

**Nama : Adi Saepul Anwar**

**Kelas : TI/3/A/M**

**Matkul : Sistem Informasi Manajement**

### **UAS Sistem Informasi Management**

#### **Soal.**

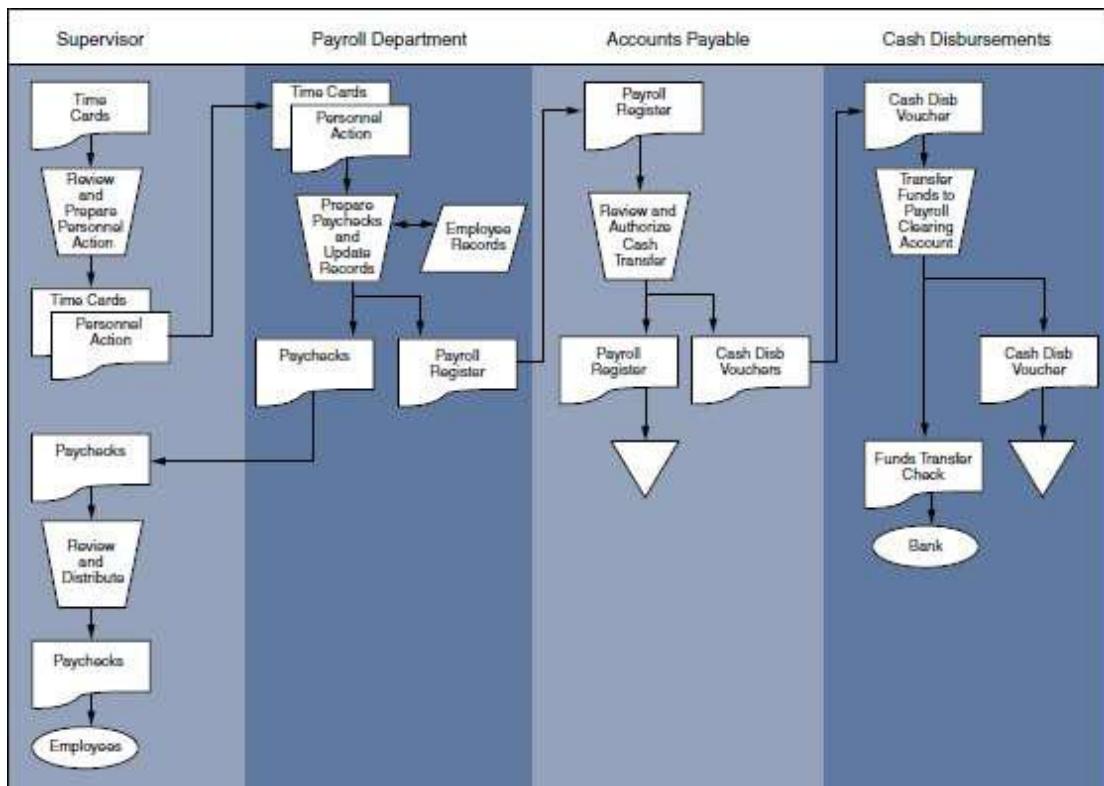
1. Cari sebuah Flowchart dari sebuah kegiatan dalam sebuah perusahaan, dan tidak boleh sama dengan teman. dari flow tersebut tentukan :
  - a. aliran datanya
  - c. tentukan kebutuhan tabel tabelnya
  - b. aliran dokument
  - d. dari tabel jawaban di C kembangkan menjadi 20 minimal tabel apa saja yang mendukung dari sebuah kegiatan bisnis perusahaan itu
2. sebuatkan metode pengembangan sebuah SIM dan berikan contohnya, kekurangannya, dan kelebihannya.

jawaban minimal 8 lembar, type timeroman, size 12, spasi 1.

di kumpulkan di waktu UAS lewat calass room

3. silahkan buatkan rancangan pengembangan dari sebuah aplikasi dari SIAKAD khusus untuk aplikasi untuk mahasiswa seperti kalian gunakan selama ini.

#### **1. Flow Payroll**



#### A. Aliran data

1. Supervisor
2. Payroll departement
3. Akunting
4. Bendahara

#### B. Aliran Dokumen

1. Time Card
2. Personal Action
3. Paycheck
4. Payroll Register
5. Cash Dist Voucher
6. Fund Trf Check

#### C. Tabel yang dibutuhkan

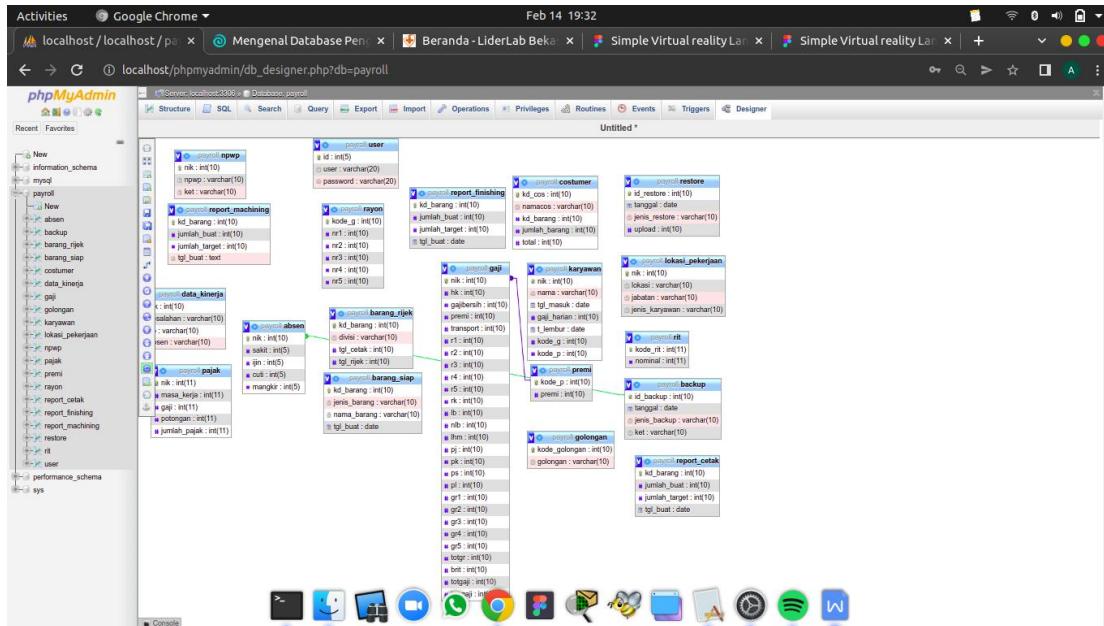
- A. User
- B. Absen
- C. Golongan
- D. Premi

E. Rayon

F. Karyawan

G. Gaji

#### D. Pengembangan Tabel



## 2. Metodologi Pengembangan Sistem Informasi Management

Metodologi pengembangan sistem adalah suatu proses pengembangan sistem yang formal dan presisi yang mendefinisikan serangkaian aktivitas, metode, best practices dan tools yang terautomasi bagi para pengembang dan manager proyek dalam rangka mengembangkan dan merawat sebagai keseluruhan sistem informasi atau software.

### Macam-Macam Metodologi Pengembangan Sistem

#### a. Metode *System Development Life Cycle (SDLC)*

Metode ini adalah metode pengembangan sistem informasi yang pertama kali digunakan makanya disebut dengan metode tradisional atau metode klasik. Metode ini digunakan untuk mengembangkan, memelihara, dan menggunakan sistem informasi.

#### b. Model *Waterfall*

Sering juga disebut model Sequential Linier. Metode pengembangan sistem yang paling tua dan paling sederhana. Cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah. Model ini menyediakan pendekatan alur hidup perangkat lunak secara sequential atau terurut dimulai dari analisa, desain, pengkodean, pengujian dan tahap pendukung.

#### c. Model *Prototyping*

Prototyping adalah proses iterative dalam pengembangan sistem dimana requirement diubah ke dalam sistem yang bekerja (working system) yang secara terus menerus diperbaiki melalui kerjasama antara user dan analis. Prototipe juga bisa dibangun melalui beberapa tool pengembangan untuk menyederhanakan proses.

d. Model *RAD* (*Rapid Application Development*)

RAD adalah penggabungan beberapa metode atau teknik terstruktur. RAD menggunakan metode prototyping dan teknik terstruktur lainnya untuk menentukan kebutuhan user dan perancangan sistem informasi. Selain itu RAD menekankan siklus perkembangan dalam waktu yang singkat (60 sampai 90 hari) dengan pendekatan konstruksi berbasis komponen.

e. Model *Spiral*

Model spiral pada awalnya diusulkan oleh Boehm, adalah model proses perangkat lunak evolusioner yang merangkai sifat iteratif dari prototipe dengan cara kontrol dan aspek sistematis model sequensial linier. Model iteratif ditandai dengan tingkah laku yang memungkinkan pengembang mengembangkan versi perangkat lunak yang lebih lengkap secara bertahap.

f. *Object Oriented Technology*

Object Oriented Technology merupakan cara pengembangan perangkat lunak berdasarkan abstraksi objek-objek yang ada di dunia nyata. Dasar pembuatan adalah objek yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas. Filosofi Object Oriented sangat luar biasa sepanjang siklus pengembangan perangkat lunak (perencanaan, analisis, perancangan dan implementasi) sehingga dapat diterapkan pada perancangan sistem secara umum: menyangkut perangkat lunak, perangkat keras dan sistem secara keseluruhan.

g. Metode *End-user Development*

Disini pengembangan dilakukan langsung oleh end-user. Keterlibatan langsung end-user sangat menguntungkan, karena memahami benar bagaimana sistem bekerja. Artinya tahap analisis sistem dapat dilakukan lebih cepat. Kelemahan adalah pada pengendalian mutu dan kecenderungan tumbuhnya “private” sistem informasi. Integrasi dengan sistem yang lain menjadi sulit.

## Tahapan Metode Pengembangan

### 1. Tahapan metode SDLC

#### Perencanaan sistem (*System Planning*)

Lebih menekankan pada aspek studi kelayakan pengembangan sistem (feasibility study). Aktivitas yang ada meliputi:

- Pembentukan konsolidasi tim pengembang
- Mendefinisikan tujuan dan ruang lingkup pengembangan
- Mengidentifikasi masalah-masalah yang ada, bisa diselesaikan melalui pengembangan sistem

- Menentukan dan evaluasi strategi yang akan digunakan dalam pengembangan sistem
- Penentuan prioritas teknologi dan pemilihan aplikasi.
- Analisis sistem (System Analysis)

Analisis sistem adalah tahap di mana dilakukan beberapa aktivitas berikut:

- Melakukan studi literature untuk menemukan suatu kasus yang bisa ditangani oleh sistem
- Brainstorming adalah tim pengembang mengenai kasus mana yang paling tepat dimodelkan dengan sistem
- Mengklasifikasikan masalah, peluang dan solusi yang mungkin diterapkan untuk kasus tersebut
- Analisa kebutuhan pada sistem dan membuat batasan sistem
- Mendefinisikan kebutuhan sistem.
- Perancangan sistem (System Design)

Pada tahap ini, features dan operasi-operasi pada sistem dideskripsikan secara detail. Aktivitas-aktivitas yang dilakukan adalah:

- Menganalisa interaksi obyek dan fungsi pada sistem
- Menganalisa data dan membuat skema database
- Merancang user interface.

### *Implementasi sistem (System Implementation)*

Tahap berikutnya yaitu mengimplementasikan rancangan dari tahap-tahap sebelumnya dan melakukan uji coba. Dalam implementasi, dilakukan aktivitas-aktivitas sebagai berikut:

- Pembuatan database sesuai skema rancangan
- Pembuatan aplikasi berdasarkan desain sistem
- Pengujian dan perbaikan aplikasi (debugging)
- Pemeliharaan sistem (System Maintenance)

Dilakukan oleh admin yang ditunjuk untuk menjaga sistem tetap mampu beroperasi secara benar melalui kemampuan sistem dalam mengadaptasikan diri sesuai dengan kebutuhan.

## **2. Tahapan metode Waterfall**

### *Analisis kebutuhan (Requirement)*

Dalam tahap ini merupakan analisa terhadap kebutuhan sistem. Pengumpul data pada tahap ini bisa melakukan sebuah penelitian, wawancara, atau studi literature. Seseorang analisis sistem akan menggali informasi sebanyak-banyaknya dari user sehingga akan tercipta sebuah sistem computer yang bisa melakukan tugas-tugas yang diinginkan oleh user tersebut. Tahapan ini akan menghasilkan dokumen user requirement atau bisa dikatakan sebagai data yang berhubungan dengan keinginan user

dalam pembuatan sistem. Dokumen inilah yang akan menjadi acuan sistem analisi untuk menterjemahkan ke dalam bahasa pemrograman.

#### Sistem desain (*Design System*)

Proses desain akan menterjemahkan syarat kebutuhan kesebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat koding. Proses ini berfokus pada struktur data, arsitektur perangkat lunak, representasi interface, dan detail (algoritma) procedural. Tahapan ini kan menghasilkan dokumen yang disebut software requirement. Dokumen inilah yang akan digunakan programmer untuk melakukan aktivitas pembuatan sistemnya.

#### Penulisan sinkode program/implementation (*Coding & Testing*)

Coding merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh computer. Dilakukan oleh programmer yang akan menterjemahkan transaksi yang diminta oleh user. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam artian penggunaan computer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat tadi. Tujuan testing adalah menemukan kesalahan-kesalahan terhadap sistem tersebut dan kemudian bisa diperbaiki.

#### Penerapan/pengujian program (*Integration & Testing*)

Tahapan ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisa, desain dan pengkodean maka sistem yang sudah dijadikan digunakan oleh user.

#### Pemeliharaan (*Operation & Maintenance*)

Perangkat lunak yang susah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan baru, atau karena pelanggan membutuhkan perkembangan fungsional.

### **3. Tahapan metode *Prototyping***

#### Pengumpulan kebutuhan

Pelanggan dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.

- Membangun *prototyping*, membangun *prototyping* dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya dengan membuat input dan format output)
- Evaluasi *prototyping*, evaluasi ini dilakukan oleh pelanggan apakah *prototyping* yang sudah dibangun sudah sesuai dengan keinginan pelanggan.
- Mengkodekan sistem, dalam tahap ini *prototyping* yang sudah disepakati kemudian diterjemahkan ke dalam bahasa pemrograman yang sesuai.

- Menguji sistem, setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus dites dahulu sebelum digunakan. Pengujian ini dilakukan dengan White Box, Black Box, Basis Path, pengujian arsitektur dan lain-lain.
- Evaluasi sistem, pelanggan mengevaluasi apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan.
- Menggunakan sistem, perangkat lunak yang telah diuji dan dites pelanggan siap untuk digunakan.

#### **4. Tahapan metode RAD**

Requirement planning

Dalam tahap ini diketahui apa saja yang menjadi kebutuhan sistem yaitu dengan mengidentifikasi kebutuhan informasi dan masalah yang dihadapi untuk menentukan tujuan, batasan-batasan sistem, kendala dan juga alternative pemecahan masalah. Analisis digunakan untuk mengetahui perilaku sistem dan juga untuk mengetahui aktivitas apa saja yang ada dalam sistem tersebut.

Design workshop

Mengidentifikasi solusi alternatif dengan memilih solusi yang terbaik. Kemudian membuat desain proses bisnis dan desain pemrograman untuk data-data yang telah didapatkan dan dimodelkan dalam arsitektur sistem informasi. Tools yang digunakan dalam pemodelan sistem biasanya menggunakan UML (Unified Modeling Language).

Implementation

Sistem diimplementasikan ke dalam bentuk yang dimengerti oleh mesin yang diwujudkan dalam bentuk program atau unit program. Tahap ini merupakan tahan meletakkan sistem agar siap untuk dioperasikan.

#### **5. Tahapan Metode Spiral**

- Tahap Liason, pada tahap ini membangun komunikasi yang efektif di antara pengembang dan pelanggan.
- Tahap Planning, pada tahap ini ditentukan sumber-sumber informasi, batas waktu dan informasi-informasi yang dapat menjelaskan proyek.
- Tahap Analisis Resiko, mendefinisikan resiko, menentukan apa saja yang menjadi resiko baik teknis maupun manajemen.
- Tahap Rekayasa, pembuatan prototype atau pembangunan satu atau lebih representasi dari aplikasi tersebut.
- Tahap Kontruksi dan Pelepasan, pada tahap ini dilakukan pembangunan perangkat lunak yang dimaksud, diuji, diinstall dan diberikan sokongan-sokongan tambahan untuk keberhasilan proyek.
- Tahap Evaluasi, pelanggan biasanya memberi masukan berdasarkan hasil yang didapat dari tahap engineering dan instalasi.

**Keunggulan dan Kelehaman**

#### **1. Metode SDLC**

Keunggulan :

- Mudah diaplikasikan,
- Memberikan template tentang metode analisis, desain, pengkodean, pengujian, dan pemeliharaan.

Kelemahan :

- Jarang sekali proyek riil mengikuti aliran sekuensial yang dianjurkan model karena model ini bisa melakukan iterasi tidak langsung.
- Pelanggan sulit untuk menyatakan kebutuhan secara eksplisit sehingga sulit untuk megakomodasi ketidakpastian pada saat awal proyek.
- Pelanggan harus bersikap sabar karena harus menunggu sampai akhir proyek dilalui. Sebuah kesalahan jika tidak diketahui dari awal akan menjadi masalah besar karena harus mengulang dari awal.
- Pengembang sering melakukan penundaan yang tidak perlu karena anggota tim proyek harus menunggu tim lain untuk melengkapi tugas karena memiliki ketergantungan hal ini menyebabkan penggunaan waktu tidak efisien.

## **2. Metode Waterfall**

Keunggulan :

- Kualitas dari sistem yang dihasilkan akan baik. Ini dikarenakan oleh pelaksanaannya secara bertahap. Sehingga tidak terfokus pada tahapan tertentu.
- Dokumen pengembangan sistem sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya. Jadi setiap fase atau tahapan akan mempunyai dokumen tertentu.

Kelemahan :

- Diperlukan majemen yang baik, karena proses pengembangan tidak dapat dilakukan secara berulang sebelum terjadinya suatu produk.
- Kesalahan kecil akan menjadi masalah besar jika tidak diketahui sejak awal pengembangan.
- Pelanggan sulit menyatakan kebutuhan secara eksplisit sehingga tidak dapat mengakomodasi ketidakpastian pada saat awal pengembangan.

## **3. Prototyping**

Keunggulan :

- Adanya komunikasi antara pengembang dan pelanggan
- Pengembang dapat bekerja lebih baik
- Pelanggan berperan aktif dalam pengembangan
- Sangat cepat dan lebih menghemat waktu
- Harapan menjadi lebih baik

Kelemahan :

- Untuk menghemat waktu, biasanya pengembang hanya menggunakan bahasa pemrograman sederhana yang mungkin rentan dari segi keamanannya.

- Tidak cocok untuk diimplementasikan pada sebuah sistem yang sangat besar dan global, seperti sistem operasi computer.

#### **4. RAD**

Keunggulan :

- RAD mengikuti tahapan pengembangan sistem seperti umumnya, tetapi mempunyai kemampuan untuk menggunakan kembali komponen yang ada (reusable object).
- Setiap fungsi dapat dimodulkan dalam waktu tertentu dan dapat dibicarakan oleh tim RAD yang terpisah dan kemudian diintegrasikan sehingga waktunya lebih efisien.

Kelemahan :

- Tidak cocok untuk proyek skala besar
- Proyek bisa gagal karena waktu yang disepakati tidak dipenuhi.
- Sistem yang tidak bisa dimodularisasi tidak cocok untuk model ini.
- Resiko teknis yang tinggi juga kurang cocok untuk model ini

#### **5. Spiral**

Keunggulan :

- Dapat disesuaikan agar perangkat lunak bisa dipakai selama hidup perangkat lunak komputer.
- Lebih cocok untuk pengembangan sistem dan perangkat lunak skala besar
- Pengembang dan pemakai dapat lebih mudah memahami dan bereaksi terhadap resiko setiap tingkat evolusi karena perangkat lunak terus bekerja selama proses
- Menggunakan prototipe sebagai mekanisme pengurangan resiko dan pada setiap keadaan di dalam evolusi produk.
- Tetap mengikuti langkah-langkah dalam siklus kehidupan klasik dan memasukkannya ke dalam kerangka kerja iterative
- Membutuhkan pertimbangan langsung terhadap resiko teknis sehingga mengurangi resiko sebelum menjadi permasalahan yang serius.

Kelemahan :

- Sulit untuk menyakinkan pelanggan bahwa pendekatan evolusioner ini bisa dikontrol.
- Memerlukan penaksiran resiko yang masuk akal dan akan menjadi masalah yang serius jika resiko mayor tidak ditemukan dan diatur.
- Butuh waktu lama untuk menerapkan paradigma ini menuju kepastian yang absolut

#### **6. Object oriented technology**

Keunggulan :

- Uniformity, OMT memungkinkan merancang user interface secara terintegrasi bersama dengan perancangan perangkat lunak sekaligus dengan perancangan basis data.
- Understandability, Kode-kode yang dihasilkan dapat diorganisasi ke dalam kelas-kels yang berhubungan dengan masalah sesungguhnya sehingga lebih mudah dipahami.
- Stability, Kode program yang dihasilkan relatif stabil sebab mendekati permasalahan sesungguhnya dilapangan.
- Reusability, Dimungkinkan penggunaan kembali kode-kode sehingga akan mempercepat waktu pengembangan perangkat lunak.

Kelemahan :

- Metode berorientasi objek merupakan konsep yang relatif baru sehingga belum ada standar yang diterima semua pihak dalam menentukan tool apa yang digunakan sebagai dasar analisis dan perancangan perangkat lunak.

## 7. EUD

Keunggulan :

- Dapat menghindari permasalahan kemacetan di departemen sistem informasi.
- Kebutuhan pemakai sistem dapat lebih terpenuhi karena dapat dikembangkan sendiri oleh pemakai.
- Menambah atau meningkatkan partisipasi aktif pemakai dalam proses pengembangan sistemnya sehingga akan ada kepuasan sendiri dari pemakai sistem.
- Dapat menambah kualitas pemahaman pemakai terhadap aplikasi yang dikembangkan serta teknologi yang digunakan dalam sistem.

Kelemahan :

- Karena pemakai sistem harus mengembangkan aplikasinya sendiri, maka dalam hal ini pemakai sekaligus pengembang sistem dituntut untuk memiliki pemahaman mengenai teknologi informasi (computer literacy) serta pemahaman tentang pengembangan sistem infomasi.
- End user computing memiliki resiko dapat mengganggu bahkan merusak sistem informasi di luar yang dikembangkan oleh pemakai sistem.
- End user computing pasti akan berhadapan dengan masalah kemampuan teknis pemakai sekaligus pengembang sistem.

## 3. Pengembangan Siakad

Pengembangannya sendiri saya lebih ke user experience yang mana masih banyak menu ataupun sub menu yang tidak sinkron ketika berbeda device itu sangat mengganggu user(contoh diatas), ada baiknya ditambah fitur responsif pada website.