

### Assignment - 3

1. 1. Unified approach combines the best practices of various object-oriented methodologies, focusing on a robust, layered, and component based architecture. It emphasizes reuse and a clear separation of concerns.

2. Layered - object oriented design: This design separates the system into various layers, ensuring that changes in one layer don't break others, making the system easier to maintain and scale.

3. User Interface (Presentation layer): Deals with how the user interface with system. eg. displaying course lists, forms for submission.

Business Logic (Domain layer): Contains core rules and logic of LMS. eg. enrolling a student in a course, calculating grades.

Data Layer (Persistence layer): Manages all interaction with database. eg. ~~storing~~ <sup>saving</sup> user profile, updating grades, retrieving course material.

3. Inheritance: How class can inherit attributes and method from parent class. This promotes code reuse.

Encapsulation: Hiding internal working of an object and exposing only necessary methods.



Reuse: The goal of O design, achieved through inheritance and creating well-designed components that can be used multiple times through the system.

2.1. UML Structural Diagram: This diagram shows the static structure of the system including its parts and their relationship.

- **Classes Diagram:** Show the classes, their attributes (data), operations (method) & the static relationship between them.
- **Component Diagram:** Show the structural relationship b/w the component of system.

2. Key Classes

- **Booking** → Booking ID, User ID, Room ID, checkIN Date, Check Out Date, Total Cost, Create Booking(), Cancel Booking().
- **User** - User ID, name, email, password, login(), view profile, update Profile().
- **Room** - Room No., type, Price Per Night, is available, check reserve room, update Price.

3. This diagram is created for visualisation of the system, which greatly improves understanding and communication among team members.

3.1. **Hardware and Software Reliability:**  
The probability that the Physical components (sensors, cameras, boards) will operate without failure for a specified time.



under stated condition.

Software Reliability: Probability that software will operate without failure for specified time under stated conditions.

## 2. Software faults:

Fault 1: Incorrect sensor Rolling Rate logic

Failure 1: Motion is detected, but the alerts trigger is delayed or missed completely.

Fault 2: Improper exception handling on camera stream failure.

Failure 2: System crashes when the camera connection is lost, preventing other sensors from working.

Fault 3: Off by one error in Timer/Scheduling code  
Failure 3: The system incorrectly schedule the "Armed" state, causing false alerts at non-scheduled time.

## 4. 1. Logistic Poisson Reliability Model (LPRM): Non-homogeneous Poisson Process model used for estimating the cumulative <sup>no. of</sup> failure or the failure intensity over time during the Testing phase.

- It assumes that the failure intensity decreases as faults are found and fixed, following a logarithmic pattern. This is often more realistic than models assuming a constant rate.

- This model estimates future failure intensity based on

the initial

2. CMM (Capability Maturity Model) describes a process

Level 1 (Initial)

Level 2 (Repeatable)

Level 3 (Defined)

and integrated

Level 4 (Managed)

processes

Level 5 (Optimizing)

ISO 9001

ISO 9000

requirements

It is a

CMM: Process

development

For health

appropriate

components

the entire



the initial intensity and no. of failure observed.

2. CMM (Capability Maturity Model): It is a functional that describes a path for improving software process maturity.

Level 1 (Initial): Processes are chaotic

Level 2 (Repeatable): Basic Project management Process are estimated

Level 3 (Defined): Processes are documented, standardized, and integrated.

Level 4 (Managed): Quantitative measures are used to control processes and products.

Level 5 (Optimizing): Continuous process improvement is enabled.

3. ISO 9001 VS CMM:

- ISO 9001: A quality management standard that specifies requirements for a quality management system. It is a generic standard applicable to any industry.

CMM: Process improvement model specifically for software development.

For health-critical system ISO 9001 is arguably more appropriate as a foundation standard. It provides a comprehensive framework managing the quality of the entire organization, not just the development process.