The Rapid Application Development (RAD) model is a software development methodology that emphasizes rapid prototyping and iterative development cycles.

It is a linear sequential approach to software development that is designed to provide a faster development cycle compared to traditional Waterfall methods.

Phases of RAD model:

1. Requirements planning: In the first phase of the RAD model, the requirements of the software system are gathered from the customer or end-user. This is done through interviews, workshops, and other techniques that help to identify the needs and objectives of the software system.

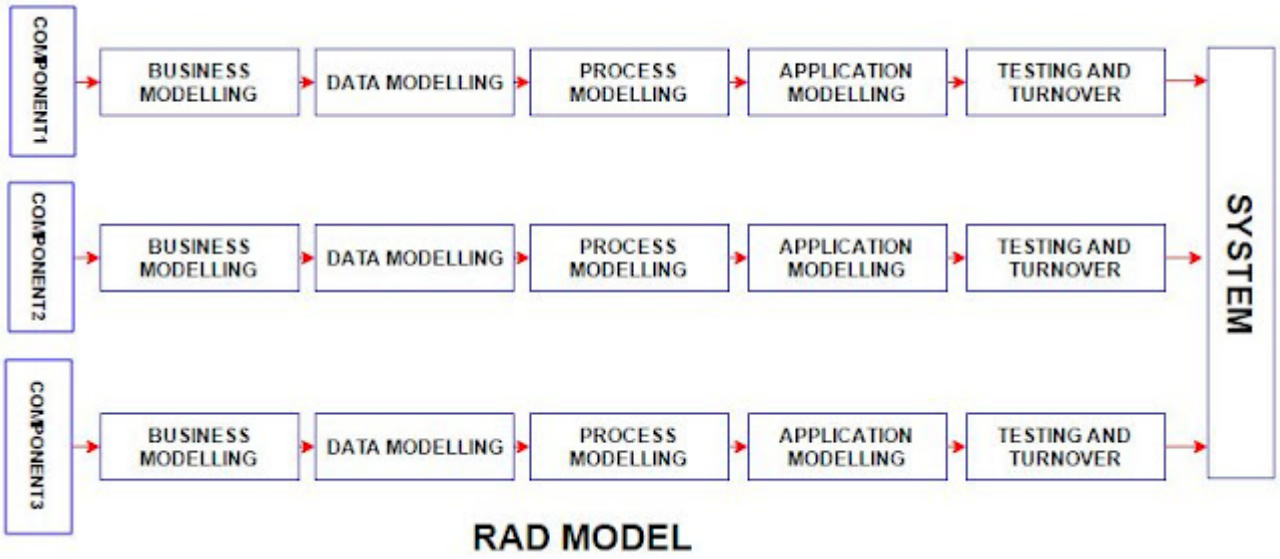
2. Rapid prototyping: In the second phase, a quick prototype of the software system is developed based on the requirements gathered in the first phase. This prototype is developed using a visual programming tool or a high-level language that can quickly generate working code. The prototype is used to demonstrate the functionality of the software system to the customer or end-user.

3. Iterative development: The third phase involves a series of iterations in which the prototype is refined and expanded until it meets the customer’s requirements. This is done through frequent feedback from the customer or end-user. The development team works on the feedback received and makes changes to the prototype until it meets the customer’s requirements.

4. Integration and testing: In the fourth phase, the prototype is integrated into a complete software system. This involves integrating the software components and performing testing to ensure that the system works as expected. Testing includes unit testing, integration

testing, system testing, and acceptance testing.

5. Deployment: In the final phase, the software system is deployed to the customer or end-user. This involves installing the software system and providing training to the end-users on how to use the system.



RAD MODEL

Advantages of RAD model:

- Faster development cycle due to the use of rapid prototyping and iterative development cycles.
- Frequent feedback from the customer or end-user ensures that the software system meets their requirements.
- Reduced development cost and time due to the reuse of existing software components.
- Increased quality of the software system due to the emphasis on testing and integration.

Disadvantages of RAD model:

- The RAD model may not be suitable for large and complex software systems.
- The quality of the software system may be compromised if the development team focuses too much on speed.
- The RAD model requires a high level of collaboration and communication between the development team and the customer or end-user.
- The use of rapid prototyping may lead to a lack of documentation and difficulty in maintaining the software system.