

Verification and Validation Report: SFWRENG 4G06 Capstone Design Project

Team #18, InfiniView-AI

Anhao Jiao

Kehao Huang

Qianlin Chen

Qi Shu

Xunzhou Ye

March 6, 2024

1 Revision History

Date	Developer(s)	Change
Mar 6	Qi Shu	Add "Comparison to existing implementations section" and "Changes due to Testing" section.

2 Symbols, Abbreviations and Acronyms

symbol	description
T	Test

[symbols, abbreviations or acronyms – you can reference the SRS tables if needed
—SS]

Contents

1	Revision History	i
2	Symbols, Abbreviations and Acronyms	ii
3	Functional Requirements Evaluation	1
4	Nonfunctional Requirements Evaluation	4
4.1	Usability	4
4.2	Performance	4
4.3	etc.	4
5	Comparison to Existing Implementation	4
6	Unit Testing	5
6.1	User Authentication Module	5
6.2	Instructor View Module	6
6.3	Practitioner View Module	7
6.4	Annotation Configuration Module	8
6.5	RTC Control Module	9
6.6	Other Modules	10
7	Changes Due to Testing	12
7.1	Change due to supervisor's feedback	12
7.2	Change due to reviewing the question in the usability survey	12
8	Automated Testing	12
9	Trace to Requirements	12
10	Trace to Modules	12
11	Code Coverage Metrics	12

List of Tables

List of Figures

This document ...

3 Functional Requirements Evaluation

FR-T1 **Type** Functional, Dynamic, Automated

Initial State Client application is running on the user's device, but the user didn't do any operations yet.

Input/Condition User clicks on the applicable identity(instructor/practitioner) button

Expected Output The live stream video window pops out on the user's screen.

Actual Output The live stream video window shows on the instructor's screen, it doesn't show on the practitioner's screen.

Result Fail

FR-T2 **Type** Functional, Dynamic, Manual

Initial State Application running on user's computer, and the user has clicked on "the instructor identity button" to indicate they are a TaiChi instructor. A window asking for permission to use the camera on the instructor's device popped out.

Input/Condition User allow/deny the webcam permission

Expected Output The webcam on the instructor's device is turned on

Actual Output The webcam on the instructor's device is turned on after allowing webcam permission, and the webcam is not turned on after denying permission.

Result Pass

FR-T3 **Type** Functional, Dynamic, Automatic

Initial State both client applications and the server are running.

Input/Condition The user clicks on the applicable identity button to indicate they are an instructor or a practitioner.

Expected Output A log message indicates connection between the user's device and the server has been established.

Actual Output

Result

FR-T4 **Type** Functional, Dynamic, Automated

Initial State The live stream Window for practitioners.

Input/Condition The user's device has established a connection with the server as a practitioner device.

Expected Output A request from the client device to the server for accessing the list of available annotation configuration.

Actual Output

Result

FR-T5 **Type** Functional, Dynamic, Automated

Initial State The selectable list of the type of annotations is rendered on the user's screen.

Input/Condition Practitioner's selection on the list of types of annotations.

Expected Output A request(that reflects user's annotation selection) from the client device to the server for updating the annotation configuration, with a log indicating the request is sent.

Actual Output

Result

FR-T6 **Type** Functional, Dynamic, Automated

Initial State The system is running and actively connected to practitioners.

Input/Condition Practitioners initiate updates to annotation configurations.

Expected Output The system receives and processes the updated annotation configurations.

Actual Output The server gets the request and user sees the annotation they selected.

Result Pass

FR-T7 **Type** Functional, Dynamic, Automated

Initial State The server is running and actively receiving annotation configuration updates.

Input/Condition In a controlled test environment, the practitioner-client initiates the update of an annotation configuration. The update is sent to the server for processing.

Expected Output The expected result is that the server correctly processes the received annotation configuration from the practitioner-client.

Actual Output The server is not able to receive annotation configuration updates while the connection has been established.

Result Fail

FR-T8 **Type** Functional, Dynamic, Automated

Initial State The server has received and processed the annotation configuration.

Input/Condition The server uses the received annotation configuration to configure machine learning pipelines.

Expected Output The machine learning pipelines are arranged and configured based on the annotation configuration.

Actual Output

Result

FR-T9 **Type** Functional, Dynamic, Automated

Initial State The machine learning pipelines are configured and active.

Input/Condition The instructor's video stream is processed with the annotation configuration.

Expected Output The instructor's video stream is rendered with accurate annotations.

Actual Output

Result

FR-T10 **Type** Functional, Dynamic, Automated

Initial State The server is actively connected to practitioner clients.

Input/Condition The annotated video stream is generated and ready for transmission.

Expected Output The annotated video stream is transmitted to each practitioner-client through their established connections.

Actual Output

Result

FR-T11 **Type** Functional, Dynamic, Automated

Initial State The signaling server is running.

Input/Condition Signaling requests for WebRTC connections are initiated.

Expected Output The signaling server consistently responds to requests and establishes WebRTC connections.

Actual Output

Result

FR-T12 **Type** Functional, Dynamic, Automated

Initial State The client application is running, but the user didn't do any operations yet

Input/Condition The user joining video stream session.

Expected Output A button to identify if a user is an instructor or a practitioner is rendered.

Actual Output

Result

4 Nonfunctional Requirements Evaluation

4.1 Usability

4.2 Performance

4.3 etc.

5 Comparison to Existing Implementation

Our project shares similarities with two existing types of implementations. The first is a project developed by a group of researchers, focused on annotating Tai Chi videos with detailed features such as skeletal outlines and semantic segmentation. Like their summer Tai Chi annotation initiative, our project also annotates videos. However,

ours operates in real-time, unlike the Tai Chi project which processes annotations offline, making ours more adept for live online learning contexts.

Additionally, our project parallels certain aspects of video conferencing tools like Zoom, which offer live annotation capabilities. While Zoom boasts a broader set of features due to its maturity as a platform, our project distinguishes itself in several key areas:

It is designed for one-directional streaming, in contrast to Zoom's multi-directional video conferencing. Unlike Zoom, where annotations are generated client-side, our project renders annotations server-side, minimizing the hardware demands on viewers' devices. Our system allows viewers the flexibility to choose their annotations, diverging from Zoom's model where annotations are unified for all viewers as determined by the video sender. These distinctions underscore our project's unique contributions to video annotation technology, especially in educational and collaborative online environments.

6 Unit Testing

6.1 User Authentication Module

UT-UA1 **Initial State** The system initializes the Auth component within a mock routing environment.

Input/Condition N/A

Expected Output The component should display the Motion Mingle banner and a logo image.

Actual Output The component displayed the Motion Mingle banner and a logo image.

Result Pass

UT-UB2 **Initial State** The system initializes the "Auth" component within a mock routing environment.

Input/Condition The user clicks on the Instructor button.

Expected Output The user should be navigated to the '/instructor' path.

Actual Output The user was navigated to the '/instructor' path.

Result Pass

UT-UC3 **Initial State** The system initializes the "Auth" component within a mock routing environment.

Input/Condition The user clicks on the Practitioner button.

Expected Output The user should be navigated to the '/practitioner' path.

Actual Output The user was navigated to the '/practitioner' path.

Result Pass

UT-UD4 **Initial State** The system initializes the "Auth" component within a mock routing environment.

Input/Condition N/A

Expected Output The slogan should be visible within the component with the correct text.

Actual Output The slogan was displayed within the component with the correct text.

Result Pass

6.2 Instructor View Module

UT-M1 **Initial State** The system is initialized with default state settings. The "Instructor" component is about to be rendered within a mock routing environment.

Input/Condition N/A

Expected Output The "Instructor" component should be rendered with the title "Motion Mingle".

Actual Output The "Instructor" component was rendered with the title "Motion Mingle".

Result Pass

UT-M2 **Initial State** The system has loaded the Instructor component, and the "SelectAnnotation" component is ready to receive user input.

Input/Condition A user selects "Skeleton" annotation from the dropdown menu.

Expected Output The dropdown menu should reflect that "Annotation 1" has been selected.

Actual Output "Skeleton" has been selected from the dropdown menu.

Result Pass

UT-M3 **Initial State** The Instructor component is rendered and operational, presumed to be in a state where broadcasting is occurring (mocked condition).

Input/Condition The user clicks the "Broadcast" button to simulate starting a broadcast and then clicks "Stop" to simulate ending the broadcast.

Expected Output The "MessageModal" mock should be displayed, indicating that the stopping of the video is to be confirmed.

Actual Output The "MessageModal" mock was displayed.

Result Pass

UT-M4 **Initial State** The Instructor component is rendered and operational. The self-video is initially off.

Input/Condition The user clicks "Start self video" to initiate the video stream.

Expected Output The system should transition to a state where the self-video is active.

Actual Output The system transitioned to a state where the self-video is active.

Result Pass

UT-M5 **Initial State** The Instructor component is rendered and operational. The self-video is active.

Input/Condition The user clicks "Close self video".

Expected Output The system should transition to a state where the self-video is inactive.

Actual Output The system transitioned to a state where the self-video is inactive.

Result Pass

6.3 Practitioner View Module

UT-PV1 **Initial State** The system is initialized with default state settings. The "Practitioner" component is about to be rendered within a mock routing environment.

Input/Condition N/A

Expected Output The "Practitioner" component should be rendered with the title "Motion Mingle", a dropdown for selecting annotations initialized to "None", and a "Connect" button.

Actual Output The "Practitioner" component was rendered with the title "Motion Mingle".

Result Pass

UT-PV2 **Initial State** The "Practitioner" component is rendered and displayed with the default selected annotation as "None".

Input/Condition The user changes the annotation selection from "None" to "Skeleton" using the dropdown.

Expected Output The dropdown menu for selecting annotations should reflect the change by displaying "Skeleton" as the current selection.

Actual Output "Skeleton" was displayed as selected in the dropdown menu.

Result Pass

UT-PV3 **Initial State** The "Practitioner" component is rendered, displaying the "Connect" button while the video stream is not connected.

Input/Condition The "Connect" button was clicked to start the video stream.

Expected Output The "Connect" button should change to "Disconnect".

Actual Output The "Connect" button changed to "Disconnect".

Result Pass

UT-PV4 **Initial State** The "Practitioner" component is rendered, displaying the "Disconnect" button while the video stream is connected.

Input/Condition The "Disconnect" button was clicked to stop the video stream.

Expected Output The "Disconnect" button should change to "Connect".

Actual Output The "Disconnect" button changed to "Connect".

Result Pass

6.4 Annotation Configuration Module

UT-AC1 **Initial State** The "SelectAnnotation" component is initialized with default state settings.

Input/Condition The component is rendered without any preselected value.

Expected Output The component should display with the label "Annotation" and a dropdown button showing the default value "None".

Actual Output The component was rendered with the label "Annotation" and a dropdown button showing the default value "None".

Result Pass

UT-BC2 **Initial State** The "SelectAnnotation" component is rendered with default settings.

Input/Condition The user clicks on the dropdown menu to view available options.

Expected Output The dropdown menu should display four options: "None", "Skeleton", "Edges", and "Cartoon".

Actual Output The dropdown menu displayed four options: "None", "Skeleton", "Edges", and "Cartoon".

Result Pass

6.5 RTC Control Module

UT-RC1 **Initial State** Before creating a new "RTCPeerConnection", no connection exists.

Input/Condition The "createPeerConnection" function is called without any specific arguments since it uses predefined settings within its implementation.

Expected Output A new "RTCPeerConnection" should be created with the configuration specified for unified-plan SDP semantics and the Google STUN server.

Actual Output A new "RTCPeerConnection" was created with the configuration specified for unified-plan SDP semantics and the Google STUN server.

Result Pass

UT-RC2 **Initial State** A "RTCPeerConnection" instance exists but has not yet established a connection nor created any offers.

Input/Condition The function "connectAsConsumer" is called with the created peer connection and "annotation" as arguments.

Expected Output The function should asynchronously create an SDP offer, set it as the local description, send it to the server, and set the received SDP answer as the remote description.

Actual Output Same as expected output.

Result Pass

UT-RC3 **Initial State** A "RTCPeerConnection" instance is ready but not yet in a broadcasting state.

Input/Condition The function "connectAsBroadcaster" is called with the peer connection object.

Expected Output The function should asynchronously create an SDP offer for broadcasting, set it as the local description, send it to the server, and set the server's SDP answer as the remote description.

Actual Output Same as expected.

Result Pass

6.6 Other Modules

UT-OT1 **Initial State** The system initializes the "NotFound" component within a mock routing environment.

Input/Condition The component is rendered without any user interaction.

Expected Output The component should display the "Not Found" message.

Actual Output The component displayed the "Not Found" message.

Result Pass

UT-OT2 **Initial State** The system initializes the "NotFound" component within a mock routing environment.

Input/Condition The component is rendered, and the existence of a link is checked.

Expected Output There should be a link that reads "GO HOME" and navigates to the root path '/' upon clicking.

Actual Output Same as expected.

Result Pass

UT-OT3 **Initial State** The "MessageModal" component is rendered with "isModalOpen" set to true.

Input/Condition The "Cancel" button was clicked.

Expected Output The "handleClose" mock function should be called once.

Actual Output The "handleClose" mock function was called once.

Result Pass

UT-OT4 **Initial State** The "MessageModal" component is rendered with "isModalOpen" set to true.

Input/Condition The "Stop Video" button was clicked.

Expected Output The "handelStopVideo" mock function should be called once.

Actual Output The "handelStopVideo" mock function was called once.

Result Pass

UT-OT5 **Initial State** The "MessageModal" component is initialized and rendered with "isModalOpen" set to true.

Input/Condition N/A

Expected Output The modal, including the warning message, should be visible.

Actual Output The modal, including the warning message, was visible.

Result Pass

UT-OT6 **Initial State** The "MessageModal" component is initialized and rendered with "isModalOpen" set to false.

Input/Condition N/A

Expected Output None of the modal's content should be present in the document.

Actual Output None of the modal's content was present in the document.

Result Pass

7 Changes Due to Testing

7.1 Change due to supervisor’s feedback

Andrew Mitchell, who is one of the supervisors of this project, has pointed out during the Revision 0 demo that the “check annotated video” button on the instructor page should appear after the instructor has started broadcasting instead of showing it directly, as clicking the check annotated video can’t show anything without input video stream. To follow the feedback from Andrew, the UI design will be changed so that the “check annotated video” button only shows up after the instructor starts broadcasting, and extra instructions will be added to the side of the instructor and practitioner view page.

7.2 Change due to reviewing the question in the usability survey

The VnV Plan, states that our group will review the questions in the usability survey as part of non-functional testing. One of the questions in the usability survey is: “Are there any specific content or learning materials you would like to see added to the application?” Some users state it is not convenient that they need to manually disconnect and reconnect to the server when selecting different annotations as a practitioner. The team decided to add an “auto-refresh” functionality to the system when an instructor or a practitioner changes their selected annotation would make the UI more intuitive and enhance user experiences.

8 Automated Testing

9 Trace to Requirements

10 Trace to Modules

11 Code Coverage Metrics

References

Appendix — Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Reflection. Please answer the following question:

1. In what ways was the Verification and Validation (VnV) Plan different from the activities that were actually conducted for VnV? If there were differences, what changes required the modification in the plan? Why did these changes occur? Would you be able to anticipate these changes in future projects? If there weren't any differences, how was your team able to clearly predict a feasible amount of effort and the right tasks needed to build the evidence that demonstrates the required quality? (It is expected that most teams will have had to deviate from their original VnV Plan.)