CSCI 5030 Course Project  
Fall 2020  
User stories / Requirements

Suppose you’re meeting with a client who wants to change the world of working from home by building team coordination and meeting software. Video conferencing is so 2019! When you meet in a room you have a desk! You have walls! You can put sticky notes everywhere! Your client’s ambition is no less than building the *virtual meeting room of the future*.

The following is a list of possible directions your course project can take. You and your team will jointly fulfill the role of the product owner in deciding what features to work on. Your only requirements are that the software allows multiple users to interact through a standard web browser window. First, two notes:

*You may assume that your users have a separate voice and webcam connection, via Zoom, Google Meet, Cisco Webex, etc.* You do not need to build these features into your software.

*Non-functional requirements are not the focus of this project*. Real projects are concerned about efficiency, security, scalability, etc. Your software should be able to handle a handful of connected clients and not do anything obviously dangerous, but in general you do not need to worry about these. You are welcome to worry about these if you want, but it’s not necessary or expected.

Possible Meeting Activities

Agile developers use a number of meeting activities to facilitate group interaction. One or two of these would be great starting points when thinking about features to implement. The names of these activities are clickable links that take you to a website with further description.

[*Scrum Board (click here)*](https://www.agilealliance.org/glossary/taskboard/) *–* A wall or whiteboard is divided into three sections – “to do,” “in progress,” and “done.” At the start of a sprint, Scrum Team members write the name of their tasks on sticky notes and place them in the “to do” column. Over the course of the sprint tasks are moved through the sequence as they are completed, or tasks may be added or removed.

[*Planning Poker (click here)*](https://www.agilealliance.org/glossary/poker)– Everyone is given a set of cards. A common set might have the values 1, 2, 3, 5, 8, 13, 21, 35, 50, 100. The speaker describes a feature or a user story that could be implemented. Each player picks a card that they think represents the difficulty of implementing the feature, and places their card face-down on the table. Once everyone has placed a card they are all flipped.

[*Safety Check (click here)*](https://adtmag.com/articles/2018/07/03/agile-safety-tips.aspx) – All members of an agile team must feel able to voice problems and concerns in order for the team as a whole to be functional and progress in self-improvement. In the Safety Check all team members **anonymously** describe how comfortable they are expressing themselves in the present meeting setting.

*Silent Writing* – Some group members may talk a lot and dominate discussions, steering the conversation away from others who may have more important things to say. Group members spend ~10 minutes writing silent observations on sticky notes, which are then publicly categorized and each cluster of observations are discussed by the group as a whole.

*Effort-Reward Charting* – Write any “to do” features on sticky notes. Draw a graph on a whiteboard with “Effort” along the Y axis and “Reward” along the X axis. Team members place the features on the graph at the appropriate places. The team discusses the completed chart and determines what work to do next.

Other Features

These are possible features that could apply to the system as a whole. Some of these might be appropriate right away- others will not be appropriate or even make sense until the system has matured somewhat. This should be a starting point- feel free to add any features you think would be neat or valuable.

* Lines – If users could draw lines on the workspace then they could create their own organizations, such as making their own Scrum Board activity that has a fourth column for “blocked” tasks that are waiting on something else to happen.
* Shapes – If users could draw circles, squares, triangles, etc. that would round out the drawing capabilities.
* Colors – It’d be great if things like sticky notes, lines, shapes etc. could have different colors to make organization easier.
* Mange/Select Items – If we want to have more thana couple items on our workspace then we need a way of selecting what item we’re currently interacting with and modifying.
* Delete Items – I’m sure someone will make a mistake and want to remove a line, shape, sticky note, etc. at some point.
* Drag Items – It’d be nice if we could drag items from place to place instead of clicking and having them magically appear, so other users would have an idea of what each other are doing.
* User List – It’d be nice if users could give themselves names when they connect and have a list of all connected users.
* Magic Hands – It’d be nice if the mouse cursor of each user was represented in the workspace by a little hand icon with a name or color or something so each connected user has a sense of “presence.”
* Connection Status – It’d be nice if the user list would give an indication of whether or not each client is currently responding to the server. The server could send a “heartbeat” signal/response to each browser every few seconds to manage this.
* Connection Status 2 – It’d be nice if your browser gave you a larger and more visible alert if you lost connection to the server.
* Panning – It’d be great if the workspace was larger than just a browser window size, and I could scroll around a larger workspace.
* Zooming – It’d be nice if users can zoom in or out to get a higher or lower level view of the board as a whole.
* Save/Load Boards – It would be nice if a board could be manually saved and downloaded to a user’s computer, and then loaded from that file for a future meeting.
* Periodic Backups – It would be nice if the server would periodically save the current state of active boards to avoid losing data during a crash or power outage.
* Automatic Restore – It would be nice if the server automatically loaded the last backup it performed when it started.
* Encrypted Communications – It would be nice if data between the browser and the server was encrypted. Note- for this class a simple encryption algorithm like ROT13 would do just fine.
* Bounding Boxes / Debug Mode – It’d be nice if we could have the user interface draw the bounding boxes that correspond to what areas are clickable or not for all objects in the workspace.