CSE 566 Assignment 4 - Identification and Classification of Risk Factors in Distributed Agile Software Development

Part 1: Hypothetical Scenario

SparkySoftware, an agile software development organization located in Tempe, Arizona, has partnered with a Taiwanese software development company, TaiwanTech, on a new product development project. SparkySoftware is a well-established company with expertise in agile methodologies and a strong track record of successful software projects. TaiwanTech, on the other hand, is a relatively newer company with a strong presence in the Taiwanese market and a growing global footprint.

The new product they are developing is a cloud-based enterprise resource planning (ERP) system that will cater to the needs of small and medium-sized businesses in the United States and Taiwan. The project is expected to have a duration of 12 months, with a distributed team of 20 developers, 5 from SparkySoftware and 15 from TaiwanTech.

1. **Requirement Elicitation:** ***Unclear Requirements in Multiple Development Sites*** The distributed nature of the team, with members located in the United States and Taiwan, can lead to challenges in requirement elicitation. Differences in time zones, cultural backgrounds, and communication styles can make it difficult to gather and align requirements across the team.

2. **Objective Statement:** ***Ambiguity in Objective Due to Cultural Differences*** The team members from SparkySoftware and TaiwanTech come from different cultural backgrounds, which can lead to ambiguity in the understanding of the project objectives. This can be exacerbated by language barriers and lack of face-to-face communication.

3. **Design:** ***Conflicts in Design*** With team members located in different geographical locations, there may be conflicts in the design of the system due to a lack of direct communication and collaboration among the team.

4. **Coding:** ***Lack of Coordination*** The distributed nature of the team can lead to a lack of coordination among the developers, especially when it comes to pair programming and knowledge sharing.

5. **Testing:** ***Unavailability of Real Testing Data*** Due to security and network issues, the team may face challenges in transferring large amounts of testing data across the distributed locations, which can impact the testing process.

6. **Release and Deployment:** ***Difference in Agile Practices and Principles at Different Sites*** The teams at SparkySoftware and TaiwanTech may follow different agile practices and principles, which can lead to challenges in the integration and deployment of the final product.

7. **Project Management:** ***Exceeded Project Time (Lower Initial Velocity)*** The distributed nature of the team and the time zone differences can make it challenging to estimate and manage the project timeline effectively, leading to exceeded project time.

8. **Communication:** ***Lack of Communication Between Client and Team*** The lack of face-to-face communication between the client (SparkySoftware) and the distributed team (including TaiwanTech) can lead to communication breakdowns and misunderstandings.

9. **Technology-Based:** ***Lack of Training*** The distributed team members may require specialized training on the tools and technologies used for the project, which can be challenging to coordinate and provide effectively across the different locations.

10. **External Stakeholder:** ***Poor Coordination Between Multiple Vendors*** The involvement of multiple vendors, including SparkySoftware and TaiwanTech, can lead to challenges in coordinating and managing the project effectively.

11. **Group Awareness:** ***Lack of Trust Between Onshore and Offshore Teams*** The distributed nature of the team and the cultural differences between the US and Taiwanese team members can lead to a lack of trust, which can impact the overall collaboration and group awareness within the team.

Part 2: Approaches to Mitigate the Risks

1. **Requirement Elicitation:** ***Unclear Requirements in Multiple Development Sites Approach:*** Establish regular video conferences and workshops with all team members and stakeholders to facilitate open communication and clarify requirements. Utilize collaborative tools and techniques, such as virtual whiteboards and user story mapping, to ensure a shared understanding of the requirements.

2. **Objective Statement:** ***Ambiguity in Objective Due to Cultural Differences Approach:*** Organize cross-cultural training sessions for the team members to foster mutual understanding and respect for different cultural perspectives. Encourage open dialogues and feedback loops to address any ambiguities or misunderstandings in the project objectives.

3. **Design:** ***Conflicts in Design Approach:*** Implement a robust design review process, with regular design meetings and collaborative design sessions, to align the team on the system design and resolve any conflicts.

4. **Coding:** ***Lack of Coordination Approach:*** Establish a structured knowledge-sharing program, such as regular code reviews and pair programming sessions, to improve coordination and collaboration among the developers.

5. **Testing:** ***Unavailability of Real Testing Data Approach:*** Explore the use of synthetic data generation tools or techniques to create realistic test data that can be shared across the distributed locations, addressing the security and network limitations.

6. **Release and Deployment:** ***Difference in Agile Practices and Principles at Different Sites Approach:*** Ensure that the team members from both organizations receive comprehensive training on the agile practices and principles to be followed. Establish clear guidelines and processes for release and deployment, with regular alignment and review sessions.

7. **Project Management:** ***Exceeded Project Time (Lower Initial Velocity)***

   Approach: Implement effective project monitoring and reporting mechanisms, with frequent status updates and feedback loops, to identify and address any delays or deviations from the planned timeline. Adjust the project plan and resource allocation as needed to maintain the project velocity.

8. **Communication:** ***Lack of Communication Between Client and Team***

   Approach: Establish a communication plan that includes regular video conferences, virtual collaboration sessions, and clear escalation procedures to ensure effective communication between the client (SparkySoftware) and the distributed team.

9. **Technology-Based:** ***Lack of Training Approach:*** Develop a comprehensive training program that covers the tools, technologies, and processes used in the project. Provide both in-person and online training sessions to ensure that all team members are equipped with the necessary skills and knowledge.

10. **External Stakeholder:** ***Poor Coordination Between Multiple Vendors***

    Approach: Appoint a dedicated project coordinator or liaison to facilitate effective communication and coordination between the different vendor organizations (SparkySoftware and TaiwanTech). Establish clear roles, responsibilities, and communication channels to streamline the collaboration.

11. **Group Awareness:** ***Lack of Trust Between Onshore and Offshore Teams***

    Approach: Organize team-building activities and virtual social events to foster a sense of trust and camaraderie among the team members. Encourage open and transparent communication, as well as recognition and appreciation of individual contributions to the project.

References:

1. https://journals.riverpublishers.com/index.php/JWE/article/view/16869