CleverSupply

Architecture Description

# Problem Background

## System Context

# systems’ mission

# system’s responsibilities

# human actors and system actors system should interact with

*Mission and system description:*

Facilitate the connection between the entities in a supply chain, thus reducing the cost of operation. Make it easier for producers to find what they need from various suppliers, and ease communication via web conferencing.

*Environment and context:*

* Targeted audience: small to medium business lines.
* Web-platform intended for use by suppliers, producers (supply consumers), retailers (product vendors). The system should make it easier for the parties involved to find and procure what they need and to maintain relations.
* Deployment technology: world wide web
* Revenue for CleverSupply: subscription and ad based income

*External entities/systems with which the system interacts:*

* payment services (for paying for the orders, and also for paying for subscriptions towards our company)

*System features and responsibilities:*

* web conference with multiple participants (video, audio, text)
* real-time offer comparison
* meeting scheduler and notification system
* conference recording and archiving
* order placement
* secure order payment
* order history
* order forecasting
* product inventory
* search by various criteria (company, product, geographical localization)
* ranking/feedback

## Business Goals

# prioritized list of business goals for system’s creation or modification.

|  |  |
| --- | --- |
| N | Description |
| 1 | Increase companies revenues by facilitating them to find business partners and negotiate contracts. |
| 2 | Reduce TCO by gathering market information, thus comparing offers. |
| 3 | Improve operational efficiency by enhancing the communication between the entities in the supply chain. |
| 4 | Increase confidence in the registered businesses through feedback and ranking. |
| 5 | Attract numerous users of the platform by offering high-quality features. |
| 6 | Provide a confident solution. |

## Stakeholders

|  |  |
| --- | --- |
| Stakeholder | Concerns |
| Suppliers | Increase market exposure. |
| Producers/Retailers | Find and contact business partners easily. |
| SysAdmin | Deployment. Recovery from failure. |
| CleverSupply shareholders | Interested in the revenues that the application generates. |
| Developers | Modularity. New technologies. |

## Architecture Driving Requirements

# prioritized list of architecture driving requirements (major functional, quality attribute, and life-cycle requirements)

|  |  |  |
| --- | --- | --- |
| N | Description | Derived from objectives |
|  | Producers/retailers can compare multiple offers, place orders, view order history and schedule recurrent orders. | 1,2,3,5 |
|  | System should support 1000 concurrent web conferences, each having a maximum of 25 participants, created in less than 1s. | 6,5,3 |
|  | Wide geographic distribution (system should be available world-wide) | 1,2,3,5 |
|  | System should have an up-time of 99.99% for an average of 100 concurrent web conferences. | 6 |
|  | In the event of an overall system crash, the data should be recoverable and the system will be up in less than 30 minutes. | 6 |
|  | System must support data import of different types of products. | 5 |
|  | The system should support multiple payment services. | 5,6 |
|  | Payments done via the system will be handled by an external secure payment service. | 6 |
|  | Only registered users can create a product inventory and only subscribing users can initiate conference and use order management features. | 6 |
|  | Access from all major web browsers (Chrome, IE, Firefox, Safari) should have the same look and feel and should function correctly. | 5,6 |
|  | Users can provide feedback and rankings on other businesses. | 4 |
|  | The communication between the users should be secure. | 6,5 |
|  | The system should not depend on external telecom vendors. |  |

# Solution

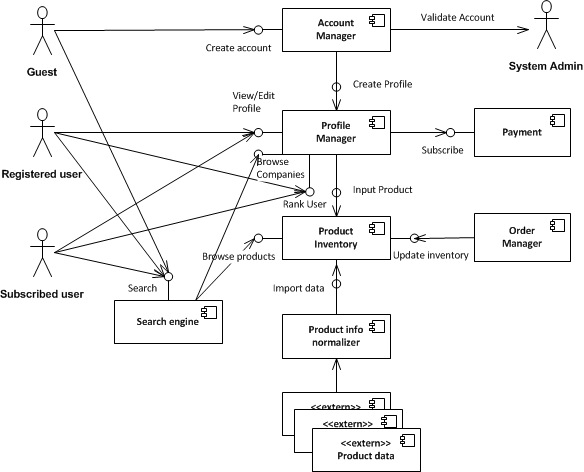
## Approach Summary

# Style, principles, patterns, design decisions

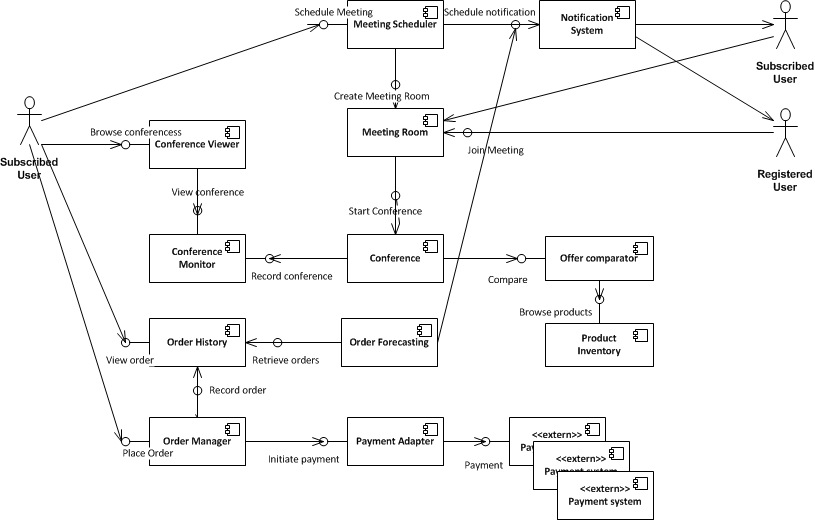
* Multi-tier web based architecture
* Principles:
  + Modularity, modifiability

## Functional View

# major system use cases



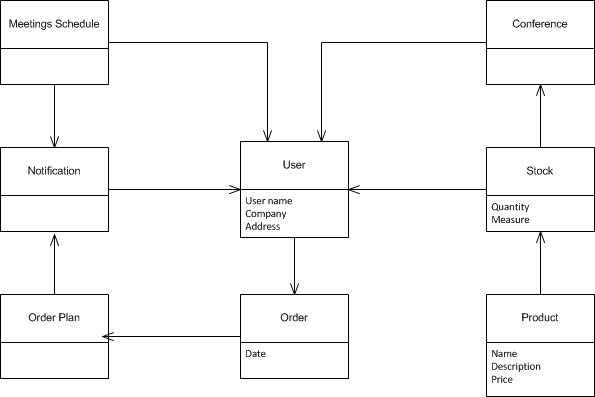
|  |  |
| --- | --- |
| Component | Description (responsibilities) |
| Account Manager | Receives registration requests and creates profiles |
| Profile Manager | View/edit company information. Shows current stock, orders and all established business partners. Allow companies to give feedback regarding interactions with other businesses. |
| Product Inventory | Companies can create a product catalog and the stock is updated automatically after order processing. |
| Product Info Normalizer | Allows companies to import large set of product records. It transforms the external input to a generic system template. |
| Search Engine | Companies can find new business partners by product offers and geographic localization. |



|  |  |
| --- | --- |
| Component | Description (responsibilities) |
| Meeting Scheduler | Companies can schedule meetings either now or for a later time and triggers invitations to participants. |
| Notification System | Sends invitations to all participants to the conference. |
| Meeting Room | Maintains attendees list and starts conference when all participants are ready. |
| Conference | Manages the video, audio and text connection between participants. |
| Conference Monitors | Records conferences. |
| Conference Viewer | Companies can review previous conferences. |
| Offer comparator | During a conference product details and compared offers can be displayed for a better negotiation. |
| Order Manager | Allows subscribed companies to place an order. |
| Order History | Records orders and allows users to view information about the business parties. |
| Order Forecasting | Based on the order history of a company, predicts future possible orders and sets up notifications for them (via notification system). |
| Payment Adapter | Interacts with different external payment services. |

## Information View

# Class diagram with key entities



|  |  |  |
| --- | --- | --- |
| Entity | Life-cycle considerations (reading/writing components, access restrictions, states, archiving) | Performance considerations (volumes, volatility) |
| Conference | Created by any subscribed user  Started by creator  Deleted when creator leaves | Hundreds-thousands, each one having tens of participants |
| Order | Added by buyer  Confirmed by payment gateway  Persisted in database |  |
| Product | Added by seller  Persisted in database  Deleted by seller |  |
| Stock | Added by seller  Updated by system and seller  Persisted in database |  |
| User | Added by registration in the system  Confirmed by administrator | Hundreds of thousands |
| Meetings Schedule | Created by system  Updated by users and system |  |

## Concurrency View

# Any concurrency considerations (state, synchronization, integrity, restart, etc)

* In order to avoid bottlenecks regarding the video streaming, multiple streaming servers will be used
* Non streaming requests will be handled by multiple web application instances on several different application servers through the use of sticky sessions on load balancers
* Load balancers will be used between application servers and streaming servers, for choosing a streaming server that is not too loaded
* For concurrent database access, optimistic locking will be used
* Socket based communication (internal protocol) will be used for inter process communication between application servers and streaming servers for establishing of a new conference
* In case of a conference failure, the conference will be restarted on the same streaming server
* In case of the failure of a streaming server, all its conferences should be moved/restarted on another streaming server

## Development View

|  |  |  |
| --- | --- | --- |
| Component  (Component type) | Implementation technology | Design standards and guidelines |
| User-Interface | HTML4.1, Jquery, Java Applet, AJAX |  |
| Database interface | JDBC, Hibernate3 |  |
| Database | OracleServer |  |
| Backend | Java, SPRING, MVC |  |
| WebConference | JMF |  |
|  |  |  |

# Source code structure

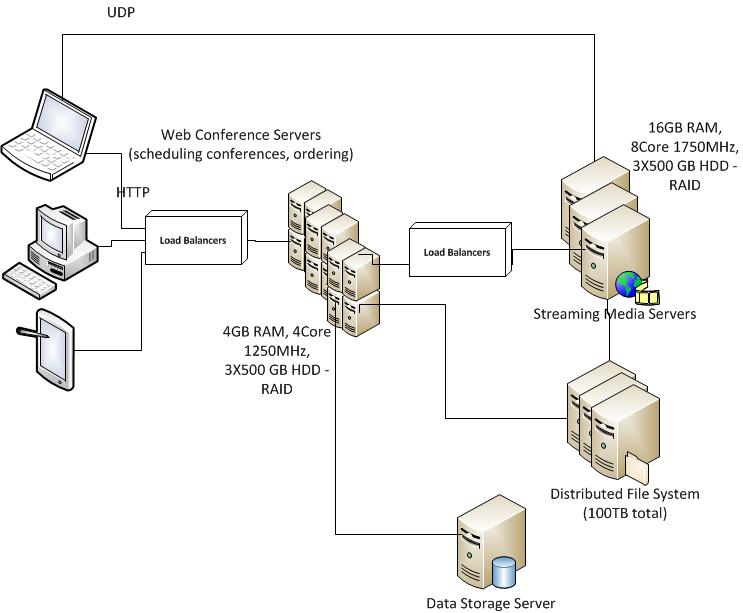
# Packaging, layering. Package diagram

# Testing guidelines

|  |  |  |
| --- | --- | --- |
| Deployable Unit | Components | Configuration options |
|  |  |  |
|  |  |  |
|  |  |  |

## Deployment View

# Deployment diagram



|  |  |  |
| --- | --- | --- |
| Node | Technology stack | Deployed Components |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Deployment options (if more than 1)

# Configuration

## Operational View

# Any operational considerations: monitoring, failure recovery, maintenance, upgrade.