

TEMA 1

Enginyeria del programari

Disciplina al voltant de la creació del programari. Permet:

- Entendre els problemes i pensar una solució.
- Definir i aplicar teories, ferramentes i mètodes adequats a cada problema.
 - Pensar i valorar alternatives.
 - Després, desenvolupar una solució de qualitat.

Conceptes bàsics:

- **Procés:** Defineixen la seqüència d'aplicació dels mètodes i l'ús de les eines adequades.
- **Mètodes i tècniques:** Tècniques/Pràctiques que indiquen com construir el programari.
- **Eines (Tools):** Donen suport a l'execució dels mètodes.

Producte: Programari i sistema informàtic

Un **producte** és un conjunt de programes que s'executen de manera ordenada i processen continguts, normalment dades, proporcionant uns resultats determinats.

El **programari** és un producte i al mateix temps un mitjà per a gestionar i proveir altres productes. Les seues característiques son:

- El programari **es desenvolupa** mentre que el maquinari es construeix i fabrica.
- **No es trenca** ni es desgasta.
- És un producte **lògic**, necessita el suport de dispositius i altres elements per a funcionar.
- **No és tangible** per al destinatari, menys la part visual (interfícies).
- **No pot funcionar de manera aïllada**, necessita d'altres components que el complementen.

Sistema informàtic: Conjunt o disposició d'elements, que inclouen maquinari i programari i que estan organitzats per portar a terme un objectiu predefinit processant dades i informació. Components:

- Programari
- Bases de dades
- Maquinari
- Eines de comunicació
- A més d'aquestes s'inclouen: Persones; Processos/Procediments; Lleis i entorn

Condicions per a desenvolupar **programari de qualitat**:

- Ha de ser **mantenible**: Poden canviar i evolucionar (modificable) i pot ser usat per diferents empreses (adaptable).
- Ha de ser **fiable**: Seguretat en el funcionament (no errors) i segur per a les persones (no accidents)
- Ha de ser **eficient**: Ús mínim de recursos i el cost d'aquests es compensa amb els beneficis.
- Ha de proporcionar una **interfície apropiada i adaptada a l'usuari**.

Classificació del programari

Depenent de l'ús o objectiu al qual es destina el programari es pot classificar en:

- De sistemes: Gestionar altres productes de programari i ajuda a executar el maquinari.
- Per a desenvolupar programari: Editors, compiladors, gestors BBDD
- Aplicacions per a negocis: Solucionar necessitats
- Programari científic: Suport a la recerca i la investigació
- Encastat (*embedded*): Inclós en un producte o sistema per a controlar-lo.
- Software d'usuari o d'oficina: Fulls de càlcul, processadors de textos
- Aplicacions en tecnologies web
- Intel·ligència artificial.

Depenent de les característiques de **la seua distribució, desenvolupament i portabilitat**, es classifica en:

- Programari de **fabricació estàndard**.
- **Legacy software**: programari fet a mida.
- Programari de **codi obert**.
- **Open World Computing**: programari que es puga usar des de diferents dispositius i en qualsevol localització.

Procés en enginyeria del programari

Cinc activitats fonamentals en un procés de desenvolupament de programari:

- **Comunicació**: establir els objectius principals, recopilar necessitats que permeten identificar les característiques i prioritats del producte a desenvolupar des d'un nivell alt.
- **Planificació**: defineix el pla del projecte que inclou les tasques tècniques que cal dur a terme, els recursos que seran necessaris, els productes i el calendari temporal del projecte.
- **Modelització**: gràfics, esquemes i documentació per a comprendre quin és el producte d'enginyeria que s'ha de construir
- **Construcció**: generació de codi i les proves necessàries per a produir un producte final de qualitat.
- **Posada en marxa**: Lliurament del producte a l'usuari. (Deployment)

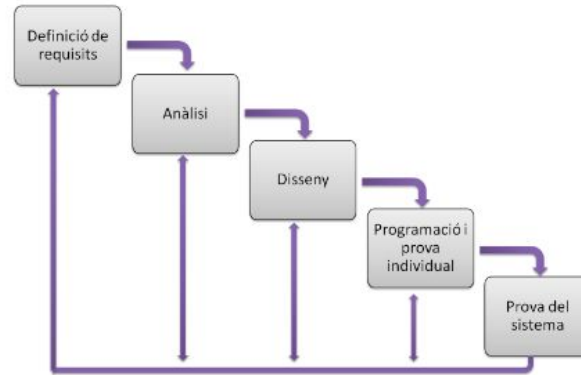
En l'assignatura, les activitats a seguir serán:

- Comunicació
- Planificació
- Definició de requisits
- Anàlisi
- Disseny

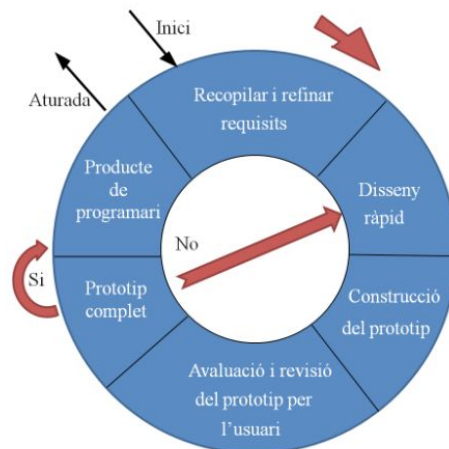
Models del procés

En l'enginyeria del programari s'han definit diferents propostes de procés que es denominen model o paradigma. Tipus:

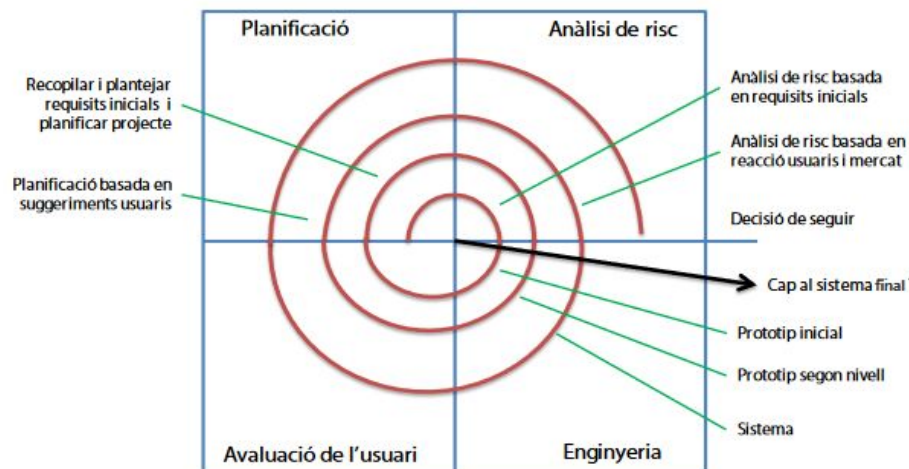
- **Model en cascada** (pasos un a continuació del altre)



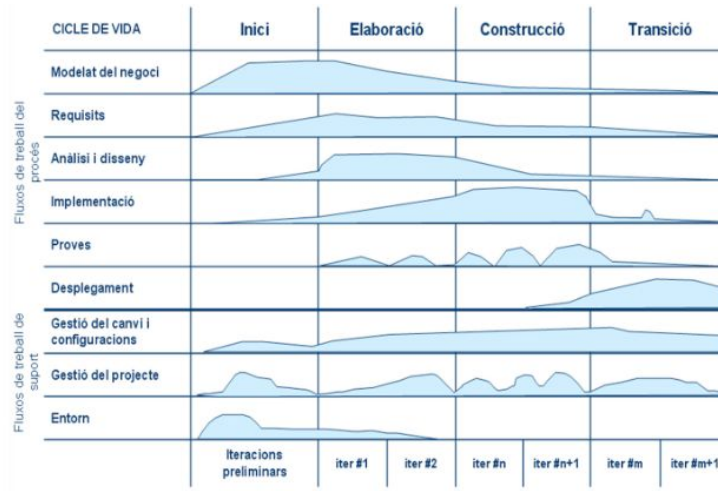
- **Desenvolupament de prototips** (fer una versió experimental)



- **Model en espiral** (cascada + prototipus)



- **The Unified Process:**



- **Mètodes àgils.** Segueixen els punts del Manifesto for Agile Software Development:
 - Individus i interaccions per damunt dels processos i ferramentes.
 - Programari que funciona per damunt d'una extensa documentació
 - Col·laboració amb el client per damunt de la negociació d'un contracte.
 - Resposta al canvi per damunt del seguiment d'un pla.

TEMA 2

Un projecte és un conjunt d'activitats organitzades en fases, tasques i subtasques agrupades amb diferents criteris, que segueixen mètodes i usen tècniques de desenvolupament de programari, amb l'objectiu de desenvolupar un producte de programari.

Inici del projecte (fase de comunicació)

- Definir objectius i abast del projecte i del producte.
- Identificar restriccions.
- Avaluar alternatives per al desenvolupament del producte de programari.

Objectiu del projecte: desenvolupar un producte que assolisca les necessitats identificades amb un temps i cost adequats.

Abast del projecte: identifica fins on s'arriba i que s'inclou dintre de les tasques del projecte i què no.

Objectius del producte: Han de definir-se des del punt de vista operatiu, tàctic i estratègic.

- Objectius estratègics: posicionament en el mercat, imatge compromesa amb la societat, obrir noves línies de negoci, etc.
- Objectius tàctics: millorar la gestió de clients, millorar l'eficàcia i qualitat del treball dels empleats de l'empresa, etc
- Objectius operatius: relacionats amb aspectes de més baix nivell pel que fa als processos de l'empresa.
-

Risc del projecte: tot allò que afecta a la finalització amb èxit del projecte.

Abast del producte: identifica quina funcionalitat ha de cobrir, fins on arribarà, àrees involucrades, les interaccions amb altres entitats u organitzacions o amb altres dispositius o sistemes informatitzats.

Identificar restriccions:

- Poden ser de temps, econòmiques, humanes, d'espai...
- Més habituals en el desenvolupament de productes de programari: costos, temps i personal disponible

*Les restriccions poden fer variar **l'abast del projecte** identificat en un inici*

Gestió de projectes

- **Direcció del projecte:** organitza, coordina i supervisa les activitats, assigna recursos, defineix estàndards de qualitat i de l'empresa per al procés i per al producte. (*Project manager*)
- **Control del projecte:** supervisa la realització correcta del projecte, tenint en compte la planificació inicial per tal de detectar possibles desviacions i corregir-les. (*Project manager i QAM*)
- **Planificació del projecte:** identifica les activitats que s'han de dur a terme en el projecte i estableix un calendari de realització d'aquestes en funció dels recursos i el temps.

Planificació del projecte (fase planificació)

Defineix les necessitats econòmiques, de recursos humans i tecnològics, com també la durada temporal del projecte. Té com a objectiu establir les activitats que s'han de realitzar per tal d'aconseguir els *objectius del projecte*.

- **Definició dels objectius** del projecte. Han de ser realistes, quantificables i factibles.
- **Identificar fases, activitats i tasques.**
- **Estimar** l'esforç hores/persona per tasca/activitat.
- **Establir relacions** entre tasques/activitats.
- **Identificar i assignar els recursos.**
- **Crear el calendari del projecte** (*diagrama de Gantt*)

Estimació del projecte

Realitza una quantificació aproximada de l'esforç o temps d'execució de cada activitat i el cost. No es una ciència exacta.

- **Mètrica:** Unitat de mesura que es defineix basada en les característiques (línies de codi, punts de funció, casos d'ús...)
- **Mesura:** Resultat del mesurament
- **Factor productivitat:** Quantes unitats de les mesurades es poden desenvolupar en una unitat temporal (50 línies de codi per hora)
- **Esforç en hores-persona:** Total d'hores que necessita una persona per desenvolupar una tasca

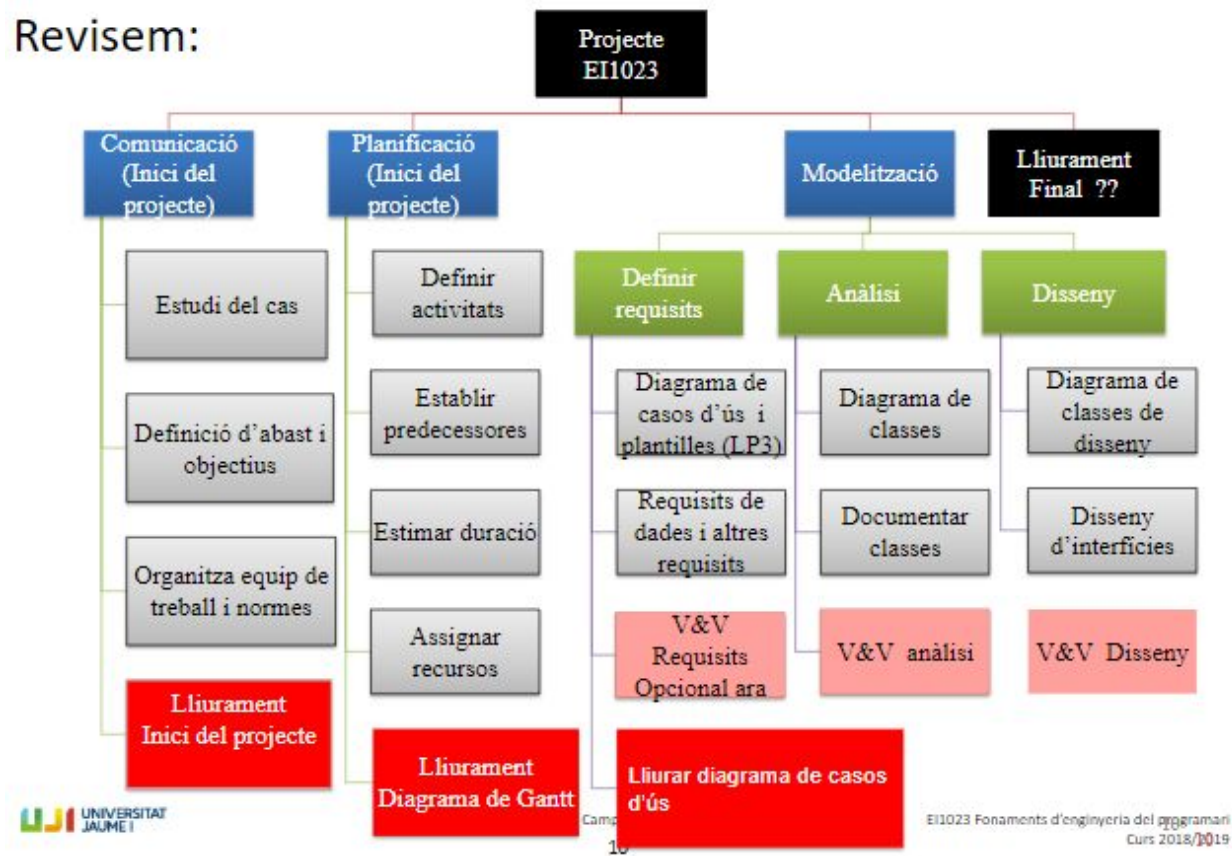
Mètodes

- **Opinió d'experts:** Basada en l'experiència
- **Per analogia:** Compara projectes similars
- **Per descomposició:** Descomposar a un nivell detallat per poder estimar l'esforç de cada subtasca descomposada.
- **Models d'estimació:** Usen equacions i fórmules
- **Punts de funció**
- **Casos d'ús (Karner)**

Tècniques de suport a la planificació

- **Diagrama de descomposició de treball (EDT/WBS):** Visualitza de manera jeràrquica l'estructura de les activitats. El projecte es descompon en fases, les fases en activitats i les activitats en tasques.

Revisem:



- **Diagrama de Gantt:** Permet fer una planificació temporal i generar un calendari amb activitats, precedències, temps i recursos.
 - **Indicar data d'inici**
 - **Organitzar esquema lògic d'activitats**
 - **Estimar durada**
 - **Identificar predecessors**
 - **Assignar recursos**

Tipus d'activitats:

- **Predecessors:** Han de finalitzar per començar altres
- **Fites:** No es pot seguir si aquestes no es duen a terme (lliuraments)
- **Tasques de resum:** Agrupen altres tasques

TEMA 3

(Fase de modelització)

Requisit: condició que el programari ha de complir o una característica que ha de tenir per tal d'assolir els objectius i abast definits, tenint en compte les restriccions. Han de:

- Estar dintre de l'abast funcional
- Ser viables tecnològic, econòmic i temporalment
- Contemplar tots els objectius i necessitats funcionals definides

Restriccions: limitacions físiques, econòmiques, organitzatives, temporals o tecnològiques que condicionen la implementació del programari, el seu abast i objectius.

ETAPES DE LA DEFINICIÓ DE REQUISITS

Investigar els requisits del sistema

S'utilitzen tècniques apropiades de recerca per a investigar, buscar i recollir la informació, es fan reunions i es recullen idees que permetran determinar els requisits del sistema proposat.

- **Revisar el sistema en funcionament:** Estudiar el sistema en funcionament proporciona una primera aproximació al coneixement dels processos de negoci dintre de l'abast del projecte. També permet identificar quins punts forts o febles té el sistema actual.
 - Recopilar documents existents
Permet identificar dades d'entrada i d'eixida del sistema que poden formar part dels requisits.
 - Observar el funcionament del sistema
Tècnica habitual i imprescindible quan es desenvolupa programari de control de producció o suport a processos industrials. És útil per identificar algunes necessitats d'un sistema d'informació.
Les observacions s'han de fer en moments d'activitat normal i en moments d'alta activitat o de realització de processos especials, per a detectar factors que poden influir en el bon funcionament del sistema.

- **Tècniques d'investigació de requisits**

- **Entrevistes:** Tenen com avantatge el contacte personal amb els futurs usuaris, que proporcionen més informació i de més valor que altres mètodes. Desavantatge: consumeix molt més temps que altres mètodes.
- **Qüestionaris:** Més impersonals que les entrevistes però més objectius.
- **Fonts externes:** Quan el sistema en funcionament té molts problemes, es vol realitzar un canvi total o bé el sistema s'implanta en una nova organització pot ser convenient utilitzar fonts externes.
- **Desenvolupament conjunt d'aplicacions:** Promou la cooperació i el treball en equip com a alternativa a les entrevistes individuals. Consisteix en organitzar i realitzar un conjunt de reunions de treball en les quals participen usuaris experimentats i enginyers informàtics.

Documentació dels requisits del sistema

Serveix per a validar el programari i dur a terme l'acceptació del producte, és a dir, comparar els resultats obtinguts amb les especificacions del que es volia obtenir. Per definir i documentar els requisits es necessita el treball conjunt dels clients/usuaris i de l'enginyer informàtic.

- **Condicions que ha de complir el document:**

- Reflectir únicament els comportaments externs del sistema, és a dir, la visió que els futurs usuaris tindrà del programari i del maquinari
- Fàcilment modificable
- Servir com a referència quan es realitzen manteniments del sistema
- Present durant tot el cicle de vida del programari
- Servir com a contracte per establir el que s'espera del producte que es vol contruir
- Incloure una taula de continguts, glossari de termes utilitzats i una possible previsió de canvis en els requisits inicialment definits
- No es un document de disseny, ha de mostrar què ha de fer el sistema sense especificar la forma o el suport en què es realitza.
- Requisits: condicions precises, sense ambigüitats ni contradiccions

- **Tècnica Diagrama de casos d'us (UML)**

Utilitzem els DCU per a representar la funcionalitat del sistema informàtic i poder documentar i analitzar els requisits del sistema. L'objectiu és documentar el comportament del sistema informàtic desde del punt de vista de l'usuari.

- Elements dels diagrames de casos d'ús
 - ❑ **Actors:** Un actor representa un rol que pot assumir un grup de persones, una persona o dispositius físics externs al sistema, que interactuen amb el producte de programari (persones que interactuen amb el sistema).
 - ❑ **Casos d'ús:** Mostra el que ha de fer el programari des del punt de vista dels usuaris, independentment de la seua implementació tecnològica (seqüència d'accions).
 - ❑ **Relacions:** Representen la connexió que existeix entre l'actor i el cas d'ús.
- Pasos per a fer el diagrama de casos d'ús
 - Identificar els actors i pensar el paper que representen.
 - Identificar els missatges necessaris entre els actors i el sistema informàtic a desenvolupar.
 - Identificar la funcionalitat que el sistema proporcionarà des del punt de vista del usuari i definir els casos d'ús.
 - Dibuixar el diagrama.
 - Descriure els casos d'ús i els actors.
 - Validar amb els usuaris competents.

Verificar i validar requisits

- **Verificar:** Comprovar que la definició de requisits s'ha fet de manera correcta, és a dir, que se segueixen els estàndards de qualitat.
 - Comprovar que cada requisit:
 - Es consistent amb els objectius i està dintre de l'abast del projecte
 - Es necessari
 - S'ha especificat a un nivell d'abstracció adequat
 - Està definit sense ambigüetat i es comprensible
 - No entra en conflicte amb altres requisits
 - Es pot comprovar una vegada implementat
 - La documentació està desenvolupada de manera adequada
- **Validar:** Comprovar que el document de requisits i els models desenvolupats reflecteixen el que els usuaris volen que el producte de programari duga a terme. Es comprovarà que cada requisit, cas d'ús i necessitat documentada correspon a una necessitat detectada pels usuaris i que cada necessitat detectada pels usuaris està registrada de manera correcta en el document de requisits.