

Test Plan

1. Test Plan Overview

1.1. The test plan will consist of 10 primary tests. Each test will involve a different aspect of the program and the database. Most tests will deal with loading data from the database into forms and saving data from the forms into the database. Other tests will make sure data can be effectively manipulated in the database. These tests will involve creating, deleting and updating data.

2. Test Cases

2.1. Create Seasons, Teams and Players Test

Input: The volleyball database will be fully populated with 10 seasons each of which will include 10 teams. Each team will then include 10 players. This will be done using the Create New Season form.

Description: The test will verify that an entry for each player will be placed into the main season table and that an empty table will be created for each new player. This will be done using the Create New Season form.

Expected Output: A new entry will be added to the season table and a uniquely named table will be created for each player added to the volleyball database.

SRS Requirements Tested: 2.2.1.1 (Create New Season), 2.2.1.1.1 (Add Player), 2.2.1.1.2 (Add Team).

2.2. Edit Seasons and Teams Test

Input: 10 different seasons and team names will be edited to include different years, locations and names.

Description: The test will verify that season and team information can be updated in the season table and the table name for each player can be renamed according to the edited information.

Expected Output: Season and team information in the season table will be updated to reflect the changes made by the user. Tables representing players affected by the changes should be renamed without modifying entries within those tables. The season, with its corresponding teams and players changes will be reflected from the drop down menus once they are changed.

SRS Requirements Tested: 2.2.1.2 (Edit Season), 2.2.1.2.2 (Edit Team).

2.3. Edit Player Test

Input: 10 different players will be edited to include different first and last names, positions and player numbers.

Description: The test will verify that player information can be updated in the season table and the table name for each player can be renamed according to the edited information.

Expected Output: Player information in the season table will be updated to reflect the changes made by the user. Tables representing players affected by the changes should be renamed without modifying entries within those tables. The player changes will be reflected in the drop down menus once the player is edited.

SRS Requirements Tested: 2.2.1.2.1 (Edit Player).

2.4. Delete Season Test

Input: 10 different seasons, each with a team and 10 players will be created and then deleted using the Delete Season form.

Description: The test will verify that seasons can be deleted; this includes deleting the teams and players within those deleted seasons.

Expected Output: All entries in the season table that correspond to the season selected by the user will be deleted. Also, all player tables for players who reside in that the deleted season will be deleted. The deleted season, with its corresponding teams and players will not be available from the drop down menus once the seasons are deleted.

SRS Requirements Tested: 2.2.1.3 (Delete Season)

2.5. Delete Team Test

Input: 10 different teams each with 10 players will be created and then deleted using the Delete Season form.

Description: The test will verify that teams can be deleted; this includes deleting the players within those deleted teams. Seasons will not be affected.

Expected Output: All entries in the season table that correspond to the team selected by the user will be deleted. Also, all player tables for players who reside in that the deleted team should be deleted. The deleted team, with its corresponding players will not be available from the drop down menus once the teams are deleted.

SRS Requirements Tested: 2.2.1.3.2 (Delete Team)

2.6. Delete Player Test

Input: 10 different players will be created and then deleted using the Delete Season form.

Description: The test will verify that only players can be deleted. Seasons and teams will not be affected.

Expected Output: All entries in the season table that correspond to the player selected by the user will be deleted. Also, all player tables that relate to the selected players will be deleted. The deleted players will not be available from the drop down menu once the players are deleted.

SRS Requirements Tested: 2.2.1.3.1 (Delete Player)

2.7. Load Team Test

Input: A team will loaded into the main statistical interface using the Load Season form.

Description: The test will verify that a team of up to 15 players can be loaded into the main statistical form and display the name and number of each player within that team in the main statistical form.

Expected Output: No tables or database information will be created, updated or deleted in this test. Once the season and team are chosen by the user, the players associated with that season and team will be loaded and displayed into the main statistical form.

SRS Requirements Tested: 2.2.4.1 (Load Season)

2.8. Input Invalid Data Test

Input: 5 seasons, teams and players will be created using the Create Season form.

Description: The test will verify that a user cannot attempt to manipulate entries from the database that do not exist. There is the possibility of manipulating the drop-down menus to load invalid data.

Expected Output: Once the user attempts to load, delete, or edit invalid data, a message will appear indicating which information is invalid. No data will be manipulated.

SRS Requirements Tested: 2.2.1.1 (Create Season), 2.2.1.1.1 (Add Player), 2.2.1.1.2 (Add Team), 2.2.1.2 (Edit Season), 2.2.1.2.1 (Edit Player), 2.2.1.2.2 (Edit Team), 2.2.1.3 (Delete Player), 2.2.1.3.1 (Delete Player), 2.2.1.3.2 (Delete Team), 2.2.1.4 (Load a Season).

2.9. Save to Database Test

Input: 5 teams will be created and loaded into the main statistical form. Once the team is loaded, a mock game will be played to update all player statistics. At the end of the game, the data will be saved to the database via the Export Data to Database menu option. The date and the opponent will then be typed in by the user. This will be done for each of the five teams.

Description: The test will verify that player statistical data can be stored into the volleyball database in each player table. It will be done simply by loading, playing a mock game and exporting to the database.

Expected Output: Once the user supplies the date and opponent, all player tables associated with the players on the team will be updated to include their own match statistics.

SRS Requirements Tested: 2.2.2.1 (Save Match).

2.10. Load from Database Test

Input: 1 team will be created with 10 players. The team will be loaded into the main statistical form and 10 mock games will be played. Each game will be

saved to the volleyball database. Then, each match will be loaded into the main statistical form from the volleyball database. The user will then input the date of the match that is to be loaded.

Description: The test will verify that previously saved matches can be loaded into the main statistical form from the volleyball database.

Expected Output: Each player table will be read and the proper match will be loaded into the main statistical form without errors. The matches will then be updated and inserted back into the database with a different date.

SRS Requirements Tested: 2.2.2.2 (Load Match)