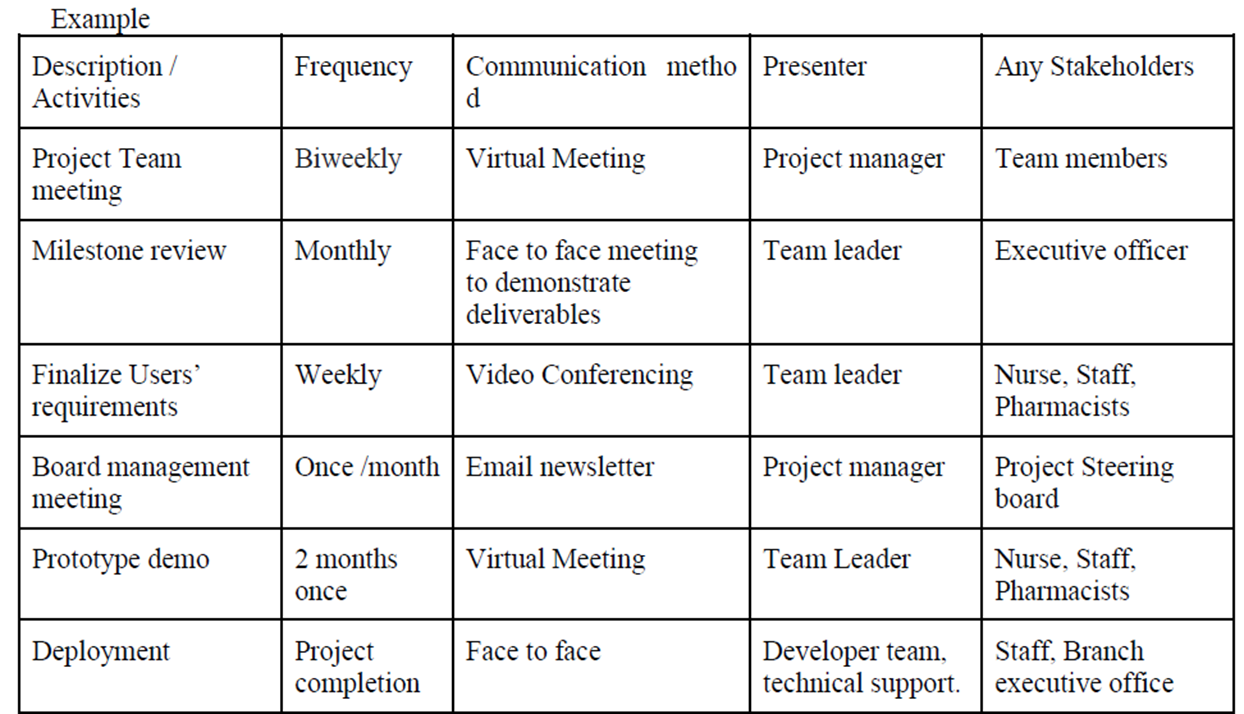
**Question 1**

a. Communication Plan in a table format

Activity, Frequency, Meeting Method (virtual, face to face, email), Participants, Stakeholder

b. Resource Management (Resource loading, resource leveling & resource assignment)

* Resource assignment - Before assigning a staff into a task, need to research his/her background (skills, knowledge, specialization) so the right person can be assigned to the right task
* Resource loading - Ensure the staff who work for particular sub task do how long the duration needed to avoid overallocation

Sample answer:

* **Resource Loading** 
  + Number of individual resources an existing schedule requires during specific time periods.
  + Help project managers and individuals to develop schedules.
* **Resource Leveling** 
  + A technique for resolving resource conflicts by delaying tasks.
  + Create a smoother distribution of resource usage.
  + Project managers examine the network diagram/PERT chart/CPM for areas of slack or float, and to identify resource conflicts.
* **Resource Assignment** 
  + To assign personnel to their projects or to acquire additional human resources needed to staff the project.
  + Use a weighted score model to acquire the right staff.
  + Project managers with strong negotiating skills are often good at getting internal people to work on their projects.

c. One possible issue if increase the number of inspection

* They feel stressed due to very tie inspection and cause **low productivity**
* **Delay the project** as the number of inspections increase and want the staff to make changes accordingly thus the time consumed will be longer.

Sample answer:

* May caused the delay on the project when the number of inspections increases to guide the staff as they need to rework the defects.
* The staff may feel stressed, low motivation. May cause low productivity on their work which can cause a project to be delayed.
* It may cause an interruption in work deliverable for staff, find out errors and correct it to ensure errors are discovered earlier and rectified before proceeding to the next stage.

**Question 2**

A.

Product operation factors − Correctness, Reliability, Efficiency, Integrity, Usability.

Product revision factors − Maintainability, Flexibility, Testability.

Product transition factors − Portability, Reusability, Interoperability

**Correctness** - The ability of a system to perform its intended function.

Measurement :

* Defect per KLOC - The number of defects/errors identified during the development divided by the size of the medicine dispensing system. The lower defect/KLOC means the medicine dispensing system has less defects.

**Usability** - It refers to how easy of a system to be used by the end user.

Measurement :

* The time required for the end user to learn how to perform a task for the first time they use the medicine dispensing system. The lesser time taken by the user, it means that the system has high usability in which users can familiarize themselves with the system’s function easily and quickly.

b.

* Maintainability relates to the ease and time/ efforts taken which the cause of an error can be diagnosed and then be fixed by developers. How much time needed to identify and fix the bugs.
* From the point of view of the developer management, mean effort to correct / repair would be a good measure.
* In this case it would be (4+4+3+2hours)/4 days i.e., 13/4 or 3.25 hours on average. To evaluate how good this might be, it would need to be compared with the figures for other software components.
* Another measurement that would need to be taken is the percentage of time that the developers are actually working on corrections and enhancements.

c. **Appraisal cost** - Need to explain how appraisal cost is being applied to this project (i.e quality review for testing) & How do you carry out inspection to enhance quality ? (i.e detect defect during testing)

Sample answer:

* This belongs to Appraisal costs are those associated with inspection and testing of both the company’s own products and the products received from suppliers.
* On software projects, the appraisal costs included testing, Fagan inspections and design reviews. As with prevention, make appraisal cost investments early in the development lifecycle.
* Indication on more resources to appraisal, the number of defects in the end-product is drastically reduced.
* For example, the more design reviews carried out during system design– the greater the chance of spotting and correcting defects on the design before the programming process begins as we can save the cost due to the incorrect designs.
* Defects in ERD models are much easier to fix and cheaper before the implementation of DBMS. The earlier we identify the defects, the cheaper the cost to fix. It will cost more to fix at the later stages especially during implementation stages.

**Question 3**

a.

* **Goal** : To increase the staff satisfaction
* **Question 1**:
  + How much time for the medicine dispensing process to be carried out?
* **Software Metrics 1** :
  + Time spent to process and complete the medicine dispensing requests
  + The number of task to complete in order for the function to be performed
* **Question 2** :
  + How many requirements are fulfilled by the staff?
* **Software Metrics 2** :
  + Number of function that fulfill the system output
  + Number of test case is used to be tested whether they are fulfill the system specification or not

Sample answer:

Goal: Increase Staff Job Satisfaction by using the new systems

Question 1: Will the medicine dispensing process be completed within a short time? Any improvement compared with the previous practice, does the new systems help to make the staff work to become more productive

Metrics:

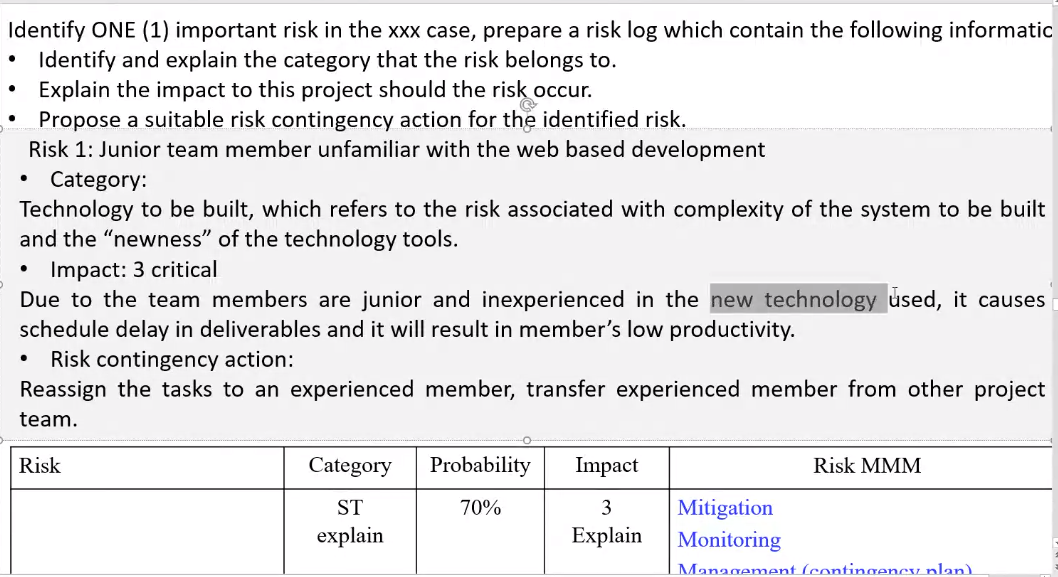
* Number of clicks for input, auto retrieval information based on relevant input for example input the patients ID to retrieved others information's
* Time taken to complete dispense requests.
* The number of errors found during medicine dispenses.
* The average waiting time for getting medicine.
* The accuracy of information generated in medicine collection.
* All these metrics will determine how productive the staff, how efficient they are

Question 2: Do the functions provided by the systems meet the nurse or staff’s needs?

Metrics:

* Number of clicks or steps to perform functions.
* Message, help, or guidance provided when the user clicked on the wrong option.
* Rate of satisfaction in terms of the functions performed as requested.
* Number of functions to complete the processing or request.
* Usability rate in terms of the ease of use for the functions (system)

b. First risk:



Sample answer:

Team member unfamiliar with web-based development tools C# and MySQL

* Category: Technology to be built, which refers to the risk associated with complexity of the system to be built and the “newness” of the technology , C# and MySQL, one is proprietary technology from Microsoft and the others is open source , connecting C# web applications to MySQL database
* Impact: If the team members are inexperienced in the new technology used to develop the system and do not get sufficient training, it will result in lower productivity, requiring more time to learn and build the systems.
* Risk monitoring: send staff for training, peer guidance support is needed. Team leader gives supervision to them.

Team member’s productivity is low.

* Category: Staff size and experience, which refers to the risks associated with the overall technical and project experience of the software engineers who will do the work. Shortage of experienced staff needed to complete the projects.
* Impact: staff may be inexperienced in the particular technology such as C# and MySQL used to develop the system and do not get sufficient training, it will result in lower productivity levels.
* In addition, the increasing frustration with the project due to their struggles with the technology may lead them to resign. Both of these will cause schedule delays.
* Risk monitoring: review staff performance by checking work deliverables, increase the number of inspections where possible before the deadline.

**Question 4**

a.

* **Fault avoidance** , try and catch block during run time errors
  + Development techniques are used to either minimize the possibilities of mistakes and/or trap mistakes before these results in system fault.
  + e.g., avoiding error-prone programming language constructs like pointers, use static analysis to detect program anomalies.
* **Fault detection**
  + Verification and validation techniques are used to discover and remove faults in a system before it is deployed for operation use.
  + e.g., implement systematic system testing and debugging. , unit testing, integration testing, systems testing, BBT, WBT.
* **Fault tolerance** 
  + Use techniques that ensure that faults in a system do not result in system errors or failures.
  + The incorporation of self-checking facilities in a system and the use of redundant system modules.
  + e.g., scan medicine code, should have redundant hardware, RAID systems, backup servers.

b. 8.3 Achieving Dependability (few strategy there)

* Avoid the introduction of errors when developing the system.
* Design verification & validation processes that are effective in discovering errors in the system. Detect the design on the models first UML, ERD for example prior to implementation or program development stages
* Design systems to be fault tolerant so that they can continue in operation when faults occur. The systems must be designed with a fault tolerance plan such as RAID, backup plan.
* Design protection mechanisms that guard against external attacks. Design must take into consideration security, secured software design.
* Include system capabilities to recognize or detect and resist cyber-attacks such as DDOS attacks.
* Include recovery mechanisms to help restore normal system service after a failure. Backup and recovery plan.

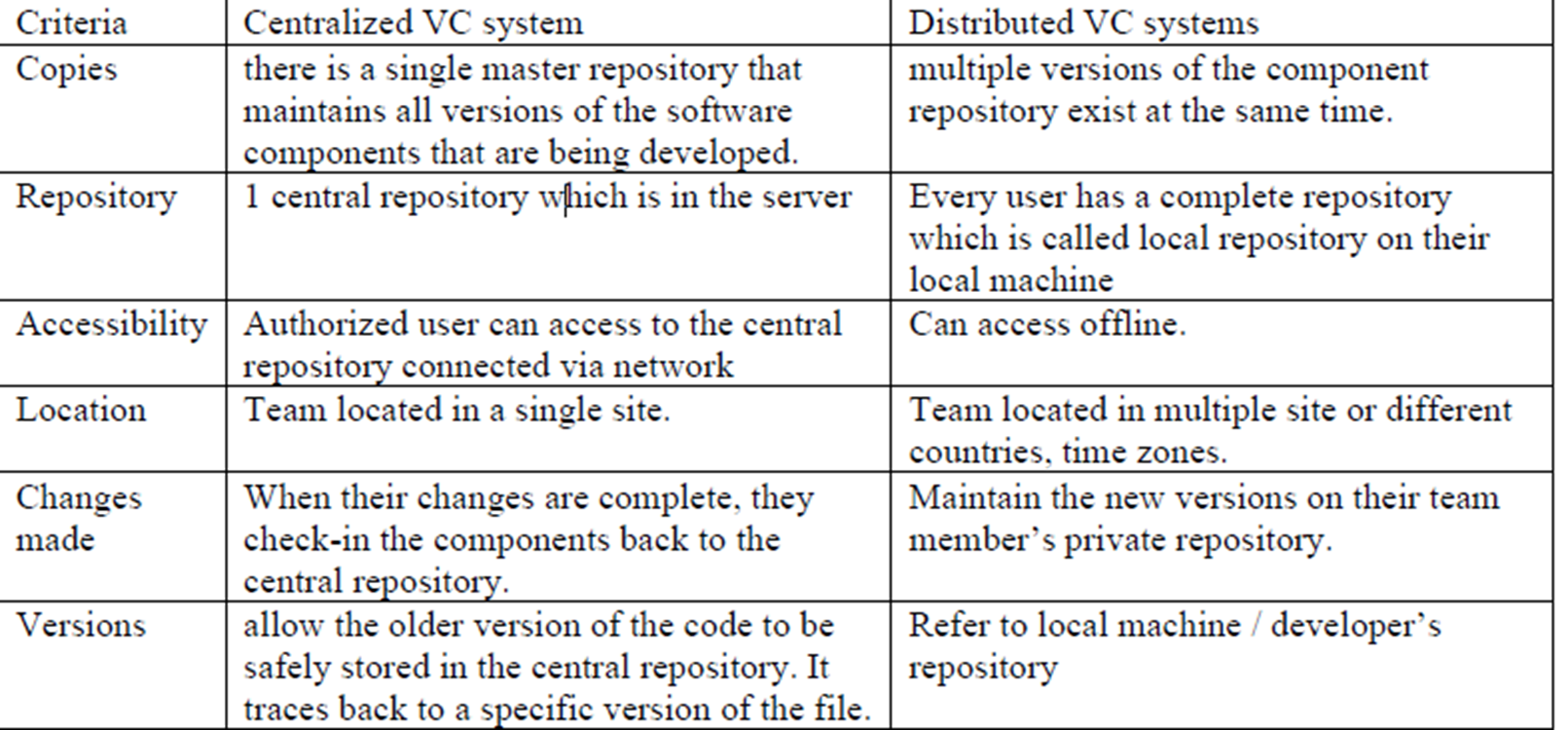
c. Repository storage management

Centralized system - Keep track different version of code that saved by different developers

Distributed system - multiple developer, how they control their version

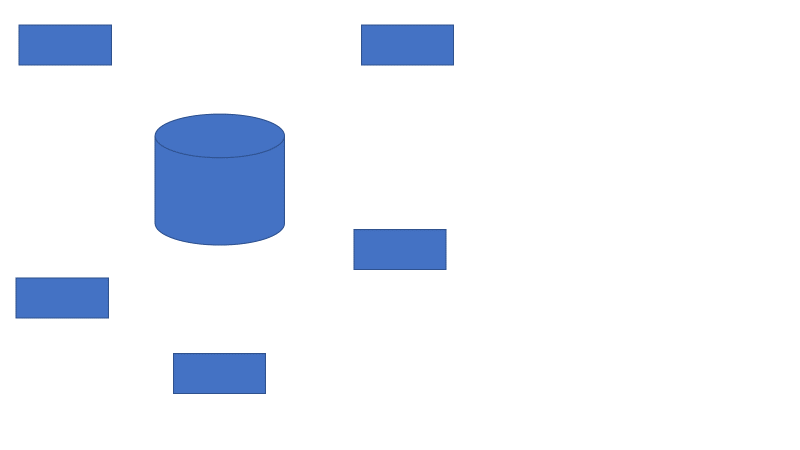
| Distributed system | Centralized system |
| --- | --- |
| Has a master copy |  |
| Developer clone the repository |  |

Sample answer:



The difference between centralized VC system and distributed VC system is the centralized VC system keeps all the history changes in a single master repository in which it maintains all versions of the software components that are being developed. This means all the team members that share the remote central server are also able to share everyone’s work (i.e source code being checked in). While for the distributed VC system, it can keep multiple versions of the repository at the same time.

**Centralized VS :**



**Distributed VS :**

