8. "Significance of Software Testing Techniques for Business Model"

*Mrs. Rohini Kurundkar**

ABSTRACT

Software testing determines the quality of software after a programmer develops it. In recent years, software testing is becoming more popular and important in the software development industry. Software testing is a broad term encircling a variety of activities along with the development and beyond, aimed at different goals.

The Purpose of testing can be quality assurance, verification and validation or reliability estimation. It is tradeoff between budget, time and quality. Testing is the single most widely used approach to ensuring software quality. Software testing helps businesses pinpoint defects in their software and make appropriate corrections. If software products are not tested properly, there will be a major problem for any Business Model. This paper describes the importance of Software testing for any Business Model.

Keywords: Software Testing, Testing Techniques, Software Quality, Software testing strategies, testing tools

I.   INTRODUCTION

A. *Software Testing-*

Software Testing is one of the broader topics. It is a very complex activity deserving a first-class role in software development. Software Testing is nothing but error detection. Thus, whenever to design and implement the computer based system or product one should keep in mind the testability.

Testing is a process of evaluating the system or its components to find out the differences between existing conditions and the required conditions.

B. *Requirement of Software Testing –*

An effective Software Testing begins with a proper plan from the user requirements stage itself. Software testability is the ease with which a computer program is tested. Metrics can be used to measure the testability of a product. The requirements for effective Software Testing are given in the following sub-sections. There are really only three categories: explicit, implicit, and latent requirements.

1) Explicit Requirements: The Things You Wrote Down

Our first type of requirement is the explicit requirement. This is the simplest type and the easiest to test. Explicit requirements are most commonly found in documents communicated by stakeholders to the development team. They might take the form of an elaborate design specification, a set of acceptance criteria, or a set of wireframes.

The first thing to remember is that even in the cases where there isn't an explicit design document, there will always be explicit requirements lying around somewhere. Look for explicit requirements in the form of claims—that is, communications to end users about things the software can do. These can typically be found in user documentation or in marketing materials.

2) Implicit Requirements: The Things Your Customers Will Expect

Implicit requirements are the second type. These are all the things that users are going to expect that were not captured explicitly. Examples include performance, usability, availability, and security. Users *expect* that their password will not be stored in plain text; that requirement need not be written down by anyone.

Consider a cloud-based storage product that lets you store your files online. The product gets a new explicit requirement: Users should be able to share private content to other users

via URL, using a share button. However, while testing it is discovered that by modifying a value in the generated URL, it's possible for other users to view all of the sharing user's private content. This violates an implicit requirement that only shared content should be accessible to other users, resulting in a show-stopping bug.

3) Latent Requirements: Things That Will Delight Your Customers

Lastly, we have latent requirements. Latent requirements represent behaviors that users do not expect based on their previous experiences but which will make them like the software more. An example: My bank has an animated transition when I transfer money between accounts. I didn't expect it to do that, but it does help me understand that I was successful, and it looks cool, so I'm delighted. Another example would be cloud-sync in gaming: When video games started allowing users to access their saved game files on any computer, users were surprised and delighted by that feature.

Some websites will auto-complete your username when you start to log in. Some will not. This is an example of a latent requirement that is, over time, becoming an implicit requirement.

II. Software Testing Techniques

Functional testing anyway has to be performed for very component and when any component is integrated to an application, regression testing has to be performed to make sure that a change is not affecting any other component or any portion of an application.

Agile Methodology seems to be the most common approach for a firm that makes frequent releases to production with new changes every time in each release. Regression testing plays a crucial role in the complete testing process as it is a type of testing that re-tests the already working parts of an application after each release. Regression testing has to be done after any improvement or when new functionality is implemented or when bug fixing is done in an application. When Regression Testing is performed after a change is made in an application, one might wonder as to why a test that has passed yesterday which did not result in any bugs when repeated results in some new bugs — the reason is that a new code broke previously working functions or introduced errors or created bugs.

It is crucial for a testing team to make sure that other functional parts in an application are still working fine as expected before the application goes to production. Regression testing should be taken as a very important testing activity as if they are testing a feature for the first time.

Regression testing is an essential part of the quality process and ensures that code changes do not hurt the existing functionality. Effective regression testing can save a company's time and money. It should become a routine procedure while developing an application. For testers in the agile environment, they should not only cut down time frames for testing but their entire approach to testing should change.

Agile development needs fully functional features to be created in relatively short time-frames. Gone are the days of creating lengthy documentations. Regardless of the length of a sprint, proper testing must be performed before a feature can be declared as completed.

Testing team should follow 'Test after change and Test often' approach which will provide the results that inform whether new code is compatible with the existing code base. Regression testing includes executing an increasing set of tests along with covering existing functionality until the product is done. Continuous regression testing help teams build software that behaves as intended and remains stable.

III.   BUSINESS MODEL

A business model is the conceptual structure supporting the viability of a business, including its purpose, its goals and its ongoing plans for achieving them.

At its simplest, a business model is a specification describing how an organization fulfills its purpose. All business processes and policies are part of that model. According to

management expert Peter Drucker, a business model answers the following questions: Who is your customer, what does the customer value and how do you deliver value at an appropriate cost?

A business model is similar to a business plan in its makeup and content. However, a business plan specifies all the elements required to demonstrate the feasibility of a prospective business, while a business model demonstrates the elements that make an existing business work successfully.

The benefits of business model documentation include maintaining a focus on corporate goals, reviewing operational practices and ensuring that the two are congruent. A representation of a company's business model can be incorporated into public relations (PR) material and is useful to share with customers and partners. A mission statement or vision statement may be included in a business model.

Traditionally, financial goals have been the main focus of such models but business sustainability and corporate culture have become increasingly integral to business plans in recent years.

A. Types of Business Models

There are many different types of business models. Direct sales, franchising, advertising-based and brick-and-mortar are all traditional business models. Brought about by the internet, there is also a click-and-mortar business model, which combines a physical presence with an online presence.

Even if two businesses operate within the same industry, they likely have different competitive advantages and disadvantages and therefore, need different business models.

Think about the shaving industry. Gillette is happy to sell its Mach3 razor handle at cost, or even lower, because the company can go on to sell you the profitable razor refills over and over. The business model rests on giving away the handle and making profits from a steady stream of high-margin razor blade sales. This type of business model is actually called the

razor-razorblade model, but it can apply to companies in any business that sell one good at a discount while the second dependent good is sold at a considerably higher price.

Companies that sell electric shavers have a different business model. Remington, for example, makes most of its money up front on the sale of the razor rather than from a stream of blade refill sales.

IV. SOFTWARE TESTING & BUSINESS MODEL

Constantly and rapidly changing digital world demands high quality software applications at all times. It is also creating a tough competition between the software firms to develop custom and unique software products as per the customer needs.

It's not just as simple as developing software and delivering. The main problem is that many software products were delivered or launched without proper testing. It would be a huge trouble for the businesses if appropriate testing is not implemented for the software products before they are delivered. With continuous increase of online customers, businesses are forced to run high performing websites or web applications.

The online customers would like to visit a website that is equipped with fully functional features. With changing customer mindsets and usage patterns, businesses continuously change features or upgrade features of an application. If any change is made to an application, there is a possibility that it can affect the other features of an application.

Businesses need to make sure that their online business platforms are up and running at all times without interrupting customer usage. Proper testing needs to be done to assure the application are of good quality. It is a right approach to start testing along with the development of the applications.

3) Increase Sales

A good product needs less promotion than one with problems, because people recommend it and word-of-mouth is the most important promotion tool there is. Offering your clients a

product that has been rigorously tested and quality checked means that you value them and go the extra mile to offer the best on the market. This will go a long way into not only getting new customers, but also retaining them.

4) Cut Costs

How do you cut costs with software testing when testing in itself is a service you have to pay for? First, software testing saves you money on the long run because it makes sure you use and sell software that is reliable and that doesn't need constant fixing and patch-work. Think of the last time you compromised on a service or product and you'll realize that you probably ended up spending more on the long run, either in actual money or in inconvenience.

Second, as mentioned in the previous paragraphs, a major benefit of software testing is that it allows you to remove errors and problems before the products get on the market. This can spare you of great headaches later when unsatisfied clients come knocking on your door. Customer support can be very expensive.

Third, by using automated software testing solutions where applicable, you gain in quality and decrease the costs of the service. Another advantage is that automated solutions produce more consistency, which means more quality for your software.

V. HOW SOFTWARE TESTING LEADS TO BUSINESS OPTIMIZATION

Software testing is actually a great tool for business optimization. Testing can be a profit source for your Business in following ways –

1) Better Quality Products

Better quality testing means better quality software, which translates into products that add more value to the customer. Remember that customers are willing to pay more for increased value. What's more, when you sell a high quality product, you gain in reputation and brand image, things that are invaluable on the long term for the growth of any business.

2) Happier Customers

As you well know, the sale of a product doesn't actually end with the sale. The client can ask for a refund if he or she is unhappy with the product. In addition, if the product is not reliable, you will have to spend money fixing or replacing it. When you add the costs, the bottom line is that it pays to produce a higher quality product, and software testing done right is the only way in which you can guarantee that what you're offering is valuable and reliable to the customers.

3) Improve User Experience

Whether the software is used internally or sold to clients, it has to be easy to use and understand. Only testers with experience can make sure that the software is designed in a way that allows users to follow a logical and intuitive path. Improved user experience also means that the software should be free of bugs and errors, which can be a source of frustration and inconvenience for users.

Choosing a professional software testing service, with a team that has the experience and know-how to produce the best possible quality for you, is guaranteed to significantly improve user experience and, as a result, to result in more satisfied clients.

VI. SOFTWARE TESTING BUSINESS MODEL OF THE FUTURE

Software testing is an imperative process of validating and verifying that a computer program/application/product meets the requirements that guided its design and development, works as expected and satisfies the needs of stakeholders. This process would be considered effective only when the resulting product is free of defects, is reliable in addition to being associated with low cost, easy to be incorporated and requires low maintenance.

A study conducted by NIST in 2002 reports that software bugs cost the U.S. economy $59.5 billion annually. Software testing itself comes with a cost associated with a variety of challenges including but not limited to availability of resources (human/infrastructure) , industry wide tools/licenses and the most important of all

being "Time". Hence it becomes necessary to develop new business models and find innovative ways to do software testing that in turn can help achieve a WIN-WIN scenario from both the Client & Industry point of view.

Challenges manifest themselves as inherent characteristics of the software testing process that can be resolved through the use of Artificial Intelligence techniques. Thus a software testing model that is based on the incorporation of Artificial Intelligence methodologies for the products being developed would truly stand the test of time and help pave the way for the industry that has been long spending billions of dollars in order to carry out software testing at a lower cost and help customers achieve more flexibility through the self testing of software using AI methodologies thereby ensuring quality deliverables.

VII. CONCLUSION

Software testing is an important part of the software development process. As with the other activities in the software lifecycle, testing has its own unique challenges. As software systems become more and more complex, the importance of effective, well planned testing efforts will only increase.

Software developed for any Business model must go through software testing. This paper shows the significance of Software testing techniques for any business model. The knowledge of the software testing techniques that contribute to quality of the software is important to the advancement in any Business Model.

The bottom line is that software testing leads to business optimization. This is the biggest benefit of all, and it encompasses all the other benefits:

More satisfied clients
Customer retention

Less costs with customer service
Less cost with fixing
Automation of processes
Better quality and more reliable products
Improved reputation and brand image
Euro-Testing has almost ten years of hands on experience with helping businesses optimize their operations through professional software testing solutions and tools

References

[1] Software Testing – Effective Methods, Tools and Techniques – Renu Rajani, Pradeep Oak Tata McGraw Hill publication

[2] Foundations of Software Testing - ISTQB CERTIFICATION – Dorthy Graham, Erik van Veenendaal

[3] Software Testing Techniques -by Boris Beizer

[4] https://www.utest.com/articles/the-importance-of-regression-testing

[5] https://intland.com/blog/agile/test-management/why-qa-testing-is-so-important/

[6] http://blog.euro-testing.ro/5-benefits-of-software-testing-that-lead-to-business-optimization/

[7] http://whatis.techtarget.com/definition/business-model

[8] https://www.investopedia.com/terms/b/businessmodel.asp

[9] http://technosoftwares.com/importance-of-software-testing-strategy-for-online-businesses/

[10] http://www.cigniti.com/blog/what-why-and-importance-of-digital-testing/

*Asst. Prof. MCA Dept.
Dr. Moonje Institute of Management & Computer Studies, Nashik - 422005