

Q&A System

This platform enables users to ask questions and receive textual answers. It is designed to faster learning, collaboration, and knowledge discovery. It streamlines information access, analyzes user sentiment, and enables feedback.

Meet Our Team (Group 2):

- Nekkala Sumanth Naidu [23CS8111]
- Narreddy Balaiahgari Sai Keerthana [23CS8112]
- Ganapathri Praphul Chandra [23CS8113]
- Authy Saha [23CS8114]
- Tanusha Pramanik [23CS8115]
- Suryam Kumar Gautam [23CS8116]
- Sumit Kumar [23CS8117]
- Anurag Prasad [118]
- Gavirni Venkata Satya Krishna Chaitanya [23CS8119]
- Yaraveda Govardhan Reddy [23CS8120]

02 Q&A System

Problem Description

- Develop a Q&A system platform where questions can be posted and answers (textual) may be received.

Primary Objectives:

- Enable basic search on the Q&A archive
- Analyze sentiment of an answer
- Assign credits/ratings to registered members
- Use of the platform for receiving class feedback

Secondary Objectives:

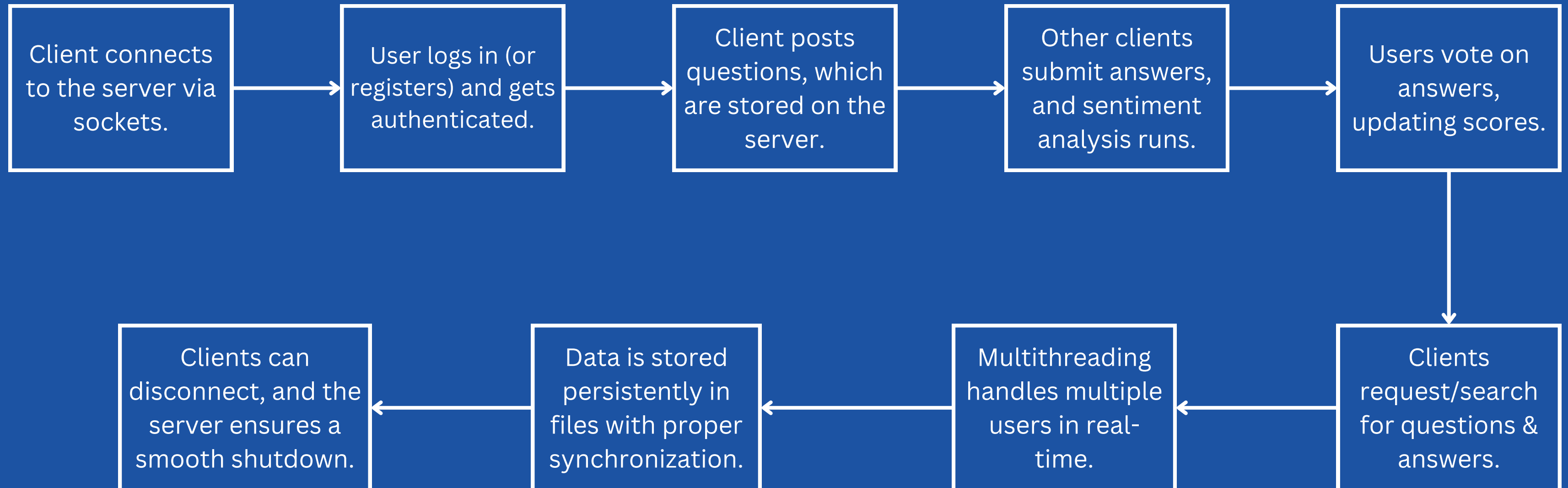
- Enables users to log in from different terminals from different physical machines
- Enables Manager/User registration and authentication (using some hash-based password)
- A user-friendly text interface
- A multithreaded platform



System Workflow



Client-Server Interaction



Text-Interface for Easy Understanding

```
1. Register
2. Login
Choice: 1
Username: tanusha1235
Password: 1235
Registration successful
```

```
1. Ask a Question
2. Answer Questions
3. Search Questions
4. Delete Content
5. Exit
Choice: 1
Enter your question: what is deadlock?
Question posted successfully with ID: 4
```

```
Available Questions:
2.  what does semaphore mean ? (Asked by: pr
3.  where is punjab ? (Asked by: praphul123)
4.  what is deadlock? (Asked by: tanusha1235)
Enter the Question ID to view answers or answ
```

```
Answers for Question:  what is deadlock?
No answers yet.
```

```
Do you want to answer this question? (y/n): y
```

```
Enter your answer: A deadlock is a situation
is waiting for a resource that another proces
```

```
Answer posted successfully with Answer ID: 1 (Sentiment: Neutral)
```

```
Enter the Question ID to view answers or answer it: 4
```

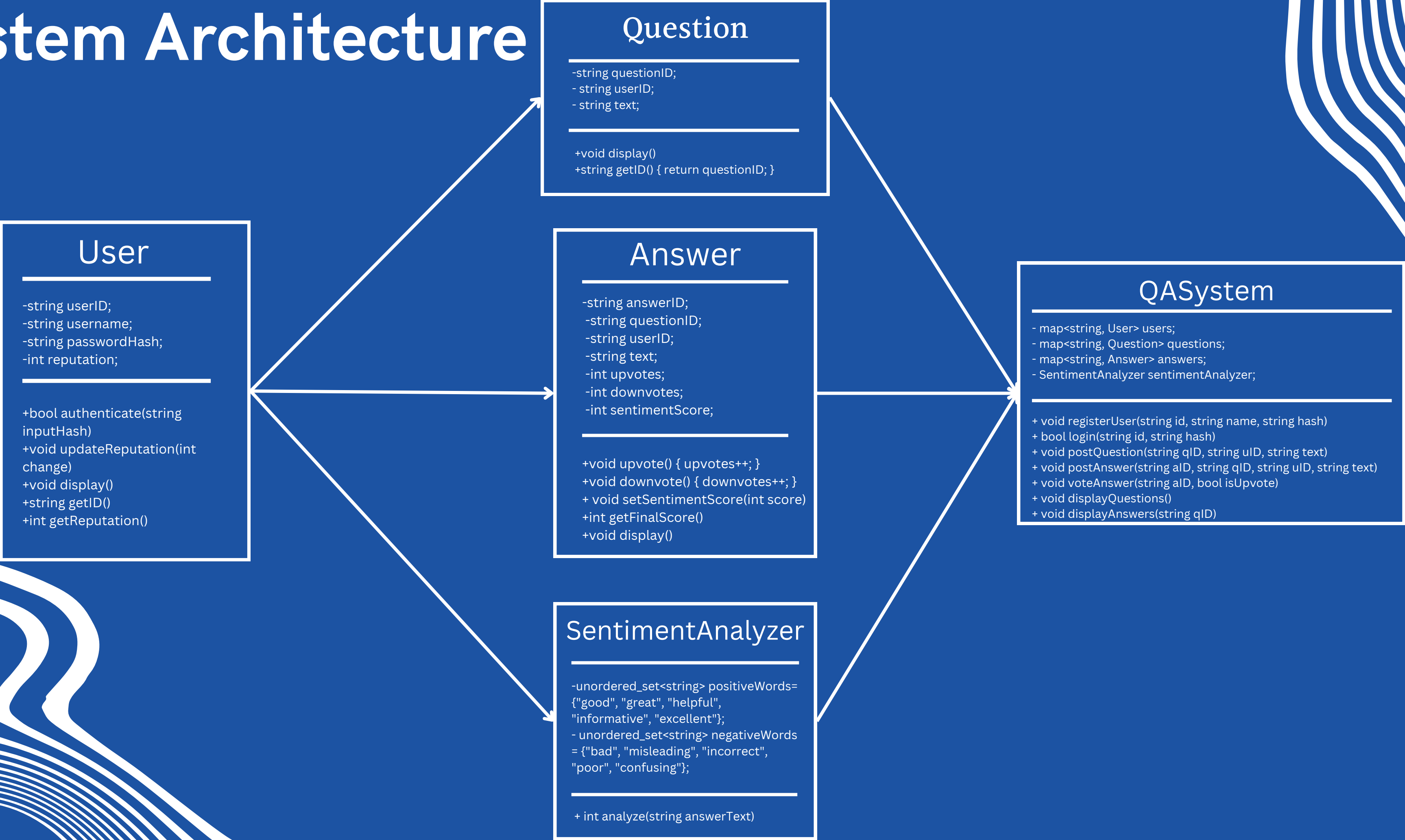
```
Answers for Question:  what is deadlock?
by tanusha1235 [0 votes, Neutral sentiment]: A deadlock
a state where each process is waiting for a resource that
Do you want to vote on this Answer? (y/n): y
Vote on Answer ID 1 (1 for Upvote, -1 for Downvote): 1
Vote recorded successfully!
```

```
Enter a word to search in questions
Questions containing 'is':
```

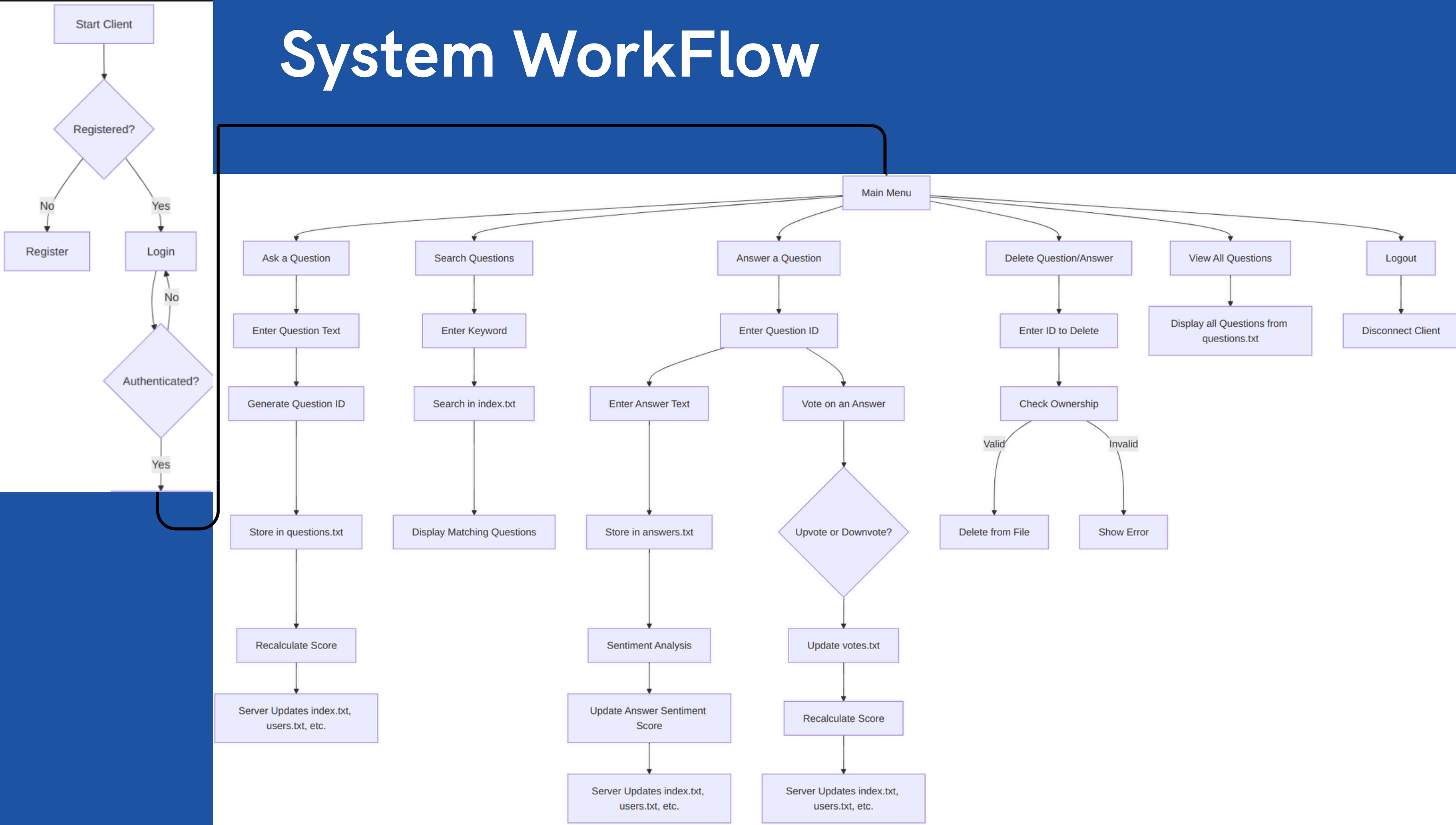
```
3.  where is punjab ? (Asked by: pr
4.  what is deadlock? (Asked by: ta
5.  Which planet is known as the "R
```

1.registration
2. posting
question and ans
3.answering que
and giving
feedback on ans
4.searching
posted que

System Architecture



System WorkFlow



System Modular Design

Client Component (client.cpp):

main(): Entry point that initializes UI and establishes server connection

connectToServer(): Creates and configures socket connection to the server

handleUserAuthentication(): Manages user login/registration processes

displayMenu(): Renders the main options menu for user interaction

sendQuery(): Formats and transmits user questions/searches to the server

displayResults(): Presents search results or question answers to the user

Server Component (server.cpp):

main(): Entry point that initializes server socket, database, and begins listening

acceptConnections(): Monitors for and accepts new client connection requests

handleClientRequest(): Processes incoming client requests and sends responses

initializeDatabase(): Loads stored data from files into memory on startup

shutdownServer(): Performs clean shutdown with data saving and connection closing

Answer Management (Answer.cpp):

Answer::Answer(): Constructor that initializes a new answer with content and metadata

Answer::getVoteCount(): Calculates and returns the current vote total

Answer::addVote(): Records a new user vote (up/down) for the answer

Answer::serialize(): Converts answer data to string format for file storage

Answer::deserialize(): Reconstructs answer object from stored string data



Question Management (Question.cpp)

Question::Question(): Constructor that creates a question with title and content

Question::addAnswer(): Associates a new answer with this question

Question::getAnswers(): Retrieves all answers linked to this question

Question::serialize(): Converts question data to string format for storage

Question::deserialize(): Reconstructs question object from stored string data

Question::search(): Determines if the question matches given search criteria

Database Management (

DatabaseManager::loadData(): Reads all data files and loads content into memory

DatabaseManager::saveData(): Persists updated data back to storage files

DatabaseManager::addQuestion(): Stores a new question in the database

DatabaseManager::addAnswer(): Stores a new answer in the database

DatabaseManager::getQuestion(): Retrieves specific question by its identifier

DatabaseManager::searchQuestions(): Finds all questions matching search parameters

DatabaseManager::updateVotes(): Updates stored vote counts for answers

Search Functionality (SearchEngine.cpp)

SearchEngine::parseQuery(): Breaks down search text into processable tokens

SearchEngine::search(): Executes search against the questions database

SearchEngine::rankResults(): Sorts search results by relevance score

SearchEngine::getTopResults(): Returns the most relevant search matches



Content Analysis (SentimentAnalyzer.cpp)

SentimentAnalyzer::analyzeSentiment(): Evaluates the emotional tone of text

SentimentAnalyzer::detectToxicContent(): Identifies potentially inappropriate content

SentimentAnalyzer::getSentimentScore(): Calculates numerical sentiment rating

Thread Management (ThreadManager.cpp)

ThreadManager::createThread(): Spawns new thread for handling client

connectionThreadManager::joinThreads(): Synchronizes by waiting for thread completion

ThreadManager::acquireLock(): Obtains mutex lock for safe resource access

ThreadManager::releaseLock(): Releases mutex lock after resource use

ThreadManager::initializeThreadPool(): Creates thread pool for connection handling

User Management (User.cpp)

User::User(): Constructor that initializes user with credentials and profile

User::authenticate(): Verifies provided login credentials against stored data

User::getReputation(): Calculates and returns user's current reputation score

User::addQuestion(): Associates a posted question with the user's account

User::addAnswer(): Associates a posted answer with the user's account

Helper Utilities (Utility.cpp)

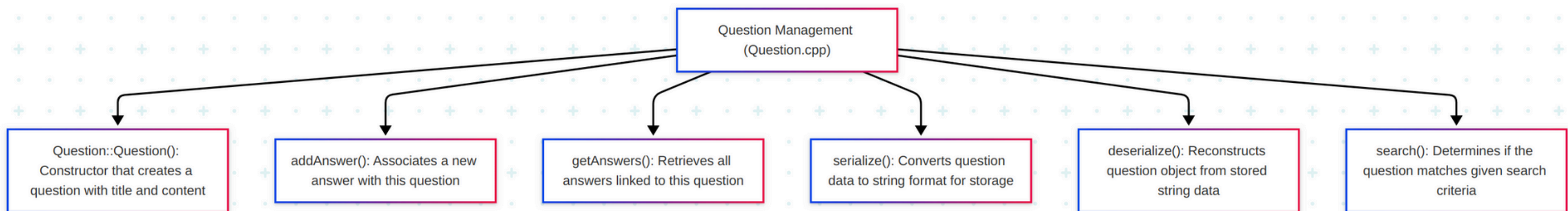
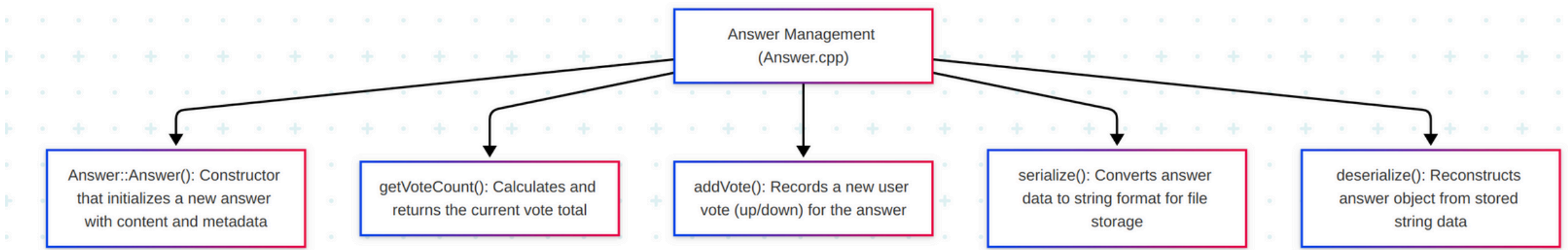
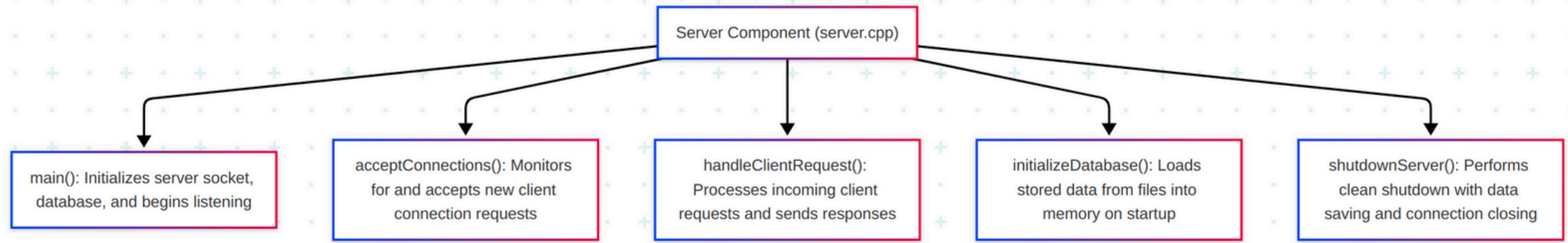
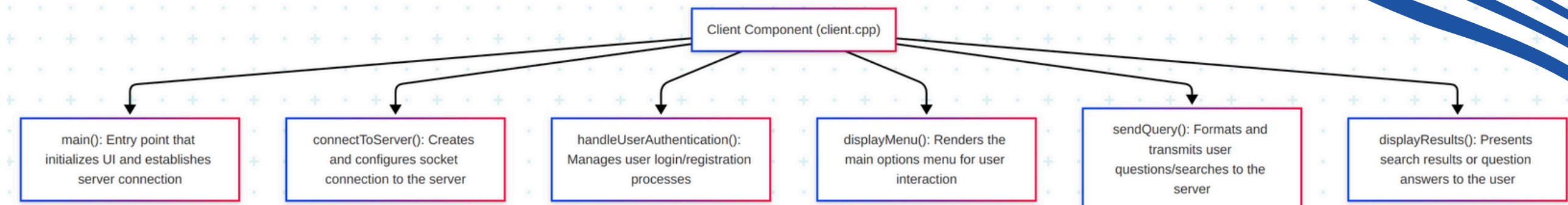
Utility::getCurrentTime(): Generates formatted timestamp for system operations

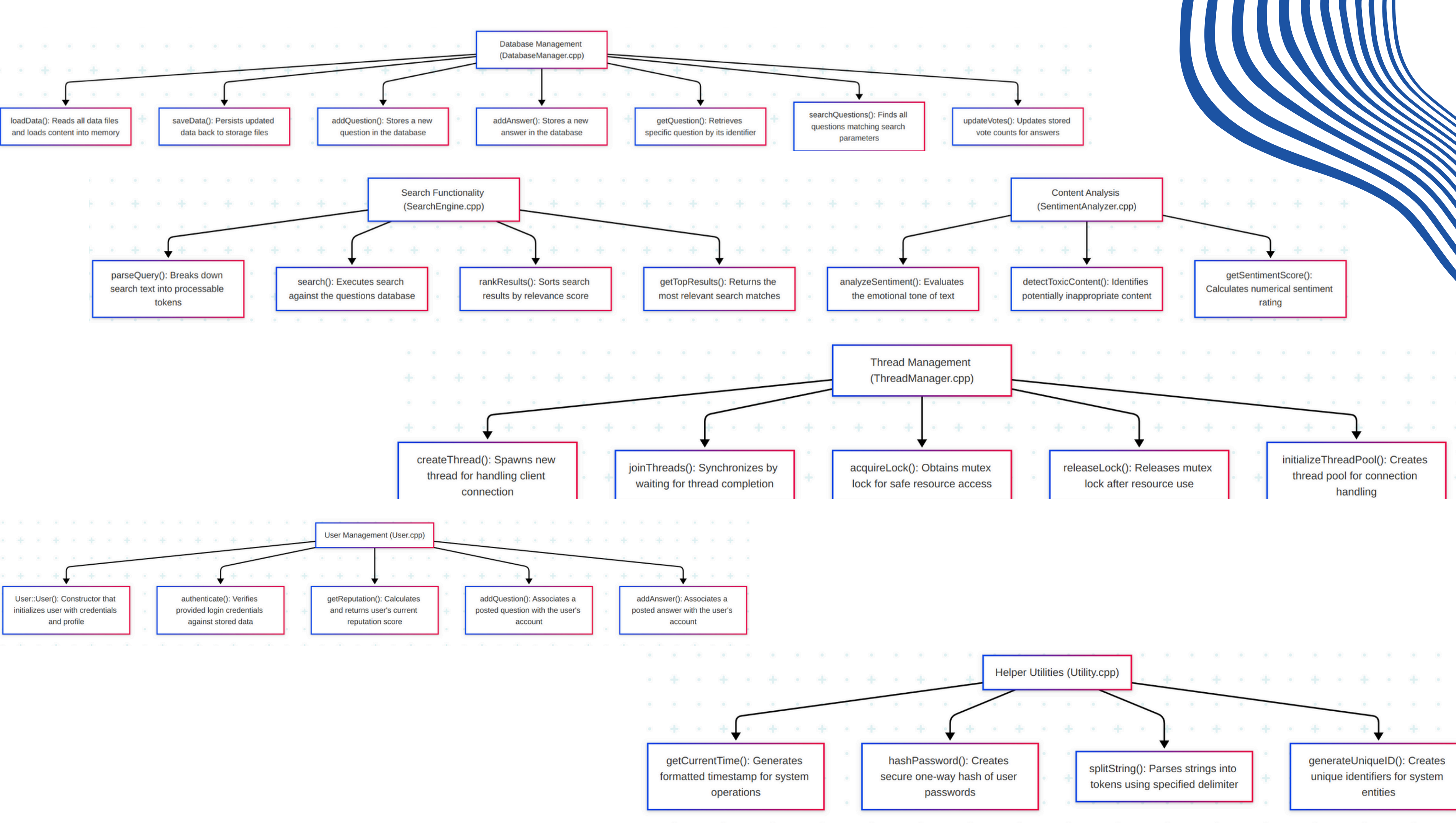
Utility::hashPassword(): Creates secure one-way hash of user passwords

Utility::splitString(): Parses strings into tokens using specified delimiter

Utility::generateUniqueID(): Creates unique identifiers for system entities

Modular Level Design





Features

01

Multithreading

Enables concurrent user interactions.

02

Sockets

Allows real-time multi-user access.

03

Shared Memory

Speeds up frequently accessed data.

04

Signals

Manages clean shutdown and process control.

05

Files & File Locking

Provides persistent storage with safe access.

06

Semaphores

Synchronizes shared resource access.



Features



User authentication



Ask Questions



Voting



Delete Question or Answer



Sentiment Analysis

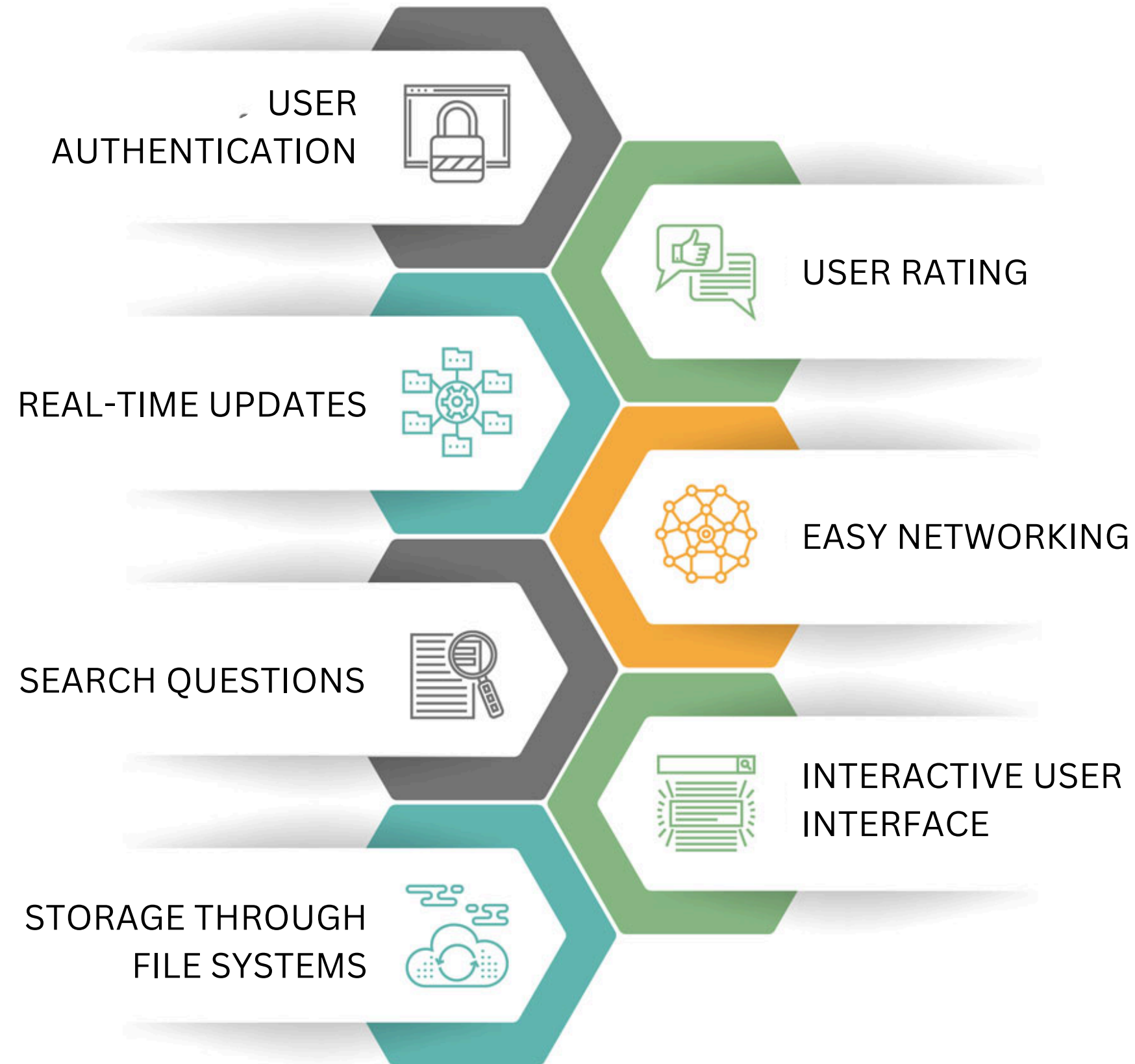


Answer existing Questions



Searching questions by Word index

Application Level Features



Timeline Goals





Future Additions

A web Interface for better usability

Upgrading the user interface from text-based interface to Web interface for better usability of the user

User password hashing


user password right now is not hashed, but would hash if further time allows

Comments section

further addition of the comments section for the user to comment on the answer if the answer is right or wrong.

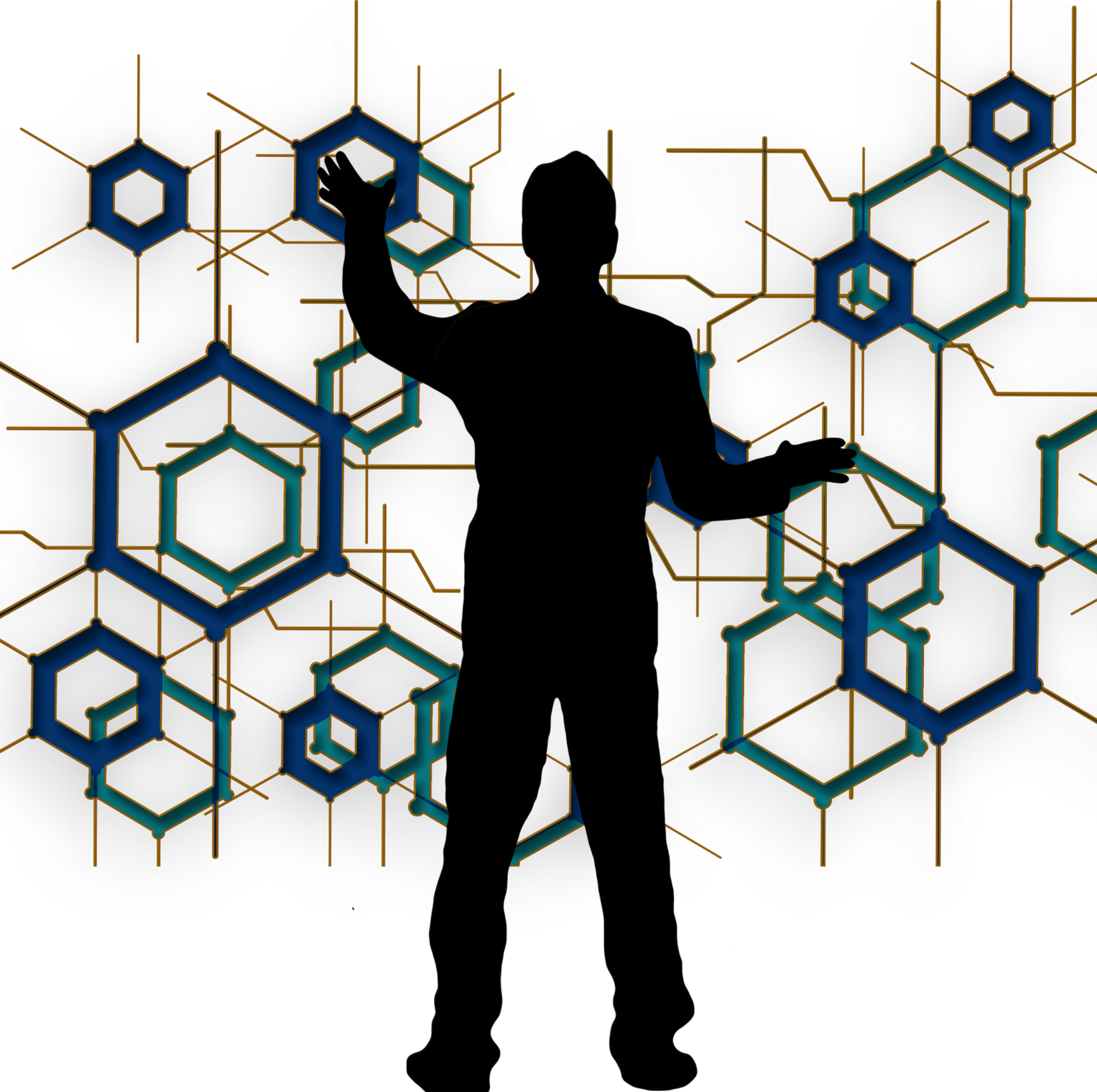
Over the Network communication

Make over the network communication possible, so that a user from another device can access the server and do real-time updates



PROJECT RESPONSIBILITIES

01	System Design	<ul style="list-style-type: none">• Anurag Prasad• Sumanth Naidu• Praphul Chandra
02	Project Planning	<ul style="list-style-type: none">• Tanusha Pramanik• Chaitanya
03	Coding	<ul style="list-style-type: none">• Anurag Prasad• Praphul Chandra
04	PPT Design	<ul style="list-style-type: none">• Govardhan Reddy• Chaitanya• Praphul Chandra
05	UI Interfacing	<ul style="list-style-type: none">• Sumit• Suryam Gautam• Govardhan Reddy
06	Resource Management	<ul style="list-style-type: none">• Keerthana• Authy saha• Praphul Chandra



THANK YOU

- By GROUP 2

