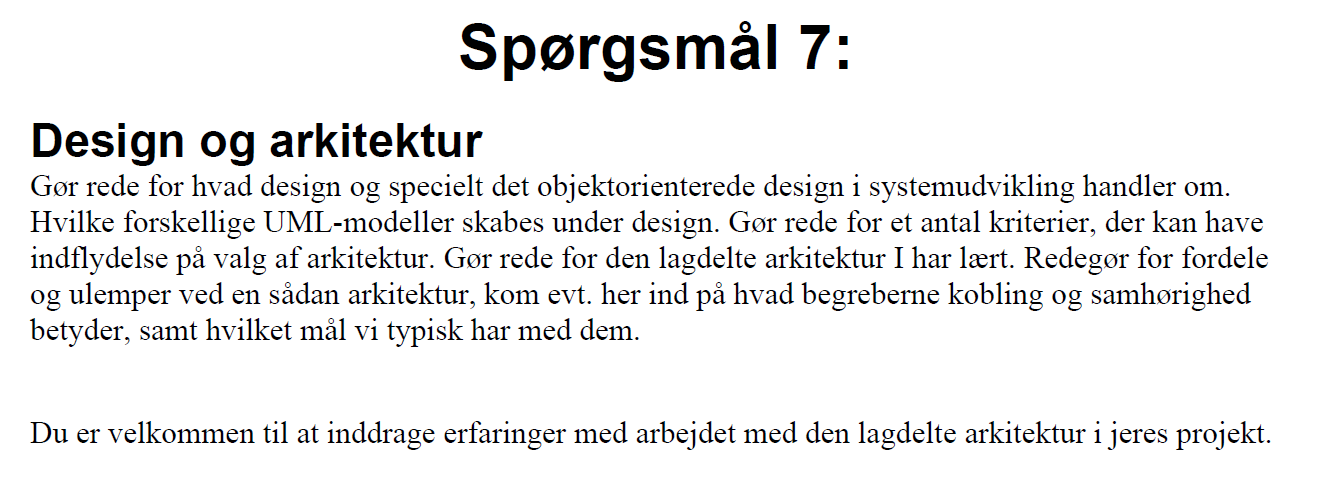
Spørgsmål 7



**Design og arkitektur**

● **Hvad handler design om**

Design handler om at få udvidet analysen, så man får en ”opskrift” på hvordan et system skal programmeres. Når vi går ind i designfasen, får vi mere fokus på, hvordan vi kan realisere modellen i forhold til designprincipper. Her skal der gøres nogle overvejelser om hvordan klasserne skal hænge sammen. For at kunne træffe nogle gode valg om logikken i systemet skal man have nogle kriterier at designe efter. Disse kriterier nedenfor vurderes i et skema med henholdsvis “Meget vigtig”, “Vigtig”, “Mindre vigtig” eller “Irrelevant”.

|  |  |
| --- | --- |
| **Design** | **Mål for** |
| **kriterier** |  |
|  |  |
| Brugbart | Tilpasningen af systemet til de organisatoriske, arbejdsmæssige og tekniske |
|  | rammer |
|  |  |
| Sikkert | Sikringen mod uønsket adgang til systemets data og facilitete |
|  |  |
| Effektivt | Udnyttelsen af faciliteterne i den tekniske platform |
|  |  |
| Korrekt | Opfyldelsen af de opstillede krav |
|  |  |
| Pålideligt | Opfyldelsen af den krævede funktionalitet med den ønskede præcision |
|  |  |
| Vedligeholdbart | Omkostningerne ved lokalisering og retning af fejl i det kørende system |
|  |  |
| Testbart | Omkostningen ved test af systemet i forhold til de opstillede krav |
|  |  |
| Fleksibelt | Omkostningen ved at ændre i det kørende system |
|  |  |
| Forståeligt | Besværet for udvikleren ved at skaffe sig overblik over og forstå systemet |
|  |  |
| Genbrugbart | Anvendeligheden af dele af systemet i andre beslægtede systemer |
|  |  |
| Flytbart | Omkostningen ved at flytte systemet til andre tekniske platforme |
|  |  |
| Integrerbart | Problemerne ved at sammenkoble systemet med andre systemer |
|  |  |

● **Hvilken betydning har arkitekturen**

Arkitekturen har en betydning på, hvordan systemet skal opbygges. Her skal besluttes hvor metoderne skal ligge, retninger på associeringer og aggregeringer, om nogle aggregeringer skal ophøjes til kompositioner. Der skal vælges arkitektur efter de valgte kriterier og prioriteringer.

fordele ved 3 lags modellen er at,

1. Det minimerer afhængighederne mellem lagene 2. Et lag kan bruges af flere overliggende lag

ulemper

1. Ekstra lag kan forringe performance

2. Ekstra lag kan give unødvendig doubletarbejde

● **Eksempel på en arkitektur I kender, og hvordan I har brugt den**

En eksempel på arkitektur er 3 lags modellen, som er den eneste arkitektur vi har brugt. 3 lags modellen går ud på at, man ikke kan bruge begreber, klasser eller metoder i lag over sig, men kun dem i samme lag eller under.

3 lags modellen består at 3 lag gui, application og storage.

Vi har brugt det i f.eks. Aarhus bryghus. Gui er vores brugergrænseflade, som kan bruge metoder fra application og gemme data i storage.

Application laget består af en controller som typisk har metoder som typisk involverer flere klasser, og opretter eller ændre en del af modellens objekter.

Model pakken indholder model klasserne med associeringer og andre forbindelser.

Storage laget indeholder metoder til at gemme objekter af klasser. Laget indeholder metoder til at hente, opdatere og slette objekter.

● **Sammenhæng og forskel mellem analyse og design, indhold af design**

I analysefasen laver man et overblik i form af analyseklassediagrammer. Når man skal begynde og lave sit system er det nødvendigt at lave et designklassediagram. Her får man typer på attributter (String, int osv.), metoder og sammenhængene mellem klasser sat på.

● **Designklassemodellen**

○ Tilføjelser og justering i forhold til analyseklassemodellen

Man tilføjer typer på attributter, sammenhænge og metoder