

1. Semesterprojekt Datamatikeruddannelsen, 2. Obligatoriske opgave, efterår 2017

Case: Svømmeklubben Delfinen

Svømmeklubben Delfinen er en mindre klub, der er i vækst. Klubbens ledelse ønsker derfor udviklet et administrativt system til at styre medlemsoplysninger, kontingenter og svømmeresultater.

Det er klubbens formand, der tager sig af nye medlemmer. Ved indmeldelse i klubben registreres diverse stamoplysninger om personen herunder alder.

Desuden registreres oplysninger om personens ønskede aktivitetsform, det vil sige aktivt eller passivt medlemskab, junior eller senior svømmer, motionist eller konkurrencesvømmer.

Klubbens kasserer tager sig af alt vedrørende kontingentbetaling. Kontingentets størrelse er betinget af flere forhold.

For aktive medlemmer er kontingentet for ungdomssvømmere (under 18 år) 1000 kr. årligt, for seniorsvømmere (18 år og over) 1600 kr. årligt. For medlemmer over 60 år gives der 25 % rabat af senior taksten. For passivt medlemskab er taksten 500 kr. årligt.

Kassereren har ønsket, at systemet kan vise en oversigt over medlemmer, der er i restance.

Konkurrencesvømmerne har tilknyttet en træner. Konkurrencesvømmerne er inddelt i 2 hold efter alder. Ungdomsholdet er for svømmere under 18 år. Seniorholdet er for svømmere på 18 og over. Hver konkurrencesvømmer er desuden registreret i forhold til hvilke svømmediscipliner, han er aktiv i.

Inden for hver svømmedisciplin registreres den enkelte svømmers bedste træningsresultat og dato løbende. For de svømmere, der har deltaget i konkurrencer, registreres stævne, placering og tid. Det er på baggrund af de enkelte svømmeres resultater, at træneren udtager svømmere til deltagelse i konkurrencer. Træneren ønsker derfor en oversigt, der kan vise klubbens top 5 svømmere inden for hver svømmedisciplin (butterfly, crawl, rygcrawl, brystsvømning og hundesvømning).

Projektkrav

Krav til ITO

Med udgangspunkt i casen om Svømmeklubben Delfinen skal der udarbejdes følgende analyser:

1. En interessentanalyse inklusiv en power/interest-matrix
2. En SWOT-analyse

Krav til Softwaredesign

Systemudviklingen skal tage udgangspunkt i Craig Larman "Applying UML and Patterns".

Det forventes, at centrale dele af casen er modelleret og dokumenteret i form af henholdsvis en use case model, en domænemodel og en designmodel (design klassediagram).

Det er et krav, at der gøres brug af:

- FURPS, herunder en liste med krav. Alternativ lav en liste med funktionelle og non-funktionelle krav.
- Liste med forretningsmæssige begreber (glossary)
- Use case diagram og tilhørende use case beskrivelser (mindst 3 fully dressed, overvej inclusions)
- Mindst 3 use cases, i brief format
- Systemsekvensdiagrammer (for tre fully dressed use cases)
- Domain model (diagram)
- Sekvensdiagrammer (1 stk)
- Designklassediagram (DCD)

Der skal desuden foreligge en:

- Afgrænsning (hvilken funktionalitet er med i jeres projekt)
- Faseplan (laves I starten af jeres projekt hvor i beskriver de forventede aktiviteter, forløb) . Brug en modificeres vandfaldsmodel, og brug KanBan til afviklingen af opgaver i Trello.
- Vision

Det gælder for alle diagrammer at de ikke behøver at være færdige, men skal afspejle jeres projekt.

I er velkomne til at beskrive jeres tanker omkring brug af modellering i konstruktionen af jeres kode og om hvordan i sikrede jer overensstemmelse mellem jeres modellering og kode.

I tilfælde af uklarheder omkring oplægget skal projektgruppen, evt. efter samtale med lærerne, beslutte sig for en passende fortolkning, som beskrives i rapporten.

Krav til Softwarekonstruktion

I skal programmere jeres design eller dele af jeres design på en sådan måde at det klart fremgår at koden er baseret på designet og lever op til den generelle opgavebeskrivelse.

Koden skal som minimum indeholde:

- Flere klasser, indeholdende indkapsling.
- Datastrukturer, såsom Array/ArrayList.
- Nedarvning.
- Filer på harddisk, således at data bevares/hentes.
- Konsol UI, formateret pænt.

Og I skal demonstrere en generel evne til at programmere basale algoritmer.

Rapporten skal som minimum indeholde:

- Afgrænsning af SWC delen af projektet.
- Et opdateret klassesdiagram.
- En grundig beskrivelse af jeres konstruktion af produktet og en velbegrundet forklaringer på de valg I har truffet under jeres udvikling.
- En beskrivelse af hvad der virker og hvad der ikke virker, og hvorfor.

Der skal være sammenhæng mellem jeres software design og software konstruktion.

Krav til Rapport og programdokumentation

Der skal udarbejdes en projektrapport (max 30 sider), der dokumenterer systemet. Rapporten og programmet samles i en afleverings fil i pdf-format i PeerGrade. I rapporten skal der være link til koden på GitHub eller BitBucket. **Gruppemedlemmernes navne skal stå på forsiden af rapporten.**

Projektet er defineret i forhold til bekendtgørelsens krav om obligatoriske individuelle afleveringsopgaver. Dette indebærer, at det i rapporten og i kildekoden skal være angivet, hvem der er ansvarlig for hvilke produkter (diverse diagrammer og programmoduler).

Projektet bedømmes bestået eller ikke bestået. Opgaven godkendes, hvis hver studerende i gruppen fremlægger en del af arbejdets resultater- samt kan svare på opklarende spørgsmål på en tilfredsstillende måde, der viser, at han/hun har deltaget i gruppens arbejde.

Projektforløb og vejledning

Projektstart tirsdag den 14. november 2017

Projektrapport afleveres på PeerGrade senest **torsdag den 7. december klokken 16.00**. Vær opmærksom på at PeerGrade lukker kl. 16.00. Deadline kan ikke udskydes og aflevering kan kun ske via PeerGrade.

Hver gruppe afleverer **en** pdf-fil som indeholder rapporten i pdf format. I pdf-filen angives link til kode på GitHub.

Der gives vejledning i den angivne projektperiode. Vejledningen finder sted i forhold til det gældende skema og aftaler med vejlederne. Vejlederne læser ikke dele af rapporten før afleveringen.

Projektgrupper

Projektet løses i grupper, der er sammensat med hjælp af lærerne.

En oversigt over gruppernes medlemmer kan ses på fronter.

Bilag 1: Orientering om krav til og udformning af opgaver på datamatiker uddannelsen

Definitioner

1.1 Normalside

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.

2 Generelle krav

I dette afsnit beskrives de generelle krav til en eksamensopgave som afleveres på datamatikerstudiet. Overholder du ikke disse krav, kan din opgave blive afvist ved aflevering.

Formkrav til det skriftlige projekt

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, hold betegnelse og dato
- Indholdsfortegnelse
- Indledning, inkl. problemformulering
- Hovedafsnit (Analyse, Design, Konstruktion)
- 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 (GitHub/BitBucket), hvor kildekoden og eksekverbar kode til produktet kan hentes
- Der skal pagineres (sidetal) på alle sider

2.1 Underskrift

Hvert enkelt gruppemedlem skal underskrive opgaven

2.2 Uddannelse

Uddannelsens navn skal angives.

2.3 Skolen

Skolens navn skal angives.

2.4 Eksamensprofil skal angives

Eksamensprofil skal angives, for eksempel "Anden obligatoriske opgave på 1. semester"

2.5 Format

Opgaven (rapporten) skal afleveres på PeerGrade som en PDF-fil samt link til kode på GitHub.