DESIGN MILESTONE

ANNIEJ
MARQUEZ
OLGASH

DATATYPE DESIGN

- Login screen
  - Textbox that allows the user to pick a username
  - Buttons that allow user to choose between using an IP address or localhost
  - Users can select a whiteboard from a dropdown list or create a new white-board
- Righthand-side panel provides tools that allows client-end control
  - Buttons that allow clients to choose between drawing and erasing
  - A pallet that holds all of Color's colors. The pallet is clickable by changing the color of the buttons
  - A slider that changes the stroke-size
  - A list that shows the usernames of collaborators
- Whiteboard-specific artsy meter in a panel at the bottom of the screen. We will implement an a algorithm to determine teh artsyness of the board
- The rest of the window is the canvas
- Classes
  - Artist class (Package: client). This class allows the client to enter a username, IP/localhost, and select or create a whiteboard.
  - Canvas class (Package: client). The GUI of the whiteboard client-end white-board. This class implements the above features and passes messages to the server.
  - Whiteboard class (Package: server). This stores the whiteboard information on the server and passes the information to a new client when they connect. Information is stored in a hashmap that allows quick lookup and mutation.

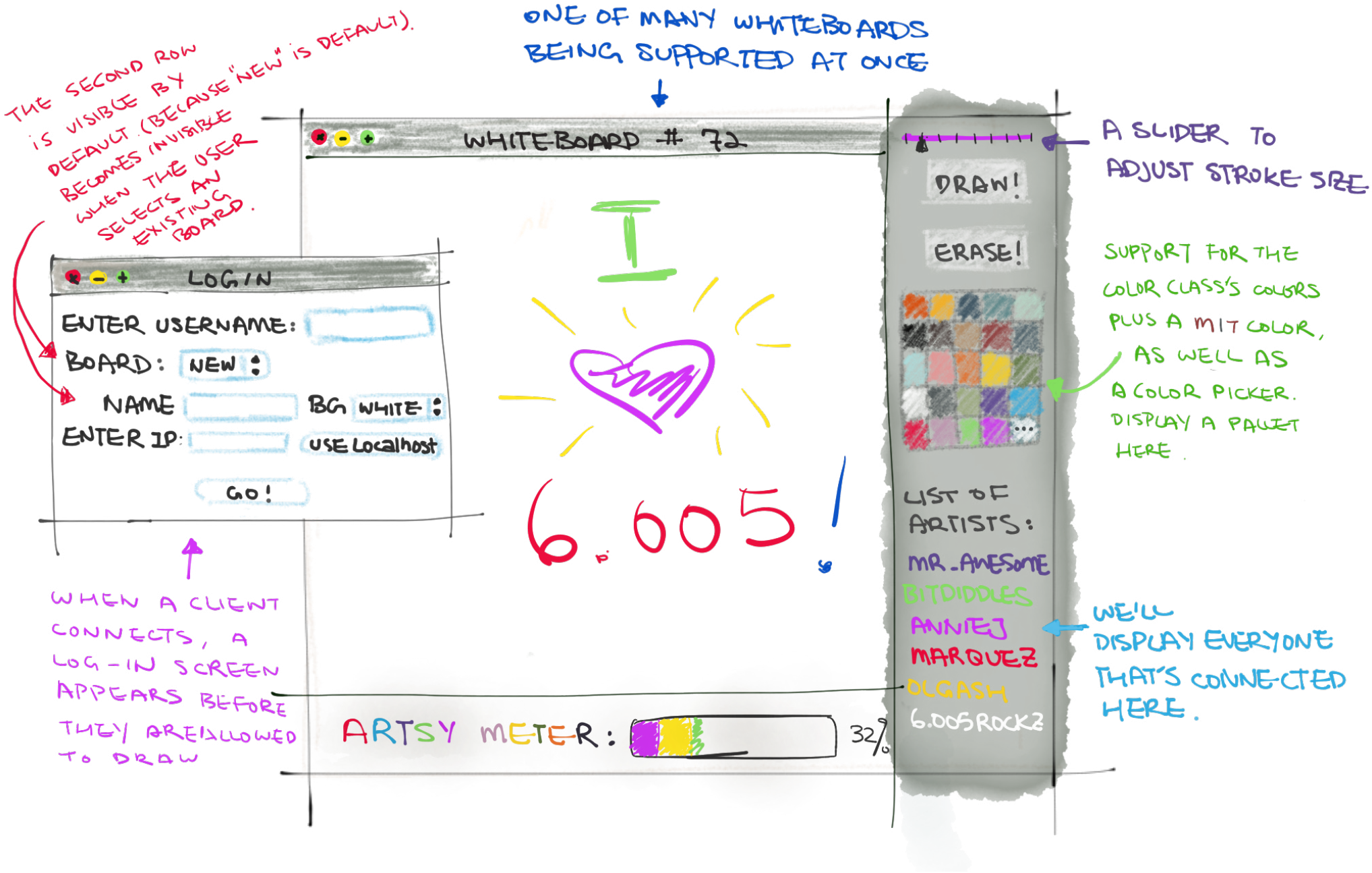
DESIGN MILESTONE

ANNIEJ
MARQUEZ
OLGASH

DATATYPE DESIGN [CONTINUED]

    * Every whiteboard has a list of its attributes
    * Background color --> also changes the eraser color
- WhiteboardServer class (Package: server)
    * The server keeps track of the list of Whiteboard objects, creates new Whiteboards, and communicates to clients

METHODS


THE SECOND ROW IS VISIBLE BY DEFAULT (BECAUSE "NEW" IS DEFAULT). BECOMES INVISIBLE WHEN THE USER SELECTS AN EXISTING BOARD.
ONE OF MANY WHITEBOARDS BEING SUPPORTED AT ONCE
WHITEBOARD # 72
A SLIDER TO ADJUST STROKE SIZE
DRAW!
ERASE!
SUPPORT FOR THE COLOR CLASS'S COLORS PLUS A MIT COLOR, AS WELL AS A COLOR PICKER. DISPLAY A PALLET HERE.
LOGIN
ENTER USERNAME:
BOARD: NEW
NAME
BG WHITE
ENTER IP:
USE localhost
GO!
6.005!
LIST OF ARTISTS:
MR_AWESOME
BITDIDDLES
ANNIEJ
MARQUEZ
OLGASH
6.005ROCKZ
WE'LL DISPLAY EVERYONE THAT'S CONNECTED HERE.
WHEN A CLIENT CONNECTS, A LOG-IN SCREEN APPEARS BEFORE THEY ARE ALLOWED TO DRAW
ARTSY METER:
32%

DESIGN MILESTONE

ANNIEJ
MARQUEZ
OLGASH

PROTOCOL

- Server - > Client
  - a list of whiteboard names and ids (WB_ID WB_NAME WB_ID WB_NAME...)
  - lines of commands for previous whiteboard state, separated by newlines ("USERS" USER_NAME USER_NAME... "PIXELS" X1 Y1 COLOR_R1 COLOR_G1 COLOR_B1 X2 Y2 COLOR_R2 COLOR_G2 COLOR_B2...)
  - new draw actions to everyone connected to a particular whiteboard ("DRAW" ARTSY_METER STROKE X Y COLOR_R COLOR_G COLOR_B)
  - new client joins ("NEWUSER" USER_NAME) to all but new client
  - change background color ("BG" COLOR_R COLOR_G COLOR_B)
  - a client leaves ("BYEUSER" USER_NAME)

- Client- > Server
  - initial connect message (to create new Person object), request whiteboard names ("HELLO")
  - select whiteboard (add user to the whiteboard, set user name, return whiteboard state) ("SELECT" WB_ID USER_NAME)
  - make new whiteboard (with color, name), like selecting, but new ("NEW" WB_NAME COLOR_R COLOR_G COLOR_B USER_ID USER_NAME)
  - new draw actions ("DRAW" WB_ID STROKE X Y COLOR_R COLOR_G COLOR_B)
  - change whiteboard bg color ("BG" WB_ID COLOR_R COLOR_G COLOR_B)
  - disconnect message ("BYE" WB_ID USER_NAME)

DESIGN MILESTONE

ANNIEJ
MARQUEZ
OLGASH

CONCURRENCY STRATEGY

- During a mouse drag event, packets of pixels and their color is sent to the server and the server modifies the Whiteboard's hashmap to reflect the changes of pixel colors.
- The hashmap is synchronized for concurrency
- The method setColor is also synchronized for concurrency
- ID numbers for users and whiteboards will use AtomicIntegers as incrementing counters
- Whiteboards are treated independently and do not share information
- Each user keeps track of their stroke history. The log in screen and whiteboard GUIs do not share objects, but the user ID and username is shared in message passing
- No shared objects; only messages passed in final sockets
- Client has a blocking queue that processes the server messages relating to the drawing/erasing of other users on the same whiteboard; incoming and outgoing messages are on separate threads
- Only call UI repaint in Swingutilities.invokeLater()

DESIGN MILESTONE

ANNIEJ
MARQUEZ
OLGASH

TESTING STRATEGY

- Practice drawing to make sure that the UI works as planned
- Receiving and sending commands both using local host and IP address
- Test multiple users sharing one whiteboard and multiple whiteboard support
- Concurrency testing
  - Make sure behavior is as expected when one user draws and another user erases common pixels
  - Multiple users drawing over common pixels
  - Strokes' real time behavior combined with erasing
  - Two strokes of different colors intersect, same result is synced across different clients
  - Acceptable lag time
- Check to see that whiteboard state is saved during disconnect/reconnect

ERROR HANDLING

- Placeholder