StreamUP

MINOR PROJECT REPORT

Submitted for the partial fulfilment of the Degree

of

Bachelor of Technology

(Computer Science Engineering)

Submitted By:

Gursimran Singh 1410617

Daljit Singh 1410820

Abhay Yadav 1410797

Submitted To:

Prof. Sumeet Kaur Sehra

Project Guide

Department of Computer Science & Engineering

Guru Nanak Dev Engineering College

Ludhiana 141006

Abstract

A messaging app is a big undertaking. Making it needs the collaboration of many technologies and computing disciplines. This report represents the background study, technology used development, implementation and testing of a messaging app called StreamUP.

StreamUP is the official messaging App of Guru Nanak Dev Engineering College. It has been envisioned, designed and implemented exclusively for Genconians. The App is the one stop solution for all communication needs for the students and faculty members inside and out of the college campus. It is a full-fledged messaging App offering a complete, rich and innovative feature set. Users can use StreamUP to instantly find your fellow mates: juniors, seniors and all of the faculty. Users don't have to suffer through any enraging registration. Log in using your college ID and password and explore the college centric social network. Here are some of the App's most lavishing features:

- Chat using multiple accounts from a single copy of the App.
- Encrypt your messages over the medium.
- Automatic conversation back up on the server, sparing your precious phone storage.
- Pre-configured classification into groups for conference chats.
- Share files and documents of any type.
- Save your contacts on your phone and / or with your account and port them to any phone via login.

Acknowledgements

The authors are highly grateful to the Dr. Manohar Singh Saini, Director Guru Nanak Dev Engineering College (GNDEC), Ludhiana, for providing this opportunity to carry out the minor project work.

The constant guidance and encouragement received from Dr. Parminder Singh H.O.D. CSE Department GNDEC Ludhiana has been of great help in carrying out the project work and is acknowledged with reverential thanks.

We would like to express a deep sense of gratitude and thanks profusely to Project Guide Mrs. Sumeet Kaur Sehra, Without her wise counsel and able guidance, it would have been impossible to complete the project in this manner.

We express gratitude to other faculty members of Computer Science Engineering department of GNDEC for their intellectual support throughout the course of this work.

Finally, we are indebted to all whosoever have contributed in this report work.

Gursimran Singh

Daljit Singh

Abhay Yadav

Table of Contents

List of Figures

List of Tables

CHAPTER 1

1 INTRODUCTION

1.1 Introduction to Project

This Service is the one stop solution for all communication needs of the students and staff/faculty members of the college campus.

It is a web based client that communicates with the server using XMPP protocol (Extensible Messaging and Presence Protocol).

XMPP servers are used worldwide for communication, chatting and messaging over internet. A messaging app is a big undertaking. Making it needs the collaboration of many technologies and computing disciplines.

XMPP is the Extensible Messaging and Presence Protocol, a set of open technologies for instant messaging, presence, multi-party chat, voice and video calls, collaboration, lightweight middleware, content syndication, and generalized routing of XML data. XMPP was originally developed in the Jabber open-source community to provide an open, decentralized alternative to the closed instant messaging services at that time. XMPP offers several key advantages over such services:

The App is the one stop solution for all communication needs for the students and faculty members inside and out of the college campus. It is a full-fledged messaging App offering a complete, rich and innovative feature set. Users can use it to instantly find your fellow mates: juniors, seniors and all of the faculty. Users don't have to suffer through any enraging registration. Log in using your college ID and password and explore the college social network. Here are some of the App's most lavishing features:

1. Chat using multiple accounts from a single copy of the WebApp. 2. Encrypt your messages over the medium.

3. Pre-configured classification into groups for conference chats.

4. Share files and documents of any type.

5. Save your contacts on your phone and / or with your account and port them to any phone via login.

1.2 Project Category

StreamUP is an Internet based web application. It is basically a web client which can be used by any user such as a student, teacher or any college employee. By using this web client anyone can chat with any other person who has a college ID.

1.3 Objectives

It is a full-fledged messaging Web Application offering a complete, rich and innovative feature set.

- To provide users a platform where they can simply log in using their college ID and password and explore the college social network.
- Students can get direct notification from Placement cell about ongoing/coming placements through this web client.
- Teachers can directly share assignments to students and can keep in touch with the students even after the college hours.

1.4 Problem Formulation

The App is the one stop solution for all communication needs for the students and faculty members inside and out of the college campus. It is a full-fledged messaging App offering a complete, rich and innovative feature set. Users can use it to instantly find your fellow mates: juniors, seniors and all of the faculty. Users don't have to suffer through any enraging registration.

1.5 Identification Of Need

The need of the project is to bridge the communication gap between teachers and students. They can use this chatting app to send/receive text messages, documents , audio, video etc.

1.6 Existing System

There is no dedicated existing system for this purpose of college social network. Although students have their own Facebook and WhatsApp groups for communication with their friends and get knowledge about the activities going on in the college.

1.7 Proposed System

The analyst is concerned with the users of the system, identifying the objects and inheritance, and thinks about the events that change the state of objects. Analysis is a creative activity that involves understanding the problem, and its associated constraints, and methods of overcoming those constraints. This is an iterative process that goes on until the problem I well

understood. Analysis is a process by which we can identify classes that play a role in achieving system goals and requirements. Objectives of Project

Objectives of the project

It is a full-fledged messaging Web Application offering a complete, rich and innovative feature set.

- To provide users a platform where they can simply log in using their college ID and password and explore the college social network.
- Students can get direct notification from Placement cell about ongoing/coming placements through this web client.
- Teachers can directly share assignments to students and can keep in touch with the students even after the college hours.

1.8 Unique Features of the System

Features that this project going to provide:

- No need of registration.
- Automatic conversation back up on the server, sparing your precious phone storage.
- Save your contacts on your phone and / or with your account and port them to any phone via login.
- Chat using multiple accounts from a single copy of the App.
- Allow user those have two different accounts can keep both accounts active such as HOD can have a HOD account and can also have a teacher account.
- End-to-end encryption with OTR.
- Send and receive images as well as other kind of files.
- Indication when your contact has read your message.
- Pre-configured classification into groups for conference chats.

CHAPTER 2

2 REQUIREMENT ANALYSIS

2.1 Feasibility study

In order to provide such service at college level scale, there are two important things that are needed.

Application or client through user can interact and talk to others. Server that can handle the traffic that uses the service.

TECHNICAL FEASIBILITY

It is possible to provide a college social network and there are living examples of the same. Many private colleges and universities have their own network from where students can talk to other students and staff.

ECONOMICAL FEASIBILITY

With the help of some authorities, the project is affordable. This project is on time investment and very small effort is required to maintain the service up and running.

Since the software used for developing this software is open source, the project is highly economical.

Software Requirement Specification Document which must include the following:

(Data Requirement, Functional Requirement, Performance Requirement ,Dependability Requirement, Maintainability requirement, Security requirement, Look and feel requirement)

2.2 Software Requirement Specification

2.2.1 Functional Requirements

R.1. Students

- R.1.1 Login/Logout
- R.1.2 Join and Create Groups
- R.1.3 Private Chat with Authenticated Users
- R.1.4 Send/Receive Files/Photos
- R.1.5 Join Departmental Groups
- R.1.6 Update Status such as online/offline/away

R.2. Teachers

R.1.1 Login/Logout

R.1.2 Create Groups according to classes

R.1.3 Join and create new users

R.1.4 Private Chat with authenticated Users

R.1.5 Send/Receive Files

R.1.6 Update Details

2.2.2 Non-Functional Requirements

N.1 Databases

A database management system that is available free of cost in the public domain should be used.

N.2 Platform

It should be platform independent.

N.3 Web Browser

We should be able to software the product from any browser.

2.3 Expected Hurdles

Some Expected Constraints are:

- If the server is down due to any reason then the client cannot function.
- There could be internet connection problems.
- Different browsers may support this software differently.

2.4 SDLC model to be used

Iterative process starts with a simple implementation of a subset of the software requirements and iteratively enhances the evolving versions until the full system is implemented. At each iteration, design modifications are made and new functional capabilities are added. The basic idea behind this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).

Following is the pictorial representation of Iterative and Incremental model:

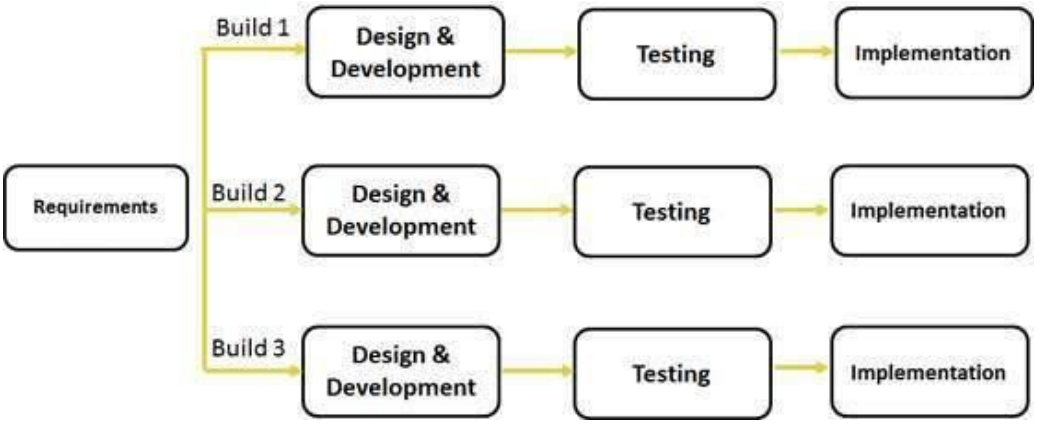

Figure 2-1 Iterative Waterfall Model

Iterative and Incremental development is a combination of both iterative design or iterative method and incremental build model for development. "During software development, more than one iteration of the software development cycle may be in progress at the same time." and "This process may be described as an "evolutionary acquisition" or "incremental build" approach."

In incremental model the whole requirement is divided into various builds. During each iteration, the development module goes through the requirements, design, implementation and testing phases. Each subsequent release of the module adds function to the previous release. The process continues till the complete system is ready as per the requirement.

CHAPTER 3

3 SYSTEM DESIGN

3.1 Design Approach

The product is an open source, community driven service which is licensed under the GNU general Public License. Product Aim is to provide an app offering a complete, rich and innovative feature set. User don't have to suffer through any enraging registration. They can login using their college ID and password and start using the service.

This service mainly is an implementation of client-server model. Where clients are: -

1. Android application dedicated to college centric social network.

2. Web client enable non-android users to interact and enjoy the services also.

And server is Ejabberd.

Ejabberd is an XMPP application server, written mainly in the Erlang programming language. It can run under several Unix-like operating systems such as Mac OS X, GNU/Linux, FreeBSD, NetBSD, OpenBSD and OpenSolaris.

Extensible Messaging and Presence Protocol (XMPP) is a communications protocol for message-oriented middleware based on XML (Extensible Markup Language). It enables the near-real-time exchange of structured yet extensible data between any two or more network entities. WhatsApp user the same server and protocol to provide their services.

3.2 DFD

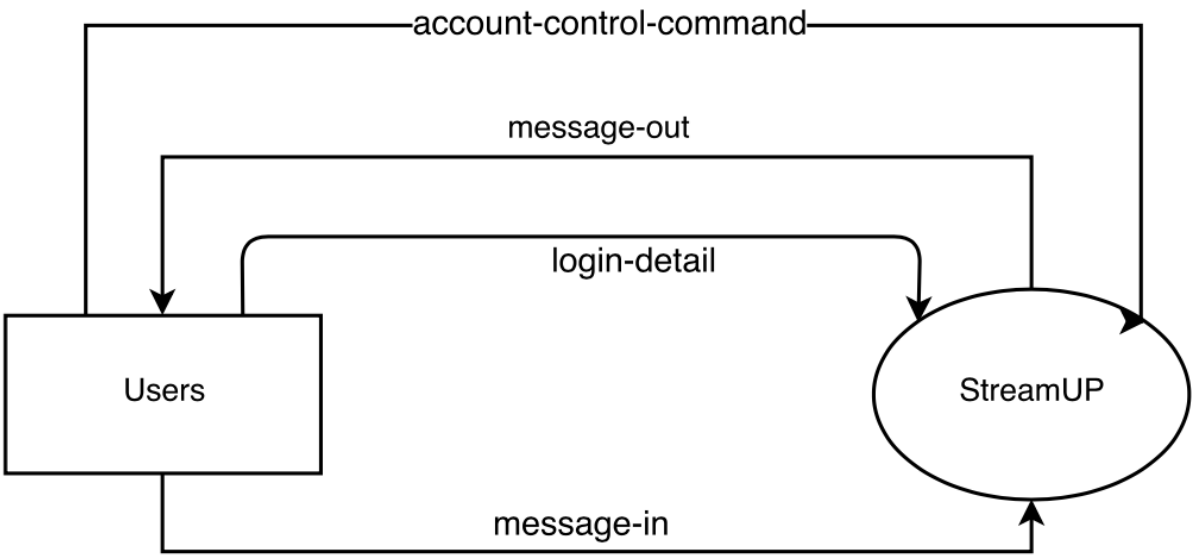

Figure 3-1 Level 0 DFD

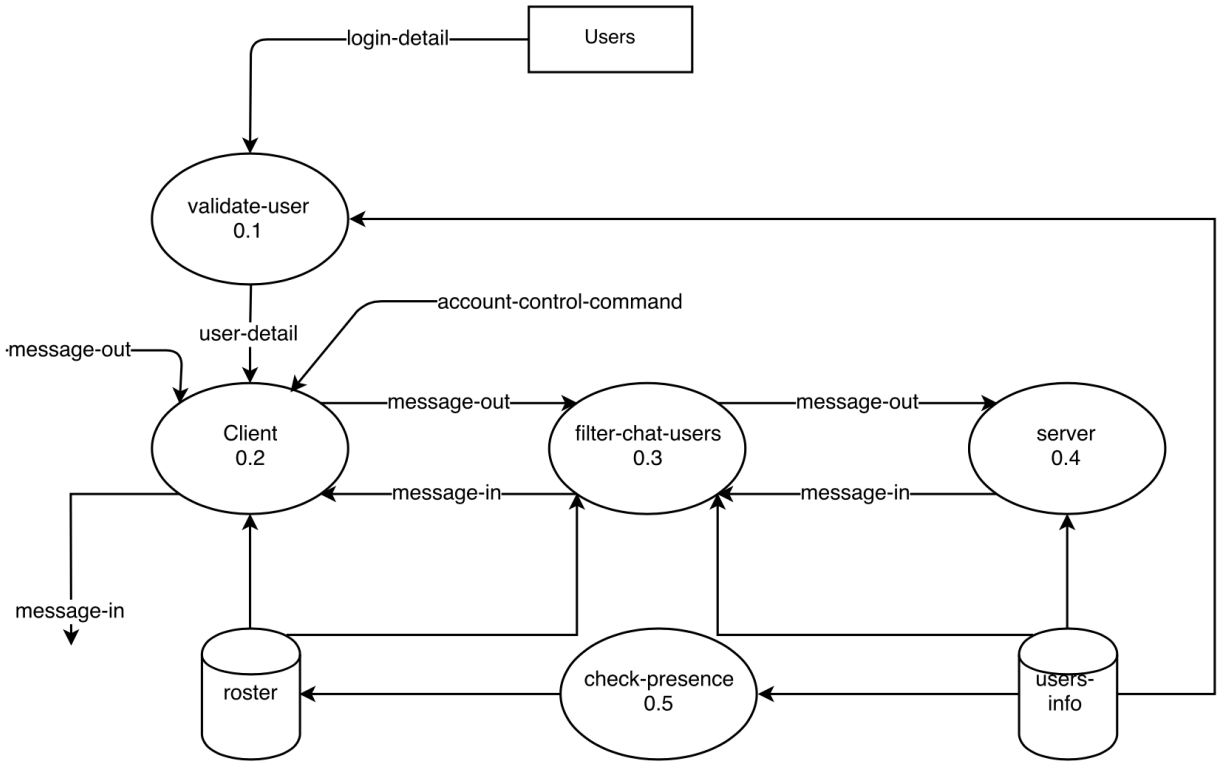

Figure 3-2 Level 1 DFD

3.3 Flow Chart

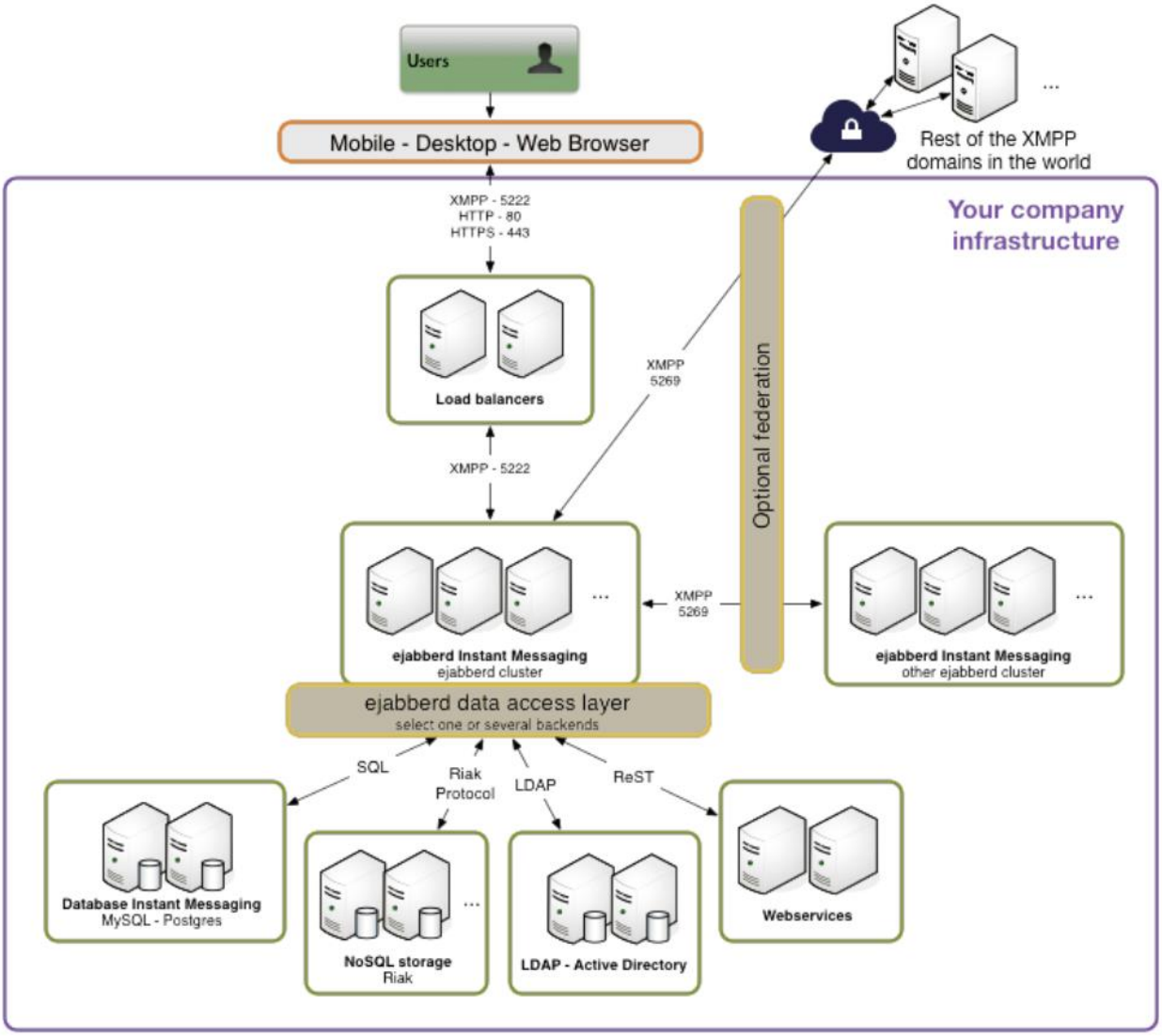

Figure 3-3 Flow Chart

3.4 Database Design

3.4.1 Table Structure

Ejabberd SQL database schema is design in order to maintain all the record about the users and messages.

Table users

Contains the information required to authenticate users.

Table 3-1Registered Users

Field	Type	Usage
username	string	User
password	string	User password, can be hashed
created_at	timestamp	When the user account was created

Table roster users

This is a quite complex table, used as a store for a quite complex protocol that is the one defined to manage rosters (groups) and subscriptions. In the common case of two users adding each other as contacts, entries in the roster table follows a series of steps as they moves from a subscription request to the final approval and bi-directional subscription being established. This process can be initiated either by the user, or by the (possible remote) peer.

Table 3-2 Users In a Group

Field	Type	Usage
username	String	User
jid	String	Contact jid
nick	String	Contact nickname
subscription	Char	'B'=both | 'T'=To | 'F'=From | 'N'=none
ask	char	'S'=subscribe | 'U'=unsubscribe | B='both' | 'O'=out | 'I'=in | 'N'=none
askmessage	string	Message to be displayed on the subscription request
server	char	'N' for normal users contacts
subscribe	string	
type	string	"item"
created_at	timestamp	Creation date of this roster entry

Table spool

Messages sent to users that are offline are stored in this table. Do not confuse this with general message archiving: messages are only temporarily stored in this table, removed as soon as the target user is back online and the pending messages delivered to it.

Table 3-3 Pending Messages

Field	Type	Usage
username	string	User
xml	blob	Raw packet
seq	integer	Unique, auto increment sequence number.
created_at	timestamp	When the message was stored

Table privacy_list_data

The table is used to store privacy rules.

Table 3-4 Privacy Settings defined By User

Field	Type	Usage
id	int	Privacy list rule id.
t	char	Privacy rule type: 'j' for jid, 'g' for group and's' for subscription.
value	string	Privacy list value for match, whose content depends on privacy list rule type.
action	char	Privacy list action 'd' for deny and 'a' for allow.
ord	int	Order for applying the privacy list rule.
match_all	boolean (0 or 1)	If true (1), means any packet types will be matched. Other matches should be false (0).
match_iq	boolean (0 or 1)	If true (1), means iq packets will be matched by rule.
match_message	boolean (0 or 1)	If true (1), means message packets type will be matched by rule.
match_presence_in	boolean (0 or 1)	If true (1), means inbound presence packets type will be matched by rule.
match_presence_out	boolean (0 or 1)	If true (1), means outbound packets type will be matched by rule.

Table muc_room

It is used to store persistent rooms, that is, rooms that must be automatically started with the server.

Table 3-5 Created Groups

Field	Type	Usage
name	string	Room name
host	string	Hostname of the conference component
opts	string	Room options, encoded as erlang terms
created_at	timestamp	Creation date

Table last

This table is used to store the last time the user was seen online.

Table 3-6 Last Seen Information

Field	Type	Usage
username	string	User
seconds	string	Timestamp for the last time the user was seen online
state	string	Why user got disconnected. Usually is empty

CHAPTER 4

4 IMPLEMENTATION, TESTING AND MAINTAINANCE

4.1 Introduction to Languages, IDE's, Tools and Technologies used for Implementation

4.1.1 Languages

HTML

Figure 4-1HTML Logo

Hyper Text Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS), and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web.[1] Web browsers receive HTML documents from a webserver or from local storage and render them into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects, such as interactive forms may be embedded into the rendered page. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as <img /> and <input /> introduce content into the page directly. Others such as ... surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page. HTML can embed programs written in a scripting language such as JavaScript which affect the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

CSS

Figure 4-2 CSS Logo

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

The CSS specification describes a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called cascade, priorities (or weights) are calculated and assigned to rules, so that the results are predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents.

MYSQL

Figure 4-3 Mysql Logo

MySQL is an open-source relational database management system (RDBMS).[6] Its name is a combination of "My", the name of co-founder Michael Widenius' daughter,[7] and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.[8] For proprietary use, several paid editions are available, and offer additional functionality. MySQL is a central component of the LAMP

open-source web application software stack (and other "LAMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python".

JAVASCRIPT

JavaScript is a high-level, dynamic, untyped, and interpreted programming language.[6] It has been standardized in the ECMAScript language specification. Alongside HTML and CSS, it is one of the three core technologies of World Wide Web content production; the majority of websites employ it and it is supported by all modern Web browsers without plug-ins. JavaScript is prototype-based with first-class functions, making it a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles.It has an API for working with text, arrays, dates and regular expressions, but does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded.

Although there are strong outward similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two are distinct languages and differ greatly in their design. JavaScript was influenced by programming languages such as Self and Scheme.

JavaScript is also used in environments that are not Web-based, such as PDF documents, site-specific browsers, and desktop widgets. Newer and faster JavaScript virtual machines (VMs) and platforms built upon them have also increased the popularity of JavaScript for server-side Web applications. On the client side, JavaScript has been traditionally implemented as an interpreted language, but more recent browsers perform just-in-time compilation. It is also used in game development, the creation of desktop and mobile applications, and server-side network programming with run-time environments such as Node.js.

BOOTSTRAP

Bootstrap is a free and open-source front-end web framework for designing websites and web applications. It contains HTML- and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Unlike many web frameworks, it concerns itself with front-end development only. Bootstrap is the second most-starred project on GitHub, with more than 100,000 stars and 45,000 forks.

Strophe.js

Strophe.js is an XMPP library for JavaScript. Its primary purpose is to enable web-based, real-time XMPP applications that run in any browser.

This library uses either Bidirectional-streams Over Synchronous HTTP (BOSH) to emulate a persistent, stateful, two-way connection to an XMPP server or alternatively WebSockets.

BOSH (protocol)

Bidirectional-streams Over Synchronous HTTP (BOSH) is a transport protocol that emulates a bidirectional stream between two entities (such as a client and a server) by using multiple synchronous HTTP request/response pairs without requiring the use of polling or asynchronous chunking.

For applications that require both "push" and "pull" communications, BOSH is significantly more bandwidth-efficient and responsive than most other bidirectional HTTP-based transport protocols and AJAX. BOSH achieves this by avoiding HTTP polling, yet it does so without resorting to chunked HTTP responses as is done in the technique known as Comet. To date, BOSH has been used mainly as a transport for traffic exchanged between Jabber/XMPP clients and servers (e.g., to facilitate connections from web clients and from mobile clients on intermittent networks).

For "push", a BOSH client starts an HTTP request, but the server postpones sending a reply until it has data to send. After receiving a reply, the client immediately makes another request on the same HTTP connection, so the server can always send data to the client without waiting for the client to poll. If, while waiting for a reply, the client needs to send data to the server, it opens a second HTTP connection. There are at most two HTTP connections open at a time, one on which the server can send data as a reply and one on which the client can send data as a POST.

"Bidirectional-streams Over Synchronous HTTP (BOSH)" and "XMPP over BOSH" are draft standards of the XMPP Standards Foundation.

The second related standard XMPP Over BOSH (XEP-0206) defines how BOSH may be used to transport XMPP stanzas. The result is an HTTP binding for XMPP communications that is intended to be used in situations where a device or client is unable to maintain a long-lived TCP connection to an XMPP server.

4.1.2 Tools

XAMPP

XAMPP is a free and open source cross-platform web server solution stack package developed by Apache Friends,[2] consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). It is a simple, lightweight Apache

distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application (Apache), database (MariaDB), and scripting language (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well.

EJABBERD

Figure 4-4 Ejjabird XMPP Server

ejabberd is an XMPP application server, written mainly in the Erlang programming language. It can run under several Unix-like operating systems such as Mac OS X, GNU/Linux, FreeBSD, NetBSD, OpenBSD and OpenSolaris. Additionally, ejabberd can run under Microsoft Windows. The name ejabberd stands for Erlang Jabber Daemon (Jabber being a former name for XMPP) and is written in lowercase only, as is common for daemon software.

Reason to choose Ejabberd:-

- FULLY OPEN SOURCE: - Everyone has the freedom to use and customize the ejabberd code, according to the GPLv2 license.
- SECURE: - Best practices are baked right into the server. Secure code runs in a trusted environment, with all SSL / TLS encryption best practices.
- FLEXIBLE: - ejabberd offers a full API to write your custom plugins and modify the server so that it works exactly as you wish, with a minimal amount of code.
- INTEROPERABLE: - ejabberd is compliant with the XMPP standard and most of the available extensions. It can be leveraged with all the available XMPP clients and libraries and can federate with other servers.
- PROFESSIONALLY MAINTAINED: - Professional release engineers manage the ejabberd release cycle, QA the full stack, and keep APIs stable. The core team has impressive credentials and 16 years of Erlang development under their belt.

- WARM & WELCOMING: - Ejabberd's helpful, kind, and supportive community spans the globe. Ejabberd's mission is to empower everyone to use and build services on top of the XMPP protocol, and more.

JSXC

Figure 4-5 JSXC Logo

XMPP, the protocol used by JSXC , identifies every user by a JID. This JID has the same format as an email address: a local part (typically user name) and a domain name, separated by @. The advantage of this notation is that organizations providing both email (typically asynchronous) and XMPP (typically synchronous) communication, can provide both services with one mnemonic ID. This allows for integrated and unified communication, where a single ID can be used for email, instant messaging, and audio/video communications; thus alleviating the need for users to remember multiple IDs, such as (non-mnemonic) telephone numbers, video conference IDs, and so on.

Even though both XMPP and email use the same address format, they are distinct services, not unlike fax, text messages (SMS), and voice telephony using a single telephone number format, even though the services differ in what they exchange, the protocol, and the typical uses.

Unlike proprietary instant messaging services, XMPP allows anyone operating a domain and a server to provide XMPP service for its users. XMPP was designed from the start to allow these servers to easily interoperate, making it easy for userA@serverA to communicate with userB@serverB, as if they were on the same server. This federated design enables privacy from the start, there is no central authority which knows what you are doing when and who your friends are. Only servers with a need to know (because they host an account for one of your friends) will ever know anything about you; third parties remain ignorant.

Sublime Text Editor

Figure 4-6 Sublime Logo

Sublime Text is a proprietary cross-platform source code editor with a Python application programming interface (API). It natively supports many programming languages and markup languages, and its functionality can be extended by users with plugins, typically community-built and maintained under free-software licenses.

Grunt

Figure 4-7 Grunt Logo

Grunt is a JavaScript task runner, a tool used to automatically perform frequently used tasks such as minification, compilation, unit testing, linting, etc. It uses a command-line interface to run custom tasks defined in a file (known as a Gruntfile). Grunt was created by Ben Alman and is written in Node.js. It is distributed via npm. Presently, there are more than five thousand plugins available in the Grunt ecosystem.

Git

Figure 4-8 Git Logo

Git is a version control system (VCS) for tracking changes in computer files and coordinating work on those files among multiple people. It is primarily used for software development,[9] but it can be used to keep track of changes in any files. As a distributed revision control system it is aimed at speed,data integrity,and support for distributed, non-linear workflows. Git was created by Linus Torvalds in 2005 for development of the Linux kernel, with other kernel developers contributing to its initial development. Its current maintainer since 2005 is Junio Hamano.

As with most other distributed version control systems, and unlike most client–server systems, every Git directory on every computer is a full-fledged repository with complete history and full version tracking abilities, independent of network access or a central server.

Github

Figure 4-9 Github Logo

GitHub is a Git repository web-based hosting service which offers all of the functionality of Git as well as adding many of its own features. Unlike Git which is strictly a command-line tool, GitHub provides a web-based graphical interface and desktop as well as mobile integration. It also provides access control and several collaboration features such as wikis, task management, and bug tracking and feature requests for every project.

Projects on GitHub can be accessed and manipulated using the standard Git command-line interface and all of the standard Git commands work with it. GitHub also allows registered and non-registered users to browse public repositories on the site. Multiple desktop clients and Git plugins have also been created by GitHub and other third parties that integrate with the platform.

The site provides social networking-like functions such as feeds, followers, wikis and a social network graph to display how developers work on their versions of a repository.

Web Browser

Figure 4-10 Chrome Logo

Google Chrome is a freeware web browser developed by Google. It was first released in 2008, for Microsoft Windows, and was later ported to Linux, macOS, iOS and Android. Google Chrome is also the main component of Chrome OS, where it serves as a platform for running web apps. Google releases the majority of Chrome's source code as the Chromium open-source project. A notable component that is not open-source is the built-in Adobe Flash Player (that Chrome has disabled by default since September 2016. Chrome used the WebKit layout engine until version 27. As of version 28, all Chrome ports except the iOS port use Blink, a fork of the WebKit engine.

4.2 Project Scheduling

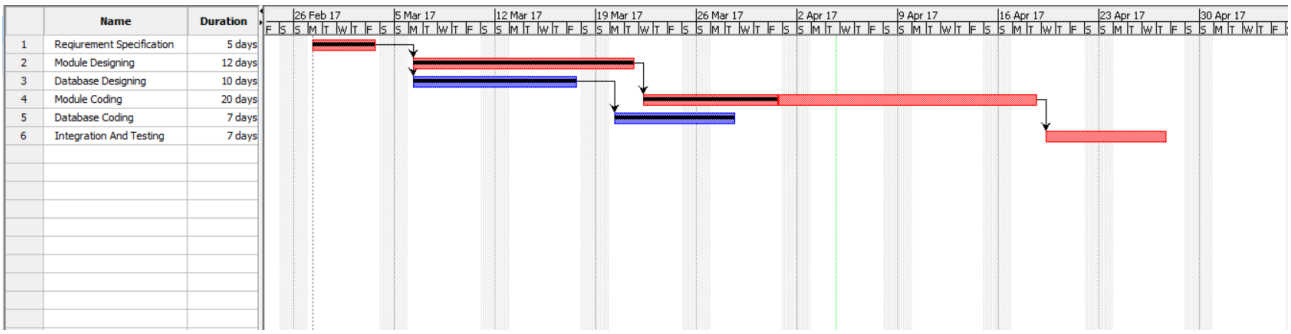

Figure 4-11 Gantt Chart

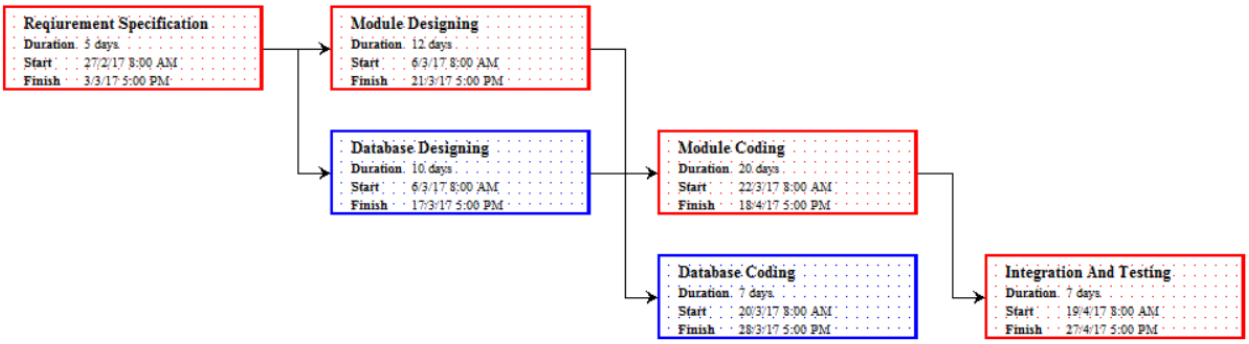

Figure 4-12 Pert Chart

CHAPTER 5

5 Results and Discussions

5.1 Snapshots of the system with brief details

5.1.1 General Information

XMPP, the protocol used by STREAMUP, identifies every user by a JID. This JID has the same format as an email address: a local part (typically user name) and a domain name, separated by @. The advantage of this notation is that organizations providing both email (typically asynchronous) and XMPP (typically synchronous) communication, can provide both services with one mnemonic ID. This allows for integrated and unified communication, where a single ID can be used for email, instant messaging, and audio/video communications; thus alleviating the need for users to remember multiple IDs, such as (non-mnemonic) telephone numbers, video conference IDs, and so on.

Even though both XMPP and email use the same address format, they are distinct services, not unlike fax, text messages (SMS), and voice telephony using a single telephone number format, even though the services differ in what they exchange, the protocol, and the typical uses.

5.1.2 XMPP Server (Ejabberd) implementation

Ejabberd is XMPP server software which can be used for instant messaging on your own server. Following are the steps to properly install and configure ejabberd on Ubuntu Server.

Login to your server via SSH as root.

Step 1 - Install ejabberd

Run the command: - apt-get -y install ejabberd

Step 2 - Setup Admin User

Run the command: - ejabberdctl register admin localhost "secret password"

Step 3 - Give Admin Privileges

By default, hostname used by ejabberd is 'localhost', which can be modified from config file.

For our example we will call our admin user "admin@localhost" and modify the following lines in /etc/ejabberd/ejabberd.cfg:

%% Admin user

{acl, admin, {user, "admin", "localhost"}}.

%% Hostname

{hosts, ["localhost"]}.

Step 4 - Restart ejabberd

Run the command: - service ejabberd restart

Now you can navigate to the ejabberd Web Admin interface on server's IP address port 5280

Now we can user a XMPP client like sunehaG to connect to XMPP server.

Use port 5222 to connect to your XMPP server

Using the LDAP server to perform authentication:

Domain example.net is using the internal authentication method while domain example.com is using the LDAP server running on the domain localhost to perform authentication:

ejabberdctl

With the ejabberdctl command line administration script you can execute ejabberdctl commands

Ejabberdctl Commands

- ejabberdctl start: Start ejabberd in background mode.
- ejabberdctl debug: Attach an Erlang shell to an already existing ejabberd server. This allows to execute commands interactively in the ejabberd server.
- ejabberdctl live: Start ejabberd in live mode: the shell keeps attached to the started server, showing log messages and allowing to execute interactive commands.
- ejabberdctl help: Get help about ejabberdctl or any available command. ejabberdctl status: Check the status of the ejabberd server.

- ejabberdctl stop: Stop the ejabberd server.
- ejabberdctl restart: Restart the ejabberd server.
- ejabberdctl mnesia: Get information about the Mnesia database.

5.1.3 Login Screen

For this section, we assume that you are using STREAMUP with unified login, i.e., logging in to your web application such as SOGo or ownCloud will also log you into your STREAMUP XMPP Chat. This is the default and recommended way to run STREAMUP. After successful login, the right hand side of your application will show the STREAMUP bar. During the first login with a browser, the STREAMUP bar will display that it is creating your security key. After a short while, the display will change to your roster. Already during key generation, you can use the main application as before. STREAMUP functions will only be available after key generation finishes. The key is safely stored in your browser and not sent to any server.

If you would like to use more screen real estate for your application, you can resize or hide the STREAMUP bar by dragging or clicking the vertical dividing line to the main application. A hidden STREAMUP bar can be restored by clicking on the right edge of the window. Your web browser will remember the bar visibility.

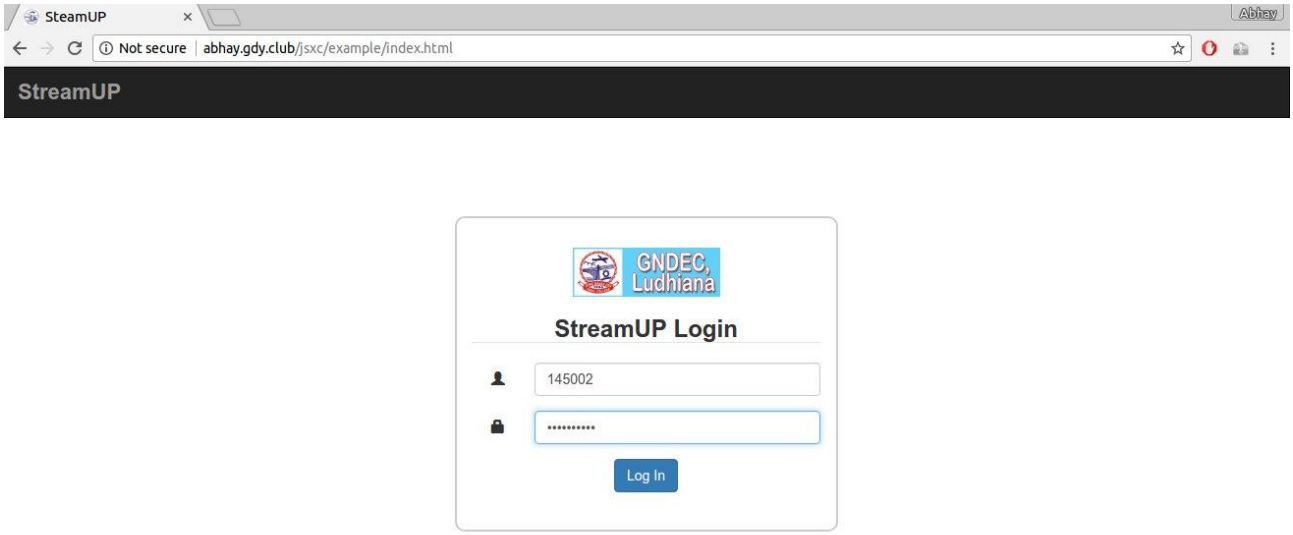

Figure 5-1 Login Screen

5.1.4 Roster Management

If you have previously used XMPP for that account, e.g., from another STREAMUP-enabled application or a native XMPP/Jabber client, your roster — the list appearing in the StreamUP bar after the login — will already be populated with the contacts. As with all XMPP clients, the roster is stored on the XMPP server, so your contact list will be available to any XMPP client, including StreamUP.

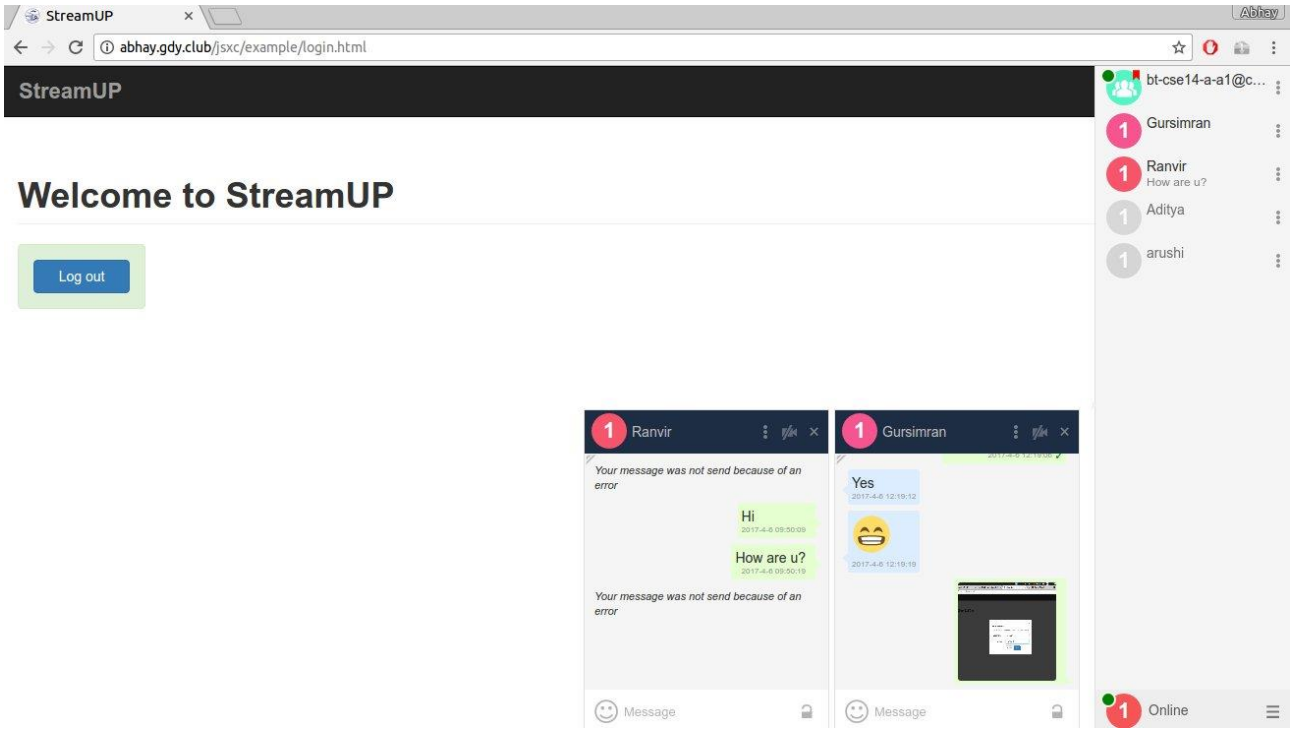

Figure 5-2 Roster Management

1. Adding a Contact

To add a contact, click on the gear icon at the bottom right corner and select Add Buddy; then enter the JID (typically the email address) of your buddy, optionally with a (nick) name under which the buddy should be shown in your roster. This nick name is only visible to you; so feel free to make them easy for you to remember. If no full name is given, then the JID will be displayed.

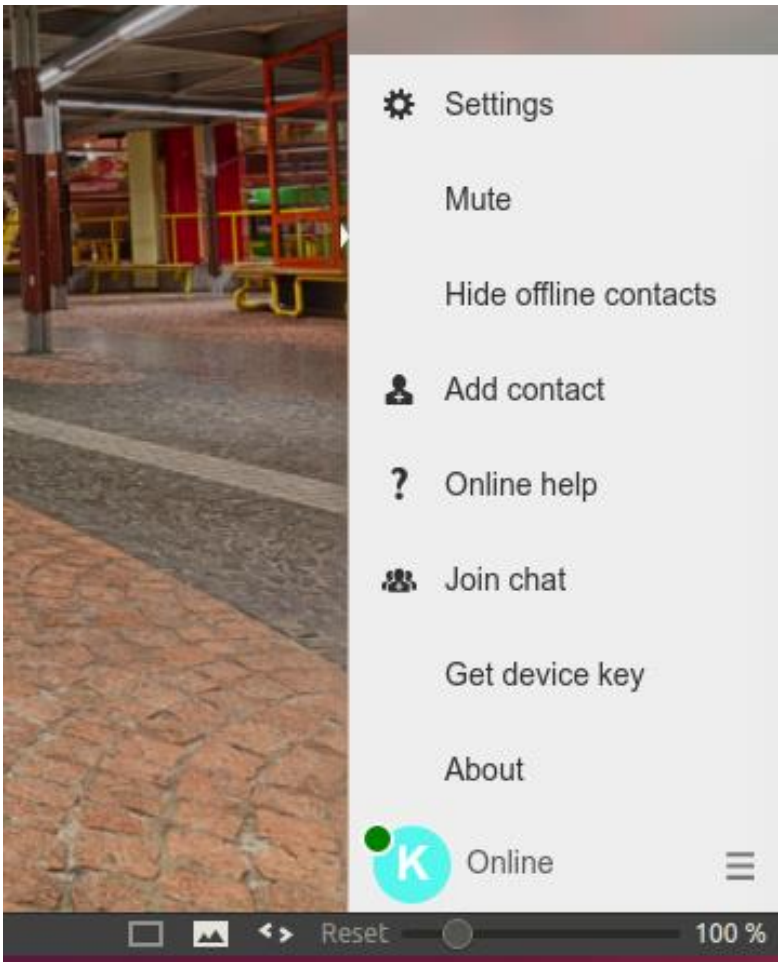

Figure 5-3 Add Contacts

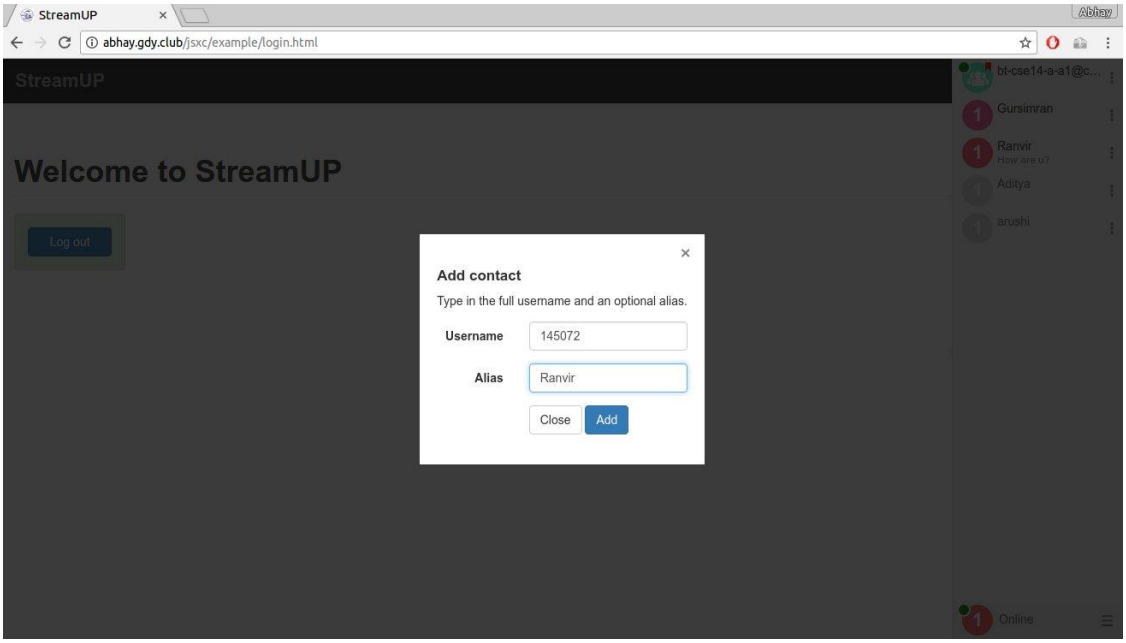

Figure 5-4 Adding Contacts

The person you just added will receive a friendship notification in their chat client. If they are offline, this request will wait for them to go online.

Once you have a lot of contacts and only want to see those who are online, you can select Hide Offline from the gear icon menu.

2. Modifying a Contact

When hovering the mouse over a contact, two icons will float in from the right: A pen to edit the name, and a cross to delete it. Clicking on the pen icon will make the name editable. Hitting return will change the entry in your roster. The name you entered will be active on all your XMPP clients connecting to this account. Other users, including the contact, will not know what you have entered there.

Figure 5-5 Modifying a Contact

If you empty the nickname, the JID will be displayed. This can also be helpful if you played around with nicknames so much that you confused yourself.

5.1.5 Notifications

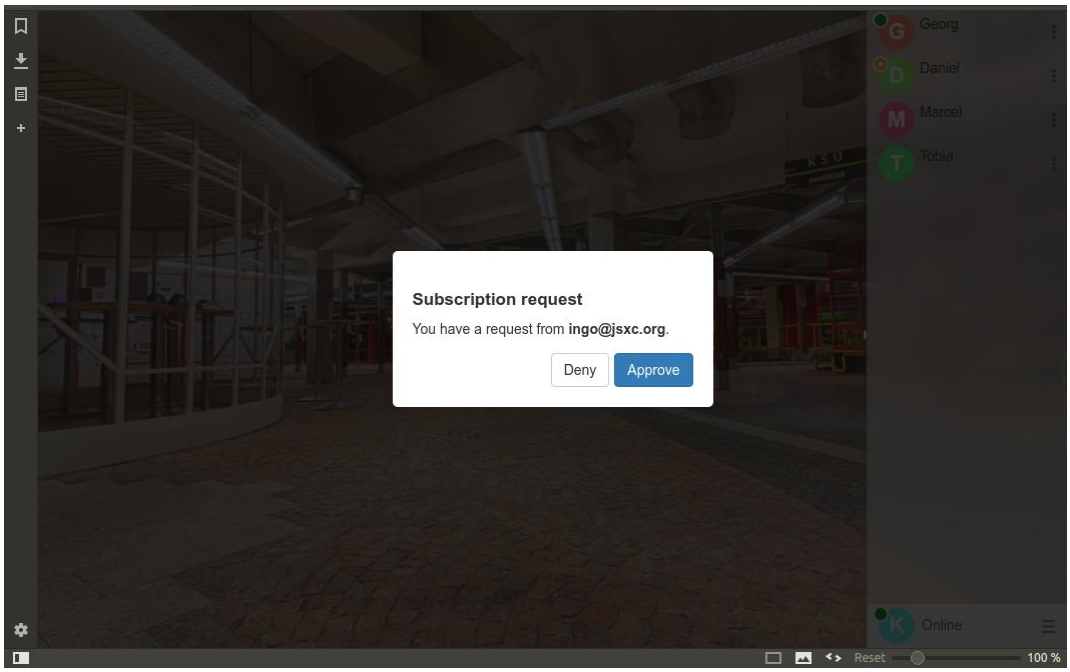

Figure 5-6 Notifications

In a chat, several events can happen. For example, someone may add you to their roster and would like to your presence status and thus be able to see whether you are online. Such events are listed at the bottom right with the pending event count on yellow background. Clicking on it reveals the requests, which you can then handle, e.g., deny or accept the friendship request.

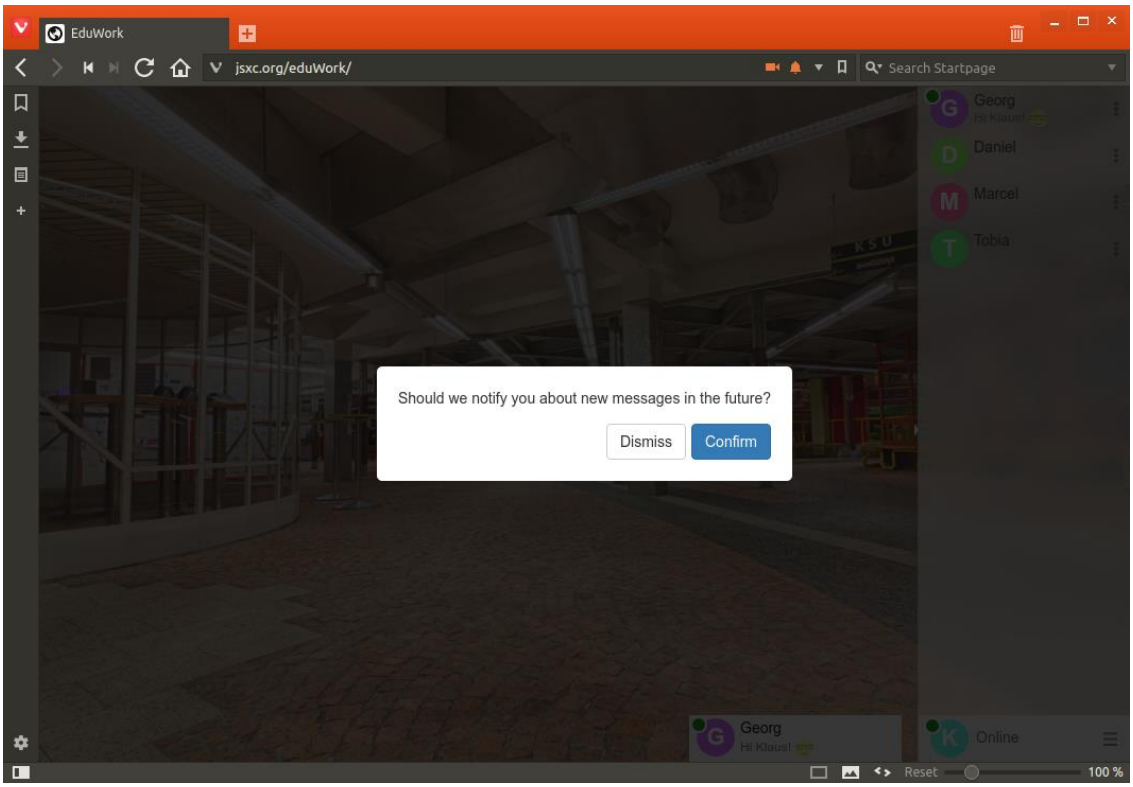

Figure 5-7 Future Notifications

5.1.6 Setting Your Status

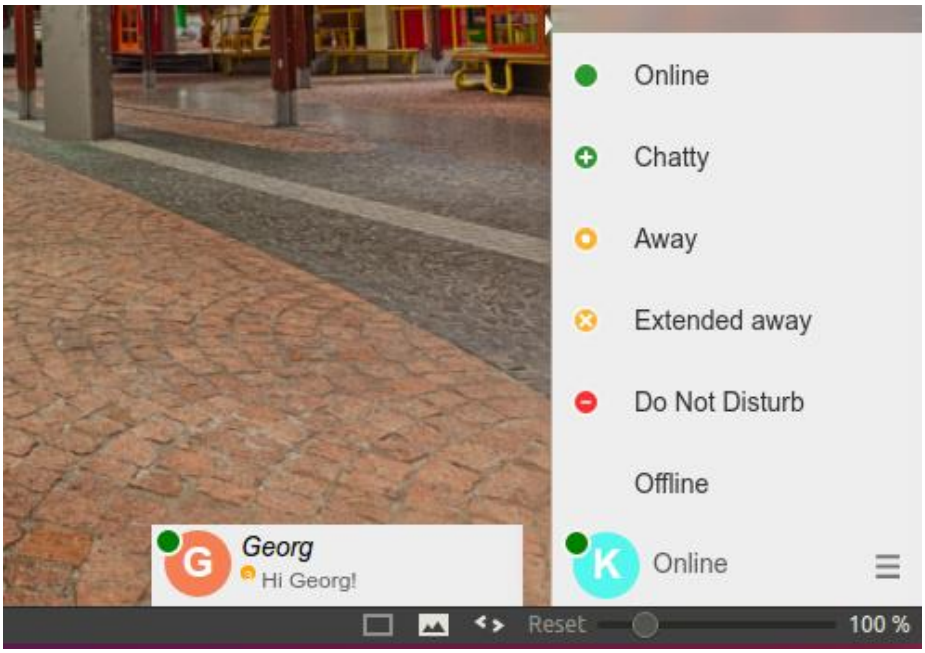

Figure 5-8 Status Setting

You can tell your contacts whether you are available, e.g. by setting the status to away or do not disturb. Your contacts will see this by the coloured bar left of their roster entry for you which will change to yellow or red as a result of this choice.

5.2 Join Chat Rooms

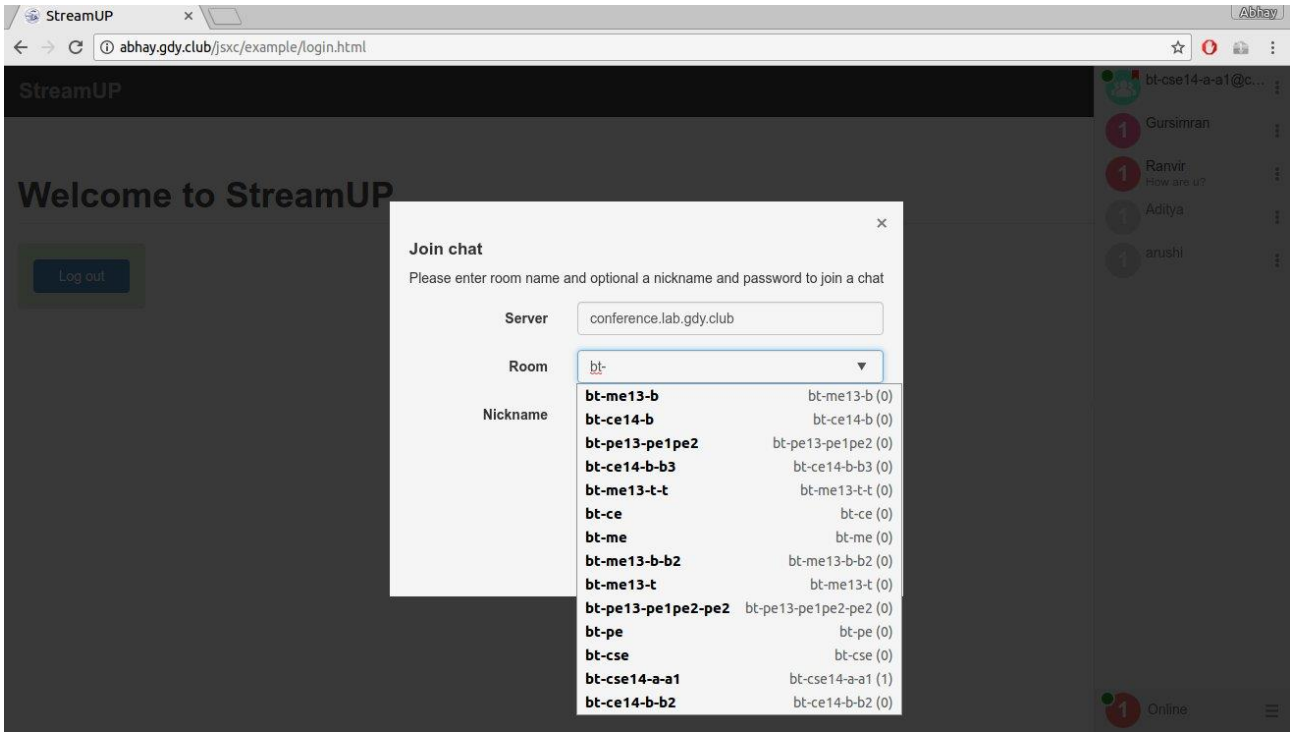

Figure 5-9 Join Chat Rooms or Groups

Chat rooms are made according to departments and these can be joined through this option.

5.2.1 Chatting

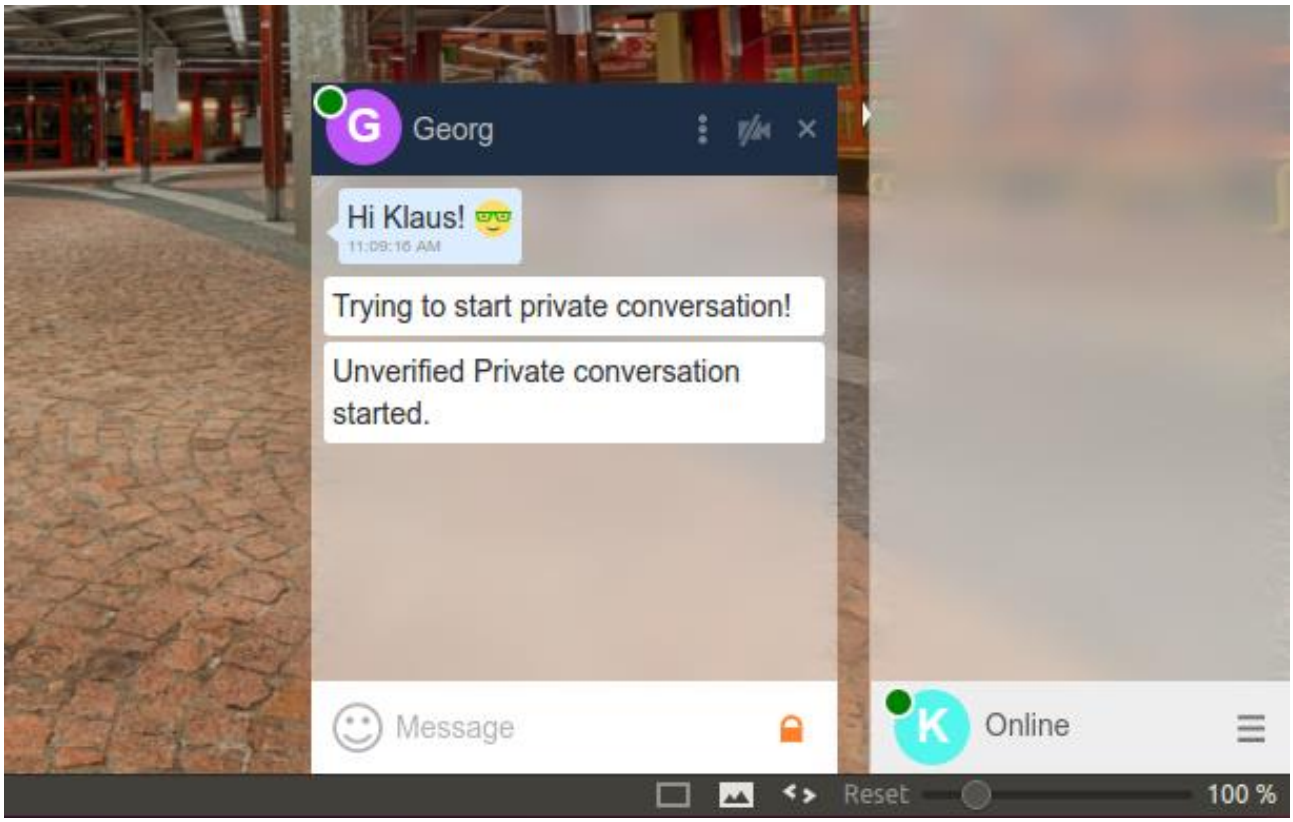

Figure 5-10 Chatting Module

Just click on one of your contacts and a chat window will open. Click on another contact, and – yes, a second chat window will open. If they are using up too much screen real estate, click at the bottom of a chat window to minimize it or on the X on the top right to close it. The bottom bar of a chat window will flash even when minimized.

Figure 5-11 Minimise Chat

5.3 Back End Representation

5.3.1 Snapshots of Database Tables with brief description

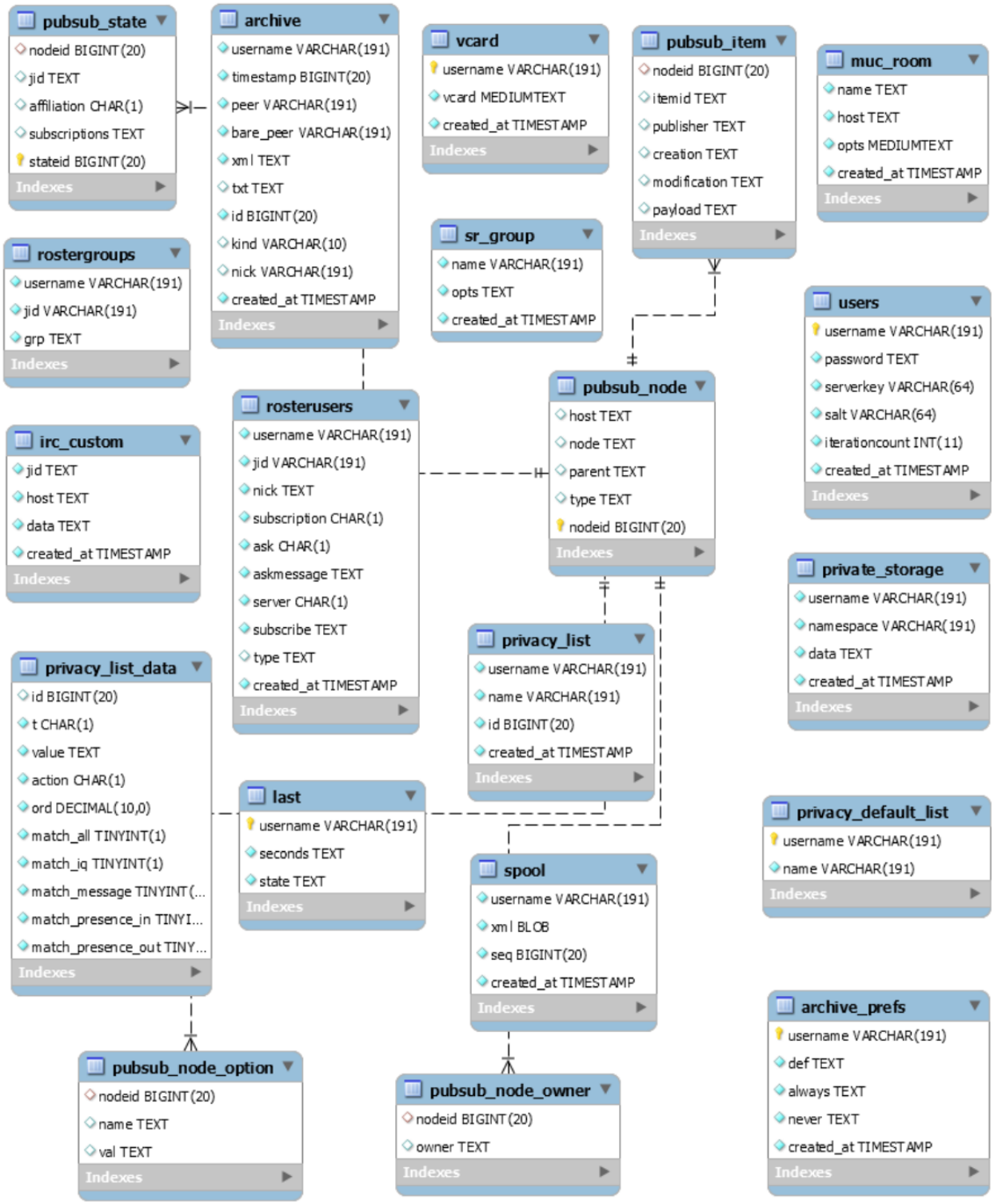

Figure 5-12 Back End Representation

6 Conclusion and Future Scope

To summarise in computing, a web application or web app is a client–server software application in which the client runs in a web browser. In this project a web application is being developed which runs on the client-server modal. This application can be useful in colleges and similar environments where there is a need for constant communication between teachers and students.

There is a lot that can be still possible in this project.

There are some features that can be added in the project like: -

- Calling both Voice and Video.
- Inbuilt Document Viewer to view shared document.
- Real-time Document editing integration that work like google docs and help user to share documents and edit them in real-time.
- Music Player integrated in the application that remove the need to have a third party music app to play music shared via this application
- A Chatbot can be added which can answer common student questions such as exam dates, last fee deposit dates, project deadline dates, holidays and other queries.

References

[1] Klaus, "STREAMUP," Dec 2017. [Online]. Available: https://www.StreamUP.org.

[2] Peter, "XMPP," 2017. [Online]. Available: https://xmpp.org.

[3] A. Shchepin, "Ejabbird," 2017. [Online]. Available: https://www.ejabberd.im/.

[4] "Stackoverflow," 2017. [Online]. Available: http://stackoverflow.com/.