Software Requirements Specification

for

Project Codename Olympia

Version 4.0 approved

Prepared by Landen Marchand, Josh King, Andrew Unger

JAL Group

04/01/2018

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Landen Marchand	02/28/2018	Added sections [1.1-1.4 , 2.2]	1.0
Landen Marchand	03/14/2018	Added sections [2.1, Appendix A]	2.0
Landen Marchand	03/30/2018	Added Header Title. Edited 2.1 and 2.2. Added Appendix A and B, Section 1.5	3.0
Landen Marchand	04/15/2018	Added sections [2.4-2.7,][3,4,5 ALL], Edited [2.1, Appendix A,B]	4.0

1. Introduction

1.1 Purpose

The purpose of *Project Codename Olympia* (**PCO**) is to have a fully automated system for the various skating events in the *Winter Olympics* (**WO**) . This automated system takes care of registration for each team, also known as country, scoring for each team and individual, and general record keeping of the events, times, and teams.

1.2 Document Conventions

Major titles, such as the project name and the event itself are first written out completely and in italics. From there, they will be denoted by their respective acronym, which is provided immediately after the full name in bold.

1.3 Intended Audience and Reading Suggestions

The intended audience for this documentation comprises the following entities: the *Olympic League* (**OL**) which oversees and organizes all of the **WO** skating events, and the software development team constructing **PCO**. There are no special reading suggestions as of version 1.0 and should be read in a straightforward manner.

1.4 Product Scope

PCO manages the registration of individuals and teams, scheduling of events, allocation of qualified judges, and manages a running database of scores and standings for the **WO** skating events which, in turn, provides a fully automated system for the **OL**.

1.5 References

Appendix A: Documentation for Data Dictionary / Class Specifications
Appendix B: Visual of Class Diagrams / Use Cases and Use Case Diagram

2. Overall Description

2.1 Product Perspective

PCO is a replacement of humans to manually coordinate the **WO** skating events. This system will be self-contained automating the registration, scheduling, allocation of judges, and statistics of the **WO** skating events. **PCO** will contain subsystems consisting of registration, scheduling athletes/judges/events, events, and a database for an aggregation of information. The definitions of these subsystems can be found in the Data Dictionary file.

2.2 Product Functions

- Register countries as teams and individuals as athletes
- Allocate each event to a specific rink
- Allocate athletes to each event
- Assign judges to each event
- Record scores/times from each event
- Display event information, scores, and current standings

2.3 User Classes and Characteristics

Officials of the WO

Official Registrant - Registers teams and athletes
Official Scheduler - Handles scheduling of events, judges, and athletes
Official Judge - Verifies and processes scores
General Official - Display current information of the WO

2.4 Operating Environment

This system is meant to be used on a Windows OS to easily manage the database and GUI. Please use Windows 7 or higher for the pest performance or Windows Server 2003 or higher. SQLite library and C# compiler are needed. Visual Studio 2015 or 2017 is preferred for maintenance.

2.5 Design and Implementation Constraints

This system is meant to be used on a Windows OS to easily manage the database and GUI. The system is solely in American English language and would have to be modified if other languages were needed. All modifications to this system once purchased must be maintained by the **WO** committee. They are free to manipulate the system as desired.

2.6 User Documentation

PCO has been designed to be a very easy-to-use and undemanding system. Upon consideration of purchase for this product, there will be an initial presentation explicitly explaining what the system does and how to utilize said system. Since the system is rather rudimentary, there are no ancillary documentation in regards to operation. JALGroup will be your external source for any and all questions.

The GUI as well as the database has terse, yet ample instructions as well as descriptions. Navigation and entry of data include buttons, drop-down menus, and text boxes. Remember, only officials are utilizing this system, and it is expected that they are knowledgeable as to how to at least operate a program on a computer.

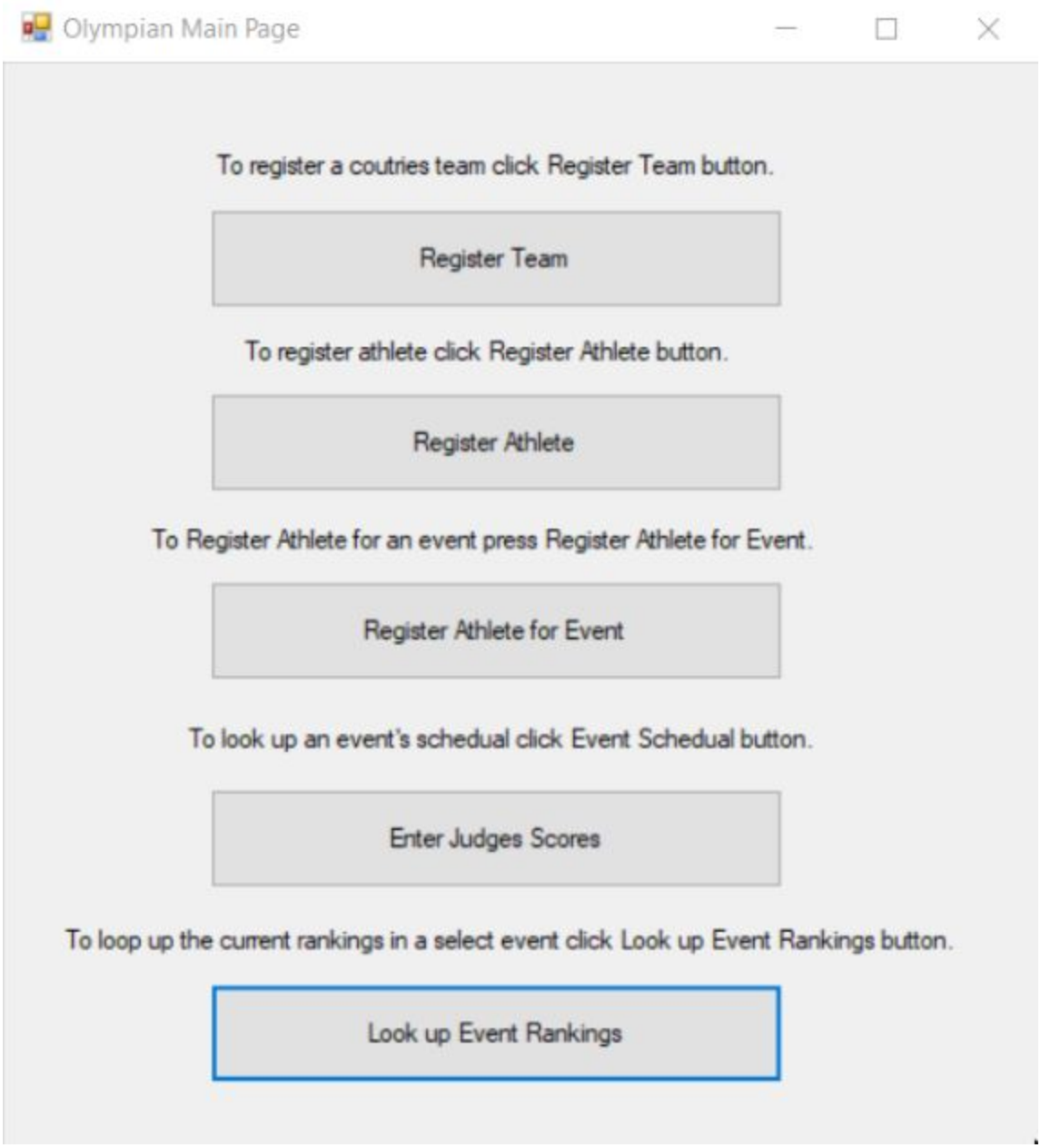
2.7 Assumptions and Dependencies

Please use a Windows OS for correct performance.
Please use SQL for a database and SQLite for the library.
This is a subsystem of the **WO**, such that officials are already logged in from a larger-scale system.

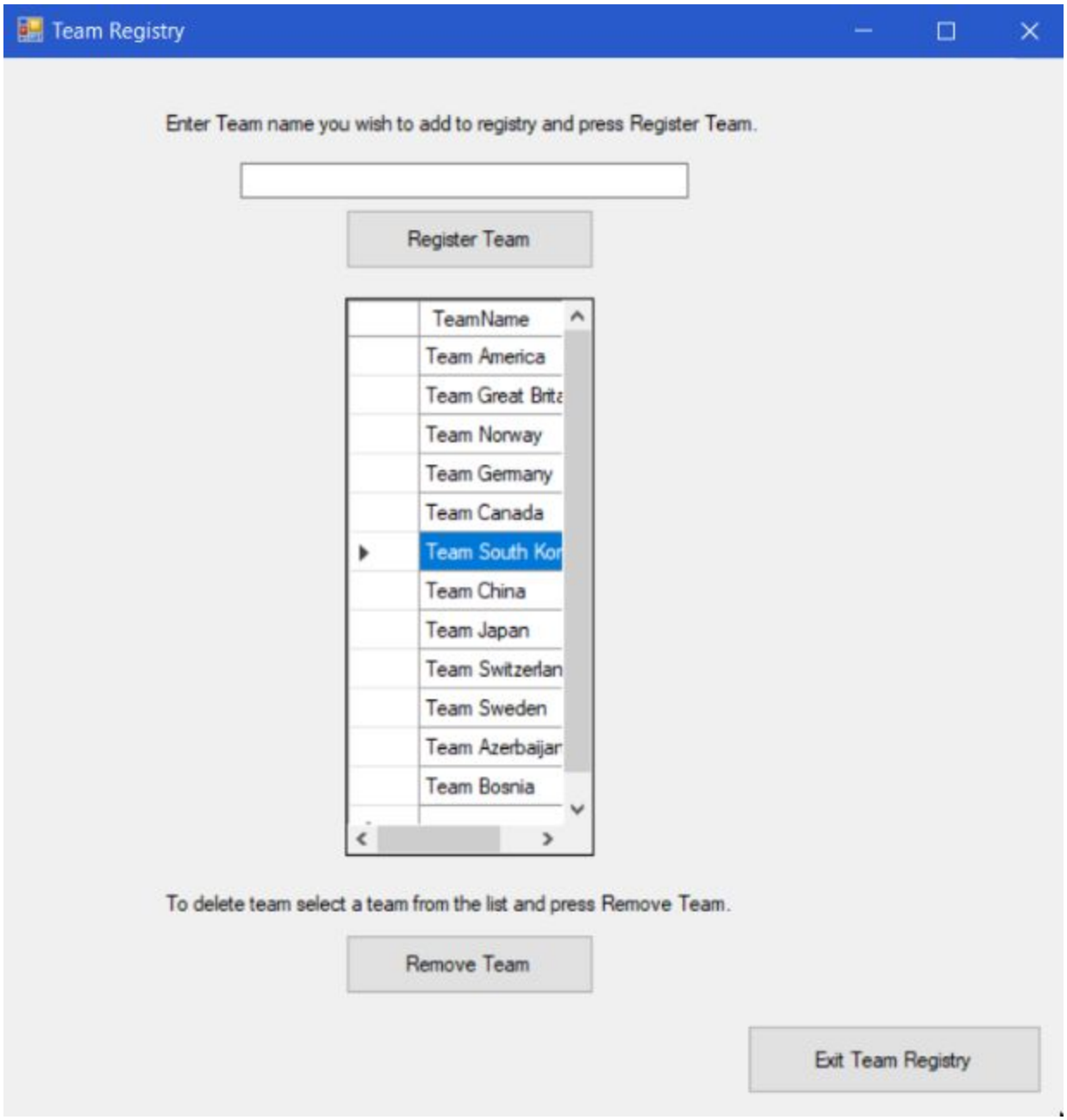
3. External Interface Requirements

3.1 User Interfaces

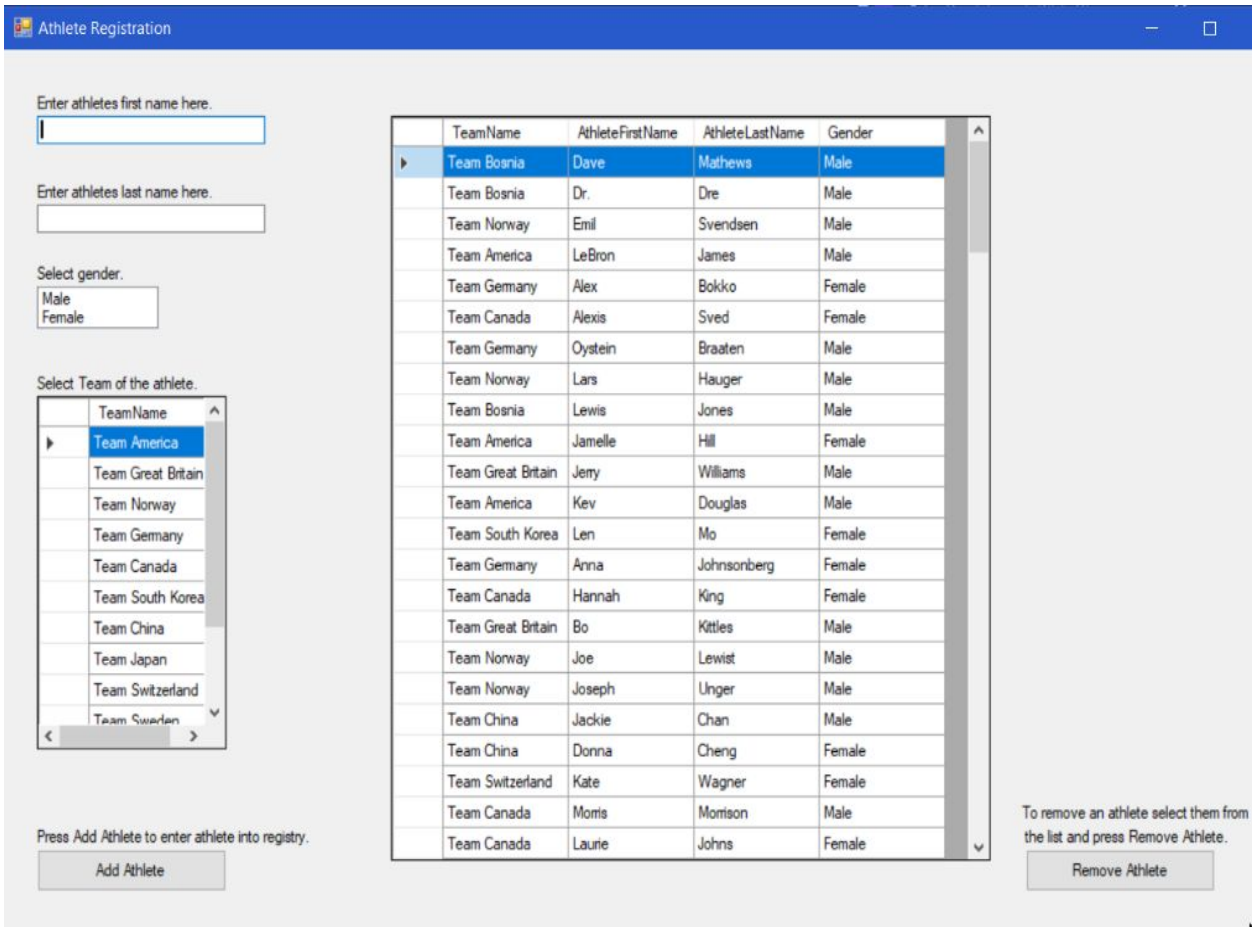
Below is the main page of the GUI. From here, the rest of the system can be accessed via one of the 5 buttons.



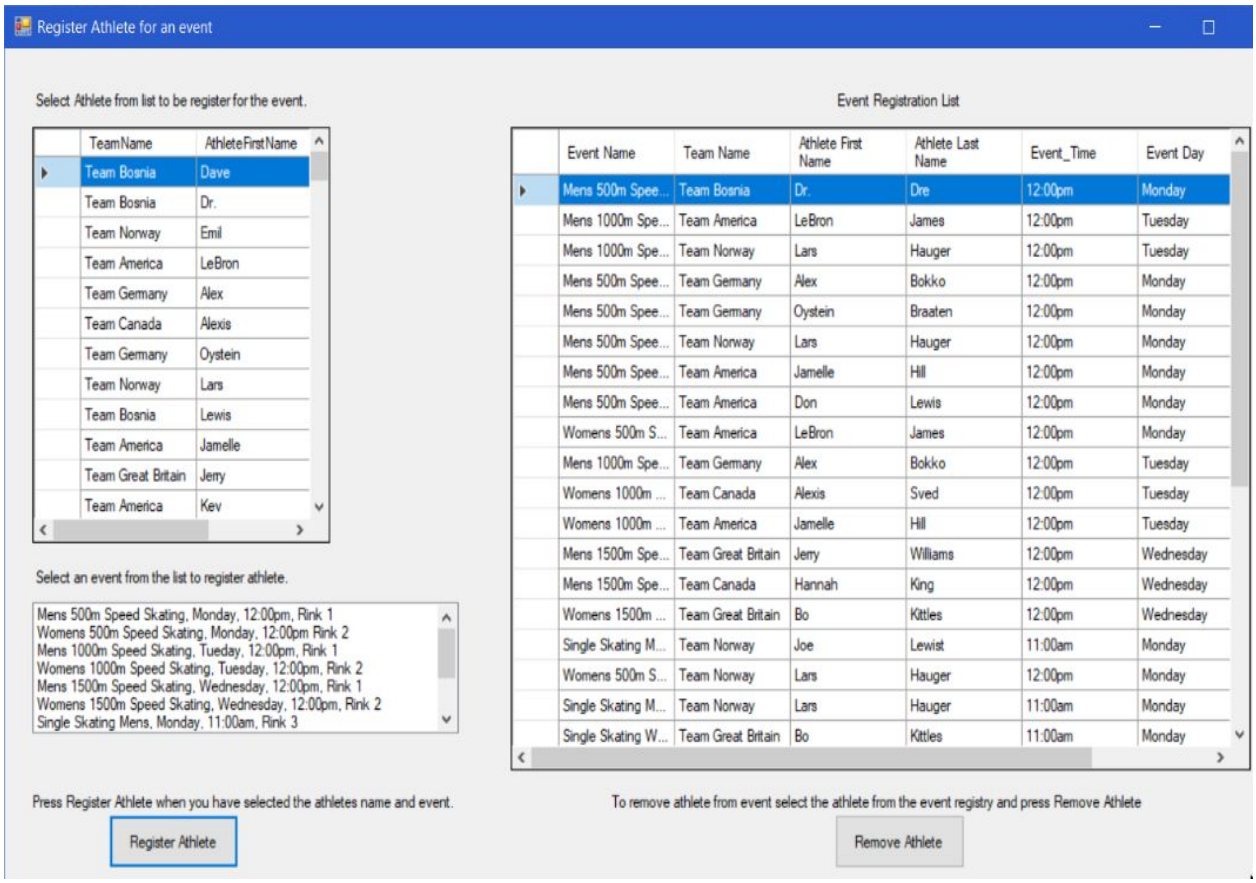
Below is the screen to add/remove a team from the **WO**. Current listing of every team is present. Exiting the screen can be done on the bottom-right.



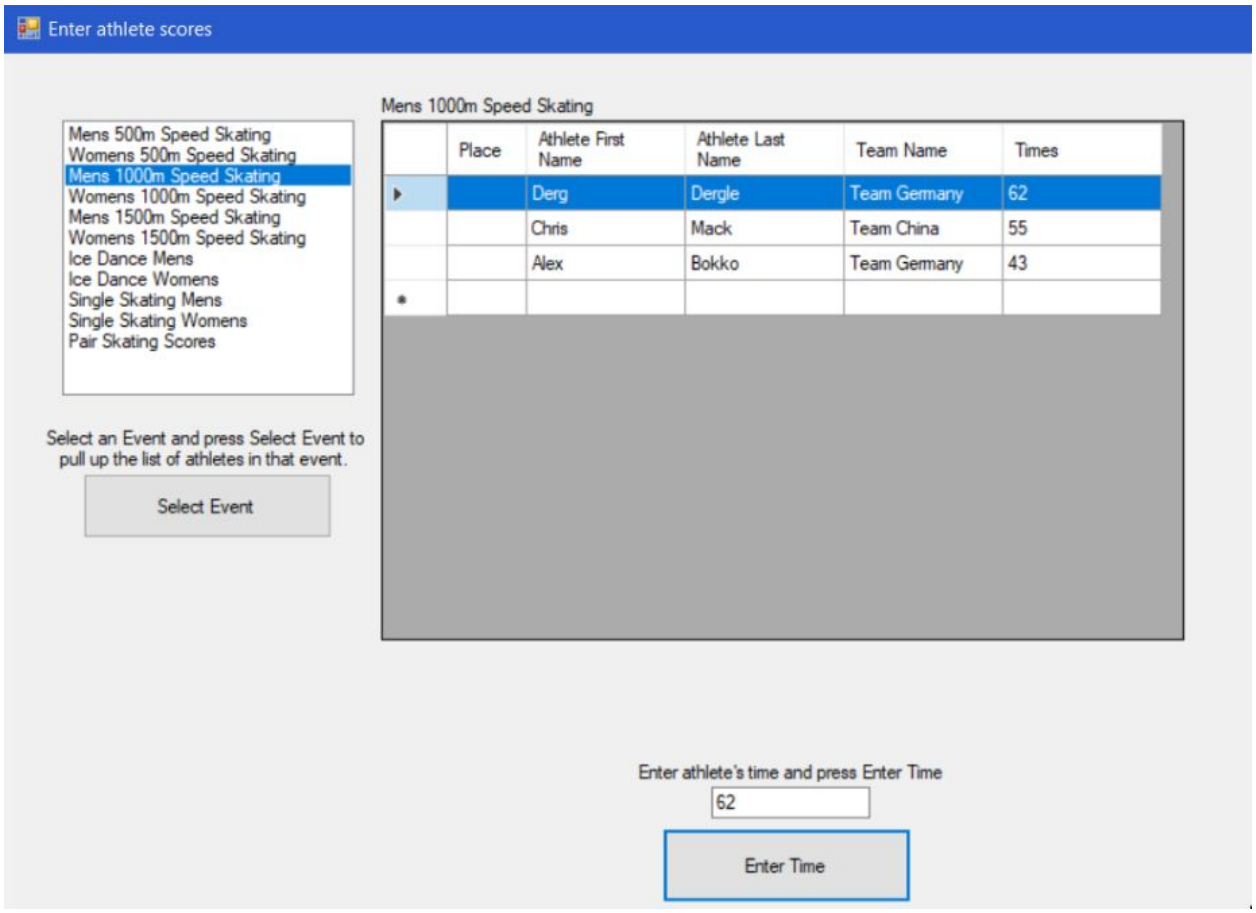
Below is the registration process for each athlete and their attributes. The current listing of each athlete in their respective team is present. There is also the listing of each team registered akin to the above screen.



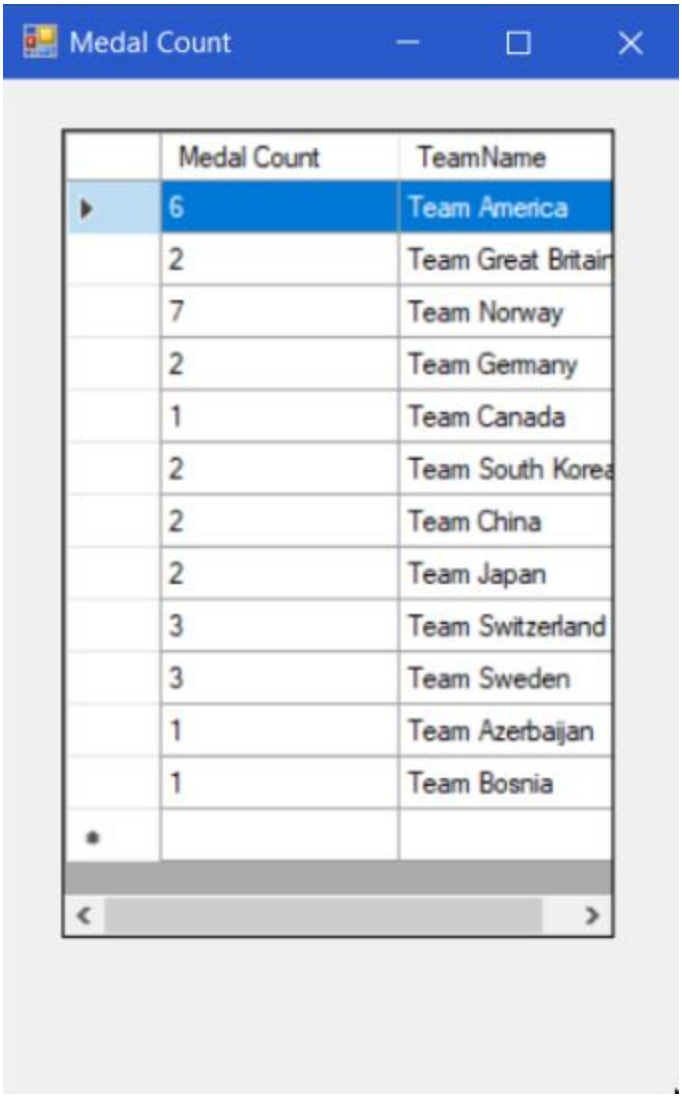
Below is the registration/deregistration of each athlete to their respective event. Each athlete is available to be chosen along with a skating event. These choices are also displayed in the Event Registration List.



Below is where the Official Judge enters the scores/times from each event. The event listing is off to the left and the scores/times may be entered toward the bottom. Entering time is straightforward. Entering Figure Skating scores consists of seven different scores which the highest and lowest are thrown out and the rest are averaged automatically.



Below is the standing medal count for each country.



3.2 Hardware Interfaces

The only hardware this system is designed for at the present time is a computer utilizing a Windows Operating system.

3.3 Software Interfaces

The system currently needs access to the SQLITE DB and any required DLLs. These can be handled by having relevant files in the root of the program. The operating system must be Windows with options to expand operations to other operating systems in the future.

3.4 Communications Interfaces

At the present time, there are no Communications Interfaces associated with **PCO**.

4. System Features

4.1 Register Team

4.1.1 Description and Priority

Ensure that each country participating in the **WO** is registered into the database. High priority.

4.1.2 Stimulus/Response Sequences

Official Registrant logs into the system from outside the skating subsystem. The official then clicks on the button labeled 'Register Team'. From there, the official then views the list of the events and chooses the event at hand. They are able to add/delete athletes as needed. Adding athlete by name, gender, team is present.

4.1.3 Functional Requirements

REQ-1: Include all the necessary DLLs in the bin/debug directory of the project
REQ-2: Include the SQLite database file in the bin/debug directory of the project
REQ-3: TBD: Unique error messages for wrong inputs
REQ-4: TBD: Unique conformation messages

4.2 Register Athlete

4.2.1 Description and Priority

Ensure that each athlete participating in the **WO** is registered into the database as well as registered to the appropriate team. High priority.

4.2.2 Stimulus/Response Sequences

Official Registrant logs into the system from outside the skating subsystem. The official then clicks on the button labeled 'Register Athlete.' From there, the official then views the list of the already-registered athletes. They are able to add/delete athletes as needed. When finished, they are able to exit team registry utilizing the button on the bottom-right.

4.2.3 Functional Requirements

REQ-1: Include all the necessary DLLs in the bin/debug directory of the project
REQ-2: Include the SQLite database file in the bin/debug directory of the project
REQ-3: TBD: Unique error messages for wrong inputs
REQ-4: TBD: Unique conformation messages

4.3 Register Athlete for Event

4.3.1 Description and Priority

Ensure that each athlete participating in the **WO** is registered to their respective event(s). High priority.

4.3.2 Stimulus/Response Sequences

Official Scheduler logs into the system from outside the skating subsystem. The official then clicks on the button labeled 'Register Athlete for Event.' From there, the official then views the list of the already-registered athletes (per event). They are able to add/delete athletes into events as needed. Register athlete confirmation is on the bottom-left while deregister athlete is on the bottom-right.

4.3.3 Functional Requirements

REQ-1: Include all the necessary DLLs in the bin/debug directory of the project
REQ-2: Include the SQLite database file in the bin/debug directory of the project
REQ-3: TBD: Unique error messages for wrong inputs
REQ-4: TBD: Unique conformation messages

4.4 Enter Judges Scores

4.4.1 Description and Priority

Input scores / times from each event in regards to each athlete in said event. High priority.

4.4.2 Stimulus/Response Sequences

Official Judge logs into the system from outside the skating subsystem. The official then clicks on the button labeled 'Enter Judges Scores'. From there, the official then views the list of the events and chooses the event at hand. To confirm the event, the confirmation button is located on the middle-right. They are able to enter scores or times per athlete in the unique event. The confirmation button is located toward the bottom-middle/right.

4.4.3 Functional Requirements

REQ-1: Include all the necessary DLLs in the bin/debug directory of the project
REQ-2: Include the SQLite database file in the bin/debug directory of the project
REQ-3: TBD: Unique error messages for wrong inputs
REQ-4: TBD: Unique conformation messages

4.5 Look up Event Rankings

4.5.1 Description and Priority

Ensure that each country participating in the **WO** has their accumulative medal count from the events displayed. Mid priority.

4.5.2 Stimulus/Response Sequences

Official logs into the system from outside the skating subsystem. The official then clicks on the button labeled ‘Look up Event Rankings’. From there, the official then ensures the medal count is correct for display. The medal count is auto-generated from the database.

4.5.3 Functional Requirements

REQ-1: Include all the necessary DLLs in the bin/debug directory of the project
REQ-2: Include the SQLite database file in the bin/debug directory of the project

5. Other Nonfunctional Requirements

5.1 Performance Requirements

There are no performance requirements at the current point in time.

5.2 Safety Requirements

There are no safety requirements at the current point in time.

5.3 Security Requirements

Due to the infancy of this system, there are no security implementations. This system is to be used solely by the officials of the **WO.** Therefore, this system is already embedded within a larger system for the entirety of the **WO**. At the present time, it is assumed the officials have already logged into the main system and accessed the subsystem from there.

5.4 Software Quality Attributes

Since the system comes complete with a GUI and a database, interaction with the system is very simple to use as well as very simple to learn. Maintainability requires only knowledge of C# programming language and SQL. System may be reused for future **WO** events, and events that are not apropos to skating.

5.5 Business Rules

Only the official **WO** committee may utilize and purchase this system through the discretion of Dr. Guercio. Only officials of the **WO** are to operate this system (Registrant, Head Judge, Scheduler, Display Manager). However, the system does have a public display.

6. Other Requirements

PCO will require a functional database in order to house all of the data accumulated from the **WO**. This includes information from teams, athletes, judges, scores, events, rinks, and medal counts. A GUI is required for an easy use of the system. Both of these are already included in the system package. This extensible system will be able to be reused for any of the other branches of the Olympics as a whole. There are no legal requirements at this point in time.

Appendix A: Glossary

PCO - Project Codename Olympia
WO - Winter Olympics
Data_DictionaryV4.docx
ClassSpecificationsV2.docx

Appendix B: Analysis Models

ClassDiagramV51Functions.png
ClassDiagramV61Uses.png
ClassDiagramV71Attributes.png
UpdatedClassDiagramV81.png
UseCases2.docx
UseCaseDiagramV4.png

Appendix C: GitHub

https://github.com/Tutlegoss/Codename-Project-Olympia

https://github.com/Tutlegoss/Codename-Project-Olympia.git