Verification and Validation Report: SFWRENG 4G06 Capstone Design Project

Team #18, InfiniView-AI
Anhao Jiao
Kehao Huang
Qianlin Chen
Qi Shu
Xunzhou Ye

March 5, 2024

1 Revision History

Date	Version	Notes
Date 1	1.0	Notes
Date 2	1.1	Notes

2 Symbols, Abbreviations and Acronyms

symbol	description
T	Test

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

Contents

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

This section will not be appropriate for every project.

6 Unit Testing

7 Changes Due to Testing

[This section should highlight how feedback from the users and from the supervisor (when one exists) shaped the final product. In particular the feedback from the Rev 0 demo to the supervisor (or to potential users) should be highlighted. —SS]

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.)