**myJym Software Requirements Specification**

**Date: 11/20/21**

| **Technical Writers:**  Hsu, Wei | |  | |  | | |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ales, Grant | |  | | |
| Hoyt, Devan |  | |  | | |

**Table of Contents**

* [1.1 Purpose](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.1-purpose)
* [1.2 Scope](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.2-scope)
  + [1.2.1 Products to be Produced](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.2.1-products-to-be-produced)
  + [1.2.2 What Will the Software Product Do](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.2.2-what-will-the-software-product-do)
  + [1.2.3 Included Core Features](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.2.3-included-core-features)
  + [1.2.4 Possible Features](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.2.4-excluded-features)
* [1.3](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3)
  + [1.3.1 Product Perspective](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.1-product-perspective)
    - [1.3.1.1 System Interfaces](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.1.1-system-interfaces)
    - [1.3.1.2 User Interfaces](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.1.2-user-interfaces)
    - [1.3.1.3 Hardware Interfaces](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.1.3-hardware-interfaces)
    - [1.3.1.4 Software Interfaces](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.1.4-software-interfaces)
    - [1.3.2 Product Functions](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.2-product-functions)
      * [1.3.2.1 User Profiles](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.2.1-user-profiles)
      * [1.3.2.2 User Scenarios](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.2.2-user-scenarios)
  + [1.3.3 User Characteristics](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.3-user-characteristics)
  + [1.3.4 Limitations](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4-limitations)
    - [1.3.4.1 Regulatory Requirements and Policies](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.1-regulatory-requirements-and-policies)
    - [1.3.4.2 Hardware Limitations](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.2-hardware-limitations)
    - [1.3.4.3 Interfaces to Other Applications](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.3-interfaces-to-other-applications)
    - [1.3.4.4 Parallel Operation](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.4-parallel-operation)
    - [1.3.4.5 Audit Functions](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.5-audit-functions)
    - [1.3.4.6 Control Functions](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.6-control-functions)
    - [1.3.4.7 Higher-order Language Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.7-higher-order-language-requirements)
    - [1.3.4.8 Signal Handshake Protocols](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.8-signal-handshake-protocols)
    - [1.3.4.9 Quality Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.9-quality-requirements)
    - [1.3.4.10 Criticality of the Application](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.10-criticality-of-the-application)
    - [1.3.4.11 Safety and Security Considerations](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.11-safety-and-security-considerations)
    - [1.3.4.12 Physical/Mental Considerations](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.12-physical/mental-considerations)
    - [1.3.4.13 External System Limitations](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.3.4.13-external-system-limitations)
* [1.4 Definitions](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#1.4-definitions)
* [3.1 My Trainer](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1-my-trainer)
  + [3.1.1 External Interfaces](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.1-external-interfaces)
    - [3.1.1.1 Google Authentication](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.1.1-google-authentication)
  + [3.1.2 Functions](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.2-functions)
    - [3.1.2.1 myJym](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.2.1-myJym)
    - [3.1.2.2 Notifications](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.2.2-notifications)
    - [3.1.2.3 My Trainer](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.2.3-my-trainer)
    - [3.1.2.4 View Trainer Profile](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.2.4-view-trainer-profile)
    - [3.1.2.5 Edit Profile](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.2.5-edit-profile)
    - [3.1.2.6 Search Trainer](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.2.6-search-trainer)
    - [3.1.2.7 Message Trainer](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.2.7-message-trainer)
    - [3.1.2.8 Message Users](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.2.8-message-users)
  + [3.1.3 Usability Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.3-usability-requirements)
  + [3.1.4 Performance Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.4-performance-requirements)
  + [3.1.5 Logical Database Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.5-logical-database-requirements)
    - [3.1.5.1 Types of information used by various functions](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.5.1-types-of-information-used-by-various-functions)
    - [3.1.5.2 Frequency of use](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.5.2-frequency-of-use)
    - [3.1.5.3 Accessing Capabilities](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.5.3-accessing-capabilities)
    - [3.1.5.4 Data entities and their relationships](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.5.4-data-entities-and-their-relationships)
    - [3.1.5.5 Integrity Constraints](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.5.5-integrity-constraints)
    - [3.1.5.6 Security](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.5.6-security)
    - [3.1.5.7 Data retention requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.5.7-data-retention-requirements)
  + [3.1.6 Design Constraints](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.6-design-constraints)
  + [3.1.7 Software System Attributes](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.7-software-system-attributes)
    - [3.1.7.1 Reliability](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.7.1-reliability)
    - [3.1.7.2 Security](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.7.2-security)
    - [3.1.7.3 Data Privacy](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.7.3-data-privacy)
    - [3.1.7.4 Maintainability](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.7.4-maintainability)
    - [3.1.7.5 Portability](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.7.5-portability)
  + [3.1.8 Supporting Information](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.1.8-supporting-information)
* [3.2 Gym Map](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2-gym-map)
  + [3.2.1 External Interfaces](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.1-external-interfaces)
  + [3.2.2 Functions](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.2-functions)
    - [3.2.2.1 Map Display](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.2.1-map-display)
    - [3.2.2.2 Map Search Function](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.2.2-map-search-function)
    - [3.2.2.3 Map Navigation](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.2.3-map-navigation)
  + [3.2.3 Usability Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.3-usability-requirements)
  + [3.2.4 Performance Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.4-performance-requirements)
  + [3.2.5 Logical Database Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.5-logical-database-requirements)
    - [3.2.5.1 Types of information used by various functions](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.5.1-types-of-information-used-by-various-functions)
    - [3.2.5.2 Frequency of use](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.5.2-frequency-of-use)
    - [3.2.5.3 Accessing Capabilities](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.5.3-accessing-capabilities)
    - [3.2.5.4 Data entities and their relationships](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.5.4-data-entities-and-their-relationships)
    - [3.2.5.5 Integrity Constraints](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.5.5-integrity-constraints)
    - [3.2.5.6 Security](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.5.6-security)
  + [3.2.6 Design Constraints](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.6-design-constraints)
  + [3.2.7 Software System Attributes](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.7-software-system-attributes)
    - [3.2.7.1 Reliability](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.7.1-reliability)
    - [3.2.7.2 Security](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.2.7.2-security)
* [3.3 MyJym Videos](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3-myjym-videos)
  + [3.3.1 External Interfaces Specification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.1-external-interfaces-specification)
    - [3.3.1.1 Database](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.1.1-database)
    - [3.3.1.2 Video Streaming](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.1.2-video-streaming)
  + [3.3.2 Functions](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.2-functions)
    - [3.3.2.1 My Video](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.2.1-my-video)
  + [3.3.3 Usability Requirement](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.3-usability-requirement)
  + [3.3.4 Performance requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.4-performance-requirements)
  + [3.3.5 Logical Database Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.5-logical-database-requirements)
  + [3.3.6 Design Constraints](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.6-design-constraints)
  + [3.3.7 Software System Attributes](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.7-software-system-attributes)
    - [3.3.7.1 Reliability](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.7.1-reliability)
    - [3.3.7.2 Security](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.7.2-security)
    - [3.3.7.3 Maintainability](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.7.3-maintainability)
  + [3.3.8 Supporting Information](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.3.8-supporting-information)
* [3.4 Login & Homepage](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4-login-&-homepage)
  + [3.4.1 External Interfaces](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4.1-external-interfaces)
  + [3.4.2 Functions](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4.2-functions)
  + [3.4.3 Usability Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4.3-usability-requirements)
  + [3.4.4 Performance Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4.4-performance-requirements)
  + [3.4.5 Logical Database Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4.5-logical-database-requirements)
  + [3.4.6 Design Constraints](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4.6-design-constraints)
  + [3.4.7 Software System Attributes](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4.7-software-system-attributes)
    - [3.4.7.1 Reliability](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4.7.1-reliability)
    - [3.4.7.2 Availability](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4.7.2-availability)
    - [3.4.7.3 Security](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4.7.3-security)
    - [3.4.7.4 Portability](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.4.7.4-portability)
* [3.5 My Workout](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5-my-workout)
  + [3.5.1 External Interface](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.1-external-interface)
    - [3.5.1.1 Database](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.1.1-database)
    - [3.5.1.2 video provider](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.1.2-video-provider)
  + [3.5.2 Functions](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.2-functions)
    - [3.5.2.1 Create Workout](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.2.1-create-workout)
    - [3.5.2.2 Find Workout](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.2.2-find-workout)
    - [3.5.2.3 Saved Workout](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.2.3-saved-workout)
    - [3.5.2.4 Workout in Progress Page](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.2.4-workout-in-progress-page)
  + [3.5.3 Usability Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.3-usability-requirements)
  + [3.5.4 Performance Requiremnts](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.4-performance-requirements)
  + [3.5.5 Logical Database Requirements](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.5-logical-database-requirements)
  + [3.5.6 Design Constraints](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.6-design-constraints)
  + [3.5.7 Software System Attributes](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.7-software-system-attributes)
    - [3.5.7.1 Reliability](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.7.1-reliability)
    - [3.5.7.2 Security](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.7.2-security)
    - [3.5.7.3 Maintainability](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#3.5.7.3-maintainability)
* [4.0 Validation](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.0-validation)
  + [4.1.1 External Interface Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.1.1-external-interface-verification)
  + [4.2.1 Map External Interfaces Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.2.1)
  + [4.2.2 Map Functions Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.2.2)
  + [4.2.3 Map Usability Requirements Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.2.3)
  + [4.2.4 Map Performance Requirements Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.2.4)
  + [4.2.5 Map Logical Database Requirements Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.2.5)
  + [4.5.1 External Interface Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.1-external-interface-verification)
    - [4.5.1.1 Database Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.1.1-database-verification)
    - [4.5.1.2 Video Provider Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.1.2-video-provider-verification)
  + [4.5.2 Functions Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.2-functions-verification)
    - [4.5.2.1 Create Workout Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.2.1-create-workout-verification)
    - [4.5.2.2 Find Workout Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.2.2-find-workout-verification)
    - [4.5.2.3 Saved Workout Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.2.3-saved-workout-verification)
    - [4.5.2.4 Workout in Progress Page Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.2.4-workout-in-progress-page-verification)
    - [4.5.2.5 My Video page Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.2.5-my-video-page-verification)
  + [4.5.3 Usability Requirements Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.3-usability-requirements-verification)
  + [4.5.4 Performance Requirements Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.4-performance-requirments-verification)
  + [4.5.5 Logical Database Requirements Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.5-logical-database-requirments-verification)
  + [4.5.6 Design Constraints Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.6-design-constraints-verification)
  + [4.5.7 Software System Attributes Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.7-software-system-attributes-verification)
    - [4.5.7.1 Reliability Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.7.1-reliability-verification)
    - [4.5.7.2 Security Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.7.2-security-verification)
    - [4.5.7.3 Maintainability Verification](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#4.5.7.3-maintainability-verification)
* [References](https://github.com/voyager1winterberry/cse372-01srs/blob/main/srsV3-0.md#references)

Table of Tables

[Table 1 Main Features of myJym 7](#_Toc89765533)

[Table 2? 7](#_Toc89765534)

[Table 3? 11](#_Toc89765535)

[Table 4? 12](#_Toc89765536)

[Table 5 Definitions 16](#_Toc89765537)

Table of Figures?

**1.1 Purpose**

The myJym app is to help people at all levels of experience feel confident using gym equipment. It also focuses on connecting users to personal trainers. Additional features include work out regiments, gym maps, and ability to track fitness goals.

**1.2 Scope**

The scope outlines what will and will not be included in the project.

**1.2.1 Products to be Produced**

The products to be produced is all the main programs that will be needed to make the product work. The myJym app will utilize:

* server
* database
* mobile myJym app
* web based myJym app

**1.2.2 What Will the Software Product Do**

This is a description of what the product to be produced will do.

* The product, the myJym app, will assist users in creating workouts, executing a workout and connecting with trainers.

**1.2.3 Included Core Features**

This is a table outlining the main features of the app with a brief description of what the feature is.

| **Feature** | **Description** |
| --- | --- |
| Login Page | The page that allows users to login with their credentials. |
| Home Page | The first page that users will see after logging into the app. |
| My Trainers | Page to mediate interaction between trainers and gym goers. |
| My Workouts | Page that allows creation and execution of workouts. |
| My Videos | Page that has a list of links to all videos used by the app. |
| Gym Map | A map showing the location of equipment, services, and gym features. |

Table 1 Main Features of myJym

**1.2.4 Excluded Features**

This is a table outlining what features are outside the scope of this project with a brief description of the feature.

| **Feature** | **Description** |
| --- | --- |
| Nutrition Information | Not intended to recommend meal plans, diets, count calories or anything else similar. |
| Video Platform | Not intended to have recorded workout videos to follow. |

Table 2?

**1.3.1 Product Perspective**

This section details the relationship of the myJym app to products that are simmilar or related to it.

**1.3.1.1 System Interfaces**

The system interfaces will be the systems that the application will be running on. The list below will be the systems used for the application.

* The myJym app will run on mobile services.
* The myJym app will run on web services.
* The myJym app will connect to google authentication for login credentials.

**1.3.1.2 User Interfaces**

The user interfaces will be the functionality of the app. This functionality is what the user will see when in the app.

* The myJym app will include a built-in messaging system to allow communication between users.
* The myJym app will include a sign-in page and a login page for the user.
* The myJym app will allow users to update their profile information.
* The myJym app will use the current systems keyboard to allow user input.
* The myJym app will have interactive buttons that will do different things according to the buttons pressed. These buttons will change the current page, modify workouts, and work with myJym’s built-in messaging.
* The myJym app will have a search bar that will allow the user to search for anything regarding the application such as different workouts. The search bar will allow for quick searches for the user's needs.
* The myJym app will have a settings page which will have everything in the page that the user can see and change in the app. Settings will include the user's profile and trainers they work with.
* The myJym app will have a workout page which will allow the user to create, delete, modify, and save workouts.
* The mJym app will have a myGym page which will find the gym you will be going to and a built-in map of that gym.

**1.3.1.3 Hardware Interfaces**

A hardware interface specifies the plugs, sockets, cables and electrical signals that pass through each line between the CPU and a peripheral device or communications network.

* Since the application will be run over the internet and cell service data, most of the hardware requirements will ensure the ios and android devices are able to connect to the internet. Since myJym will be a fitness app, features of myJym shall require users to have the feature of location tracking for personal coach and exercise equipment tutorial videos on their mobile device. Preferably, iPhone 6 and later gen, for the purpose of app store update reasons. The version of the iOS system shall be 15.1.1, and android system shall be 11.

**1.3.1.4 Software Interfaces**

Software interfaces (programming interfaces) are the languages, codes and messages that programs use to communicate with each other and to the hardware. Examples are the Windows, Mac and Linux operating systems, SMTP email, IP network protocols and the software drivers that activate the peripheral devices.

* Specify the use of other required software products.
  + Operating System: iOS and Android devices, and desktop applications, and the content will be operating within the mobile application.
  + Linkage: Users shall be allowed to sign up myJym with their existing personal email address to login to their accounts.

**1.3.1.5 Communications Interfaces**

* Communication interfaces shall include communication between the users.
* Communication interfaces shall include communication between the **trainers**, including a means of messaging within the app.
* Communication shall include a means of communicating with myJym to receive help.

**1.3.1.6 Memory**

* **Primary memory** shall be used to store the user’s data such as **profile** and user agreements. **Secondary memory** will not be utilized.

**1.3.1.7 Operations**

* Operations by the user shall include:
  + Creating a profile upon first use of the app
  + Connect with trainers that can offer insight and assistance into the workout schedule of the user
  + Learn how to use gym equipment by scanning a QR code on the machine and watching the associated video.
  + Operations by the Trainers shall include:
  + Creating a profile upon first use of the app
  + Receiving approval from myJym to contact and train users through an approval process -Connect with users through messaging and other means through the app to assist with training goals

**1.3.1.8 Site Adaption Requirements**

The system must be available in the following formats.

* Location (either through HW or User input), Network capability (connect to internet to connect to the server and download media), and I/O (Screen/Video, Keyboard, Mouse/Touch, Sound), Camera (to read QR codes)
* On a device without location services, the myJym app shall allow the user to type in the location.
* On a device without network capabilities, the myJym web app shall be unable to open.
* On a device without network capabilities, the myJym mobile app shall use cached...
* On a device without a camera to read QR codes, the myJym app shall allow the user to type in the url associated with the QR code.

**1.3.1.9 Interfaces with Services**

The different services that the system will interact with.

* Video links from a video provider links to the various workout videos must be accessible
* The server will connect accounts to the main database to access the account information

**1.3.2 Product Functions**

For the product functions of the myJym application the following User Profiles and User Scenarios will be as follows:

**1.3.2.1 User Profiles**

The following User Profiles define the target demographics for the myJym application:

* A returned missionary attending BYU-Idaho, in their early twenties, has a busy schedule with work and a heavy course load.
* A student in their junior year, also attending BYU-Idaho, in their early twenties, they enjoy playing football with their friends and family.
* A new freshman, halfway through the semester wanting to lose weight, but unsure of where to start.
* A couple that likes to spend time together by playing sports and working out. The application will be available to all age groups and demographics, however these groups will serve as the focus the application.

**1.3.2.2 User Scenarios**

The following User Scenarios serve to describe the use cases that have been elicited to be applicable to the product functions of the myJym application:

* Only being able to attend the gym in the early morning, with no previous time to set a workout routine or research workout machines.
* Brand new to exercising, there is a fear of feeling inadequate and confused in front of others, and a desire to be helped in a discreet way.
* Getting burnt out with their current workout routine, and desiring new machines and techniques that can help them develop new routines.
* A student desires additional experience with a variety of exercises including strength training, endurance training, weight loss, and cardio training to be able to better meet their fitness goals.

**1.3.3 User Characteristics**

This table is for the user characteristics from the perspective of a trainee user.

| ***User Group*** | ***Trainee*** |
| --- | --- |
| User Profile | This User group will be using the myJym application to connect to a personal trainer, as well as obtain information on workout equipment and techniques. The Trainee will have to register on the myJym application and have time available to meet with a Personal Trainer, learn workout equipment and routines |
| ***Demographic Criteria*** |  |
| Age | 18 - 70 |
| Income | Middle class income |
| Education | All forms of education |
| Geography | Cities with public gyms and access to personal trainers locally |
| City Size | Large enough city with access to gyms and trainers |
| Disabilities | Many workouts require the user to be a fully functional adult |
| Gender | Any gender |
| Technical Expertise | Competent, able to navigate through most applications |
| ***Psychographic Criteria*** |  |
| Information Availability and Content | Information availability is key information if the user wants to connect to the Personal Trainer. If user wants to understand gym equipment and make a workout routine. Simple information is required about the user like age, sex, and location, but no detailed information about the user is required |
| Ease of Use | Users require and easy-to-use application, that can be operated with mobile devices |
| Privacy and Security | Security, as well as user information security, need to be protected via encryption or password. |
| Graphic Style | Application appearance/graphics should be functional but most importantly readable, there should be no small fonts or distracting colors or images to allow easy use, as well as allow for people with disabilities and international constraints |
| Fulfillment and reliability | Reliability is essential. The application should always be functional since the ability to constantly communicate between users and trainers should be available |

Table 3?

This table is for the user characteristics from the perspective of a trainer user.

| ***User Group*** | ***Personal Trainer*** |
| --- | --- |
| User Profile | This Personal Trainer group will be using the myJym application to connect to trainees, as well as obtain information on available equipment and trainees looking for a personal trainer. The Personal Trainer will have to register on the myJym application, and have time available to meet with a trainee, and have a clear understanding of equipment and techniques, and able to teach and make a workout routine for the trainee |
| ***Demographic Criteria*** |  |
| Age | 18 - 50 |
| Income | Middle class income |
| Education | Personal Trainer Certificate |
| Geography | Cities with public gyms and access to personal trainers locally |
| City Size | Large enough city with access to gyms |
| Disabilities | Many workouts require the user to be a fully functional adult |
| Gender | Any Gender |
| Technical Expertise | Competent, able to navigate through most applications |
| ***Psychographic Criteria*** |  |
| Information Availability and Content | Information availability is key information if the user wants to connect to the Personal Trainer. The trainer should have a public profile on the myJym application which displays the trainer's availability and method of teaching, but no detailed information about the user is required |
| Ease of Use | Users require and easy-to-use application that can be operated with mobile devices |
| Privacy and Security | Security, as well as user information security, need to be adequate. Though no strong security or privacy settings are applicable |
| Graphic Style | App appearance/graphics should be functional but most importantly readable, there should be no small fonts or distracting colors or images to allow easy use |
| Fulfillment and reliability | Reliability is essential, the application should be always functional since ability to constantly communicate between users and trainers should be available |

Table 4?

**1.3.4 Limitations**

This section will focus on providing a general description of any aspect of the project that could result in a supply or hardware restriction limitation.

**1.3.4.1 Regulatory Requirements and Policies**

HIPPA / GDPR - Restrictions on collecting personal information. Important to be transparent. Ensure that personal information does not have to be given to operate, only given voluntarily.

**1.3.4.2 Hardware Limitations (e.g. signal timing requirements)**

Hardware requirements are a camera, internet access, and a smart phone. The computers of the development team are a limiting factor.

**1.3.4.3 Interfaces to Other Applications**

One limitation of interfacing with other applications would be the licensing of a third-party video service. The video service might also run ads, this hurt the user experience. The messaging service could require the user to create an account, this could deter some users from using the app.

**1.3.4.4 Parallel Operation**

Calorie tracking will require the use of a smart watch or fitness tracker which not everyone will have. Bluetooth would be used to connect the devices; Bluetooth could cause problems for less tech savvy users.

**1.3.4.5 Audit Functions**

Bug report tool/ user feedback. The system may not get relevant data from the user feedback, and the system will need to handle user feedback.

**1.3.4.6 Control Functions**

Errors in the code of the myJym app could limit the control functions from functioning properly.

**1.3.4.7 Higher-Order Language Requirements**

Since this is a mobile application two different apps need to be written, one for android and one for apple. This will require the use of Kotlin and Swift so the developers will need to learn or know at least one of these languages. This could also cause problems in ensuring that the apps are the same. A way to get around this would be to use a web app using Django, HTML, CSS, and JS.

**1.3.4.8 Signal Handshake Protocols**

Internet access will be the limitation for all signal handshake protocols. The protocols themselves will be limit the signal handshakes that are happening.

**1.3.4.9 Quality Requirements (e.g. reliability)**

The app needs to be easily accessible, efficient, and able to handle data.

**1.3.4.10 Criticality of the Application**

Most of the app is dependent upon personal trainers and YouTube. If either of these fail the app will have premade workouts, in the form of written descriptions, for the gym goer to reference.

**1.3.4.11 Safety and Security Considerations**

One of the limitations is that the system must store personal user data such as usernames, passwords, names, weights, and heights.

**1.3.4.12 Physical/Mental Considerations**

The system will need to consider the physical limitations of the gym goer. The system will need to include a liability waiver in the terms and conditions.

**1.3.4.13 External System Limitations**

The limitations of a video streaming service will apply to the app.

**1.4 Definitions:**

This is a comprehensive list of needed definitions used by the myJym application:

| **Word** | **Definition** |
| --- | --- |
| Account | An arrangement allowing a user personalized access to the myJym application, requiring the use of a username and password to access. |
| Admin | The people that run the app. They will connect the user's location to the gym and create QR codes that can be scanned by the user. |
| Android 11 | The minimum android version that will be required for the project. |
| Audit Functions | Functions of the application that allow for user feedback and error handling. |
| App | myJym mobile application. |
| Client | see *trainee* |
| Control Functions | The interface with which the user will interact with and control the app. |
| Criticality of the Application | Parts of the application have dependencies that can fail because of them. The most critical parts of the application. |
| The Database | A database that will store all information needed to run the application contraining the following tables: Admin, UserData, Trainer, and WorkOut. |
| Equipment | The machines at the student gym. |
| Email | A method of exchanging messages over the internet between electronic devices. |
| Embedded Video | Embedded Videos: Videos in applications that play a video inside of the app without having to go to the place containing the original video. |
| Error Message | A message stating that a condition is incorrect and requires that the condition be different. |
| Forms | Digital way to collect relevant information from the user. |
| Google | An American company that specializes in internet-related services and products. |
| Google Authentication Key | A key provided by Google to authenticate users. |
| GUI | GUI (Graphical User Interface) is a user interface that includes graphical elements, such as windows, icons, and buttons. The purpose is to appeal to the user using visuals. |
| Gym | A location with workout machines available. |
| Gym Hours | The hours that all partner gyms are open. |
| Hardware Limitations | Physical technological factors that might limit or slow the progress of the app and its construction. |
| Higher-Order Language Requirements | Knowledge of high-level programming languages such as Kotlin and Swift. |
| Home Page | The view which allows access to myTrainer, myWorkout, myVideos, and myGymMap views. |
| Interface | A point where two systems, subjects, organizations, etc. meet and interact. |
| Interfaces to other applications | The ability to interact with third party applications. |
| Invalid Credentials | Login credentials that do not match those associated with an account. |
| iOS | Internet Operating System or iPhone Operating System. It is the operating system used on Apple products, such as iPhone and iPad. |
| IOS 14.1 | The minimum IOS version that will be required for the project |
| Jared's Senior Project Class | A project class that the project sponsor will be taking at some point in the future. |
| Login Credentials | The username and password associated with an account. |
| Login Page | A view that requests that the user login. |
| macOS | The Macintosh Operating System is an operating system designed by Apple Inc. to be installed and operated on the Apple Macintosh series of computers. It is a (GUI) based OS that has since been released in multiple different versions. |
| Mnemonic | The device that uses patterns of letters, associations to assist users to remember a specific feature and its functionality. |
| Mobile Application (myJym) | The application that will be used to allow communication/digital connection between the user and personal trainer. It will also hold the data and information about the equipment available through QR codes. |
| Modal Income | Is the income amount that divides the population into two groups, half having the income above average and the other half below the average. |
| The myJym app | An application that provides a variety of features for the purpose of making the gym a more approachable experience. |
| Name | The name of the software’s feature. Generally, the name of the feature needs to be self-explanatory and easy to understand. |
| Parallel Operation | The ability of an application to run in parallel to other applications without getting in their way or being interrupted by them. |
| Partner Gyms | These are gyms that are partnered with the myJym app. |
| Password | A key word or phrase associated with an account. |
| Password Recovery Email | An email the gym-goer and/or trainer provides with account creation to recover their account if they forget their password. |
| Personal Trainer | Individuals who have earned certification that demonstrates they have achieved a level of competency for exercising safely and efficiently along with creating safe and effective exercise programs for those who have the medical clearance to participate. Personal trainers assist clients. |
| Personal Trainer Certificate | Individuals who have earned certification that demonstrates they have achieved a level of competency for exercising safely and efficiently along with creating safe and effective exercise programs for those who have the medical clearance to participate. Personal trainers can assist clients, and can do so while working in conjunction with the myJym app. |
| Physical/mental considerations | Considerations that pertain to the physical and mental health of the application. |
| Primary memory | Memory utilized by the app on the individual devices |
| Profile | A personalized account created by the user that will be associated with them and required to utilize the app. |
| QR Scanning | The ability of a mobile device to capture QR codes in their camera and then open the associated links. |
| Quality Requirements | The reliability of the application, as well as the other quality metrics that represent how well it fulfills its tasks. |
| Regulatory Requirements and Policies | Application requirements that need to comply with legal regulations and policies. |
| Safety and Security Considerations | Keeping the user’s physical and personal safety in mind, including sensitive or confidential information |
| Secondary memory | removable or external memory used by the individual devices. |
| Signal Handshake Protocols | How the app interacts securely with other applications and the internet. |
| Source | A specific reference for the research and citation purpose to avoid plagiarism. |
| Specification number | For the purpose of organization feature within the SRS documentation. |
| The System | The backend code that processes the server requests to send or return data as requested by the server through the user's requests. |
| Trainers | see *Personal Trainer* |
| Trainee | A gym-goer being trained by a personal trainer. |
| User | A gym-goer and/or a trainer. |
| User-admin | User that acts as an administrator. |
| User-trainer | User who instructs gym-goers. |
| Version number | A set of unique numbers that are assigned to a specific release of a software project. In this case, it will be myJym version 1.0.0 |
| Working camera | camera capable of taking clear photos |
| Workout Plan | A set of exercises, times, and practices a user will complete that are displayed in the application. This can be completed individually or with a personal trainer |

Table 5 Definitions

**2.0 References**

**3.0 Functions**

**3.1 My Trainer**

The My trainer portion is a page accessible through the homepage.

**3.1.1 External Interfaces**

The external interfaces section will cover 3rd party programs that will be used.

* 

**3.1.1.1 Google Authentication**

Google authentication will go over all the requirements for the 3rd party interface. This is all the things that google will be doing for the application.

***3.1.1.1.1*** Google Authentication API shall be used by gymgoers and trainers to login to their accounts.   
***3.1.1.1.2*** All users shall be able to login to their account using a google account.   
***3.1.1.1.3*** All users shall be able to create an account using a google google account.

****

**3.1.2 Functions**

The myJym application will allow users to view the My Trainer page and receive system notifications.

***3.1.2.1*** The myJym application shall send users Notifications.   
***3.1.2.2*** The myTrainer page shall allow users to view trainer profiles.   
***3.1.2.3*** The myTrainer page shall allow users to search for trainers.   
***3.1.2.4*** The myTrainer page shall allow users to update their profile.   
***3.1.2.5*** The myTrainer page shall allow users to message trainers.   
***3.1.2.6*** The myTrainer page shall allow trainers to message users.

**** **3.1.2.1 Notifications**

Notifications will allow users to receive notifications on their device about scheduled workouts and sent messages.

***3.1.2.1.1*** A notification shall be sent to the user’s devices when a user sets a time for their workout.   
***3.1.2.1.2*** A notification shall be sent to the user’s device when the user receives a message.

****

**** **3.1.2.2 View Trainer Profile**

The trainer profile will provide a viewable page for users with the trainer's photo, short bio, video links from a video provider, a follow button, a link to workouts created by the trainer, and a button to message the trainer.

***3.1.2.2.1*** Trainer profile shall be viewable by gymgoers.   
***3.1.2.2.2*** Trainer profile shall display a picture of the trainer.   
***3.1.2.2.3*** Trainer profile shall display a short biography of the trainer.   
***3.1.2.2.4*** Trainer profile shall display a clickable link to the trainer’s videos on a video provider.   
***3.1.2.2.5*** “View Workouts” shall be an option on the Trainer profile that allows the user to view a trainer’s created workouts.   
***3.1.2.2.6*** “Message Trainer” shall be an option on the Trainer profile that allows the user to send a message to the trainer.   
***3.1.2.2.7*** Trainer profiles shall store a trainers first name   
***3.1.2.2.8*** Trainer profiles shall store a trainer’s last name.   
***3.1.2.2.9*** Trainer profiles shall store a trainer’s email.   
***3.1.2.2.10*** The system shall create a unique ID for each trainer.   
***3.1.2.2.11*** The system shall create a personal key for each trainer.

****

**** **3.1.2.3 Update Profile**

The update profile function will allow any user to make changes to their profile.

***3.1.2.3.1*** Update Profile shall allow any user to update their own picture.   
***3.1.2.3.2*** Update Profile shall allow any user to update a biography text-field.   
***3.1.2.3.3*** Update Profile shall allow any user to update their personal video links.   
***3.1.2.3.4*** Update Profile shall allow any user to update their first and last name.

****

**** **3.1.2.4 Search Trainer**

Search trainer will allow users to search for trainers by name, location, and workout.

***3.1.2.4.1*** The system shall have a search bar.   
***3.1.2.4.2*** A search bar shall be viewable by the user.   
***3.1.2.4.3*** Search trainer shall allow a user to search for a trainer by name.   
***3.1.2.4.4*** Search trainer shall allow a user to search for a trainer by location.   
***3.1.2.4.5*** Search trainer shall allow a user to search for a trainer by workouts.

****

**** **3.1.2.5 Message Trainer**

Message trainer will allow a user to communicate with a trainer.

***3.1.2.5.1*** Message Trainer shall give a user the ability to send messages to a trainer.   
***3.1.2.5.2*** Message Trainer shall list messages in chronological order.   
***3.1.2.5.3*** Messages Trainer shall be able to handle messages containing text.   
***3.1.2.5.4*** Message Trainer shall be able to handle messages containing links.

****

* 

**3.1.2.6 Message Users**

Message Users shall allow the trainer to communicate with users.

***3.1.2.6.1*** Message users shall give trainers the ability to send messages to users.   
***3.1.2.6.2*** Message users shall create a chat between trainer and users.   
***3.1.2.6.3*** Message users shall list messages in chronological order.   
***3.1.2.6.4*** Messages Trainer shall be able to handle messages containing text.   
***3.1.2.6.5*** Message Trainer shall be able to handle messages containing links

****

**** **3.1.3 Usability Requirements**

Requirements that are part of the user experience. It is the different aspects of the system such as understandability, accessibility, task failure and more that impact how a user views and uses the system.

***3.1.3.1*** Ease of learning. The system shall be easy to learn for both novices and users with experience from similar systems. Users familiar with similar systems shall be able to create a new account with minimal errors.   
***3.1.3.2*** Task efficiency. The system shall be designed to be simple to use for frequent users. Frequent users shall be given an option to be able to access their account without having to sign in each session. Tasks shall be able to be completed by the frequent user quickly and efficiently.   
***3.1.3.3*** Ease of remembering. The system shall be clear and well defined. Users experienced with this system shall be able to login to their account quickly.   
***3.1.3.4*** Understandability: A new user shall be able to navigate to any desired option quickly and efficiently.   
***3.1.3.5*** Subjective satisfaction. The system shall receive a high rating by test groups when released.   
***3.1.3.6*** Task failure. Task failure by the system shall be kept to a minimum.   
***3.1.3.7*** Requirements to undo. The system should be able to allow users are able to try things and undo them. The ability to undo should be incorporated into any part of the system that stores information, or task.   
***3.1.3.8*** Accessibility. The system shall be created with elements that ensure that it is accessible to adults ranging from 18-40. These elements include but are not limited to: color, text, complexity, and exclusivity.

****

* **3.1.4 Performance Requirements**

The performance requirements are showing the performance of unique features of my trainer page. The list will be a list of the features of my trainer page.

***3.1.4.1*** My trainer page shall load when the user navigates to my trainer page.   
***3.1.4.2*** The trainer shall receive messages soon after the author sends the message.   
***3.1.4.3*** The trainer shall be able to send messages after clicking the send button in the built-in messenger.   
***3.1.4.4*** Notifications for the trainer shall be received at the same time as when the message comes in.   
***3.1.4.5*** Notifications from the trainer shall be received at the same time the recipient receives the message.   
***3.1.4.6*** The trainer profiles on the web shall be loaded soon after the settings button has been clicked.   
***3.1.4.7*** The trainer profiles on the mobile app shall be loaded soon after the settings button has been clicked.   
***3.1.4.8*** Trainer search results on the web shall load up soon after the search has been entered in.   
***3.1.4.9*** Trainer search results on the mobile app shall load up soon after the search has been entered in.

**3.1.5 Logical Database Requirements**

**** **3.1.5.1 Types of information used by various functions**

The system shall include several types of data for each account.

***3.1.5.1.1*** MyTrainer shall store a trainers first name.   
***3.1.5.1.2*** MyTrainer shall store a trainer’s last name.   
***3.1.5.1.3*** MyTrainer shall store a trainer’s email.   
***3.1.5.1.4*** The system shall create a trainer\_ID for each trainer.

****

** 3.1.5.2 Frequency of use**

Data shall not be reused between accounts. This will require all emails and primary keys to be unique to each account.

***3.1.5.2.1*** Email data types shall be used for a single account.

****

** 3.1.5.3 Accessing Capabilities**

The following is the different access that should be given between accounts.

***3.1.5.3.1*** Trainers shall have access to UserData first\_name.   
***3.1.5.3.2*** Trainers shall have access to UserData last\_name.   
***3.1.5.3.3*** Trainers shall have access to UserData goals.   
***3.1.5.3.4*** Trainers shall have access to UserData W/O\_plan.   
***3.1.5.3.5*** Trainers shall have access to UserData gym.

****

** 3.1.5.4 Integrity Constraints**

The following are constraints and regulations on the data within the database.

***3.1.5.4.1*** Data entries shall be possible entries.   
***3.1.5.4.2*** Impossible data entries shall not be allowed.   
***3.1.5.4.3*** Data entries shall be verified to ensure it is possible.   
***3.1.5.4.4*** The database shall have entity integrity.

****

* **3.1.5.5 Security**

The following refers to account security and the different measures that should be taken to ensure account safety.

***3.1.5.5.1*** Trainer accounts shall utilize google sign-in to access their account.   
***3.1.5.5.2*** User accounts shall utilize google sign-in to access their account.   
***3.1.5.5.3*** Trainers shall only have one account.   
***3.1.5.5.4*** Gymgoers shall only have one account.   
***3.1.5.5.5*** Admin accounts shall have access to all accounts.

* **3.1.6 Design Constraints**

Design constraints are conditions that need to happen for a project to be successful. Design constraints help narrow choices when creating a project.

***3.1.6.1*** The interactable map features shall provide sufficient Pixels Per Inch that the average human is able to cleanly select them.

* **3.1.7 Software System Attributes**

Software Quality Attributes: Features that facilitate the measurement of performance of a software product by Software Testing professionals, and include attributes such as availability, interoperability, correctness, reliability, learnability, robustness, maintainability, readability, extensibility, testability.

**** **3.1.7.1 Reliability**

***3.1.7.1.1*** Applications shall not have errors within the codes to cause applications to close or crash.   
***3.1.7.1.2*** There will be functions within the software that performs verifications to notice errors before the application break or force closes.   
***3.1.7.1.3*** The system shall verify the information that matches from the server and the application.

****

**** **3.1.7.2 Security**

***3.1.7.2.1*** Password Protection and Encryption   
***3.1.7.2.2*** User account creation   
***3.1.7.2.3*** User information shall be available to users and those have access.   
***3.1.7.2.4*** User shall be required to provide email, and password for the purpose of account creation.

****

**** **3.1.7.3 Data Privacy**

***3.1.7.3.1*** User’s data shall be hidden from all other users and parties.   
***3.1.7.3.2*** Login Information   
***3.1.7.3.3*** Creating an account shall be safe and secure.   
***3.1.7.3.4*** All users shall be allowed to login securely.   
***3.1.7.3.5*** Once a user logs out from the account, all user information shall be removed from the login screen.

****

**** **3.1.7.4 Maintainability**

***3.1.7.4.1*** Users shall be allowed to report to the administration team if theyforget their password, and need to have a reset password email sent totheir personal email that has been registered on their myJym account.   
***3.1.7.4.1*** An automatic email shall be sent to the user's email address to allow the user to reset their password.

****

**** **3.1.7.5 Portability**

***3.1.7.5.1*** System Database shall store user information.   
***3.1.7.5.2*** myJym application shall run on both Android and iOS mobile interface.

****

* **3.1.8 Supporting Information**

Description of the problems to be solved by the software: A problem that this software will solve is to give users the ability to connect with trainers. Trainers will be able to provide information to users about workouts and feedback on workout plans. Trainers will be able to access user data to be able to help them choose workout plans and give them the motivation to meet their goals.

**3.2 Gym Map**

The map of the gym used by the MyJym application and it's requirements will be detailed in this section.

**3.2.1 External Interfaces**

***3.2.1.1*** The map shall utilize the System Database to obtain needed information for the map generation.   
***3.2.1.1*** The map search bar shall obtain user data from the **System Database**.

**3.2.2 Functions**

Functions of the gym map and it's associated features are detailed below:

**3.2.2.1 Map Display**

The map display will be responsible for how the various graphical display elements will be handled. These are as follows:   
***3.2.2.1.1*** The map shall organize gym equipment according to a JSON file detailing their locations.   
***3.2.2.1.2*** The map shall organize gym services according to a JSON file detailing their locations.   
***3.2.2.1.3*** When an abnormal state is encountered the map shall stop and refresh.   
***3.2.2.1.4*** The map shall support modification to facility locations by a system administrator.   
***3.2.2.1.5*** A simple tutorial of using the map shall display the first time a user opens to that view.

**3.2.2.2 Map Search Function**

The map features a search bar with the following capabilities:   
***3.2.2.2.1*** The map search bar shall perform validation on inputs matching them to gym functions.   
***3.2.2.2.2*** The map shall provide a search bar used to locate gym functions.   
***3.2.2.2.3*** The map search bar shall be a modeless interface not restricting user input to other parts of the application.   
***3.2.2.2.4*** Facilities in the gym shall be searchable by name through the map search bar.

**3.2.2.3 Map Navigation**

The requirements for navigating on the map are as follows:

***3.2.2.3.1*** The map shall be scrollable edge to edge.   
***3.2.2.3.2*** The map shall support zooming, detecting when a zoom request is made.   
***3.2.2.3.3*** Gym function icons shall be interactable to view options related to that function.

* **3.2.3 Usability Requirements**

For the myGym app the usability requirements serve to increase the usability for the user include the ease of learning, task efficiency, ease of remembering, understandability, task failure, and accessibility. These are criteria which serve to increase the usability of myJym.

These are as follows:   
***3.2.3.1*** Ease of learning- The map shall be intuitive allowing the user to easily form a mental model of its workings. This will be facilitated by a clean organization for the display elements of the map.   
***3.2.3.1*** Task efficiency- The map shall facilitate a user quickly identifying where they should be in the gym to complete their desired activity.   
***3.2.3.2*** Ease of remembering. The map shall utilize simple command formats to ensure users are capable of creating a mental model of the map functionalities.   
***3.2.3.3*** Understandability: The user shall understand the purpose of the map. Map features shall use formats akin to those found on similar mobile applications. New users will be greeted by a tutorial upon first entering the map view.   
***3.2.3.4*** Task failure. When encountering a task failure the map shall stop its operation and refresh.   
***3.2.3.5*** Accessibility. Map iconography shall be created based upon commonly recognized symbols.

**3.2.4 Performance Requirements**

The performance requirements of the system for the myJym map feature will define the performance capabilities and how various conditions will impact their operation.

These are as follows:   
***3.2.4.1*** The map shall prioritize loading of graphical display elements prior to lower priority tasks.   
***3.2.4.2*** The map search functionality shall return results in higher performant step timing than\*\* O\*\*(n log n).   
***3.2.4.3*** The map shall allow interaction with elements after a predetermined time post loading.

**3.2.5 Logical Database Requirements**

The data used by the gym map and it's state will be detailed here as follows:

**3.2.5.1 Types of information used by various functions**

The following will be the information used by the myGym map to perform it's required functions:   
***3.2.5.1.1*** The map shall store the icons for the map as a data construct.   
***3.2.5.1.2*** The map shall use search inputs to retrieve data from the **System Database**.

**3.2.5.2 Frequency of use**

The following will define how often the various functionalities of the myJym map will be accessed.   
***3.2.5.2.1*** Map icons shall be loaded into memory from the **System Database** when the view is selected.   
***3.2.5.2.2*** The map search functionality shall be accessible on demand.

**3.2.5.3 Accessing Capabilities**

The map will be able to be accessed from and have access to the following:   
***3.2.5.3.1*** The map shall be accessible from the home page of the myJym application.   
***3.2.5.3.2*** The map shall have access to the **System Database** to access gym equipment locations.   
***3.2.5.3.3*** The map shall have access to the **System Database** to return search results.

**3.2.5.4 Data entities and their relationships**

The data entities will have the following relationships:   
***3.2.5.4.1*** The map shall handle data for the exercise machine locations.   
***3.2.5.4.2*** The map shall handle data for the exercise machine images.   
***3.2.5.4.3*** The map shall handle data for the map search bar results.

**3.2.5.5 Integrity Constraints**

The data stored by the map will be subject to the following integrity constraints:   
***3.2.5.5.1*** Map icons shall be entries to the database.   
***3.2.5.5.2*** Map data shall be formatted to match the conventions of the database.

**3.2.5.6 Security**

The map data security will apply as follows:   
***3.2.5.6.1*** Data used for the map shall not be accessible by other functions of myJym.

**3.2.6 Design Constraints**

The design constraints of the map, the limitations on it's design, will be as follows:   
***3.2.6.1*** The interactable map features shall provide sufficient **Pixels Per Inch** that the average human is able to cleanly select them.

**3.2.7 Software System Attributes**

The system attributes used by the myJym app will detail the reliability and availability of the map:   
***3.2.7.1*** The map shall be available when the application has been opened to the home page.   
***3.2.7.2*** The map shall be available offline when not internet connectivity is available.

**3.2.7.1 Reliability**

The factors of reliability for the myJym map shall be as follows:   
***3.2.7.1.1*** The maps delivered to the user on MyJym shall deliver accurate data based on the information in the **System Database.**   
***3.2.7.1.2*** The MyJym map shall be retained in memory while the myJym application is focused on a separate page.

* **3.2.7.2 Security**

The following act to maintain the integrity and security of the myJym map:   
***3.2.7.2.1*** The map shall only be editable by an admin.

**3.3 MyJym Videos**

This Specification section explains the MyJym application in detail, the specification goes over eight categories. Such as external interfaces, functionality, usability, performance, database, design constraints, and software system attribute requirements for the MyJym application.

**3.3.1 External Interfaces Specification**

This section defines the inputs and outputs of the MyJym application.

**3.3.1.1 Database**

This shows the what columns will be included to create the database. The specific column requirements are listed in section 3.5  
***3.3.1.1.1*** The database shall store workout title  
***3.3.1.1.2*** The database shall store workout video links  
***3.3.1.1.3*** The database shall store workout description  
***3.3.1.1.4*** The database shall store the image links

**3.3.1.2 Video Streaming**

The API for a video streaming website will be used to share videos on workout techniques and forms, and how to use gym exercise equipment. This will be used according to the product owner and stakeholder’s decisions.

***3.3.1.2.1*** Gymgoers shall have the ability to access workout videos

**3.3.2 Functions**

This section is responsible for explaining MyJym’s ability to perform to the User’s standard. These functions are split up into sections that explain the procedure of the MyJym application in how to perform or complete tasks.

**3.3.2.1 My Video**

The purpose of the following section provides the capabilities of the video page. This represents an individual user for the MyJym application.

***3.3.2.1.1*** The My Video page shall display a workout title.  
***3.3.2.1.2*** The My Video page shall display an equipment name.  
***3.3.2.1.3*** The My Video page shall display a workout video.  
***3.3.2.1.4*** The My Video page shall provide a save video button.

**3.3.3 Usability Requirements**

This section lists the quality and use requirements for the MyJym application. Including measurable effectiveness and satisfaction criteria for the My Video feature.   
***3.3.3.1*** User shall be able to save a selected video  
***3.3.3.2*** User shall be able to view the workout video  
***3.3.3.3*** User shall be able to view saved videos

**3.3.4 Performance requirements**

For the MyJym application to meet performance standards, there are specific requirements that are needed to be met, specified below.  
***3.3.4.1*** User shall be able to save a selected video  
***3.3.4.2*** The user shall only be able to save one video link per exercise.  
***3.3.4.3*** The My Video page shall support a configurable amount of workouts  
***3.3.4.4*** The My Video page shall support a configurable amount of workouts videos

**3.3.5 Logical Database Requirements**

The MyJym database shall store all the tables that are necessary for storing information required by the application.

***3.3.5.1*** The workout table shall contain an exercise title column  
***3.3.5.2*** The workout table shall contain a description column  
***3.3.5.3*** The workout table shall contain a video link column  
***3.3.5.4*** The workout table shall contain a workout id column that acts as a primary key

**3.3.6 Design Constraints**

This section specifies the constraints with the myJym application that would arise from project limitations, external standards, and specific requirements.  
***3.3.6.1*** The user shall only be able to save workout the video

**3.3.7 Software System Attributes**

**3.3.7.1 Reliability**

Specifies factors required to establish reliability of the myJym application at the time of delivery.

***3.3.7.1.1*** The system shall exit immediately when a user finds an error.

***3.3.7.1.2*** The app shall check that locally stored user information matches what is stored in a database.

**3.3.7.2 Security**

Specifies the factors required to guarantee a defined level of availability for the myJym application in regards to the workout videos.   
***3.3.7.2.1*** The user shall only be able to add approved video links to an exercise.

**3.3.7.3 Maintainability**

Specifies the factors required to define a level of maintainability for the myJym application.   
***3.3.7.3.1*** App administrators shall be able to remove inappropriate content.

**3.3.8 Supporting Information**

N/A

**3.4 Login & Homepage**

The first pages that the user will see. The user will login and then use the homepage to access the various features of the application.

**3.4.1 External Interfaces**

Communication with third party applications. The login page requires external interfaces to function. The following list details necessary external interfaces.

*3.4.1.1* The login page shall provide the ability to login to an account using login credentials.   
*3.4.1.2* The system shall send an automated password recovery email to the user on request

**3.4.2 Functions**

Operations that the system can perform. The following list details operations that are necessary for the login and homepage to properly function.

*3.4.2.1* The home page shall provide access to myTrainer.   
*3.4.2.2* The home page shall provide access to myWorkouts.   
*3.4.2.3* The home page shall provide access to myVideos.   
*3.4.2.4* The home page shall provide access to myGymMap.  
*3.4.2.5* The login page shall provide the user with the ability to recover a lost password.   
*3.4.2.6* The login page shall provide the user with the ability to create an account.   
*3.4.2.7* The login page shall display an error message when invalid credentials are submitted.   
*3.4.2.8* The system shall check the login credentials against the database.

**3.4.3 Usability Requirements**

Requirements relating to the ease of use of the application.

*3.4.3.1* The login page shall inform the user of password requirements.   
*3.4.3.2* The login page shall allow the user to enter login credentials.

**3.4.4 Performance Requirements**

Requirements relating to the performance metrics of the application.

*3.4.4.1* The system shall have the ability to process login credentials.

**3.4.5 Logical Database Requirements**

This section will define what the database will store as the application receives data.

*3.4.5.1* The UserData database table shall have a Google authentication key from the Google third party login.   
*3.4.5.2* The UserData database table shall store usernames.   
*3.4.5.3* The UserData database table shall store the passwords.   
*3.4.5.4* The database shall be accessed by the login page when user credentials are submitted through the login page.   
*3.4.5.5* The database shall keep the login credentials of users confidential.

**3.4.6 Design Constraints**

Standards that will define the design of the application.

*3.4.6.1* The myJym app shall be completed in its entirety by the end of Jared's Senior Project class.   
*3.4.6.2* The budget for the myJym app shall be $0.00.

**3.4.7 Software System Attributes**

Metrics used to describe myJyms’ features.

**3.4.7.1 Reliability**

Applications distinct types of use.

*3.4.7.1.1* The app shall require the compliance of personal trainers.  
*3.4.7.1.2* The app shall require an administrator.

**3.4.7.2 Availability**

When myJym will be available for use.

*3.4.7.2.1* The myJym app shall be available on demand of the user.

**3.4.7.3 Security**

Measures of what will be done to protect user’s assets

*3.4.7.3.1* The system shall encrypt user accounts.  
*3.4.7.3.2* The system shall use a protocol to secure online communications.

**3.4.7.4 Portability**

What will be done to distribute the application.

*3.4.7.4.1* The application shall be built with a programming language that can be ported to both iOS and Android.

**3.5 My Workout**

The workout function shall have the ability to let users create their own workouts.

The workout function shall have the ability to let users find premade workouts.

The workout function shall have the ability to let users view previously saved workouts.

**3.5.1 External Interface**

External interfaces are any interfaces outside the main products that will be created during this project.

**3.5.1.1 Database**

This list outlines what columns shall be included to create the database. Specific columns and requirements of the database can be found in section 3.5.

***3.5.1.1.1*** The database shall store workout **title**.  
***3.5.1.1.2*** The database shall store workout video links.  
***3.5.1.1.3*** The database shall store workout description.  
***3.5.1.1.4*** The database shall store workout **sets**.  
***3.5.1.1.5*** The database shall store workout **reps**.  
***3.5.1.1.6*** The database shall store workout **tags**.

**3.5.1.2 Video Provider**

The **API** for the video provider will be used to share videos on workout techniques and forms, and how to use gym exercise equipment. This will be used according to the product owner and stakeholder’s decisions.

***3.5.1.2.1*** **Gymgoers** shall have the ability to access workout videos.

**3.5.2 Functions**

This section is responsible for explaining what the myJym app is expected to perform. These functions are divided into sub-sections that explain the procedure, performance and tasks of the myJym app.

**3.5.2.1 Create Workout**

Create workout is a page within the app that allows users to create workouts to their own specifications. The following list is what is needed to achieve that functionality.

***3.5.2.1.1*** The workout creation shall allow the **user** to specify a workout name.  
***3.5.2.1.2*** The workout creation shall allow the user to specify a muscle group to workout.  
***3.5.2.1.3*** The workout creation shall allow the user to specify an amount of **reps**.  
***3.5.2.1.4*** The workout creation shall allow the user to specify an amount of **sets**.  
***3.5.2.1.5*** The workout creation shall allow the user to embed a video link.

**3.5.2.2 Find Workout**

Find workout is a page within the app that allows users to find workouts that other users have already created. The following list is what is needed to achieve that functionality.

***3.5.2.2.1*** The find workout page shall have the ability to search by predetermined **tag**s.  
***3.5.2.2.2*** One of the tags that can be used to search by shall be **author**.  
***3.5.2.2.3*** One of the tags that can be used to search by shall be **name of workout**.  
***3.5.2.2.4*** One of the tags that can be used to search by shall be **length**.  
***3.5.2.2.5*** One of the tags that can be used to search by shall be **workout type**.  
***3.5.2.2.6*** One of the tags that can be used to search by shall be **difficulty**.  
***3.5.2.2.7*** One of the tags that can be used to search by shall be **equipment type**.  
***3.5.2.2.8*** One of the tags that can be used to search by shall be **muscle groups**.  
***3.5.2.2.9*** The user shall have the ability to save a workout they find to their saved workouts list.

**3.5.2.3 Saved Workout**

Saved workout is a page within the app that allows users to save a workout that they created or found. The following list is what is needed to achieve that functionality.

***3.5.2.3.1*** The user shall have the option to edit the workout.  
***3.5.2.3.2*** Selecting edit workout shall redirect the user back to the create workout page with all the workout information filled in.  
***3.5.2.3.3*** The user shall have the option to immediately start the workout.  
***3.5.2.3.4*** Selecting start workout shall redirect the user to the workout in progress page.  
***3.5.2.3.5*** The user shall have the option to delete a previously saved workout.

**3.5.2.4 Workout in Progress Page**

Workout in progress is a page within the app that displays the current workout in progress. The following list is what is needed to achieve that functionality.

***3.5.2.4.1*** The workout in progress page shall guide the user through their workout.  
***3.5.2.4.2*** The workout in progress page shall display the workout information for the selected workout.

**3.5.3 Usability Requirements**

This section explains the usability and quality of the use requirements and objectives for the myJym application including measurable effectiveness and satisfaction criteria for the my workout feature.

***3.5.3.1***The user shall be able to create a workout.  
***3.5.3.2***The user shall be able to find a workout.  
***3.5.3.3***The user shall be able to save a workout.  
***3.5.3.4***The user shall be able to start a workout.  
***3.5.3.5***The user shall be able to see workout progress.

**3.5.4 Performance Requirements**

For the myJym application to meet performance standards there are specific requirements that need to be met. These requirements are specified below.

***3.5.4.1*** The user shall only be able to save one video link for a specific exercise.  
***3.5.4.2*** The system shall support a configurable amount of workouts.  
***3.5.4.3*** The system shall support a configurable amount of tags per workout.

**3.5.5 Logical Database Requirements**

The myJym database shall store all the tables that are necessary for storing information required by the application.

***3.5.5.1*** The workout table shall contain an exercise title column.  
***3.5.5.2*** The workout table shall contain a tag column.  
***3.5.5.3*** The workout table shall contain a rep column.  
***3.5.5.4*** The workout table shall contain a set column.  
***3.5.5.5*** The workout table shall contain a description column.  
***3.5.5.6*** The workout table shall contain a video link column.  
***3.5.5.7*** The workout table shall contain a workout id column that acts as a \*\*primary key\*\*.

**3.5.6 Design Constraints**

This section specifies the constraints with the myJym application that would arise from project limitations, external standards, and specific requirements.

***3.5.6.1*** The user shall only be able to add approved video links from a video provider to an exercise.  
***3.5.6.2*** The user shall only be able to input an integer into the rep field.  
***3.5.6.3*** The user shall only be able to input an integer into the set field.  
***3.5.6.4*** The user shall have to add a title to the workouts they create.

**3.5.7 Software System Attributes**

This section specifies the required attributes of the myJym application explaining the reliability, security, and maintainability.

**3.5.7.1 Reliability**

Specifies factors required to establish reliability of the myJym application at the time of delivery.

***3.5.7.1.1*** The system shall exit immediately when user finds an error.  
***3.5.7.1.2*** The app shall check that locally stored user information matches what is stored in database.

**3.5.7.2 Security**

Specifies the factors required to guarantee a defined level of availability for the myJym application in regards to the workout videos.

***3.5.7.2.1*** The user shall only be able to add approved video links from a video provider to an exercise.

**3.5.7.3 Maintainability**

Specifies the factors required to define a level of maintainability for the myJym application.

***3.5.7.3.1*** App administrators shall be able to remove **inappropriate** content.

**3.8.3 Supporting Information**

Not applicable for the my workout feature.

**4.0 Validation**

This section will validate that all features from section 3 will be in working order.

**4.1.1 External Interfaces Verification**

This section will define the verification for the external interfaces for the myTrainer section.

***4.1.1.1*** Verify the My trainer portion is a page accessible through the homepage. [3.1.1.1] (Technique: demonstration)

**4.1.1.1 Google Authentication Verification**

Verify the google authentication.

***4.1.1.1.1*** Verify that google authentication API shall be used by users to login to their accounts. [3.1.1.1.1] (Technique: demonstration)   
***4.1.1.1.2*** Verify that all users shall be able to login to their account using a google account. [3.1.1.1.2] (Technique: demonstration)   
***4.1.1.1.3*** Verify that all users shall be able to create an account using a google google account. [3.1.1.1.3] (Technique: demonstration)

**4.1.2 Functions Verification**

This section will verify all the functions of the my trainer page.

**4.1.2.1 Notifications Verification**

This section defines the verification for notifications.

***4.1.2.1.1*** Verify the myJym application shall send users Notifications. [3.1.2.1.1] (Technique: demonstration)   
***4.1.2.1.2*** Verify the myTrainer page shall allow users to view trainer profiles. [3.1.2.1.2] (Technique: demonstration)

**4.1.2.2 View Trainer Profile Verification**

This section defines the verification for viewing trainer profiles.

***4.1.2.2.1*** Verify the myTrainer page shall allow users to search for trainers. [3.1.2.2.1] (Technique: demonstration)   
***4.1.2.2.2*** Verify the myTrainer page shall allow users to update their profile. [3.1.2.2.2] (Technique: demonstration)   
***4.1.2.2.3*** Verify the myTrainer page shall allow users to message trainers. [3.1.2.2.3] (Technique: demonstration)   
***4.1.2.2.4*** Verify the myTrainer page shall allow trainers to message users. [3.1.2.2.4] (Technique: demonstration)   
***4.1.2.2.5*** Verify a notification shall be sent to the user’s devices when a user sets a time for their workout. [3.1.2.2.5] (Technique: demonstration)   
***4.1.2.2.6*** Verify a notification shall be sent to the user’s device when the user receives a message. [3.1.2.2.6] (Technique: demonstration)   
***4.1.2.2.7*** Verify trainer profile shall be viewable by gymgoers. [3.1.2.2.7] (Technique: demonstration)   
***4.1.2.2.8*** Verify trainer profile shall display a picture of the trainer. [3.1.2.2.8] (Technique: demonstration)   
***4.1.2.2.9*** Verify trainer profile shall display a short biography of the trainer. [3.1.2.2.9] (Technique: demonstration)   
***4.1.2.2.10*** Verify trainer profile shall display a link to the trainer’s YouTube channel. [3.1.2.2.10] (Technique: demonstration)   
***4.1.2.2.11*** Verify “View Workouts” shall be an option on the Trainer profile that allows the user to view a trainer’s created workouts. [3.1.2.2.11] (Technique: demonstration)   
***4.1.2.2.12*** Verify “Message Trainer” shall be an option on the Trainer profile that allows the user to send a message to the trainer. [3.1.2.2.12] (Technique: demonstration)   
***4.1.2.2.13*** Verify trainer profiles shall store a trainers first name [3.1.2.2.13] (Technique: demonstration)   
***4.1.2.2.14*** Verify trainer profiles shall store a trainer’s last name. [3.1.2.2.14] (Technique: demonstration)   
***4.1.2.2.15*** Verify trainer profiles shall store a trainer’s email. [3.1.2.2.15] (Technique: demonstration)   
***4.1.2.2.16*** Verify the system shall create a unique ID for each trainer. [3.1.2.2.16] (Technique: demonstration)   
***4.1.2.2.17*** Verify the system shall create a personal key for each trainer. [3.1.2.2.17] (Technique: demonstration)

**4.1.2.3 Update Profile Verification**

This section defines the verification for updating profiles.

***4.1.2.3.1*** Verify update Profile shall allow any user to update their own picture. [3.1.2.3.1] (Technique: demonstration)   
***4.1.2.3.2*** Verify update Profile shall allow any user to update a biography text-field. [3.1.2.3.2] (Technique: demonstration)   
***4.1.2.3.3*** Verify update Profile shall allow any user to update their personal YouTube link. [3.1.2.3.3] (Technique: demonstration)   
***4.1.2.3.4*** Verify update Profile shall allow any user to update their first and last name. [3.1.2.3.4] (Technique: demonstration)

**4.1.2.4 Search Trainer Verification**

This section defines the verification for searching trainers.

***4.1.2.4.1*** Verify the system shall have a search bar. [3.1.2.4.1] (Technique: demonstration)   
***4.1.2.4.2*** Verify a search bar shall be viewable by the user. [3.1.2.4.2] (Technique: demonstration)   
***4.1.2.4.3*** Verify search trainer shall allow a user to search for a trainer by name. [3.1.2.4.3] (Technique: demonstration)   
***4.1.2.4.4*** Verify search trainer shall allow a user to search for a trainer by location. [3.1.2.4.4] (Technique: demonstration)   
***4.1.2.4.5*** Verify search trainer shall allow a user to search for a trainer by workouts. [3.1.2.4.5] (Technique: demonstration)

**4.1.2.5 Message Trainer Verification**

This section defines the verification for messaging trainers.

***4.1.2.5.1*** Verify message Trainer shall give a user the ability to send messages to a trainer. [3.1.2.5.1] (Technique: demonstration)   
***4.1.2.5.2*** Verify message Trainer shall list messages in chronological order. [3.1.2.5.2] (Technique: demonstration)   
***4.1.2.5.3*** Verify messages Trainer shall be able to handle messages containing text. [3.1.2.5.3] (Technique: demonstration)   
***4.1.2.5.4*** Verify message Trainer shall be able to handle messages containing links. [3.1.2.5.4] (Technique: demonstration)

**4.1.2.6 Message Users Verifictaion**

This section defines the verification for messaging users.

***4.1.2.6.1*** Verify message users shall give trainers the ability to send messages to users. [3.1.2.6.1] (Technique: demonstration)   
***4.1.2.6.2*** Verify message users shall create a chat between trainer and users. [3.1.2.6.2] (Technique: demonstration)   
***4.1.2.6.3*** Verify message users shall list messages in chronological order. [3.1.2.6.3] (Technique: demonstration)   
***4.1.2.6.4*** Verify messages Trainer shall be able to handle messages containing text. [3.1.2.6.4] (Technique: demonstration)   
***4.1.2.6.5*** Verify message Trainer shall be able to handle messages containing links [3.1.2.6.5] (Technique: demonstration)

**4.1.3 Usability Requirements Verification**

This section defines the verification for usability requirements.

***4.1.3.1*** Verify the system is easy to learn for both novices and users with experience from similar systems through user testing. [3.1.3.1] (Technique: demonstration)   
***4.1.3.2*** Verify users familiar with similar systems is able to create a new account with minimal errors through user testing. [3.1.3.2] (Technique: demonstration)   
***4.1.3.3*** Verify the system is designed to be simple to use for frequent users through user testing. [3.1.3.3] (Technique: demonstration)   
***4.1.3.4*** Verify that frequent users are given an option to be able to access their account without having to sign in each session. [3.1.3.4] (Technique: demonstration)   
***4.1.3.5*** Verify tasks are able to be completed by the frequent user quickly and efficiently through user testing. [3.1.3.5] (Technique: demonstration)   
***4.1.3.6*** Verify the system is clear and well defined through user surveying. [3.1.3.6] (Technique: demonstration)   
***4.1.3.7*** Verify users experienced with this system are able to login to their account quickly(within 5 seconds of opening the app) through user testing. [3.1.3.7] (Technique: demonstration)   
***4.1.3.8*** Verify a new user is able to navigate to any desired option quickly and efficiently through user testing. [3.1.3.8] (Technique: demonstration)   
***4.1.3.9*** Verify the system receives a high rating by test groups. [3.1.3.9] (Technique: demonstration)   
***4.1.3.10*** Verify task failure by the system is kept to a minimum through user testing. [3.1.3.10] (Technique: demonstration)   
***4.1.3.11*** Verify the system is able to allow users to try things and undo them through user testing. [3.1.3.11] (Technique: demonstration)   
***4.1.3.12*** Verify the ability to undo is incorporated into any part of the system that stores information, including tasks. [3.1.3.12] (Technique: demonstration)   
***4.1.3.13*** Verify the system is created with elements that ensure that it is accessible to adults ranging from 18-40. [3.1.3.13] (Technique: demonstration)

**4.1.4 Performance Requirements Verification**

This section defines the verification performance requirements.

***4.1.4.1***Verify that my trainer page shall load when the user navigates to my trainer page. [3.1.4.1] (Technique: test)   
***4.1.4.2***Verify that the trainer shall receive messages soon after the author sends the message. [3.1.4.2] (Technique: test)   
***4.1.4.3***Verify that the trainer shall be able to send messages after clicking the send button in the built-in messenger. [3.1.4.3] (Technique: test)   
***4.1.4.4***Verify that notifications for the trainer shall be received at the same time as when the message comes in. [3.1.4.4] (Technique: test)   
***4.1.4.5***Verify that notifications from the trainer shall be received at the same time the recipient receives the message. [3.1.4.5] (Technique: test)   
***4.1.4.6***Verify that the trainer profiles on the web shall be loaded soon after the settings button has been clicked. [3.1.4.6] (Technique: test)   
***4.1.4.7***Verify that the trainer profiles on the mobile app shall be loaded soon after the settings button has been clicked. [3.1.4.7] (Technique: test)   
***4.1.4.8***Verify that the trainer search results on the web shall load up soon after the search has been entered in. [3.1.4.8] (Technique: test)   
***4.1.4.9***Verify that the trainer search results on the mobile app shall load up soon after the search has been entered in. [3.1.4.9] (Technique: test)

**4.1.5 Logical Database Requirements Verification**

This will verify all of the items for logical databases.

**4.1.5.1 Types of information used by various functions verification**

This section defines the verification for types of information used by various functions.

***4.1.5.1.1*** Verify MyTrainer shall store a trainers first name [3.1.5.1.1] (Technique: demonstration)   
***4.1.5.1.2*** Verify MyTrainer shall store a trainer’s last name. [3.1.5.1.2] (Technique: demonstration)   
***4.1.5.1.3*** Verify MyTrainer shall store a trainer’s email. [3.1.5.1.3] (Technique: demonstration)   
***4.1.5.1.4*** Verify The system shall create a trainer\_ID for each trainer. [3.1.5.1.4] (Technique: demonstration)

**4.1.5.2 Frequency of use verification**

This section defines the verification for grequency of use.

***4.1.5.2.1*** Verify Email data types shall be used for a single account. [3.1.5.2.1] (Technique: demonstration)

**4.1.5.3 Accessing Capabilities verification**

This section defines the verification for accessing capabilities.

***4.1.5.3.1*** Verify Trainers shall have access to User first\_name. [3.1.5.3.1] (Technique: demonstration)   
***4.1.5.3.2*** Verify Trainers shall have access to User last\_name. [3.1.5.3.2] (Technique: demonstration)   
***4.1.5.3.3*** Verify Trainers shall have access to User goals. [3.1.5.3.3] (Technique: demonstration)   
***4.1.5.3.4*** Verify Trainers shall have access to User W/O\_plan. [3.1.5.3.4] (Technique: demonstration)   
***4.1.5.3.5*** Verify Trainers shall have access to User gym. [3.1.5.3.5] (Technique: demonstration)

**4.1.5.4 Integrity Constraints verificaion**

This section defines the verification for integrity constraints.

***4.1.5.4.1*** Verify Data entries shall be possible entries. [3.1.5.4.1] (Technique: demonstration)   
***4.1.5.4.2*** Verify Impossible data entries shall not be allowed. [3.1.5.4.2] (Technique: demonstration)   
***4.1.5.4.3*** Verify Data entries shall be verified to ensure it is possible. [3.1.5.4.3] (Technique: demonstration)   
***4.1.5.4.4*** Verify The database shall have entity integrity. [3.1.5.4.4] (Technique: demonstration)

**4.1.5.5 Security verification**

This section defines the verification for security.

***4.1.5.5.1*** Verify Trainer accounts shall utilize google sign-in to access their account. [3.1.5.5.1] (Technique: demonstration)   
***4.1.5.5.2*** Verify User accounts shall utilize google sign-in to access their account. [3.1.5.5.2] (Technique: demonstration)   
***4.1.5.5.3*** Verify Trainers shall only have one account. [3.1.5.5.3] (Technique: demonstration)   
***4.1.5.5.4*** Verify Gymgoers shall only have one account. [3.1.5.5.4] (Technique: demonstration)   
***4.1.5.5.5*** Verify Admin accounts shall have access to all accounts [3.1.5.5.5] (Technique: demonstration)

**4.1.6 Design Constraints Requirements Verification**

This section will verify the design constraints requirements.

***4.1.6.1*** Verify system users shall specify the requirements for myTrainer and ensure to understand these requirements meet their demands and needs. [3.1.6.1] (Technique: demonstration)   
***4.1.6.2*** Verify users shall specify changes to the requirements. [3.1.6.2] (Technique: demonstration)   
***4.1.6.3*** Verify system managers shall use the requirements documentation to plan a bid for the system to plan the system development process for myTrainer. [3.1.6.3] (Technique: demonstration)   
***4.1.6.4*** Verify system engineers shall use the requirements to understand what system is being developed. [3.1.6.4] (Technique: demonstration)   
***4.1.6.5*** Verify System testing engineers shall use the requirements for the purpose of development of validation tests for the system of myTrainer. [3.1.6.5] (Technique: demonstration)   
***4.1.6.6*** Verify System maintaince managers shall use requirements to understand the system and the relationship between its parts. [3.1.6.6] (Technique: demonstration)

**4.1.7 Reliability Verification**

This section defines the verification for reliability.

***4.1.7.1*** Verify applications shall not have errors within the codes to cause applications to close or crash. [3.1.7.1] (Technique: inspection)   
***4.1.7.2*** Verify there will be functions within the software that perform verifications to notice errors before the application is crashed. [3.1.7.2] (Technique: inspection)   
***4.1.7.3*** Verify the system shall verify the information that is match between the server and the application. [3.1.7.3] (Technique: inspection)   
***4.1.7.4*** Verify user information shall be available to users and other users that have access to specific user accounts. [3.1.7.4] (Technique: inspection)   
***4.1.7.5*** Verify users shall be required to provide an email and password during the account creation process. [3.1.7.5] (Technique: inspection)   
***4.1.7.6*** Verify user’s account and personal information shall be hidden from all other unauthorized users and parties. [3.1.7.6] (Technique: inspection)   
***4.1.7.7*** Verify the user account creation process shall be safe and secure. [3.1.7.7] (Technique: inspection)   
***4.1.7.8*** Verify that the login process to user’s personal accounts is secure. [3.1.7.8] (Technique: inspection)   
***4.1.7.9*** Verify that once a user logs out from their personal account, all user information shall be removed from the login screen and cannot be accessed without user login with their personal username and password. [3.1.7.9] (Technique: inspection)   
***4.1.7.10*** Verify users shall be allowed to report to the administration team during circumstances such as when the user wishes to reset their password. [3.1.7.10] (Technique: inspection)   
***4.1.7.11*** Verify that a password resetting email shall be sent to the user’s email address that is provided by the user during the account creation process. [3.1.7.11] (Technique: inspection)   
***4.1.7.12*** Verify system database shall store user’s information. [3.1.7.12] (Technique: inspection)   
***4.1.7.13*** Verify myTrainer application shall run on both android and iOS mobile devices. [3.1.7.13] (Technique: inspection)

???

**4.2.1 Map External Interfaces Verification**

This section defines the verification for the gym map external interface requirements.

***4.2.1.1*** Verify that the map utilizes the System Database to generate the map. (Technique: Inspection) [3.2.1.1]   
***4.2.1.2*** Verify that the map search bar obtains user data from the System Database. (Technique: Inspection) [SRS 3.2.1.2]

**4.2.2 Map Functions Verification**

This section defines the verification for the gym map functions.

***4.2.1.1*** Verify the map organizes gym equipment according to a layout file. (Technique: Inspection) [3.2.2.1.1]   
***4.2.1.2*** Verify the map organizes gym services according to a layout file. (Technique: Inspection) [3.2.2.1.2]   
***4.2.1.3*** Verify the map properly refreshes when an error occurs. (Technique: Test) [3.2.2.1.2]   
***4.2.1.4*** Verify an administrator can modify the map layout. (Technique: Demonstration) [3.2.2.1.4]   
***4.2.1.5*** Verify a map tutorial opens when the map is first opened. (Technique: Demonstration) [3.2.2.1.5]   
***4.2.2.1*** Verify that the map search bar performs validation on inputs matching them to gym functions. (Technique: Inspection)[3.2.2.2.1]   
***4.2.2.2*** Verify that the search bar has functionality to locate gym functions. (Technique: Demonstration) [3.2.2.2.2]   
***4.2.2.3*** Verify that the map search bar is a modeless interface allowing for interaction with other page elements while it is open. (Technique: Inspection) [3.2.2.2.3]   
***4.2.2.4*** Verify that facilities in the gym are searchable by name through the map search bar. (Technique: Demonstration) [3.2.2.2.4]   
***4.2.2.1*** Verify that the map has a scrollable edge. (Technique: Demonstration) [3.2.2.3.1]   
***4.2.3.2*** Verify that the map supports zooming, detecting when a zoom request is made. (Technique: Demonstration) [3.2.2.3.2]   
***4.2.3.3*** Verify that gym function icons are interactable to view options related to that function. (Technique: Demonstration) [3.2.2.3.3]

**4.2.3 Map Usability Requirements Verification**

This section defines the verification for the gym map usability requirements.

***4.2.3.1*** Verify the map is intuitive so the user can easily form a mental model of its workings. (Technique: Inspection) [3.2.3.1]   
***4.2.3.2*** Verify a gym goer can quickly identify where they should be in the gym to complete their desired activity. (Technique: Demonstration) [3.2.3.2]   
***4.2.3.3*** Verify the map icons and interfaces use simple command formats to ensure users understand map functionalities. (Technique: Inspection) [3.2.3.3]   
***4.2.3.4*** Verify when a task failure occurs the map stops its operation and refreshes. (Technique: Test) [3.2.3.4]   
***4.2.3.5*** Verify map iconography is consistent with commonly recognized symbols. (Technique: Inspection) [3.2.3.5]

**4.2.4 Map Performance Requirements Verification**

This section defines the verification for the gym map performance requirements.

***4.2.4.1*** Verify the map prioritizes loading the graphical display elements prior to lower priority tasks. (Technique: Inspection) [3.2.4.1]   
***4.2.4.2*** Verify the map search functionality returns results in higher algorithm efficiency than\*\* O\*\*(n log n). (Technique: Test) [3.2.4.2]   
***4.2.4.3*** Verify the map allows for interaction with elements after a predetermined time post loading. (Technique: Test) [3.2.4.3]

**4.2.5 Map Logical Database Requirements Verification**

This section defines the verification for the gym map logical database requirements.

***4.2.5.1*** Verify the map icons are loaded into memory from the \*\*System Database\*\* when the view is selected. (Technique: Test) [3.2.5.2.1]   
***4.2.5.2*** Verify the map search functionality is accessible on demand. (Technique: Demonstration) [3.2.5.2.2]   
***4.2.5.3*** Verify that the map is accessible from the homepage of the myJym application. (Technique: Demonstration) [3.2.5.3.1]   
***4.2.5.4*** Verify the map has access to the \*\*System Database\*\* to access gym equipment locations. (Technique: Inspection) [3.2.5.3.2]   
***4.2.5.5*** Verify that the map has access to the \*\*System Database\*\* to return search results. (Technique: Inspection) [3.2.5.3.3]   
***4.2.5.6*** Verify that the map handles data for the exercise machine locations. (Technique: Inspection) [3.2.5.4.1]   
***4.2.5.6*** Verify that the map handles data for the exercise machine images. (Technique: Inspection) [3.2.5.4.2]   
***4.2.5.6*** Verify that the map handles data for the map search bar results. (Technique: Inspection) [3.2.5.4.3]   
***4.2.5.7*** Verify that map icons are entries to the database. (Technique: Inspection) [3.2.5.5.1]   
***4.2.5.8*** Verify that map data is formatted to match the conventions of the database. (Technique: Inspection) [3.2.5.5.2]   
***4.2.5.9*** Verify that data used for the map is only accessible by the functions of myJym. (Technique: Test) [3.2.5.6.1]   
***4.2.5.10*** Verify that the interactable map features provides sufficient \*\*Pixels Per Inch\*\* that the average human is able to cleanly select them. (Technique: Inspection) [3.2.6.1]   
***4.2.5.11*** Verify that the map is available when the application has been opened to the home page. (Technique: Demonstration) [3.2.7.1]   
***4.2.5.12*** Verify that the map is available offline when no internet connectivity is available. (Technique: Demonstration) [3.2.7.2]   
***4.2.5.13*** Verify that the maps delivered to the user on MyJym delivers accurate data based on the information in the \*\*System Database.\*\* (Technique: Inspection) [3.2.7.1.1]   
***4.2.5.14*** Verify that the MyJym map is retained in memory while the myJym application is focused on a separate page. (Technique: Test) [3.2.7.1.2]   
***4.2.5.15*** Verify that the map only be editable by an admin. (Technique: Inspection) [3.2.7.2.1]

**4.4 Login & Homepage Verification**

Verify the Login & Homepage requirements

**4.4.1 External Interfaces Verification**

Verification of the interfaces that the application will output to the user.   
***4.4.1.1*** Verify that the login page shall provide the ability to login to an account using login credentials. [3.4.1.1]  
Method: Test  
What: Attempt to login to an account using both valid and invalid credentials and verify that the happy path is achieved.  
***4.4.1.2*** Verify that the system shall send an automated password recovery email to the user on request. [3.4.1.2]   
Method: Test  
What: Attempt to recover a password and confirm that the recovery email is sent.

**4.4.2 Functions Verification**

Verification of the different functionalities within the application.  
***4.4.2.1*** Verify that the home page shall provide access to myTrainer. [3.4.2.1]  
Method: Test  
What: The application shall take the user to the myTrainer page after the myTrainer button has been selected while on the home page.  
***4.4.2.2*** Verify that the home page shall provide access to myWorkouts. [3.4.2.2]  
Method: Test  
What: The application shall take the user to the myWorkouts page after the myWorkouts button has been pressed while on the home page.  
***4.4.2.3*** Verify that the home page shall provide access to myVideos. [3.4.2.3]  
Method: Test  
What: The application shall take the user to the myVideos page after the myVideos button has been pressed while on the home page.  
***4.4.2.4*** Verify that the home page shall provide access to myGymMap. [3.4.2.4]  
Method: Test  
What: The application shall take the user to the myGymMap page after the myGymMap button has been while on the home page.  
***4.4.2.5*** Verify that the login page shall provide the user with the ability to recover a lost password. [3.4.2.5]  
Method: Test  
What: The application shall provide the user with a temporary password when requested. The password will go through email and shall only become the current active password when it is used on the app after the user views it in their email. The user will then be required to set a new password immediately after logging in.  
***4.4.2.6*** Verify that the login page shall provide the user with the ability to create an account. [3.4.2.6]  
Method: Test  
What: The application shall allow the user to create an account with their individual credentials and access the apps features.  
***4.4.2.7*** Verify that the login page shall display an error message when invalid credentials are submitted. [3.4.2.7]  
Method: Test  
What: When a user attempts to log in with invalid credentials, the login page shall display an error message that the credentials were invalid and re-prompt for valid credentials.  
***4.4.2.8*** Verify that the system shall check the login credentials against the database. [3.4.2.8]  
Method: Test  
What: The user shall only be able to login and view the home page after they provide valid login credentials.

**4.4.3 Usability Requirements Verification**

Verification of the requirements related to usability.  
***4.4.3.1*** Verify that the signup page shall inform the user of password requirements. [3.4.3.1]  
Method: Test  
What: When creating an account verify that the message for password requirements appears.  
***4.4.3.2*** Verify that the login page shall allow the user to enter login credentials [3.4.3.2]  
Method: Test  
What: Attempt to enter login credentials on the login page.

**4.4.4 Performance Requirements Verification**

Verification of the system's ability to process user requests.  
***4.4.4.1*** Verify that the system shall have the ability to process login credentials. [3.4.4.1]  
Method: Test  
What: The system shall store the login credentials by allowing the user to re-login multiple times.

**4.4.5 Logical Database Requirements Verification**

Verification of the operation of the database.  
***4.4.5.1*** Verify that the UserData database table shall have a Google authentication key from the Google third party login. [3.4.5.1]  
Method: Inspection  
What: Inspect the UserData table to ensure that it can collect and store google authentication keys.  
***4.4.5.2*** Verify that the UserData database table shall store usernames [3.4.5.2]  
Method: Inspection  
What: Inspect the UserData table to ensure that it can collect and store usernames.  
***4.4.5.3*** Verify that the UserData database table shall store the passwords [3.4.5.3]  
Method: Inspection  
What: Inspect the UserData table to ensure that it can collect and store passwords.  
***4.4.5.4*** Verify that the database shall be accessed by the login page when user credentials are submitted through the login page. [3.4.5.4]  
Method: Test  
What: Have the tester attempt to login with a valid and invalid credentials. Check if the system allows valid credentials to login and doesn't allow invalid credentials to login.  
***4.4.5.5*** Verify that the database shall keep the login credentials of users confidential. [3.4.5.5]  
Method: Test  
What: Have a "hacker" try to access personal information in the database. Run this test until the database is secure from all attacks.

**4.4.6 Design Constraints Verification**

Verification of the metrics set to limit the application.  
***4.4.6.1*** Verify that the myJym app shall be completed in its entirety by the end of Jared's Senior Project class. [3.4.6.1]  
Method: Demo  
What: Show that the myJym app is completed within the given timeframe.  
***4.4.6.2*** Verify that the budget for the myJym app shall be $0.00. [3.4.6.2]  
Method: Demo  
What: Show that there has been no money spent on the development of the application.

**4.4.7 Software System Attributes Verification**

Verification of the software system attribute requirements

**4.4.7.1 Reliability Verification**

Verification of the dependability of the application.  
***4.4.7.1.1*** Verify that the app shall require the compliance of personal trainers. [3.4.7.1.1]  
Method: Demo  
What: Demonstrate the compliance of personal trainers in the use of the application.  
***4.4.7.1.2*** Verify that the app shall require the adminitrator. [3.4.7.1.2]  
Method: Demo  
What: Demonstrate the administrator requirements in the use of the application.

**4.4.7.2 Availability Verification**

Verification of the systems uptime for users.  
***4.4.7.2.1*** Verify that the myJym app shall be available on demand of the user. [3.4.7.2.1]  
Method: Demo  
What: Access the myJym application at various times to ensure that it is available whenever needed.

**4.4.7.3 Security Verification**

Verification of the applications ability to protect user assets.  
***4.4.7.3.1*** Verify that the system shall encrypt user accounts. [3.4.7.3.1]  
Method: Inspection  
What: Inspect the code to ensure that the user account information is properly encrypted.  
***4.4.7.3.2*** Verify that the system shall use a protocol to secure online communications. [3.4.7.3.2]  
Method: Inspection  
What: Inspect code to ensure that a protocol is properly used to secure online communication.

**4.4.7.4 Portability Verification**

Verification of the applications ability to work on different operating systems.  
***4.4.7.4.1*** Verify that the application shall be built with a programming language that can be ported to both iOS and Android. [3.4.7.4.1]  
Method: Inspection  
What: Inspect the code to ensure that it can be ported to iOS and Android.

**???**

**4.5.1 External Interface Verification**

This section defines the verification for external interfaces for the myJym application.

**4.5.1.1 Database Verification**

This section defines the verification for the database for the my workout function.

***4.5.1.1.1*** Verify that the database shall store workout **title**. [3.5.1.1.1]  
***4.5.1.1.2*** Verify that the database shall store workout video links. [3.5.1.1.2]  
***4.5.1.1.3*** Verify that the database shall store workout description. [3.5.1.1.3]  
***4.5.1.1.4*** Verify that the database shall store workout **sets**. [3.5.1.1.4]  
***4.5.1.1.5*** Verify that the database shall store workout **reps**. [3.5.1.1.5]  
***4.5.1.1.6*** Verify that the database shall store workout **tags**. [3.5.1.1.6]

**4.5.1.2 Video Provider Verification**

This section defines the verification for the video provider integration for the my workout function.

***4.5.1.2.1*** Verify that the **gymgoers** shall have the ability to access workout videos. [3.5.1.2.1]

**4.5.2 Functions Verification**

This section defines the verification for functionality for the my workout feature.

**4.5.2.1 Create Workout Verification**

This section defines the verification for the create workout function for the my workout feature.

***4.5.2.1.1*** Verify that the workout creation shall allow the **user** to specify a workout name. [3.5.2.1.1] (Technique: demonstration)  
***4.5.2.1.2*** Verify that the workout creation shall allow the user to specify a muscle group to workout. [3.5.2.1.2] (Technique: demonstration)  
***4.5.2.1.3*** Verify that the workout creation shall allow the user to specify an amount of **reps**. [3.5.2.1.3] (Technique: demonstration)  
***4.5.2.1.4*** Verify that the workout creation shall allow the user to specify an amount of **sets**. [3.5.2.1.4] (Technique: demonstration)  
***4.5.2.1.5*** Verify that the workout creation shall allow the user to embed a video link. [3.5.2.1.5] (Technique: demonstration)

**4.5.2.2 Find Workout Verification**

This section defines the verification for the find workout function for the my workout feature.

***4.5.2.2.1*** Verify that the find workout page shall have the ability to search by predetermined **tag**s. [3.5.2.2.1] (Technique: demonstration)  
***4.5.2.2.2*** Verify that one of the tags that can be used to search by shall be **author**. [3.5.2.2.2] (Technique: demonstration)  
***4.5.2.2.3*** Verify that one of the tags that can be used to search by shall be **name of workout**. [3.5.2.2.3] (Technique: demonstration)  
***4.5.2.2.4*** Verify that one of the tags that can be used to search by shall be **length**. [3.5.2.2.4] (Technique: demonstration)  
***4.5.2.2.5*** Verify that one of the tags that can be used to search by shall be **workout type**. [3.5.2.2.5] (Technique: demonstration)  
***4.5.2.2.6*** Verify that one of the tags that can be used to search by shall be **difficulty**. [3.5.2.2.6] (Technique: demonstration)  
***4.5.2.2.7*** Verify that one of the tags that can be used to search by shall be **equipment type**. [3.5.2.2.7] (Technique: demonstration)  
***4.5.2.2.8*** Verify that one of the tags that can be used to search by shall be **muscle groups**. [3.5.2.2.8] (Technique: demonstration)  
***4.5.2.2.9*** Verify that the user shall have the ability to save a workout they find to their saved workouts list. [3.5.2.2.9] (Technique: demonstration)

**4.5.2.3 Saved Workout Verification**

This section defines the verification for the saved workout function for the my workout feature.

***4.5.2.3.1*** Verify that the user shall have the option to edit the workout. [3.5.2.3.1] (Technique: demonstration)  
***4.5.2.3.2*** Verify that selecting edit workout shall redirect the user back to the create workout page with all the workout information filled in. [3.5.2.3.2] (Technique: demonstration)  
***4.5.2.3.3*** Verify that the user shall have the option to immediately start the workout. [3.5.2.3.3] (Technique: demonstration)  
***4.5.2.3.4*** Verify that selecting start workout shall redirect the user to the workout in progress page. [3.5.2.3.4] (Technique: demonstration)  
***4.5.2.3.5*** Verify that the user shall have the option to delete a previously saved workout. [3.5.2.3.5] (Technique: demonstration)

**4.5.2.4 Workout in Progress Page Verification**

This section defines the verification for the workout in progress function for the my workout feature.

***4.5.2.4.1*** Verify that the workout in progress page shall guide the user through their workout. [3.5.2.4.1] (Technique: demonstration)  
***4.5.2.4.2*** Verify that the workout in progress page shall display the workout information for the selected workout. [3.5.2.4.2] (Technique: demonstration)

**4.5.2.5 My Video Page Verification**

This section defines the verification for the workout in progress function for the my workout feature.

***4.5.2.5.1*** Verify that the My Video page shall display a workout title. [3.3.2.1.1] (Technique: demonstration)  
***4.5.2.5.2*** Verify that the My Video page shall display an equipment name. [3.3.2.1.2] (Technique: demonstration)  
***4.5.2.5.3*** Verify that the My Video page shall display a workout video. [3.3.2.1.3] (Technique: demonstration)  
***4.5.2.5.4*** Verify that the My Video page shall provide a save video button. [3.3.2.1.4] (Technique: demonstration)

**4.5.3 Usability Requirements Verification**

This section defines the verification for the usability requirements for the my workout feature.

***4.5.3.1*** Verify that the user shall be able to create a workout. [3.5.3.1] (Technique: demonstration)  
***4.5.3.2*** Verify that the user shall be able to find a workout. [3.5.3.2] (Technique: demonstration)  
***4.5.3.3*** Verify that the user shall be able to save a workout. [3.5.3.3] (Technique: demonstration)  
***4.5.3.4*** Verify that the user shall be able to start a workout. [3.5.3.4] (Technique: demonstration)  
***4.5.3.5*** Verify that the user shall be able to see workout progress. [3.5.3.5] (Technique: demonstration)

**4.5.4 Performance Requirements Verification**

This section defines the verification for the performance requirements for the my workout feature.

***4.5.4.1*** Verify that the user shall only be able to save one video link for a specific exercise. [3.5.4.1] (Technique: test)  
***4.5.4.2*** Verify that the system shall support a configurable amount of workouts. [3.5.4.2] (Technique: test)  
***4.5.4.3*** Verify that the system shall support a configurable amount of tags per workout. [3.5.4.3] (Technique: test)

**4.5.5 Logical Database Requirements Verification**

This section defines the verification for the logical database requirements for the my workout feature.

***4.5.5.1*** Verify that the workout table shall contain an exercise title column. [3.5.5.1] (Technique: inspection)  
***4.5.5.2*** Verify that the workout table shall contain a tag column. [3.5.5.2] (Technique: inspection)  
***4.5.5.3*** Verify that the workout table shall contain a rep column. [3.5.5.3] (Technique: inspection)  
***4.5.5.4*** Verify that the workout table shall contain a set column. [3.5.5.4] (Technique: inspection)  
***4.5.5.5*** Verify that the workout table shall contain a description column. [3.5.5.5] (Technique: inspection)  
***4.5.5.6*** Verify that the workout table shall contain a video link column. [3.5.5.6] (Technique: inspection)  
***4.5.5.7*** Verify that the workout table shall contain a workout id column that acts as a **primary key**. [3.5.5.7] (Technique: inspection)

**4.5.6 Design Constraints Verification**

This section defines the verification for the design constraints for the my workout feature.

***4.5.6.1*** Verify that the user shall only be able to add approved video links from a video provider to an exercise. [3.5.6.1] (Technique: demonstration)  
***4.5.6.2*** Verify that the user shall only be able to input an integer into the rep field. [3.5.6.2] (Technique: demonstration)  
***4.5.6.3*** Verify that the user shall only be able to input an integer into the set field. [3.5.6.3] (Technique: demonstration)  
***4.5.6.4*** Verify that the user shall have to add a title to the workouts they create. [3.5.6.4] (Technique: demonstration)

**4.5.7 Software System Attributes Verification**

This section defines the verification for the software system attributes for the my workout feature.

**4.5.7.1 Reliability Verification**

This section defines the verification for the reliability of the my workout feature.

***4.5.7.1.1*** Verify that the system shall exit immediately when user finds an error. [3.5.7.1.1] (Technique: demonstration)  
***4.5.7.1.2*** Verify that the app shall check that locally stored user information matches what is stored in database. [3.5.7.1.2] (Technique: inspection)

**4.5.7.2 Security Verification**

This section defines the verification for the security of the my workout feature.

***4.5.7.2.1*** Verify that the user shall only be able to add approved video links from a video provider to an exercise. [3.5.7.2.1] (Technique: demonstration)

**4.5.7.3 Maintainability Verification**

This section defines the verification for the maintainability of the my workout feature.

***4.5.7.3.1*** Verify that the app administrators shall be able to remove **inappropriate** content. [3.5.7.3.1] (Technique: demonstration)

**4.8.3 Supporting Information Verification**

Not applicable for the my workout feature.

**References**

[1] S.R.O., Eccam. “Example Software Requirements Specification (SRS) ReqView Documentation.” ReqView, 2018, [www.reqview.com/doc/iso-iec-ieee-29148-srs-example](http://www.reqview.com/doc/iso-iec-ieee-29148-srs-example).

[2] IIEEE, " 29148-2018 - ISO/IEC/IEEE International Standard - Systems and software engineering -- Life cycle processes -- Requirements engineering " pp. 56-87, Jul. 2018. [Online] Available: <https://ieeexplore-ieee-org.byui.idm.oclc.org/document/8559686>

[3] “Macintosh Operating System (Mac OS).” Techopedia, techopedia, 2012, [www.techopedia.com/definition/2639/macintosh-operating-system-mac-os](http://www.techopedia.com/definition/2639/macintosh-operating-system-mac-os).

[4] <https://github.com/voyager1winterberry/cse372-01srs/blob/main/refDocs/Week10Assignments.docx>