Testing the IMS system

The IMS system consists of several components:

- The management console is a web interface used by experiment operators.
- The watch app which runs the set up experiment sessions.
- The infrastructure that contains communication and persistence.

Functional requirements:

1. Testing the watch user interaction and feedback

Test cases:

1. Input handling

1.1 detect swipes:

Steps:

- Open watch app
- Swipe on numbers to choose id

Expected result: the numbers change upon swipe

1.2 detect taps:

Steps:

- Open watch app
- Choose id
- Tap on the confirmation button

Expected result: id should be selected, and the user should be redirected to the lobby.

1.3: detect managers input commands

Steps:

Participant on the watch:

- Open watch app

- Choose id
- Tap on the confirmation button
- Wait on lobby

Operator on the web interface:

- Create a lobby with the chosen participant id

Expected result: the lobby on the watch should now show the assigned lobby id.

2. Feedback Mechanisms

Session setup steps:

Participant on the watch:

- Open watch app
- Choose id
- Tap on the confirmation button
- Wait on lobby

Operator on the web interface:

- Create a lobby with the chosen participant id
- Set up the experiment sessions

Participant on the watch:

- Tap ready

Operator on the web interface:

- Launch experiment

Expected result: a session should run on the participant's watch.

2.1 visual feedback:

Participant on the watch during sessions:

2.1.1 water ripples session:

Steps:

- Tap on the middle button

Expected result: a ripple effect of a growing moving circle should show.

2.1.2 wine glass session:

Steps:

- Swipe in a cyclic manner around the edge of the screen

Expected result: an orange block should show under the finger's location.

2.1.3 flour mill session:

Steps:

- Swipe in a cyclic manner around the wheel

Expected result: an orange block should show under the finger's location.

2.1.4 flower garden session:

Steps:

- Tap on the middle button

Expected result: 5 water droplets/plants should show up around the button.

2.2 haptic feedback:

Participant on the watch during sessions:

2.2.1 water ripples session:

Steps:

- Tap on the middle button until sync occurs

Expected result: during synchronization, a vibration should occur.

2.2.2 flower garden session:

Steps:

- Tap on the middle button until sync occurs

Expected result: during synchronization, a vibration should occur.

2.3 audio feedback:

Participant on the watch during sessions:

2.3.1 water ripples session:

Steps:

- Tap on the middle button until sync occurs

Expected result: when synchronization occurs, a short sound should be played.

2.3.2 wine glass session:

Steps:

- Swipe in a cyclic manner around the edge of the screen until sync occurs

Expected result: during synchronization, a continuous harmonious sound should be played.

2.1.3 flour mill session:

Steps:

- Swipe in a cyclic manner around the wheel until sync occurs

Expected result: during synchronization, a grinding sound should be played.

2.1.4 flower garden session:

Steps:

- Tap on the middle button until sync occurs

Expected result: when synchronization occurs, a short bell sound should be played.

3. User polling

3.1 Feedback polling between sessions:

Steps:

- Participant is part of a set up experiment
- The participant finishes a session

Expected result: before the next session, a scale survey of 2 feedback questions should appear with the option to submit.

3.2 Redirecting to feedback polling at the end of the experiment:

Steps:

- Participant is part of a set up experiment
- The participant finishes all the sessions in the experiment

Expected result: a barcode for a more extensive feedback questionnaire should appear on the screen.

2. Communication and Synchronization

1.1 Real-Time Communication

Test Case 1.1.1: Verify the watch establishes a connection to the game server.

Steps:

1. Start watch client app.
2. Observe the client attempting to connect to the game server.
   Expected Result:
   Connection is successfully established, and the client is moved to the waiting to join a lobby screen.

Test Case 1.1.2: Verify the web interface can control the game session.

Steps:

1. Log into the web interface as an operator or admin.
2. Create a lobby.
3. Add sessions to the lobby.
4. Remove half of the sessions.
   Expected Result:
   The lobby screen shows exactly which sessions were added minus the ones that were removed and no error was presented to the user

1.2 Synchronization Detection

Test Case 1.2.1: Verify synchronization is detected based on timing and position.

Steps:

1. Begin a session.
2. Have two participants perform a coordinated action (e.g., tapping the screen at the same beat).
   Expected Result:
   The system accurately detects when the timing and positions align and shows the sync feedback

Test Case 1.2.2: Verify appropriate feedback is triggered.

Steps:

1. Simulate two participants performing synchronized and then unsynchronized actions.
2. Observe the system's feedback.
   Expected Result:
   Correct visual/audio feedback is given for synchronized actions; no synchronization feedback is give for unsynchronized actions.

2. Data Collection and Storage

2.1 Sensor Data Acquisition

Test Case 2.1.1: Verify interaction data is correctly captured.

Steps:

1. Perform tap and swipe gestures during a session
2. Check the logged interaction data on the manager session data menu.
   Expected Result:
   The system correctly records timings and positions of interactions

Test Case 2.1.2: Verify physiological data is correctly captured.

Steps:

1. Start a session while wearing the watch.
2. Check the logged interaction data on the manager session data menu.
   Expected Result:
   The system correctly records physiological metrics.

2.2 Data Storage and Analysis

Test Case 2.2.1: Verify data is sent to the remote collection server.

Steps:

1. Complete a session.
2. Check the remote data server logs for session data.
   Expected Result:
   Interaction and physiological data successfully transmitted to the server.

Test Case 2.2.2: Verify session details are correctly stored.

Steps:

1. Complete a session.
2. Query the data storage to retrieve session details (e.g., duration, sync attempts).
   Expected Result:
   Stored session data is accurate and matches the actual game activity.

3. Connectivity and Stability

Test Case 3.1.1: Verify smartwatch login with user ID.

Steps:

1. Log into the app with a specific user ID.
2. Look at the manager console and see which participants are connected
   Expected Result:
   The watch is linked to the correct participant and the participant shows up in the manager console

3.2 Connection Management

Test Case 3.2.1: Verify auto-reconnection after disconnection.

Steps:

1. Intentionally disconnect the device from the network.
2. Wait for the system to attempt to reconnect automatically.
   Expected Result:
   Connection is successfully re-established without manual intervention.

3. Testing the User Interface

The user interface (UI) is a critical component of the IMS Manager application, driving all experiment-control workflows. Below is a detailed manual testing plan and results for each screen and feature.

UI Testing Process

- Method: Manual exploratory testing in Chrome (latest) by clicking through each view and performing all supported user actions.
- Scope: Authentication, main-menu navigation, lobby management, participant/operator CRUD, session listings and filtering, and detailed per-session data visualizations.

Test Cases

5.1 Login Functionality

- Steps:
- Navigate to /login.
- Enter valid admin credentials and click Login → Participants and Operators screens are accessible.
- Enter valid non-admin credentials and click Login → Participants and Operators screens are not accessible.
- Enter invalid credentials and click Login.

Expected Results:

Admin credentials redirect to Main Menu at /main_menu with access to Participants and Operators screens.

Non-admin credentials redirect to Main Menu at /main_menu without access to Participants and Operators screens.

Invalid credentials keep the user on the login page.

5.2 Main Menu Navigation

- Steps:
- From any page, click the home icon → returns to Main Menu.
- Click each blue button (Lobbies, Participants, Operators, Session Data) → navigates to that screen.
- Click Disconnect (red) → returns to /login and clears the session token.

5.3 Lobby Management

5.3.1 View Active Lobbies

- Steps:
- Click Lobbies → table of active lobbies (ID and players) appears.
- Single-click a row → highlights it and sets selected_lobby_id.
- Double-click a row → opens /lobby?lobby_id=....

5.3.2 Create & Remove Lobby

- Steps:
- Click Create Lobby → modal opens listing participants with checkboxes.
- Select two participants and click Create Lobby → new lobby appears in the table.
- Select a lobby row and click Remove Lobby → lobby is removed.

5.3.3 Experiment Control

Scenario A – Start / Stop Experiment

- Open lobby detail view; confirm Start Experiment is disabled until both watches are Ready.
- When both are Ready, click Start Experiment → button text changes to Stop Experiment.
- Click Stop Experiment → button reverts to Start Experiment; lobby resets to idle.

Scenario B – Add Session

- Click Add Session → modal with Game Mode, Duration, etc., appears.
- Select, for example, Water Ripples, set parameters, and click Create Session.
- New session row appears in the table.

Scenario C – Remove Session

- Select a session row, click Delete Session, confirm.

- Row disappears from the table.

Scenario D – Re-order Sessions

- Click anywhere within a table row to start dragging.
- Drag the row to the desired location within the table.
- Release the row, and the table updates the order column automatically.

Edge-Case Checks:

- There must always be two participants to start a session, and the Create Session button remains disabled if both participants are not marked as Ready.
- Create Session button stays disabled until all required fields are completed (invalid fields highlighted red).

5.4 Participant Management

- Steps:
- Click Participants → table of participants (ID, name, age, etc.) appears.
- Click Add Participant, complete form, submit → new participant listed.
- Select a row, click Remove Participant → participant deleted.

Permissions: Only admin users can access Participants.

5.5 Operator Management

- Steps:
- Click Operators → operator list appears.
- Add Operator / Remove Operator works like Participants.

Permissions: Only admin users can access Operators.

5.6 Session Data (All Experiments)

- Steps:
- Click Session Data → table of past experiments (Exp ID, participants, timestamp) appears.
- Click any row → opens /session_data/single?session_id=….

5.7 Sessions within a Lobby Data

- Steps:
- Scroll to Sessions table → all sessions for that lobby are listed (game type, duration, etc.).
- Use Game Type dropdown → table filters instantly.
- Click a session row → opens single-session view.

5.8 Single-Session Data Visualizations

Charts

- For click-based games (e.g., Water Ripples): Clicks & Sync Markers will appear as the primary graph type.
- For circular games (e.g., Flower Mill, Wine Glasses): Finger Frequency, Sync Intervals, and Finger Angle graphs will render.
- Common across all games: Heart Rate, HR Variation, Latency, and Jitter charts.
- Steps:
- Load /session_data/single?... → all relevant charts render without console errors.
- Hover or tap any data point → tooltip displays correct timestamp and value.

Non-Functional requirements:

1. Speed and Performance

- **Requirement:** The latency time for data transfer between 2 users will be minimal and less than 20ms, so the synchronization be optimal.
  - **Test Case:** Make sure that the average latency in the experiment does not cross the threshold, on different Wi-Fi connections.
    - **Steps:**
      1. Running various experiments on connections to the university's Internet network, to other networks, and in different locations.
      2. Check the latency in the management console during the experiment.
    - **Expected Result:** The average latency in the experiment will be below 20ms.

- **Requirement:** The system will meet the capacity of 4 users at the same time.
  - **Test Case:** Make sure that the average latency in the experiment does not cross the threshold of 20ms, when 2 pairs play simultaneously.
    - **Steps:**
      1. Run 2 experiments simultaneously on 4 people connected to the same network/different networks.
      2. Check the latency in the management console during the experiment.
    - **Expected Result:** The average latency in both experiments will be below 20ms.

- **Requirement:** In the case of a short communication failure, the system will continue to operate with an error-corrector, but in the case of a long failure, the session will stop.
  - **Test Case 1:** Running an experiment with controlled short disconnections for testing.
    - **Steps:**
      1. Running an experiment when one of the subjects is near an area with no Wi-Fi reception, and walks there frequently and briefly.
      2. The second subject should not experience any disconnections at all, and should feel continuous movement of the partner at all times.
    - **Expected Result:** The session must go on and not stop.

  - **Test Case 2:** Running an experiment with controlled long disconnections for testing.
    - **Steps:**
      1. Running an experiment when one of the subjects was near an area with no Wi-Fi reception.
      2. The subject needs to walk to that area during the session without coming back.
    - **Expected Result:** The session should be terminated with error message.

- **Requirement:** The watches are supposed to reliably record physiological data.
  - **Test Case:** Make sure that the heart rate changes according to changes in activity.
    - **Steps:**
      1. Running an experiment where one of the subjects changes states throughout the experiment between: resting, walking, running.
      2. At the end of the experiment, check the heart rate graph of that subject.
    - **Expected Result:** The subject's heart rate had to change depending on the order of the situations he was in during the experiment.

2. Safety

- **Requirement:** The data collected will be stored in an encrypted database with limited access.
  - **Test Case:** Verify data encryption and access control.
    - **Steps:**
      1. Access the database and inspect data format.
      2. Attempt access from unauthorized user.
    - **Expected Result:** Data is encrypted, access denied for unauthorized users.

3. Security

- **Requirement:** Only the researchers and those on their behalf will be able to access the data collected on the subjects.
  - **Test Case 1:** Attempt to access internal links.
    - **Steps:**
      1. From a browser where you previously logged in to the system, we will try to access internal links in the management console.
    - **Expected Result:** The site will send the user to the login window.

  - **Test Case 2:** Verify encryption and secure storage of sensitive data.
    - **Steps:**
      1. Verify encryption standards for data at rest and in transit.
    - **Expected Result:** All sensitive data is encrypted and securely stored.

4. Usability

- **Requirement:** A user will be able to operate the application without prior experience or training.
  - **Test Case 1:** Usability test with new users.
    - **Steps:**
      1. Provide app to users unfamiliar with the system.
      2. Show them the installations video.
      3. Observe their usage.
    - **Expected Result:** Users play the session-modes successfully, without human guidance.