**CSE325**

**Software Requirements Specification**

**Document**

**( Sahil Narula (12BCE0416) Poorva Arora (12BCE0277) )**

**Cloud Based Music Player**

**Version: (n)** **Date: (02/17/2015)**

**Table of Contents**

**1. Problem Statement 3**

**2. Background Information 3**

**2.1 Previously Known Applications 3**

**2.2 Limitations of Previously Known Applications 3**

**3. Stakeholders 4**

**3.1 Stakeholders for project 4**

**4. Requirements 4**

**4.1 Functional Requirements 4**

**4.2 Priority ID with Requirements 5**

**4.3 Non Functional Requirements 5**

**4.4 Use Case Diagram 7**

1. Problem Statement

*We are planning to build an online audio distribution platform that enables its user to access and create their favourite music collection remotely. Many cloud service provide generic or data-specific Cloud storage (e.g., Google, Picasa or SoundCloud).*

*Although both Cloud storage service types have the data storage in common, they present heterogeneous characteristics: different interfaces, accounting and charging schemes, privacy and security levels, functionality and, among the data-specific Cloudstorage services, different data type restrictions. This project focuses on p2p connection between client to server or client to client. Cloud data storage is used to maintain distributed index files.*

*Software clients in different operating system will use the same lower level restful APIs. So they can access their favourite music collection on the go. The multimedia database created or remote server will be categorized on the basis of various genres.*

**2. Background information**

*A wide variety of Cloud Services (CS) are available today, such as Amazon EC2 , SkyDrive , Google App Engine, or Dropbox. Both types of Cloud services — with generic and data-specific storage — show one common aspect: data is stored on CS’s servers.*

**2.1Previously Known Applications**

*There are a lot of Cloud Services available these days but applications that are similar to ours would be Sound Cloud, GrooveShark, Pandora because they are also providing music streaming as we are going to provide.*

**2.2****Limitations of Previously Known Applications**

*Both types of CSs (generic and data specific) are heterogeneous: they offer different APIs, different accounting and charging schemes, different privacy and security levels, different functionality, and, among generic Cloud storage services, they present different data type restrictions. This project will contain single entry point to access or store data but data is clustered using various indexing to access the data effectively.*

**3. Stakeholders**

*A stakeholder in an organisation can be defined as any group or individual who can affect or is affected by**the achievement of the organisation’s objectives.*

*Stakeholder can be any person either an individual or a group who can be affected by the particular project or has an interest in the project i.e. the development of the system software.*

**3.1 Stakeholders for project**

*The various stakeholders involved in our project are:*

* *Project team members (2 members in our team)*
* *Software developer and designer*
* *Project tester*
* *Project customers (clients)*
* *End users: Our software reaches a wide variety of users. It can be used by any user to send a vital piece of information to his/her friend. This software can be used by the government officials, private companies or daily users.*
* *Administrator who has the privileges to create, update or delete current records.*

**4. Requirements**

*A****software requirements specification****(SRS) is a description of a software system to be developed, laying out functional and non-functional requirements, and may include a set of use cases that describe interactions the users will have with the software.*

**4.1 Functional Requirements**:

As we know that functional requirements are the statement of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations. The various functional requirements of our software are listed below:

* *The system requires the user to first create an account on the widget in order to access the Cloud service.*
* *Dashboard for users to create or update playlist by uploading or deleting songs respectively.*
* *Proper admin panel for administrators where they can monitor the current usage traffic.*
* *A local music player to access the buffered songs even if the network is not available.*
* *Server spawns only one thread that consumes very less dyno/hours and still providing a reliable channel for data transfer.*
* *Admin has the privileges to modify or authorize the current users list.*
* *Admin has the privileges to create, modify and delete the current playlist.*
* *Admin can update users of the new tracks added and latest music buzz around the globe through RSS feed.*
* *Users will get a secure way to register and further login whenever needed*.

**4.2 Priority ID with Requirements**

1. *Login using user ID*
2. *Accessing the playlist on the server*
3. *Uploading, creating new playlist*
4. *Efficient storage in back-end*
5. *Tagging friends*
6. *Sending Recommendations*
7. *Changing the passwords*
8. *Deleting the account*
9. *Logout Mechanism*

**4.3 Non-functional Requirements:**

*These comprise of the constraints on the services or functions offered by the system such as timing constraints, constraints on the development process, standards, performance and platform requirements such as response time, scalability, usability, security, etc. The various non-functional requirements of our project are listed below:*

* ***Performance:***

*The system must be interactive and the delays involved must be less .So in every action-response of the system, there are no immediate delays. In case of opening windows forms, of popping error messages and saving the settings or sessions there is delay much below 2 seconds.*

*Also when connecting to the server the delay is based editing on the distance of the 2 systems and the configuration between them so there is high probability that there will be or not a successful connection in less than 20 seconds for sake of good communication.*

* ***Safety:***

*Information transmission should be securely transmitted to server without any changes in information.*

* ***Reliability:***

*As the system provide the right tools for discussion, problem solving it will be made sure that the system is reliable in its operations and for securing the sensitive details.*

* ***Availability:***

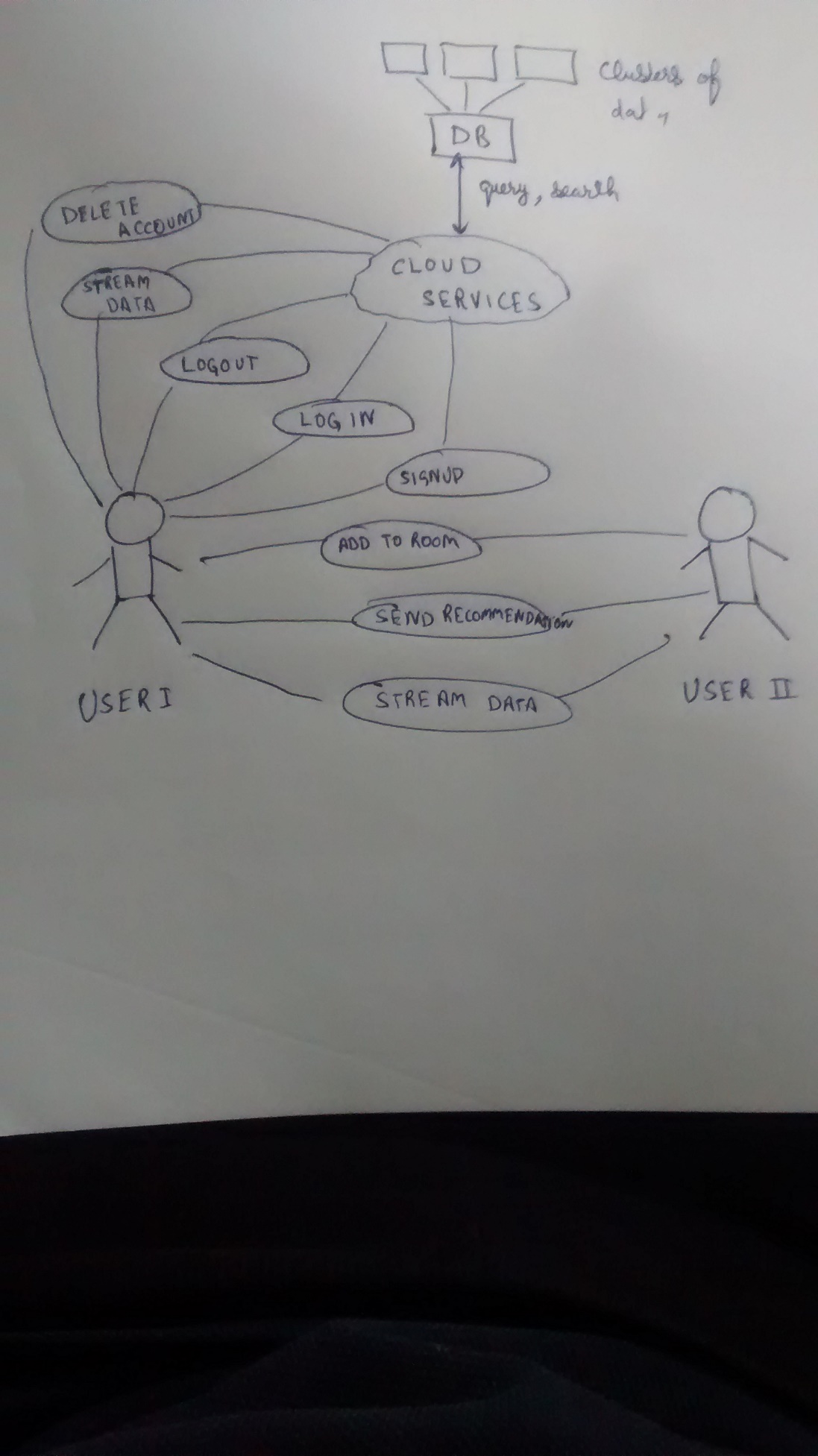
*24/7 Availability because the music player is working on remote server.*

* ***Security:***

*The main security concern is for users account hence proper login mechanism should be used to avoid hacking. The user id registration is way to spam check for increasing the security. Hence, security is provided from unwanted use of recognition software.*

* ***Usability:***

*As the system is easy to handle and navigates in the most expected way with no delays. In that case the system program reacts accordingly and transverses quickly between its states.*

**5. Use case diagram:**