
Computer Networks: UE16CS301

Ishita Bhandari: 01FB16ECS143

Kavya Varma: 01FB16ECS162

Kusumanjali Ramesh: 01FB16ECS173

KIK: Text Application

OVERVIEW

KIK is a messaging application that allows the sending and viewing of text messages and images.

The user can add profile picture, status, add friends, send messages, etc.

The service requires the clients to provide Name and Password.

It uses TCP.

Programming language used: Python.

GUI

We have used Tkinter.

Tkinter is the standard GUI library for python.

Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

Python when combined with Tkinter provides a fast and easy way to create GUI Applications.

MODULES (14)

Authenticating Login:

Client sends the username and password to the server. The server checks if the username and password exists in its database. If it exists then it logs in the user and shows the user's their list of friends. Otherwise server tells client that user does not exist, in turn interface asks user to check and retype his information.

Sign Up:

Client takes input from user and server collects it and puts this in its database. This is reflected in the client. Upon signing in, it creates a user account giving each user a unique user id.

Check if Username Exists:

Client sends the server the username, server goes through its entire database. If the username already exists it returns username already used and asks the client for a new one..

Get Contacts:

When user makes a request for its own contacts, the server sends the required information back.

Add Contacts:

Client sends the uid of a new contact to be added as their friend. The server in turn updates its database

Validity of User ID :

When one user is trying to add another user as their friend, the client sends this uid to the server. The server checks for this uid in its database and returns valid if uid from client matches with any of the uid's from there server's database.

Send Message:

Client collects data from user. through tcp send function. The entire data goes to the server as string and is stored in a message data structure.

View Conversation:

Every pair of users has a conversation data structure on the server which contains the message ids of all the messages of that conversation. When a user asks for the conversation the messages corresponding to these message ids are retrieved.

Automatic Updation of Chat Box:

When a user has a conversation with another user open, and a new message is sent on this chat box, the server sends a message saying that a change has occurred.

Notification:

When a user receives a new message, a notification is sent from the server.

Add Profile Picture :

User can choose the picture to be uploaded from a dropdown box. The client then sends the picture to the server (base64 encoding). The server then decodes the image and stores it.

Get Profile Picture:

When a user requests to see his friend's profile picture, server sends profile picture to the friend/client.

Add Status :

This function is called from client to server, it adds the user's status on the server

Get Status:

When a user requests to see his friend's status, the client sends a request(uid of friend) to the server, the server fetches the status and returns it to the client.

Project Output:

The application successfully establishes a conversation between two users. It manages to create dynamically changing chat boxes along with notification. Logging in and signing up has also been successfully implemented with each user being defined by their username, profile picture and status.