested to attend the meeting | | 20 |
| 3.2.3.1 | SDMS shall send an update to all the participants whenever there is any change in the meeting request. | | 26 |
| 3.2.3.2 | SDMS shall allow the meeting initiator to change the location of the meeting based on the preference requests from the important participants | | 19,21 |
| 3.2.3.3 | SDMS shall allow the meeting initiator to reschedule a meeting in the event of irresolvable conflicts. | | 15,22,13 |
| 3.2.3.4 | SDMS shall allow the meeting initiator to add more details to the meeting description. | | 23 |
| 3.2.3.5 | SDMS shall allow the meeting initiator to add more participants at any time before the meeting. | | 24 |
| 3.2.3.6 | SDMS shall allow the meeting initiator to remove existing participants at any time before the meeting. | | 24 |
| 3.2.3.7 | SDMS shall allow the meeting initiator to add more equipment resources to the meeting, based on requests from the active participants. | | 18,25 |
| 3.2.3.8 | SDMS shall be capable of cancelling a meeting by itself (based on a pre-defined criteria) or by allowing the meeting initiator to cancel it. | | 31,32 |
| 3.2.4.1 | SDMS shall allow the administrator to accept a user’s request to sign up into the system. | | 34 |
| 3.2.4.2 | SDMS shall allow the administrator to unlock the user accounts which have been locked due to more than three attempts of incorrect password. | | 35 |
| 3.2.4.3 | SDMS shall allow the administrator to disable a user’s account which is temporarily not in use. | | 36 |
| 3.2.4.4 | SDMS shall allow the administrator to enable (activate) a user’s account when it is needed again. | | 37 |
| 3.2.4.5 | SDMS shall allow the administrator to delete the user accounts which are no longer in use. | | 38 |
| 3.2.5.1 | A user should be able to update his calendar with exclusion set and preference set. | | 30 |
| 3.2.6.1 | • If there is a date conflict with an important participant, the system cancels the meeting and asks the initiator to re-plan the meeting.  • If there is a date with an active participant, the system will inform the initiator about it. The initiator can decide on whether changing the important participant or re-schedule the meeting.  • If there is a date conflict with an optional participant, the system automatically withdraws him from the meeting. | | 31,32,33 |
| 3.2.7.1 | The system shall send meeting remainders to all the accepted meeting attendees 30 minutes before the meeting. | | 28 |
| 3.2.8.1 | SDMS shall allow the potential attendees of the meeting to upload a summary of the meeting or any concerning documents of the meeting. | | 27 |
| 3.3.2.9 | The system shall be available 24X7 to schedule meetings. | | 39 |
| 3.3.2.10 | The system shall be secure. | | 1,2,3,4,5 |

### Traceability between Different Scenarios and System Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Scenario | Requirement ID | |
| Administrator Login - Success | * Administrator clicks on the Admin Login * Administrator fills in the password * Administrator clicks on the login button * Administrator is successfully logged | | 3.2.2.9.1 |
| Administrator Login - Failure | * Administrator clicks on the Admin Login * Administrator leaves the password blank * Administrator clicks on the login button | | 3.2.2.9.1 |
| Administrator Login - Failure | * Administrator clicks on the Admin Login * Administrator fills in wrong password * Administrator clicks on the login button | | 3.2.2.9.1 |
| Administrator View Contacts - Success | * After logging in, administrator clicks on View Contacts. * Administrator is successfully able to see all the users already registered. | | 3.2.4.6 |
| Administrator Register Contact - Success | * After logging in, administrator clicks on View Users. * Administrator sees a list of users pending approvals. * Administrator sets Locked to False. * User is successfully registered. | | 3.2.4.1 |
| Administrator Edit Contact –  Success | * Administrator logs in, clicks on Edit/Delete Contacts. * Administrator chooses the contact to be edited. * Administrator makes the required changes to the contact information. * Administrator clicks on the edit button. * The contact information is successfully edited. | | 3.2.4.7 |
| Administrator Delete Contact –  Success | * Administrator logs in, clicks on Edit/Delete Contacts. * Administrator chooses the contact to delete. * Administrator click on the Delete button. * The contact is successfully deleted. | | 3.2.4.5 |
| Administrator Log out –  Success | * Administrator is already logged in. * Administrator clicks on Log Out. * Administrator successfully logs out. | | 3.2.2.9.3 |
| New User Login -  Success | * User click on the “New User Sign up for new Account”. * User fills in the detailed information. * User clicks on the Submit button. * User account is successfully set up. | | 3.2.2.9.2 |
| Existing User Login -  Success | * User clicks on Login. * User fills in the username and password. * User is successfully logged in and reaches the welcome screen. | | 3.2.2.9.1 |
| View Invitations -  Success | * User successfully logs on. * User clicks on the inbox. * User successfully views the invitations, if any. | | 3.2.2.6 |
| View Initiated Meetings -  Success | * User successfully logs on. * User successfully views the meetings initiated by him, if any. | | 3.2.2.5 |
| View Confirmed meetings -  Success | * User successfully logs on. * User clicks on the inbox. * User successfully views the confirmed meetings, if any. | | 3.2.2.7 |
| View meeting details  - Success | * User successfully logs in. * User clicks on Minutes of Meeting. * User successfully views the minutes of the meetings. | | 3.2.8 |
| View Address Book - Success | * User successfully logs in. * User clicks on Address Book. * User successfully views all the contacts currently registered. | | 3.2.9 |
| User Initiates meeting – Success | * User successfully logs in. * User clicks on Initiate meeting. * User fills in the required details of the meeting. * User clicks on the Send Request button. * Meeting is successfully initiated. | | 3.2.1 |
| Initiator Confirms meeting - Success | * User logs in * User clicks on a meeting from the “View Initiated meetings” section * Users sees responses from invitees (all accepted) * User clicks on “Get confirmed Date” * A confirmed meeting date is given | | 3.2.2.9 |
| Initiator Confirms meeting – Failure1 | * User logs in * User clicks on a meeting from the “View Initiated meetings” section * User sees responses from invitees(some rejected) * User chooses whether to confirm meeting with accepted participants or to cancel it based on their importance to the meeting. | | 3.2.2.8, 3.2.1.9 |
| Initiator Confirms meeting – Failure2 | * User logs in * User clicks on a meeting from the “View Initiated meetings” section * User sees responses from invitees(all accepted, date conflict) * User clicks on “Get confirmed Date” * User sees a message showing the participants whose dates are conflicting | | 3.2.1.8,3.2.1.9 |
| User Responds to Meeting Invites | * User logs in * User Views Invitations in Inbox * User selects preference and exclusion dates from date range * User responds to meeting by filling the comment box where he can specify location requests or equipment requests or if he wants to attend partially. | | 3.2.1.7 |
| User Initiates meeting - Failure | * User successfully logs in. * User clicks on Initiate meeting. * User does not fill all the details required for initiating a meeting. * User clicks on the Send Request button. * Error message is displayed | | 3.2.1 |
| User Initiates meeting - Failure | * User successfully logs in. * User clicks on Initiate meeting. * User fills in with error formats of the required details * User clicks on the Send Request button. * Validation Error message is displayed | | 3.2.1 |
| User updates calendar -  Success | * User successfully logs in. * User clicks on Update Calendar. * User selects the preferred and the exclusion dates. * User clicks on Update Calendar. * The calendar is successfully updated. | | 3.2.5 |
| Log Out -  Success | * User is already successfully logged in. * User clicks on Log Out. * User successfully logs out. | | 3.2.2.9.3 |

# Prototype Screen Shots



Initial Welcome Page



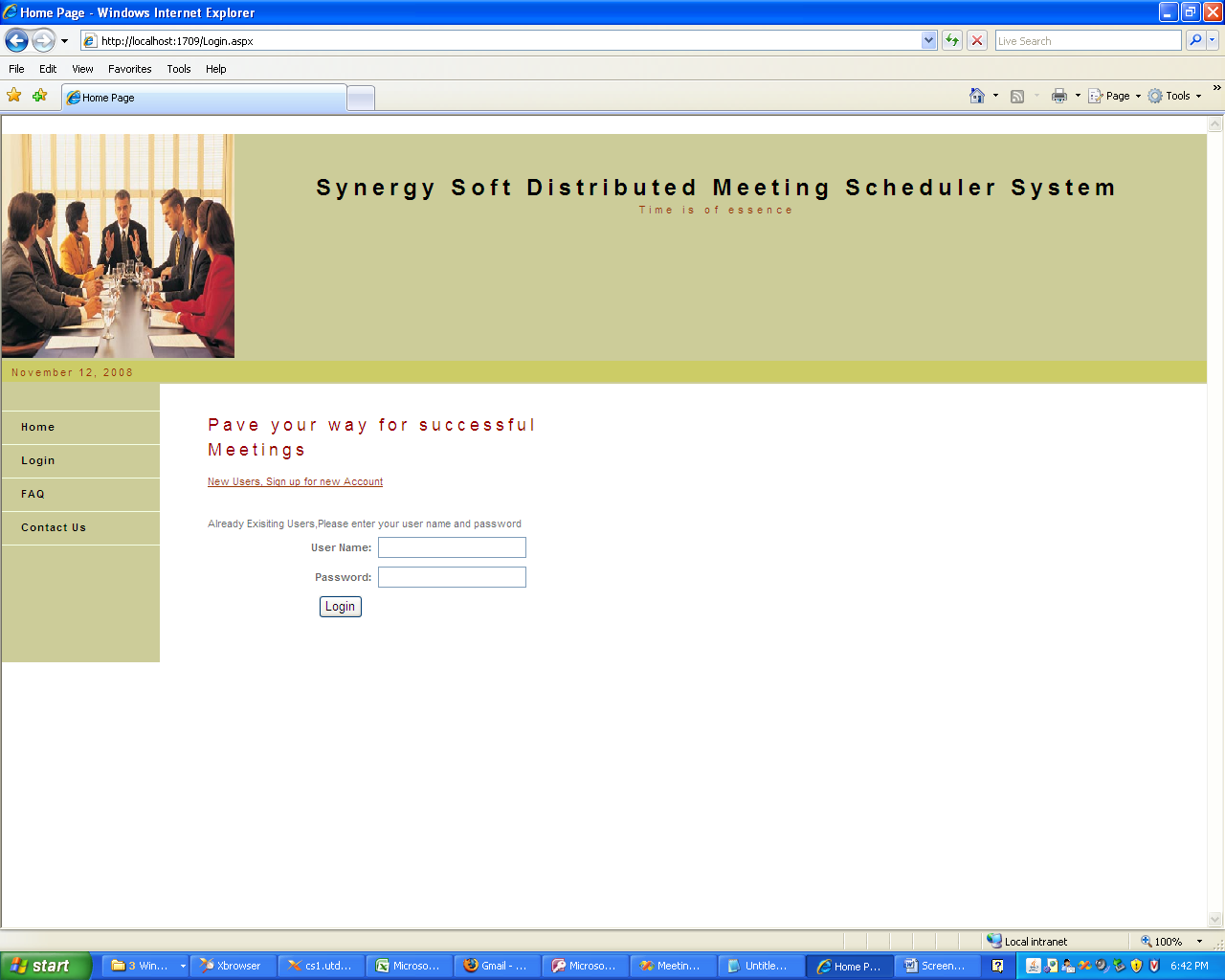
Admin Screen



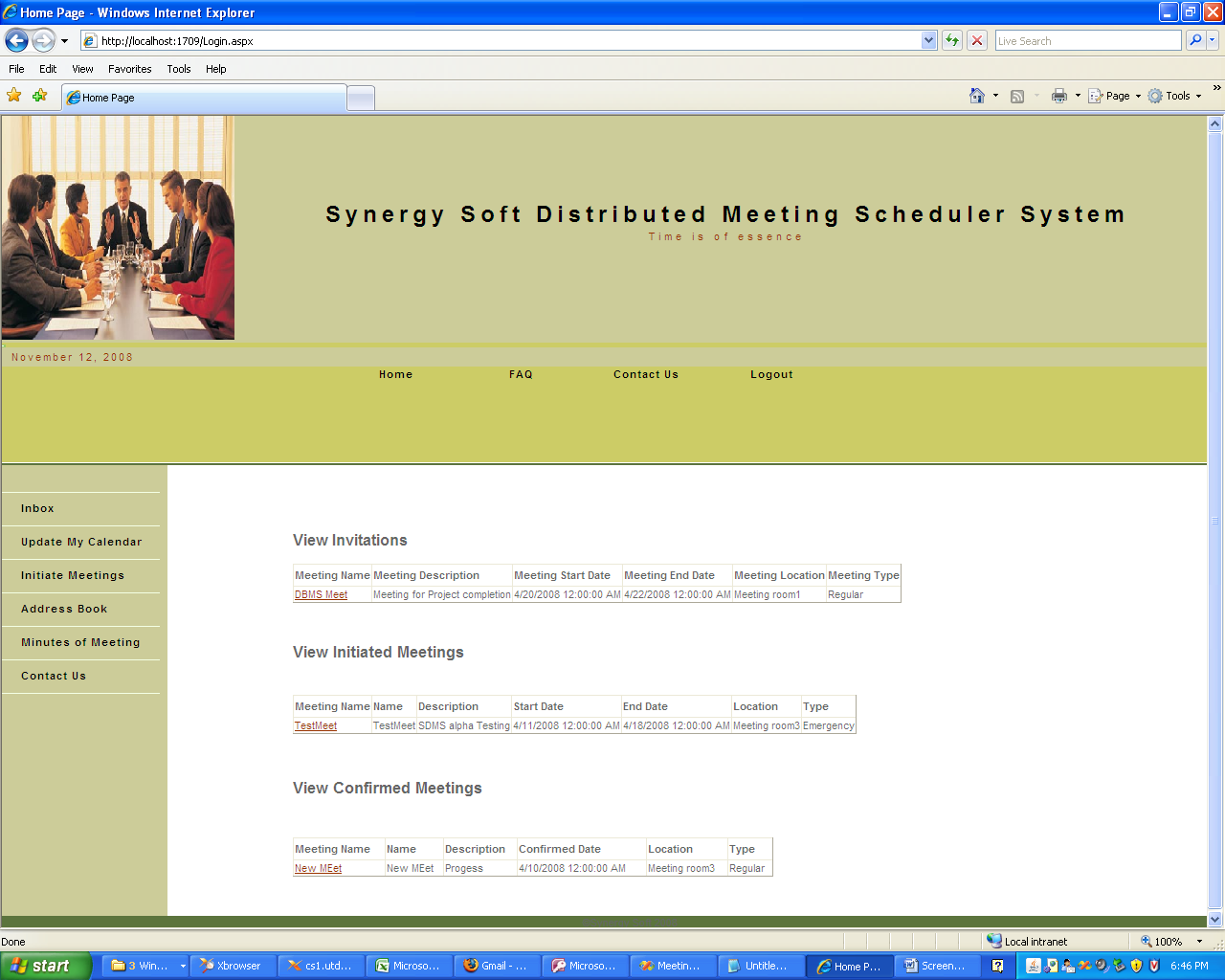
Admin – View Contacts



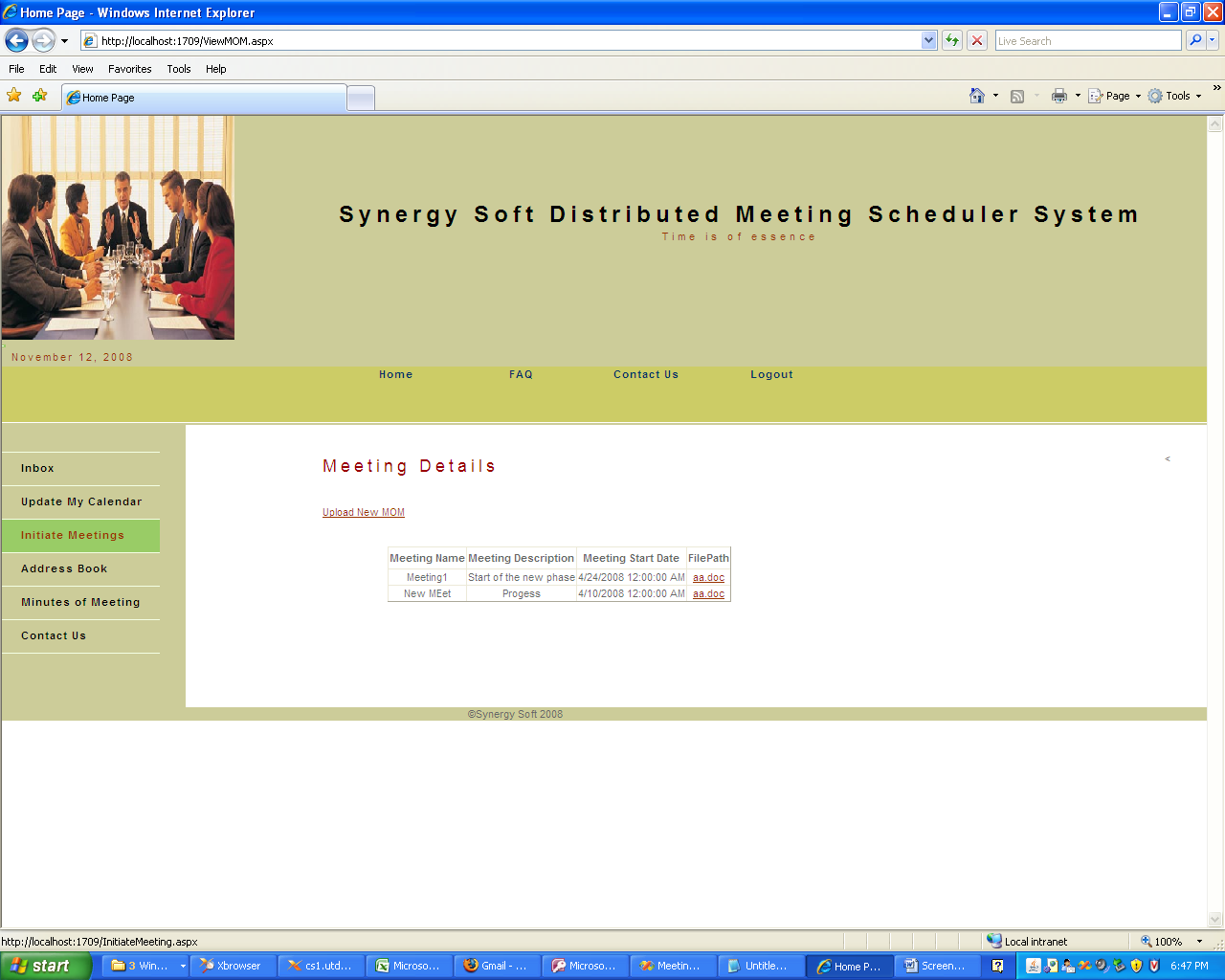
Admin – Edit/ Delete Contact



User - Login



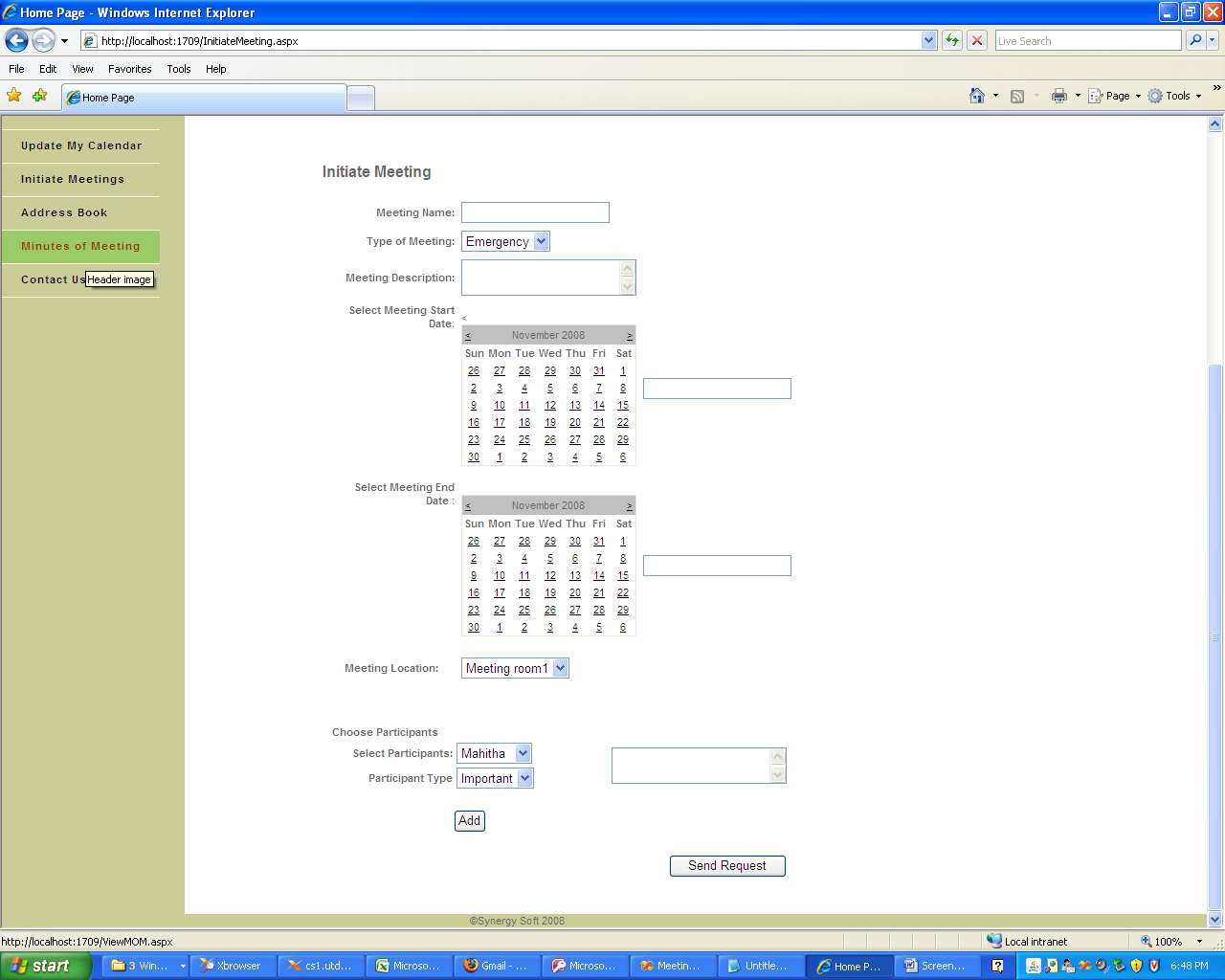
Welcome Page for User with meeting and invitation details



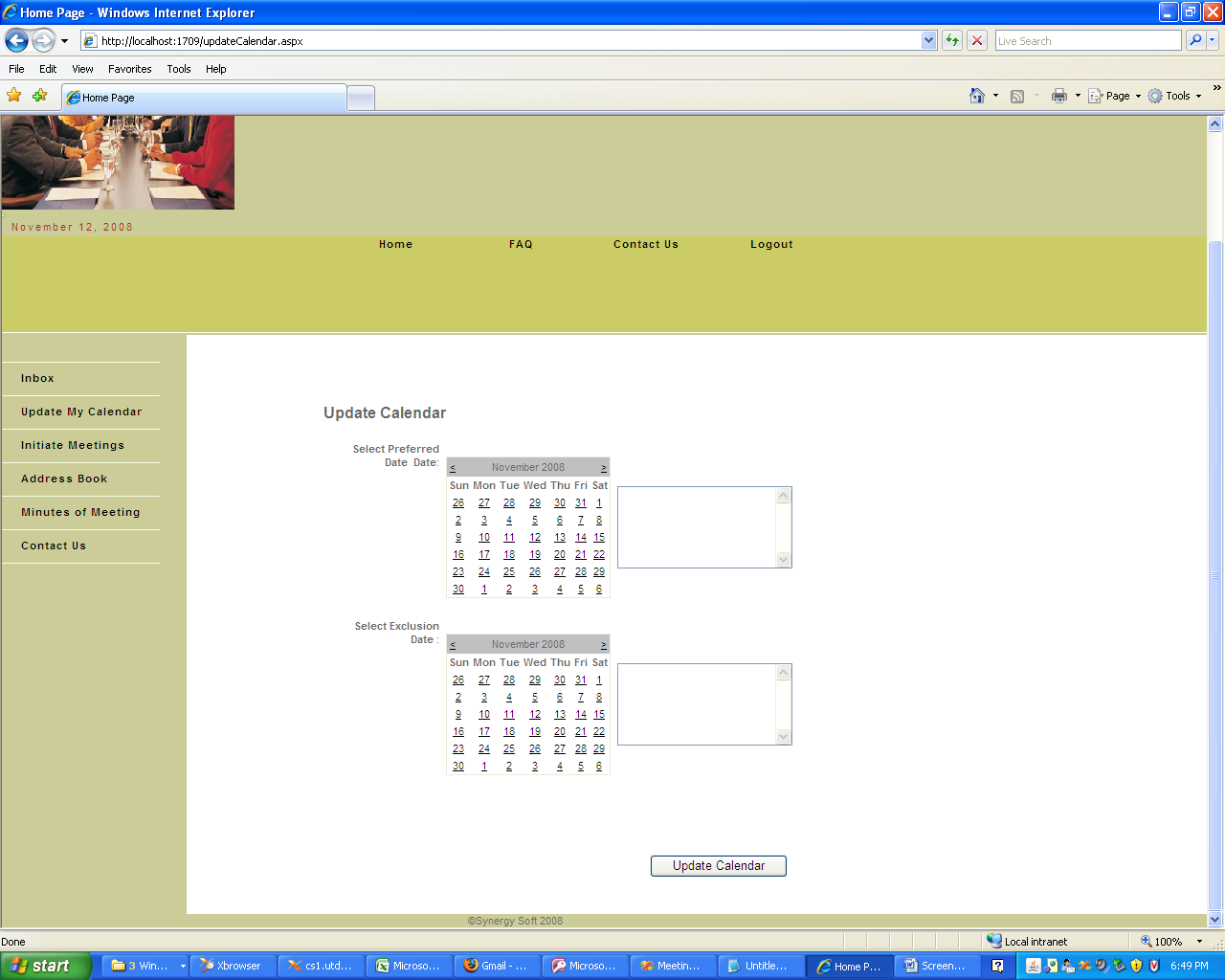
Minutes of Meetings



Address Book



Initiate Meeting



Update Calendar