PLEXSHARE

Design Specifications

Lab Session Monitor

CS5617 Software Engineering
Instructor: Ramaswamy Krishnan Chittur

Ishwar Govind
Technical Architect

Contents

DESCRIPTION

The education sector is beginning to increasingly employ video conferencing services to assist instructors. There are many inconveniences one may encounter when attempting to conduct online lab sessions using such services (having to deal with an independent submission portal, monitoring students and many more). A dedicated monitoring tool would greatly ease the burden of instructors.

PlexShare is a lab session monitor that enables communication and file sharing between teacher and students. It also supports multi-user screensharing. A public whiteboard for students and teachers to draw diagrams and illustrations. A dashboard module for session management along with additional features like telemetry and session summary. The lab submissions are maintained in the cloud.

KEY FEATURES

- Client – Server architecture model.
- User Authentication using OAuth.
- LAN communication using sockets.
- Clients have options to screenshare to server.
- Clients can screenshare at 24 fps.
- Server can view 9 screenshare tile at most.
- Public Whiteboard model with permission for edit access.
- Whiteboard module provides user with options to draw curves and standard shapes.
- Dashboard with sessions management, telemetry and Session summarizer.
- Exam mode to restrict client to client communication.
- Users can send chat/files publicly or to other users.
- Lab submissions maintained in the cloud

OVERVIEW OF COMPONENTS

PlexShare has 7 modules.

- Networking
- Cloud
- Dashboard
- Content

- Screenshare
- Whiteboard
- UX

Every Module except for Networking module has their own UX implemented by using MVVM (Model View View-Model) Design Pattern. Each module has their own specs for detailed information.

NETWORKING

The networking module is used for communication between the client and the server. It serializes the objects and sends them as byte packets. The module has priority queues for enabling packet priority required for smooth transactions in the program. It gives higher priority to whiteboard, screenshare over content and cloud module. The network module provides functions for server to client, server to client packet transactions.

The network module has the information related to mapping for User ID to network address.

CLOUD

The cloud module is exclusively for the purpose of final lab submission. The cloud module saves each sessions metadata. The metadata is later used to retrieve a previous session's submission from cloud for evaluation purposes.

CONTENT

The content module gives the user access to an inbuilt messaging application with file sharing. With additional features like broadcast messaging, private messaging, emoji reacts, reply/edit/delete messages. There is a state manager for content module to store the messages in both client and server. Client to client content messaging is allowed based on the mode (lab / exam) selected at the beginning in dashboard by the server. The files are stored in the server.

WHITEBOARD

Whiteboard module gives the user the option to create a session which will give them access to a whiteboard with an option to edit. If the user is in read-only mode they can only view the changes happening to the whiteboard and can snapshot a copy of it. If given edit access by server, then the user can create, edit and delete shapes and draw random curves. The user can adjust the size, change color/gradient of the brush. Priorities are given to objects drawn by users. Lower priority users cannot edit/delete objects created by high priority Users. A user can request a snapshot of the whiteboard at any time and save as file.

SCREENSHARE

The ScreeShare module will take full screen snapshots of client's screen and send to the server. Every user once joined will be provided with the option to share their screen at the rate of 24 frames per second to the instructor. The module processes the snapshots and reduces it to a lower resolution with cursor tracking. Students (clients) cannot view the shared screen of other clients. The maximum screenshare tiles the instructor can view per page is 9.

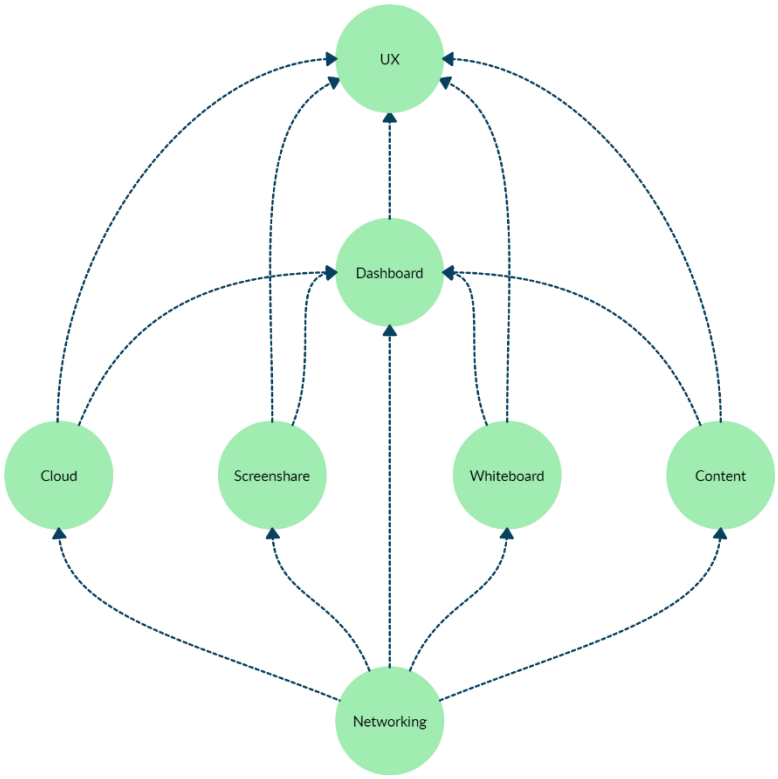
DASHBOARD

The Dashboard module is mainly used for Session Management. It stores the participant list in both user and server modules. It has a telemetry feature which stores user metrics data which can be later used for data analytics and improving the health of the software and user experience. The dashboard module also has a summarising feature which will provide the summary of an entire session using the chat from content module at the end of a session. The telemetry data is sent to the client when a client requests it.

UX

The main UX module is responsible for joining all the UX components of other modules (excluding networking). The allocation of resources especially UI grid spacing and tabs for modules are done in the main UX module. The home page and authentication are implemented by the same. The module has the implementation required for starting the server and loading previous sessions.

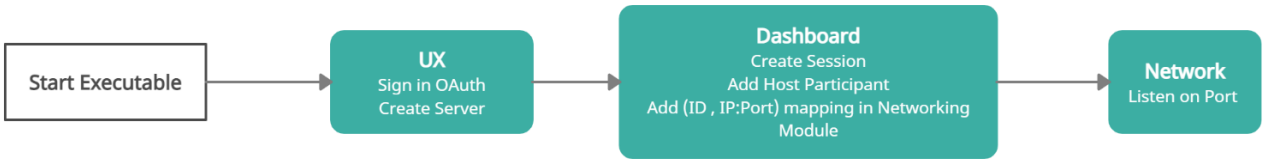
Dependency Diagram



Activity Diagrams

Server Activity

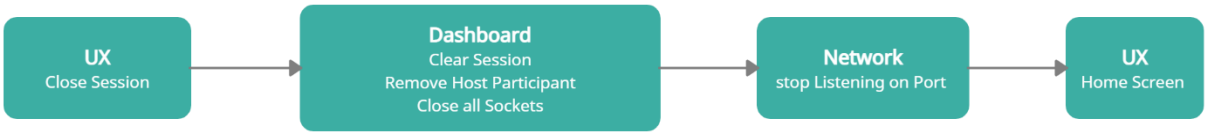
Lab Session Start



The Instructor starts the executable (plexshare.exe) which loads the home screen requiring OAuth authorization. The UX module need server IP and Port to host a session. The Instructor also has the choice to enables / disable Exam Mode which will disable client to client (student to student) transactions in the Content Module. The Dashboard is

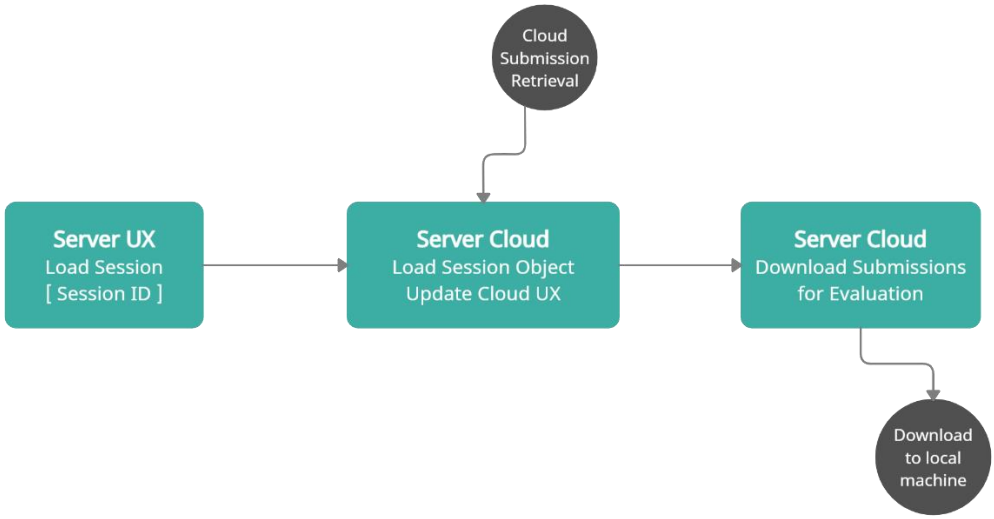
responsible for setting up the Session Management and adding User ID, IP: PORT mapping to the Network Table. The Server starts listening PORT for establishing connections with more users.

LAB SESSION END



The Instructor signals close session in the Dashboard. The Dashboard clears the session and generates the summary of the whole session as a file. The Networking module is given signal to stop listening on port. The UX module will take back the server to the Home Screen.

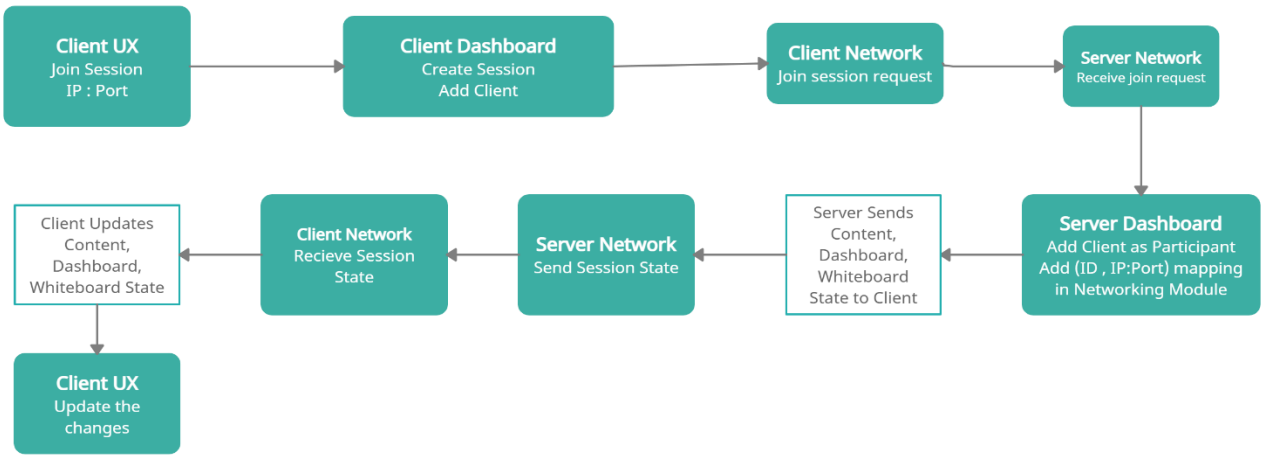
LOAD SESSION SUBMISSIONS



The Instructor can load previous sessions submission's for evaluation through the login page of the application. This will give an entry point through the UX to the Cloud Module through which the server can download submissions made by clients in the Session.

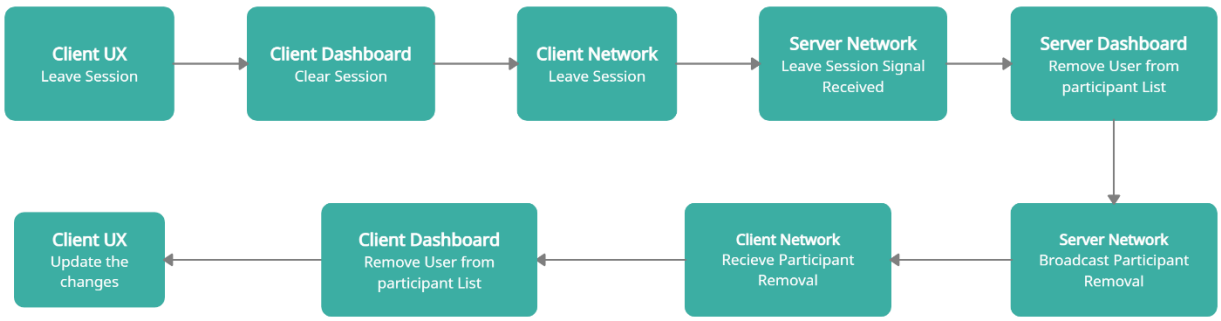
Client Activity

Lab Session Join



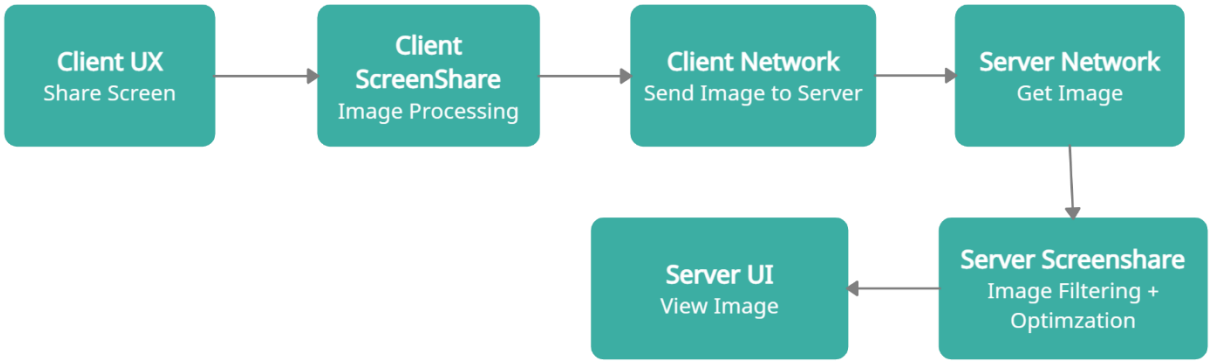
The Client can join the lab session by filling the form in login page (session ID, IP: Port of Host server) after OAuth Authorization. This will spawn UX module with tabs having Dashboard, Content, Submission and Whiteboard UI. To avoid the Client from not losing out on current state of the whiteboard and other content modules, the state is retrieved from the server. The UX changes are made using the retrieved states.

Lab Session Quit



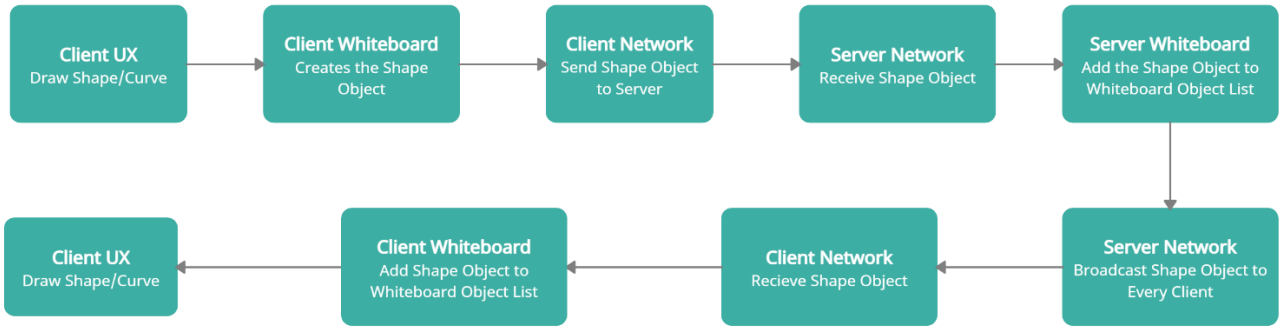
When the client leaves the session the client's dashboard clears its sessions and send data to server to remove user from participant list of every client connected to the session.

SCREENSHARE



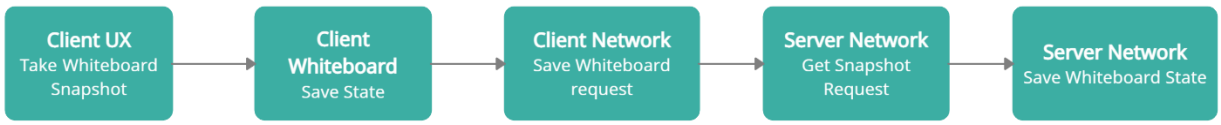
Screensharing is done by consecutively sending screenshots of the user's screen at constants intervals of time. The Screenshare module will process the image and filter images if the adjacent changes are null.

WHITEBOARD DRAW



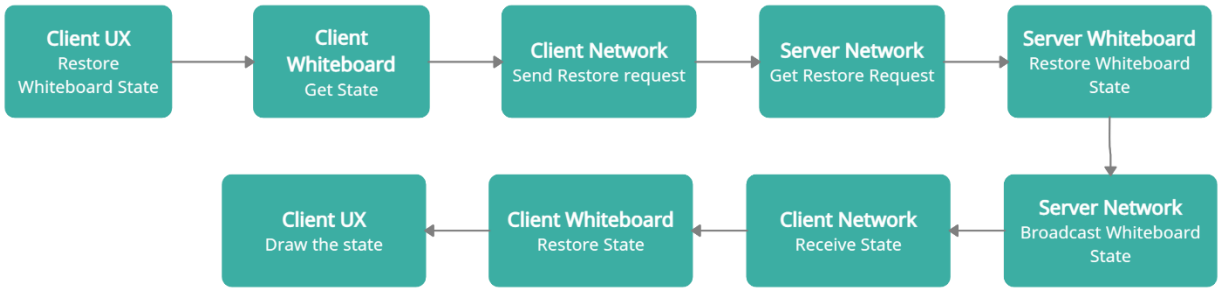
When a client draws a shape on the whiteboard, the shape object is created in the whiteboard module and is send to the server. The server uses the timestamp to attach the shape at the right position (above / below other shapes). Then broadcasts the shape to all other clients and the Whiteboard UX is updated for every client.

WHITEBOARD SNAPSHOT



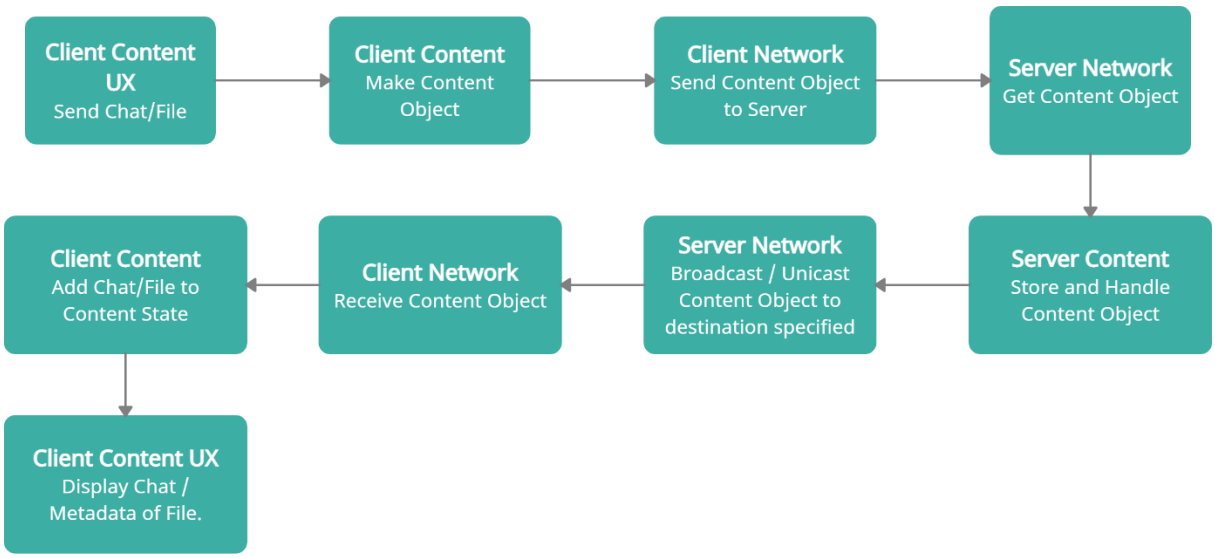
Users can snapshot any moment in the whiteboard during any time. The snapshot is saved as a list of whiteboard shape objects in the server which can later be retrieved through any client.

WHITEBOARD RESTORE



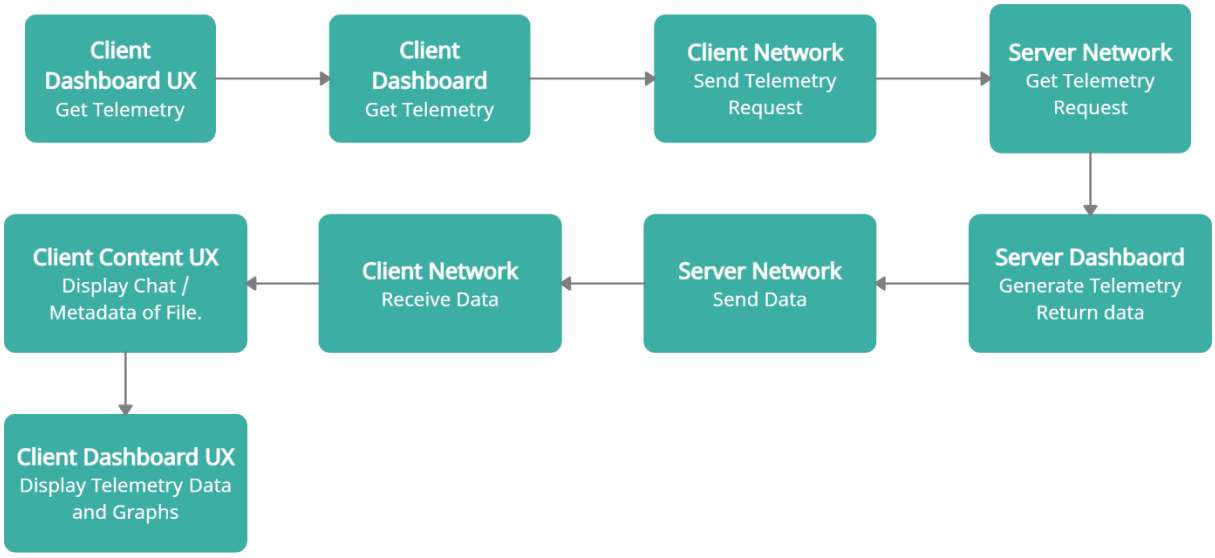
The snapshot previously saved in whiteboard can be retrieved by requesting for it to the server. The server updates the whiteboard state with the snapshot list of objects and broadcasts the state to all clients, updating the Whiteboard UX.

CHAT / FILE SHARING



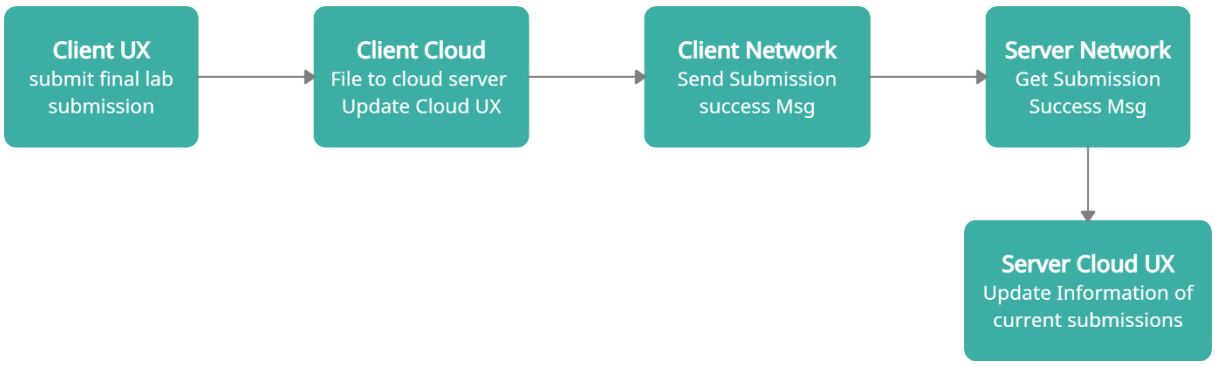
In Normal Lab mode clients can send chat/files to any other client whereas during Exam mode this feature is disabled. This setting is set before the beginning of the session in the UX. When a client sends message to another client or server the message which could be a file or text message or reaction to a message is send to the server and processed. If the message is to general chat it is broadcasted. If the message is a private message to another client, the server sends the content message object to the client.

TELEMETRY DATA RETRIEVAL



Telemetry Data is retrieved when a client requests for it through the UX of Dashboard. The retrieved Telemetry information is displayed in the UX component of Dashboard module.

CLOUD LAB SUBMISSION



Cloud Submissions are done by the client through the UX. The submissions are directly uploaded to the cloud server rather than sending to the server. On successful submission a message is send to server to notify the instructor regarding the user's submission.

Analysis

Peer-to-Peer V/S Cloud-Server

- Peer-to-peer implementation would require more complex to develop based on the module requirements.
- Client-Server has lower Fault Tolerance.
- Client-Server model has a more stable network form.

Due to the difficulty of programming the peer-to-peer model, we have decided to go for the Cloud-Server methodology.

Client-Cloud V/S Client-Server-Cloud

- Client-Cloud method would require less time to send the client's submission to the cloud (reduction in 1 hop delay).
- Client-Server-Cloud model causes flooding of the queue with submission packets in the network model delaying other low priority packets as well in the low priority PriorityQueue in the networking module.

Using Client-Cloud model for lab submissions as it better in terms of performance.

Conclusion

PlexShare can be used for monitoring lab sessions of multiple students concurrently. The exam mode can be used for monitoring lab exams to prevent malpractices. Lab mode can be used by tutors to conduct tutorial sessions and workshops. The Whiteboard helps students to work together more effectively.

References

1. **HTTPS://CITESEERX.IST.PSU.EDU/VIEWDOC/DOWNLOAD?DOI=10.1.1.671.5028&REP=REP1&TYPE=PDF** - Peer-to-peer vs Client-server
2. **HTTPS://WWW.NETWORKSTRAINING.COM/PEER-TO-PEER-VS-CLIENT-SERVER-NETWORK/** - Peer-to-peer vs Client-server