TalkSee

Team 6

Mark Higa

Chris Bauschka

Ashley Braun

Aaron Petry

Problem Statement

Our team project is a Windows Phone application utilizing the Microsoft Research Hawaii SDK, designed to aid people that may have hearing conditions or may simply be in an environment where it is difficult to hear the phone speaker. As users communicate through voice or video-call, the app will provide speech-to-text captions for the receiver, providing a visual means/supplement for communicating. Users will also be able to manually type and send texts communications during the voice/video-call.

Background Information

The application was inspired after observing a friend sign to his deaf mother through FaceTime on an iPad to iPhone connection. Seeing them communicate directly without the help of an added human interpreter was amazing, but there were still many limitations. The first and most immediate was that while his mother could also see me, my inability to sign still kept me from communicating with her. But if an app could quickly translate speech into text as captions, she could read what I said even though she could not hear me. Additionally, signing usually requires users to be in a stationary environment typically requiring both hands to sign. An effective speech-to-text caption application would enable them to travel with phone in hand while seeing the other user sign and being able to speak back.

We quickly realized that this concept could be improved and expanded to help an even greater number of people, particularly the elderly who may have difficulty hearing. This would allow them to communicate without special phones or hearing aids (if they don't feel like wearing one just to take a call). Even with proper hearing aids, it could provide a visual confirmation of the verbal communication. Additionally, users with adequate hearing who are in noisy environments might use this as a supplemental communication aid. And by having the captions available in a transcript form, users could review the conversation if they missed what the other user said.

There is a service called CapTel that both ATT and Sprint offer that adds captions to voice calls. However, this solution involves a call center with human translators who actively listens to the call and transcribe it to text. A new application for iOS was recently introduced called ClearCaptions, but we do not know how well it performs. We are currently unaware of any applications that provide automated speech-to-text captions for voice calls on the Windows Phone platform or speech-to-text captions for video-calls on any mobile platform. We believe that with the cross-platform nature of the new Windows environment, and the plethora of smartphone, tablet, and computer devices soon to be released by Microsoft, our application will provide an excellent solution for a very wide market.

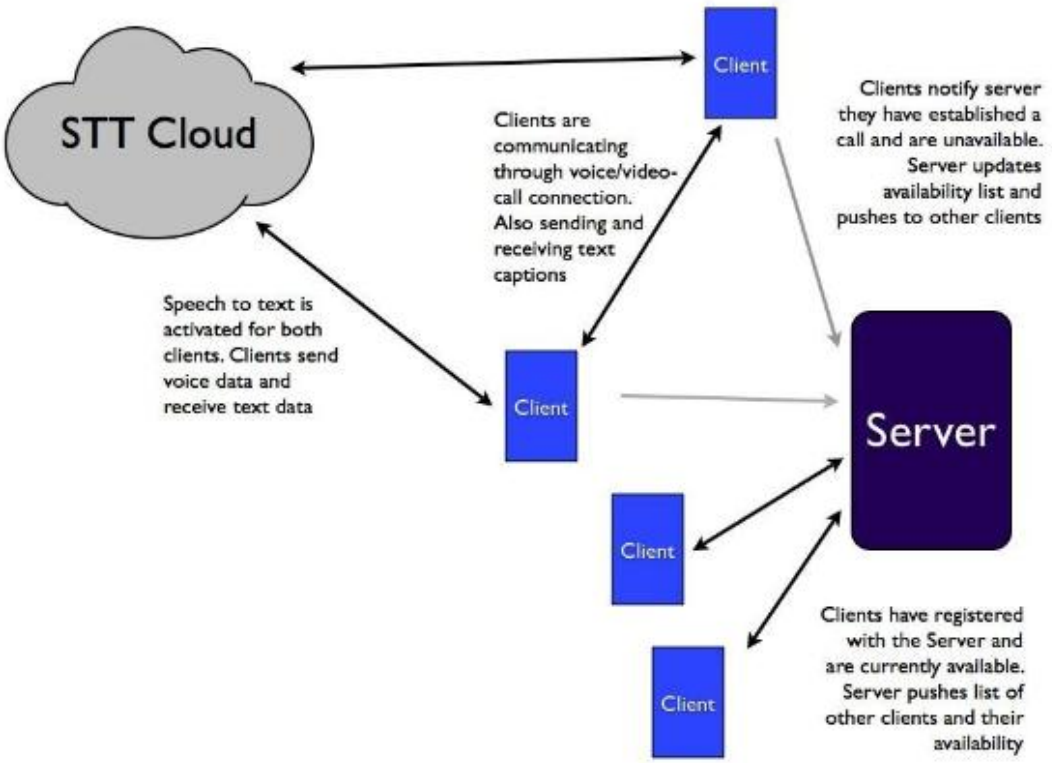
We are exploring multiple voice and video-call platforms with public SDKs and APIs to find an ideal solution that we will be capable of implementing. We are also prepared to develop custom solutions with possibly limited functionality if necessary. We will use the Microsoft Hawaii SDK to provide the speech-to-text service.

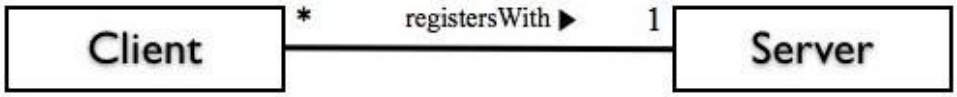
Environment and System Models:

The TalkSee application will utilize a client/server architecture. The server will communicate with the client application on Windows Phone devices through an internet connection. The client will also communicate with Microsoft's speech-to-text (STT) Cloud service when text captions are activated.

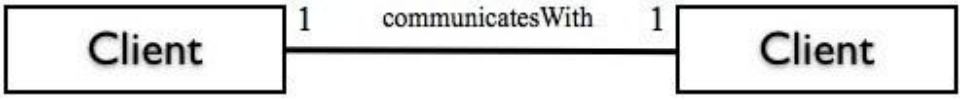
Each client will establish a connection with the server and send information about itself which the server will store. Each client will keep the server updated with its availability and connection status which the server will store and share with all other connected clients.

An available client can initiate a call to another available client and begin voice/video-call communication. If speech-to-text captions are activated, the currently speaking client will continually record and send voice clips to the STT Cloud. After translating the speech to text, the STT Cloud will send the text back to the speaking client, which will immediately send it to the currently listening client. Both clients will store the sent and received text as a transcript while they are connected.

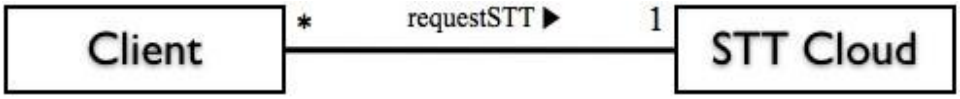




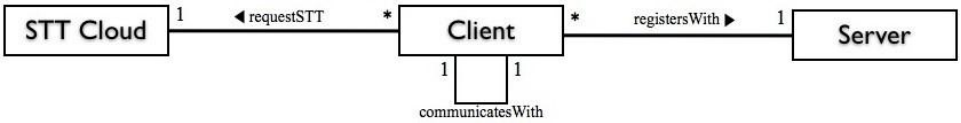
Multiple clients can register with Server.



Each client can only connect and communicate with one other client.



Multiple clients may request speech-to-text translations from STT Cloud



Multiple clients can register with Server. Each client can only connect and communicate with one other client. Multiple clients may request speech-to-text translations from STT Cloud.

Functional Requirements:

I) Interaction between the client and the directory server

1. The server will initialize with an empty list of clients.
    a. The server will periodically send this list to any connected phones.
2. The server will listen forever for clients that wish to connect.
3. A client will establish a connection with the server.
    a. The client will send its relevant connection information to the server (name, IP address).
    b. The server will identify this client with a unique id.
    c. The server will record this information into its directory.
    d. The server will send to the client a list of all available clients and connection information.
4. A client will tell the server that it is available.
    a. The client will be added to the available list.
    b. The server will push the updated list to all connected clients.
5. A client will periodically tell the server that it is still connected.
    a. If the client times out, it will be removed from the server's connected list and available list.
    b. The availability list will be pushed out to all connected devices.
6. A client will tell the server when it is unavailable (in a call).
    a. The server updates the list of available devices.
    b. Both clients are marked as in a call on the list.
    c. The server will push out the updated list to all connected clients.
7. The client will tell the server that the call has ended, and that it is now available.
8. A client will tell the server that it is disconnected when the client is closed.
    a. The server will update the availability list and the connected list.
    b. The server will push out the updates to all connected phones.

II) Interaction between the client and the STT Cloud

1. The speaking client will send recorded voice data to the STT Cloud.
2. The STT Cloud will return the text version of the voice data to the speaking client.
3. The speaking client will send the text thought the message box to the listening client.

III) Interaction between the client and another client

1. The calling client will send a call request to a receiving client.
    a. The server will remove both clients from the available list.
    b. The calling client will have the option to cancel the call before the call goes through.
2. The receiving client will either accept or cancel the call.
    a. Accept: both clients begin communication.
    b. Cancel: both clients are marked as available again and exit the call.

Quality, Platform, and Process Requirements:

Quality:

Regarding quality, there are a few specific benchmarks that we would like to strive for in order to provide quality to any user of our application. We would like for our video to be above 20 frames per second and only between 2-4 seconds of delay. We would like our speech to text to take no longer than 3-5 seconds to load. Also, we would like our sound to be in sync with the video up to 1 second of delay. We would prefer the whole application, including the server, to load quality and efficiently in under a second.

Platform:

We will be developing the client application using the Project Hawaii SDK 2.0 for Windows Phones. This application will be tested and deployed on a Windows Phone 7.5 Operating System (Mango). The Project Hawaii SDK allows Windows Phone applications to take advantage of research cloud services. Using this SDK, we have access to a variety of tools. We will specifically be using these cloud services for the speech-to-text feature of our application.

Our client server will be developed using the Python Programming Language. Although our server won't require any direct user interaction, Python is able to be used on Windows, Linux/UNIX, and Mac OS X machines.

Reliability:

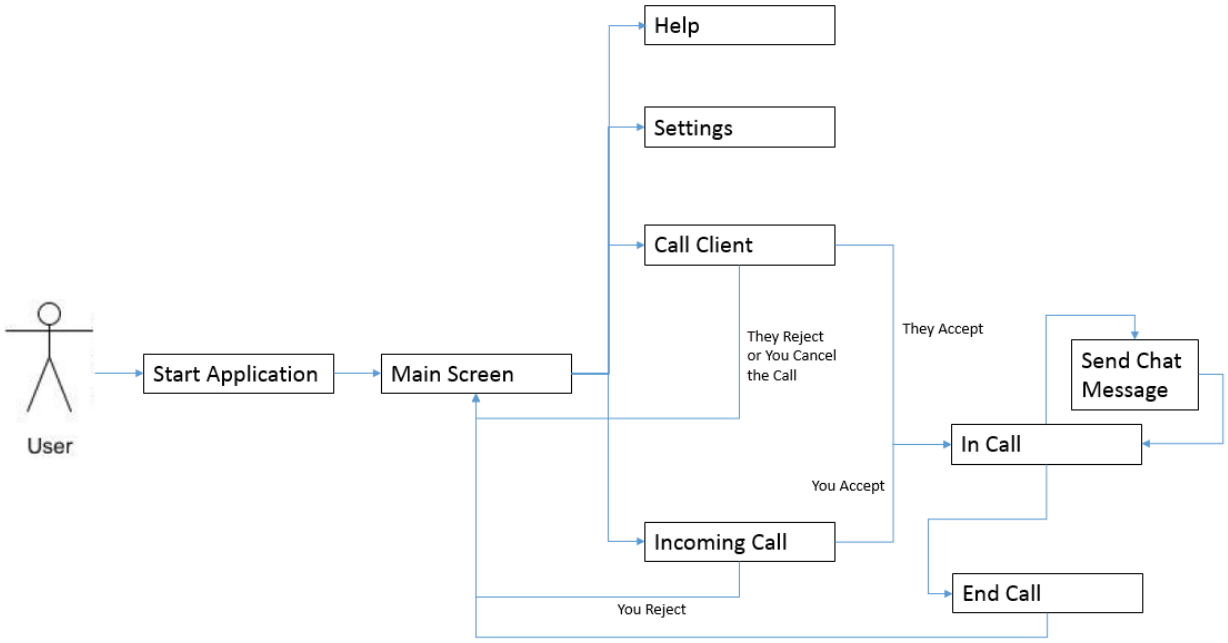
In order to create a positive experience for our user, reliability is an important issue. First and foremost, our server shouldn't crash and should be able to handle multiple clients. Having a reliable server increases the availability and convenience for the user. Regarding the client, we are striving for it to handle multiple calls and not have any issues with crashing. We will be using UDP (User Datagram Protocol) for faster communication and lower delay. Finally, our application should be secure in order to protect our users.

Usability:

Because of the nature of this application, our goal is always to make it simple and easy for any person using it. Settings are a very important part of this. We will implement settings that allow a user to turn certain functions on/off, change their name, adjust the font, and change the server and port.

Making these changes and using the application itself should and will be easy and intuitive. The User Interface will also be very simple allowing for easy use and navigation.

Use Cases:



Name: Start Application

Pre-conditions: The user has the application installed on the Window's Phone

Steps:

User actions:

1. Selects the application from the phone's list of apps.

System Responses:

2. The application begins running

Post-conditions: The user will be directed to the main screen of the application

Name: Main Screen

Pre-conditions: The application has started or is running successfully

Steps:

User actions:

1. User navigates to main screen by opening the app, or by navigating back from other processes within the app.

System Responses:

2. The application displays the main screen.

Post-conditions: The application will display the main screen

Name: Help

Pre-conditions: The user is located at the main screen.
Steps:

User actions:

1. User selects the help button

System Responses:

2. The application displays the help screen.

Post-conditions: The application will display the help screen

Name: Settings

Pre-conditions: The user is located at the main screen.
Steps:

User actions:

1. User selects the settings button

System Responses:

2. The application displays the settings screen.

Post-conditions: The application will display the settings screen

Name: Call Client

Pre-conditions: The user is not already in a call
Steps:

User actions:

1. User selects an available name to call
2. User then presses "Call"

System Responses:

3. The client sends a request to the server to connect with the selected client
4. The server connects to the requested client.
5. The client is asked to either accept or reject the call

Post-conditions: The application will display a "Calling ____ …" message, with the option to cancel on the user's phone.

Name: Incoming Call

Pre-conditions: The user is not already in a call, and a calling client has requested to connect to the user.

Steps:

User actions:

2. The user will either accept or reject the call.

System Responses:

1. The user is asked to either accept or reject the call

Post-conditions: If the user accepts the call, the connection will be established, both users will be removed from the available list, and the call will proceed. If they reject the call, they will be returned to the main screen.

Name: In Call

Pre-conditions: The user and the other client have accepted the call

Steps:

User actions:

1. The user will carry on a conversation normally.
5. The user sends chat message(s), or ends the call.

System Responses:

2. The application will record the user's voice and send the data to the Hawaii STT service.
3. The STT service will send back the speech converted into text, and the client will send the sound and the text to the listening client.
4. The application will display the text in the chat history.

Post-conditions: If the user sends a chat message, it will proceed to that process. If they end the call, the connection will be broken and they will return to the main screen.

Name: Send Chat Message

Pre-conditions: The user is in a call

Steps:

User actions:

1. The user types in a message and hits enter.

System Responses:

2. The client will connect to the listening client and send the text.
3. The server will connect to the other client and send the text.
4. The application will display the text in the chat history on both phones

Post-conditions: Text is displayed in the chat history, and the call continues as normal.

Name: End Call

Pre-conditions: The user is in a call

Steps:

User actions:

1. One of the users ends the call

System Responses:

2. The connection will be broken
3. Both users will be added back to the available list
4. The users will be returned to the main screen.

Post-conditions: Users will both be back on the main screen of the application