MSSE SOFTWARE, INC.

Test Plan for
GolfScore

Contents

1. Introduction

1. Objective

The Test Plan is an aggregation of information, which describes the entire test activity for this project. It covers the entire testing effort (unit, development test, system verification test, and Beta). It identifies the product requirements, schedules, resource requirements (people, effort and equipment), quality, assumptions, exclusions, and risks.

A preliminary Test Plan is prepared for the Project Team during the System Phase of PEAQ Process. This Test Plan will be updated in the earliest possible time of the Implementation Phase, so that progress can be tracked during implementation.

2. Project Description

This project aims to process scores from a golf tournament and produce reports showing who won said tournament, as well as how the golfers performed on each course played.

3. Process Tailoring

For this project, there will be unit testing, as well as Design Verification Test, System Verification Test and final review with the client.

As for the unit tests, they will be handled by the developers. The Design Verification Test will be broken up into 2 phases:

- Integration testing which will also be handled by the developers.
- Functional testing.

The System Verification Plan will be done to ensure that the system works in its entirety. For this purpose, there will be:

- Specification testing,
- Compatibility testing,
- Documentation review.

As for the references, we used:

1. Software Requirements Specification for GolfScore, July 18, 2017.

2. Assumptions/Dependencies

In order to begin testing of the GolfScore program, the following needs to happen:

- GolfScore is available and installed on a PC running Windows 2000 or any later version.
- The units tests and integration testing have been done by the developer team.

3. Test Requirements

- Calling GolfScore
    - Correct format for calling GolfScore
- Input
    - Formatted text file containing the following records:

        - Course Records
            - Course name
            - Course identifier
            - Par for holes in single integer
        - Golfers Records
            - Name
            - Stroke count for each of the 18 holes
    - Golf courses from 1 to 5
    - Golfer playing one time on each course
    - Number de golfers from 2 to 12
    - Each course has 18 holes
    - Par for each hole is 3, 4 or 5 strokes
    - Stroke count is sum of strokes for one course
    - Score is from 0, 1, 2, 4, 6 and sums up
    - If the stroke count of one hole is over the par of this hole, score is 0
    - If the stroke count of one hole is equal to the par, score is 1
    - If the stroke count of one hole is 1 under the par, score is 2
    - If the stroke count of one hole is 2 under the par, score is 4
    - If the stroke count of one hole is 3 or more under the par, score is 6
- Output
    - 3 types of reports are created:
        - Tournament Ranking Report: trank.rep
            - Name, score for each course and total score, and final standing
            - Best golfer first
        - Golfer Report: golfer.rep
            - Same as tournament ranking report but alphabetically by last name
        - Course Report: course.rep
            - One section per course
            - For each course:
                - List of all the golfers name
                - Hole-by-hole stroke count for that course
                - Total score for that course (descending order of score)
    - Each type of records is delimited by
- Error handling
    - Parameter errors: program stops with adequate error message
    - Input data errors
        - Non-numeric data where it's supposed to be, non correct values => program stops with appropriate error message

    - Golfer with records of playing twice on the same course: the second record is ignored
- Output errors
    - Ask to overwrite if the file already exists.

4. Test Tools

Hardware is needed:

- Processors,
- Monitors

5. Resource Requirements

Since I'm still only a student, I have no insight on this.

6. Test Schedule

Same, being still a student, it feels overwhelming to invent something here.

7. Risks/Mitigation

As for the risks, malfunctions of the program is the higher risk here. Miscalculation that could lead to a false tournament ranking is to be avoided.

8. Metrics

The following metrics data will be collected. Some will be collected prior to, and some after product shipment.

Prior to shipment:

Effort expended during DVT, SVT and Regression

# of defects uncovered during DVT, SVT and Regression, and development phase each defect is attributable to

Test tracking S-Curve

PTR S-Curve

After shipment:

# of defects uncovered and development phase each defect is attributable to

Size of software

Test cases:

1	Number of golfer ‘1’ -> error
2	Number of golfer ‘2’ -> pass
3	Number of golfer ‘5’ -> pass
4	Number of golfer ’12’ -> pass
5	Number of golfer ’13’ -> error
6	Number of golfer ‘2.5’ -> error
7	Number of golfer ‘abc’ -> error