Software Game Testing

Nimisha Karumathil

Student, Audyogik Shikshan Mandal's Institute of Management & Computer Studies, Thane, Maharashtra, India

ABSTRACT

The purpose of this paper is to analyze in general the process of testing games and describe in more detail all the aspects of test phases in a game development process.

The game industry has grown significantly in the last decade and many companies have started to use software engineering techniques in the game development process.

All developing processes and products are subjected to test, for validation and verification.

Modern software testing techniques can be easily used in game development as well, however it is not easy to test a game due to its complexity, but customers and managers are putting hard constraints on a game quality more and more frequently.

Effective testing techniques can reduce the testing times and improve game quality and production time.

KEYWORDS: Game Testing, Verification, Validation, Static and Dynamic Methodologies, Key Challenges of Software

INTRODUCTION:

Testing is a process used to help identify the correctness, completeness and quality of developed software.

Testing helps in Verifying and Validating if the Software is working as it intended to be working.

Testing should start at the early stage to reduce the cost and time to rework and produce error free software.

Testing involves using Static and Dynamic methodologies to Test the application.

MOST IMPORTANT LIFE-CYCLE

Software Testing Life Cycle   Testing Life Cycle
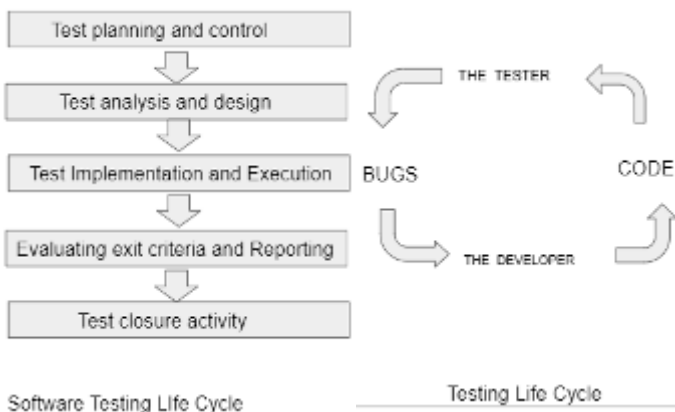
Testing Types and Techniques

1. Functionality Testing: This literally means the method to identify bugs or errors in a game that may affect the user-experience.

2. Clean Room testing: The game team test strategy in a cleanroom test is to produce the test that plays the game the way the user will play it.

3. Compatibility Testing: This is used to find whether a game is functioning properly or not with respect to the hardware, graphics and software configuration that the device is built with. It is one of the essential mobile app testing services that checks if a game title is able to run on specific devices.

4. Tree Testing: This type of testing is almost the same as usability testing. The only things needed are trees (menu). Improves the overall understanding of the complex features in the game.

5. Regression Testing: This testing is done after any kind of modification. Checking the impact of changed part on unchanged part of the application after modification.

6. Ad hoc testing: This is quite an unplanned testing method generally used to break down the system. Testers randomly test the app without test cases or any documents.

7. Load testing: It is a type of performance testing to determine the performance of a system under real-time loads. Load testing shows the reaction of an app when multiple users use the app simultaneously.

8. Play Testing: Play testing is the method of game testing by playing the game to analyse non-functional features like fun factors, difficulty levels, balance, etc. Here a selected group of users plays the unfinished versions of the game to check the workflow.

9. Localization Testing: Localization Testing is defined as a software testing technique, where the product is checked to assure that it behaves according to the local culture or settings. (drawbacks)

These are mainly used testing while testing any games. The technique which is used here is Black Box Testing and White Box Testing White Box Testing is used by Developers mainly and while doing Automation Testing testers use this. Black Box Testing is mainly used by Testers to test each functionality (functional and non-functional features) of the game.

LITERATURE SURVEY

According to research in 2017 & 2018 there were a total 15 games which were utter disappointment.

They weren't absolutely terrible games, however, each of them missed to match up to what the gamer's desire.

For a huge number of reasons, these 15 games didn't meet players' deepest desires, and in spite of the fact that fun value can surely be found inside them, they didn't accomplish the significance they were supposed to offer. From all of the 15 games which didn't workout, I found a loophole in the games because of which it was lacking to meet the Quality and Requirements of the end-users.

The key challenges of Software Game Testing
1. Testing considered late in the project.
2. Requirements not tested.
3. Integration is done after all components have been developed: This might result in full testing not being covered.
4. Complete testing is not possible.
5. Tester Focusing on finding easy bugs.
6. The testing always should be under time constraint.
7. There is a lack of skilled testers.
8. The requirements may keep on changing.
9. There can be some lack of resources, tools and training.
10. A complete application has to be tested.
11. A proper understanding of the requirements, domain knowledge and the business user perspective understanding.
12. To know which tests need to be executed first.
13. Not giving proper knowledge about the product or not giving any kind-off documentation to junior testers/ freshers.

Problems and solution to overcome all these challenges
1. Testing should start at the beginning of Software Development, right from the BRS document (Business Requirement Specification)
Read whatever documentation is available to you Think critically about what you have read and formulate questions about it Proactively engage with people to get your questions answered Developers for technical questions Product managers for feature questions Field people to understand how this is going to work with your customers in the real world Assemble the information you gathered into understandable and logically flowing test cases

2. SRS (Software Requirement Specification) Documents which is made by Senior Business Analyst or Technical Analyst should also be tested and reviewed
3. Each Component or Module should be tested as soon as they develop it by using the dummy module also called as (stub) / also can use the module of the previous project.
4. There is a criteria which we need to keep in mind while testing that complete testing is not possible.
100% test coverage.
80% should be prepared.
20% test can be failed.
5. Sometimes it happens that organizations reward testers based on a number of bugs (very bad approach to judging Testers Performance) then some testers only concentrate on finding easy bugs that don't require deep understanding and testing. A hard or subtle bug remains unnoticed in such a testing approach.
6. Some companies don't give proper knowledge about the product and also don't even give any kind-off documentation to junior testers/ freshers. At that time by using Exploratory testing we can do the testing here.
7. When there is lack of tools and training at that time Test Control should take place by Test Manager, and should add or remove the resource according to need.
8. It is obvious, Before the release of any Software Application it should be tested completely.

Here by the help of Complete Testing/ Testing Principle we can test the Software Completely.

V-MODEL / V_V Model / SEQUENTIAL MODEL
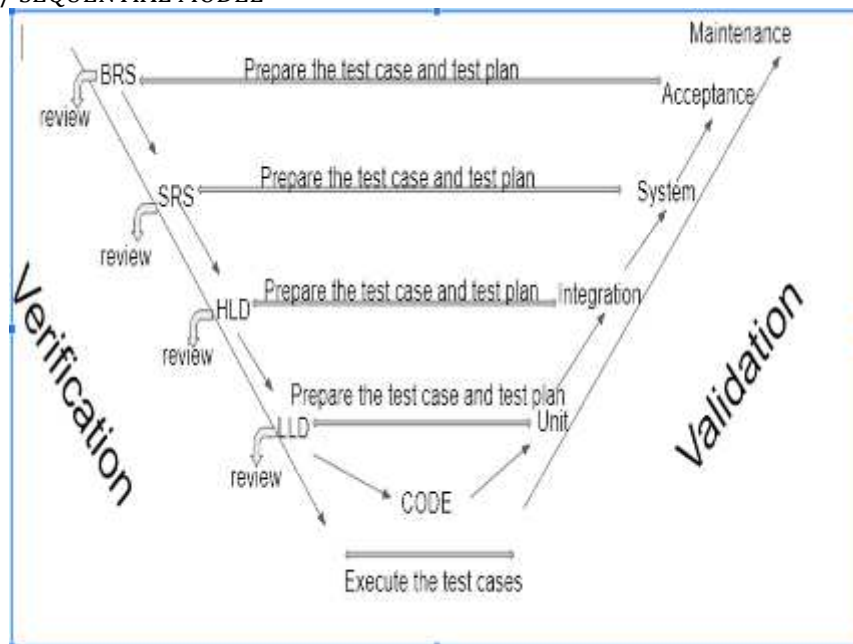
Principle of Testing / Complete Testing
#1) Testing Shows the Presence of Defects
#2) Early Testing
#3) Exhaustive Testing is Not Possible
#4) Testing is Context-Dependent
#5) Defect Clustering
#6) Pesticide Paradox
#7) Absence of Error

Conclusion

Testing a Game Software includes a lot many processes. In this work we have tried to give an overview of Software Game Testing. We have also introduced two most important life cycles of testing.

The type of testing which we have listed above are most commonly used testing types while testing a game. Also there are some major key challenges we have listed here which we face while testing a Gaming Software. While testing a gaming software we have a lot of things which need to be done manually. By using the V-V Model and Principle of Testing we can overcome the challenges which we are facing while testing the game manually. An average gaming development team spends more time on testing than any other application development team. Also there are many other advanced automation tools. But game development process and testing mostly based on black box testing.

References:

[1] https://pdfs.semanticscholar.org/4361/a1882ca8ea29 6ff6411dbbaa90ca5fbc3ed4.pdf

[2] https://scholarworks.rit.edu/cgi/viewcontent.cgi?artic le=10170&context=theses