

# **SDLC**

## **(Systems Development Life Cycle)**

**Nasrul, S.Pd.I, S.Kom, M.Kom**  
**[nasrul@nurulfikri.com](mailto:nasrul@nurulfikri.com)**

# SDLC ?

## **Systems Development Life Cycle**

**(Daur hidup pengembangan sistem) :**

Kerangka kerja yang menggambarkan proses aktifitas setiap tahapan pengembangan software

### **Fase-Fase pengembangan**

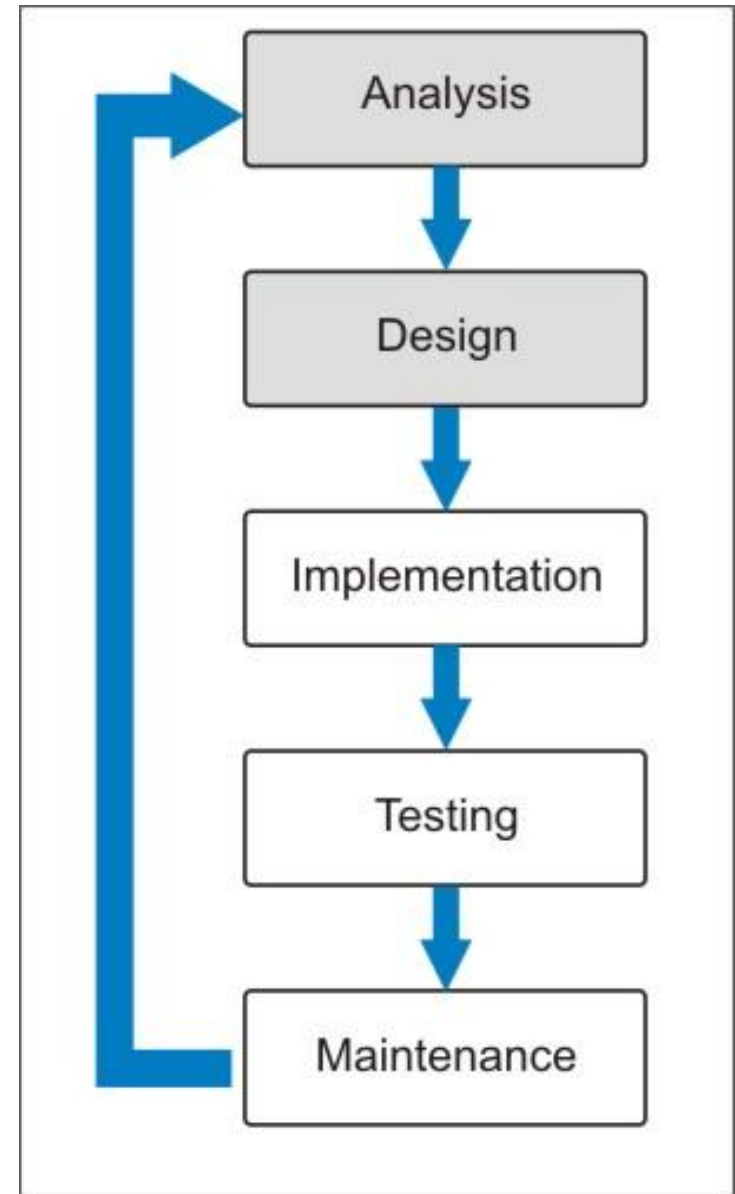
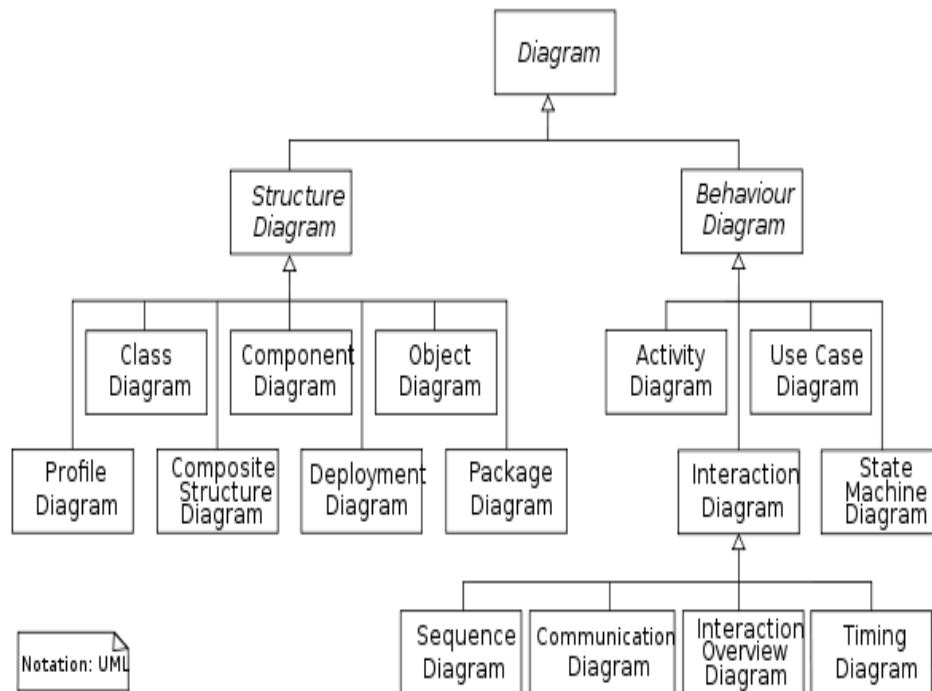
Planning	Definition	Analysis	Design	Build	Transition	Warehouse
----------	------------	----------	--------	-------	------------	-----------

# Who use SDLC ?

## Systems Development Life Cycle

Proses-proses logik yang digunakan oleh **system analyst** untuk mengembangkan **sistem informasi** yang didalamnya termasuk requirement, validasi, training, dan user ownership

# Diagram -Tahapan SDLC



# SDLC :: Analysis

- Tujuan: memahami permasalahan secara menyeluruh dan detail, melalui requirement yang jelas
- Fokus : sistem dengan logika tampilan -> apa yang akan dihasilkan oleh sistem
- Input: berbagai studi kelayakan
- Output: Spesifikasi Requirement

## SDLC :: Analysis activities

- Mengumpulkan informasi tentang semua sistem yang digunakan saat ini dan mendapatkan kebutuhan dari sistem yang akan dibangun
- Melakukan pemodalan sistem yang ada (optional)
- Melakukan pemodelan sistem yang akan diajukan untuk dikembangkan menjadi sistem yang komplit sesuai definisi kemampuan sistem
- Membuat solusi-solusi alternatif

# SDLC :: Analysis Model

- Structured analysis: Data Flow Diagrams, data dictionary, entity-relationship diagrams, process specification
- Data analysis: purely data oriented, entity-relationship diagrams dan teknik pemodelan data lainnya
- Object-oriented analysis: use-case diagrams, class diagrams

## SDLC :: Design

- Tujuan: menghasilkan desain untuk sebuah sistem yang sesuai dengan identifikasi requirement
- Fokus : Tampilan fisik, bagaimana sistem sesuai dengan requirement
- Input: Spesifikasi Requirement
- Output: Dokumen spesifikasi desain



# SDLC :: Design activities

- Hal-hal yang dilakukan pada fase desain:
  - desain forms and reports
  - desain interfaces and dialogues
  - desain logical database
  - desain physical files dan databases
  - desain system and struktur program : tools: structure charts, pseudo-code
  - desain distributed processing systems

# SDLC :: Implementation & Testing

- Tujuan : Sebuah sistem yang bekerja yang terinstall dan digunakan
- Input: Dokumen spesifikasi desain
- Output: Dapat dilakukan testing pada sistem yang terinstall

# SDLC :: Implementation & Testing activities - 1

- Create test data
- Kode program
- Test program
  - unit testing (each module)
    - system testing (build up larger and larger portions of the system)
    - acceptance testing (very formal, keyed to user of the system)
- Dokumentasi program

# SDLC :: Implementation & Testing activities - 2

- Training users
- Persiapan rencana instalasi
- Convert data files ke sistem baru
- Install new hardware and software platforms
- Install the information system
- Perform final testing (UAT : User acceptance test)

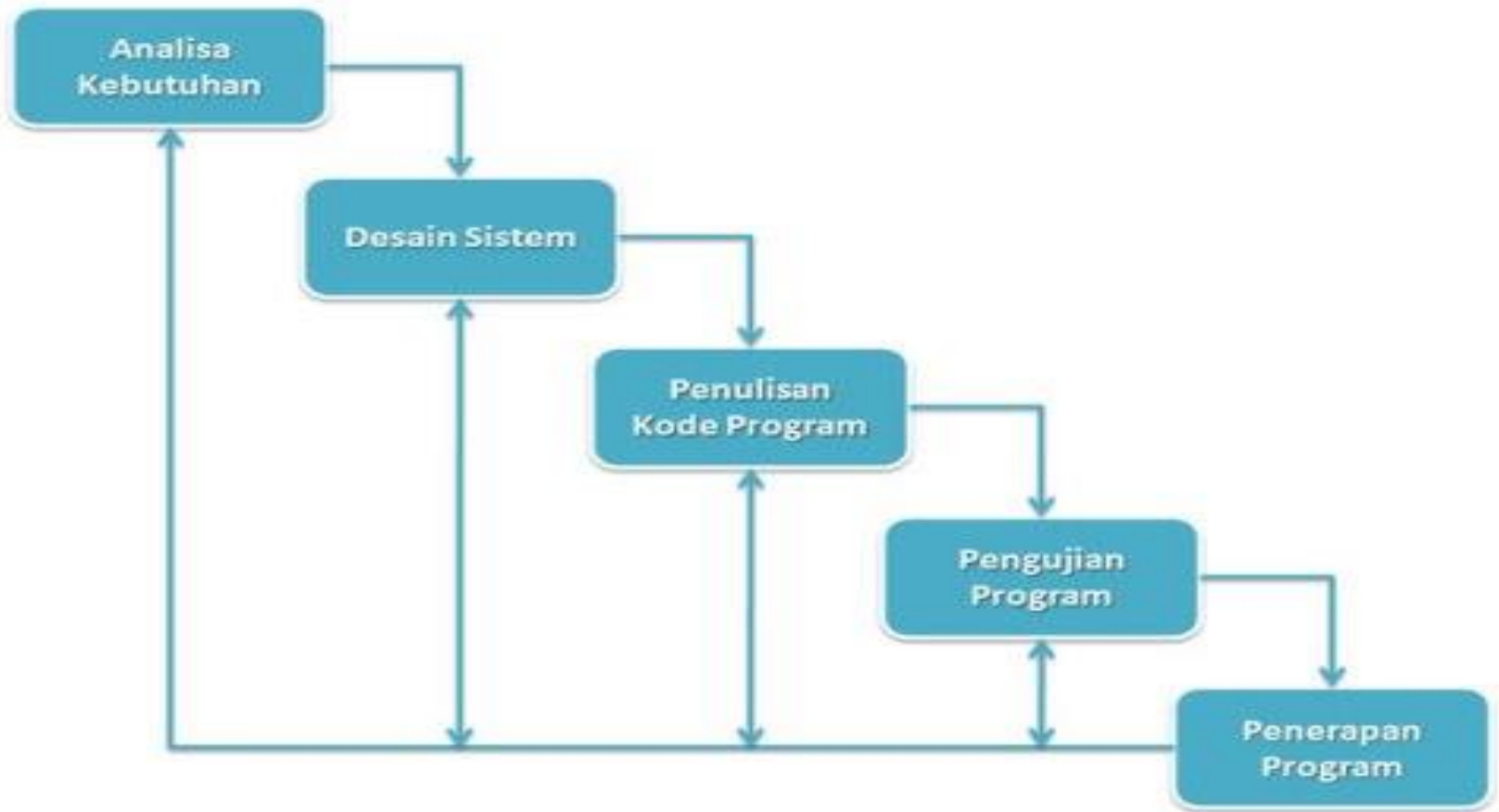
# SDLC :: Maintenance

- Secara sistimatis memperbaiki dan memperluas kinerja system
  - Penambahan fitur
  - Membetulkan errors
  - Penyesuaian external sistem (Hardware , sistem operasi )

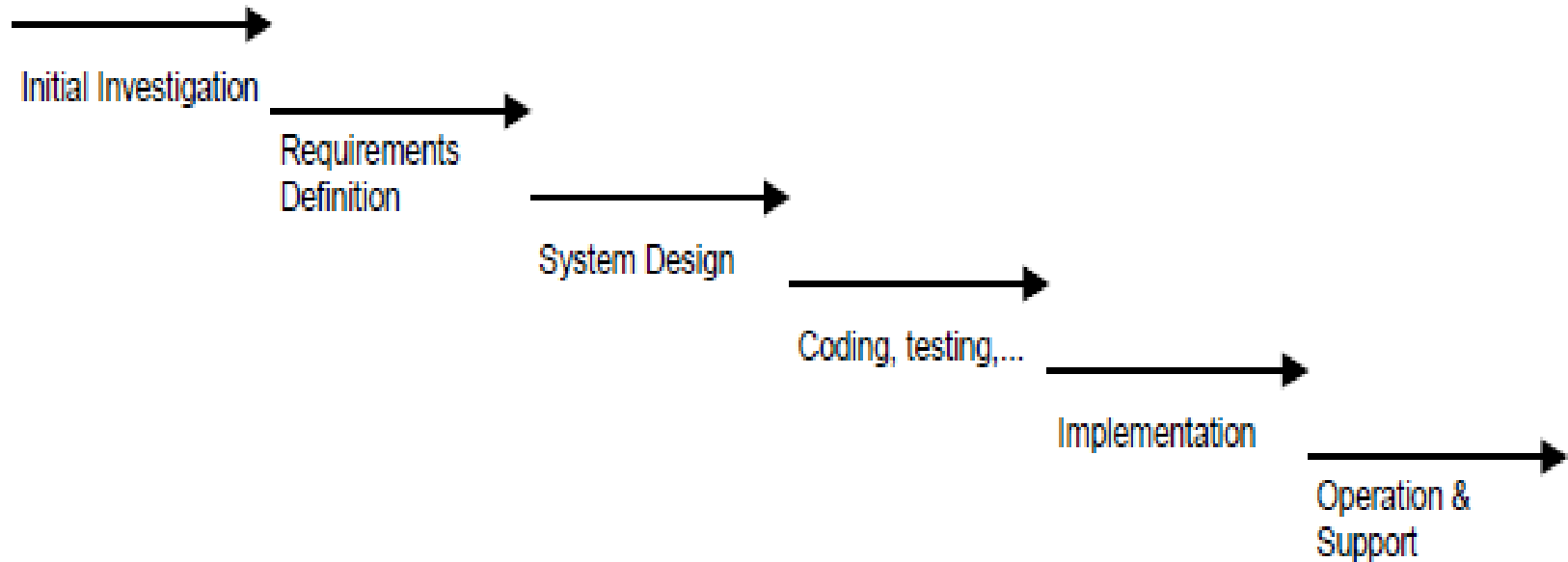
# Metodelogi SDLC

- 1) Waterfall
- 2) Incremental
- 3) Prototype
- 4) RAD
- 5) Agile Development
- 6) Etc

# Waterfall Model



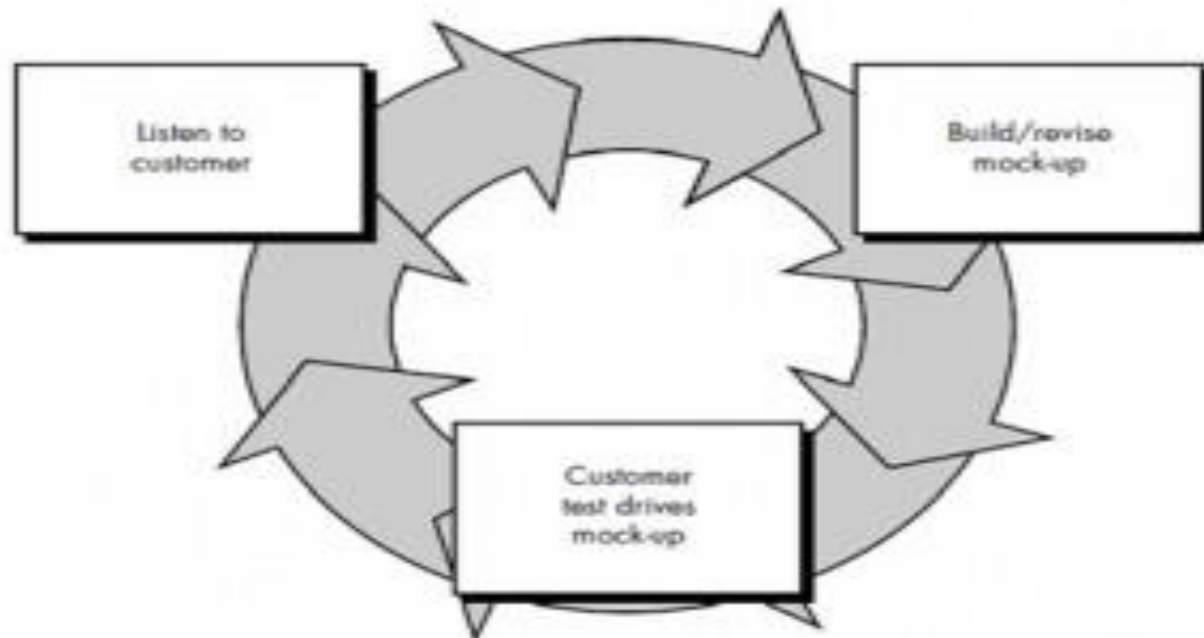
# Waterfall : Linear



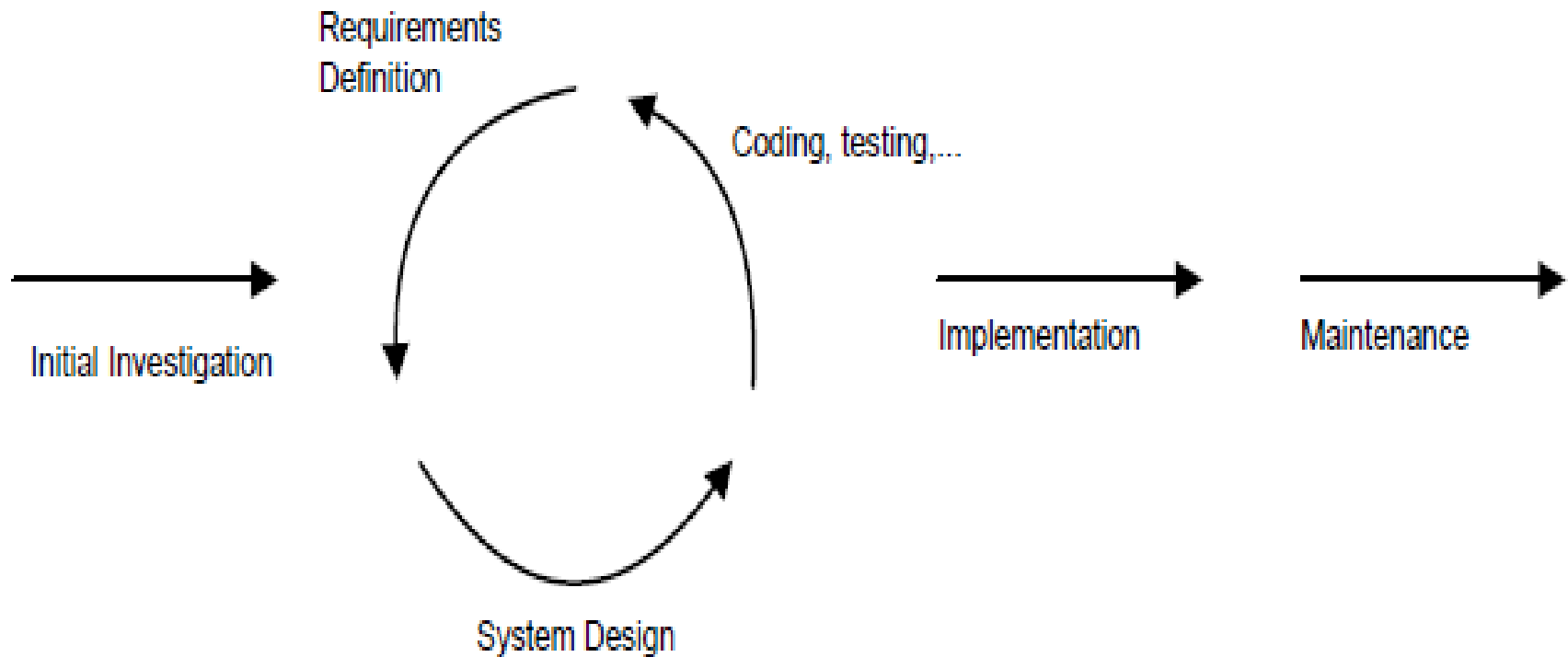


# Prototype

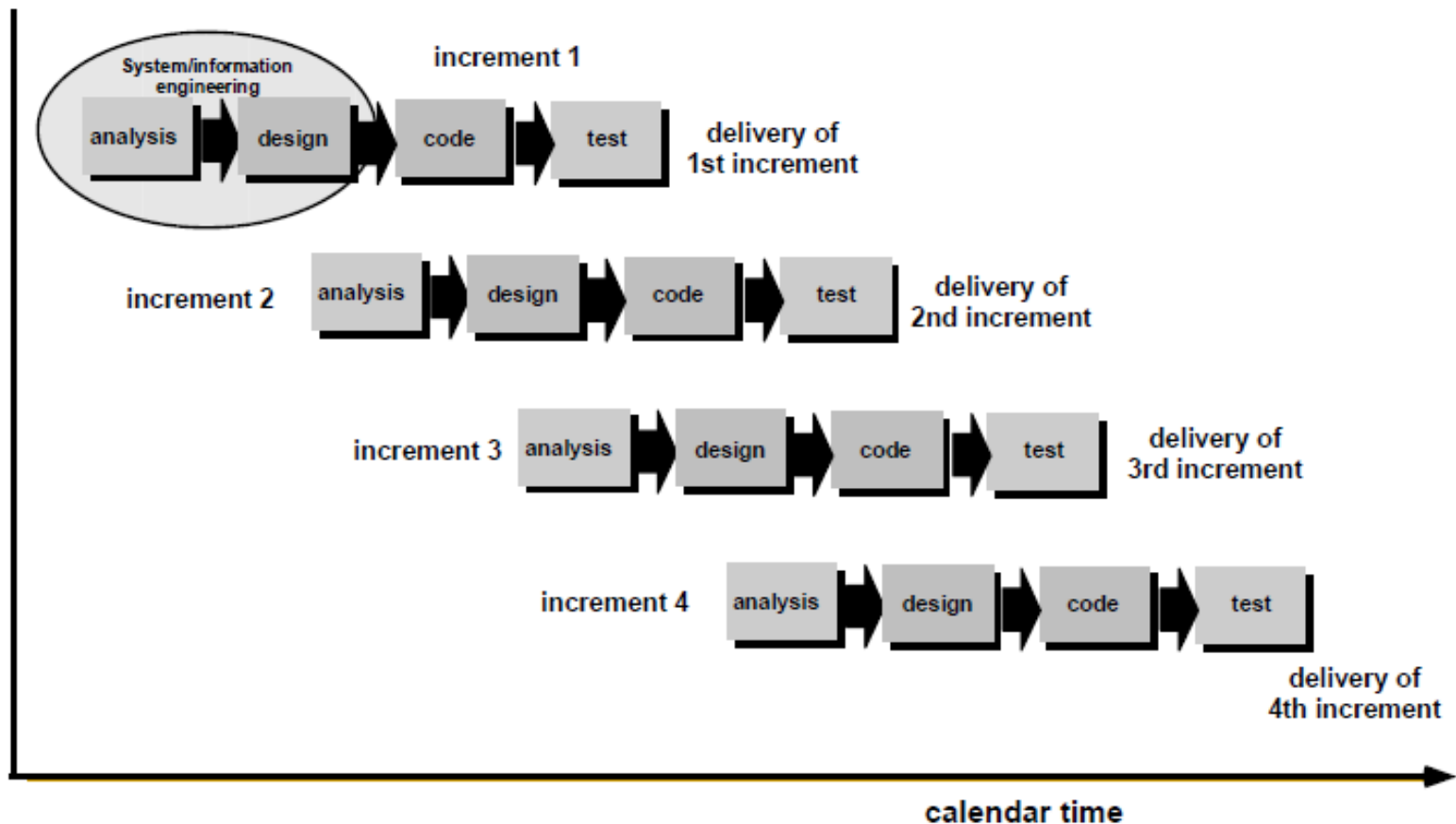
- Membuat sebuah contoh prototipe untuk menunjukkan kebutuhan dan desain pemakai



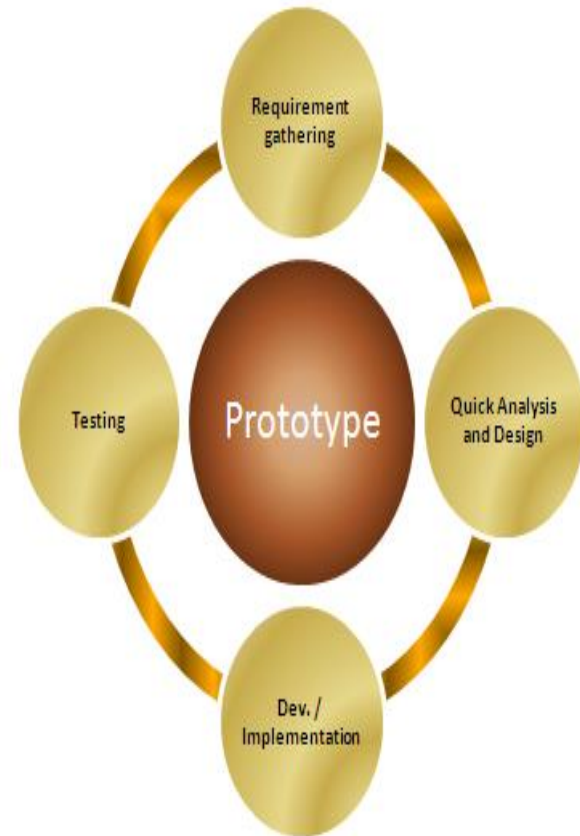
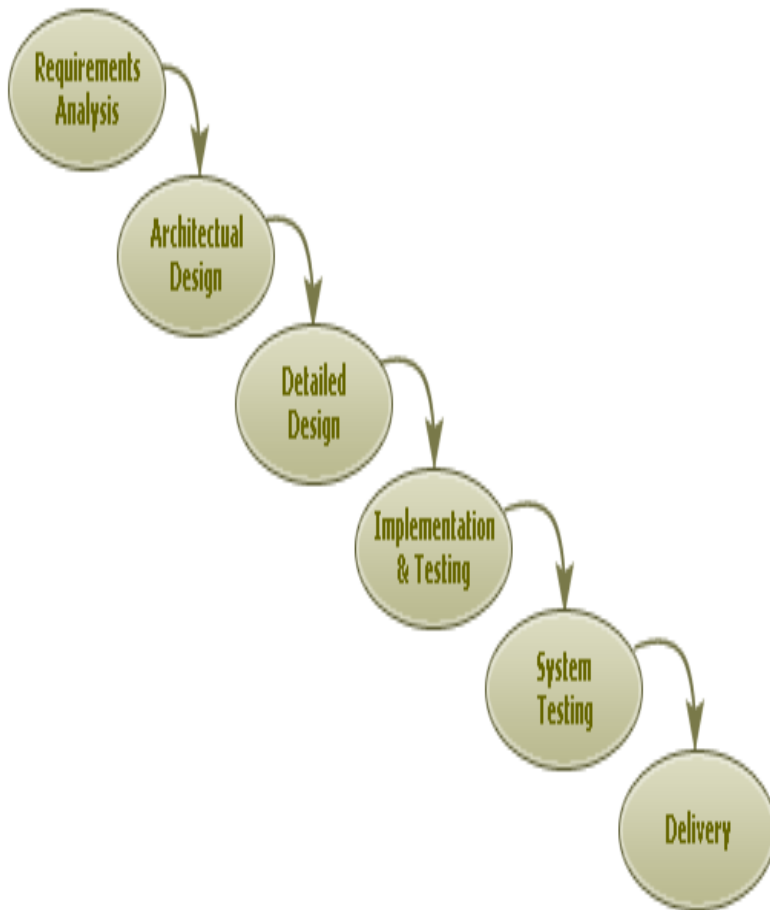
# Prototype :: iterative



# Incremental : Kombinasi linear dan iterative



# RAD : iterative

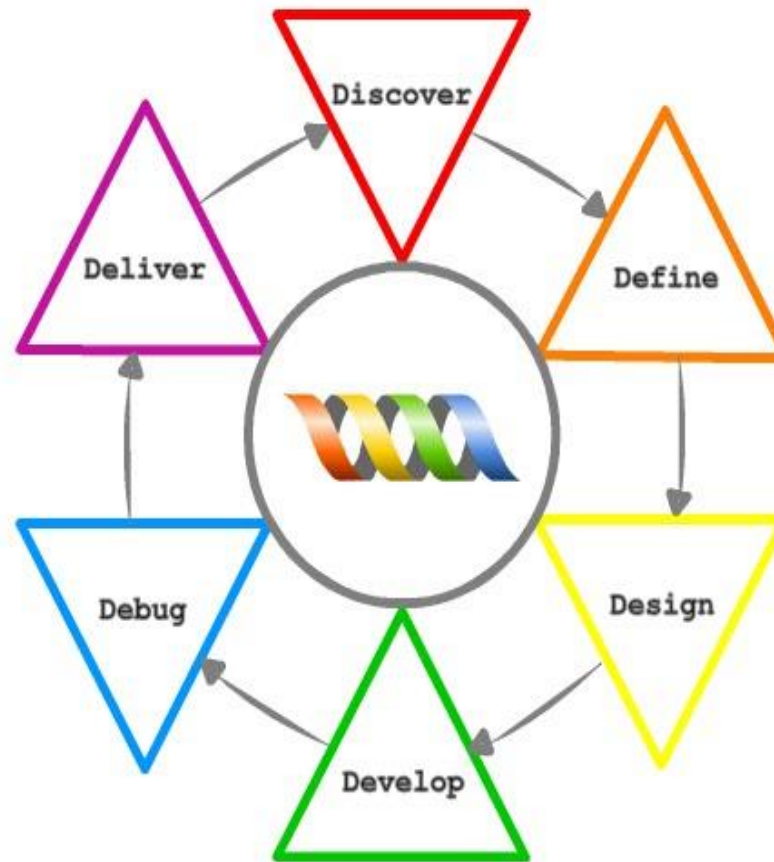


# Agile Software Development

Kumpulan metode pengembangan perangkat lunak berdasarkan iteratif dan incremental.

Agile software development bercirikan pengembangan software dengan melibatkan secara aktif owner project / customer dengan tim pengembang ( kolaborasi )

# Agile-Development



ADAPT  
(Agile DNA-Activities Process Template)

# Agile Process: Discover

- Actor
  - Stakeholders / Customer
  - Project Manager
- Outputs
  - Executive Summary
  - Abstract
  - RFP (Request For Proposal)
  - Project Description

# Agile Process : Define

- Actor
  - Stakeholders / Customer
  - Project Manager
  - Business Analyst
- Outputs
  - Requirement
  - UML Diagram
  - Manual
  - Mockup



# Agile Process : Design

- Actor
  - Architect / Tech Lead
  - Business Analyst
- Outputs
  - Database Diagram
  - API / Code Skeleton
  - Solution / Project
  - Architecture
  - Unit-test

# Agile Process : Develop

- Actor
  - Developer
  - Designer
- Outputs
  - Business Logic / intelligence
  - UI / Grapics
  - Validation
  - Navigation
  - Security

# Agile Process : Debug

- Actor
  - Stakeholder / Customer
  - QA Testers
  - Business Analyst
  - Developer & Designer
- Outputs
  - Bug list
  - Resolved issue list

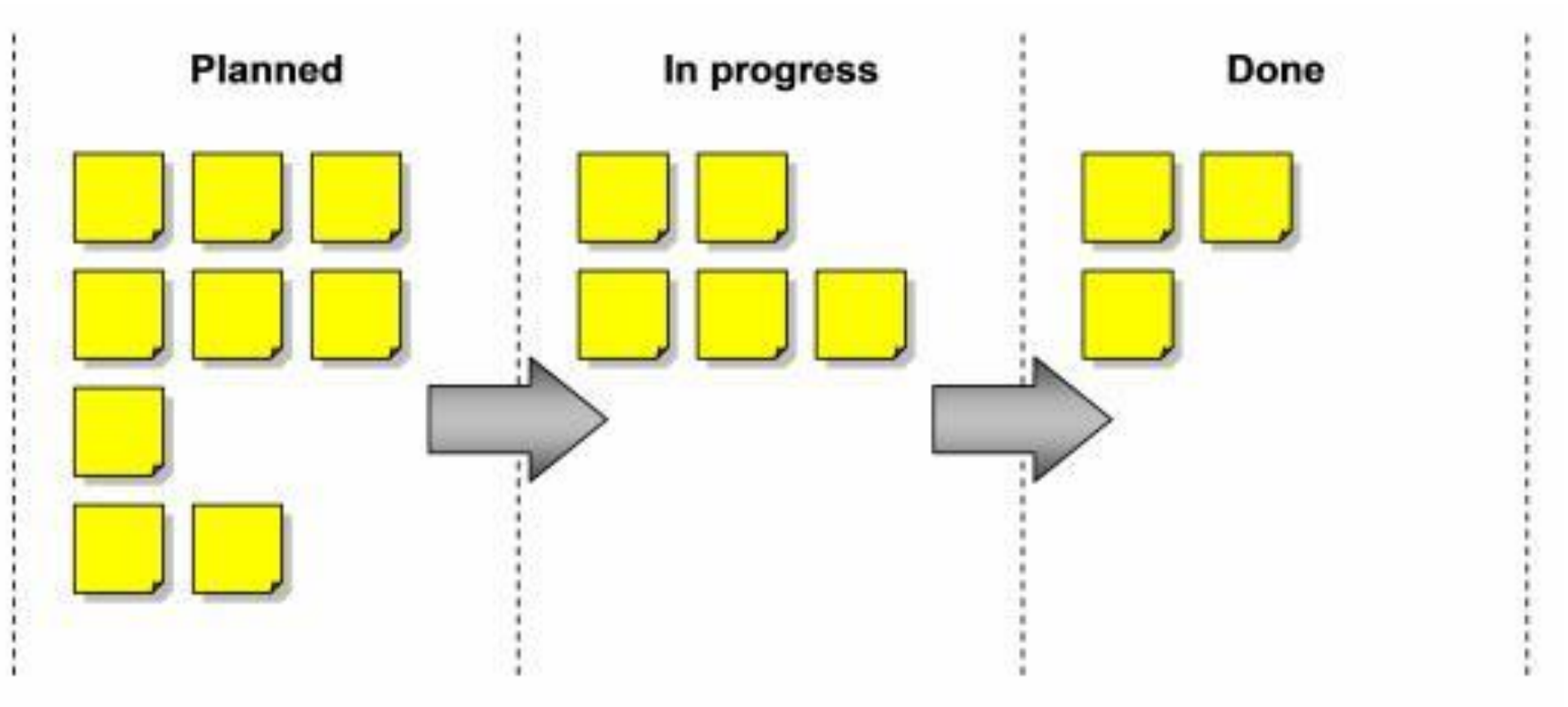
# Agile Process : Deliver

- Actor
  - Build Manager
  - Tech Leads
  - Developer
- Output
  - Deployed Application

# Agile Process Model

- 1) **XP** - Extreme Programming
- 2) **ASD** – Adaptive Software Development
- 3) Dynamic Systems Development method
- 4) **SCRUM**
- 5) Agile Modelling

# POST IT Agile Development



# POST IT Agile Development



# POST IT Agile Development

Agilar Taskboard

http://taskboard.agilar.org/taskboard/team/6

Smart Bookmarks - Google Translate - Gmail - Inbox - Google Calendar - Google Reader (100+) - Wikipedia - Plesk 8.1.0 - BIT-HDTV - Home o... - site.dev.agilar.org - Taskboard DEV - iLean DEV

Wikipedia, the free encyclop... - Scrum Alliance - Certified Sc... - Agilar Taskboard - Dashboard [Hudson] - RFQ detail - Gmail - Inbox - xquesada@g...

Product Backlog - Add Task

PLEASE ANALYSE - PLEASE TEST - DONE - BLOCKED - HIGH PRIORITY - WAITING - DELEGATED - BUG

FRANCISCO - XAVIER - MARCOS

Agilar Dev Team

AA018  
Legacy Issues And Unplanned Items  
Priority: 3000  
Size: 0

AP005  
Training section and module - dirt road  
Priority: 2885  
Size: 5

NOT STARTED	IN PROGRESS	FINISHED
<p>Fix Bubbles</p> <p>Deploy to staging should backup the production database and test a production deploy</p>	<p>HUC PLEASE TEST HUC deploys know about the status of the last good build FRANCISCO</p> <p>BUG Sponsorship listings. MARCOS N</p>	<p>When clicking everything goes to the homepage</p> <p>Check alignment of forms</p>
<p>Add tests for active resource (see ActiveResource::HttpMock)</p> <p>Add tests to agora training controller (upcoming and past)</p> <p>Add AJAX calendar to start/end date</p>	<p>ADD variations on start / end date MARCOS N</p> <p>PLEASE TEST Camera URL de produccion: #course_name MARCOS N</p> <p>PLEASE TEST Mail to trainer and organizer when new registration occurs MARCOS N</p>	<p>Change Trainings heading to "Public Trainings"</p> <p>Add tobias training</p>
<p>"You trainings" sh BUG co-trainer but not main trainer</p> <p>You can stop public BUG your profile site even if you have published trainings</p> <p>Possible input sam BUG and co-trainer</p>	<p>Only DONE ants who their profile on the Agilar website should be eligible for trainings</p> <p>BUG DONE de traine training (website) - link apunta a...</p> <p>PLEASE TEST Support for external registration mechanism (URL field) MARCOS N</p>	
<p>The owner of a training should be able to edit and delete it</p>	<p>PLEASE TEST Cannot Edit existing trainings MARCOS N</p>	

Done