TSH Video Platform Domain Model

## Introduction

This document describes the current definition of the domain model for the video platform. Inspired by Domain Driven Design (DDD), it’s purpose is to define and elaborate the key business domain concepts and terminology (“ubiquitous language”) in a format that is understandable by subject matter experts, product management, design, and technical audiences.

The evolution of this document will start with a high level definition of the business domain concepts and be refined and elaborated over time as our understanding of the business requirements develop. Our emphasis will be on the development of the concepts that inform the data model, and avoid aspects of the concepts already covered by design thinking concepts (e.g. personas).

## High Level Definitions of Key Concepts

Course A *course* describes an offering (paid or unpaid) that is analogous to a course in a university. It describes the overall content (e.g. “Moving Together, Sitting Only”) manner of delivery (e.g. 24 sessions each 1 hour taught by a live instructor via interactive video), cost, participant requirements, etc.

Course Catalog A place, such as a web page, that lists all current courses. The audience is the general public who are interested in learning more about courses offered by TSH.

Class A *class* defines the specifics of offering a course to which a set of participants have signed up for. Drawing on the school analogy, a course might be “Composition 101”, whereas the class might be “The fall class of Composition 101 meeting 10 AM Tuesday and Thursday every week in room 200”

Session A *session* is a particular meeting of a class, at a specific place and time. For example, “The 10AM Tuesday, October 5th session of the fall class of Composition 101”

Instructor The person teaching a class or session

Coordinator A person supporting the instructor

Participant A person signed-up for a class and attending its sessions

Prospect A person being considered for participation in a research study, pilot, or customer program.

Caregiver A person accompanying a participant and attending a session

Dyad A caregiver and participant who participate in classes together

Login Alternatively an “Id” or “Identity” or “user”, this refers to the unique identity that the system associates with a specific user of the system.

## 

## Detailed Definitions of Concepts

### Course

Much like a course in any school, a TSH course describes the details of the course, the goals of the course, and prerequisites and capabilities of the participants and their equipment, etc.

A course has at least a full name, description, and acronym (brief abbreviation used by convention to name certain resources associated with the course).

#### Some other thoughts

Define a roster of qualified instructors. This could drive the available instructors shown when assigning instructors to a class

Define default schedule parameters for classes, including:

1. Number of times it meets per week
2. Duration of session and lobby

Define states for a course to reflect availability in the course catalog:

Waitlisting The course is visible in the course catalog , and thus is visible to the public. However, prospective students cannot sign up for the course either because all scheduled classes are full, or there are currently no future class schedules. Prospective students can indicate interest in the course and request some type of notification when space becomes available.

Open The course has future scheduled classes with space available; prospective participants can sign up for a class.

Closed The course is closed for signups or waitlisting.

### 

### Class

The definition of a class includes these values:

1. Full name, description, and acronym for the class
2. The current instructor teaching the class
3. The list of participants who are currently signed up for the class
4. The generic schedule for sessions (number and duration of sessions, specific dates and starting times, chronological pattern for scheduling classes)
5. The maximum number of participants for the class
6. Start date
7. The current assigned seating (order that participants appear in group views)
8. The course this class is teaching (one-to-many relationship from course to class)
9. Other default parameters of the sessions, including duration of pre/post group sessions, etc

#### Some Other Thoughts

For the future, we will likely add concepts of *open* classes (classes that are available for signup of participants) vs. closed classes (no longer available for signup). We may also support waitlisting, giving the users the ability to be waitlisted for filled classes.

### 

### 

### Session

##### Definition

The *definition* of a session holds these values:

1. Full name and acronym for the session
2. The class and course associated with the session (a one-to-many relationship from class to session)  
   Note: We expect that ad-hoc sessions will need to be created that are not associated with any class. This would be the case when a user joins a practice session with a coordinator for pre-class checkout.
3. The nominal time at which instruction is scheduled to start (“scheduled time”), the duration of the class, the “lobby” time before and after the instruction. Whereas a class sets the expected schedule of its sessions, each session holds the actual date/time that the session was (or will be) held. As an example, this might be different from the class’ schedule if the session was moved due to a holiday.
4. The instructor who will teach the session. When class instructors change, the instructor associated with the class will reflect the new instructor, whereas the instructor associated with the session will reflect the instructor who actually conducted (or is expected to conduct) the session.
5. ~~The list of participants who actually attended the class.   
   Note: We will have to define a precise definition of attendance. A participant is determined to have attended the class if they join the session is Ready or In Progress.~~Note:We initially used this to store class attendance, but moved to an append-only attendance table that records when a user joins a session.
6. The setting of any class-level values that have been overridden for the session. This might include duration, recording settings, etc. Currently, only the instructor and help messages can be overridden at the session level.

##### 

##### Time-Based States

A session has these states, *which are based solely on the current time and are not mutually exclusive*:

Scheduled The session will be conducted some time in the future and cannot currently be joined.

Open Participants are able to join the class and participate in whatever activity is currently underway. Generally, the class is open during the lobby time and when the class is In Session.

Closed The session is not Open

In Session The session goes to this state when the scheduled time has passed, and continues in this state for a length of time equal to the duration of the class. The instructor should be actively instructing the class and participants will generally be viewing the instructor (“instructor only view”)

Completed The session is closed (not open) and the time for the lobby to close has passed.

Expired The session is closed, and all in-memory information about the session is removed from the server

This time-based state controls the presentation of session information in the landing page. For scheduled sessions, users are shown information, including the time remaining until a session becomes open. For open sessions, the user is invited to enter the session. Completed sessions are not displayed on the landing page.

##### 

##### Dynamic State

In addition, we keep additional stateful information about the *current state* of the session which depends on activities and events within the session:

Active Users The users who are currently participating in the session (note that users can only be active while the session is open).

Active The session is active if it is open, and there are active users.

Current View The current view defines the contents of the classroom for instructors. This includes *instructor view* (participants see only the instructor), spotlight view (participants see specific “spotlighted” user) and group view (participants see all other participants and the instructor).

Recording Session recording has three states: Off, recording, and paused. Recording is initially off until there are active users, and is automatically turned off when the session becomes inactive. In addition, instructors can pause recording, which stops recording independent of the active users.

### 

### Prospects

Prospects are people who express interest in attending a class as part of a research program, pilot, or commercial offering. Generally, they will respond to targeted outreach such as advertisements, plan promotion, selection via medical records based on likely dementia diagnosis, etc.

Generally, prospects will become participants for a research study, pilot, or client commercial offering as the result of two processes whose details and complexity depend on the context:

Enrollment Users are screened in terms of their cognitive abilities, suitability of equipment, and possibly their consent to abide by the applicable terms and conditions. The successful outcome for pilots is that a prospect is enrolled in a Moving Together class.   
  
For research studies, the process is generally more complex and requires several steps over time, involving multiple interactions with the AV platform for consent, evaluation, equipment and connectivity testing. These interactions will generally be managed and conducted by trial managers and class coordinators for research studies, but might also also be self-directed.   
  
For research studies, enrollees are randomly selected to receive the “intervention” (Moving Together class) or control group (no participation). Note that for blind studies, trial managers who are responsible for evaluating the progress of enrollees are not able to determine which enrollees are part of the control vs. intervention groups.  
  
For pilots, the enrollment process may be as simple as a single step where prospects complete an evaluation questionnaire, conduct a self-directed equipment and internet test, select a class, and enter payment information. In this case, enrollment and onboarding might be combined into a single step.

Onboarding This process converts the prospect into a participant. Depending on the context, this process is generally a single step which involves final assessment by an instructor or coordinator, equipment setup, class selection, and payment.

Assessment This process samples and compares measures of cognitive abilities and levels of satisfaction taken over time. The baseline may be defined as part of the enrollment or onboarding process, or be the initial step of the evaluation process.  
  
For research studies, the steps are triggered by the timelines of the study (e.g. after 12 and then 24 weeks) and are conducted by trial managers.  
  
For pilots these will be a series of self-reported satisfaction surveys completed by participants triggered at specific milestones in their class.

### 

### Passwordless Login

The term passwordless login refers to the ability of a user to follow a link that takes the user to Moving Together. Information embedded within the link is used to *identify* *the user* once they arrive at the site. It is important to note that the link *only* identifies the user; it does not in itself determine what they can do once they arrive at the landing page.

Instead, the landing page looks up the user identified by the link in the class database to determine which classes, if any, that user is signed up for, displays the class(s), and allows the user into the class if a session is open.

Because the link is not specific to a class (or any other activity on the site), user’s don’t need to be given separate links to enable them to attend a class or test sessions. A single link reduces any confusion or errors that might be caused by following the wrong link; users have only one link to manage.

Passwordless links are communicated and stored via insecure means, and because of this lack of security, they infer limitations on the types of access that the user has once authenticated via the link.

##### Tickets and Tokens

A passwordless *ticket* is simply a long, randomly generated unique identifier for a user in the system. Their length and randomness make them difficult to guess since the search space is large (and sparsely populated) and the sequence of generated identifiers are not predictable.

The ticket is attached to a link to the landing page; when the user arrives at the site, the site can read the ticket and look up information about the ticket in a table. The table contains the actual user’s identifier, the role that the user will assume once authenticated, and other restrictions of their participation in the site. Their role will set *permissions* on general actions that they can (or more typically, cannot!) perform. For example, the roles given to prospects and participants will not give them permissions to perform administrative actions, view other user’s attendance records, etc. Furthermore, prospects will not be given permissions that allow them to attend Moving Together classes.

A *token* is a set of information about a user that is used as an efficient way to describe the user, some of their attributes (unique identifier, nickname, email address, etc) and permissions that the user has by virtue of the role(s) they have assumed. It’s a lot like having a passport that gives your name, your citizenship, and has a visa attached giving you permission to enter and stay in a certain country. When you arrive in another country with your passport, authorities there can inspect the passport, verify your photo (and other security features), and let you into their country without having to call up your home country to verify the information.

When a user follows the link with a ticket, the ticket is looked up in a table to determine the user and additional information (including permissions) and generate a token with all of this information in a standard format. This token is passed back to the browser which presents the token with subsequent requests. Digital signatures are used to keep the token from being modified while in the browser, and so that the server can trust the token by verifying the signature. The server will then limit your actions to those defined by your token.

The use of the token has the advantage that once a token is generated, it works just like tokens generated through a full username/password login, and it can be passed between systems to represent the user and their permissions without the danger of it being tampered with. Also, since tokens are self-contained, there is no need to go back to the table containing all of the ticket information every time it is needed.

Tokens also have a limited lifetime; that is, after a defined period of time they expire and are no longer trusted by the server. This means that if we make changes to the permissions associated with a passwordless ticket (by modifying data in the table), we can limit to how long tokens with the old permissions can be in effect.

##### Username/Password Login

For instructors, coordinators, trial managers, and even for participants that need to perform actions that require stronger assurance of their identity, we require that they authenticate via username/password instead of using a passwordless link.

The secure and trusted management of basic user information, including password storage, is performed by a third party identity management system, Auth0. Inside this system, each user is assigned a unique identifier (“user id”) that can be used to link together all the information about a user. Auth0 also implements a full suite of identity management functions, including two-factor authentication (2FA), email- and SMS-based password reset, etc.

This arrangement also provides centralized sign-in (often called *Single-Sign On*, or SSO) so that our platform can be built of separate components, all sending users to Auth0 for login once, and then receiving a secure token (signed by Auth0) that contain trusted information about the user and their permissions on subsequent logins.

For prospective users that require access limited to the AV platform, and who do not require full username/password authentication, we will not store their identities in Auth0, or use Auth0 tokens. Instead, these users will be assigned a user ID specific to the AV platform, which will issue its own tokens. These user IDs are created when we create a passwordless ticket for the user as a prospect.

Users that are enrolled in a Moving Together class (i.e. become participants) will have their identity provisioned into Auth0 and their passwordless ticket updated with the new user Id. For these users, their existing passwordless link will continue to work. When they are performing actions that do not require full username/password authentication, they will continue to use a passwordless token issued by the AV platform. If they need to be fully authenticated, they will be redirected to Auth0 authentication, and be redirected back to the AV platform with an Auth0 token containing updated permissions.

Administrators, coordinators, auditors, and trial manages will always require full username/password authentication against an Auth0 user id.

#### More Information

1. The actions available to a specific user is controlled by a Role Based Access Control (RBAC) model. Briefly, that model assigns a role to each user (i.e. “Instructor” and “Participant” are roles), and gives each role a set of permissions. These permissions define and guard the actions that a user can perform. See [RBAC in Together1](https://docs.google.com/document/d/16VoI9QtNRQ-1KDx2t7f91enHbiQABgwTFVS76SWGt30/edit?usp=sharing) for more details.
2. From a software architecture perspective, splitting up data into the aspects required by each major component (the AV platform being one component) is an important concept to achieve modularity. In this case, the AV data, specific to the AV platform, is stored in the AV component’s database. Likewise for the user’s data specific to onboarding would be in a separate component for onboarding, and user data specific to marketing and recruitment are stored in their respective components, each linked back to their central identity in Auth0.   
     
   Those familiar with legacy database modeling approaches might find this idea foreign, since they would be used to a single, unified, and highly coupled model (and storage) of the data in normalized, relational form. However, decoupling the data into separate storage for each component allows us to choose completely different technologies for each, and allows them to evolve independently. A side benefit is that personally identifying information is separated from the information in each component, improving privacy compliance and security.

### 

### **Ad-Hoc Sessions**

Ad-hoc class sessions include AV sessions conducted outside of a formal Course/Class schedule for training, troubleshooting, clinical evaluation, or prospect/participant communication. It uses the AV platform to provide an identical experience to class participation, leveraging that knowledge or providing opportunities for training on the use of the platform.

We expect to support the following use cases:

1. Scheduled session: The session is scheduled in advance at a specific time and pre-defined duration. The prospect/participant follows their passwordless link, and they are prompted to enter the class at the appropriate time.
2. Unscheduled (“On-the-Fly”) Session: The session is NOT scheduled in advance. The user acting as instructor tells the prospect/participant to follow their link, where they see the usual notice that class is in session.

In both cases, the session is participated in by a trial manager, coordinator, or instructor acting in the instructor role, and with a specific assigned prospect or participant. For scheduled classes, the specific person fulfilling the instructor may or not be assigned when the session is scheduled; if assigned, we may need to re-assign at any time.

Future additions in support of observer roles, or ad-hoc session participants will also be supported.

The prospect or participant should be able to join the session using their user-specific passwordless link to the landing page and it works the same way it does for MT classes.

##### Overview of Existing Model for Moving Together

(Static) Data Model

1. Courses have a 1-to-many relationship to classes
2. Classes have an assigned instructor and list of participants, which determines assigned seating.
   1. Participants not on the participant list are not able to join the class
   2. Need to add support for session participation by observers and coordinators (visible and invisible) which is not pre-planned
3. No definition of role for instructor at course or class level
4. Classes have a 1-to-many relationship with sessions
   1. Instructors are assigned at the session level with the default being the instructor for the class. Instructors can be reassigned at the session level
   2. There is no assignment of participants at the session level
   3. Sessions are subdocuments of classes
5. Landing page query
   1. Participants: query classes based on participant, then filter sessions based on date/time
   2. Instructors query instructor at class level and at session level, then filter sessions based on date/time

In-Memory Model

1. Class and session (static) data is available in-memory with the session. Static data is not modified
2. Data model defines instructor and assigned seats. No way to reassign instructor in runtime
3. Active list of participants is maintained as they join/leave the session
4. Inconsistent use of instructor assigned to class and roles/permissions to enable and determine instructor
5. Currently an API to get data for all active sessions

##### Approaches to Supporting Ad-Hoc Sessions

1. Create new Ad-Hoc Sessions for AV Platform
   1. Keep a separate ad-hoc sessions table which combines current class + session
      1. Properties include participant, instructor/role, scheduled time, duration, lobby time (before/after) mostly identical to a MT session
      2. Keep track of reason for the session (Consent, Goals Assessment, Baseline Assessment). This is analogous to the class/course
   2. Modify landing page logic to find ad-hoc sessions and display with others
      1. For participants: Look for sessions for which I am the participant, show times like MT sessions
      2. Same for trial manager/coordinator, but also allow instructors to find participants via admin tool
   3. Create the runtime session when the first person joins, mostly like MT session, but from ad-hoc session data
   4. Will be efficient to find sessions based on time, participant, instructor, role
   5. For on-the-fly
      1. Trial manager initiates from Admin tool when searching for prospect, creates in-memory session on the server
      2. Prospects/participants also search in-memory sessions in landing page
2. One Participant per Class
   1. Participant list is the single participant
   2. Sessions are added with appropriate date/times when they are scheduled
   3. For participant, landing page logic is unchanged
   4. For instructor landing page, assign role to course, and make instructor at the season level optional
   5. Course can be specific to task or process type
   6. Prune unnecessary courses and completed sessions
3. Multiple Participants per Class, One Course per Task Type
   1. Define session-specific participants for ad-hoc sessions.
      1. Overrides assigned seating at class level from the session
   2. For participant landing page, query first by class, then filter sessions for the date/time, and also participant (filtering based on ad-hoc vs MT session type)
   3. For instructor landing page, assign role to course, and make instructor at the season level optional. Filter classes by role or instruct at class level, and at session level by date/time and instructor
   4. Course is specific to process type
   5. Prune unnecessary participants on each class (don’t have any sessions) and completed sessions
      1. Potential for performance issues with many participants assigned to a class

In all cases:

1. On-the-fly sessions can be handled by the person acting as an instructor first creating a scheduled session that starts immediately
2. Acting instructor can go to a separate landing page for ad-hoc sessions; participants cannot
3. Differentiate ad-hoc courses, classes and sessions as ad-hoc vs MT-type

## 

## User Stories

1. For new prospects in RedCap, Trial Manager needs to **create prospect** by entering basic information about a prospect and remember to schedule/track consent session for a future time
   1. Enter basic prospect information (SID, Name, email, phone)
   2. The prospect show up in list “Ready to Schedule Consent”
   3. Passwordless link is generated and available to email to the user
   4. TOG-285
2. For new prospects in RedCap, Trial Manager, Class Coordinator, Instructor needs **to schedule consent meeting** (first step in the process)
   1. Go to “Schedule My Sessions Today” (actual name TBD)
   2. Find and filter by session type
   3. Go to prospect and ready to be scheduled list and find prospect to schedule meeting for
   4. Select date/time for consent meeting
   5. Copy and manually email user link
   6. The trial manager may also note the time in the personal calendar or shared TSH calendar
   7. Once prospect consent meeting is scheduled, they are removed from “Ready to Schedule Consent” list and added to “Conduct Consent”
   8. TOG-286
3. Trial manager (Class Coordinator or instructor) needs to see **what sessions they have scheduled** for today/tomorrow/this week so that they can organize their time
   1. Go to “My Sessions Today” (actual name TBD)
   2. See list of sessions scheduled for today/tomorrow/this week, sorted by time.
      1. See name, nickname, other contact info (email, phone) type of session, link to more information about prospect and session
      2. Filter by type of session
      3. Text search by name/contact info
   3. Click on session to go to the AV landing page
   4. TOG-287
4. Trial Manager wants to **conduct scheduled session** they have noted on their personal calendar
   1. Log in to landing page on AV platform at the time of the session and sees the scheduled sessions, including type and basic information about the prospect; shows if the session is open, or how long until it starts
   2. Click on a meeting and enter
   3. TOG-288
5. Trial manager needs to **reschedule a meeting for a specific prospect**
   1. Go to prospect list and see meetings and update if needed
   2. Find prospect via name, SID, contact info, etc
   3. (Somehow) see uncompleted scheduled meeting and reschedule
   4. TOG-289
6. See meetings scheduled for a prospect
   1. Go to prospect list and see meetings for prospect and update if needed
   2. Find prospect via name, SID, contact info, etc
   3. (Somehow) see uncompleted scheduled meeting and reschedule
   4. TOG-290
7. Reschedule meetings in My Sessions Today
   1. Go to “Schedule My Sessions Today” (actual name TBD)
   2. Find and filter by session type
   3. Go to prospect and ready to be scheduled list and find prospect to schedule meeting for
   4. Select date/time to reschedule meeting
   5. Copy and manually email user link
   6. The trial manager may also note the time in the personal calendar or shared TSH calendar
   7. Once prospect consent meeting is scheduled, they are removed from “Ready to Schedule Consent” list and added to “Conduct Consent”
   8. TOG-291

--------------------------

1. Trial manager (or instructor) wants to **keep track of sessions that have been completed** and note the outcome
   1. The session is marked as complete automagically based on duration
   2. The trial manager has the option to mark the outcome of the session when leaving, based on type of session
   3. Based on the type of meeting and the outcome, the prospect may be marked as rejected and removed from active prospects list, or be marked as ready for the next session in the process
   4. Conducted consent meeting status is updated
2. Trial manager want to note **that a prospect is rejected or drops from the project** to focus active prospect list
   1. Find prospect via name, SID, contact info, etc
   2. Mark prospect as rejected (or removed for some other reason).
   3. Prospect is removed from the active prospects list and waitlisted state

### **Note: I don’t think that we need additional information on these concepts at this time:**

### Instructor

### Coordinator

### Participant

### Trial Manager

### Caregiver