Mobile Information Systems (CSE 536)
Major Project Report
CollabNote

Adeel Zafar * 201205503

Juhi Kulshreshtha * 201205534

Sandeep Sharma * 201205605

Pallavi Gupta ‡ 201205543

Asmita Metrwar †‡ 201205544

Bhagyashree Jadhav ! 201205552

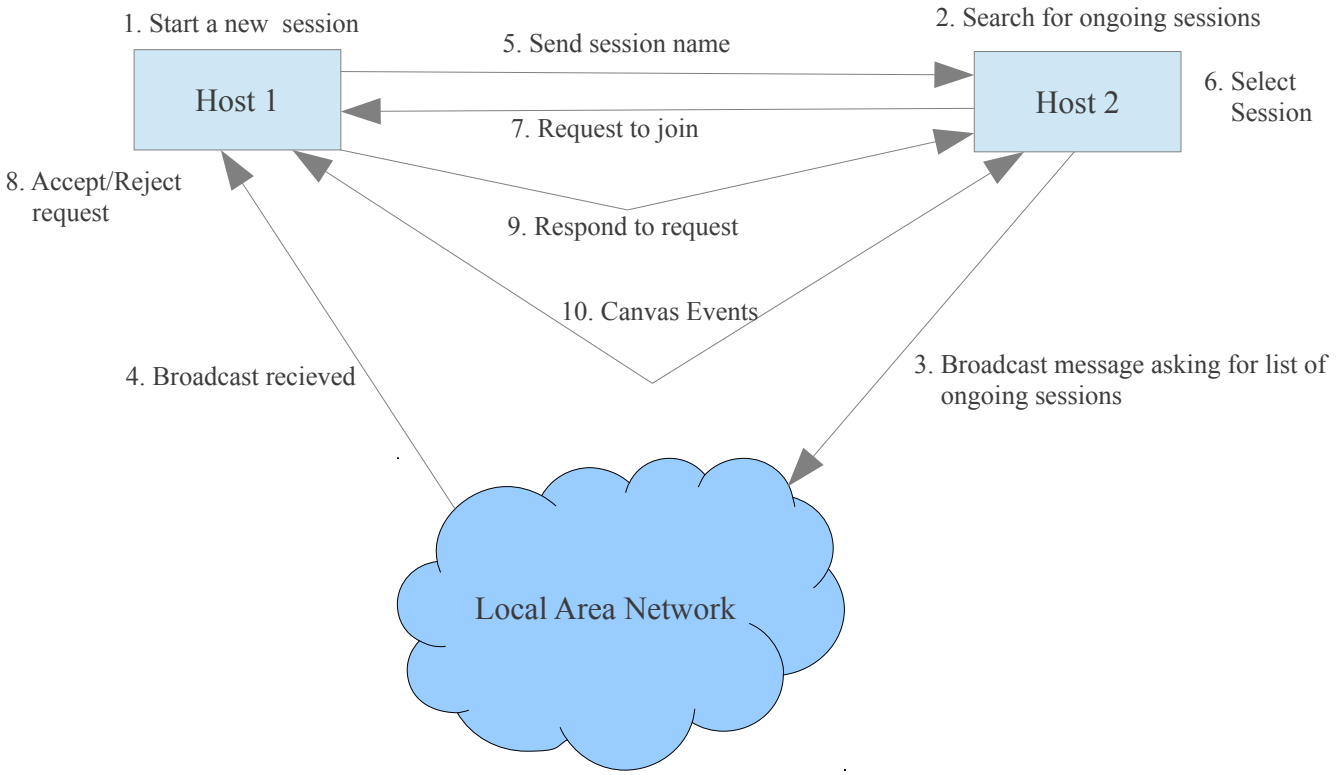
Sayeda Mohiuddin ! 201205598

† Team Lead * Canvas Module ‡ Networking Module ! UI Module

Description:

Our Project, CollabNote, is an android based collaboration tool that you can use to discuss proposals, jot down notes, collaborate with others or let the world know about your ideas. A simple to use, intuitive interface and a server-less architecture allow for the app to be used in any environment. All that is needed is a wi-fi connection, no internet is required. Any host (android phone) can start a session and others on the same network can join in. Upon start of a session, users can draw on a common canvas which is updated in real-time to all the other hosts. Implementing extra functionalities such file sharing and sticky notes will also be on the agenda.

A typical working scenario with two hosts:



The above is a simplified scenario where there are only two hosts on the network and only one session is hosted on that network. This over-simplification is for exposition purposes only. Our final application would support multiple hosts and any number of session on the network.

Use Cases:

1. **Start a session**: The user clicks on a button to start a new session, asking for a name of the session. The name of the session should be unique over the network. A canvas is opened and the host immediately goes into master mode.
2. **Join a session:** Upon clicking of the join button, a list of all ongoing sessions on the network is brought up. The user can select any of these sessions to join in.
3. **Accept or reject a join request:** The master node user can accept or reject a join request from a slave node user.
4. **Draw on Canvas:** Any user, master or slave, can draw on the canvas. Shapes such as circles, rectangles or lines and other objects such as text boxes can be drawn. The changes to the canvas will immediately be reflected onto other nodes in the same session.
5. **Add sticky note:** A long touch on the canvas causes a selection menu to come up where one can create a sticky note. The sticky note is positional and will appear on the same place where the long touch took place. This sticky note is not visible to others in the session.
6. **Add file:** A long touch on the canvas causes a selection menu to come up where one can select a file to upload. This file is transmitted to others in the session.

Network Communication Model:

We follow a master-slave model for all our communications for the canvas updates and file uploads. The node that starts a session is designated as a master and any other node that joins in afterwards is denoted as session. An update on any node in the session is first pushed onto the master which then pushes the changes to all the nodes in that session one by one. Thus the master would keep track of who all are in the network.

The different network message types are:

1. **Get session:** This message is broadcast by a host when it wants to get a list of all the sessions active on the network.
2. **Send session:** This message is sent to a host by a master node upon receipt of the "Get session" message. It includes the name of the session currently active on the master host.
3. **Request join:** A host sends this message to a master node when it wants to join a session active on that master host.
4. **Response join:** A master node responds to the "Request join" message with a "yes" or a "no" depending upon what the user of the master node selects.
5. **Canvas events:** Any canvas event that happens on any node in the session is sent to the master node that hosted the session. This master node then distributes these to the rest of the nodes.

Modules:

Networking Module: The Networking Module is responsible for handling end to end communication between applications running in different devices. It has 2 components:-

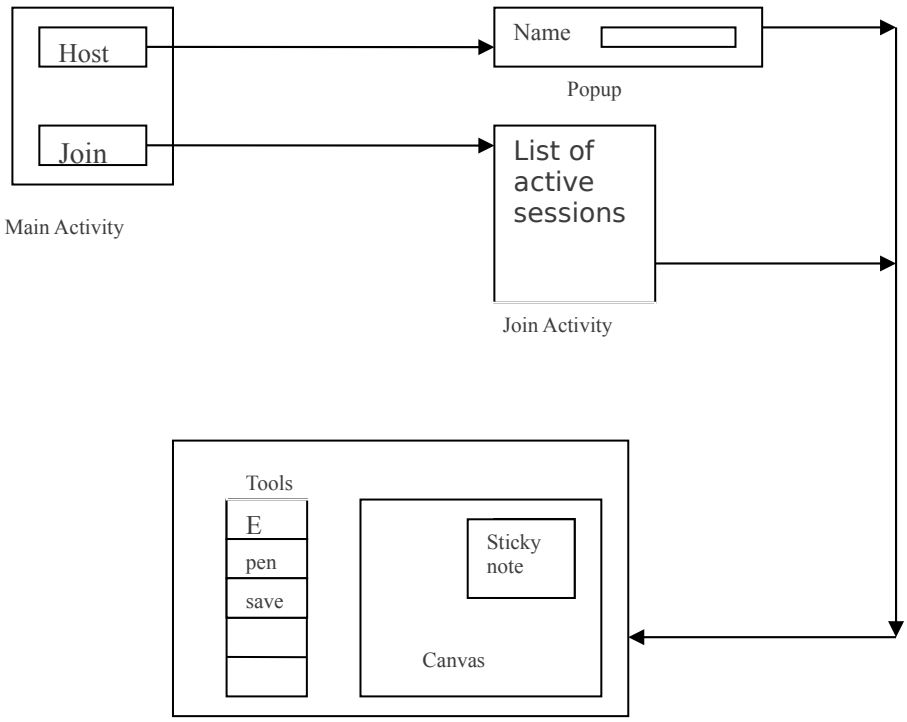
1. A separate server service is started which listens at port 5525 to receive message from local area network and pass them to the application.
2. When an activity wants to send any message it spawn a new client thread to send the message to a particular IP.

Canvas Module : The Canvas Module is responsible for rendering the image on the canvas of each node and invoking a separate thread to send updated pixels. It also provides option to select various available drawing tools like brush, pencil, eraser and colors.

UI Module : UI Module handles all UI components of activities in application. The various activities within its purview are:

1. Main Activity :- Has options for hosting a new session as master or joining an active session as a slave.
2. Join Activity :- Shows all the active sessions as a list and gives an option for user to join in.
3. Canvas Activity :- It allows user to draw collaboratively. Interacting with the Canvas layer it renders all the images drawn by user on canvas and sends it to other device using the transportation layer. It also provide the user the option of saving current canvas and creating a new blank canvas.

UI Mockup:


Host
Join
Main Activity
Name
Popup
List of active sessions
Join Activity
Tools
E
pen
save
Sticky note
Canvas