`

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | USN |  |  |  |  |  |  |  |  |  |  |  |  |   **School of Computer Science and Engineering**  **B. Tech (Hons.)**  **Midterm Question Paper – Set 1**  **Academic Year 2024-2025** | | |
| **Course: Agile Software Engineering and DevOps** | | **Course Code: CS2004** | **Semester: IV** |
| **Time: 9:30am to 11am** | **Duration: 90 minutes** | **Date: March 14, 2025** | **Max Marks: 25** |

**Notes/ Instructions:**

1. **Answer all questions**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl. No.** | **PART A – (MCQs) Max Marks (5)** | **Marks** | **L1-L6** | **CO** |
|  | Apply your understanding of Epics, Features, and Stories in Agile to illustrate their relationship?  A. Stories are part of Epics, and Epics are part of Features  B. Features are part of Epics, and Epics are part of Stories  C. Epics, Features, and Stories are unrelated  D. Epics are broken down into Features, and Features are broken down into Stories | 1 | L3 | CO2 |
|  | Apply your understanding of DevOps to identify which of the following is NOT a core component  A. Continuous Integration  B. Continuous Delivery  C. Manual Documentation  D. Automation | 1 | L3 | CO4 |
|  | “Burnt Down Chart” primarily tracks the following  A. The number of bugs found during testing  B. The amount of work remaining and time left to complete it  C. The team's mood throughout the sprint  D. The number of meetings held during the sprint | 1 | L2 | CO2 |
| 4. | Who is responsible for facilitating PI Planning in SAFe?  A. Scrum Master  B. Product Owner  C. Release Train Engineer (RTE)  D. Portfolio Manager | 1 | L2 | CO2 |
| 5. | Identify a key best practice for creating an effective test plan?  A. Ignore project requirements  B. Focus only on functional testing  C. Clearly define the scope of testing  D. Avoid documenting potential risks | 1 | L2 | CO3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl. No.** | **PART B – Max Marks (20)** | **Marks** | **L1-L6** | **CO** |
| **6.** | **a.** Why is the "Definition of Done" critical in Scrum? Give 1 example **(2 Marks)**  **b.** Apply your understanding of the INVEST acronym to explain its meaning for user stories and its significance **(3 Marks)** | 5 | L3 | CO2 |
| **7.** | Consider a scenario where you need to create a new branch named "feature-x" in Git, switch to that branch, make some changes, add the changes to the staging area, commit the changes, and finally push the branch to a remote repository named "origin".   1. Give the sequence of Git commands and their purpose to accomplish this task and **(2 Marks)** 2. You have completed your work on the "feature-x" branch and give the sequence of commands to merge it into the "main" branch **(3 Marks)** | 5 | L3 | CO4 |
| **8.** | Consider a scenario where you have the following tasks:   |  |  |  | | --- | --- | --- | | Task | Cost of Delay | Duration | | A | 60 | 20 | | B | 120 | 30 | | C | 50 | 10 |  1. Explain Weighted Shortest Job First (WSJF) **(1 Marks)** 2. Calculate the WSJF score for each task and rank them in order of priority   **(3 Marks)**  **c.**Describe the key characteristics of an Agile Release Train **(1 Mark)** | 5 | L3 | CO2 |
| **9.** | **a.** Distinguish between integration testing and unit testing in terms of their scope and objectives by giving one example **(3 Marks)**  **b.** Illustrate the concept of non-functional testing by providing two relevant examples **(2 Marks)** | 5 | L3 | CO3 |

Course Outcomes

|  |  |
| --- | --- |
| CO 1 | Evaluate the advantages and disadvantages of Agile development compared to traditional models |
| CO 2 | Assess various Agile methodologies such as Scrum, XP, Lean, and Kanban, and determine their appropriate applications |
| CO 3 | Create software requirements, design specifications, test plan and Analyze test coverage, requirements traceability for a software project |
| CO 4 | Utilize and implement various DevOps tools (e.g., Git, GitHub, Docker) in a software project |
| CO 5 | Develop a mini software project using Agile Scrum methodology, simulating its roles, meetings, processes, and artifacts |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Marks Distribution** | | | | | | | | | |
| **L1** | **L2** | **L3** | **L4** | **L5** | **L6** | **CO1** | **CO2** | **CO3** | **CO4** |
| 0 | 3 | 22 | 0 | 0 | 0 | 0 | 13 | 6 | 6 |