**Discuss with your team Advantage and Disadvantage of these life cycle model. and which project ?**

1. **BUILD AND FIX MODEL:**

|  |  |
| --- | --- |
| **Advantages** | **Disadvantages** |
| 1. Requires less experience to execute or manage other than the ability to program. 2. Suitable for smaller software. 3. Requires less project planning. | 1. No real means is available of assessing the progress, quality, and risks. 2. Cost of using this process model is high as it requires rework until user’s requirements are accomplished. 3. Informal design of the software as it involves unplanned procedure. 4. Maintenance of these models is problematic. |

=> This model is suitable for small projects or prototypes where speed is more important than long-term maintainability or scalability. It's often used in situations where requirements are not well-defined or are likely to change frequently. However, it's not recommended for large, complex projects where a more structured approach is needed to manage complexity and ensure long-term success.

2. RAPID PROTOTYPING MODEL:

|  |  |
| --- | --- |
| **Advantages of Rapid Prototyping** | **Disadvantages of Rapid Prototyping** |
| 1. Speed: Quick product development with 3D printing. | 1. Material Suitability: Limited to printing metal and plastic parts. |
| 2. Automated Testing: Utilizes CAD/3D modeling software for virtual mechanical tests. | 2. Not Suitable for Assemblies: Only produces single components. |
| 3. Low Development Costs: Simplified workflow reduces overall costs. | 3. Expense: Initial investment in equipment and materials can be costly. |
| 4. New Testing Opportunities: Allows for experimentation with materials and geometries. |
| 5. Discovery of Design Problems: Identifies issues early in the production process. |

=> Rapid prototyping is particularly suitable for projects where quick iterations, testing, and validation of designs are crucial

**3. INCREMENTAL MODEL:**

|  |  |
| --- | --- |
| **Advantages of Incremental Process Model** | **Disadvantages of Incremental Process Model** |
| 1. Breakdown of tasks facilitated by divide and conquer approach. | 1. Requires thorough planning and design. |
| 2. Lower initial delivery cost. | 2. Costlier than the waterfall model. |
| 3. Incremental resource deployment. | 3. Requires a complete and clear definition of the system. |
| 4. Errors can be detected early due to the use of core modules. | 4. Increased complexity as each increment is added. |
| 5. Suitable for projects with known upfront requirements. | 5. Higher overall project costs due to multiple planning and testing phases. |
| 6. Suitable for projects with lengthy development schedules. | 6. Difficulty in tracking overall project progress. |
| 7. Suitable for projects involving new technology. | 7. Increased communication overhead due to multiple development teams. |
| 8. Suitable for projects with specific needs like funding schedules. | 8. More time spent on testing due to multiple testing phases. |
| 9. Generates working software early in the lifecycle. |  |
| 10. Flexibility to change scope and requirements. |  |
| 11. Each iteration is easily managed. |  |
| 12. Risk management facilitated by iterations. |  |

=> The Incremental Process Model is suitable for projects where requirements are well-defined but may evolve over time, and where it's beneficial to deliver working software incrementally such as an website,..

**4. EXTREME PROGRAMMING:**

|  |  |
| --- | --- |
|  |  |
| **Advantages** | **Disadvantages** |
| Error avoidance through pair programming | Relatively high costs |
| No overtime, teams work at their own pace | Requires version management |
| Changes can be made at short notice | Requires self-discipline to practice |
| Code is clear and comprehensible at all times |  |

=> Extreme Programming (XP) is a software development methodology that emphasizes teamwork, communication, simplicity, and feedback. It is particularly suitable for projects that require rapid development, flexibility to accommodate changing requirements, and a high degree of collaboration

**5. COMPONENT-BASED SOFTWARE ENGINEERING:**

|  |  |
| --- | --- |
| **Advantages of CBSE** | **Disadvantages of CBSE** |
| 1. Reusability: Components can be reused in multiple projects, saving time and effort in development. | 1. Compatibility Issues: Ensuring compatibility between components from different sources can be challenging. |
| 2. Faster Development: Development time is reduced by utilizing pre-built components, leading to faster time to market. | 2. Limited Flexibility: Customizing third-party components may be limited, leading to constraints in functionality or design. |
| 3. Higher Quality: Components are thoroughly tested and validated, enhancing the overall quality of the software. | 3. Dependency Management: Managing dependencies between components can be complex and require careful planning. |
| 4. Cost-Effectiveness: Reusing components reduces the cost of development and maintenance over time. | 4. Lack of Control: Dependency on third-party components may lead to a loss of control over critical aspects of the software. |
| 5. Scalability: Components can be easily scaled and integrated into larger systems as needed. | 5. Learning Curve: Developers may require training to effectively utilize and integrate components into their projects. |
| 6. Maintenance Ease: Updates or fixes to components can be applied uniformly across multiple projects. | 6. Vendor Lock-in: Dependence on specific component vendors may limit future flexibility and options. |

Projects Suitable for CBSE:

1. Enterprise Software Systems: Projects involving the development of large-scale enterprise software systems, such as customer relationship management (CRM) or enterprise resource planning (ERP) systems, can benefit from CBSE. The reusability of components allows for faster development and easier maintenance of complex systems.
2. Web Application Development: Web applications often consist of various modules or components, such as user authentication, data processing, and user interfaces. CBSE can streamline web application development by leveraging pre-built components and libraries, leading to faster deployment and improved quality.
3. Embedded Systems Development: Embedded systems, such as those found in automotive, aerospace, or consumer electronics, often require efficient use of resources and adherence to strict performance requirements. CBSE enables developers to leverage optimized components for common functionalities, reducing development time and ensuring reliability.

**Discuss with your team Advantage and Disadvantage of these approaches**

* + - 1. Traditional system development life cycle:

| **Advantages of SDLC** | **Disadvantages of SDLC** |
| --- | --- |
|  |  |
| 1. Efficient with regard to costs | 1. Project may take longer and cost more without proper planning. |
| 2. Efficacious in terms of time | 2. Correcting code problems can cause delays if numerous. |
| 3. Enhances teamwork and coordination |  |
| 4. Defines suitable roles for employees |  |
| 5. Increases workplace transparency |  |
| 6. Minimal danger when project is implemented |  |

Exercise:

**5.6** What did you do in software projects? recently? It is a commercial project & a project for studying or human resources project. Please write a model development life cycle that you did. It is guaranteed to have structure or not intended. Do you make models different if you start this project again.

* At the previous time, I created a website name CDM which is stand for Car Dealership Management, it built to mange car dealership include 3 roles: Manager, Staff and Customer.
* It’s just a final project in a course in University.
* We used Scrum/Agile for this project, we also used Jira to manage, monitor the project progress.
* It’s guaranteed.
* No, Scrum/Agile is one of the most popular SDLC that implemented at the business now.

**5.7** You are an IT manager and have responsibility with a large system, but unfortunately, it error on Monday morning. How can you handle it if:

1. You just have only 2 hours.

2. If the library has compatible functionality for a few days without system software

ANSWER:

* + - * Identify the exact problem: Check and determine the specific cause of the unexpected error in the library system. You may contact the technical support team or experts within the organization to analyze and identify the root cause
* Assess the impact: Evaluate the extent of the error's impact on the organization's operations. Determine whether it affects end-users' ability to work and the urgency of the repair.
* Implement temporary measures: If possible, implement temporary measures to minimize the error's impact on the organization's daily operations. For example, provide alternative means or manual processes to continue work.
* Contact software vendor or technical partners: If necessary, contact the software vendor or technical partners to seek assistance and guidance on how to fix the issue.
* Execute repair plan: Based on the analysis of the cause and impact, develop and execute a specific repair plan to address the error as quickly as possible without causing new issues.
* Notify and update stakeholders: Inform stakeholders about the status of the error and the actions being taken to rectify it, including end-users and other departments within the organization.
* Test and evaluate: After the repair, conduct testing and evaluation of the system to ensure that the issue has been fully resolved and no secondary issues have arisen.
* Learn and improve: Use the experience from this incident to improve system management processes and prevent similar incidents in the future.