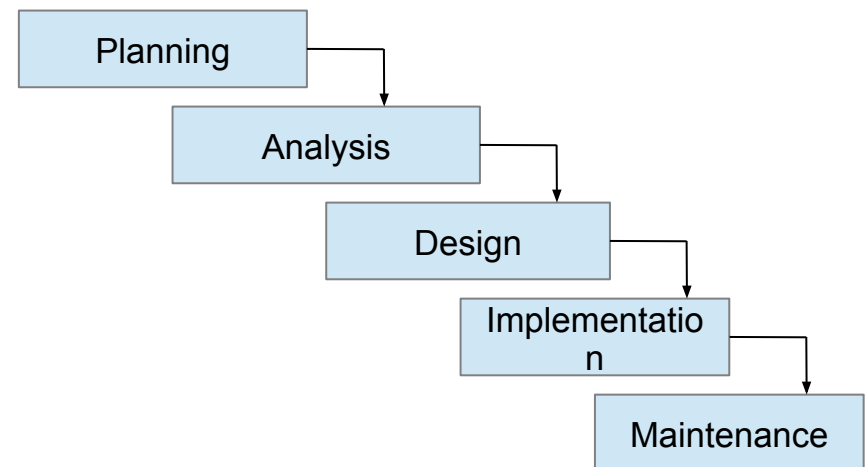


# Metodologi Pengembangan Sistem Informasi

# Metode Waterfall

- Merupakan pendekatan tradisional
- One big project
- Fase yang lain dimulai setelah fase sebelumnya selesai (sequential process)
- Tanpa backtracking dan looping (segala requirement harus dinyatakan dengan jelas).
- Cocok untuk situasi dimana perubahan adalah tidak dimungkinkan.
- Internal



# Permasalahan pada Pendekatan Waterfall

- System requirement “dikunci” setelah ditentukan (tidak dapat diubah)
- Keterlibatan user terbatas (hanya pada fase requirement)
- Terlalu fokus pada deadline jadwal dari setiap fase SLDC sehingga mengorbankan praktek pengembangan yang sehat.

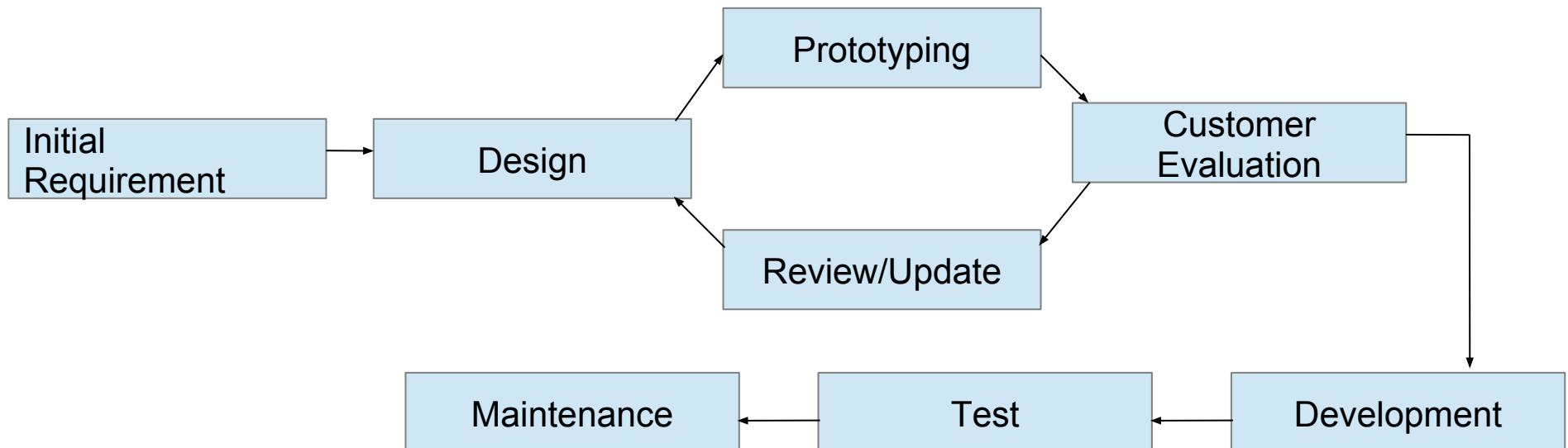
# Alternatif terhadap Tradisional Waterfall

- Prototyping
- Rapid Application Development (RAD)
  - Case tools
  - Joint Application Design (JAD)
- Agile Methodologies
  - extreme Programming

# Prototyping

Merupakan proses pengembangan bersifat iteratif:

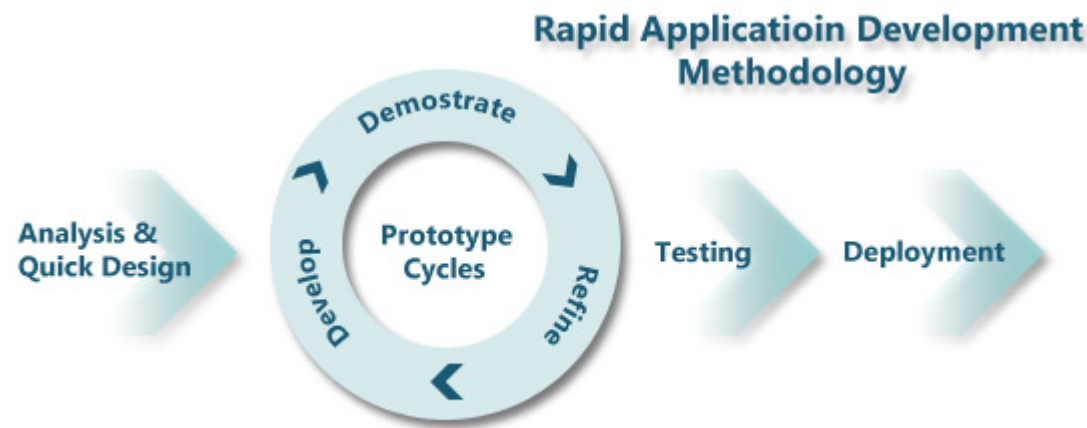
- Requirement dengan cepat dikonversi menjadi suatu sistem yang bekerja.
- Sistem direvisi secara terus-menerus
- Kolaborasi erat antara user dan analyst.



Prototyping sering digunakan pada fase analisa pada SDLC tradisional untuk mempercepat development untuk menghasilkan model prototype yang dapat digunakan tanpa analisa dan design yang detail. Prototype dapat menggunakan Spreadsheet, maupun MS Access.

# Rapid Application Development

- Metodologi untuk mengurangi waktu perancangan dan implementasi.
- Termasuk: prototyping, JAD, CASE tools dan code generator

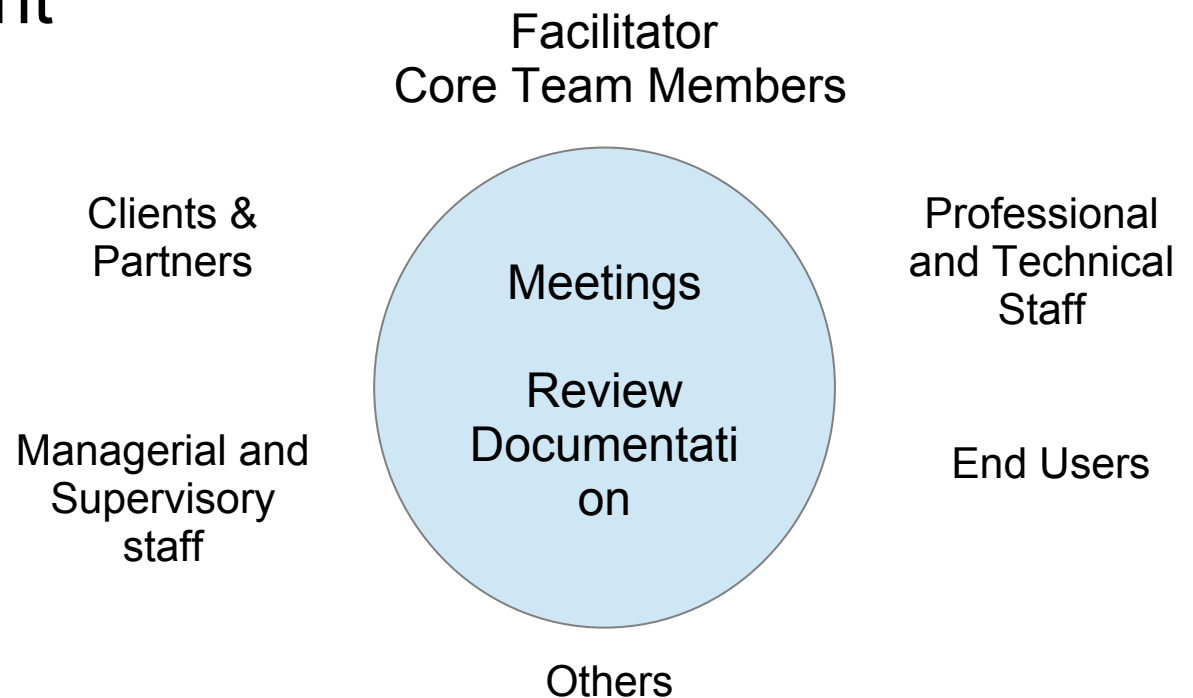


# Case Tools

- Computer Aided Software Engineering
- Software tools yang menyediakan dukungan otomatisasi untuk pengembangan sistem.
- Project dictionary/workbook-system description and specification
- Diagramming tools
- Contoh produk: Oracle designer, Rational Rose

# Joint Application Development

- Proses terstruktur yang melibatkan user, analyst, dan manager
- Sesi pertemuan workgroup intensif beberapa hari
- Tujuan – untuk menspesifikasikan atau me-review system requirement

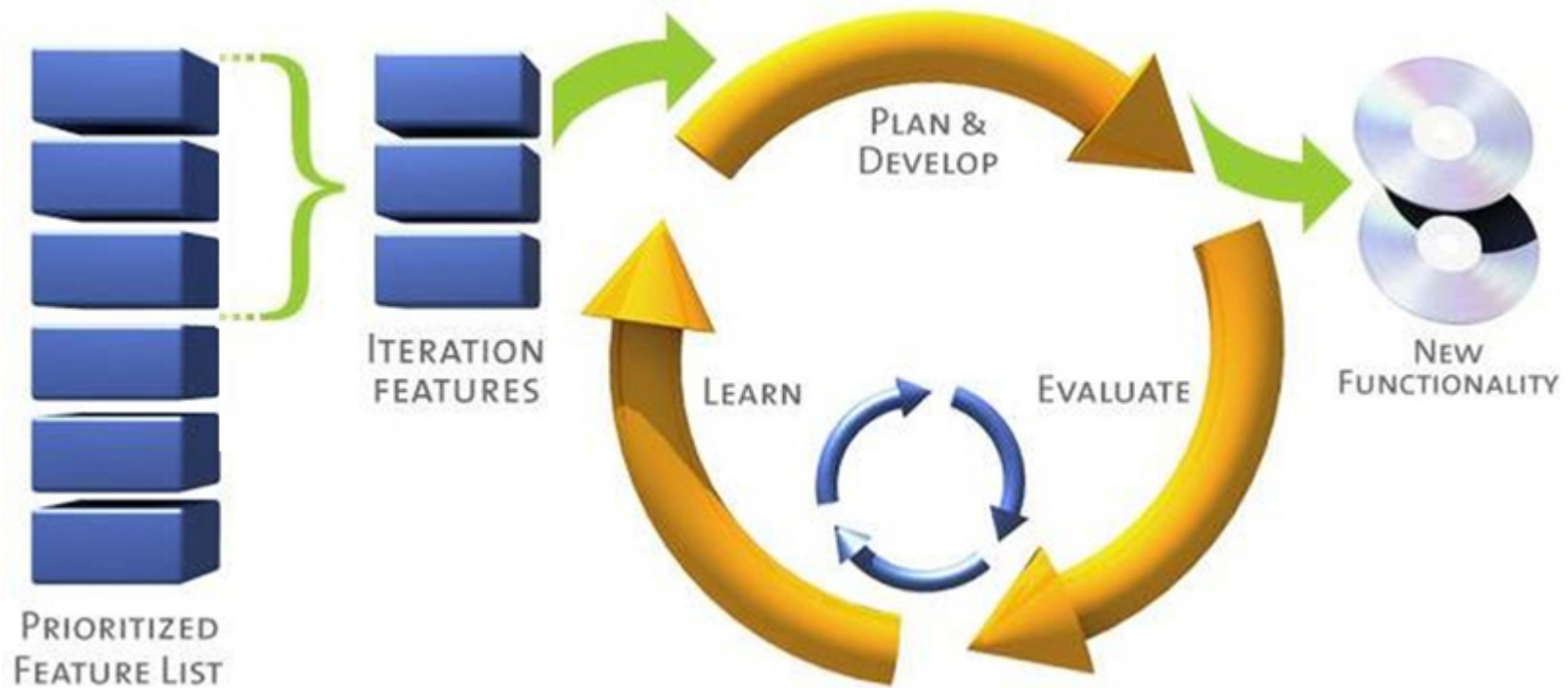




# Metodologi Agile

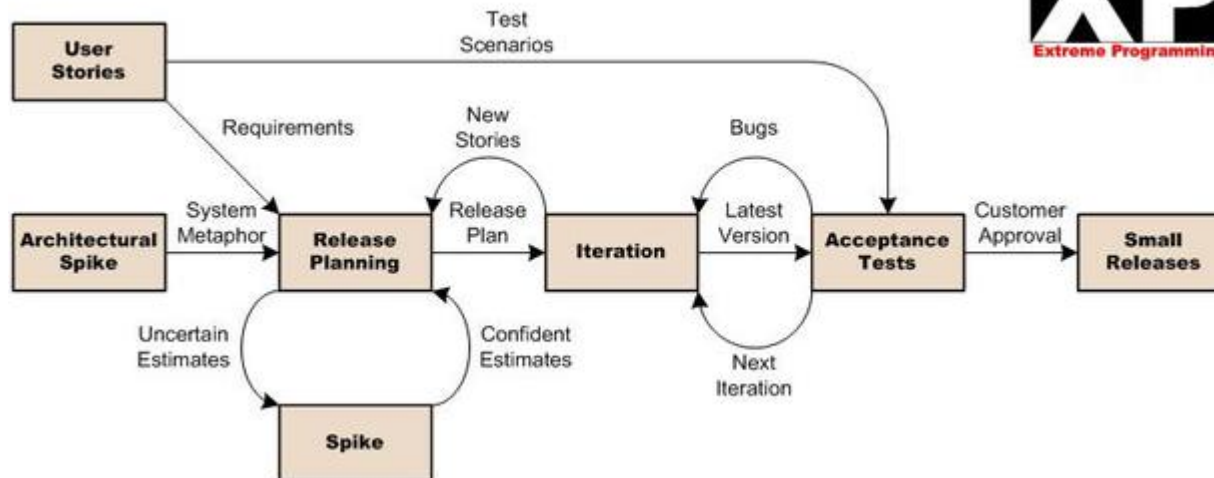
- Dimotivasi dengan menyadari bahwa pengembangan software seperti fluida, sulit diprediksi, dan dinamis.
- Fleksible
- Many small project, dan highly collaborative
- Cocok untuk yang menginginkan continuous improvement.
- Melibatkan user
- Sebuah proses dimana requirement dapat berevolusi dan berubah.
- Tiga prinsip utama:
  - Bersifat adaptif daripada prediktif
  - Menekankan kepada orang daripada aturan
  - Proses bersifat self-adaptive

# Metodologi Agile (lanjutan)



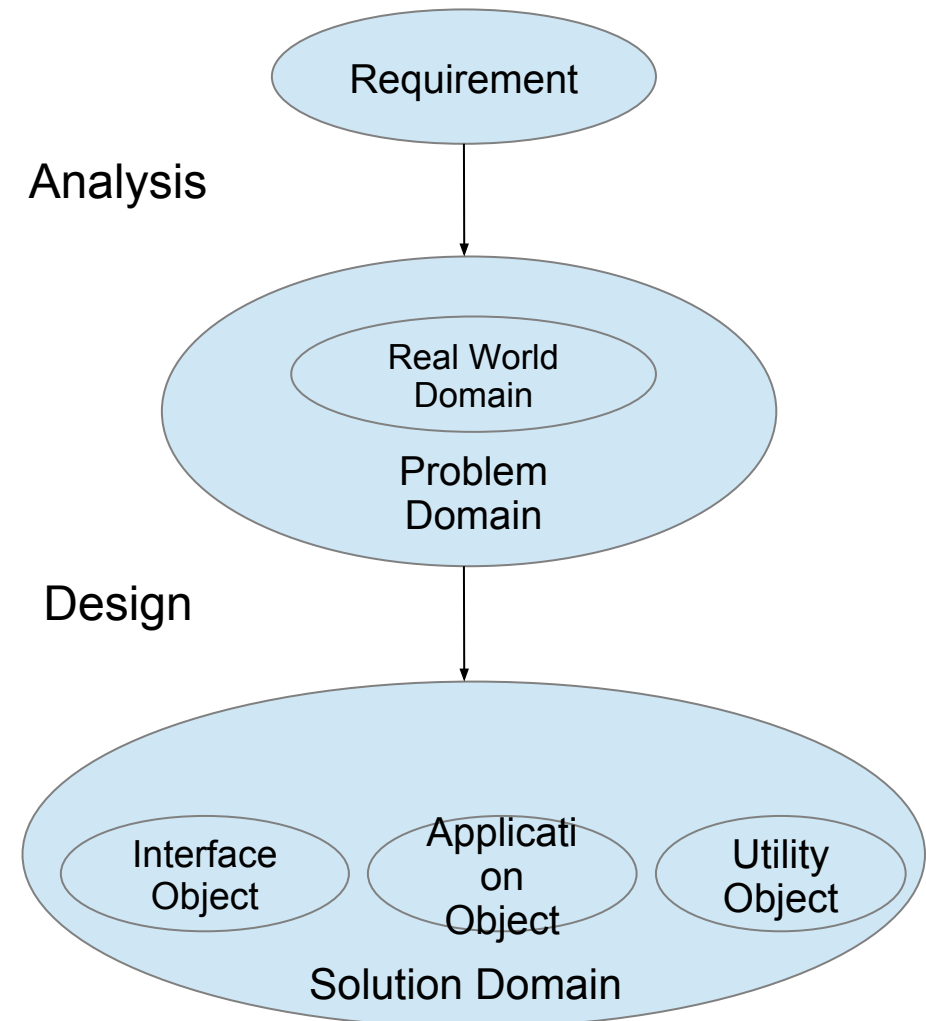
# eXtreme Programming

- Pendek, siklus pengembangan bersifat incremental.
- Otomatisasi test
- Tim pengembangan Two-person (Pair Programming)
- Koding dan Testing dioperasikan secara bersamaan.
- Keuntungan:
  - Komunikasi antar developer
  - Produktivitas tingkat tinggi
  - Koding berkualitas tinggi



# Object Oriented Analysis and Design

- Menekankan pada objek dari pada data atau proses.
- Objek – suatu struktur enkapsulasi atribut-atribut dan perilaku dari suatu entitas dunia nyata.
- Kelas Objek – suatu pengelompokan logika dari objek-objek yang berbagi atribut-atribut dan perilaku-prilaku yang sama.
- Inheritance – pengaturan hirarki dari kelas-kelas yang memperbolehkan sub-sub kelas untuk mewarisi properti dan metode dari superclass.



# Pemilihan Metodologi

Pilih	Ketika
Metodologi Waterfall	<ul style="list-style-type: none"><li>• System telah dikembangkan dan didokumentasi dengan SDLC</li><li>• Penting untuk dokumentasi pada setiap langkah</li><li>• Manajemen merasa lebih cocok dan aman menggunakan SDLC</li><li>• Sumber daya dan waktu tersedia untuk menyelesaikan semua tahapan SDLC</li><li>• Komunikasi mengenai bagaimana sistem baru bekerja adalah penting</li></ul>
Metodologi Agile	<ul style="list-style-type: none"><li>• Aplikasi perlu dikembangkan dengan cepat untuk respon terhadap lingkungan yang dinamis</li><li>• Suatu tindakan penyelamatan (sistem mengalami kegagalan dan tidak ada waktu untuk mencari apa yang terjadi)</li><li>• Pelanggan puas dengan pengembangan incremental</li><li>• Eksekutif dan analyst setuju dengan prinsip methodology Agile.</li></ul>
Metodologi OOP	<ul style="list-style-type: none"><li>• Permasalahan telah dimodelkan sebagai class</li><li>• Organisasi mendukung pembelajaran UML</li><li>• System dapat ditambahkan secara berkala, satu sistem setiap waktu</li><li>• Pemakaian kembali software yang ditulis sebelumnya dimungkinkan</li><li>• Adalah dapat diterima untuk menyelesaikan masalah yang sulit terlebih dahulu</li></ul>

# Apa yang sama

Walaupun masing-masing metodologi kelihatannya berbeda, tetapi persamaannya adalah:

1. Analyst perlu memahami cara kerja organisasi terlebih dahulu.
2. Team project perlu membuat anggaran dan jadwal dan membuat proposal project.
3. Perlu dilakukan proses interview dan pengumpulan data dengan questionnaire, dan contoh2 dokumen dan laporan untuk mengetahui bagaimana bagaimana transaksi bisnis berjalan.

# End User SDLC

Pengetahuan user terkait Excel dan MS Access dapat menyebabkan user melakukan pengembangan tools mandiri yang dapat menyebabkan permasalahan:

1. Dokumentasi yang kurang
2. Masing-masing user mengembangkan sistem sesuai dengan keinginan masing2.
3. Menggunakan tools yang tidak kompatibel dengan sistem perusahaan.
4. Dukungan terputus jika yang bersangkutan tidak lagi pada posisi pekerjaant ersebut.
5. Sulit diupgrade.
6. Sulit di maitenance.

# Quiz

Pendekatan \_\_\_\_ pada pengembangan sistem sering disebut sebagai model “waterfall”.

- a. SDLC
- b. Prototyping
- c. RAD
- x. XP



# Quiz

Model Waterfall pada software development adalah:

- a. Suatu pendekatan yang masuk akal kalau requirement terdefinisi dengan baik.
- b. Suatu pendekatan yang baik ketika suatu program yang bekerja diperlukan dengan cepat.
- c. Pendekatan terbaik untuk project dengan tim pengembangan besar.
- d. Model ketinggalan jaman dan jarang digunakan lagi.

# Quiz

Model Prototyping pada pengembangan software adalah:

- a. Suatu pendekatan yang masuk akal kalau requirement terdefinisi dengan baik.
- b. Suatu pendekatan yang baik ketika customer tidak dapat mendefinisikan requirements dengan jelas.
- c. Pendekatan terbaik untuk project dengan tim pengembangan besar.
- d. Model ketinggalan jaman dan jarang digunakan lagi.

# Quiz

Model Incremental pada software development adalah:

- a. Suatu pendekatan yang masuk akal kalau requirement terdefinisi dengan baik.
- b. Suatu pendekatan yang baik ketika suatu core produk diperlukan dengan cepat.
- c. Pendekatan terbaik untuk project dengan tim pengembangan besar.
- d. Suatu model revolusional yang tidak digunakan untuk produk komersial.

# Quiz

Model spiral pada software development:

- a. Berakhir dengan menghasilkan produk software.
- b. Lebih chaos dibandingkan dengan model incremental.
- c. Melibatkan evaluasi resiko project pada setiap iterasi.
- d. Semua yang disebutkan diatas.

# Quiz

Manakah teknik berikut ini dimana tim pemakai, dan spesialis IS terlibat didalam sesi intensif untuk meminimalkan total waktu untuk mengumpulkan informasi dari banyak peserta.

- a. RAD
- b. JAD
- c. CASE
- d. Piloting

# Quiz

Software tool yang digunakan untuk otomatisasi dari sejumlah langkah didalam metodologi pengembangan software adalah:

- a. RAD
- b. Piloting
- c. JAD
- d. CASE

# Quiz

Berikut ini adalah keuntungan pemakaian metodologi RAD, kecuali:

- a. Mengurangi waktu pengembangan sistem
- b. Kualitas sistem jauh lebih baik dari SDLC
- c. Fokus pada kebutuhan sistem yang essential
- d. Kemampuan cepat berubah terkait dengan perubahan design oleh user.

# Quiz

Berikut ini yang bukan merupakan karakteristik dari XP:

- a. Merupakan salah satu jenis dari pendekatan agile.
- b. Programmer menulis program secara berpasangan (pair) dan melakukan pengujian koding.
- c. Koding dan testing dilakukan secara bersamaan
- d. Sangat bermanfaat ketika mengembangkan sistem skala besar, kompleks TPS.



# Quiz

Ketika prototyping dimasukkan pada pendekatan SDLC, umumnya dimasukkan sebagai bagian dari langkah berikut:

- a. Planning
- b. Analisa
- c. Design
- d. Implementation
- c. Maintenance

# Quiz

Berikut ini adalah karakteristik dari metode Agile:

- a. Short releases and iterations
- b. Melibatkan user
- c. Dokumentasi yang minimal
- d. Semua yang diatas.

# Quiz

Berikut ini yang merupakan ciri dari pendekatan prototyping:

- a. Digunakan oleh perusahaan untuk pendekatan kepada teknologi yang lebih baru.
- b. Dokumentasi kurang lengkap dibandingkan dengan pendekatan SDLC.
- b. Fitur keamanan dan kendali yang lebih rendah dibandingkan dengan pendekatan SDLC tradisional.
- d. Semua yang disebut diatas

# Quiz

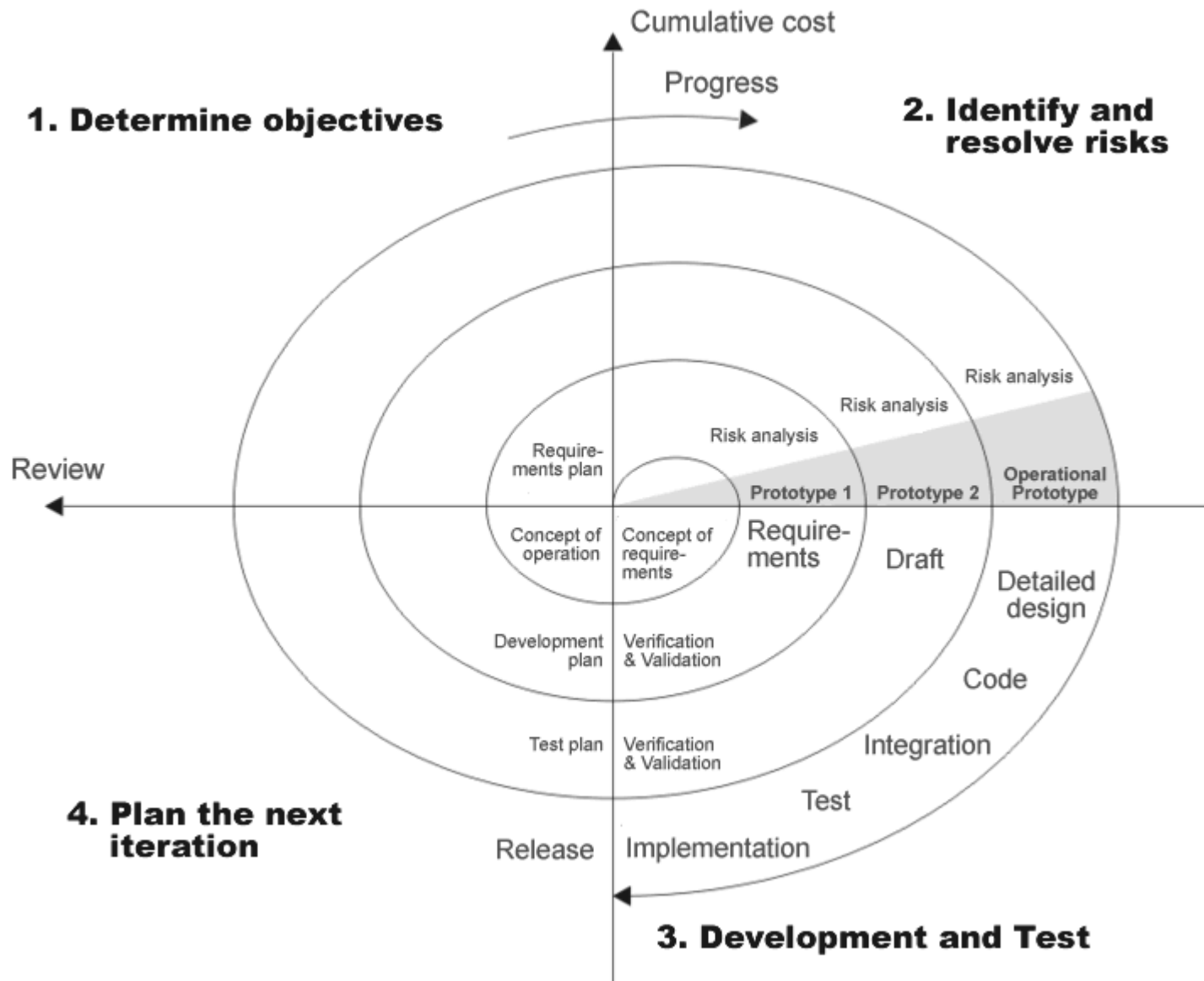
Salah satu praktek utama pada XP adalah memastikan semua kode yang diproduksi ditulis oleh dua programmer yang bekerja pada satu fasilitas komputer. Ini adalah langkah revisi yang membawa kepada extreme. Semua kode selalu direview oleh minimal satu orang:

- a. Pair programming
- b. Test driven developoment
- c. Design improvement (refactoring)
- d. Semua yang diatas salah.

# Quiz

Pendekatan evolusioner terhadap software development yang diusulkan oleh Barry Boehm yang terdiri dari 4 kuadran, dimana proyek software melintasi masing2 kuadran sebagaimana berkembang secara incremental.

- a. Linear sequential model
- b. Spiral model
- c. Incremental model
- d. eXtreme Programming



# Quiz

Suatu kelompok metodologi software development yang menghasilkan software didalam iterasi pendek dan memperbolehkan perubahan besar atau design:

- a. Agile
- b. Tradisional Model
- c. a dan b benar
- d. Tidak ada pilihan yang benar

**Sekian & Terima Kasih**