**SOFTWARE DEVELOPMENT LIFECYCLE (SDLC)**

The **Software Development Life Cycle (SDLC)** is like a roadmap for building software, guiding teams through each step from start to finish. The SDLC is a well-defined process of constructing software solutions that will enable the realization of a structured procedure from the initiation phase up to the deployment phase. An Explanation of how a Requisition Management System can be developed using this SDLC stage and applies to the Requisition System Assesment\_PartB is presented below.

**1. Planning**: Think of this as laying the foundation. The team figures out what the project is about, what it needs to achieve, and the resources and time required.

**2. Requirement Analysis**

**• Goal:** Know what is required from the system.

Acquiring requirements of the requisition system like recording information about the staff, handling of the requisition request, estimations of the costs, approval or rejection of the request, and providing the statistics.

**For the Requisition System:**

Requirements would include a method of entering staff information and items by which costs are related. A system through which all the requisitions in the course of an organization can be captured and retrieved. A method for automatically green flagging or pending flagging requisitions beyond, or below, a particular cost ceiling.

**3. System Design**

• **Goal**: Organize within the design the system architecture and layout, and how the specified requirements will be met.

• Activities: the structures for the requisitions, approval processes, and database structure if necessary in addition to the flow, the user interface if it is a web or desktop application, and the logic for the requisitions and approval processes. Explain on how it will deal with staff details, how it will take requests, and how approvals will be done.

**For the Requisition System:**

• Establish a class known as RequisitionSystem, to respond to user information input, performance of calculations, and granting of approvals. Decide whether to use a basic data structure such as a simple list or an array in memory (similar to the Python list) or persist it to the external storage system such as a database or a file. Decide on the format to be used to indicate the status of requisitions, whether it is approved or still pending, and how the information on the staff is to be stored.

**4. Implementation (Coding)**

**• Goal:** To write the actual code according to specified design documents.

Some of the acts that should be performed include; translation of the design into code. Ensure that coding standards and best practices are followed as each function is developed and test the function.

**For the Requisition System:**

• Design the class RequisitionSystem that shall have methods that will manage staff info, process requisition, and generate stats.

• Take an evaluation of how the sistem is capable of handling multiple requests from different users or where requisitions should be pending or not approved.

**5. Deployment**

**• Goal:** Put it out into the field and make it usable.

Since the system is to be used, it should be fitted in the environment of use. Provide documents (handbook) and if need be orient the users.

**For the Requisition System:**

• If the system is standalone, build it as a script that users will be able to execute.

• If it is a web or a desktop application, then upload it on a server or desktop.

• If it’s a document it should be made available to the appropriate employees/staff and other interested parties.

**6. Maintenance**

• Goal: Ensure the software is always active and at the same time apply for updates as may be required.

Address any issues that occur after the software has been deployed in an organization. Sometimes it is necessary to add new modifications or improvements to the software (for example creating a connection with a database or including a login).

**For the Requisition System:**

• They may need more options such as export requisitions or work with approval processes or can save data permanently. Add-ons may involve changes in the approval procedure, or the criteria used to set auto-approval for an order.

**7. Testing:**

In this stage, unit, integration, system, and user acceptance testing will be done to check on system quality.

**Requisition System Problem Solution**

In this requisition system, SDLC assists by providing a framework or procedures to follow hence coming up with well-developed software solution that is competent. It is a step-by-step approach to planning and executing the work and makes sure none of the vital activities like, for example, requirement gathering, system designing, or testing, can be omitted. Storing requisitions in the memory did not happen which would have been beneficial but using a database rather than memory helps in persisting the data sessions. There is always a way to at least make a web as simple as ‘enter the information of requisition to order from staff’, which should be helpful in increasing user experience.