Aantekeningen 11-09-2018

Functioneel Ontwerp

In deze fase specificeert het ontwerp, de Wat vraag (Wat moet het systeem gaan doen?)

Eindgebruiker en opdrachtgever specificieren dat. (Nooit aannemen)  
  
SDLC (System Development Lifecycle), zeer belangrijk in studie. (Software Development Lifecycle dekt niet de hele lading)  
  
Het idee begint vooral door vraag naar verbetering, of realizatie dat iets helpt voor iemand.  
  
Haalbaarheidsstudie: Het idee komt binnen, en wordt bestudeert dat het technisch haalbaar is, of het economisch haalbaar (is het rendabel), en of er alternatieven zijn? (Bv: bestaande process aanpassen, of gelijk product, outsources, etc)

Analysefase: Interviews, enquetes met gebruikers and projectgever en documenten onderzoek om informatie te verkrijgen om de specificaties van het ontwerp   
  
Functioneel Ontwerp:  
Lijst van specifcaties worden gebruikt om het ontwerp gemaakt  
  
Technisch Ontwerp:  
Hoe het systeem moet functioneren (Welke server, code, etc.)  
  
Functioneel Ontwerp + Technisch ontwerp = System Design  
  
Bouwfase (realizatiefase):  
Bouwen (coderen) van het ontwerp, daarnaast ook documenteren en handleidingen schrijven voor de eindgebruiker (Documentatie is zeer belangrijk)  
Zonder documentatie is het nodig om te reverse engineeren, wat zeer tijdrovend is.  
Verder test je het te bouwen systeem ook.  
  
Implementeren:  
fysiek op een server zetten (met code), en de eindgebruikers introduceren (cursussen, etc), plus vragen of dit acceptabel voor de eindgebruiker is (Gebruikersacceptatietest)  
Hierna wordt het overgedragen aan het systeem(of applicatie) beheer  
  
Onderhouden:  
Het beheer onderhoudt het systeem en houdt het up-to-date. Verder kunnen gebruikers ook vragen voor extra functies.  
  
Kill/Uitfaseren:  
Als het applicatie/systeem niet meer nodig is wordt deze uitgefaseerd, en bepalend op de aanleiding hiervan begint de cyclus weer (Bv. Betere systeem cyclus gaat door verandering behoeften bedrijf, cyclus stops)  
  
Tijdens de eerste twee fases maak je een plan van aanpak.  
  
De eindatum van een project is wanner de implementatie gedaan is.

Leren van cyclus: Uitleggen wat er in en uit de fase komt, en wat er gebeurt  
  
Wie is verantwoorderlijk voor uitvoeren?  
1 Plan: Het Idee, intiate  
2 Ontwerpen: Alles tot de implementatie, door de ontwikkelende organizatie  
3 Onderhouden: Onderhouden door de ondersteunende organizatie (beheerders)   
  
Informatieanalyst stelt een vooronderzoek in of informatieplan.  
  
Informatieplan : Welke informatie heb je echt nodig  
  
Functioneel ontwerp: Wat moet er gedaan worden op basis van informatieplan  
  
Technisch ontwerp: Hoe moet het gedaan worden  
  
Waarom FO: omdat je geen huis bouwt zonder ontwerp

**Functioneel Ontwerp**

Inleiding: beschrijving bedrijf, doel van ontwerp  
  
Beschrijving nieuwe infosysteem, korte, duidelijke uitleg van het infosysteemGegevensmodel, het database  
De gewenste uitvoer: Wat moet er uit de database komen?  
De gewenste invoer: Wat moet er ingevoerd worden?  
De benodigde invoer: Welke gegevens zijn er nodig om de uitvoer mogelijk te maken  
Menustructuur en autorisatie: Wie mag bij de gegevens gebruiken, en hoe.  
Organisatorische consequenties: welke consequenties zijn er voor de organisatie, de gebruiker, etc.  
Technische consequenties: Wat is er nodig om het systeem te laten functioneren (server, extra kabels, etc.)  
  
IS (informatiesysteem)  
Invoer van data word geprocessed naar informatie en dat wordt geoutput naar informatie producten

Algemene aanpak voor het maken van FO:

1. Begin met het lezen van de aanwezige documentatie
2. Daarna houdt interviews met opdrachtgever en gebruiker.
3. Daarna beschrijf gewenste uitvoer.
4. Daarna her gegevensmodel.
5. Daarna de invoerfuncties
6. Menustructuur en autorisatie
7. Technische en organisatorische consequenties (kunnen verdeeld worden, zijn niet afhankelijk van elkaar)
8. Beschrijving IS
9. Inleiding

Een entiteit is een object, person, gebeurtenis of plaats.  
Een attribuut is alles wat je over een entiteit kan/wilt vast leggen.  
  
Primaire sleutel: unieke kenmerk van een entiteit, bv. Klantnummer, studentennummer.  
Attributen zijn uit datatypen en er wordt plaats gemaakt wordt voor dit datatype in het database. Dit wordt gedaan om sneller dingen te laten draaien.  
Bv: Klantennummer, Numeriek, 9(6) (Dus, 9\*6, of 999999 verschillende klantennummers)

Relaties  
Entiteiten worden gekoppeld bij een andere entiteit.  
Regel 1: De primaire sleutel van 1-side van een relatie kan gevonden worden als attribuut in de entiteit van de n-side  
Regel2: Veel op veel relaties mogen niet omdat de database anders niet weet wat uniek is. (Een klant kan meerdere fracturen hebben, maar verschillende klanten kunnen niet dezelfde fractuur hebben) Om dit te voorkomen wordt er vaak een tussenstop gemaakt.  
  
Entiteit relatie diagram (ERD) laat zien hoe entiteiten zijn gekoppeld.  
  
Kraaienpoot in een diagram betekent veel, gewone streep = 1  
  
Definitie van een entiteit wordt gedaan in een bepaald format (staat op Blackboard)  
  
Ontwerk richtlijnen voor uitvoerproduct:

* Goede titel
* Datum en tijd
* Gebruik secties en rubrieken om info te ordenen
* Duidelijke labels voor alle velden en kolommen
* Inclusief legenda voor afkortingen
* Alleen benodigde info
* Rapporteer info in her formaat dat niet handmatig bewerkt moet worden
* Info evenwichtig verdelen over pagina, scherm

Ontwerp process uitvoerproduct

1. Identificeer uitvoerproduct en bekijk logische eisen
2. Specificeer de fysieke output eisen
3. Indien nodig ontwerp ieder voorgedrukt formulier
4. Ontwerp, valideer en test outputs door gebruikt to maken van layout tools

Uitvoerproduct bestaat uit:

* Code
* Naam overzicht
* Gebruiker
* Doel
* Frequentie (tijd)
* Sortering
* Selectie
* Af te drukken
* Opmerkingen

Infoschets van uitvoerproduct wordt gemaakt zodat de gebruiker het begrijpt en het bijgestuurd kan worden

Invoertak

* Code
* Naam
* Autorisatie (Wie kan het invoeren?)
* Doel
* Beschrijving
* Frequentie (tijd)
* Bestanden
* Schermen (Welke schermen mogen dit worden ingevoerd?)

Geen schets nodig.

Beschrijf gedtailleerd wat en hoe wordt getest  
  
Gebruikersacceptatie: hoeveel % gebruikers accepteert het nieuwer systeem

Conversie: hoe wordt oude data geconventeerd

Trainingen en opleiding: moeten werkers worden omgeschoold om te kunnen werken?

Verandering in organisatie