

Et program til statistik for en fodboldklub i Premier League

En fodboldklub i Premier League, der hedder [vælg selv hvilken her], er interesseret i et nyt system til at registrere informationer om deres kampe. Man håber at den nye applikation ved hjælp af den registrerede data kan medvirke til at træffe bedre beslutninger om en række forhold.

Applikationen skal kunne registrere:

1. Hvem der spiller på holdet og deres position på banen.
2. Hvor meget klubben giver i løn til hver spiller.
3. Hvornår klubben har spillet og hvornår de skal spille og mod hvem.
4. Hvis en kamp er spillet, vil man kunne se resultatet, og hvem der har scoret hvornår.
5. Hvis en kamp er spillet, hvilke spilsystem, som klubben har spillet.
6. Hvor mange redninger indenfor målrammen, som målmanden i den givne kamp har foretaget.
7. Hvor mange kampe hver spiller har spillet.
8. For hver spiller, hvor mange kampe der er vundet, og hvor mange der er tabt ud af de kampe, som spilleren har været med i.

Nogle af de spørgsmål, man håber at få svar på ved brug af det nye program er:

- Findes der et optimalt lønniveau for den enkelte spiller? Man har fundet frem til at det for nogle fodboldspillere gælder, at hvis de tjener meget, er de mindre tilbøjelige til at yde deres bedste, simpelthen fordi de mister motivation. Man håber derfor med det nye program, at man ved at registrere hvor mange mål angriberne scorer i forhold til hvor meget de tjener, og hvor mange mål som målmanden har reddet, ville kunne identificere et optimalt lønniveau, hvor lønnen motiverer spillerne mest muligt.
- Hvem er det, der har den generelt mest positive effekt på holdet? Man har den teori, at nogle spillere engang imellem har en positiv effekt på holdet som helhed, selvom de ikke scorer mål. Men hvem drejer det sig om over en længere periode? Tidligere har man især fokuseret på den effekt anføren har på holdet som helhed, men dermed har man måske overset at andre spillere har en større positiv effekt. Man vil gerne forstærke den positive effekt – positive reinforcement som det hedder – ved at gøre den helt rigtige spiller til anfører på holdet. Derfor er man interesseret i en top tre liste over de spillere, som har deltaget i procentvist flest vundne kampe, for teorien er at det er dem der alt andet lige har den mest positive effekt på holdet som helhed.
- Hvad er det optimale spilsystem mod hvem? Ved at se på hvilke kampe der er vundet eller tabt mod hvilke klubber, vil man kunne træffe beslutninger om hvilke spilsystemer man skal vælge til hvilke modstandere.

Krav til Software Design

Det forventes, at centrale dele af casen er modelleret og dokumenteret i form af henholdsvis en Use case model, en Domænemodel og en Designmodel.

Det er et krav, at der gøres brug af:

- Supplerende kravliste
- Use case diagram og tilhørende use case beskrivelser (mindst 6 use cases skal være beskrevet, heraf 1 i fully dressed format)
- Domænemodel, der beskriver de konceptuelle klasser
- Systemsekvensdiagrammer (SSD) for mindst 4 af de beskrevne use cases
- Kommunikationsdiagrammer *eller* sekvensdiagrammer for de 4 use cases, der er modelleret i SSD
- Designklassediagram

Der skal desuden foreligge en:

- Faseplan
- Papir-prototype

Krav til Introduktion til Objekt Orienteret Programmering

Hver gruppe skal programmere en applikation, hvor der gives en implementation af mindst 4 use cases beskrevet i designdelen af rapporten.

Programmet skal demonstrere brugen af følgende:

- Datastrukturer såsom arrays eller ArrayList
- Arv
- Input fra brugeren igennem konsollen
- Persistens ved at gemme data til og hente data fra filsystemet
- Pakker
- Arkitektur, foreksempel MVC.

Rapporten skal indeholde

- En grundig beskrivelse af den udviklede applikation og en begrundet forklaring af de valg, som der er truffet undervejs, herunder gerne fordele og ulemper ved de forskellige mulige løsninger som har været overvejet.
- En beskrivelse af de særligt interessante dele af koden.
- Hvis der er dele af systemet, som ikke fungerer korrekt skal disse beskrives. Beskriv problemet som I ikke har kunnet løse og evt. hvor langt I er nået mht. at løse problemet.
- En beskrivelse af hvordan systemet afprøves, hvis det ikke er indlysende for

en førstegangsbriker.

Der skal være en tydelig sammenhæng mellem software design og programmering dvs. kodningen skal tage udgangspunkt i de i software design, udviklede modeller.

Den udviklede kode skal være velkommenteret undtagen for de trivielle dele.

Hvert gruppemedlem skal være ansvarlig for programmeringen af en del af applikationen og udvælges af gruppen i fællesskab ud fra kompleksiteten og den enkelte studerendes kvalifikationer. Det er tilladt at programmere centrale/komplekse dele af systemet i fællesskab.

Det skal angives i rapporten og i kommentarer i koden hvilke dele af programmet hver enkelt studerende har haft hovedansvaret for at programmere.

Krav til rapporten og grupperne

Hver gruppe skal bestå af mindst 4 og max 5 personer. De studerende sammensætter såvidt muligt selv grupperne. Det anbefales stærkt at skrive en gruppekontrakt med mål og forventningsafstemning i forhold til arbejdet. Hver gruppe skal lave en projektrapport à max 30 normalsider, der dokumenterer applikationens udvikling og implementation. Rapporten afleveres samt kildekoden til projektet afleveres onsdag den 9. december kl 12.00.

Hver gruppe afleverer én og kun én .zip fil, der indeholder selve rapporten i pdf format samt kildekoden til projektet. Zipfilen uploades som gruppeaflevering på Fronter, hvor der er en afleveringsmappe i Projekt mappen for 1. semester. Vær opmærksom på at mappen lukker ved deadline og at deadline ikke kan rykkes. Hvis en gruppe ikke har afleveret ved deadline vil samtlige gruppemedlemmer dumpe prøveeksamen.

Underviserne står til rådighed i de respektive timer med vejledning under projektperioden. Underviserne læser ikke rapporten eller dele heraf før rapporten afleveres.

Uddrag af formkrav taget fra Datamatiker studieordningens beskrivelse af førsteårsprøve. Vejledende for prøveeksamen på første semester:

- Der skal afleveres en projektrapport og et produkt.
- Projektrapporten, som udgør den skriftlige del af prøven skal minimum indeholde:
 - Forside med titel, navn og fødselsdato
 - Holdbetegnelse og dato.
 - Indholdsfortegnelse
 - Indledning, inkl. problemformulering
 - Hovedafsnit

- Konklusion
 - Litteraturliste (inkl. alle kilder, der er lavet henvisninger til i projektet)
 - Bilag (inkluder kun bilag, som er centrale for rapporten)
 - Vedlæg kildekode eller angivelse af sti til versionsstyringsserver, hvor kildekoden og eksekverbar kode til produktet kan hentes
 - Der skal pagineres (sidetal) på alle sider
- En normalside er 2.400 tegn inkl. mellemrum og fodnoter.
 - Forside, indholdsfortegnelse, litteraturliste samt bilag tæller ikke med heri.
 - Bilag er uden for bedømmelse.
 - Hver enkelt figur eller diagram tæller 800 tegn.