

DELFINEN

Gr. 1: Rasmus Kibshede, Kamille Nikolajsen og Michala Nybroe



27. MAJ 2021

KEA DAT21V1

Indholdsfortegnelse

Indledning:.....	2
Interessentanalyse:.....	2
Risikoanalyse:.....	4
Domæne model:.....	5
Use case model og beskrivelse:	6
Manage Members	6
Expected Income	6
Manage Member Debt	7
Manage Competitors	7
Design klassediagram:	8
Member administration	9
Economy administration.....	9
Competitive administration	9
Samlede design klasse diagram	10
Afgrænsning for proces	10
Status af kode	11
GitHub	11

Indledning:

Dette projekt stilles af klubben Delfinen, der ønsker et administrativt system til at navigere en forøgende klubs størrelse. Systemet giver adgang til håndtering samt overblik af de forskellige medlemstyper og økonomien bag klubben.

Interessentanalyse:

I det følgende fremvises dette projekts interessentanalyse. Her tages udgangspunkt i projektet og ikke virksomheden Delfinen. Denne blev lavet ved projektets begyndelse.

Interessentanalyse						
Interessenter	Deres mål	Tidlige reaktion	Hvad der kan forventes	Indvirkning pos./neg.	Mulig fremtidig reaktion	Ideer
Delfinen	At få et godt administrations program.	Spændte	Bliver nervøse, hvis ikke de ser en klar linje og gode løbende resultater.	Negativ, hvis mål ikke nås. Positiv, når deres system kan afhjælpe den voksende klub.	Kan nægte at betale, hvis ikke kravene imødegås.	Sørge for at give præsentation er løbende – dels for at sikre en fælles forståelse af mål og behov samt til at skabe tryghed om udviklingsprocessen.
Undervisere	At se eleverne få et godt læringsudbytte.	Tillidsfulde og spændte.	Rolige og hjælpende/vejledende.	Negativ hvis ikke fulde læringspotentiale opnås. Positiv idet projektet har givet stort læringsafkast for eleverne.	Review og hermed give brugbar feedback for en videre udvikling og sparring til denne type projekter.	Sørge for løbende at henvende sig med konkrete problemstillinger eller diagrammer og hermed få brugbar feedback.

Produktionschef – Michala	At planerne overholdes og processen understøtter gruppens udvikling.	Spændt	Rammesætning og håndtering af de ændringer som hænder undervejs. Et overblik over projektets fremgang.	Negativ ved stress som for meget pres på rammesætning eller stagnering. Positiv ved adrenalin og udvikling.	Kan blive nødt til at indlægge flere pauser, hvis gruppemedlemmerne bliver for udkørte.	Sørge for plads til at produktionschefen også kan dele sine tanker løbende med holdet.
Kommunikationschef – Kamille	At projektet forløber gnidningsfrit og give plads til den enkelte medlemmer således rammerne er bedst mulige for udvikling.	Ekstrem positivitet og blik for de enkelte medlemmers helbred.	En positiv tilgang der opmuntrer og støtter op om gruppens læringsgrundlag. Skaber et rum, hvor ideer kan få lov at florere således læringsudbyttet bliver hævet.	Negativ, hvis tiltag affejes eller fokus forsvinder for helbred. Positiv hvis gruppen har et godt forløb og er tilfredse i sidste ende.	Kan blive nødt til at uddelegere arbejde, hvis presset bliver for tungt. Kan finde på at sige stop, hvis forholdene ikke er optimale.	Holde pauser løbende gennem processen og skabe et rum samt tid til at dele meninger og tanker.
Teknikchef – Rasmus	At skabe en glidende arbejdsproces ved at minimere tekniske komplikationer og desuden være bevidst om ressourcebalance.	Spændt og ambitiøs	Sikre at designet er gennemtænkt inden produktionsfasen går i gang. Desuden minder gruppen om brugen af Git.	Negativ, hvis design formår at blive for komplekst eller hvis tidsplanen skrider grundet design på bekostning af overblik. Positiv sikre overblik over programmet	Kan finde på at hive design tilbage, hvis fokus erstattes med usikkerhed. Vil øge opmærksomheden på brugen af rammer, hvis for mange ressourcer går til deling af kode eller andre tekniske komplikationer.	At lave en god start med enighed om design og herefter løbende tjekke ind for at se hvorvidt behovet for ændringer indfinder sig. Give plads til fællesvisioner og læring som processen skrider fremad.

				et og gør det let at udføre projektet, hvilket er med til at sikre læringsudbytte og tilfredshed i sidste ende.		
--	--	--	--	---	--	--

Risikoanalyse:

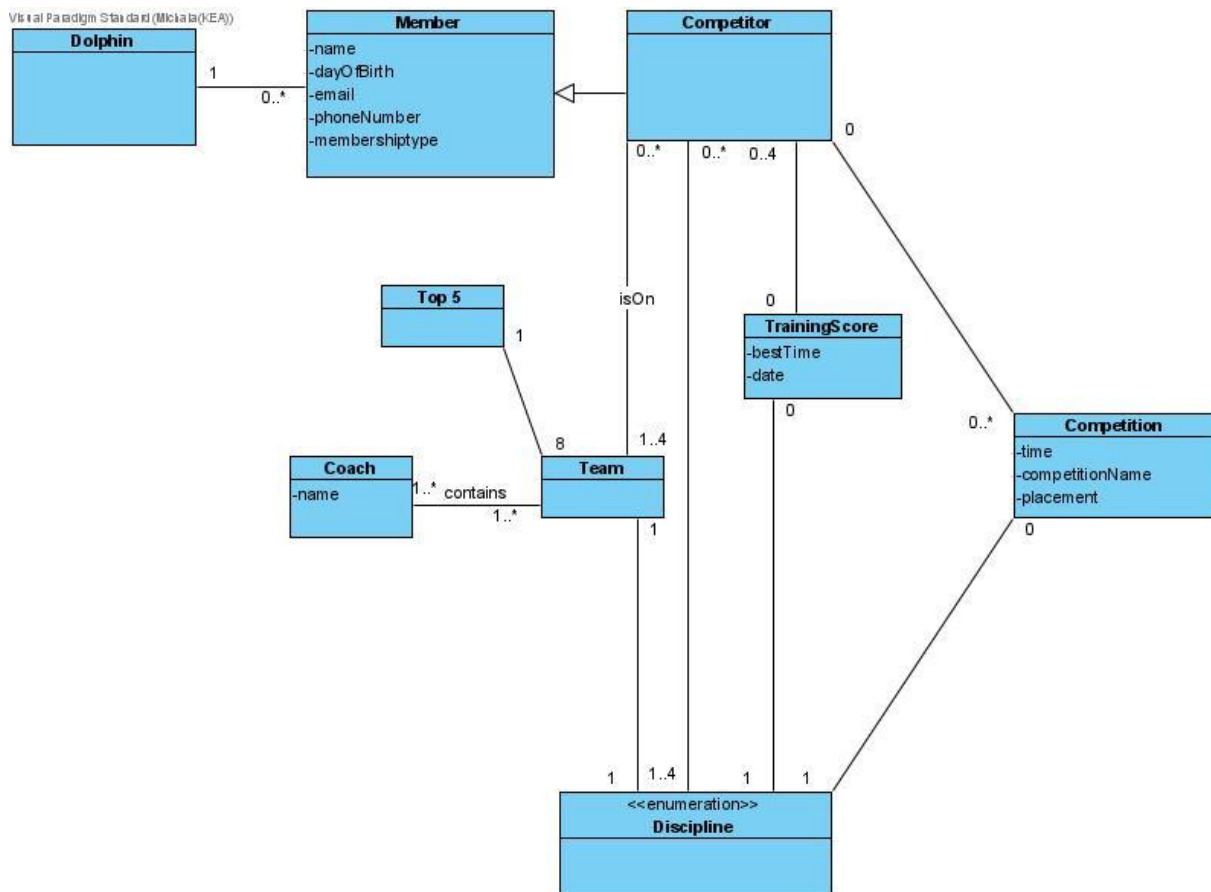
I det følgende fremvises dette projekts risikoanalyse. Denne analyse blev ligesom interessant analysen foretaget ved projekt start, men har måtte undergå tilføjelser som følge af projektets frem skridning.

Udvidet risikotabel for Marios-pizzeria-projekt							
Risikomoment	Sandsynlighed	Konsekvens	Produkt	Præventive tiltag	Ansvarlig	Løsningsforslag	Ansvarlig
Sygdom længerevarende (<u>corona</u>)	2	7	14	Følg restriktioner og pas på sig selv. Daglige opdateringer.	Michala.	<u>Carepackage</u> til den syge. Reevaluering af arbejdsfordelingen.	Kamille
Sygdom kortere forløb	4	3	12	Tøj, der passer til vejret. Gode sove- og spisevaner. Daglige opdateringer.	Michala.	Bliv hjemme. Kommunikation evt <u>outsourcce</u> . Fokus på eget helbred.	Kamille.
GitHub går ned	1	3	3	Huske at <u>push</u> og <u>pull</u> . Backup.	Rasmus.	ZIP-filer til hinanden. <u>CodeWithMe</u> .	Rasmus.
Vores arbejde forsvinder lokalt	3	1	3	Backup. Push og <u>pull</u> .	Rasmus.	Benyt sig af GitHub, evt <u>dropbox</u> el.lign.	Rasmus.
Lærer ændrer <u>projekt beskrivelse</u>	1	3	3	Kommunikation. Veltænkt <u>program</u> , der laves i iterationer.	Rasmus.	Kommunikation frustrationer til undervisere. Arbejd evt i weekenden.	Michala.
Mister adgang til Teams	1	1	1	Klar kommunikation i gruppen. Herunder	Kamille.	Benyt andre platforme evt <u>discord</u> .	Rasmus.

				alternative løsninger.			
Gruppemedlem forlader gruppen	1	7	7	Være søde og respekterende over for hinanden. Huske at lytte.	Michala.	Opdel gruppemedlems arbejde imellem resterende <u>gruppemedlemmere</u> . Tag fat i gruppemedlem og underviser.	Gruppen.
Gruppemedlem mister/har elendigt internetforbindelse	5	3	15	Skifte udbyder. Have GB på mobil klar.	Kamille.	Finde andet net. Mødes hos en i gruppen.	Kamille og Michala.
Gruppedynamik	2	3	6	Lytte til hinanden – gå nysgerrigt til værks. Lave løbende tjek-ins. Lav gruppekontrakt.	Kamille.	Referere tilbage til gruppekontrakten og prøv at kommunikere alle sider ud.	Kamille.
Underviser bliver syg	4	3	12	Sørge for løbende at spørge undervisere omkring opgaven og fra start sikre en forståelse af	Michala	Spørge andre ressourcer og omstrukturere arbejde rundt om de ressourcer som er til stede. Altså plan ændringer.	Michala
				omfanget af krav.			

Domæne model:

Efter samtale med kunde, fremstilles følgende domænemodel.



Use case model og beskrivelse:

I det følgende fremsættes programmets fire use case beskrivelser efterfulgt af use case model. Dette blev udarbejdet til start, således blev et overblik over projektets indhold tydeliggjort. Dette gjorde afgrænsning af projektet overskueligt om end senere opdateringer måtte finde sted.

Manage Members

The chairman uses the system to register information about person wanting to become a member. The system records and updates information about members.

Manage Members - Includes the possibility to:

- Add member.
- Remove member.
- Edit member.
- Show members.

Expected Income

The cashier uses the system to check expected income of the club. The system displays the expected total of all members membership prices.

Expected Income – includes the possibility to:

- Show expected income

Manage Member Debt

The cashier uses the system to manage members in debt. The system displays, record and updates information about members in debt.

Manage Member Debt – includes the possibility to:

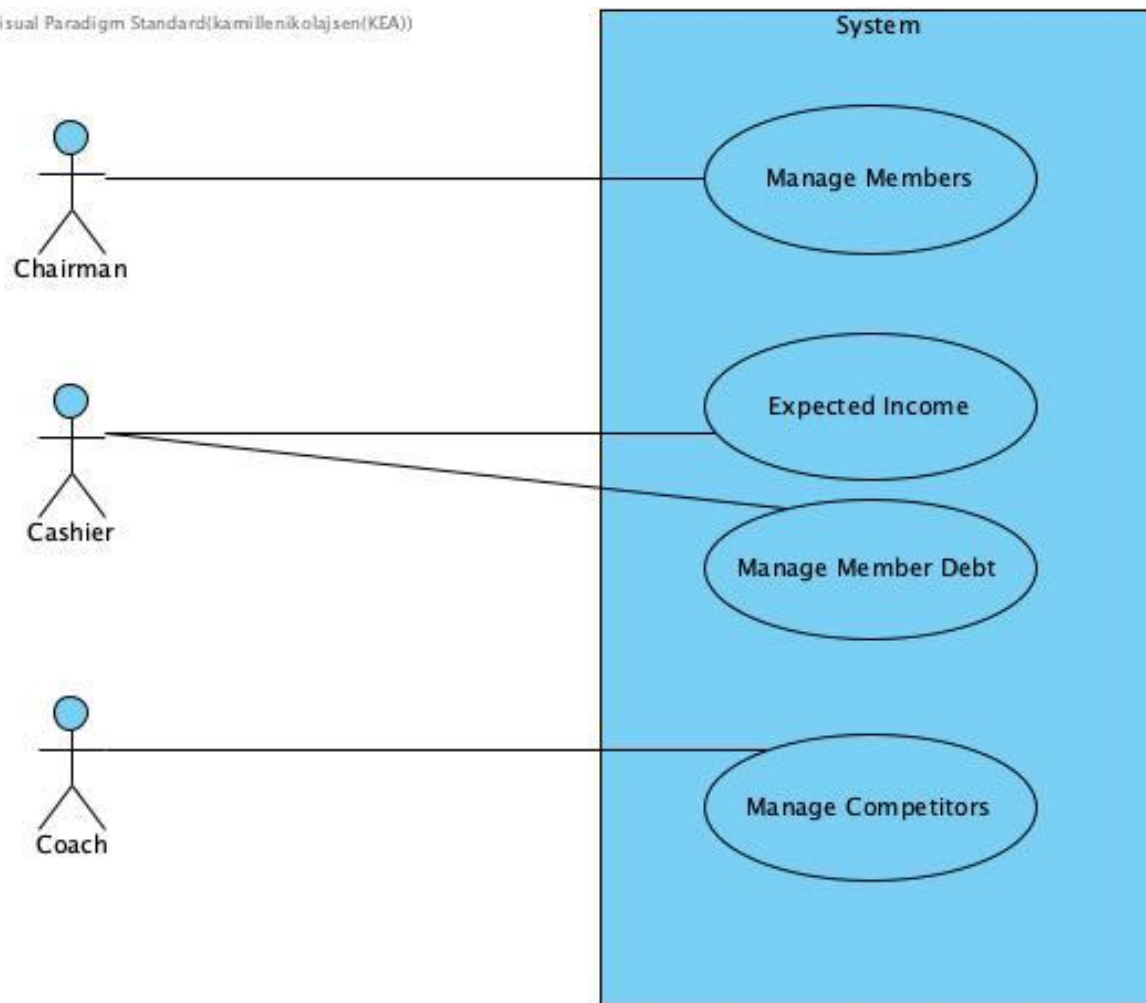
- Show people in RKI
- Edit member in debt
- Move member to RKI
- Remove person from RKI

Manage Competitors

A coach uses the system to manage competitors disciplines and to view a list of each teams top 5, and from there register competitions.

Manage Competitors – includes the possibility to:

- Show competitors
- Show team
- Show top 5
- Register training result
- Register competition
- Remove discipline
- Add discipline



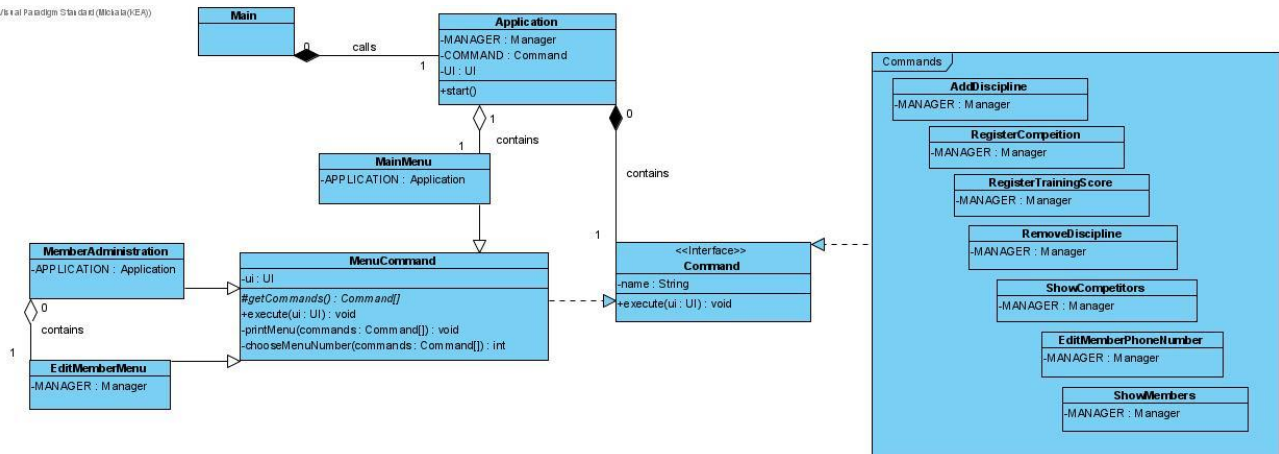
Design klassediagram:

Her fremsættes projektets klassediagrammer med undtagelse af tidligere viste domæne model. Idet projektet viste sig at være omfattende udarbejdes fire efterfølgende klassediagrammer.

Nedenstående vises i rækkefølgen: Member administration, economy administration, competitive administration og samlede diagram.

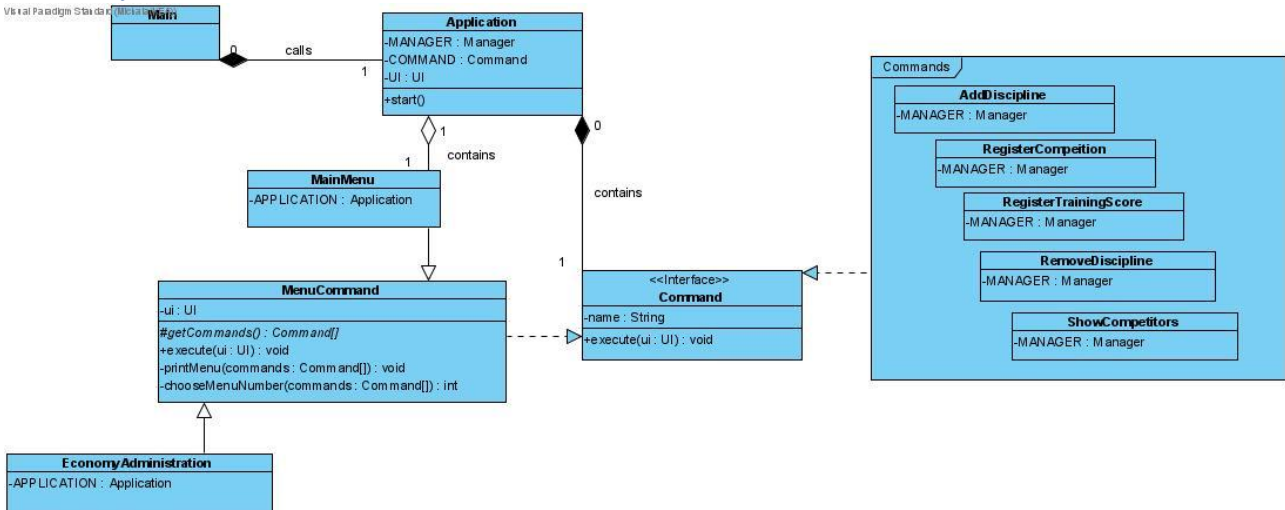
Member administration

Visual Paradigm Standard (UML 2.5.1)



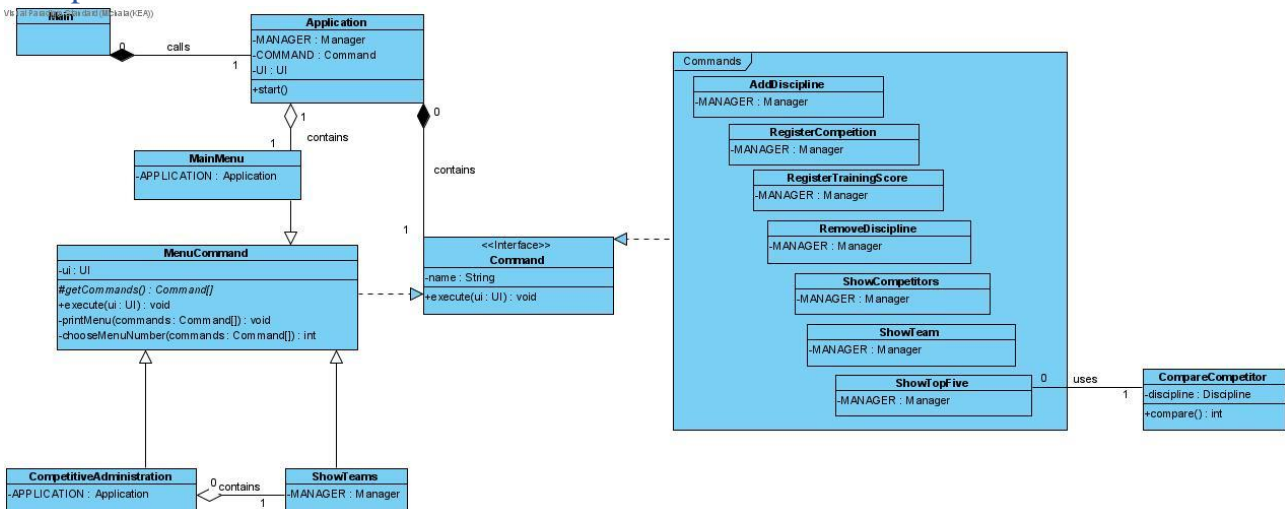
Economy administration

Visual Paradigm Standard (UML 2.5.1)



Competitive administration

Visual Paradigm Standard (UML 2.5.1)



Følgende diagram findes på GitHub ligesom de andre, men denne virker særligt nødvendigt at se i et større format.



I design fasen bliver det dog afgjort at der er behov for en videre udarbejdelse af designet således programmet ikke bliver forankret i en klasse. Hertil benyttes command pattern, hvorved programmets funktioner implementerer interfacen command. Første iteration bliver således at skabe strukturen for programmet, og anden iteration bliver implementeringen af første use case. Når dette er færdiggjort, er planen at tage fat på tredje iteration og lave cashierens menu samt de use cases som følger heraf. Herunder ville kunne argumenteres for ManageDebt og ShowIncome som to underlæggende use cases. Valg af rækkefølge på hvilke use cases, der gribes an, hvornår gik først og fremmest på at en member er nødvendig at oprette før vi kan teste og checke vores program, hvorefter den økonomiske del blev valgt som næste iteration således Delfinen, ville være garanteret det økonomiske overblik en lille virksomhed i vækst er så afhængig af. Den sidste del af problemformulering om coach skulle være det sidste at gå

til, hermed skulle den fjerde iteration afslutte selve programmeringsfasen før refactoring ville blive de følgende iterationer.

Sidste use case ManageCompetitors bliver dels mere uspecifik og dels langt mere krævende end de øvrige samt en byggesten på foregående use cases, idet om man er excersiser eller competitor er ud fra at være et aktivt medlem, hvorfor denne iteration anslås til at tage langt længere tid end foregående.

Status af kode

Programmet lever op til kundens formodede krav dels fra problemformulering og dels for samtale med kunden (Tine). Ved mere tid kunne en cleanup af koden, således metoder kun gør én ting og en implementering af loginsystem for forskellige brugere være en overvejelse at implementere.

Yderligere dokumentation og test kunne udføres til bedre at give en forståelse af programmet og sikre nemt at finde fejl på sigt.

Bugs:

Når man indtaster fødselsdato på member vil man skulle følge den formattering som er sat. Desuden er det muligt at indtaste andet end meningsgivende telefonnumre, navne og emails etc.. Dette vurderes som mindre bugs idet systemet er internt og det formodes at være acceptable kompromis. Hermed er der generelle bugs i systemet, grundet mangel på validering, der ligeledes er nedprioriteret grundet systemets interne funktionalitet.

GitHub

Rasmus-Kibshede

KamilleNikolajsen

MichalaNybroe

Alle tabeller og modeller er at finde på GitHub.

<https://github.com/Rasmus-Kibshede/Dolphin.git>