You've finished your peer reviews

Well done! You sent 13 peers feedback that will help them. If you have time, please review one or two more. Every review you do helps another peer complete the course! Your fellow learner has submitted their assignment anonymously and your review will be anonymous to them. All names are still visible to course instructors. Project Scenario 1 response by Anonymous Learner

April 17, 2020

□ Flag this submission

PROMPT

What software development methodology would you

suggest for this situation and why? Step 1: Start analyzing the scenario by identifying the logic behind the selection of characteristics. For example, you may identify "User Needs Unknown" as a characteristic based activationative Type 1. The selection of characteristics and specified the correct logic?

O pts

Didn't identify this characteristic

Didn't identify this characteristic based on statement X, Y and Z in the scenario. oased on statement X, Y and Z in the scenario.

• Step 2: Select a model that best fits the characteristics you identified in step 1. Justify your choice by providing by the logic behind your selection. For example, you may say that since the scenario has characteristics X and Y, models A and B are potential candidates. reference statement to support this characteristic is "...with the exact same functionality. Thus, the requirements from client perspective are Additionally, since the scenario has characteristic Z, model A is the best option. very well known and do not need to change" Step 1: Characteristics of this situation and specify the logic behind the selection of characteristics.

2 pts Identified the characteristic and specified the correct logic primary and most relevant characteristic is "with the exact same functionality. ...requirements from the client perspective are very well known and do not something similar) as one of the characteristics? need to change."

• Re-architect the system and provide the

• 0 pts exact same functionality. (Based on Statement; "To support the anticipated demand, the company needs to re-architect 1 pt Didn't identify this characteristic the system and provide the exact same Identified the characteristic but the logic / the system and provide the exact same functionality.")
Product demand was not anticipated in the original design and current system cannot support increased demand. This is the logic citied in the production of an percentage. needs to be changed in the system to support the growing demand is also well understood" critical to the selection of an incremental Model. (Based on statements: "The product is loved by clients and is growing in popularity.

The level of product demand was not anticipated, and the current system

2 pts Identified the characteristic and specified the correct logic architecture cannot support the rising Changes required to support increased demand well understood. (Based on statement; "What needs to be changed in the system to support the growing demand the system to support the growing demand is also well understood.")

• 0 pts • Single component requiring highly scalable architecture is causing greatest pain requires immediate action. Focusing first on deploying the component causing the most pain will benefit both the customer and their users (Based on statement; "Out of the 4, one of them has caused the most pain and the organization could benefit greatly if that component could be replaced first with a new, highly scalable architecture."

• O pts
Didn't identify this characteristic

1 pt Identified the characteristic but the logic / reference statement to support this characteristic was incorrect. The correct logic / reference statement to support this characteristic is "Out of the 4, one of them has caused the most pain and organization could benefit greatly if that component could be replaced first with a new, highly scalable architecture." new, highly scalable architecture.")

• Product is 2 years old and well understood as a

component could be replaced fir highly scalable architecture." result this is a **known solution**. Client's perspective is that functional requirements do not need to change. (Based on statements; "Zenith Healthcare is a new company in the market and has launched its product two years market and has launched its product two years ago.", "requirements from the client perspective are very well known and do not need to change.")

• Four fairly independent components must be rearchitected. (Based on statement; "The product has 4, fairly independent components. All 4 components need to be re-architected.")

• Significant effort to migrate to new platform. All 4 components need to be re-arcuneaueu, ,
Significant effort to migrate to new platform.
(Based on statements; "The work of migrating")

1 pt

1 pt to a new platform is a tedious job and the deployment of a new system will involve a lot of external communication, managing customer expectations, etc." "The technical architect and one project manager will be Learner selected the right model: the Incremental Learner selected a model that will work but is not the preferred model (e.g. "Unified Process") and the preferred model (e.g. "Unified Process") are the preferred model (e.g. "Unified Process"). 2 pts
Learner selected the right model: the Incremental
Model

6 apray, but most of the team who will be
doing the coding for the migration will be
offshore in Belarus. The testing team will also
be in Belarus.")

7 pts
Learner selected the right model: the Incremental
Model

8 pts
Learner selected the right model and the right
variation of it: the most basic incremental model

- all phases are completed in each increment.
This allows us to replace the most pain-inducing Step 2: Select a model that best fits the the standard of this customer. In this case, meet the needs of this customer. In this case, because the requirements are well known I would select from a predictive model coefficielly the locromental Model. of them has caused the most pain and the organization could benefit greatly if that component could be replaced first with a new, highly scalable architecture." the beginning and end of each increment, ensuring the requirements and design of each increment are compatible with the What is the overall quality and detail of the response each increment are compatible with the overall goals of the project (all phases completed in each increment.)

We would set up four increments each focusing on a separate component with the first being the component requiring the highly scalable architecture. architecture.
 Communications and the new platform migration efforts might require the Enough detail migration efforts might require the development of phase gates and specialized sections of the section of the s box/creative thinking Any other open feedback for this question? PROMPT RUBRIC For the selected model, take us through a simulated / fictitious journey on how this project will be fictitious journey on how this project will be completed all the way from defining requirements to deployment. You are free to make up characters as you feel appropriate to fit your story. Please watch the video on "Model Selection" to get an idea. The you feel appropriate to it your story, rease masses the video on" Model Selection" to get an idea. The video stays at high level, but you can go in further details as you feel necessary. In your story, please details as you feel necessary. In your story, please
make sure to talk about artifact and practices
followed by the team on this project.

1 pt
Identified the characteristi Because we are using an incremental model, the first step would be to initiate a full re-validation of the requirements. During this process we would look at existing requirements to confirm they are still valid and develop any additional new requirements identified in the discovery phase. While the scenario indicates no need to all t and develop any additional new requirements identified in the discovery phase. While the scenario indicates no need to change client-based requirements, we should conduct a full review to ensure we are not missing anything critical to our product delivery. The output of this process would include a full set of current requirement specifications. Once the team is fully eatified in here include a full set of current requirement specifications. Once the team is fully satisfied it has valid and complete requirements for Increment one, we can move on to the design phase.

In the design phase, assuming our initial assessments. remain valid, we would likely divide the work into the four "fairly independent" components and look at both the individual components and the interfaces between each of these components (as needed). increment in our implementation phase. component is causing more issues. As a result, we would significantly reduce our pain, by focusing the first increment on designing and deploying a new, highly scalable solution. This increment would also experience with upgrading the overall system. phases (implementation through deployment) of this first increment prior to full implementation of the second phase. Phase gate reviews during the first increment would include a go-no go decision for the follow-on increments. Prior to a go decision, risk based decisions to perform certain low risk activities can be considered case-by-case. Increments two through four – Using the results of testing and the lessons we learned in the first increment implementation phase we would time the initiation of Increments two, three, and four to minimize impact on technical, communications, design, and customer experience. Any updates to the Increment two through four requirements and design would need to be validated and verified at the beginning of each increment. If risks are low and design changes are minimal, we may be able to execute the second, third, and fourth increments using a Sashimi model. While this is not critical to the success of the project, it could decrease costs and significant, we can choose to delay the third and fourth increment implementation phases as needed. \\ In order to ensure we minimize risk and maximize Requirements Review, Design Review, along with Testing entry and exit reviews. Each review would include both a Go / No Go decision and feedback to provide lessons learned to the follow-on increments. end of each increment. requirements, design, implementation, testing, and PROMPT RUBRIC Baseline testing should be performed to ensure we appropriate for the example? have a complete understanding of the total system prior to the execution of any increment.

• 0 pts

There are no suggested types (JUSTIFICATION) These baselines will help us to components performed during the incremental development. Baseline data will be maintained throughout the full development lifecycle. Each increment representing changes performed. Once the system meets all project goals and is fully operational a re-baseline would be performed. increment representing changes performed. Once the system meets all project goals and is fully operational a re-baseline would be performed.

Regression testing would be performed to ensure that previously developed and tested software still performs after a change. (JUSTIFICATION) This type of the still be should be the still be stil of testing should be performed after any significant change to a system this includes bug fixes, (with at most one exception), but the justification change to a system una mount of the hardware.
enhancements, upgrades to software or hardware.
Basically any change to the base configuration should initiate a set of regression tests.

does not us to a 8000-1-1 necessary.

8 pts enhancements, upgrades to software or hardware. does not do a good job of explaining why they are Functional testing should confirm that all desired functions are available and working as designed and expected. Separate tests would need to be developed to support verification and validation of all system functions (both those impacted by the current increment and the total system.) (JUSTIFICATION)
This is important as we want to ensure any changes made in each increment have the desired functional impact. We would need to compare results in each increment to our original baseline testing and investigate any unplanned results. We would want to confirm no significant impact to current functionality and confirm all new functions are fully operational. Integration testing should be performed for both the internal component integration points and the integration between each of the incremental builds.

(JUSTIFICATION) This testing should confirm that relationships between the components and subrelationships between the components and sub-components remain stable and operational. We would want to confirm that each Increment is fully integrated into the final solution. User testing we might consider a couple of different approaches for user testing. (JUSTIFICATION) First, we may need to synthetically confirm that the system scales appropriately to the desired system design. We would likely want to stress the system as much as possible and if possible attempt to identify the threshold at which the system degrades or becomes non-operational. Depending on these results we may need to make adjustments to the design. In addition, we may want to consider offering our trusted and experienced users an opportunity to Alpha and Beta test either the individual components or the full system prior to full deployment to the entire user Security Testing to ensure any identity, access, and privacy controls are properly functioning prior to full deployment to the user community. (USTIFICATION) Security and privacy are critical elements of any system, even more important to systems that impact customers/users. Confirming prevents potential damage from loss of data, integrity of the system, ultimately ensuring we maintain the highest reputation and positively impacting our users Failure testing. (JUSTIFICATION) We want to know how the system will respond as load is generated or should we experience a catastrophic failure of the underlying infrastructure. We need to ensure the new system meets our design requirements for catastrophic failure at all levels (platform failure, service failure, component failure, database failure, PROMPT RUBRIC Write a few example of test cases or a descriptive narrative for what you expect the testing team to use when testing this product.

Do the test cases or narrative provided make sense relative to the project at hand?

• 0 pts O pts
 There are no test cases or narrative system prior to any changes. Recause testing changed. system prior to any changes. Because testing should be performed periodically throughout the life of any system, we would use the tools developed for the critical system as undated throughout the systems. system, we would use the tools developed for the original system as updated throughout the systems life. This could include ping testing to confirm connectivity of the network, CPU threshold testing to confirm impact of systems and services on the compute capability of the system, application tests specific to the functions of the application itself. (for example, a test data set might be used to perform example, a test data set might be used to perform calculations confirming the results match a known good data set).

Regression tests - In the development of each component we will want to regularly perform regression testing to confirm that changes we made to our system do not negatively impact dependent components and services. To perform these tests we may want to build a "test system" that represents in part or whole the full system capability for the deployed system. We may need to build this test part or whole the full system capability for the deployed system. We may need to build this test project partially, or is incorrect in some major way system to scale or may build simple representations
of certain critical functions
9 pts system to scale or may build simple representations of certain critical functions.

Functional tests - Within the component there may be many individual functions. We will want to generate test results demonstrating the system is fully functional after all changes are made. We may need specialized tools to perform these tests, or we may need to develop artificial data sets to confirm we are getting the proper results from our functional after applicability. approached. are getting the proper results from our functional testing. We will also want to look at automated test tools to reduce the time and human interactions for testing. For example, if 1+1=2 as our current system functions we would want to demonstrate that the function of adding these two numbers results in the automated tool generating the question and confirming the results. four main components. Within the components there may be subcomponents. In all cases these components and subcomponents must operate together in order to achieve the desired results.
Connecting these components and subcomponents are many interfaces. We will want to have tools that are many interfaces. We will want to have tools that confirm that the interfaces are communicating properly and that the system generates the correct results. This testing can be performed using automated tools generating artificial data flows. The goal here is to have as much automation in the tools used for testing as possible in order to reduce the time and human offers required to employe the critical to overall satisfaction with our system. We will want to develop tools and capabilities around identifying the impact of change on the user experience. In this case we can develop "artificial" intelligence tools that represent users (singularly or in very large numbers) allowing us to confirm individual experience and what happens to the user experience mature the system and begin deployment we may want to develop a set of "trusted" users for the purpose of Alpha and Beta testing, where we allow these users early access to the new system with the understanding that they will provide feedback (both positive and negative). Security Tests - There are many tools available for security testing. We will want to confirm that user data is protected, that we meet all government regulations, that access is only provided to authorized users and that the user is provided only enough access to perform tasks they are fully authorized to perform (for example a System Administrator has significant rights to perform system specific activities, but a user would only need access to user specific areas.). Failure tests - We will want to develop tools that will simulate critical system failures, loss of data, loss of access, loss of hardware, etc. These tools would randomly introduce a fault in the system based on our desired goals. Our desire is to create a system capable of accurately determining the failure and rapidly responding to recover the system. We also want to confirm that messages to system

Waterfall Models
Incremental Models
Iterative Models

Applying traditional software development models

Video: Phase Gates / Stage Gates 8 min

Peer-graded Assignment: Project Scenario 1 2h

Review Your Peers: Project Scenario

✔ Peer-graded Assignment: Project Scenario 1

Reviews 13 complete

Sut

Comments

Comments left for the learner are visible only to that learner and the person who left the comment.

Share your thoughts...