Govt. Women Engg. College, Ajmer
Mid Term Exam II - Apr.3, 2018
Software Testing and Validation (IT - VIII Sem)

Max. Marks: 20 Time: 1Hr.

Q.1 What do you understand by Agile Testing? Can we apply Agile methodology on Waterfall Model? Explain with reason. [7 marks]

Q.2 Suppose you are developing training and placement portal for your college then explain steps for usability testing for it. [7 marks]

Q.3 Describe object oriented testing metrics. [6 marks]

Govt. Women Engg. College, Ajmer
Mid Term Exam II - Apr.3, 2018
Software Testing and Validation (IT - VIII Sem)

Max. Marks: 20 Time: 1Hr.

Q.1 What do you understand by Agile Testing? Can we apply Agile methodology on Waterfall Model? Explain with reason. [7 marks]

Q.2 Suppose you are developing training and placement portal for your college then explain steps for usability testing for it. [7 marks]

Q.3 Describe object oriented testing metrics. [6 marks]

Govt. Women Engg. College, Ajmer
Mid Term Exam II - Apr.3, 2018
Software Testing and Validation (IT - VIII Sem)

Max. Marks: 20 Time: 1Hr.

Q.1 What do you understand by Agile Testing? Can we apply Agile methodology on Waterfall Model? Explain with reason. [7 marks]

Q.2 Suppose you are developing training and placement portal for your college then explain steps for usability testing for it. [7 marks]

Q.3 Describe object oriented testing metrics. [6 marks]

Govt. Women Engg. College, Ajmer
Mid Term Exam II - Apr.3, 2018
Software Testing and Validation (IT - VIII Sem)

Max. Marks: 20 Time: 1Hr.

Q.1 What do you understand by Agile Testing? Can we apply Agile methodology on Waterfall Model? Explain with reason. [7 marks]

Q.2 Suppose you are developing training and placement portal for your college then explain steps for usability testing for it. [7 marks]

Q.3 Describe object oriented testing metrics. [6 marks]

Govt. Women Engg. College, Ajmer

Mid Term Exam II - Apr.3, 2018

Software Testing and Validation (IT - VIII Sem)

Max. Marks: 20 Time: 1Hr.

Ans. 1 In the world of software development, the term agile typically refers to any approach to project management that strives to unite teams around the principles of collaboration, flexibility, simplicity, transparency, and responsiveness to feedback throughout the entire process of developing a new program or product. And agile testing generally means the practice of testing software for bugs or performance issues within the context of an agile workflow.

In a waterfall approach, any given project's workflow proceeds in a linear series of sequential steps, progressing down the chain of production from Requirements to Design to Implementation to Verification to Maintenance. in a waterfall process the "design" and "implementation" stages exist prior to, and distinct from, the "verification" and "maintenance" stages. This split between software developers and software testers, positioning them as separate entities at different points along a production cycle, is one of the fundamental problems that Agile seeks to resolve. In the Agile approach, developers and testers are seen as two sides of the same production coin, two parallel lines that should always meet and compare notes daily. From an Agile perspective, efficient production is severely hampered if your developers are striving to refine their code to a state of perfection before passing it on to a separate testing team, who then strive to break it in as many ways as they can before sending their damage report back to the dev team. This two-step process requires time, money, and frequently leads to internal division between a given company's developers and testers. Instead, Agile suggests that these two essential functions be merged—not necessarily in terms of people, but in terms of time and process—thus bridging the illusory divide between code creators and code breakers, and even reducing the need for robust testing teams, while still respecting the necessity of both roles.

One could even say that in Agile, developers are encouraged to think more like testers, continually checking their own code for potential errors, and testers are encouraged to think more like developers, tempering their natural destructive tendencies to engage more fully in the creative process itself. Barber refers to this as creating a "product delivery team with a unified vision of delivering production-worthy code the 'first' time by integrating developer thinking and tester thinking throughout the code writing process." This integration implies both that developers enhance their tester thinking skills and embrace the notion of direct interaction, even pairing, with those who specialize in tester thinking while coding.

In short, Agile requires a depth of continuous collaboration between stakeholders—including managers, developers, testers, and customers—throughout the production process that isn't found in a traditional waterfall workflow. Testing becomes an essential component of each and every phase of the developmental process, with quality being "baked in" to a product at every stage of its development through constant feedback from everyone holding a vision of the final product.

Q.2 Suppose you are developing training and placement portal for your college then explain steps for usability testing for it. [5 marks]

Ans 2. Usability testing process contains the following steps:

Step #1) Identifying the users to perform the usability test – it helps to choose the set of users that is close to how the real-time users are going to be. Care has to be taken not to pick experts or complete newbies. The experts are going to simply run through the entire process and the novices need lots of background training to even get started- neither situation is optimum.

Step #2) Designing the tasks that the users are going to perform on the application –A list of situations that the users are going to use the application for are to be made prior to starting the test. This can include something like: 'Search for an X-box and buy it" or 'submit a customer care question" etc. on an eCommerce site. The tasks should closely represent the real transactions the users would use the site for.

Step #3) Facilitating the testing – The usability team will have the users perform the tasks on the site and are going to gather information regarding the test progress and results. It really makes a huge difference when the design team participates while the users are working on the application. It gives them a better picture of how the app was used and where it did not deliver what the user wanted etc, firsthand.

Step #4) Analyze results – At the end of the test, we might end up with the time it took to perform tasks, whether the task was successful or not etc, so basically raw data. The results have to be presented to all the stakeholders and analyzed for identification of potential problem areas.

Ans 3. OBJECT ORIENTED TESTING METRICS

Testing metrics can be grouped into two categories: encapsulation and inheritance. Encapsulation
Lack of cohesion in methods (LCOM) - The higher the value of LCOM, the more states have to be tested.
Percent public and protected (PAP) - This number indicates the percentage of class attributes that are public and thus the likelihood of side effects among classes.
Public access to data members (PAD) - This metric shows the number of classes that access other class's attributes and thus violation of encapsulation
Inheritance
Number of root classes (NOR) - A count of distinct class hierarchies.
Fan in (FIN) - FIN > 1 is an indication of multiple inheritance and should be avoided.
Number of children (NOC) and depth of the inheritance tree (DIT) - For each subclass, its superclass has to be re-tested. The above metrics (and others) are different than those used in traditional software testing, however, metrics collected from testing should be the same (i.e. number and type of errors, performance metrics, etc.).