

Synergy Distributed Meeting   
Scheduler

TEAM

**M**eeting **V**iew**P**oint

URL: [www.utdallas.edu/~jdv052000](http://www.utdallas.edu/~jdv052000)

Team Members

Bojan Knezevic – [bxk062000@utdallas.edu](mailto:bxk062000@utdallas.edu)

Haibo Shi – [hxs088000@utdallas.edu](mailto:hxs088000@utdallas.edu)

Hector Irizarry – [hector.irizarry@utdallas.edu](mailto:hector.irizarry@utdallas.edu)

Junia Valente – [jdv052000@utdallas.edu](mailto:jdv052000@utdallas.edu)

Mark Huber – [mah038000@utdallas.edu](mailto:mah038000@utdallas.edu)

Yuhan Tseng – [yxt083000@utdallas.edu](mailto:yxt083000@utdallas.edu)

Chun Du – [cxd081000@utdallas.edu](mailto:cxd081000@utdallas.edu)

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 11/05/2008 | 1.0 | Initial document. | Junia Valente  Bojan Knezevic |
| 11/05/2008 | 1.1 | Added content to section 1.3, 2.2 & 2.3. | Hector Irizarry |
| 11/07/2008 | 1.2 | Added content to section 2.1 and started section 3.2. | Yuhan Tseng  Junia Valente |
| 11/07/2008 | 1.3 | Finished section 3.2 and added 3.1, 3.3 and 3.4. | Hector Irizarry |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 5

1.1 Purpose 5

1.2 Scope 5

1.3 Definitions, Acronyms, and Abbreviations 5

1.4 References 5

1.5 Overview 5

2. Positioning 5

2.1 Business Opportunity 5

2.2 Problem Statement 5

2.3 Product Position Statement 5

3. Stakeholder and User Descriptions 5

3.1 Market Demographics 5

3.2 Stakeholder Summary 5

3.3 User Summary 5

3.4 User Environment 5

3.5 Stakeholder Profiles 5

3.5.1 <Stakeholder Name> 5

3.6 User Profiles 5

3.6.1 <User Name> 5

3.7 Key Stakeholder or User Needs 5

3.8 Alternatives and Competition 5

3.8.1 <aCompetitor> 5

3.8.2 <anotherCompetitor> 5

4. Product Overview 5

4.1 Product Perspective 5

4.2 Summary of Capabilities 5

4.3 Assumptions and Dependencies 5

4.4 Cost and Pricing 5

4.5 Licensing and Installation 5

5. Product Features 5

5.1 <aFeature> 5

5.2 <anotherFeature> 5

6. Constraints 5

7. Quality Ranges 5

8. Precedence and Priority 5

9. Other Product Requirements 5

9.1 Applicable Standards 5

9.2 System Requirements 5

9.3 Performance Requirements 5

9.4 Environmental Requirements 5

10. Documentation Requirements 5

10.1 User Manual 5

10.2 Online Help 5

10.3 Installation Guides, Configuration, and Read Me File 5

10.4 Labeling and Packaging 5

A Feature Attributes 5

A.1 Status 5

A.2 Benefit 5

A.3 Effort 5

A.4 Risk 5

A.5 Stability 5

A.6 Target Release 5

A.7 Assigned To 5

A.8 Reason 5

# Introduction

[The purpose of this document is to collect, analyze, and define high-level needs and features of the <<System Name>>. It focuses on the capabilities needed by the stakeholders and the target users, and **why** these needs exist. The details of how the <<System Name>> fulfills these needs are detailed in the use-case and supplementary specifications.]

[The introduction of the **Vision** document provides an overview of the entire document. It includes the purpose, scope, definitions, acronyms, abbreviations, references, and overview of this **Vision** document.]

## Purpose

[Specify the purpose of this **Vision** document**.**]

The purpose of this document is to collect, analyze, and define high-level needs and features of the ***Synergy Distributed Meeting Scheduler.*** It focuses on the capabilities needed by the stakeholders and the target users, and why these needs exist. The details of how the ***Synergy Distributed Meeting Scheduler*** fulfills these needs are detailed in the use-case and supplementary specifications

## Scope

[A brief description of the scope of this **Vision** document; what Project(s) it is associated with and anything else that is affected or influenced by this document.]

This Vision Document applies to the ***Synergy Distributed Meeting Scheduler,*** which will be developed by the

Meeting ViewPoint (MVP) Development team. The MVP team will develop a customizable, decentralized system that allows individuals or organizations to easily, efficiently, and precisely schedule meetings in accordance with practical limitations of virtual and real-world meetings. The SDMS will be a web based application designed to support the meeting scheduling needs of an organization. It will not require a client based application; instead it will be accessible and conform to standard HTML Web Application practices. The SDMS will interface with and utilize third party resources to facilitate all of the users’ meeting scheduling needs (e.g. database services, email communication, etc.).

## Definitions, Acronyms, and Abbreviations

**Active participant:** A user, who is also an attendee, whose role in the meeting requires them perform an action during the meeting (speaker, demo driver, etc). This user may also be asked provide requirements for equipment.

**Administrator:** is a privileged user who is responsible for managing user accounts, and managing resources (ex. adding or removing users, rooms, equipment, etc).

**Attendee:** a user, who receives a meeting invite, and is responsible for either accepting or declining the invite. In the case the invite is accepted, the attendee is required to provide an exclusion and preference set. An attendee can be furthermore classified as important or active participant.

**Concurrency:** the ability to handle more than one meeting requests at same time.

**Confirmation:** A notification sent to attendees by the initiator confirming the final meeting arrangements.

**COTS:** Commercial of-the-shelf. A software product that is ready-made and available for sale.

**Customer:** Synergy Soft Inc.

**Date conflict:** occurs when no available date can be found in the stated date range.

**Date range:** time interval specified by the initiator in which the meeting should take place, this also serves as the boundaries for the exclusion and preference sets.

**Date set:** a pair of input values, including calendar date and time period.

**End customer:** person, or organization, that buys the SDMS software.

**Equipment:** Any type of resource (e.g. projector, microphone, etc) that can be used in a meeting or event. They are further classified as movable or fixed. Movable equipment refers to equipment that can be transported from one location to another without requiring technician (hardware technician, electrician, handyman, etc) intervention. Fixed equipment refers to equipment that is assigned to a location (overhead projector, podium microphone, etc) wherein moving it to another location involves an installation that requires technician intervention.

**Exclusion set:** a set of dates on which the attendees are not available to attend a meeting.

**GUI:** Graphical User Interface.

**Internationalization (I18N)**: The process of designing a software application so that it can be adapted to various languages and regions without engineering changes.

**Important participant:**

A user, who is also an attendee, whose attendance at the meeting is necessary for the meeting to take place. This user may also be asked to provide their meeting location preferences.

**Initiator:** The user who calls for the meeting. The initiator is responsible for performing the meeting scheduling activities, or to delegate an initiator representative to perform this on their behalf.

**Initiator representative:** A user who is delegated to act on behalf of the initiator.

**Invite:** A meeting request sent by an initiator or representative to the potential attendees, which includes meeting topic, date range and requires attendees to respond with their preferences regarding date. For active participants the invite will require the attendee to provide equipment requirements. For important participants the invite will require the attendee to provide location preferences.

**Localization (L10N):** The process of adapting software for a specific region or language by adding locale-specific components and translating text.

**Mediator:** A user who has privileges to schedule resources (e.g. locations & equipment). This user also is tasked with determining meeting priority in the event of an irresolvable scheduling conflict.

**Meeting scheduling activities:** The tasks required in order to schedule a meeting. These usually involve the following tasks: planning the meeting, sending the invites, monitoring the responses, resolving conflicts, and confirming the final arrangements.

**Nomadicity:** The ability to move from one location to another and start communications from any location.

**Preference set:** a set of dates on which the attendees would prefer the meeting to take place.

**Private meeting:** a meeting that concerns only to the user. The attendee’s availability is marked as unavailable in their calendar and no details are given to other users.

**Professional meeting:** A meeting that concerns the user’s organization. The attendee’s availability is marked as unavailable in their calendar and general information about the meeting is visible to other users.

**SDMS:** Synergy Distributed Meeting Scheduler

**Strong date conflict:** Thisoccurs when no date can be found within the date range and outside all exclusion sets.

**Strong location conflict:** This occurs when there are no available locations which coincide with acceptable dates.

**Time interval:** a period of time with defined limits. For the purposes of the system, limits are defined in 15 minutes increments (e.g. 8:15 am, 8:30 am, 8:45 am & 9:00am)

**UML:** Unified Modeling Language

**User:** A person who interacts directly with the product. A user can have different roles with respect to the system (e.g. administrator, mediator, regular user) and meeting events (e.g. initiator, attendee, active participant, or important participant).

**Virtual location:** A meeting place which corresponds to a non–physical location where the meeting could take place (e.g. teleconferencing).

**Weak date conflict:** Thisoccurs whendates can be found within the date range and outside all exclusion sets, but no date can be found which coincides with all preference sets.

**Weak location conflict:** This occurs when the available locations do not coincide with the preferred locations.

## References

[This subsection provides a complete list of all documents referenced elsewhere in the **Vision** document. Identify each document by title, report number if applicable, date, and publishing organization. Specify the sources from which the references can be obtained. This informati*on may be provided by reference to an appendix or to another document.]*

[1] <http://utdallas.edu/~chung/RE/vision-doc-UTDCS-17-04.pdf>

[2] <http://www.ts.mah.se/RUP/RationalUnifiedProcess/>

## Overview

[This subsection describes what the rest of the **Vision** document contains and explains how the document is organized.]

The vision document is composed of the Positioning, Stakeholder and User Descriptions, Product Overview, Product Features, Constraints, Quality Ranges, Precedence and Priority, Other Product Requirements, and Documentation Requirements.

Positioning briefly describes the business opportunity being met by this project and provides a statement summarizing the problem being solved by this project.

Stakeholder and User Descriptions section provides a profile of the stakeholders and users involved in the project, and the key problems that they perceive to be addressed by the proposed solution.

Product Overview section provides a high level view of the product capabilities, interfaces to other applications, and system configurations.

Product Features section describes features and capabilities of the system that are necessary to deliver benefits to the users.

Constraints section describes design constraints, external constraints or other dependencies

Quality Ranges describes performance, robustness, fault tolerance, and usability.

Precedence and Priority section describes priority of the different system features.

Other Product Requirements describes standards, hardware or platform requirements, performance requirements, and environmental requirements

Documentation Requirements section describes the documentation that must be developed to support successful application deployment

# Positioning

## Business Opportunity

[Briefly describe the business opportunity being met by this project.]

A facility for scheduling meetings has many potential applications, such as scheduling courses and flights, room assignments at hospitals and hotels, scheduling national and international meetings, logistics, job scheduling in production systems, as well as command and control systems. The particular type of systems this project is intended for is supporting people to schedule their meetings. Many software vendors are eager to offer such a system, especially one with a powerful vantage point (cf., Microsoft, IBM-Lotus, etc.). In particular, SynergySoft, Inc. aims to provide such a facility which would outperform any such system that is currently available in the highly competitive market. Synergy Distributed Meeting Scheduler is aimed towards organizations with frequent meeting scheduling, organization, and administration needs. The SDMS will facilitate meeting management for both traditional and distributed meeting styles to meet the needs of modern work environments.

## Problem Statement

[Provide a statement summarizing the problem being solved by this project. The following format may be used:]

|  |  |
| --- | --- |
| The problem of | Difficulty in organizing meetings. |
| affects | Any organization or individuals that need organized meetings in order to fulfill their goals. |
| the impact of which is | Entities spend a significant amount of time and resources organizing meetings due:  -difficulty in contacting attendees and collecting their availability and preferences data.  -complexity of processing and analyzing data in order to make a decision about meeting arrangements.  - Re- planning and interactions that occur as result negotiation activities.  The effects of all the factors above are amplified as the number of participants’ increases. |
| a successful solution would be | Contact and collect participant availability data. Aid the user to decide on the meeting arrangement by ranking alternatives based user configurable criteria. Allow for re-planning and support interactions for negotiation activities. |

## Product Position Statement

[Provide an overall statement summarizing, at the highest level, the unique position the product intends to fill in the marketplace. The following format may be used:]

|  |  |
| --- | --- |
| For | Any organization or individuals. |
| Who | That needs organized meetings in order to fulfill their goals. |
| The (product name) | Is a software product |
| That | Supports the meeting organization activities. |
| Unlike | Unlike IBM Lotus Notes and Microsoft Outlook will provide support for conflict negotiation and make suggestions for meeting arrangements based on user defined criteria. |
| Our product | Provides mechanisms to distribute meeting requests and collects participant responses. Ranks meeting alternatives based user configurable preferences. Allow for re-planning and provides a mechanisms to perform conflict resolution. |

[A product position statement communicates the intent of the application and the importance of the project to all concerned personnel.]

# Stakeholder and User Descriptions

[To effectively provide products and services that meet your stakeholders’ and users' real needs, it is necessary to identify and involve all of the stakeholders as part of the Requirements Modeling process. You must also identify the users of the system and ensure that the stakeholder community adequately represents them. This section provides a profile of the stakeholders and users involved in the project, and the key problems that they perceive to be addressed by the proposed solution. It does not describe their specific requests or requirements as these are captured in a separate stakeholder requests artifact. Instead, it provides the background and justification for why the requirements are needed.]

## Market Demographics

[Summarize the key market demographics that motivate your product decisions. Describe and position target market segments. Estimate the market’s size and growth by using the number of potential users or the amount of money your customers spend trying to meet needs that your product or enhancement would fulfill. Review major industry trends and technologies. Answer these strategic questions:

• What is your organization’s reputation in these markets?

• What would you like it to be?

• How does this product or service support your goals?]

The target market includes organizations with members or subdivision distributed across several geographic locations. The users are expected to be familiar with basic computers usage tasks and popular business software suites (MS Office, IBM Lotus, etc).

Our customer SynergySoft Inc has a well-established reputation as a software solutions provider across the industry and is looking to enter this new market with the SMDS.

## Stakeholder Summary

[There are a number of stakeholders with an interest in the development and not all of them are end users. Present a summary list of these non-user stakeholders. (The users are summarized in section 3.3.)]

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Responsibilities** |
| SynergySoft Inc | This stakeholder that contracted a requirements engineer of a consulting firm to refine the scheduling meeting system requirements. | Ensures that the consulting firm provides such a facility which would outperform any other software vendors that is currently available in the highly competitive market.  Ensure that the consulting firm will come up with detailed requirements description that captures customers’ real needs and wants as precisely, concisely and conceptually as possible. |
| Technical Reviewer | This is a stakeholder that must be involved regularly to maintain the development cycle. | Review and provide feedback concerning the development process.  Reviews and provides feedback on work products. |
| System analyst | This is a stakeholder that works with the stakeholders to gather their needs. and delimiting | Leads and coordinates requirements elicitation efforts.  Leads and coordinates use-case modeling by outlining the system's functionality the system |
| Requirement Specifier | This is a stakeholder that works with the Analysts to correctly translate requests/needs into requirements to be used for design. | Specifies the details of one or more a parts of the system's functionality by describing one or the aspects of the requirements, this will include functional and non-functional. |
| Software Architect | This is a stakeholder that is primary for leading the system development from a technical perspective. | Generates software architecture artifacts, derived from key technical decisions that constrain the overall design and implementation for the project.  Ensures that the system is going to be maintainable and the architectural solution supports the functional and non-requirements. |
| Project Manager | This is a stakeholder that is primary for leading the system development from a management perspective. | Plans, manages and allocates resources, specifies priorities, coordinates interactions with customers and users, and keeps the team focused. Also establishes a set of practices that ensure the integrity and quality of project artifacts. |
| Software Developer | This is a stakeholder that is primary for producing the actual software products. | Generates software artifacts according to the design.  Follows the project process. |
| Software tester | This is a stakeholder that is primary for validation and verification of the system. | Ensures that the system is works correctly and fulfills the design specifications.  Generates test plans and procedures.  Documents and report bugs encountered.  Follows the project process. |
| Software Maintenance Developer | This is a stakeholder that is primary for resolving problems with software after released. | Generates software patches of fixed to resolve problems (non-related with installation of configuration) with system between releases.  Generates software patch installation procedures.  Documents changes and updates design documents to reflect changes.  Follows the project process. |
| Market Analyst | This is a stakeholder that will assist our abilities to position our product successfully. | Ensures that there is going to be a market demand for the product's features. |

## User Summary

[Present a summary list of all identified users.]

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Description** | **Responsibilities** | **Stakeholder** |
| Customer | Entity that buys the software. | <TBD> | Self |
| Users | Primary end user of the software. | schedule meetings, respond to meetings invites, view meetings, send/receive meeting confirmations, specify contact information, assign representative, negotiate/solve conflicts | Self |
| System administrator | End user of the software. | In addition to the regular User responsibilities. Manages user’s accounts, locations and equipment. Installs, repairs, starts and stops system. | Self |

## User Environment

[Detail the working environment of the target user. Here are some suggestions:

Number of people involved in completing the task? Is this changing?

How long is a task cycle? Amount of time spent in each activity? Is this changing?

Any unique environmental constraints: mobile, outdoors, in-flight, and so on?

Which systems platforms are in use today? Future platforms?

What other applications are in use? Does your application need to integrate with them?

This is where extracts from the Business Model could be included to outline the task and roles involved and so on.]

Typically only one user is involve performing a task (send invite, respond to invite, assign delegate, etc) and should take from 5-30 minutes.

Since system will follow client-server architecture there are two operating environments to be consider client and server side. The client-side is platform independent and only requires web browsing capabilities. The server-side platform requirements correspond to those of the underling application server (Microsoft IIS 6.0 Web Server).

Users are expected to access the system through a browser-enable device and have network access to server.

System administrators depending on the task are expected to access the system remotely through its web interface or locally.

The SMDS interacts with:

#### Microsoft SQL Server for database interactions.

* Microsoft IIS 6.0 Web Server to deliver HTML content to clients.
* Microsoft Active Directory via the LDAP protocol for user authentication.
* Microsoft Exchange Servers e-mail notification and calendar synchronization.

## Stakeholder Profiles

[Describe each stakeholder in the system here by filling in the following table for each stakeholder. Remember that stakeholder types can be as divergent as users, departments, and technical developers. A thorough profile would cover the following topics for each type of stakeholder.]

### <Stakeholder Name>

|  |  |
| --- | --- |
| **Representative** | [Who is the stakeholder representative to the project? (Optional if documented elsewhere.) What we want here is names.] |
| **Description** | [A brief description of the stakeholder type.] |
| **Type** | [Qualify the stakeholder’s expertise, technical background, and degree of sophistication—that is, guru, business, expert, casual user, and so on.] |
| **Responsibilities** | [List the stakeholder’s key responsibilities with regard to the system being developed—that is, their interest as a stakeholder.] |
| **Success Criteria** | [How does the stakeholder define success?  How is the stakeholder rewarded?] |
| **Involvement** | [How is the stakeholder involved in the project? Relate where possible to Rational Unified Process roles—that is, Requirements Reviewer and so on.] |
| **Deliverables** | [Are there any additional deliverables required by the stakeholder? These could be project deliverables or outputs from the system under development.] |
| **Comments / Issues** | [Problems that interfere with success and any other relevant information go here.] |

## User Profiles

[Describe each unique user of the system here by filling in the following table for each user type. Remember user types can be as divergent as gurus and novices. For example, a guru might need a sophisticated, flexible tool with cross-platform support, while a novice might need a tool that is easy to use and user-friendly. A thorough profile needs to cover the following topics for each type of user.]

### <User Name>

|  |  |
| --- | --- |
| **Representative** | [Who is the user representative to the project? (Optional if documented elsewhere.) This often refers to the Stakeholder that represents the set of users, for example, Stakeholder: Stakeholder1.] |
| **Description** | [A brief description of the user type.] |
| **Type** | [Qualify the user’s expertise, technical background, and degree of sophistication—that is, guru, casual user, and so on.] |
| **Responsibilities** | [List the user’s key responsibilities with regard to the system being developed— that is, captures details, produces reports, coordinates work, and so forth.] |
| **Success Criteria** | [How does the user define success?  How is the user rewarded?] |
| **Involvement** | [How is the user involved in the project? Relate where possible to Rational Unified Process roles—that is, Requirements Reviewer, and so on.] |
| **Deliverables** | [Are there any deliverables the user produces and, if so, for whom?] |
| **Comments / Issues** | [Problems that interfere with success and any other relevant information go here. These would include trends that make the user’s job easier or harder.] |

## Key Stakeholder or User Needs

[List the key problems with existing solutions as perceived by the stakeholder or user. Clarify the following issues for each problem:

• What are the reasons for this problem?

• How is it solved now?

• What solutions does the stakeholder or user want?]

[It is important to understand the **relative** importance the stakeholder or user places on solving each problem. Ranking and cumulative voting techniques indicate problems that **must** be solved versus issues they would like addressed.

Fill in the following table—if using Rational RequisitePro to capture the Needs, this could be an extract or report from that tool.]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Need** | **Priority** | **Concerns** | **Current Solution** | **Proposed Solutions** | |
| Broadcast messages |  |  |  | |  |

## Alternatives and Competition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Product Name | Equipment | Location | Email | Preference set | Negotiation | Exclusion set | Representative set |
| MS Outlook | X  (connected with MS Exchange) | X (connected with MS Exchange) | X |  |  | X | X |
| IBM Lotus Notes |  | X | X |  |  | X | X |
| Google  Calendar |  |  | X |  |  |  |  |
| SDMS | X | X |  | X | X | X | X |

# Product Overview

[This section provides a high level view of the product capabilities, interfaces to other applications, and system configurations. This section usually consists of three subsections, as follows:

• Product perspective

• Product functions

• Assumptions and dependencies]

This section provides a high level view of the SDMS capabilities, interfaces to the external Users and database, and the system configuration.

## Product Perspective

[This subsection of the **Vision** document puts the product in perspective to other related products and the user’s environment. If the product is independent and totally self-contained, state it here. If the product is a component of a larger system, then this subsection needs to relate how these systems interact and needs to identify the relevant interfaces between the systems. One easy way to display the major components of the larger system, interconnections, and external interfaces is with a block diagram.]

The SDMS will interface with the existing Users and database as shown in the context diagram below (see Figure TBD).

The SDMS will consist of a thin-client component and server component as illustrated in Figure TBD. The server component runs on Microsoft ISS Server 7.0. The server component must interface with the Users through web interface and to a Microsoft SQL Server 2008 running.

The client component is accessible to any web browser that supports ASP.NET and .Net Framework 3.5. The web interface should be compliant to current and emerging HTML designs. A valid username and password must be entered in order for access to be granted.



## Summary of Capabilities

[Summarize the major benefits and features the product will provide. For example, a **Vision** document for a customer support system may use this part to address problem documentation, routing, and status reporting without mentioning the amount of detail each of these functions requires.

Organize the functions so the list is understandable to the customer or to anyone else reading the document for the first time. A simple table listing the key benefits and their supporting features might suffice. For example:]

**Table 4-1 Customer Support System**

|  |  |
| --- | --- |
| **Customer Benefit** | **Supporting Features** |
| New support staff can quickly get up to speed. | Knowledge base assists support personnel in quickly identifying known fixes and workarounds. |
| Customer satisfaction is improved because nothing falls through the cracks. | Problems are uniquely itemized, classified and tracked throughout the resolution process. Automatic notification occurs for any aging issues. |
| Management can identify problem areas and gauge staff workload. | Trend and distribution reports allow high level review of problem status. |
| Distributed support teams can work together to solve problems. | Replication server allows current database information to be shared across the enterprise. |
| Customers can help themselves, lowering support costs and improving response time. | Knowledge base can be made available over the Internet. Includes hypertext search capabilities and graphical query engine. |
| Access to up-to-date schedule information | The system accesses the database for the latest schedule information.  The Users will be able to review their schedule. |
| Instant feedback on meeting status. | All responses and interactions are instantly capture and made availability. |
| Secure and confidential | A valid username and password is required to access to the SDMS.  Users’ information and schedule protected from unauthorized access. |
| Access from web enable device. | Users may access the SMDS from any computer or from mobile device via the internet.  No installation is required since it accessed through a web interface. |
| Easy and timely access to meeting schedule | Users can view their schedules in by providing their username and password.  Users may access the SMDS from any computer or from mobile device via the internet. |
| Easy way to schedule meetings | Users can provide the meeting data and system will provide suggestions on potential date/location options.  The system will distribute meeting invites and collect all responses. |
| Support for conflict resolution | Users can choose from several conflicts resolution options including mediation and conventional methods.  The system will support conflict resolution by managing user iterations and providing special privileges to the Mediator user. |

## Assumptions and Dependencies

[List each of the factors that affect the features stated in the **Vision** document. List assumptions that, if changed, will alter the **Vision** document. For example, an assumption may state that a specific operating system will be available for the hardware designated for the software product. If the operating system is not available, the **Vision** document will need to change.]

The following assumptions and dependencies relate to the capabilities of the SDMS as outlined in this Vision Document:

* It is assumed that the organization will operate and support the Microsoft IIS Server 7.0.
* It is assumed that the organization will operate and support the Microsoft SQL Server 2008.
* It is assumed that popular web browser vendors will support ASP.net 3.5 and .NET 3.5 Framework.

## Cost and Pricing

The product will be part of open source under GNU GPL (GNU General Public License).

Optional support fee for subscribed users of $15,000 per year will be provided.

## Licensing and Installation

[Licensing and installation issues can also directly impact the development effort. For example, the need to support serializing, password security or network licensing will create additional requirements of the system that must be considered in the development effort.

Installation requirements may also affect coding or create the need for separate installation software.]

No professional installation required.

# Product Features

[List and briefly describe the product features. Features are the high-level capabilities of the system that are necessary to deliver benefits to the users. Each feature is an externally desired service that typically requires a series of inputs to achieve the desired result. For example, a feature of a problem tracking system might be the ability to provide trending reports. As the use-case model takes shape, update the description to refer to the use cases.

Because the **Vision** document is reviewed by a wide variety of involved personnel, the level of detail needs to be general enough for everyone to understand. However, enough detail must be available to provide the team with the information they need to create a use-case model.

To effectively manage application complexity, we recommend for any new system, or an increment to an existing system, capabilities are abstracted to a high enough level so 25-99 features result. These features provide the fundamental basis for product definition, scope management, and project management. Each feature will be expanded in greater detail in the use-case model.

Throughout this section, each feature will be externally perceivable by users, operators or other external systems. These features need to include a description of functionality and any relevant usability issues that must be addressed. The following guidelines apply:

• Avoid design. Keep feature descriptions at a general level. Focus on capabilities needed and why (not how) they should be implemented.

• If you are using the Rational RequisitePro toolkit, all need to be selected as requirements of type for easy reference and tracking.]

## <aFeature>

## <anotherFeature>

# Constraints

[Note any design constraints, external constraints or other dependencies.]

# Quality Ranges

[Define the quality ranges for performance, robustness, fault tolerance, usability, and similar characteristics that are not captured in the Feature Set.]

# Precedence and Priority

[Define the priority of the different system features.]

# Other Product Requirements

[At a high level, list applicable standards, hardware or platform requirements, performance requirements, and environmental requirements.]

## Applicable Standards

[List all standards with which the product must comply. These can include legal and regulatory (FDA, UCC) communications standards (TCP/IP, ISDN), platform compliance standards (Windows, UNIX, and so on), and quality and safety standards (UL, ISO, CMM).]

## System Requirements

[Define any system requirements necessary to support the application. These can include the supported host operating systems and network platforms, configurations, memory, peripherals, and companion software.]

## Performance Requirements

[Use this section to detail performance requirements. Performance issues can include such items as user load factors, bandwidth or communication capacity, throughput, accuracy, and reliability or response times under a variety of loading conditions.]

## Environmental Requirements

[Detail environmental requirements as needed. For hardware- based systems, environmental issues can include temperature, shock, humidity, radiation, and so forth. For software applications, environmental factors can include usage conditions, user environment, resource availability, maintenance issues, and error handling and recovery.]

# Documentation Requirements

[This section describes the documentation that must be developed to support successful application deployment.]

## User Manual

[Describe the purpose and contents of the User Manual. Discuss desired length, level of detail, need for index, glossary of terms, tutorial versus reference manual strategy, and so on. Formatting and printing constraints must also be identified.]

## Online Help

[Many applications provide an online help system to assist the user. The nature of these systems is unique to application development as they combine aspects of programming (hyperlinks, and so forth) with aspects of technical writing, such as organization and presentation. Many have found the development of an online help system is a project within a project that benefits from up-front scope management and planning activity.]

## Installation Guides, Configuration, and Read Me File

[A document that includes installation instructions and configuration guidelines is important to a full solution offering. Also, a Read Me file is typically included as a standard component. The Read Me file can include a "What's New With This Release” section, and a discussion of compatibility issues with earlier releases. Most users also appreciate documentation defining any known bugs and workarounds in the Read Me file.]

## Labeling and Packaging

[Today's state-of-the-art applications provide a consistent look and feel that begins with product packaging and manifests through installation menus, splash screens, help systems, GUI dialogs, and so on. This section defines the needs and types of labeling to be incorporated into the code. Examples include copyright and patent notices, corporate logos, standardized icons and other graphic elements, and so forth.]

# A Feature Attributes

[Features are given attributes that can be used to evaluate, track, prioritize, and manage the product items proposed for implementation. All requirement types and attributes need to be outlined in the Requirements Management Plan, however, you may wish to list and briefly describe the attributes for features that have been chosen. The following subsections represent a set of suggested feature attributes.]

## A.1 Status

[Set after negotiation and review by the project management team. Tracks progress during definition of the project baseline.]

|  |  |
| --- | --- |
| Proposed | [Used to describe features that are under discussion but have not yet been reviewed and accepted by the "official channel," such as a working group consisting of representatives from the project team, product management, and user or customer community.] |
| Approved | [Capabilities that are deemed useful and feasible, and have been approved for implementation by the official channel.] |
| Incorporated | [Features incorporated into the product baseline at a specific point in time.] |

## A.2 Benefit

[Set by Marketing, the product manager or the business analyst. All requirements are not created equal. Ranking requirements by their relative benefit to the end user opens a dialog with customers, analysts, and members of the development team. Used in managing scope and determining development priority.]

|  |  |
| --- | --- |
| Critical | [Essential features. Failure to implement means the system will not meet customer needs. All critical features must be implemented in the release or the schedule will slip.] |
| Important | [Features important to the effectiveness and efficiency of the system for most applications. The functionality cannot be easily provided in some other way. Lack of inclusion of an important feature may affect customer or user satisfaction, or even revenue, but release will not be delayed due to lack of any important feature.] |
| Useful | [Features that are useful in less typical applications will be used less frequently or for which reasonably efficient workarounds can be achieved. No significant revenue or customer satisfaction impact can be expected if such an item is not included in a release.] |

## A.3 Effort

[Set by the development team. Because some features require more time and resources than others, estimating the number of team or person-weeks, lines of code required or function points, for example, is the best way to gauge complexity and set expectations of what can and cannot be accomplished in a given time frame. Used in managing scope and determining development priority.]

## A.4 Risk

[Set by development team based on the probability the project will experience undesirable events, such as cost overruns, schedule delays or even cancellation. Most project managers find categorizing risks, as high, medium, and low, is sufficient, although finer gradations are possible. Risk can often be indirectly assessed by measuring the uncertainty (range) of the projects team’s schedule estimate.]

## A.5 Stability

[Set by the analyst and development team, this is based on the probability that features will change or the team’s understanding of the feature will change. Used to help establish development priorities and determine those items for which additional elicitation is the appropriate next action.]

## A.6 Target Release

[Records the intended product version in which the feature will first appear. This field can be used to allocate features from a **Vision** document into a particular baseline release. When combined with the status field, your team can propose, record, and discuss various features of the release without committing them to development. Only features whose Status is set to Incorporated and whose Target Release is defined will be implemented. When scope management occurs, the Target Release Version Number can be increased so the item will remain in the **Vision** document but will be scheduled for a later release.]

## A.7 Assigned To

[In many projects, features will be assigned to "feature teams" responsible for further elicitation, writing the software requirements, and implementation. This simple pull-down list will help everyone on the project team to understand responsibilities better.]

## A.8 Reason

[This text field is used to track the source of the requested feature. Requirements exist for specific reasons. This field records an explanation or a reference to an explanation. For example, the reference might be to a page and line number of a product requirement specification or to a minute marker on a video of an important customer review.]