Svømmeklubben delfinen

Gruppe
"Get it done!"

Gruppen Medlemmer

Simon Melbye Andersen - Arhqirah

Christian Rodrigo - RodrigoPerea

Jonathan Willumsen - JWillumsen89

Sille Løkke Lorentzen - silleloekke

Github repository
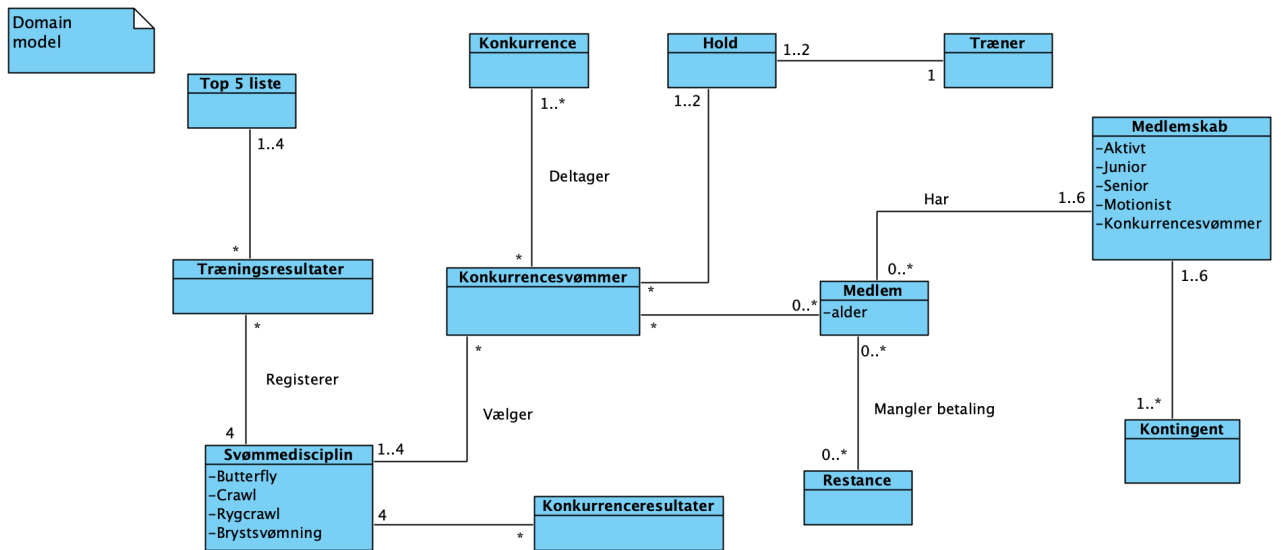https://github.com/JWillumsen89/Delfinen.git

Indholdsfortegnelse

*UML diagramerne findes som billeder inde i vores githup repository, hvis de skulle være svære at aflæse
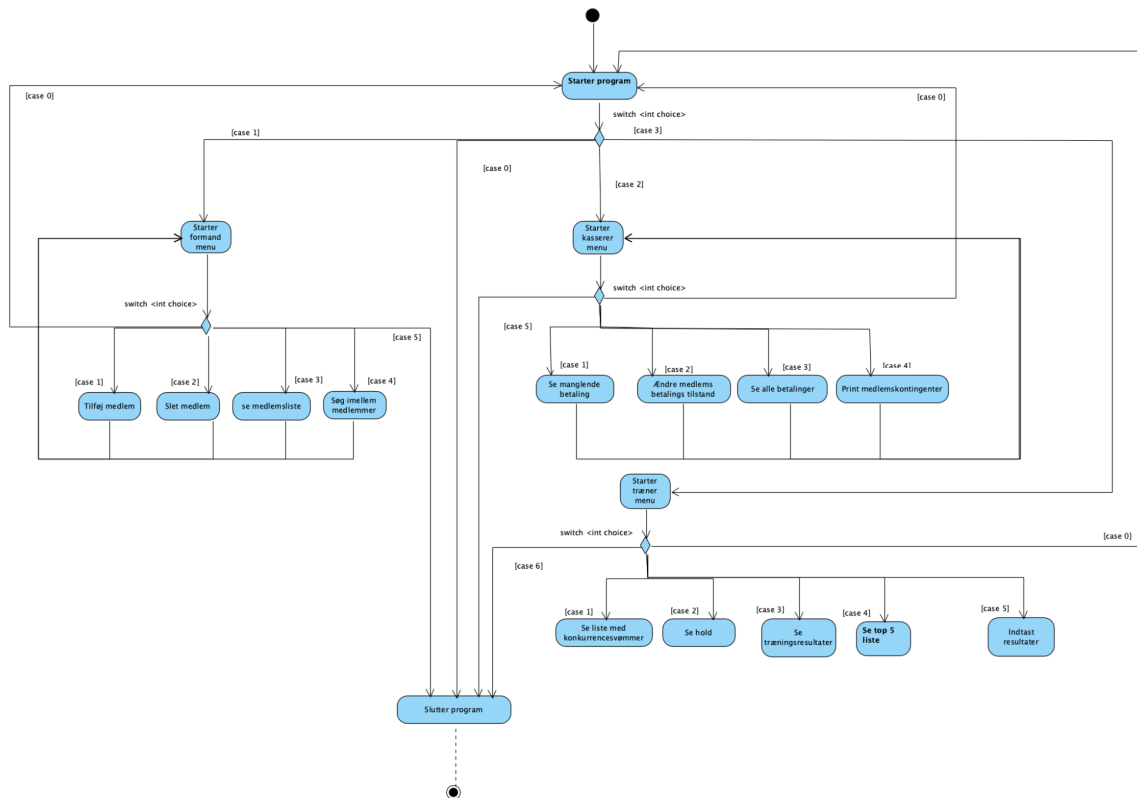
User stories

1. Formand: As a user, I should be able to register new members, with their information.
2. Kassere: As a user, I want the system to set payment status to "not paid" automatically, upon registration.
3. Coach: As a user, i must assign a swimming coach to a member, if i mark the membership as "konkurrencesvømmer"
4. Coach:As a user i have to choose from a list of swimming disciplines upon registration of a "competition swimmer"
5. Formand: As a user, i should be able to see a list of all membership status (Aktivt/passivt)
6. Kassere: As a user i should be able to change members payments status from not paid, to paid and vice versa
7. Kassere: As a user i should be able to get a print of total amount of income for a year based on all members.
8. Kassere: As a user i should be able to get a print of all members, which payment status is not paid
9. Kassere: As a user i should be able to see a list of members, who are in arrears
10. Kassere: As a user i should be able to get a list of all the membership fees
11. Kassere: As a user i want the system to calculate automatically membership fees based on their age
12. Træner: As a user I should be able to choose a swimmer from a list, and register a date, and time for each swimmer on a training-score-sheet.
13. Træner: As a user I should be able to register a swimmer's result from a competition and in what discipline they were competing in.
14. Træner: As a user I want to see the swimmer's name, time and category when I print the top 5 list.

Domain model

Domain model

Top 5 liste

1..4

Konkurrence

Hold 1..2 **Træner**

1..2 1

Medlemskab
−Aktivt
−Junior
−Senior
−Motionist
−Konkurrencesvømmer

1..*

Deltager

*

Træningsresultater

*

Har 1..6

0..*

Medlem
−alder

1..6

Konkurrencesvømmer

* 0..*

* *

1..6

Registerer

Vælger

4 0..*

Svømmedisciplin
−Butterfly
−Crawl
−Rygcrawl
−Brystsvømning

1..4

Mangler betaling

Kontingent

1..*

4

Konkurrenceresultater

0..*

Restance

*

Activity diagram

Class diagram

Filehandler

FileHandler
- -fileName : String
- -fileNameID : String

DatabaseException

delfin

Main
- +main(args: String[]): () : void

Controller
- -name : String
- -email : String
- -breaststroke : String
- -butterfly : String
- -backcrawl : String
- -crawl : String
- -final team : String
- -final coach : String
- -phoneNumber : int
- -paymentCategory : double
- -active : char
- -answerCom : char
- -competitorOrRegular : char
- -paidOrNot : char
- -age : LocalDate
- -programRunning : boolean
- -running : boolean
- -intError : boolean
- -correctInput : boolean
- -added : boolean
- -members : ArrayList<Member>
- -searchedForMembers : ArrayList<Member>
- -trainingScores : ArrayList<TrainingScore>

ui

UserInterface
- -choice : String

colors

<<static>>
color

members

Member
- -name : String
- -email : String
- -active : String
- -passive : String
- -paid : String
- -notPaid : String
- -butterfly, breaststroke, backcrawl, crawl : String
- -coach : String
- -teams : String
- -age : int
- -phoneNumber : int
- -memberID : Integer
- -activeOrPassive : char
- -competitorOrRegular : char
- -paidOrNot : char
- -memberCategory : double
- -paymentCategory : double
- -members : ArrayList<Member>
- -restance : ArrayList<Member>
- -added : boolean
- -curDate : LocalDate
- +toString() : String

finance

MembersFee
- -final fees : int[]
- -category : double

competitors

Competitor
- -MemberID : Integer
- -name : String
- -discipline : String
- -type : Char
- -competitionLocation : String
- -competitionResult : int
- -milliseconds : int
- +toString()() : String

Controller   Member   Membersfee   FileHandler   UserInterface

1: addMember();

Sequence diagram of addMember(), der befinder sig inde i chairman

1.1: typeName();

1.2: typeDOBException();

1.3: typeDOB();

1.3.1: calculateAge();

1.3.2: return int age;

1.3.3: paymentCategory();

1.3.3.1: paymentCategoryCalculator

1.3.3.2: return double category

1.4: typeEmail();

1.5: typePhoneNumber();

1.6: addMemberID();

1.6.1: loadMemberID();

1.6.2: return Integer.parseInt(String data);

1.7: typeMemberStatus();

1.8: scannerBugFix();

loop [if active == "A"]

1.9: typeCompetitionOrNot();

loop [do]

If statements

loop [if answerCom == comp]

[else if answerCom == regular]

[else]

1.10: wrongInput

[while !answer]

1.11: scannerBugFix();

loop [if answerCom == "C"]

1.12: pickDiscipline()

1.12.1: printDisciplineMenu();

1.12.2: returner printDisciplineMenu();

alt

1.12.3: case 1 -> butterfly

1.12.4: case 2 -> crawl

1.12.5: case 3 -> backcrawl

1.12.6: case 4 -> breaststroke

1.12.7: case 5 -> running

1.12.8: default -> wrongInput();

1. switch case

1.13: saveRegularMember();

alt

1.13.1: case "Y" -> createNewMember();

1.13.1.1: <<member>>

1.13.1.2: return Member member

1.13.1.3: save();

1.13.1.3.1: saveDatabase();

2. switch case

1.13.1.3.1.1: saveMembersToFile();

1.13.1.3.1.2: return Member member

Returner tilbage til saveMember, som er den første switch case

1.13.1.3.1.3: saveMemberID();

1.13.1.3.1.4: return Member member

1.13.2: case "D" -> System.out.println

1.13.3: case "E" -> editMember

alt

1.13.4: case "N" -> changeName();

1.13.4.1: typeName();

Returner til editMember(), som er den anden switch case inde i den første

1.13.5: case "D" -> saveMember();

1.13.6: case "E" -> changeEmail();

1.13.6.1: typeEmail();

1.13.7: case "P" -> changePhoneNumber();

1.13.7.1: typePhoneNumber();

1.13.8: case "M" -> changeActiveOrPassive();

1.13.8.1: typeMemberStatus();

1.13.9: case "EXIT" -> chairman();

1.13.10: default -> editMember();