Prototype Model: SDLC (software development life cycle)

Prototype Model: SDLC (software development life cycle)

Prototype Model: SDLC (software development life cycle)

- In this model a small sample is made after taking a requirement from the user.
- And that sample is shown to the user.
- If the customer (user) tells it right.
- After that further process takes place.
- In this, feedback is taken from the user.
- Whereas the waterfall model does not happen like this.
- Therefore the waterfall model is the risk software development model.

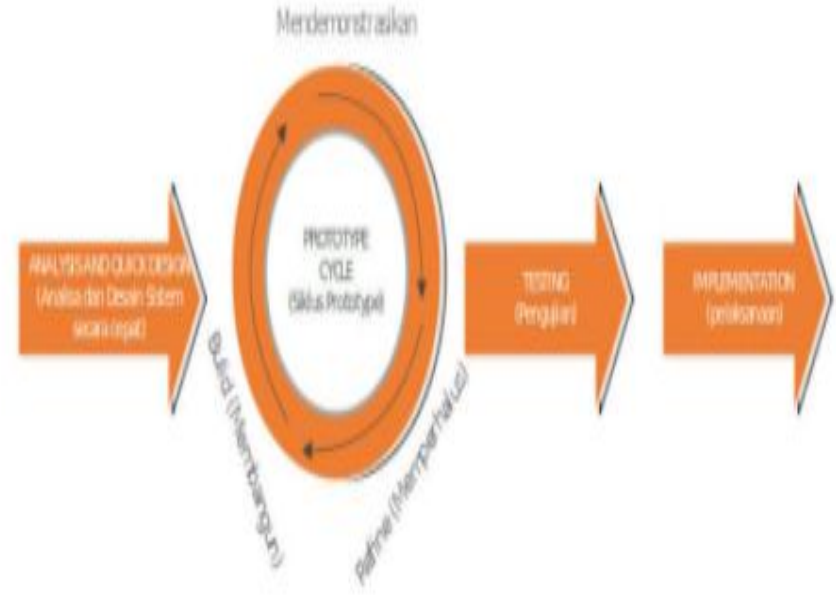# Prototype Model: SDLC (software development life cycle)

WORKING

- Prototyping models are not made with hard planning.
- This is an early approximation.
- In this, the model of the product and the software system is made.
- Which is shown to the customer.

Prototype Model: SDLC (software development life cycle)

Types phase of prototyping model

1.Requirement gathering and Analysis

- The first requirement in this model is gathering and analysis.
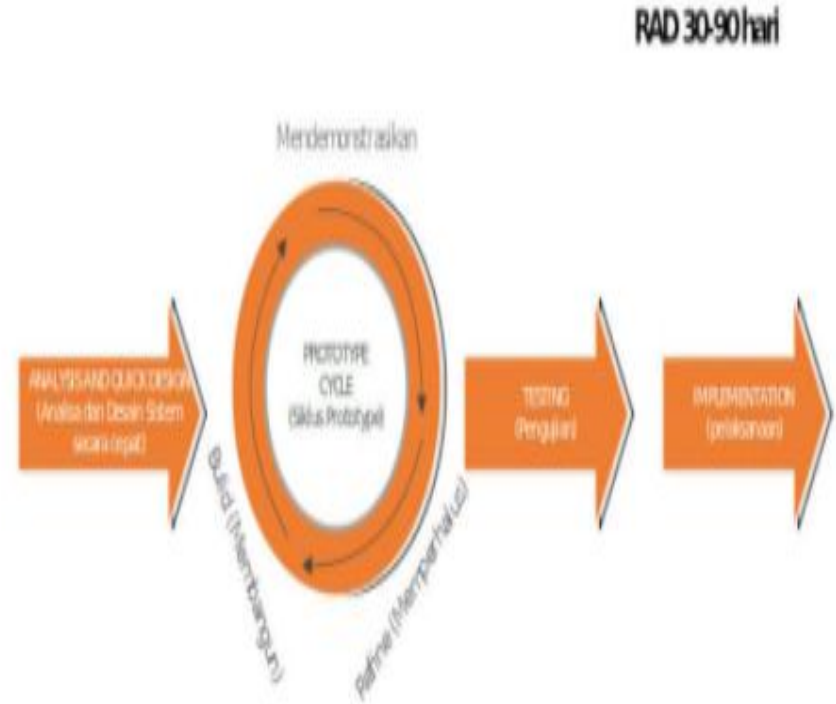- In this, the software engineer and the user sit together and analyze the requirement.

RAD 30-90 hari

Mendemonstrasikan

ANALYSIS AND QUICK DESIGN
(Analisa dan Desain Sistem secara cepat)

PROTOTYPE CYCLE
(Siklus Prototipe)

Build (Membangun)

Refine (Memperbaiki)

TESTING
(Pengujian)

IMPLEMENTATION
(pelaksanaan)

Prototype Model: SDLC (software development life cycle)

Types phase of prototyping model

2. Design

- Quick design is prepared in this model.
- This is not a detailed one.
- There are some important points in this.
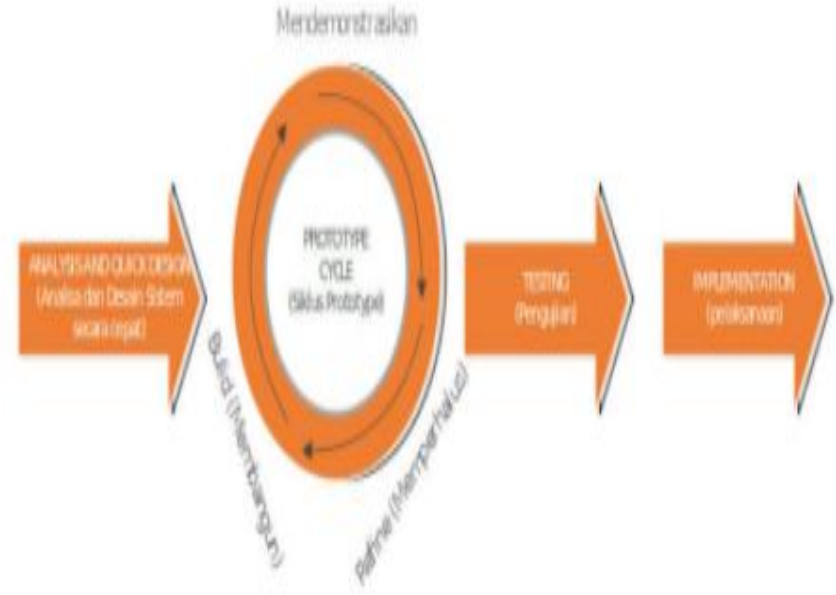- So that the user can know whether the design is correct or not.

RAD 30-90 hari

Mendemonstrasikan

ANALYSIS AND QUICK DESIGN
(Analisa dan Desain Sistem
secara cepat)

PROTOTYPE
CYCLE
(Siklus Prototipe)

TESTING
(Pengujian)

IMPLEMENTATION
(pelaksanan)

Build (Membangun)

Refine (Memperbaiki)

Prototype Model: SDLC (software development life cycle)

Types phase of prototyping model

3. Build prototype

- The information which is taken from quick design.
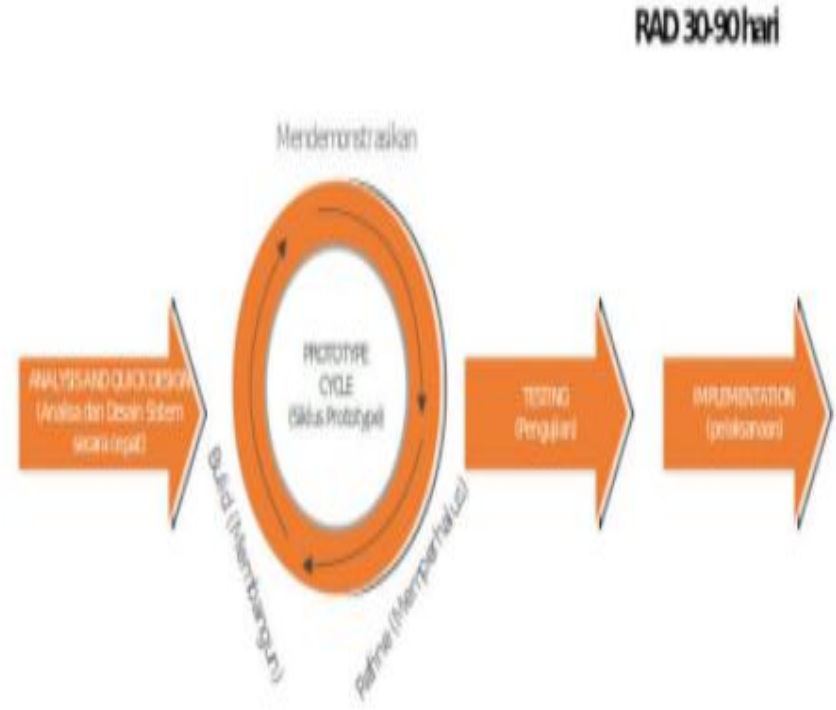- A prototype is prepared from it.

RAD 30-90 hari

Mendemonstrasikan

PROTOTYPE
CYCLE
(Siklus Prototype)

ANALYSIS AND QUICK DESIGN
(Analisa dan Desain Sistem secara cepat)

Build (Membangun)

Refine (Memperhalus)

TESTING
(Pengujian)

IMPLEMENTATION
(pelaksanaan)

Prototype Model: SDLC (software development life cycle)

Types phase of prototyping model

4. Customer Evolutionary

- The system that has been discovered in customer Evolution.
- It has to be given training on how to use it.
- After that we will take evolution from the customer.
- Will try to know whether there is a need for any change in the software that has been created.

RAD 30-90 hari

Mendemonstrasikan

ANALYSIS AND QUICK DESIGN
(Analisa dan Desain Sistem secara cepat)

PROTOTYPE CYCLE
(Siklus Prototipe)

TESTING
(Pengujian)

IMPLEMENTATION
(pelaksanaan)

Build (Membangun)

Refine (Memperbaiki)

Prototype Model: SDLC (software development life cycle)

Types phase of prototyping model

5. Review and update

- If the user says that any changes are required in it, then it will be modified in this phase.

- As soon as the customer will be satisfied.
- After that the next phase will be applied. Like coding, testing, maintenance.
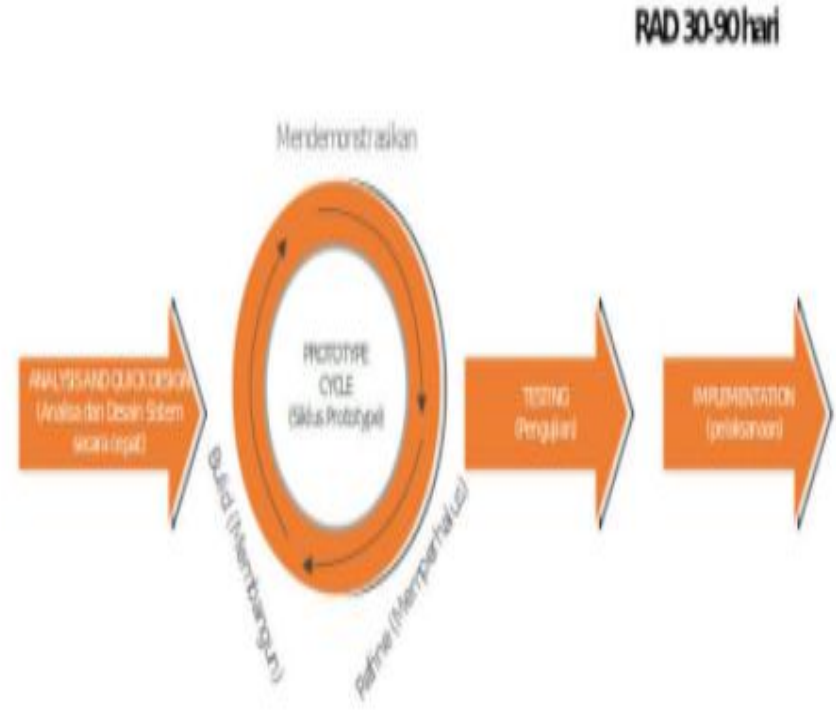
Prototype Model: SDLC (software development life cycle)

SOME POINTS

- Not all features are being added in this phase. But we are providing a basic idea to the user that what type of software you will have.
- We use this model when the accurate requirement and solution are not known. That is, the user does not already know what kind of software he needs.
- The requirements which are understood in this. It is included in the prototype. After understanding the rest of the requirement, it is added. That is, by making a prototype of the requirement that is understood, it is shown to the customer.

Prototype Model: SDLC (software development life cycle)

SOME POINTS

- Whereas in the waterfall model, all the requirements are known first. Only then will the software development take place while it is not possible that the customer should tell all the requirements together, in that case the prototype model is used.
- In this, multiple iteration and requirement analysis and design are repeated equally.
- When the user is satisfied, then the code of the prototype is used to make the final software.
- This model is not based on strict planning.

Prototype Model: SDLC (software development life cycle)

<u>Advantage of prototyping model</u>

- It is easy to identify the requirements of the customer.

- In the waterfall model, the customer's feedback is taken in the last. But the prototype here is shown to the customer after it is designed at the beginning itself.

- Here the customer understands from the very beginning how the software will be made.

- In this the user is involved. In the software development process, whereas in the waterfall model, the customer does not get the user until the software is ready.

Prototype Model: SDLC (software development life cycle)

Advantage of prototyping model

- Here the error is solved by finding it. Whereas this does not happen in the waterfall model.

- Here quick feedback is received from the customer, due to which the better solution is found and addressed.

- If anything is missed in the prototype, then it can be easily found. Because the feedback keeps on getting from the customer.

- It will take less time to make this project. Because once we have done the requirement analysis from the customer, then it will take less time to develop the project.

Prototype Model: SDLC (software development life cycle)

Disadvantage of prototyping model

- It may contain insufficient analysis. Because the software engineer will not analyze the requirement after making the prototype.

- The documentation here may not be made complete.