



INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X

Impact factor: 4.295

(Volume3, Issue3)

Available online at www.ijariit.com

Agile Development Testing Paradigms

Saurabh Srivastava

LPIT Software Solutions Pvt. Ltd

srbh.spn@gmail.com

Abstract: Software testing in the agile development is a too complex issue in the Information Technology world. Since agile methods don't focus much on testing activities and agile software development doesn't include testing practices required for the quality of the project, the process of finding the suitable method for testing, mainly for large scale projects is challenging. In this paper, we propose a concept of integrated agile software testing in a large-scale project. This research defines the software testing process technologies and principles for the agile software testing and identifies the best practices for testing software products in the agile development process.

Keywords: Software Development, Agile Development, Agile Testing, E-commerce.

I. INTRODUCTION

Software testing practices that follow the principles of Agile software development is called Agile Testing. Agile is a repeated development methodology, where requirements evolve through an association between the customer and self-organizing teams and agile aligns development with customer needs. The standard of consumer demand for software products is increasingly high, accordingly, factors such as shorter delivery times and quality of the product or service became more and more crucial. Problems such as high cost, high complexity, the difficulty of maintenance, and a disparity between the needs of users and the product being developed, become increasingly evident in the processes of software development.

If the software product isn't tested until the different end of a long release, and the feedback loops will be enlarged and can be measured in months. That's too long. Shorter feedback loops increase Agility. Fortunately, on agile projects, the software product is ready to test almost from the beginning. And the Agile teams typically employ several levels of testing to uncover different types of information.

Principles of Agile Testing

- Testing is NOT a stage:
- Testing Moves project forward:
- Everyone Tests
- Shortening Feedback Response Time
- Clean Code
- Reduce Test Documentation.
- Test Driven

Agile development methodologies According to Ambler, the waterfall process of software development limits developers. Sometimes being a cumbersome and expensive development process, many organizations - especially small ones - end up choosing not to use any type of consolidated process. This fact highlights the need to use agile methods that are not extremely focused on documentation. Projects using Agile methodologies assume that change is common in software projects (and software-heavy projects) and thus value ongoing planning, emphasizing human aspects and adaptability to rapid changes in features and scope. While other processes try to address the problems in the waterfall model with interactions and an incremental approach (e.g. Rational Unified Process - RUP) they are still viewed as too rigid and bureaucratic. Currently, there are numerous approaches to agile software development; some of the most popular models are Extreme Programming (XP), Scrum, Lean, and Kanban. No less important and also used by a variety of organizations are Crystal and Agile Modeling (AM), but all have the following in common:

- Continuous planning;
- Involving the client in all phases of the project;
- Interactive and incremental process;
- Clear definition of roles;
- Iterations costs and scope well defined;
- Discipline in the workflow.

Software product testing is an investigation conducted to provide stakeholders with information about the quality of the software product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. The test techniques include the process of running a program or application with the intent of finding software bugs.

Software testing involves the implementation of a software component or system component to calculate one or more properties of interest. In general, these properties specify the extent to which the component or system under test:

- meets the requirements that guided its design and development,
- responds correctly to all kinds of inputs,
- performs its functions within an acceptable time,
- is sufficiently usable,
- can be installed and run in its intended environments, and
- achieves the general result its stakeholder's desire.

Manual regression tests take enlarged to execute and because a human must be available, may not begin immediately. The feedback time extended to days or weeks. Manual testing, particularly manual exploratory testing, is still important. However, the agile teams mostly search that fast feedback manage by automated regression is key to catch problems instantly, thus reducing risk and rework.

Problem Statement:

- Describes the different types of testing mechanism
- The learning mechanism is used to reduce the time of product development

Constraint:

- A set of software testing mechanism
- Complexity in Functional testing
- Wasted resources
- Inaccurate estimates
- Product Security

II. LITERATURE SURVEY

This section introduces some previous work done:

1. Testing in an Agile Product Development Environment: An Industry Experience Report

Our experiences in this set of projects (along the last two and a half years) highlight the importance of testing within agile methodologies and of implementing good practices following the whole-team approach. Although we built our proposal on top of Scrum, our findings and suggestions can be applied to different agile methods. Testing practices support several other agile practices and values, such as continuous integration, refactoring, and mainly the adaptability to change during development. All increments of software must be validated by sets of test cases for unit, integration, and acceptance testing to validate functionality exploratory tests and tests related to the operation of the system as a whole also should be used to criticize the system and to make sure that non-functional requirements are met.

2. Agile Software Testing in a Large-Scale Project

In a traditional project, “everyone is responsible for quality,” but in agile projects, everyone actually writes tests. This includes all developers, business analysts, and even the customer. For the IAF, this was a radical shift from existing practices. The organization implemented the practice after developers received the required extra training, but the customer took part only in defining and reviewing some test cases. Having everyone test offered several project advantages. First, it removes the team’s assurance on the single, assigned project tester, who would have otherwise constituted a major bottleneck. This verified highly important because—for reasons unrelated to the project—the tester was replaced twice during the first three releases. Second, because developers were responsible for writing tests for each new factor, their test-awareness increased and they prevented or quickly

Analysis of Various Testing Mechanism

- Top-down and bottom-up
- Traditional waterfall development model
- Agile or Extreme development model

III. AGILE SOFTWARE TESTING VS TRADITIONAL SOFTWARE TESTING

There is no IT meeting that does not talk and debate endlessly about the traditional Waterfall model vs. Agile development methodologies. Feelings run strong on the subject with many considering Agile ‘so of the moment’, just so right, while Waterfall is thought to be passé! But, before deciding which is more appropriate, it is essentially important to provide a little background on both models.

A classically linear and sequential approach to software design and systems development, each waterfall stage is assigned to a separate team to ensure greater project and deadline control, important for on-time project delivery. A linear approach means a stage by stage approach for product building

Agile methodology means cutting down the large picture into puzzle size bits, fitting them together when the time is right e.g. Coding, design, and testing bits. So, while there are reasons to support both the waterfall methods and agile methods, however, a closer look clarifies why many web design and software companies make the more appropriate choice of employing Agile methodology. The following table enumerates the *raison d’être* for choosing Agile methodology over the Waterfall method.

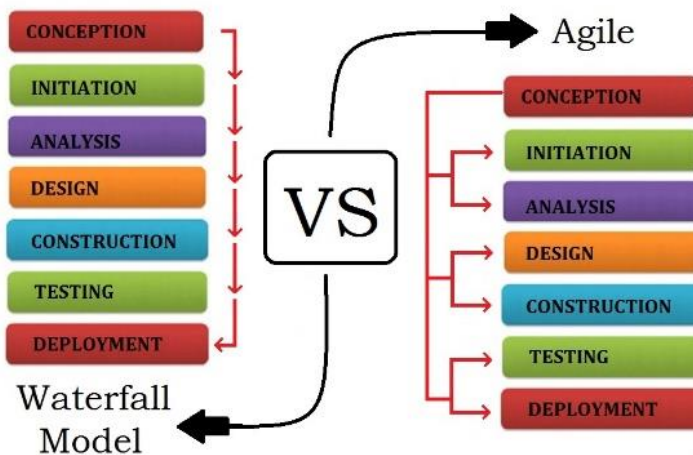


Figure 1: Agile vs. Waterfall

- The agile software developing methods are becoming an important part of the software development evolution process. Leading companies in the software development adopt these methods fast in order to make their products to be completed early and efficient.
- Testing methods in the agile development phases play an important role when a company wants to offer a quality product. If team project members want to use the agile software testing methods and principles, first of all, they must have a vision about what should be achieved and how it can be done in a less time and more effortlessly. The list below shows the most common principles of agile software testing.

Some of the major aspects of agile development are as below:

1. Testers in the agile software testing team must adapt to fast changes.
2. At the end of each of the iterations, there must be provided a working version of the software.
3. The customer must be included in testing process in order to accept the new features that had been made.
4. Agile software testing requires testing in the early stage and testing parallel with software development process.
5. Testing must be performed at each of the iterations.
6. Continuous integration is one of the most important steps in agile software testing and development because the development team should try to provide not the only functionality of the individual tasks, but the business logic.
7. Any changes in the process must be accepted by customers. However, using these principles does not guarantee success. Everything depends on the team structure, project size, and customer requirements. The key factor for success is the proper combination of these principles that will be aligned properly with the company development policies.
8. The progress of the work can be calculated with the help of planned activities for a current period of time.
9. Testing process should be automated because it will provide hot testing results and efficient work progress.

IV. SYSTEM IMPLEMENTATION

LogicKits (E-commerce Application) is based on these tools:

- Edraw Max- is used to creating and publish various kinds of diagrams to represent any ideas.
- Apache - is web server software, that playing a key role in the initial growth of the World Wide Web.
- PHP- is a server-side scripting language designed for web development.
- MySql - is a relational database management system. The program runs as a server providing multi-user access to a number of databases.
- Sublime Text – Sublime Text is a sophisticated text editor for code, markup, and prose.
- WordPress- is open-source and free content management system (CMS) based on PHP and MySQL

The e-commerce module fulfills all requirements for operating a standard business-consumer web service. It offers a fully-functional category-based shopping cart system and real-time credit card billing, just as you would find in any large popular e-commerce site. However, this is simply the base of the e-commerce module; it offers a whole host of handy utilities, including (but not limited to):

1. Security
2. User tracking
3. Gift certificates
4. Mailing lists
5. An auditing system
6. Order fulfillment
7. Order tracking
8. Special offers
9. Online reporting
10. User-based product rating & professional reviews
11. Product association (linking)
12. Pre-ordering

The following is a quick whistle-stop tour of the interesting e-commerce module components, and the functionality they offer.

1. The shopping cart keeps track of user shopping sessions and orders. Orders can be saved and completed later.
2. Product selection is simplified through the use of the catalog engine, which enables categorization of all products into (multiple) categories, subcategories, and subcategories. Site administrators can upload product data in bulk from a structured data file.
3. The real-time credit card billing system allows orders to be authorized on the spot, while you wait. The card is billed once the order is shipped.
4. The e-recommendation engine allows products to be mapped to certain products for all or only some class of user.
5. Products can be reviewed either professionally, or through customer contributions.
6. User tracking provides a comprehensive history of a user's purchasing history, and also the products they have viewed. It can also integrate with the rest of the ACS to provide information about the user's activity on bulletin boards and discussion forums.
7. Content management is fully audited and allows the site administrators to enter or modify existing data. The look and feel of the visual interface can also be completely redesigned without coding or re-programming because the module is template-based.
8. Users can easily find the products they want using the intelligent parametric and free-text search engine.
9. User classes can be defined by the site administrators, and then used to determine the discounts and rates that will be offered to individuals, institutions, or students, for example.
10. The object-oriented design of the site facilitates easy customization of product information fields so that extra attributes can be described where appropriate.
11. The gift certificate system audits and administrates the allocation of gift certificates to customers.

The e-commerce module will be used by five main groups of users:

1. Site Administrators administer and customize the e-commerce module to suit their requirements. They use administration interface to add new products and categories and edit existing ones. They resolve any conflicts and perform maintenance such as updating product recommendations and (dis)approving customer reviews.
2. Site Designers customize the look and feel of the site. They have to be familiar with how to write HTML, with simple extensions for variable inclusion.
3. Customer Service Representatives issue gift certificates, write the template e-mails that are sent out to customers and perform the customer care activity available in the customer service module.
4. Order Processors fulfill orders and record shipment details so that customers and administrators can track the shipments later.
5. Customers browse the site for products they want to buy and purchase them. They also write reviews about the products and participate in other community activities on the site.

Use Cases: Adding electronic components to the shopping cart.

Roles: Customer, Visitor

Description:

1. The application provides the user with a list of electronic components.
2. The user selects a category.
3. The application returns the list of components that belongs to the selected electronic components.
4. The user can also select the component using search tag.

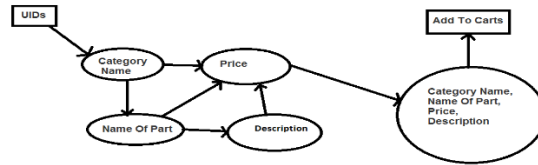


Figure 2: Example of requirements specification

To support users(visitors, customers, administrators) In the purchase of electronic components and the management of the system throughout the internet.

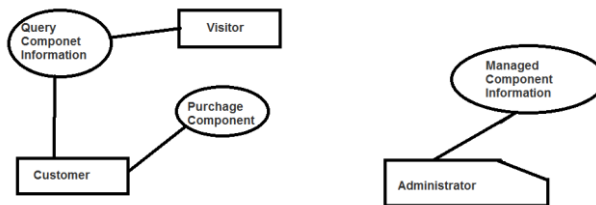


Figure 3: Audience Classification

Audience Characterization

1. Visitor views the different types of components.
2. Visitor view component description.
3. Visitor searches the product.

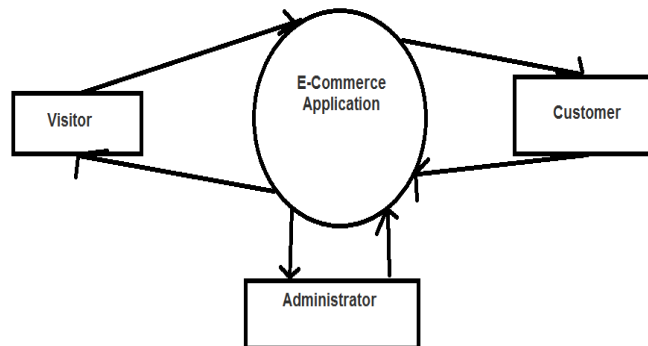


Figure 4: System Scope

Table 1: Events

Source Entity	Event Name
Visitor	Query Component Information
Customer	Purchase Component, Query Component Information
Administrator	Manage product Information

The upper side of this figure shows the use case diagram. The lower side shows the activity diagram associated with the use case find the electronic component.

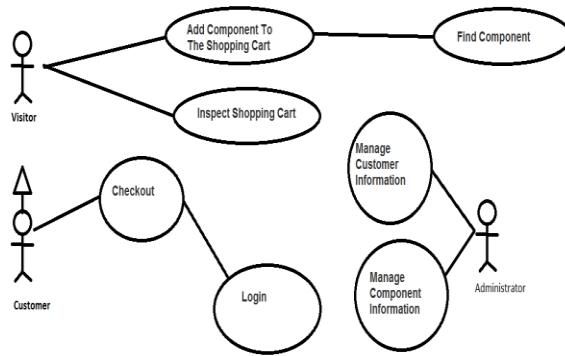


Figure 5: Activity Diagram

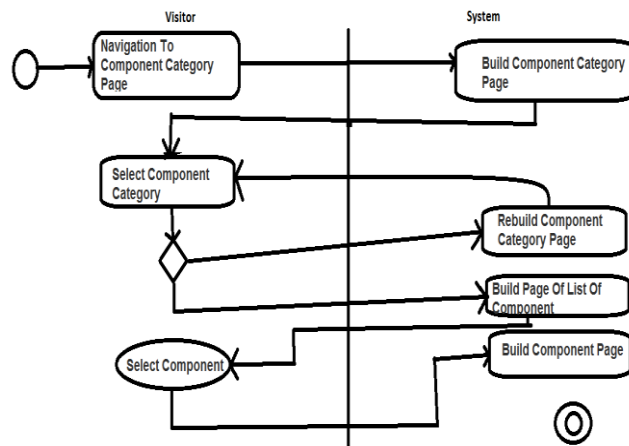


Figure 6: System User Interaction

Visitor

Description: it represent all the user that can access the public part of the web application

Profile data: No profile required

Object access in reading mode: product

Object access in write mode: shopping cart

Relevant usage: query product information, add component to shopping cart

Collect Products



Figure 7: Activity Diagram (Collect products)

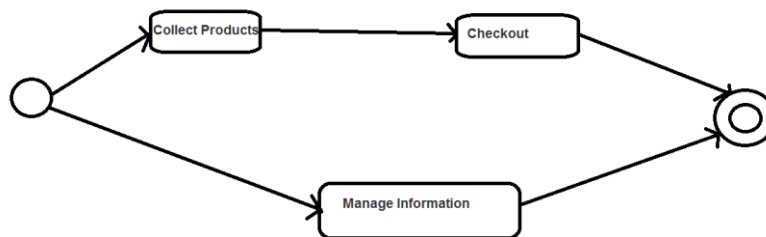


Figure 8: Functional Use Case Diagram

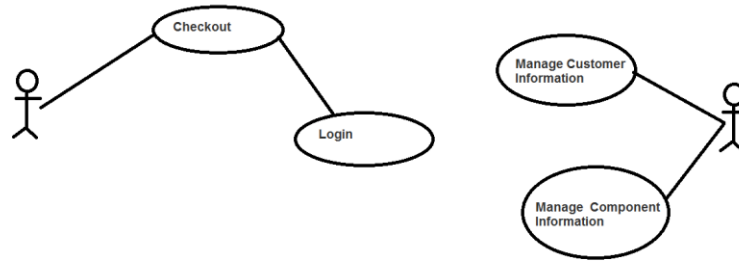


Figure 9: Navigational Use Case Diagram

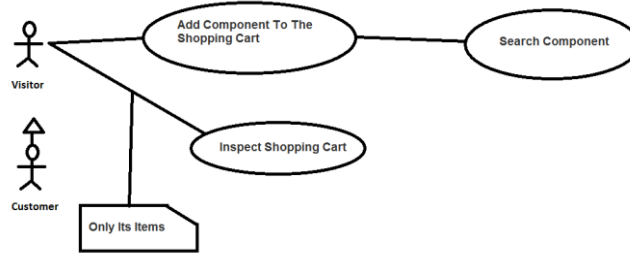


Figure 10: Interaction Use case Diagram

Testing an Ecommerce Shopping Cart Site

1. Browsers
2. The Cart
3. Cookies
4. Customers
5. Registration
6. SSL: Secure Pages
7. Credit Cards
8. Sessions
9. Servers
10. Regression testing

Browsers

1. Know what browsers your users have
2. Stay current on browser releases
3. Be conscience of what parts of your site are affected by the browser and browser version

The cart: adding items

Verify the item is added to the cart

1. Where in the store was the item added to the cart? For example is there a special page, discount box, banner ad, etc.
2. Is the buying functionality the same throughout the store? For example, if the customer buys from the banner ad versus from a regular item page - is the buying functionality the same.

The cart: changing item quantity`

Verify item quantity is increased/decreased

1. Are there quantity restrictions? For example, your site may restrict the quantity that may be purchased.
2. Are there restricted products? For example, some products may not be shippable in all states. Products such as alcohol have many restrictions

Verify the item is removed from the cart

1. Are there fees, taxes or discounts to be adjusted for example when the consumer removes an item, (or adds or changes the item quantity) are the corresponding delivery/shipping fees, taxes, and coupons adjusted as needed.
2. Is there item specific details that need to be removed? For example, does an item have gift wrapping or special handling/shipping details or substitution information that needs to be removed from the order details if the item is removed from the order

Cookies: What they are

According to O'Reilly's book JavaScript: The Definitive Guide, "A cookie is a small amount of named data stored by the web browser and associated with a particular web page or website". Cookies serve to give the web browser a memory so that it can use data that was input on one page on another page, or so it can recall user preferences or other state variables when the user drop a page and returns. Cookie data is automatically transmitted between a web browser and web server, so CGI scripts on the server can read and write cookie values that are stored on the client.

Cookies: Facts to know

1. Cookies are small files save on the client side. What information they contain varies completely by the site.
2. Cookies have an expiry date. Some cookies are only valid for the length of the session.
3. Cookies have a domain associated with them. This means cookies by your company and your company's domain is not readable by other sites.
4. Cookies have a path which specifies on which pages a cookie can be read.
5. Cookies have a security setting. Most cookies are available on insecure pages but in some cases, a cookie could be valid and read only on a secure page.

Cookies: What to test

1. Test your site's cookies according to the content of the cookie
2. Test with cookies and without.
3. Test the expiry date, domain, path, and security setting.
4. Test with a different type of browsers.

Cookies: Where to find them and how to remove them

1. Internet Explorer The cookie is stored in a separate file and the location depends on the operating system.

Windows 7:

Now you will be able to see the two real locations of Windows Cookies folders at the following address in Windows 7:

- C:\Users\username\AppData\Roaming\Microsoft\Windows\Cookies
- C:\Users\username\AppData\Roaming\Microsoft\Windows\Cookies\Low

Windows 8 and Windows 8.1:

Now you will be able to see the real locations of Windows Cookies folders at the following address in Windows:

- C:\Users\username\AppData\Local\Microsoft\Windows\INetCookies

Customers

1. 2. What information is maintained about each customer?
2. 3. Is your site customized?
3. 4. What information is stored encrypted?
4. 5. What are the rules for sending emails?

Registration

1. Unique user id Verify each user has a unique account.
2. Address information does your application restrict service or delivery based on which state or addresses the order is being shipped to? Does your application validate the street address during registration?
3. Passwords Are they stored encrypted? Are they encrypted in your user logs? Registration
4. The overall process how does the process work? Does registration handle user that use the back and forward button in the browser?
5. SSL Is the registration process secure? Can the user break out of registration – do the pages rotate from secure to insecure as needed?
6. Field validation Are required fields working as expected? Is there JavaScript for field validation?

SSL: What it is

According to O'Reilly's book Web Security & Commerce: SSL is a layer that exists between the raw TCP/IP protocol and the application layer. While the TCP/IP protocol sends an anonymous error-free stream of information between two computers (or between two processes running on the same computer), SSL adds numerous features to that stream, including:

- Authentication and Non-repudiation of the server, using digital signatures
- Authentication and Non-repudiation of the client, using digital signatures
- Data confidentiality through the use of encryption
- Data integrity through the use of message authentication codes

SSL: Where and how to test

1. Verify pages that should be secure are secure
2. Verify secure pages cannot be bookmarked

3. Verify secure pages work correctly on each server
4. Verify secure pages work on at least one version of IE and Netscape

The Credit Card

1. Check the credit card types allowed
2. Check the expiration dates of the card whether adding or updating the card info
3. Check the credit card billing address whether the address is updated or the card is updated

Table 2: The Credit Card number

Card	Prefix	Length
MasterCard	1-55	16
Visa	4	13 or 16
American Express	34, 37	15
Discover	6011	16

Session

1. Session timeout
2. Unique session IDs
3. Session caching
4. User logs

Servers

1. Server re-directs
2. SSL
3. Configuration files
4. User logs

Regression Testing

1. Create a base set of user scenarios that must be tested each release.
2. Identify for each release the areas of greatest risk and code that could be affected.
3. Balance risk vs likelihood and plan testing accordingly.

CONCLUSION

Agile Development is being incorporated by a majority of software development companies nowadays. Testing in an Agile development is one of the challenging paradigms. In this paper, we have presented a software development and testing strategy on our own e-commerce application which deals with electronic components. The application has been developed by our software development team using agile methodology.

REFERENCES

- [1] Mark Harman, Yue Jia, and Yuanyuan Zhang, "Achievements, open problems and challenges for search-based software testing", ICST 2015.
- [2] Achievements, open problems and challenges for search-based software testing, "Achievements, open problems and challenges for search-based software testing",
- [3] Bojana Koteska, Anastas Mishev, "Agile Software Testing Technologies in a Large-Scale Project"
- [4] Lu Luo, "Software Testing Techniques"
- [5] Andreia M. dos Santos, Börje F. Karlsson, André M. Cavalcante, Igor B. Correia and Emanuel Silva, "Testing in an Agile Product Development Environment: An Industry Experience Report"
- [6] David F. Rico, "Effects of Agile Methods on Electronic Commerce: Do They Improve Website Quality?"
- [7] Bojana Koteska, Anastas Mishev, "Agile Software Testing Technologies in a Large-Scale Project"
- [8] Jani Kukkohovi, "AGILE DEVELOPMENT AND TESTING IN EMBEDDED SYSTEMS"
- [9] Oscar Pastor and Silvia Abrahão and Joan Fons, "Building E-Commerce Applications from Object-Oriented Conceptual Models"
- [10] P.Salini and S.Kanmani, "IMPLEMENTATION OF MOSRE FRAMEWORK FOR A WEB APPLICATION - A CASE STUDY"
- [11] Heba Elshandidy, Sherif Mazen, "Agile and Traditional Requirements Engineering: A Survey"
- [12] Mohamad Kassab, "An Empirical Study on the Requirements Engineering Practices for Agile Software Development" Software Engineering and Advanced Applications (SEAA), 2014 40th EUROMICRO Conference.