Indian Institute Of Technology, Kharagpur
Department of Computer Science and Engineering

Cricket Tournament Management System (CTMS)

Test Plan Document

Authors:
Arudeep Kumar (22CS10011)
Vaasu Singla (22CS10078)
Gaurang Gupta (22CS30025)

Group - **GARLIC NAAN**

7th of April, 2024

Contents:

1. Test Plan Identifier:

CTMS-TP-001

2. References:

The only document that is referenced here is the **Software Requirement Specification (SRS)** document.

3. Introduction:

The testing process for the Cricket Tournament Management System will be performed to ensure that the system is functioning correctly and that all of the use cases specified in the project description are implemented properly.

3.1 Overall strategy:

The testing process will involve several steps, including unit testing, and system testing.

1. In unit testing, each module or component of the system will be tested independently to ensure that it functions correctly.
2. And ,in system testing ,the entire system will be tested as a whole to ensure that it meets the requirements and specifications. The testing process will involve just manual testing.

The testing process will be documented in a test plan document, which will outline the testing strategies, testing schedule, and the expected outcomes of each testing phase.

4. Software Risk Issues

Integration Risks: Difficulty in integrating the CTMS with existing systems, such as databases or third-party APIs for live cricket data.
Technology Compatibility: Issues with the software running on different platforms or browsers, especially given the diversity of user devices.
Scalability Limitations: The system may not handle the load during peak times, such as high-profile tournaments, leading to performance degradation.
Security Vulnerabilities: Potential for data breaches or unauthorized access, particularly concerning player and team information.

Strategies for Mitigation

To address these risks, the development team should adopt several strategies:

Risk Assessment: Conduct a thorough risk assessment at the outset and at regular intervals throughout the project to identify and prioritize risks.
Prototyping and Testing: Early and frequent prototyping and testing can help uncover technical and usability issues before full-scale implementation.
Stakeholder Engagement: Regularly engage with stakeholders, including potential users, to ensure the system meets their needs and expectations.
Agile Development Practices: Adopting agile methodologies can help manage project management risks by allowing for flexibility and iterative progress.
Security and Legal Reviews: Regular security audits and legal reviews to ensure the system complies with all relevant laws and standards

5. Features to be tested (Test Items):

5.1 Use-case based testing:

The purpose of unit testing is to validate that each unit of the software works as intended and meets the requirements.

● Test the functionality of generating random player and team details.

● Test the functionality of generating a random match schedule.

● Test the functionality of generating random match details for a particular match.

● Test the accuracy and completeness of player statistics displayed , including runs scored, strike rate, average, and wickets taken.

● Test the accuracy and completeness of team statistics displayed, including matches played, total matches won, highest run scorer, highest wicket-taker, and last 5 match results.

● Test the accuracy and completeness of match scorecards displayed.

● Test the optional functionality of displaying tournament statistics, including the five highest wicket-takers and five highest run-scorers.

● Test the system's ability to store and retrieve player, team ,and match details accurately and securely.

● Test the system's ability to handle various inputs and scenarios, including unexpected user inputs and errors.

5.2 Performance testing:

●Verify the time it takes to run functionalities such as match and match details generation with large number of teams

Verify the generation of statistics in cases of large number of teams

6. Features not to be tested:

- Editing randomly generated player and team details (unnecessary to do once the details have been generated).
- Editing randomly generated match schedule (unnecessary to do once the schedule has been generated).
- Editing randomly match details for a particular match (unnecessary to do once the details have been generated).5

- Editing player statistics including runs scored, strike rate, average, and wickets taken (unnecessary to do once the statistics has been generated).
- Edit the tournament statistics, including the five highest wicket-takers and five highest run-scorers (unnecessary to do once the statistics has been generated).

7. Approach

7.1 Testing Levels

The testing approach for the Cricket Team Management System will encompass Unit, System/Integration (combined), and Acceptance test levels. While acknowledging budget and time constraints, the primary responsibility for testing will rest with the test manager, with the development team's involvement.

• **UNIT:** Developers will execute unit tests, collectively approved by the development team. Comprehensive unit test details will be shared with the testing team for their reference.

• **SYSTEM/INTEGRATION:** The test manager will lead System/Integration testing in collaboration with the development team. No specialized test tools are required. System/Integration testing will commence once critical defects have been addressed. Programs may possess up to two Major defects, provided they don't impede testing and have available workarounds.

• **ACCEPTANCE:** Actual end users will conduct Acceptance testing under the guidance of the test manager and developers. Prior to testing, end users will be briefed on acceptance test cases. Acceptance testing will commence once all critical and major defects are resolved. Programs may have one major defect, provided it doesn't obstruct testing and has a workaround. Before finalizing acceptance testing, all open critical and major defects must be resolved.

7.2 Defect Reporting

During the Unit Testing phase of the Cricket team Management System, the development team will assume responsibility for identifying and documenting any defects within the codebase. Defects will be meticulously documented, detailing their source, severity, proposed solutions, and estimated resolution time. This comprehensive information will be promptly shared with the testing team and project manager for further evaluation and resolution.

As testing progresses into subsequent phases, such as System/Integration and Acceptance testing, the focus will shift towards identifying defects that may have been overlooked during Unit Testing or arise from interactions between different modules or components of the system. Special attention will be paid to defects that should have

been captured during Unit Testing but were not, as these may pose greater risks or complexities to resolve at higher testing levels.

Ultimately, the overarching goal of the testing process is to ensure that the Cricket team Management System adheres to the required quality standards and remains free from critical defects that could compromise user experience or the integrity of the databse's data. Efforts will be directed towards achieving thorough defectidentification, resolution, and continuous improvement throughout the testing lifecycle.

8. Item pass/fail criteria:

8.1 Suspension Criteria:

On encountering any of the following cases, the test may be considered suspended:

1. software produces garbage/incorrect output
2. software doesn't respond/produce any output
3. software crashes
4. software takes long to respond/produce valid output

8.1 Approval Criteria:

The test case will be regarded as "**approved"** if the software produces accurate /expected results in expected time

9. Test Deliverables

- Acceptance test plan
- System/Integration test plan
- Unit test plans
- Test cases document

10. Environmental Needs

The successful execution of testing for the Cricket team management system requires access to the following elements:

• Access to the the full database system which consists of player details ,match details, team details, which are all stored in different database.

• Access to a testing dataset that includes a variety of scenarios and data configurations to thoroughly test the system’s functionalities.

11. Glossary

CTMS : Cricket team management system

SRS : Software Requirement Specification