Team: Church of CVN
Members: Mackenzie Glynn (mwg2126), Nathan Reitinger (nr2645), Riva Tropp (rtt2114), Brett Landau (bl2719)

Ada

Part 0:

Language: Java
Platform: Linux

Summary:

When coordinating work or social interaction across geographic locations, it can be difficult to communicate effectively. Real-time chat closes this gap, enabling users to have a near in-person experience without leaving their location. Remote communication can also enable people with mobility issues to engage further in work and social settings.

For our project, we are going to be building a chat application for linux desktop. Our project will support at least two people chatting with each other and offer text, voice, and potentially video communication options. We plan on having a central server that keeps communication logs in a relational database---a stretch goal is to have this server route connections whenever a user chooses to host a chat room. Another goal is to utilize Google's speech-to-text API and have the server create text transcriptions of voice and video chat sessions to give the users written records of their chats.

Our code will be written in Java, which likely means it will be able to support users on other operating systems as well. Some preliminary research has pointed us toward several Webcam capturing APIs (namely, sarxos/webcam-capture) which claim to support the major operating systems. Java has abstracted wrappers around network sockets which we will use to facilitate the chat room communication and there are Java libraries for both MySQL and PostgreSQL which we will use on the backend.

Technologies[1]:

Unit Test Framework: JUnit
Static Analysis Tool: PMD
Database: PostgreSQL or MySQL
Version Control: Github

[1] Subject to change.

Team: Church of CVN
Members: Mackenzie Glynn (mwg2126), Nathan Reitinger (nr2645), Riva Tropp (rtt2114), Brett Landau (bl2719)

User Stories:

1. As someone who changes location often, I would like to be able to send messages to people even when we are not in the same room, so that I can always have the option of communication.

   ~my conditions of satisfaction are~
   - I can send and receive one or more text messages
   - My conversation partner can respond to me
   - I can send messages to anyone I like
   - I can send and receive as many messages as I like
   - I can't send messages to numbers that don't exist, and if I do I will be told it is unsupported

2. As someone who needs and enjoys "private" time, I would like to have an account, so I can log off when I do not feel like talking to anyone (and log on when I do).

   ~my conditions of satisfaction are~
   - I can choose a username to represent myself
   - No one else can use my username in a chat
   - I can determine who my conversation partner is by username
   - When I am logged off, I am not bothered by messages
   - If I am logged off and someone sends me a message, I can still retrieve it when I log on

3. As a person who often uses long-distance communication to contact people I care about, I want to be able to see a picture of my friend when I chat with them online, so that I can enjoy seeing my friends' faces.

   ~my conditions of satisfaction are~
   - I can select a digital picture that is associated with me
   - The picture can be of anything I like (permitting size restraints)
   - If I don't choose a picture, a picture is chosen for me
   - This picture is the one my friends see when they talk to me
   - People that I don't know can't see this picture
   - I can change this picture whenever I like
   - When I change a picture, my friends see the new picture

4. As an avid chatter (i.e., someone whose messaging can span days or weeks), I would like to see my older messages, so I can pick up a thread of conversation where I left it.

   ~my conditions of satisfaction are~
   - I can see old messages that I have sent, or that were sent to me
   - I cannot see other peoples' messages and other people cannot see my messages
   - I can tell which person sent which message
   - The messages appear in chronological order
   - I can see the time that the message was sent respective to my current time zone

5. As someone who is vision-impaired, I want to be able to have text messages read aloud to me, so that I don't miss anything.

   ~my conditions of satisfaction are~
   - I can navigate to a given message without having to rely on my eyes (perhaps by typing into a control bar)
   - A long message is read in its entirety
   - Punctuation marks do not impact the voice speaking the text
   - Older messages may be read out loud, not just the current message
   - I can adjust the volume of the speaker

6. As a person who generates lots of messages (and who is also unorganized), I would like to search for a given message in my history, so that I can remember exactly what I said.

   ~my conditions of satisfaction are~
   - I can type in a search term to navigate to a message that was sent by myself or one of my friends
   - I can determine who sent the message
   - I can determine the time the message was sent
   - I can have the message read aloud to me
   - No one else can search through my history

7. As a bilingual user, I would like to have non-English language support, so that I can talk to my friends in our preferred language.

   ~my conditions of satisfaction are~
   - I can type messages into the chat in languages other than English
   - Other languages are dealt with in some way by the voice program (number 5)
   - My chat partners can see the messages in our non-English language (i.e., the messages are not changed in any way)
   - I can search for messages in my preferred language

8. As a person who likes “reading people,” I want to see and hear my friends when I talk to them online so that I can get the subtext of the conversation.

   ~my conditions of satisfaction are~
   - My friends’ faces move when I speak to them online (vice versa)
   - I can see my friends faces clear enough to make out expressions
   - When I am done talking, the video turns off
   - I can talk to at least one person
   - I can’t talk to no one
   - If I talk very quickly, my friends can still hear me
   - Background noise does not overpower my talking

User Story Wishlist:

- As someone who works remote often, I would like my video chat sessions to feel responsive and for framerate to be high (i.e., frame delta compression)
- As someone who says the same thing often, and doesn’t want to have to type it out over and over, I’d like my chat to remember my top-5 “sayings” so I can just use one of those in response rather than type the whole phrase out

Acceptance Testing:

1. Chat Service (User Story: 1)
   a. verify that a message sent from Alice to Bob appears on Bob's machine in plain text format and all visible characters appear unchanged.
      i. Input handling:
         1. one-word message succeeds
         2. message with interior whitespace succeeds
         3. message with liminal whitespace (e.g., <space>word<space>) succeeds
         4. unicode (particular character supported or not, with error messages appearing appropriately) succeeds
         5. message with unprintable characters rejected
         6. message with punctuation succeeds
         7. message as long as our database will allow succeeds
         8. message with sql injection statements rejected or modified to be safe
   b. Alice receives positive feedback that the message was sent (may be implicit given the lack of error response)
   c. if a message is too long to store, Alice receives a responsive error
   d. if a message fails due to a server error, Alice receives a responsive error

e. if a message fails due to a non-existent recipient, Alice receives a responsive error
f. if Alice sends too many messages at the same time (i.e., traffic flooding), then the messages fail, Alice gets a responsive error
g. if Alice sends a large, but supportable number of messages, the messages succeed
h. after Alice sends a message to Bob, Bob can send a message to Alice (with the same integrity requirements and inputs as above)
    i. messages appear in the order that they were sent
    ii. two messages may be sent at the same time

2. User Account (User Stories: 2, 3)
    a. account (User Story: 2)
        i. log in and log out
            1. Alice registers a username and password
            2. Alice can log in with that username and password
            3. Alice can see her own account information with her username and password
            4. Alice can log out and can then no longer see her own data
        ii. username validation tests
            1. username with supported characters succeeds
            2. username with unsupported characters does not succeed
            3. username already in the database does not succeed
            4. username of the appropriate length succeeds, and too long or too short does not succeed
        iii. password validation tests
            1. correct username and password pair succeeds
            2. username with incorrect password does not succeed
            3. unsupportable number of attempts at password guessing fails
            4. password storage is not plain text (secure with appropriate encryption and salt)
            5. passwords are the appropriate length and rejected if not
        iv. verify that a message fails without usernames of the sender and receiver
        v. verify that a message cannot be sent without credentials
    b. photo (User Story: 3)
        i. icon
            1. size of photo is appropriate (not over 10MB)
            2. type of photo is appropriate (.JPEG or .PNG)
            3. administrators with approval have the ability to change photos (user is notified)
            4. random photo assigned for users who do not upload a photo
            5. users may upload as many photos as they like

6. past photos do not persist, users do not have a history of photos uploaded
7. only a single photo is associated with a user

ii. UI

1. photo is viewable by the user and the people who the user communicates with
2. photo is not viewable to people the user does not communicate with
3. photo is uploaded by selecting the photo on the users' file system

3. History and Search (User Stories: 4, 6)
    a. history log (User Story: 4)
        i. verify that Alice's history (in our database) contains:
            1. a message Alice has sent to Bob
            2. a message Bob has sent to Alice
            3. messages stored in the order they were sent
            4. include support for messages sent in different time zones
                a. User views timestamp per their respective time zone
        ii. message sent from Alice to Bob in central time is stored in milliseconds, in EST
        iii. message with a negative timestamp is rejected
        iv. timestamps do not change over time (immutable)
        v. Alice can see a message that was received by her
        vi. Alice cannot see a message that was not sent or received by her
    b. keyword searching (User Story: 6)
        i. user may search through their own history for a keyword (use fake database for this)
            1. multiple matches are returned in full
            2. no messages without the search term are retrieved
            3. a "no match" is reported to the user as a failed match
            4. matching is on exact word typed (case sensitive)
            5. only keyword search is supported (not regex)
        ii. inputs are handled the same as input in "chat service" acceptance test (see acceptance test number 1)
        iii. history with hundreds of entries does not take more than 5 seconds to return search results

4. Speech-to-text (User Story: 5)
    a. sending, receiving, history, and search functions may be triggered with text-based commands (e.g., `send_message: contents`)
    b. unpronounceable characters (e.g., an emoji) are made known to the user (error condition)
    c. the speaker does not say more than is written

   d. the speaker says exactly what is written
   e. empty messages are not spoken
   f. a long message is read in its entirety
   g. repeating a message twice (or as many times as the user would like) is possible per user request
      i. scrubbing the message playback is not possible
   h. punctuation marks are emphasized in some way by the speaker (either audible variations or pronouncing the particular type of punctuation used)

5. Non-English Language Support (User Story: 7)
   a. input handling with non-English support succeeds on the “chat service” acceptance tests (see acceptance test number 1)
   b. a default language is set in the database (English)
   c. a set of non-English languages are supported (not all languages)
   d. a user with no language preference cannot exist in the database (i.e., a user cannot remove their language preference without inserting another)
   e. a nonexistent or unsupported language preference cannot exist in the database
      i. UI dropdown selector is used to select language