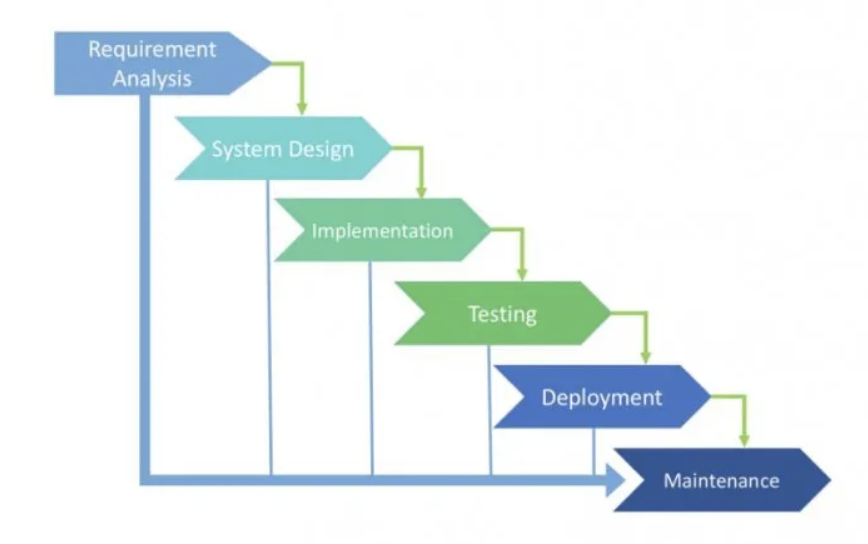
Pengembangan perangkat lunak sendiri dapat dilakukan melalui berbagai metode. Sedikitnya beberapa metode yang paling banyak digunakan oleh para pengembang perangkat lunak. beberapa metode tersebut akan dibahas satu per satu berikut ini.

**Metode *Waterfall***



Waterfall merupakan metode pengembangan perangkat lunak tradisional yang sistematis. Metode ini memiliki lima tahapan proses, di antaranya Communication, Planning, Modeling, Construction, dan Deployment.

Communication merupakan fase di mana pelanggan atau pemilik proyek menyampaikan kebutuhan dan permasalahannya kepada pengembang. Lalu, bersama-sama mengumpulkan data-data yang diperlukan dan merumuskan fitur-fitur perangkat lunak.

Selanjutnya, menginjak pada proses perancangan. Dimulai dengan merumuskan estimasi kerja, kebutuhan sumber daya, serta perencanaan alur kerja. Berlanjut dengan tahap perancangan struktur data, arsitektur, tampilan, dan algoritma perangkat lunak.

Rancangan kemudian coba diaplikasikan pada perangkat keras komputer dalam bentuk bahasa pemograman. Construction juga mencakup tahapan uji coba pengoperasian perangkat lunak untuk mengetahui kelemahannya.

Setelah berhasil dibuat, perangkat lunak disebarluaskan untuk diimplementasikan pada perangkat pengguna secara umum. Temuan-temuan dari pengguna, akan menjadi bahan bagi pengembang untuk mengevaluasi dan memperbaiki perangkat lunak lebih jauh lagi.

**Kekurangan dan Kelebihan Metode Waterfall**

Metode waterfall melibatkan berbagai proses yang sistematis dan komprehensif. Sumber daya dan tahapan pengerjaannya dikumpulkan secara lengkap sehingga dapat mencapai hasil maksimal. Sayangnya, proses tersebut memakan waktu lebih lama.

Sistem dalam metode waterfall merupakan proses yang baku, sehingga pengembang sulit melakukan improvisasi. Itulah mengapa metode ini dianggap kurang efektif dan seringkali hanya dipakai dalam pengembangan perangkat lunak atau sistem berskala besar

#### ****Metode****Prototype

#### 

Prototype dalam bahasa Indonesia diartikan dengan istilah purwarupa. Istilah tersebut berarti model awal atau rancangan sementara yang masih membutuhkan berbagai penyesuaian sebelum dinyatakan telah memenuhi hasil yang diinginkan.

Terdapat lima tahapan dalam pengembangan perangkat lunak menggunakan metode prototype. Dimulai dengan pengumpulan informasi dari pelanggan ke tim pengembang. Lalu, tim akan merencanakan sistem dan mengerjakan purwarupa perangkat lunak.

Hasilnya kemudian diserahkan kepada pelanggan untuk dievaluasi. Jika terdapat permasalahan, tim akan merevisi sistem tersebut hingga benar-benar sesuai dengan kebutuhan pelanggan.

Ketika revisi selesai dan telah diterima, berarti perangkat lunak telah siap diterjemahkan ke dalam perangkat keras. Kemudian, dilanjutkan dengan proses uji coba dan berbagai revisi sebelum mulai dapat digunakan.

**Kekurangan dan Kelebihan Metode Prototype**

Metode prototype digunakan apabila pemilik proyek tahu benar apa yang diinginkannya, tetapi tidak mengetahui bagaimana cara mengaplikasikannya. Kuncinya terletak pada komunikasi yang baik antara pelanggan dan pengembang.

Pelanggan harus menyampaikan kebutuhannya secara jelas. Sementara pengembang juga diharapkan mampu menerjemahkan informasi tersebut agar dapat menghasilkan perangkat lunak yang sesuai. Jika tidak, prototype ini tidak akan menjadi metode yang efektif.

Pengembang memperoleh tantangan besar dari pelanggan. Sebaliknya, pelanggan dapat terpuaskan jika pengembang berhasil memenuhi kebutuhannya. Kerja sama kedua pihak akan saling menguntungkan.

#### ****Metode Spiral****

#### 

Metode spiral menggabungkan dua metode pengembangan yang telah dibahas sebelumnya, yaitu prototype dan waterfall. Pengembang melaksanakan prototyping dengan cara sistematis khas metode waterfall.

Umumnya metode spiral diterapkan dalam pengembangan perangkat lunak berskala besar, sekaligus membutuhkan sistem yang kompleks. Setiap prosesnya selalu disertai dengan analisis mendalam mengenai tingkat risiko dan keberhasilan pengembangan.

Pelaksanaan metode spiral dilakukan dalam lima langkah. Pertama adalah komunikasi, yaitu pemilik proyek menyampaikan kebutuhannya kepada pengembang perangkat runak. Dilanjutkan dengan perencanaan mendetail tentang proyek yang digarap.

Langkah perencanaan diikuti dengan analisis untuk mengidentifikasi berbagai kemungkinan yang bisa terjadi selama pengembangan. Kemudian, pengembangan perangkat lunak mulai dijalankan dan setelah jadi akan mendapatkan evaluasi dari pelanggan.

Kelima langkah tersebut dilakukan secara berulang-ulang pada setiap tahapan pengembangan. Sejak dari pengembangan konsep, pengembangan prototype, perbaikan, perubahan, hingga pemeliharaan sistem yang telah jadi.

**Kekurangan dan Kelebihan Metode Spiral**

Metode spiral menerapkan alur kerja yang kompleks, panjang, dan memakan waktu lama. Metode ini tidak cocok untuk proyek kecil-kecilan, apalagi yang berbujet rendah. Seba

Selain itu, metode spiral juga kurang tepat jika dilakukan oleh pengembang perangkat lunak yang belum berpengalaman. Prosesnya bisa menjadi terlalu kompleks bagi pemula yang masih memerlukan banyak latihan.

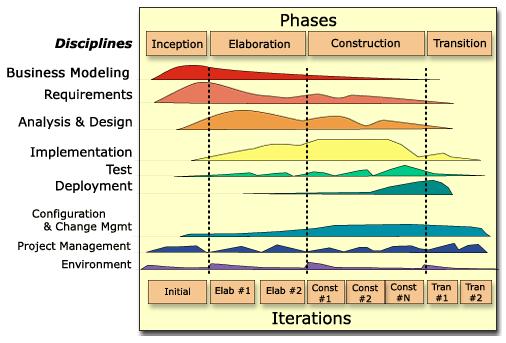
**RUP (Rational Unified Process**

Rational Unified Process (RUP) merupakan suatu metode rekayasa perangkat lunak yang dikembangkan dengan mengumpulkan berbagai *best practises* yang terdapat dalam industri pengembangan perangkat lunak. Ciri utama metode ini adalah menggunakan *use-case driven* dan pendekatan iteratif untuk siklus pengembangan perankat lunak. Gambar dibawah menunjukkan secara keseluruhan arsitektur yang dimiliki RUP.

RUP menggunakan konsep *object oriented*, dengan aktifitas yang berfokus pada pengembangan model dengan menggunakan *Unified Model Language* (UML). Melalui gambar dibawah dapat dilihat bahwa RUP memiliki, yaitu:

* **Dimensi pertama** digambarkan secara horizontal. Dimensi ini mewakili aspek-aspek dinamis dari pengembangan perangkat lunak. Aspek ini dijabarkan dalam tahapan pengembangan atau fase. Setiap fase akan memiliki suatu *major milestone* yang menandakan akhir dari awal dari phase selanjutnya. Setiap phase dapat berdiri dari satu beberapa iterasi. Dimensi ini terdiri atas *Inception*, *Elaboration*, *Construction*, dan *Transition*.
* **Dimensi kedua** digambarkan secara vertikal. Dimensi ini mewakili aspek-aspek statis dari proses pengembangan perangkat lunak yang dikelompokkan ke dalam beberapa disiplin. Proses pengembangan perangkat lunak yang dijelaskan kedalam beberapa disiplin terdiri dari empat elemen penting, yakni *who is doing*, *what*, *how* dan *when*. Dimensi ini terdiri atas

*Business Modeling, Requirement, Analysis and Design, Implementation, Test, Deployment, Configuration* dan *Change Manegement, Project Management, Environtment.*



Gambar Arsitektur *Rational Unified Process*

**Fase RUP**

* Inception/insepsi
* Elaboration/elaborasi
* Construction/konstruksi
* Transition/transisi
* Inception
  + Menentukan Ruang lingkup proyek
  + Membuat ‘Business Case’
  + Menjawab pertanyaan “apakah yang dikerjakan dapat menciptakan ‘good business sense’ sehingga proyek dapat dilanjutkan
* Elaboration
  + Menganalisa berbagai persyaratan dan resiko
  + Menetapkan ‘base line’
  + Merencanakan fase berikutnya yaitu construction
* Construction
  + Melakukan sederetan iterasi
  + Pada setiap iterasi akan melibatkan proses berikut: analisa desain, implementasi dan testing
* Transistion
  + Membuat apa yang sudah dimodelkan menjadi suatu produk jadi
  + Dalam fase ini dilakukan:
    - Beta dan performance testing
    - Membuat dokumentasi tambahan seperti; training, user guides dan sales kit
    - Membuat rencana peluncuran produk ke komunitas pengguna

**Peran Use Case Pada Setiap Fase**

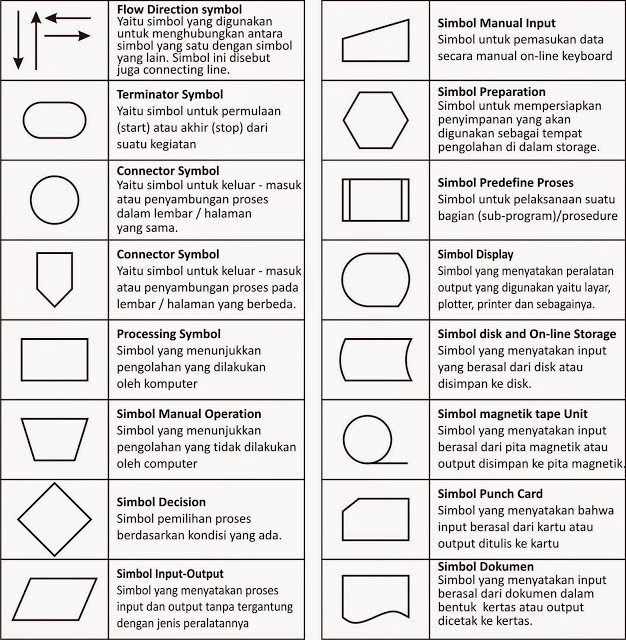
* Inception
  + Menolong mengembangkan scope proyek
  + Menolong menetapkan penjadwalan dan anggaran
* Elaboration
  + Menolong dalam melakukan analisa resiko
  + Menolong mempersiapkan fase berikutnya yaitu konstruksi
* Construction
  + Melakukan sederetan iterasi
  + Pada setiap iterasi akan akan melibatkan proses berikut: analisa desain, implementasi dan testing
* Transistion
  + Membuat apa yang sudah dimodelkan menjadi suatu produk jadi
  + Dalam fase ini dilakukan:
    - Beta dan performance testing
    - Membuat dokumentasi tambahan seperti; training, user guides dan sales kit
    - Membuat rencana peluncuran produk ke komunitas pengguna

**Flowchart**

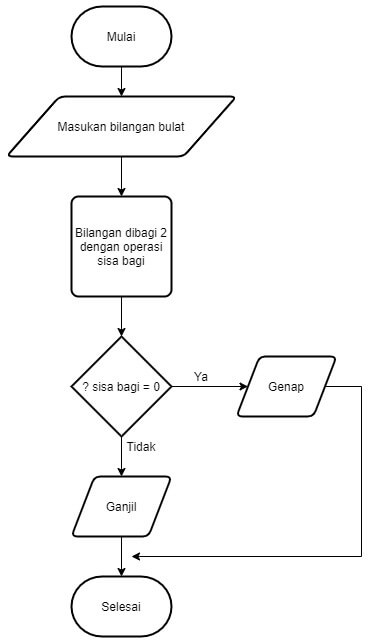
Flowchart adalah sebuah diagram yang menjelaskan alur proses dari sebuah program. Dalam membangun sebuah program, flowchart berperan penting untuk menerjemahkan proses berjalannya sebuah program agar lebih mudah untuk dipahami

Setiap langkah digambarkan dalam bentuk diagram dan dihubungkan dengan garis atau arah panah.

Flowchart berperan penting dalam memutuskan sebuah langkah atau fungsionalitas dari sebuah proyek pembuatan program yang melibatkan banyak orang sekaligus. Selain itu dengan menggunakan bagan alur proses dari sebuah program akan lebih jelas, ringkas, dan mengurangi kemungkinan untuk salah penafsiran. Penggunaan flowchart dalam dunia pemrograman juga merupakan cara yang bagus untuk menghubungkan antara kebutuhan teknis dan non-teknis.



contoh flowchart sederhana untuk menentukan apakah bilangan yang dimasukan ganjil atau genap



##### **Prosedur Pemunjaman Buku**

* + 1. Peminjam memilih buku sendiri, kemudian membawanya ke tempat pelayanan.
    2. Menunjukkan Kartu Mahasiswa dan menyebutkan nama terangnya.
    3. Menuliskan nama, NIM/NIP di slip pengembalian (halaman belakang/depan buku).
    4. Petugas memproses di komputer dan memberi cap tanggal kembali di slip.
    5. Menggunakan kartu/namanya sendiri.

