Name: Wildan Luqmanul Hakim  
Enrollment ID: BDSE04-0322

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Module No:** | Application Implementation | **IU No:** | 1 | **Exercise No.** | 1 |

|  |  |
| --- | --- |
| **Lab Assessment Statement** | **Question is part of Module Project**  Envision Executing a Software Project  Develop a project plan (list of tasks in correct order), for each of Spiral, Waterfall, Agile, V-models to execute the software project. (2 pages). |
| **Technical Environment** | - |
| **Guidelines** | - |
| **Duration** | 20 mins |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Module No:** | Application Implementation | **IU No:** | 1 | **Exercise No.** | 2 |

|  |  |
| --- | --- |
| **Lab Assessment Statement** | **Question is part of Module Project**  Assume you are executing a project with 100’s of pages & processes.   1. List at least 2 disadvantages of each of the Spiral, Agile & V-Models while executing such a large software project, in a tabular format. 2. List the at least 2 advantages why using the Waterfall Model will be a good choice in this project scenario. |
| **Technical Environment** | - |
| **Guidelines** | - |
| **Duration** | 20 mins |

**Project scenario:**

 ABC Jobs Pte Ltd as a website developer to develop a community portal for Software Developers.

**Project Objectives:**

The Scope of the Project is to design a Community Portal Similar to Linkedin.com. , a Community Portal website where the website requires a Registration Page, Login Page, See Profile Users and Update User Profile.

**SDLC Models:**

1. **Waterfall Model**

The waterfall model is a classical model used in system development life cycle to create a system with a linear and sequential approach. It is termed as waterfall because the model develops systematically from one phase to another in a downward fashion. This model is divided into different phases and the output of one phase is used as the input of the next phase. Every phase has to be completed before the next phase starts and there is no overlapping of the phases.

1. **Requirement & Gathering Analysis**
2. Capture all requirements
3. Interview with clients/costumers
4. Conduct survey to stakeholders
5. Analysis the requirements

Discussing concepts with team members to better understand the requirement

1. Documenting the requirements
2. Develop the project Initiation document
3. Prepare the Project Plant document
4. Create work breakdown structure and work package
5. Estimation time and cost
6. Resources management
7. Project scheduling
8. Etc.
9. Create Quality Plan and Feasibility Test plan
10. Create Project Risk Management Plan Document
11. Document the Software Requirement Specification
12. Use case documentation
13. Requirement Traceability Matrix document
14. Do the Feasibility Test
15. Document Revision
16. **System Design**
17. As per the Requirement, create the design
18. Create the business procedure.
19. Produce use case diagrams, class diagrams, and activity diagrams
20. Design the system architecture.
21. Designing the UI / UX
22. Create a usability testing system
23. Capture the hardware and software requirement
24. Document the design
25. Record the system design.
26. Report the design software document
27. Write an instruction manual.
28. Write a maintenance manual
29. Create a training schedule.
30. Conduct Usability Testing
31. **Implementation**
32. Prepare the code guideline documentation
33. As per the design develop the software programs / code
34. Build all the functionalities / feature as per the requirement:
35. Registration
36. Login and Logout
37. Forget password
38. Homepage
39. See and update own profile
40. Search and see other profile
41. Administration user data
42. Etc.
43. Integrate the code for next phase
44. Documenting the software program
45. Create software development document
46. Create integrate document
47. Develop test case specification document
48. Conduct Unit Testing for each component
49. Revise the code and the documentation
50. **Testing**
51. Capture the test case document to perform testing
52. Perform all the testing as per the documentation
53. Perform Functional testing (Integration testing, acceptance testing, etc.)
54. Perform Non-Functional testing (load testing, performance testing, etc.)
55. Track the progress on testing through the traceability matrix documentation
56. Gather and Analysis the test result as per meet the requirement
57. Report all the testing activities and their result
58. **Deployment**
59. Prepare test case documentation for Beta testing
60. Create user manual if needed
61. Checking up
62. Make sure the environment is ready
63. Make sure there are no severe defect open
64. Make sure the test exit criteria are met
65. Deploy the application in the respective environment
66. Testing up after deployment
67. Perform sanity check in the environment
68. Documenting
69. Create version control document
70. Create post implementation review report
71. Develop any change request documentation
72. **Maintenance**

Maintenance will be carried out if there are feature updates or fixing errors found when the system is used directly by the user. d. Update the environment with the latest feature

1. **Spiral Model**

The spiral model is a systems development lifecycle ([SDLC](https://www.techtarget.com/searchsoftwarequality/definition/systems-development-life-cycle)) method used for [risk management](https://searchcompliance.techtarget.com/definition/risk-management) that combines the [iterative development](https://www.techtarget.com/searchsoftwarequality/definition/iterative-development) process model with elements of the Waterfall model. The spiral model is used by software engineers and is favored for large, expensive and complicated projects.

1. **Planning**
2. Capturing all requirements
3. Interview with clients/costumers
4. Conduct survey to stakeholders
5. Research similar product
6. Analysis the requirements  
   Discussing concepts with team members to better understand the requirement
7. Documenting the requirements
8. Develop the project Initiation document
9. Prepare the Project Plant document

* Estimation time and cost
* Resources management
* Project scheduling

1. Document the Software Requirement Specification
2. Requirement Traceability Matrix document
3. Feasibility Study

* Prepare the feasibility test plan
* Conduct feasibility test

1. **Risk Analysis**
2. Examined the specifications and identified any potential risks s
3. Calculate the risks.
4. Record the risk management system.
5. **Development and Testing**
6. As per the Requirement, create the design

* Create the business procedure
* Create Class diagram, activity diagram, and use case diagram
* Create the system architecture
* Designing the UI / UX
* Develop usability test plant

1. Capture the hardware and software requirement
2. Document the design

* Create a system design document.
* Make a user guide.
* Make a maintenance guide
* Create a training schedule

1. Conduct Usability Testing
2. Document Revision

* Revised requirement traceability matrix
* Revised test plant document
* Revised the system architecture

1. **Evaluation**
2. Customer and other stakeholder feedback on the software
3. An examination of the design
4. Documentation of any requested changes
5. Revision of version control documentation for the design
6. **Agile Model**

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product.

**Iteration 1 : Registration Epic (first module)**

1. **Plan**Meeting with a team member to evaluate the work
2. **Develop**Develop the feature / functionality as per the scope defined

* Registration
* Login and logout
* Homepage, view profile, and update profile

Prepare unit test plan and test cases

1. **Test/QA**

* Conduct a feasibility study and report the findings, along with any changes to the metrics.
* Perform unit testing and report the documentation of the results.
* Update risk management

1. **Deliver**

Show the stakeholders the current state of the project or its development.

1. **Assess**

* Obtain all of the stakeholder feedback.
* Take note of your change request for the following sprint.
* The most recent traceability metrics
* Collect data that will be used in the upcoming sprint

***Iteration 2 : Login Epic*** ***(second module)***

1. **Plan**Meeting with a team member to evaluate the work
2. **Develop**Develop the feature / functionality as per the scope defined

* Registration
* Login and logout
* Homepage, view profile, and update profile

Prepare unit test plan and test cases

1. **Test/QA**

* Conduct a feasibility study and report the findings, along with any changes to the metrics.
* Perform unit testing and report the documentation of the results.
* Update risk management

1. **Deliver**

Show the stakeholders the current state of the project or its development.

1. **Assess**

* Obtain all of the stakeholder feedback.
* Take note of your change request for the following sprint.
* The most recent traceability metrics
* Collect data that will be used in the upcoming sprint

***Iteration 3 :***

1. **Plan**Meeting with a team member to evaluate the work
2. **Develop**Develop the feature / functionality as per the scope defined

* Registration
* Login and logout
* Homepage, view profile, and update profile

Prepare unit test plan and test cases

1. **Test/QA**

* Conduct a feasibility study and report the findings, along with any changes to the metrics.
* Perform unit testing and report the documentation of the results.
* Update risk management

1. **Deliver**

Show the stakeholders the current state of the project or its development.

1. **Assess**

* Obtain all of the stakeholder feedback.
* Take note of your change request for the following sprint.
* The most recent traceability metrics
* Collect data that will be used in the upcoming sprint

**Additional Support**

Conduct scrum meeting for every 3 hour per day between stakeholders team and the client leaded by scrum master

1. **V-models (Sequential):**

**Left-Hand Side**

1. **Requirement Analysis**
2. Capturing all requirement

* Interview with clients/costumers
* Conduct survey to stakeholders
* Research similar product

1. Analysis the requirements  
   Discussing concepts with team members to better understand the requirement
2. Verification activities  
   Requirement review
3. Validation activities  
   Develop of UAT (User Acceptance Testing)
4. Artifact produced

* Understanding documentation in terms of requirements
* Document for the start of a project
* Test case for Project Plant document UAT
* The documentation of business requirements

1. **System Requirement / High-level design**
2. Investigate on how the requirement could be implemented
3. Verification activities  
   Design review
4. Validation activities:

* Develop of test facilities and cases
* Develop of a traceability metric

1. Artifact produced:

* System test case
* System test plan
* Feasibility report
* Hardware-Software requirement
* Modules to be created

1. **Architectural design**
2. Based on the system requirement, created

* Software architecture
* Modules, their relationship, and dependencies
* Architectural diagram
* Database tables
* Technologies details

1. Verification activities  
   Design review
2. Validation activities  
   Integration test plant and test cases
3. Artifacts produced

* Understanding documentation in terms of requirements
* Document for the start of a project
* Test case for Project Plant document UAT
* The documentation of business requirements

1. **Module design / Low-level design**
2. Documentation for each of module of the software component design

* Interface
* Class
* Method
* Data type
* Etc.

1. Verification activities  
   Design review
2. Validation activities  
   Creation and review of unit test cases
3. Artifact produced  
   Unit test cases
4. **Implementation / Code**
5. Implementing the code as per the module

* Register new user
* Login to community portal
* Logout from community portal
* See and Update user profile
* Forgot password
* Search another user
* Administer user data

1. Verification activities

* Code review
* Test case review

1. Validation activities

Creation of functional test cases

1. Artifact produced

* Test cases
* Review documentation
* Development documentation

**Right-Hand Side**

1. **Unit Testing**
2. Running each unit test case
3. Artifacts produced

Documentation of unit test execution results

1. **Integration Testing**
2. Executing the integration test cases defined in the architectural design
3. Artifacts produced  
   Documentation of integration test results
4. **System Testing**
5. Perform all system test cases

* Functional testing and Non-Functional testing
* Logged all defect
* Report progress

1. Track and update the traceability metric
2. Mitigated the risk
3. Artifacts produced

* Test result
* Defect report
* Test summary report
* Updated traceability metrics

1. **User Acceptance Testing**
2. To meet business needs, conduct acceptance testing.
3. Conduct some non-functional testing and compatibility testing.
4. Artifacts produced

* Documentation of UAT results
* Measures of updated business coverage

**List at least 2 disadvantages of each of the Spiral, Agile & V-Models while executing such a large software project, in a tabular format.**

|  |  |  |
| --- | --- | --- |
| **Sl.No.** | **SDLC Models** | **Disadvantages** |
| **1.** | **Spiral Model** | 1. Risk of not meeting schedule or budget 2. Not suitable for smaller or low-risk projects 3. Aspects of project risk analysis may require special skills |
| **2.** | **Agile Model** | 1. Unclear End Product 2. Rely on Team High Commitment 3. Incomplete Documentation |
| **3.** | **V-Model** | 1. Poor model for long, sustainable projects 2. No system can be used until one cycle is complete 3. Not a good model for complex, object-oriented projects |

**Two Advantages of using Waterfall Model in this project scenario:**

1. The waterfall model produces a clear work structure by concentrating on a specific and defined set of phases. Before moving on to the next phase, teams must finish the previous one in its entirety. So, if there are any barriers to completion, they are clearly evident.
2. With the help of this approach, you can easily and intuitively see how the project is progressing. This also implies that no formal training or certification is necessary for project management.