Cade Dillon

ISTA 322 – Project Step 9

7 October 2018

YogNatomy Software Requirements Specifications (SRS)

# 1. Purpose

The purpose of this document is to provide a comprehensive outline of software requirements for the YogNatomy application. YogNatomy is an application that seeks to make Yoga practice intuitive, understandable, and accessible for practitioners of all skill levels. YogNatomy will be a web application built in the MVC framework and hosted with Microsoft Azure. The scope of this document includes database requirements, business logic requirements, and presentation layer requirements.

# 2. Product Description

YogNatomy is a web application that seeks to bring the practice of Yoga to all people, as a means of maintaining their health and wellness, or in the case of people with prior injuries, regaining it through rehabilitation. YogNatomy contains a database of Yoga poses that the user can browse through, keyword search, or view by collection. The information in the database will be presented via wikia style web pages. Other features include account registration, exercise completion history, exercise assignment from trainer users to trainee users, interactive navigation graphics, and new exercise requesting. Distinct users are: trainees, trainers, and database administrators.

# 3. System Requirements

## 3.1 Account Registration

The system shall allow both trainee users and trainer users to register accounts. The registration feature will be located on the home view, and users will enter identifying information such as email address, name, and date of birth. Users will create a unique user name and password. Users will also indicate some basic exercise history such as self-reported fitness level, Yoga experience, and any trainer information they have.

*Confirmation Testing –* User information will be stored into the YogNatomy database upon account registration, which may be confirmed with a database query to return new user information.

## 3.2 Log-In Functionality

The system shall allow returning users to access their account’s via entering their previously created user names and passwords. This will allow them to view their exercise history, manage their trainer-client relationships, send requests to the administrator for new exercises, and change their existing account information.

*Confirmation Testing –* The successful retrieval of stored user information via the log in feature would constitute a passed test. Testing should confirm user account information is persistent in the YogNatomy database.

## 3.3 Exercise Requesting

The system shall allow trainees and trainers to request the database administrator implement new exercises. This will be a simple messaging system, and users will provide the exercise name and a brief description. The database administrator will then see these messages as a notification upon his next login to the application.

*Confirmation Testing –* The successful test case of this feature would be the user being able to enter the request into the system, and then the database administrator successfully receiving it.

## 3.4 Exercise Assignment

The system shall allow the trainer users to assign exercises to their clients, and only their clients. The trainer will access this feature through a menu option in their profile management view. The trainee will receive notification of assigned exercises upon their next log in.

*Confirmation Testing –* The successful test case of this feature would be the trainer being able to view their client listing, select the assign exercise feature, assign the exercise, and then the user receiving notification of and being able to view the assigned exercise.

## 3.5 Browse Exercises

The system shall allow all users to browse through the exercises contained in the YogNatomy application. YogNatomy will feature a blended navigation system including browsing by hyperlinks, keyword searches, and the interactive muscle diagram.

*Confirmation Testing –* Test case will be considered to be passed if all exercise pages are reachable via all three methods of navigation.

## 3.6 Database Maintenance

The system shall allow the database administrator to add or change information in the database. These changes will be persistent across all versions of the database. These include: adding new exercises, updating information on existing exercises, and updating user profile information.

*Confirmation Testing –* The test case for this requirement will be to add or change some information in the database, and then check to ensure that the changes have persisted through the application.

# 4 User Interface MockUps

## 4.1 Home Controller View

## 4.2 Exercise Information View

## 4.3 Account Registration View