INL/EXT-19-55789

Light Water Reactor Sustainability Program

Report for 2.2.1 Task 4: Software-Based Tools to Support Human-System Evaluation Studies

September 2019

U.S. Department of Energy

Office of Nuclear Energy

INL/EXT-19-55789

Report for 2.2.1 Task 4: Software-Based Tools to Support Human-System Evaluation Studies

Michael Hildebrandt, Jens-Patrick Langstrand, Hoa Nguyen

Jeffrey C. Joe, Project Manager

September 2019

Prepared for the
U.S. Department of Energy
Office of Nuclear Energy

IFE/INL-196543

SOW 14512

Report for 2.2.1 Task 4: Develop and document software-based tools that will support and enhance the modeling, collection, and analyses of eye-tracking data that are collected in human-system evaluation studies

Institute for Energy Technology

	KJELLER	HALDEN
Address	NO-2027 Kjeller, Norway	NO-1751 Halden, Norway
Telephone	+47 63 80 60 00	+47 69 21 22 00
Telefax	+47 63 81 63 56	+47 69 21 22 01

Report number	Date
IFE/INL-196543	2019-09-12

Report title and subtitle	Number of pages
Report for 2.2.1 Task 4: Develop and document software-based tools that will support and enhance the modeling, collection, and analyses of eye-tracking data that are collected in human-system evaluation studies	26

Project/Contract no. and name

Contract no. 196543

Client/Sponsor Organization and reference

Idaho National Laboratory, USA

Abstract

INL contracted IFE to develop custom software functionality to support INL's control room simulator data collections. The tool described in this report provides functionality for authoring, managing and executing so-called *micro task* data collections (Hildebrandt and Eitrheim, 2015), where operators receive a long list of operational tasks, and the speed and accuracy of their response is measured. The tool consists of an editor, player and observer module. This report contains methodological background and a user guide.

	Name	Date	Signature
Author(s)	Michael Hildebrandt Jens-Patrick Langstrand Hoa Nguyen	2019-09-12	
Approved by	Andreas Bye	2019-09-12	

1 Introduction

Studies of operator behavior in process control rely on a wide variety of Human Factors methods (Stanton, Salmon, Walker, Baber & Jenkins, 2013). The most common approach to process control studies involves the use of scenarios, and in many cases specifically emergency scenarios. Scenarios typically last for 30 minutes to two hours. Researchers often evaluate task performance, workload, communication, teamwork, or other measures of interest. The advantage of scenario-based studies is that they closely approximate real control room operations and therefore provide a high degree of ecological validity (Brewer, 2000). The problem with scenario-based studies is that they generate relatively few data points per unit of time. For process control studies, it is common to have access to 4-8 crews, and very rare to have access to more than 10 crews. Each crew typically participates for between half a day and two days, during which they will perform between 2-6 scenarios. Given the degree of crew-to-crew variability, this makes it difficult to perform experimental studies that can result in statistically significant differences between conditions. Researchers must consider the trade-off between the number of experimental factors they want to investigate (e.g., interface A vs. B in simple and complex scenarios), and the reduction in statistical power as the number of conditions increases. Although there are approaches to increase the amount of data points, such as dividing the scenario into phases, scenario-based studies can be considered low-density data collection methods.

To address this issue, Hildebrandt and Eitrheim (2015) proposed a high-density data collection method called *micro tasks* (MTs) to supplement traditional scenario-based methods. It is important to emphasize that MTs are not intended to replace scenario-based methods, but work together with these more contextual methods by providing large amounts of data about the efficiency and reliability with which operators can use the control room interfaces.

2 Micro Task Method

The micro task (MT) method is an interface evaluation method. It is based on usability assessment methods from the human-computer interaction domain, where participants are presented with a series of short, well-defined tasks, such as for example “Find the cheapest flight from London to New York on October 15.” For process control studies, a typical micro task would be “Check if all reactor coolant pumps are running.” (with a single response button labeled “Checked”), or “Are all reactor coolant pumps running?” (with response buttons “Yes” and “No”).

The following subsections contain methodological guidance that researchers should consider when developing MT studies.

2.1 Task and Block Duration

MT questions are typically designed to be answered within 3-30 seconds. Questions that take longer than half a minute to answer should be selected with care, as they risk increased response time variability and thereby reduce the statistical power of the study (unless countermeasures such as increased sample size are used). Another reason to keep the questions short and concise is to maximize the amount of questions that can be asked. Experience from previous studies has shown that individual blocks of micro tasks should be kept to 10-15 minutes duration. A reasonable number of tasks for this duration is 30-50. This assumes that most questions are static MTs that can be answered with a single glance, such as “Is the pressurizer level above 80%?”. For dynamic MTs, where for instance a trend must be monitored over time, the number of questions should be reduced accordingly.

2.2 Question Phrasing

It is important that MT questions are phrased as unambiguously as possible. As response time is the primary measure in MT studies, any confusion by the participants will adversely affect the data quality. A good starting point for question generation is to refer to the operating procedures. In most cases it is advisable to word the questions in accordance with the procedure text as much as possible. The reason is

that operating procedures have already been optimized for clarity, and the MT data can be mapped directly on an operational task (which can be important, for instance if the data is used in the context of Human Reliability Assessment).

There are, however, circumstances when it is necessary to deviate from this guideline. One example is when a study evaluates not just the speed, but also the accuracy of the response. Operating procedures are typically written with the instruction "check" or "verify". However, by using the response button "Checked", it is not possible to evaluate if the participant checked the values correctly or incorrectly. In such cases, the question can be reformulated, for instance "Are all reactor coolant pumps running?" (yes/no), or "What is the level in pressurizer?" (numerical response). Another example where it is necessary to go beyond the procedure text are knowledge-based questions. Such questions are typically used when evaluating how well a control room interface supports higher-level cognitive activities, such as situation awareness (e.g., "Are there conditions for a reactor trip?").

2.3 Individuals and Groups

One of the strengths of the MT method is that multiple operators can run through MTs independently of each other, but in the same simulator at the same time. Often only a single simulator is available, and running MTs concurrently with 3-4 operators (instead of sequentially with single operators) leads to much better simulator utilization. There are very few disadvantages of running MT studies with multiple, independently-working operators in the same room, provided they are instructed appropriately so that operators that finish first do not distract those who take more time.

While the MT method is primarily designed for data collection with individual operators, it can be used on a crew level as well. In that case, the MTs are presented to one operator, usually the shift supervisor, who will call out the task to the crew, direct appropriate crew actions, and enter the response.

2.4 Instructions

Since the MT method relies so heavily on task performance time, it is essential that participants are instructed appropriately and consistently. It is recommended that the main instructions about the nature of the study are delivered verbally by the experimenter, for example:

"*In this study, we are interested in how efficiently you can gather information from the panels. We will give you a set of 40-50 relatively easy questions, similar to what you would find in a procedure. We want you to work through them individually, and not talk with each other. We are interested in how quickly you can work through these tasks, so please work accurately, but also at a good, steady pace. If you can't find the information, or get confused by a question, just move on."*

After the instructions, the experimenter should verify that the participants have understood what is expected of them, and potentially reiterate the expectation to work accurately but at a good, steady pace.

The observer function in the micro task tool (see section 4.6) allows the experimenter to identify if a participant works through the MTs excessively fast or excessively slow. If such issues are detected early in the trial, the experimenter must decide if the participant should be allowed to continue or not. Especially if several operators work independently in the same room, an excessively slow participant can lead to long delays for the others.

2.5 Redundancy

As the MT method can generate large amounts of data, it is recommended to include multiple questions on the same issue in order to make the dataset more robust and increase statistical power. For instance, if pump symbols are of interest, 5-8 questions about pumps should be included instead of just one.

2.6 Simulator Support

To answer the MT questions, participants need to have process displays available to them. This can be achieved in anything from a full-scope control room simulator to a single screen. In a full-scope simulator, it is important to consider if questions should be grouped so that they relate to individual or adjacent panels. If participants must walk between panels to collect the required information, the researchers must consider if the transit time to walk between panels will adversely affect the data. For digital interfaces, on the other hand, the researchers should consider if the time needed to navigate between process displays is a relevant factors, and if MTs should be grouped so that they do not require navigation. To lessen any adverse effects of such factors, a larger pool of randomized MTs, with enough redundancy among questions, should be used.

2.6.1 Live Simulator vs. Static Screen Shots

MT studies do not require a live, running simulator. They can be conducted either with the simulator in "freeze" state, or just with screen shots of panels or interfaces. In fact, static interfaces make it easier to create identical conditions for all participants. With the simulator running, especially with a dynamic scenario, participants may receive a given MT at different points in the process evolution, thereby creating a different context for the task, which may influence performance times. In addition, an evolving process makes it more difficult to evaluate if a question has been answered correctly, as the correct answer cannot be pre-defined. Note that the MT tool can be modified to receive live data from the simulator and automatically evaluate response accuracy based on that data. However, this functionality is not within the scope of the current delivery.

2.6.2 Operator Actions

The MT method was developed primarily to evaluate how efficiently an operator can access information from a control room interface. However, for some studies, researchers may want to include operator actions as well (e.g., "trip the reactor"). The micro task tool can be modified to receive signals from the simulator to evaluate if the action has been performed, and performed correctly. Two factors should be considered here.

First, is it actually necessary for the study to receive the data from the simulator? The operators can be instructed to press "Action completed" on completion of the action. While this will not provide automatic evaluation of correctness, the response times will most likely be proportional to the timing data received from the simulator.

The second issue to consider is that if operators can act on the process, they may change the task context for other users. For instance, if three operators run MTs independently of each other, but in the same simulator and at the same time, if one operator trips the reactor, the other operators will not be able to perform this task as the reactor is already tripped.

3 Micro Task Environment

This report describes "Synquesticon" a software tool developed to support the authoring, management and execution of MT studies.

3.1 Features

The core features of the Synquesticon system are:

- Graphical user interface for authoring, management, and stimulus presentation
- Web-based server-client architecture resulting in platform independence
- Database support for storage of study designs and experimental data
- Multi-user support

- Reusable task definitions
- A tagging system for tasks and sets
- Text and image stimuli
- Single-task and multi-task presentation formats
- Organization of tasks into sets
- Randomization of task presentation order
- Different user interface schemes
- Adaptive training based on number of correct responses
- Measurement of total task completion time and time to first response
- Automatic evaluation of response correctness
- Eye tracking support
- "Area-of-interest" editor for eye tracking stimuli
- Live observation functionality
- Real-time annotations by observers
- Progress indicator
- Observers can remotely pause and resume data collections
- Data export as comma-separated values

3.2 Modules

Synquesticon consists of three modules:

- The **editing module** supports the authoring of MT questions and the editing of experimental blocks.
- The **player module** supports the data collection, i.e., presentation of MTs to participants.
- The **observer module** allows researchers to monitor study progression in real-time (including eye movements where available), pause and restart a study, and add comments to data points.

3.3 Core Concepts

In Synquesticon, a study consists of a series of tasks, which can be organized into sets.

3.3.1 Task

The task is the elemental unit of a MT study and can be a question (e.g., "Are all reactor coolant pumps running"), an instruction, or an image. Tasks also have user-definable response buttons, for example "Yes" / "No" in case of a MT question. By default, every task is displayed on a separate page, and tasks are presented in sequence. This behavior can be overridden, as explained below.

3.3.2 Set

Tasks can be organized into sets. Organizing tasks into sets makes it easier to manage larger experimental designs. For instance, an experiment can be divided into sets such as "Demographic questions", "Instructions", "Practice trials", and "Data collection". Tasks within a set are by default presented in sequence, but this can be changed to random presentation. This allows the researcher to present some tasks in sequence (e.g., the demographic questions and instructions), while randomizing other sets (e.g., the main data collection tasks). Sets can be nested, i.e., contain other sets.

As noted above, tasks are by default presented on separate pages. However, for some types of questions, such as questionnaires, the researchers may prefer to present several items on the same page. To support this, sets have a setting whereby all items in this set are presented on the same page.

3.3.3 Tags

Tags can be added to both tasks and sets. Tags are saved in the results files and can help simplify the data analysis. For instance, task tags can describe the type of component or the kind of question (e.g., “pump”, “valve”, “identification”, “decision”). For sets, tags can be used to describe the experimental condition (e.g., “analog” vs. “digital”).

3.3.4 Global Variables

During the data analysis, it is often necessary to group data by various categories, such as the participant ID, the position of the participant (e.g., “reactor operator”, “shift supervisor”), or crew. To support this, there is an option called “make global variable” in the task definition dialog. If set to “global”, every line in the data file will include the value set here (e.g., “crew3”).

4 User Guide

4.1 System Architecture

Synquesticon is a server-client application. For both authoring and data collection, users connect through a web browser. Therefore, no additional software needs to be installed by the client. The preferred browsers for iOS is Safari (as it supports full-screen web pages without a navigation bar), and Chrome for all other operating systems and platforms.

The server-side code can be installed on a laptop (for a local installation), on an intranet (for an institute-wide installation), or on a web server (for remote collaboration or data collection). When run on a laptop, a Wi-Fi router can be used to give access to clients.

4.2 Installation Instructions

The Synquesticon software code can be downloaded as a Github repository using the link that was provided to INL. The following dependencies must be installed:

- Node.js (https://nodejs.org/en/)
- MongoDB (https://www.mongodb.com/)
- Crossbar (https://crossbar.io/)

Once these dependencies are installed, go to the directory that contains the Synquesticon software code. Go to the directory `WebEntry`, and from there open a command console. Enter the command `npm install`. Then go the directory `WebEntry/backend` and enter the command `npm install`. All required components are now installed.

To run the application, double-click on the file `launcher.bat` in the `WebEntry` directory. It can take up to half a minute to launch the server. Synquesticon can now be accessed through a web browser (see section 4.3).

4.3 Home Screen

The MT application (Figure 1) is accessed through a web browser using the installation URL and the port 3000. For a local installation, the URL would be `localhost:3000`, for installation on a local Wi-Fi network, the URL would take the form `<hostIP>:3000` (the `hostIP` can be found by running the command `ipconfig` on the host computer’s console), and for a web server installation, the access URL would be `<serverName>/<installationPath>:3000`.

Figure 1. Synquesticon home screen.

In the left-hand column, under the heading "Studies", a list of all available studies is shown. Any task set that contains the tag "experiment" is listed here. Next to the study title is a play button, which will launch the player module described in section 4.5.

The pen icon next to the "Studies" label will open the editor module described in section 4.4.

The dialog "Eye Tracker" at the bottom of the left-hand column will show a list of all connected eye trackers.

In the center of the screen is the observer module, described in section 4.6. When a study is run, messages about the participant's progress will be displayed here. To the right of the label "Observer" is a "Pause / resume" button whereby the experimenter can interrupt and resume the study on all connected clients.

Next to the "Pause / resume" button is the "Data retrieval" icon for downloading data, as described in section 4.7.

In the top-right corner is the "Settings" button, described in section 4.3.1.

4.3.1 Settings

The global settings are accessible through the gear icon in the top-right corner of the screen. It contains the following functions:

- *Device ID.* A global identification and role label can be set for the device. These labels are used in the observer messages (section 4.6). By default, the ID is set to *Anonymous*.
- *Crossbar Settings* (Figure 2). Crossbar is the software platform used to broadcast and receive messages. While an experiment can be run without Crossbar support, the observer function will only work if a Crossbar server is installed and the proper settings are made on all connected clients. Generally, the Crossbar IP address is the same IP address as the computer that runs Synquesticon (if uncertain, use the command `ipconfig` in the command console of the computer running the Crossbar server). By default, Crossbar runs on port `8080`, and this setting should only be changed if that port is blocked. The default channel for messages is `realm1`, and there is usually no reason to change this setting.

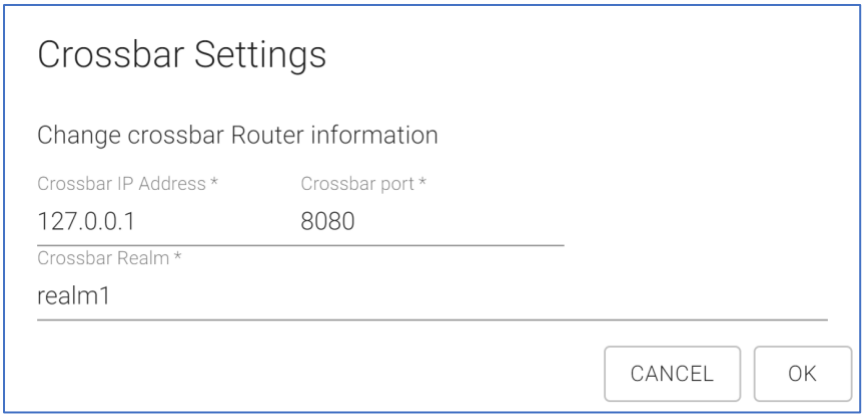

Figure 2. Crossbar settings.

- *Dark Theme.* This function toggles between two user interface themes, a dark theme and a light theme (Figure 3). The selected theme applies to all modules, including the player module. The theme is only changed on the client itself, not on all other connected clients.

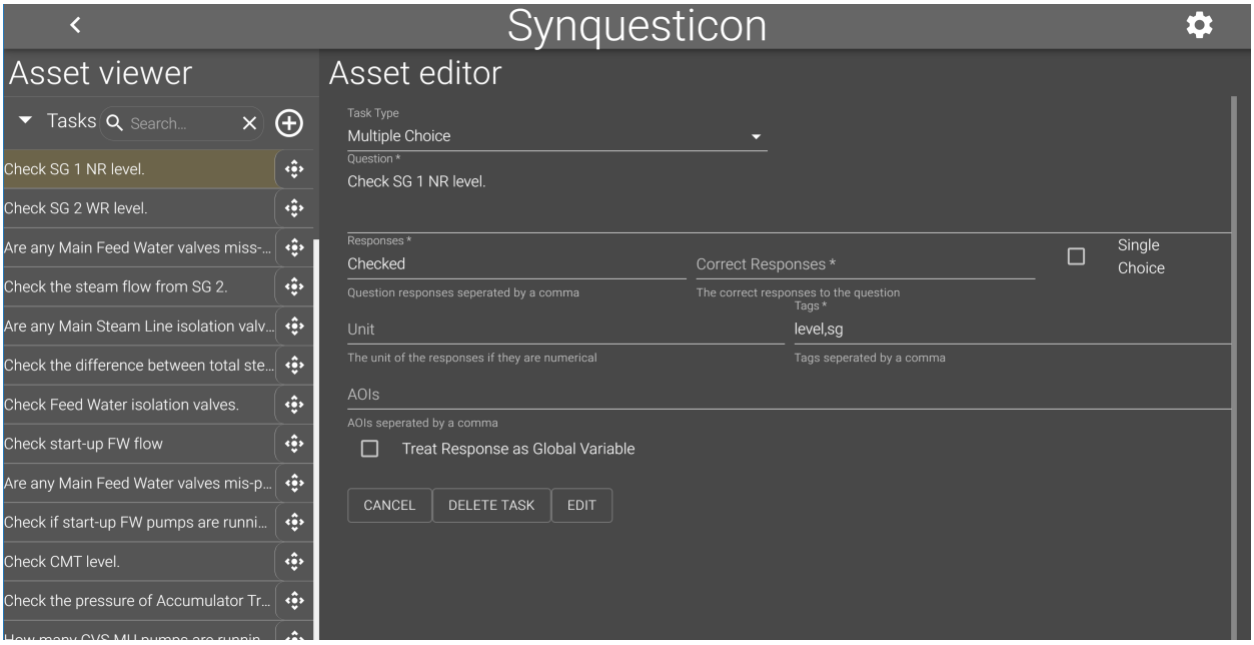

Figure 3. Example of "Dark Theme" user interface.

- *Full screen*. Use this function to switch into full screen mode, removing the web browser's navigation bar. This function is especially important when setting up a device to collect data, as the participant will not be able to reload, go back, or disrupt the experiment flow in any other way.

4.4 Editor Module

The editor module (Figure 4) is where studies are authored and edited. It is here that tasks are defined and assembled into sets (see section 3.3). It consists of an asset viewer section (on the left) and an asset editor section (on the right).

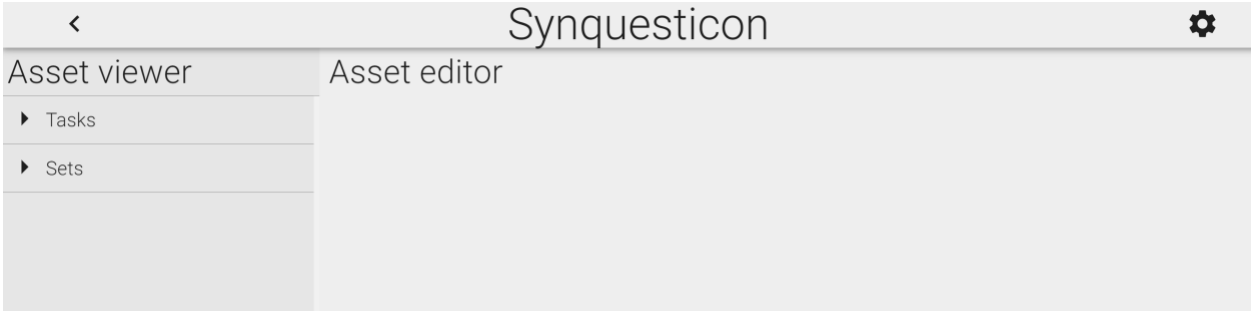

Figure 4. Editor module.

4.4.1 Asset Viewer

The asset viewer provides access to existing tasks and sets, and is the entry point to creating new tasks and sets.

4.4.1.1 Task List

Clicking the triangle next to "Tasks" expands the task list and shows a list of all tasks that have been created (Figure 5). At the top of the task list is a search bar whereby tasks can be filtered by tag or question text. Next to the search bar is an "Add" button. It is used to create a new task.

Clicking on a task in the list opens the task editor for this item (see section 4.4.2.1). Next to the task text is a "drag" button. It can only be used to drag a task into a set (see section 4.4.2.2). It cannot be used to re-order the task list.

Figure 5. Asset viewer with task list.

4.4.1.2 Set List

The set list shows all available task sets (Figure 6). Note that this list is different from the list of experiments on the home screen (see section 4.3). The list on the home screen shows only sets tagged with "experiment", whereas the set list in the asset viewer shows all sets regardless of labels. This usually includes subsets such as "WarmupQuestions" or "DemographicQuestions". The set list provides similar functionality to the task list (search, create new set, edit set, drag set into another set list).

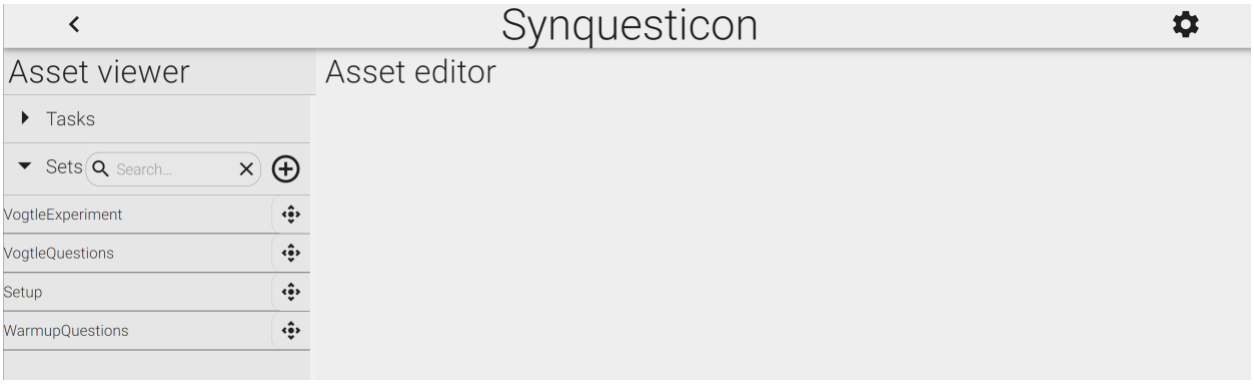

Figure 6. Asset viewer with set list.

4.4.2 Asset Editor

Tasks and sets are created and modified in the asset editor.

4.4.2.1 Task Editor

To create a new task, click the "+" symbol in the task list (see Figure 5). To modify an existing task, click on the task in the task list.

There are four different task types in Synquesticon (see Figure 7):

- Instructions
- Multiple Choice
- Text Entry
- Image

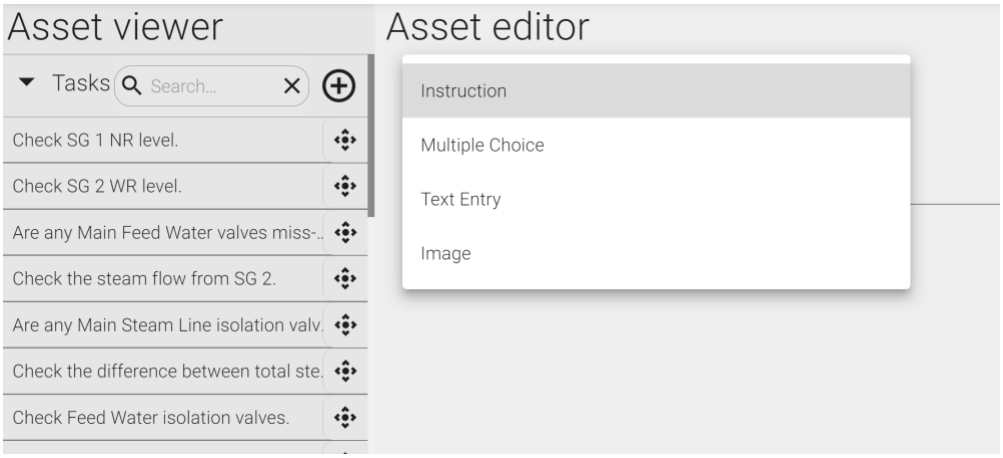

Figure 7. Task types in Synquesticon.

These task types will be described in more detail in the following sub-sections. Note that edits in Synquesticon are not automatically saved. When creating a new task, remember to click "Create" when finished. When editing a task, remember to click "Edit" when finished. Otherwise any changes will be lost. All required fields are denoted with an asterisk (*).

4.4.2.1.1 Instructions

"Instruction" is the simplest task type in Synquesticon (see Figure 8). There are no response options for such tasks, and no right or wrong responses. During data collection, the instructions are shown to the

participant and a “Next” button added at the bottom of the page. The participant’s viewing time will be recorded in the data logs.

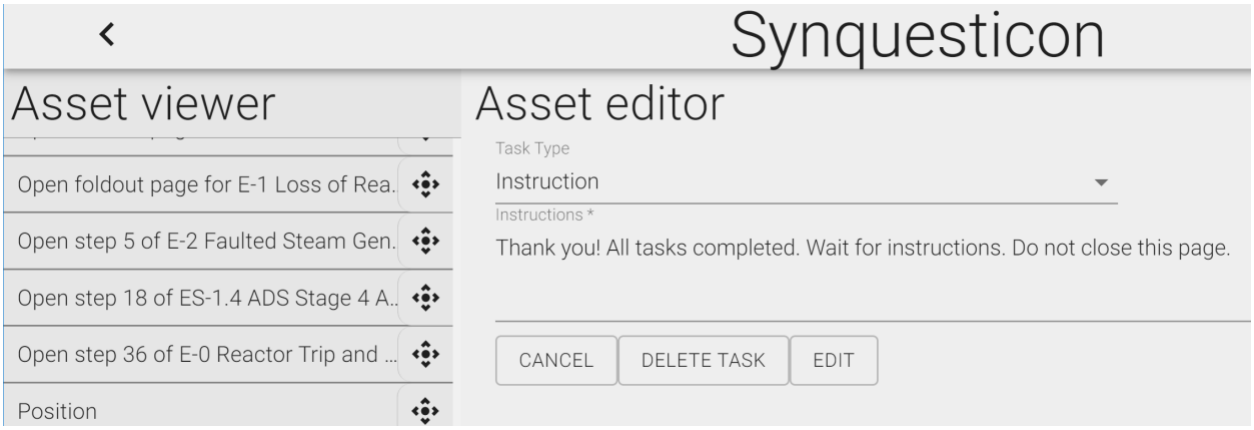

Figure 8. Editor for task type “Instruction”.

4.4.2.1.2 Multiple Choice

“Multiple Choice” is a task type that presents participants with a task text and one or more response buttons (see Figure 9 and Figure 16). For simplicity, this category contains all button-based response, such as

- Single-response tasks, such as “Check that all reactor coolant pumps are running” (response button “Checked”)
- Multi-response, single choice tasks, such as “Are all reactor coolant pumps running?” (response buttons “Yes” and “No”). The participant will only be able to select one button. To enable this category, tick “Single choice” in the task editor.
- Multi-response items, such as “Which reactor coolant pumps are running?” (response buttons “A”, “B”, and “C”). The participant can select one or more response buttons.

Figure 9. Editor for task type “Multiple Choice”.

The question text can be entered in the “Question” field. Below it is the “Responses” field, where the experimenter can define the response buttons as a comma-separated list of labels (e.g., “Yes,No”). To the right of the “Responses” field, the correct responses can be defined as a list of comma-separated labels. These must exactly match the labels in the “Responses” field.

Below the “Responses” field is the “Tags” field. Tags can be entered as a list of comma-separated labels. Any label entered here will appear in the data log for this task.

If the option “Treat Response as Global Variable” is checked, then in the data log the participant’s response to this question will not only be logged to this task, but to every data point for this participant. This function is often used to label every data point with categories such as the crew ID or the participant’s position (see Figure 10).

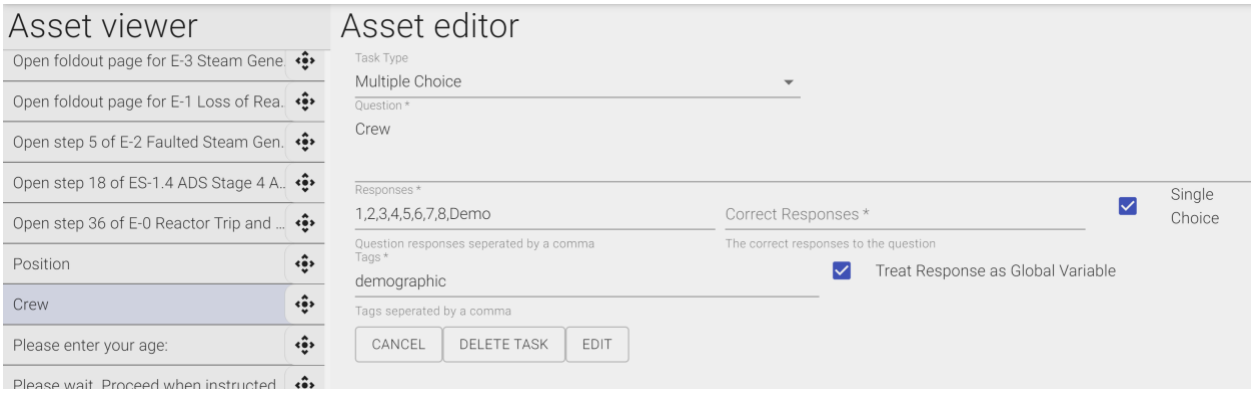

Figure 10. Example of a task where the response becomes a global variable.

4.4.2.1.3 Text Entry

Whereas in the previous task type pre-defined response options are presented to the participant, in the “Text entry” task type, participants can enter values (see Figure 11). This is typically used for questions such as “*What is the pressurizer level?*”. The participant can enter a response through a numerical keypad. The tool currently only supports numerical entry. It also does not support the definition of a correct response. Determining the correct response in process control tasks can be difficult, as analog gauges make it difficult to obtain an absolute accurate reading, and participants may round values to the nearest full number. It will therefore be necessary to provide functionality for entering a range of correct responses. This functionality will be added in future versions of the tool.

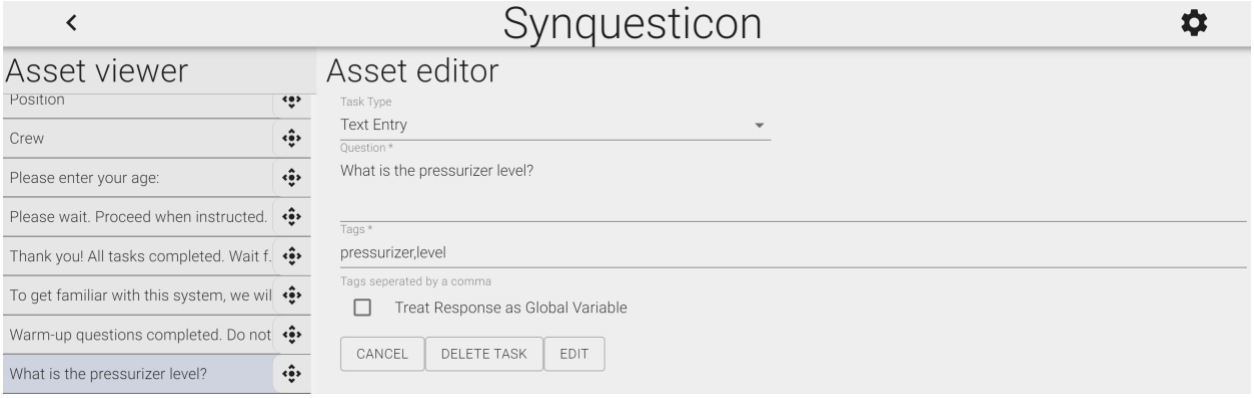

Figure 11. Editor for task type “Text entry”.

4.4.2.1.4 Image

The “Image” task type (see Figure 12) will present the participant with a single static image per page. An image can be uploaded using the “Choose File” dialog and named using the “Task Name” field. Once the image is uploaded, it is shown in the editor.

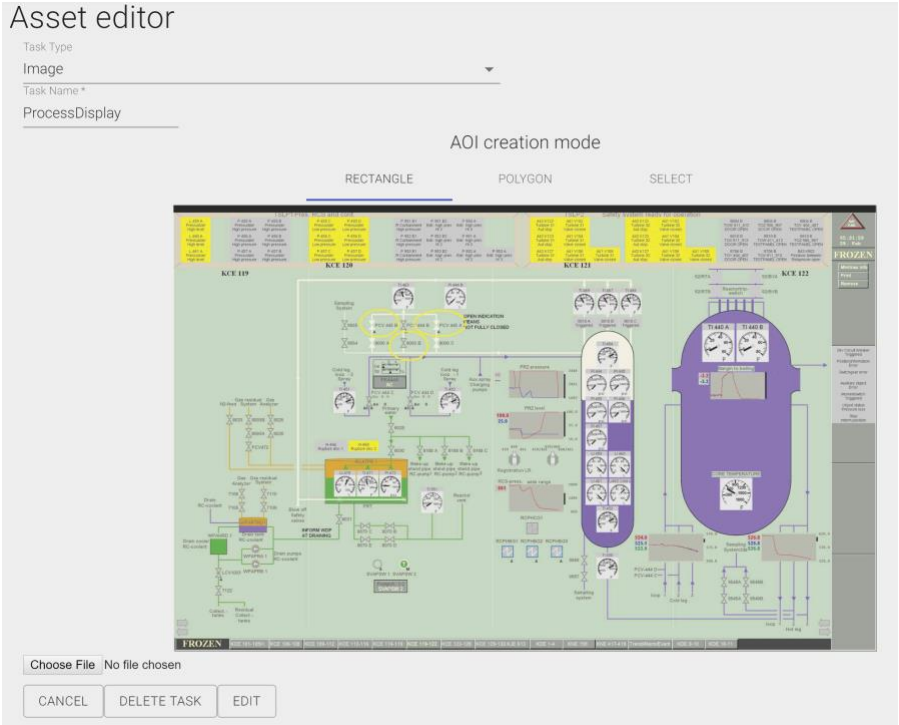

Figure 12. Editor for task type "Image".

4.4.2.1.4.1 Area of Interest Editor

The editor provides functionality for defining areas of interest (AIOs) for use with eye trackers (see Figure 13). AOIs can be drawn into the image as either rectangles or polygons. AOIs can be renamed or deleted using the "Select" dialog. Remember to save changes.

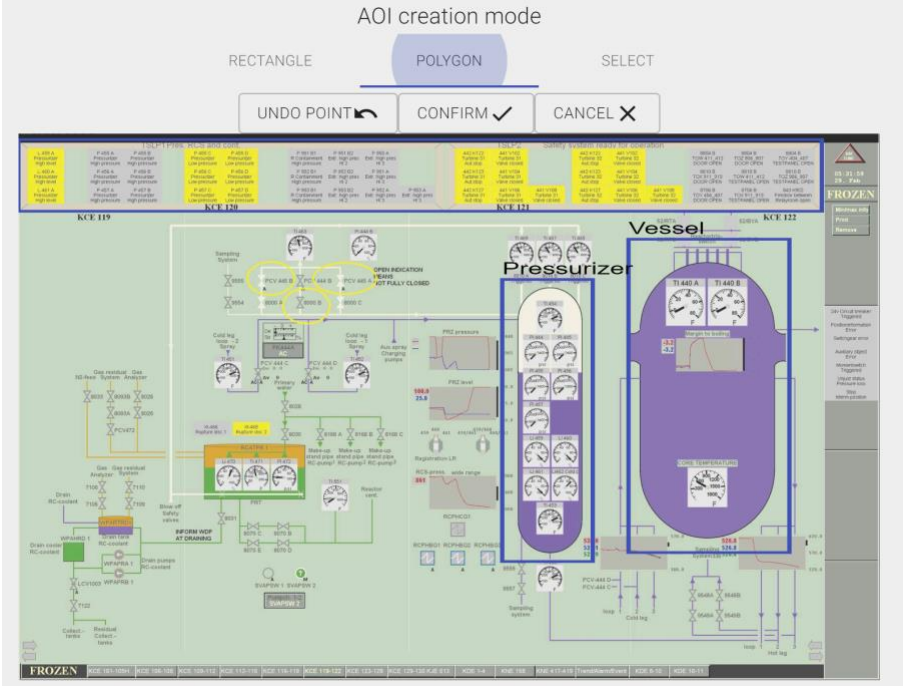

Figure 13. AOI editor.

4.4.2.2 Set Editor

The set editor provides functionality for organizing tasks into sets (see Figure 14).

A set name can be defined in the first line of the editor. Below that, a comma-separated list of tags can be added. If the tag list contains the label “experiment”, the set will be shown on the home screen (section 4.3), and from there can be launched in player mode.

The function “Repeat Set Threshold” (RST) provides support for adaptive training. The system will keep track of how many correct responses a participant made in this set. If the number of correct responses is equal to or larger than the number in the RST field, the experiment run will continue as normal. Otherwise the set is repeated, and the counter of correct responses is reset to zero. Note that a task must have a correct response defined within it, and that only first-order tasks (i.e. direct child-nodes in the set, and not tasks within nested subsets) are considered.

Below the RST function are two check boxes. The first is for randomizing the presentation order of items within the set (by default, items are presented in the order in which they appear in the “Set Tasks” list).

The second checkbox is used to display multiple items on the same page. By default, each task is presented on a separate page. In some cases, it might be desirable to present multiple items on the same page. Examples are demographic questions, questionnaire items, or tasks where an image and a MT question are presented together.

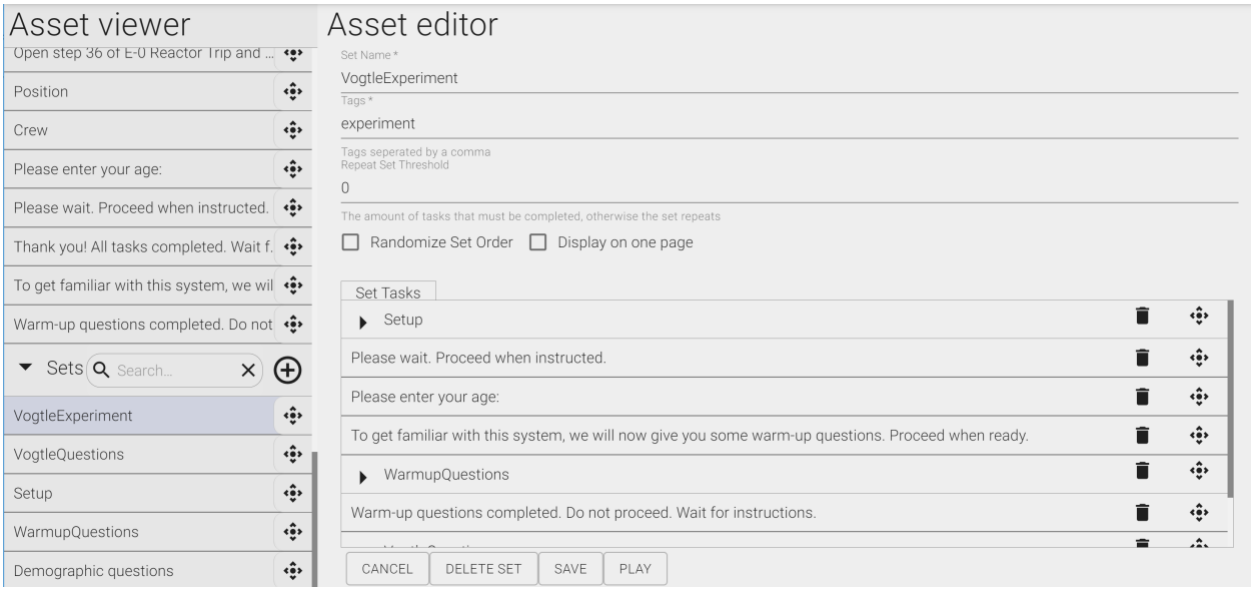

Figure 14. Set editor.

4.4.2.2.1 Task List

The lower half of the set editor contains the task list (see Figure 15). The user can drag either tasks or sets from the asset viewer on the left into the task list. Once in the task list, the order of items can be changed by dragging the item up or down using the drag handle. Items can be deleted using the trashcan icon. Subsets are identified by the triangle icon. Clicking it will show the items in the subset. Note that items in subsets cannot be re-ordered. To re-order items in a subset, open the subset itself by selecting it from the asset viewer on the left.

Note that all changes must be saved using the “Save” button at the bottom of the page. In the same location you will find functions for deleting the set, and to run the set for testing (“Play” button).

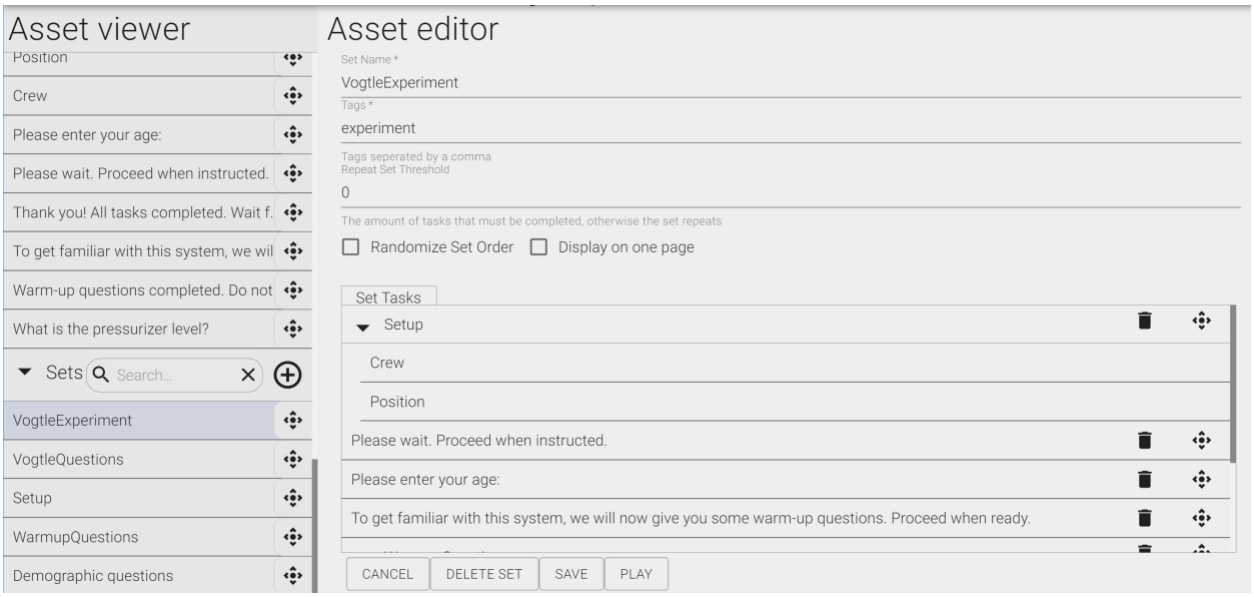

Figure 15. Task list (lower right-hand side of the screen) with subset "Setup" open.

4.5 Player Module

The player module presents tasks to study participants and records their responses. An experiment is started by pressing the start button (triangle) next to the experiment name on the home screen (see Figure 1). It is recommended to run the player in full screen mode (see section 4.3.1).

When tasks are presented during data collection, each page has a "Skip" button in the lower-right corner that participants can click if they do not know the answer or want to skip a question for some other reason. Once the participant provides a response, this button changes from "Skip" to "Next".

Figure 16 shows an example of a page in player mode where two tasks are displayed on the same page. Figure 17 shows an example of a "text entry" task.

Figure 16. Example of a multi-item page in the player module.

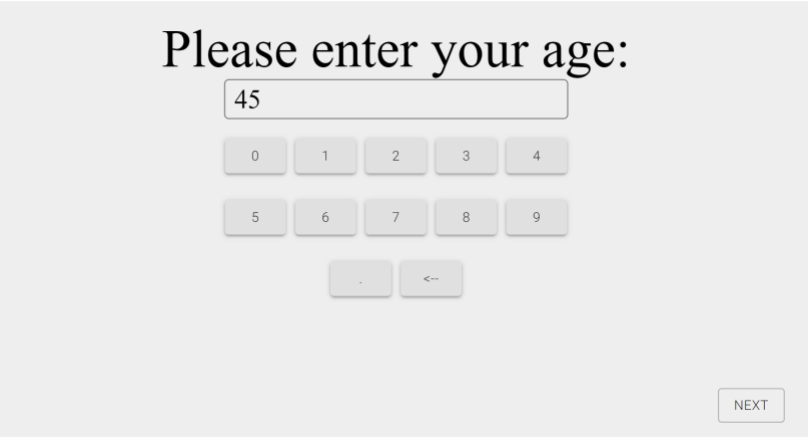

Figure 17. Example of a "text entry" task type with on-screen keyboard.

4.6 Observer Module

The observer module provides progress information for all connected data collection clients (Figure 18). The observer module can be opened on multiple clients, so that multiple researchers can monitor progress. For the observer module to work, it is necessary that the correct Crossbar settings have been entered on all connected clients, i.e. both the data collection clients (player module) as well as the observer clients (see section 4.3.1).

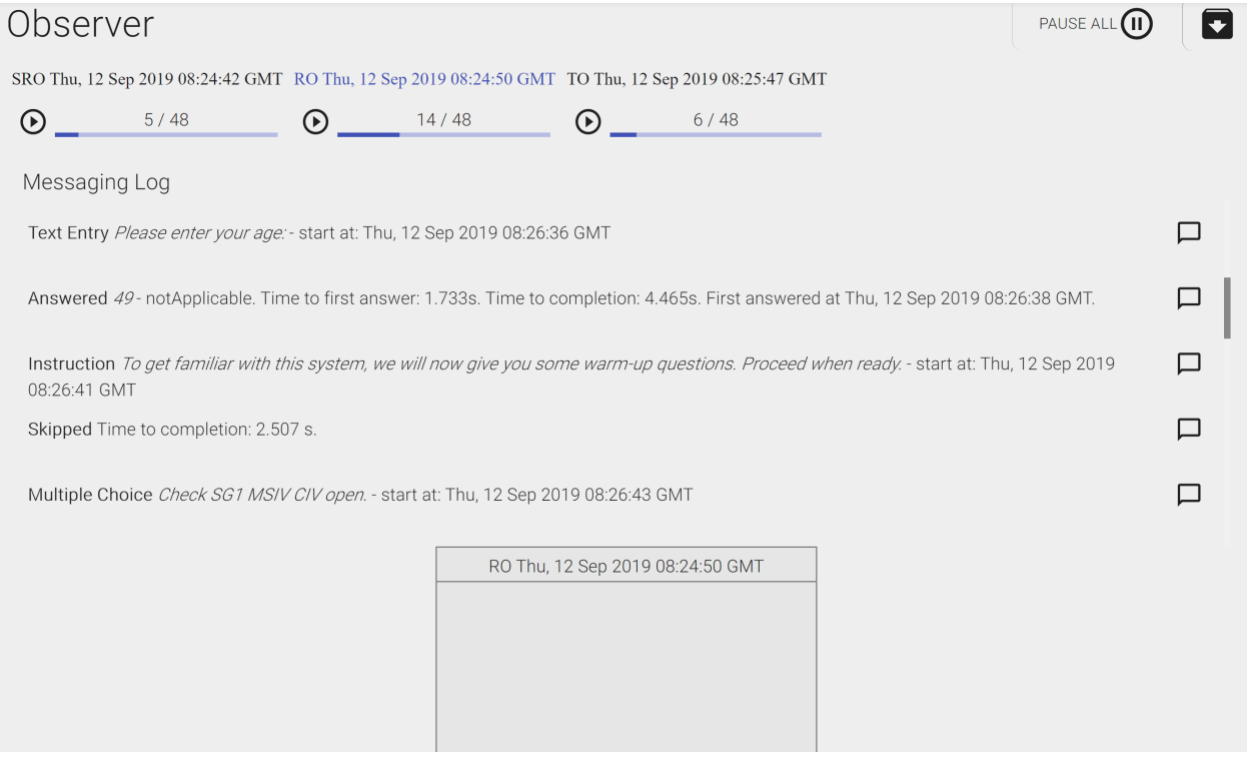

Figure 18. Observer module, showing monitoring of three participants (SRO, RO, TO).

The observer module is empty by default. Once data collection is started, a tab is opened for each data collection client. The tab shows the client ID, as set through the settings dialog (see section 4.3.1), as well as a timestamp for when the data collection started.

Below the participant information is a progress indicator. It shows the number of completed tasks and the number of total tasks both as numbers and as a progress bar.

4.6.1 Pause Function

To the left of the progress bar is a "pause / resume" button that pauses progress on a participant's client (Figure 19). This function can be used in case the participant struggles with the task and needs instructions from the experimenter. In addition to the pause function for each participant, there is a global pause function for all participants via the "Pause all" button in the top right-hand corner of the screen.

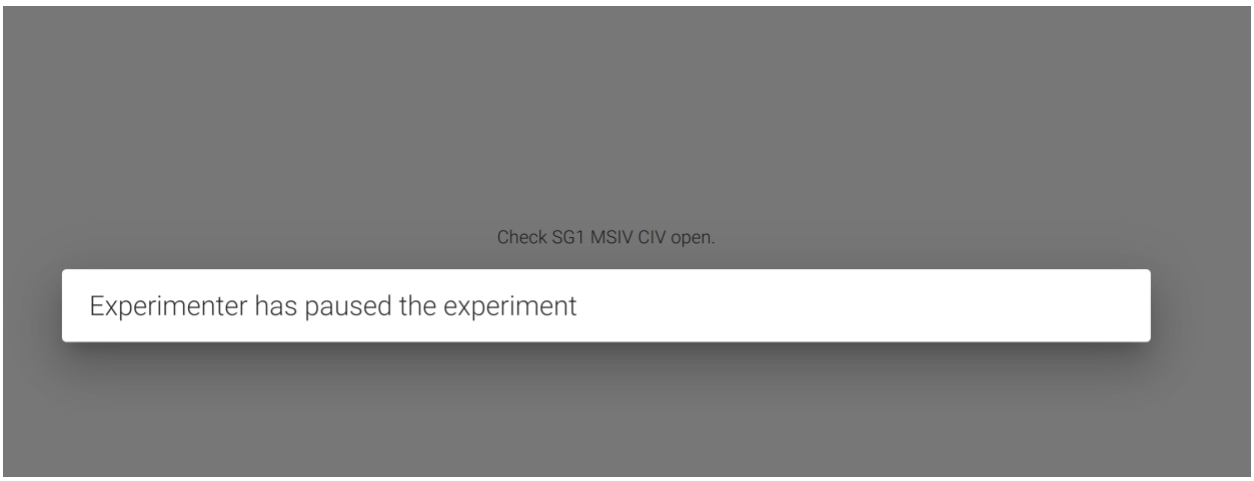

Figure 19. Example of paused experiment on the participant's device.

4.6.2 Observer Log

At the center of the observer screen is the observer log. There are separate logs for each participant, and the observer can change between them using the participant tabs. The log will only show messages starting from when the observer log was opened. In other words, if an observer page is opened halfway through a data collection session, the first half of the message logs will not be displayed (although the data is of course logged). If the observer log is reloaded or refreshed, all previous messages are lost.

For each task, there is one log message created when the task is presented, and another one when the task is completed.

The message at the start of a task shows the task type, the task text, and a time stamp, for example:

```
Multiple Choice Check SG1 MSIV CIV open. - start at: Thu, 12 Sep 2019
08:26:20 GMT
```

The message for when a task is completed starts with "Answered" followed by the answer provided. Next comes the evaluation if the question was answered correctly or not. If no correct answer is defined for the task, the value "notApplicable" is shown.

After this comes two performance times. The first is the time to first response (the first input from the participant), followed by time to completion (the time when the participant clicks "Next").

Finally, there is the timestamp for time of first response.

An example of a log message for a completed task is:

```
Answered Checked - notApplicable. Time to first answer: 0.972s. Time to
completion: 2.372s. First answered at Thu, 12 Sep 2019 08:28:03 GMT
```

4.6.3 Comment Function

Next to each message is a speech bubble icon. Clicking it opens a dialog where the observer can add a message for this task (Figure 20). Messages are stored in the data logs.

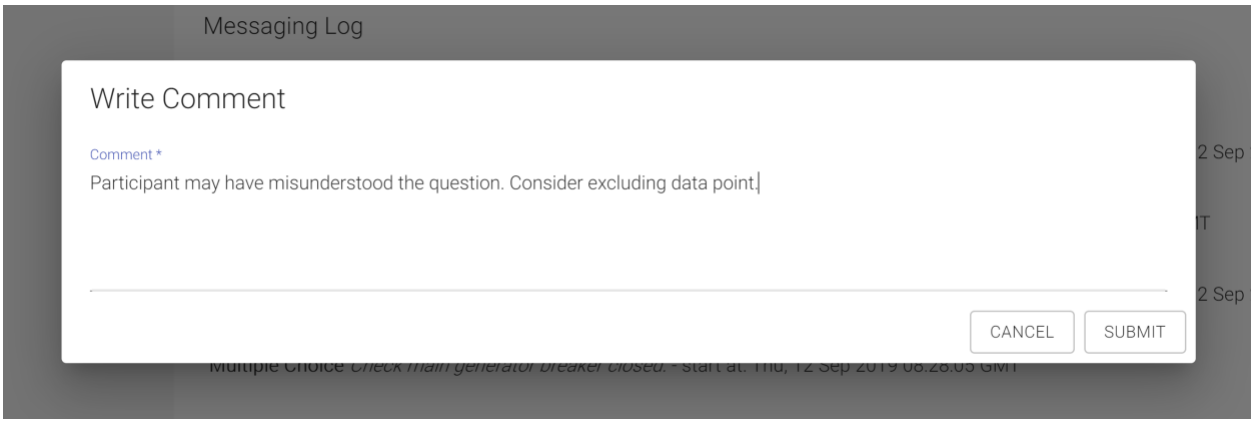

Figure 20. Comment function in the observer logs.

4.6.4 Eye Tracking Live View

At the bottom of the observer screen, there is an eye tracking live view box that will display eye coordinates in the form of moving dots if an eye tracker is connected. This function helps identify any data quality issues with the eye tracking recording in real-time.

In its current version, the live view function does not display the screen content or areas of interest.

4.7 Data Retrieval

Each participant generates a separate log file. The log files are accessible via the download icon on the home screen (see section 4.3). The labels shown in this list are the variables that have been set to "global" (see section 3.3.4), usually crew number and position.

The relevant recordings can be selected and exported (see Figure 21). The data is saved as a comma-separated text file in the folder `WebEntry/backend/exported_data/`.

Figure 21. Data export dialog with three crews selected for export.

4.7.1 Variable Labels

The saved `.csv` data files can be imported into a statistics software. The first line of every file contains the variable labels. The following data is available:

- `Global variables`. A colon-separated list of global variables.
- `Family tree`. Information about the sets and subsets in which this task occurred, e.g. "VogtleExperiment_VogtleQuestions".
- `Task type`. "SingleItem" or "MultiItem", depending on whether the task was presented by itself or together with other items.
- `Task content`. The task text or image name.
- `Start timestamp`. Timestamp when the task was presented.
- `First response timestamp`. Timestamp of first response.
- `Time to first answer`. Time from when the task was presented to first response.
- `Time to completion`. Time from when the task was presented to when the participant clicked "Skip" or "Next".
- `Answer`. The participant's response.
- `Correctly answered`. "Correct", "incorrect", or "notApplicable" (in case no correct answer was defined).
- `Comments`. Any comments that were entered through the observer module (see section 4.6.3).

5 Appendix A: Sample Micro Task Questions

Warm-up questions

1. Check SG1 MSIV CIV open.
2. Check MFW pump C.
3. Open foldout page for E-3 Steam Generator Tube Rupture.

Micro task questions

1. Check SG 1 NR level.
2. Check SG 2 WR level.
3. Check the steam flow from SG 2.
4. Are any Main Steam Line isolation valves open?
5. Check the difference between total steam flow and total feed flow.
6. Check Feed Water isolation valves.
7. Check start-up FW flow
8. Check if start-up FW pumps are running.
9. Check CMT level.
10. Check the pressure of Accumulator Train B.
11. How many CVS MU pumps are running?
12. Is the Passive Core Cooling System line-up correct for this plant condition?
13. Check the total flow from the CVS MU system.
14. Check the IRWST upper NR level.

15. Are any CMT isolation valves mis-positioned?
16. Check SG2 MSIV Bypass CIV closed.
17. Check all Switchgear Busses energized by offsite power.
18. Activate Critical Safety Function Status Trees.
19. Are all main turbine stop valves open?
20. Check main generator breaker closed.
21. Check CMT A narrow range level.
22. Check Reactor Coolant Pumps running.
23. Check RNS pump A.
24. Is the IRWST aligned properly?
25. Check Makeup CIV MOV.
26. Check turbine stop valve.
27. Check at least one Nuclear Island Switchgear Bus energized.
28. Check Trn A Containment Recirc Fan A.
29. Open E-0 Reactor Trip or Safeguards Actuation.
30. Open ES-1.3 ADS Stage 1-3 Actuation Response.
31. Open ES-0.1 Reactor Trip Response.
32. Open foldout page for E-1 Loss of Reactor or Secondary Coolant.
33. Open step 5 of E-2 Faulted Steam Generator Isolation.
34. Open step 18 of ES-1.4 ADS Stage 4 Actuation Response.
35. Open step 36 of E-0 Reactor Trip and Safeguards Actuation

6 References

Brewer, M. (2000). Research Design and Issues of Validity. In Reis, H. and Judd, C. (eds). *Handbook of Research Methods in Social and Personality Psychology*. Cambridge: Cambridge University Press.

Hildebrandt, M., Fernandes, M. (2016). Micro task evaluation of Analog vs. Digital power plant control room interfaces. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, Vol. 60*, pp. 1349-1353.

Hildebrandt, M., Eitrheim, M. (2015). A Micro task Method for Assessing Performance Effects of Innovative Interface Elements. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, Vol. 59*, pp. 1759-1763.

Stanton, N. A., Salmon P. M., Walker G.H., Baber C. & Jenkins D.P. (2013). *Human factors methods: A practical guide for engineering and design*. Ashgate.