**Subject: Data Mining**

Initial Post 1:

Simple Random Sampling is that every item of the sample is selected from the group that the probability of being selected for all the items is the same. In other words, sampling units are selected at random so that the opportunity of every sampling unit being included in the sample is the same. It is the primary sampling; in simple random sampling, all the items that can be chosen to have an equal probability of occurrence. The two different types of samplings, probability, and non-probability are different in a subtle way. Simple random sampling is a type of probability sampling in which all the items have the same probability of picking up. If there are 1000 items and 20 needs to be chosen, all 20 have the same probability of selection; probability sampling. This method is easy to understand but hard to execute.

It is possible to sample data instances using a distribution different from the uniform distribution. The non-uniform distribution is a different method from a uniform distribution, and Cluster Sampling is an example. Cluster sampling works by clustering a group of items basing on their characteristics; then, the items are selected from the cluster. Clustering is one of the approaches where there is nonequivalent allocation of probability. If a research group wants to collect data of a particular number of people using a cluster sample, they will form clusters of people with particular characteristics and select people from that cluster. Expert Judgement is another way of sampling, in which items are sampled based on expert knowledge.

Initial Post 2:

Sampling Approaches are of several kinds, and the most popular of these is simple random sampling. Simple Random Sampling is one of the most straightforward sample data instances for distribution difference, along with an example of a probability distribution. In the case of data instances, there are different ways to uniform distribution as an equal probability. In this method of sampling, the selection of an item is an equal probability. In this type of random sampling, two variations are – sampling without replacement and sampling with replacement. Every item is chosen and removed from the object set considered as the population in the first type. In the second type, objects remain in the population set. Hence, there is a crucial issue here since the same object can be used for more than one sample. The population includes different types of objects, and simple random sampling will show that a different type of objects where simple random sampling is adequately for types of objects is less frequent. Hence, simple random sampling provides adequate population sets for appropriate data set to be determined and ensuring quality data mining (Etikan, Musa & Alkassim, 2016).

Yes, it is possible to use different distribution in sample data instances for uniform distribution. Problems in terms of analysis are seen for proper representation in all the object types. The building classification models for their classes should be considered critical. Using sample design distances is possible to develop a different distribution from the uniform distribution. For example, in excel, the sample data set can be obtained by creating 20 elements with the standard normal distribution placing the formula in the first cell, A1. The cells' range should be 20, that A1: A20 for providing the distribution (Zhu et al., 2018).

**Subject: Operational Excellence**

Initial Post 1:

Software developments models describe the phases of the software cycle and order; it also helps determine system requirements and the overall system architecture, this process takes care of the developed product in the shortest possible time and reduces its business impact to the customer. Team members are also the developers who design, program, and deploy the system (Rajkumar et al., 2016).

The formal method describes the steps that are taken to achieve a result. System integration results in an end product

This procedure is also known as the construction or encapsulation of the solution. The building blocks of software technology are usually assembled or put together. It gives the company a complete, high-quality, and safe, functional product. The "soft process" is the formal method that builds functional software applications for integration with business applications. When a customer delivers an integration solution to the integration laboratory, the laboratory performs a formal process to SOURCE the requirements. In Software Verification, the software development and integration process include more than just putting software together. The software development process and the software integration process are the completion of the final functional software application. This "complete" software application, the complete high-quality end-user application, allows the application to be deployed, managed and used by a user. This "final" software application would need to be tested for safety, performance, installation, and end-user support to make any software specification or legal statements. The users, developers, and auditors involved in implementing and maintaining the "final" software application would be taking the software specification and internal testing and validation information to a new level (Yilmaz et al., 2016).

Initial Post 2:

The Process of New Software Development: The process of software development is complex (Clarke, O'Connor & Leavy, 2016). Any mistake in the process can cause inevitable outcomes in the entire business or product quality. It is a lengthy process requiring step-by-step techniques. The process of software development involves:

Planning: In this process, the specialists gather the right requirements while analyzing the software's aspects, even in the future. Hence, developers must understand the client's requirements, the exact need, and the issues during software development. Communication is critical between the project teams, stakeholders, and users.

Software analysis: Here, the development team define the project and check its feasibility. It divides the workflow into segments such that the developers, designers, testers, and managers can evaluate their tasks effectively. The software is defined in terms of its reliability, time, functioning, and cost (Al-Zewairi et al., 2017).

Software designing: The design of the software should be clear and creative. Designing involves the overall product design along with database design and data structure.

Software programming: This stage is the most critical process of software development. Many brains are put into work for coding to deliver the desired software.

Implementation and integration: The software contains various programs that need careful implementation and software product integration that should be step-by-step. In this process, the developers check whether the software runs in multiple systems. In case they find bugs, they should fix them (Clarke, O'Connor, & Leavy, 2016).

Software testing: After coding completion, the software is tested. The quality of the software is tested as well as its performance. If the software is error-free, it proceeds to the next process.

Installation and maintenance: This is the final process. Clients can now install the software on their devices. In case the client needs software modification, the process of maintenance begins.

One of the main differences between software development and methodologies is the phase's sequence in which software development is being completed (Al-Zewairi et al., 2017). Methods use the linear approach. Processes must be done sequentially.

Initial Post 3:

Differences between software development and methods. Software development:

Software development is a collective and iterative logical technique concerned with creating a computer coded or programmed software to address a unique business or personal objective, purpose, or process. It is a planned initiative that consists of more than a few steps or ranges that result in the creation of operational software. The software program's software development is executed via programming and includes countless methods such as preliminary research, data glide design, method glide design, flow charts, technical documentation, software program testing, debugging, and other software architecture techniques. It is referred to as the Software Development Life Cycle (SDLC).

Software Development methods:

To design software to meet the business need, the software program's improvement to meet the precise graph, and software deployment to production, SDLC includes some methodology. The software development methodology is a procedure or collection of strategies used in software program development. (Kleidermacher & Kleidermacher, 2012). In the modern-day world, quite a few methodologies exist for developing the software; generally, it takes the structure of described phases. It is designed to describe the show of the existence cycle of a piece of software. The Waterfall model, Rapid Application Development model, Component-based development, The formal strategies model, Agile model, Spiral model, and Prototype model.

The waterfall model, the unique SDLC method, is linear and sequential, commonly following these degrees: Identify required software, analysis of the software requirements (Mayer, J. 2005), Detailed specification of the software requirements, Software design, Programming, Testing, Maintenance.

Numerous models are frequently mixed into some hybrid methodology, as is the case with open-source software development (OSSD).

Initial Post 4:

The overall process of developing software is based on the requirement, then the feasibility, then the selection of methodology to adopt for the software development, and there will be various stages like requirement intake, development, implementation, and testing phase for the successful delivery of the software to the client. Successful projects are managed well. To effectively deal with a project, the manager or development team must pick the software development methodology that will work best for the current project. All methodologies have various qualities and shortcomings and exist for various reasons (Langer, 2018).

Agile development methodology:

Teams utilize the agile development methodology to limit risk (for example, bugs, cost overwhelms, and evolving necessities) while including new usefulness. In every single agile technique, teams build up the software in iterations that contain scaled-down additions of the new usefulness. There is a wide range of types of agile development strategies, including scrum, crystal, extreme programming (XP), and feature-driven development (FDD) (Langer, 2018). The essential benefit of agile software development is that it permits software to be discharged in iterations. Iterative discharges improve effectiveness by permitting teams to discover and fix defects and adjust their desires early. They likewise permit clients to acknowledge software benefits prior, with gradual visit improvements.

DevOps deployment methodology:

DevOps is a development methodology and a set of practices that support a traditional culture. DevOps deployment focuses on hierarchical change that upgrades coordinated effort between the offices liable for various development life cycle portions, for example, development, quality confirmation, and tasks. DevOps is centered on improving time to advertise, bringing down the disappointment pace of new discharges, shortening the lead time among fixes, and limiting disturbance while expanding unwavering quality. To accomplish this, DevOps associations intend to computerize continuous deployment to guarantee everything happens quickly and dependably. Organizations that utilize DevOps methods benefit by fundamentally diminishing time to advertise and improve customer satisfaction, item quality, and representative profitability and efficiency (Langer, 2018).

Initial Post 5:

Developing the new software process starts with the strategic planning and gathering of required resources. Many software development tools are available in the modern business era that better assist the critical requirements needed for developing new software. Before developing the new software, it is essential to understand the right process, which suits the business. Since technologies in recent days are drastically changing, it is essential to work on the latest business trends and develop the software accordingly. The following are the key steps necessary while developing the new software: (Silva & Gualotuna, 2018).

* Analysis and strategic planning
* Gathering of requirements
* Design and prototype
* Software development process
* Testing the plan thoroughly
* Deployment process

Maintenance and work on the updates:

While developing the new software, managers must ensure that they are going through the right strategic planning and target the business goals. Following the best communication channels and keep all the staff on the same page is another important strategy in the new software development process to avoid communication breakdown issues and other concerns. Management ensures that they provide adequate resources to develop the best software and control the unproductive activities to minimize the expenditure (Silva & Gualotuna, 2018).

Software development is a series of activities that include creating, designing, deploying, and supporting the software. While developing the software, it is essential to select the methodology, which gives the perfect outcomes and utilize the resources optimally. The software development method sets up the business's strategies and framework and ensures that the outcomes will be positive. To utilize all the resources optimally in the software development process is essential to work on effective methodologies that give positive outcomes and keep control over the unwanted cost (Wasserman, 2016).

Initial Post 6:

The System Development Life Cycle is the methodology for developing new software. This process is very much structured and is designed on the risk-averse for managing the projects, which include the multiple systems and programmers which impacts the organization at large; The phases of the SDLC are:

Preliminary Analysis: This stage reviews the client's request so that the possible solution could be developed and checks the alternatives too. This stage examines the feasibility of the software as per the organization or client requirement. The technical, economic, legal feasibility is checked under this stage (Sharma & Singh, 2015).

System Analysis: In this stage, the system analysts get together with the stakeholders for determining the particular requirements of the new software. Here, the process is documented, the team is interviewed, and data requirements are developed to understand what the system is expected to do.

System Design: In this stage, the designer or the developer uses the documents of the system requirements, which were developed in the last stage, and then the technical details are developed, which is needed for the system. Here, the requirements of the business are translated to the technical requirements. Designing the database, user interface, reporting, input, and output of data is done.

Programming: Here, the coding is done, as per the document of the system design the developer creates the software.

Testing: In this phase, the software developed is tested with specific structured tests. Unit Test is done for coding regarding the bugs and errors. Then the system is a test that tests the system components. Then the test of user acceptance is done for ensuring the system standards.

Implementation: Once the software is tested, it is implemented in the client's system or organization. Here, the users are trained, documents are provided, and conversions are made on the system as per the new software requirements (Susko & Barisic, 2020).

Maintenance: This is the final stage, where the bugs are fixed after implementation, if any.

The SDLC methodology is known as the Waterfall Methodology for representing how every step is separated in software development.

**Instructions**:

* Total of 8 Responses
* Minimum of 200 words for each response
* Minimum of 2 APA References for each response
* Responses should cover:
	+ Ask a captivating, thoughtful question about the topic.
	+ Provide extensive additional information on the topic
	+ Explain, define, or analyze the topic in detail
	+ Make an argument concerning the topic.
* Please follow the format below:
* Response 1, References
* Response 2, References
* .
* .
* Response 8, References