

Delfin projekt

Grp 7, Adil, Magnus og Morten

28-05-2017

Projektrapport Delfin – projekt:

Vi har i vores opgave startet med at lave use cases. Til dette projekt har vi valgt at lave brief use cases, dvs. uden alternative scenarier. Use casene er forskellige handlingsbeskrivelser af hvad vores program skal kunne.

Herefter har vi lavet et use case diagram. Her har vi forbundet system/admin til de forskellige use cases, for at få os et bedre overblik.

Herefter har vi ud fra vores use cases lavet en navneord liste og en udsagnsord liste.

Vi har fundet alle navneordene i vores use cases og skrevet i en tabel, og det samme har vi gjort med udsagnsordene i vores use cases.

Disse lister har vi brugt som udgangspunkt til vores domain model. De navneord som vi vurderede til at skulle bruges som klasser i vores program, dem puttede vi ind i domain modellen. Udsagnsordene har vi brugt som bindeled mellem de forskellige bokse i domain modellen.

Herefter har vi så udvalgt en use case til at lave et SSD diagram. Her beskriver vi interaktionen mellem systemet og admin / cashier. Da der ikke er så meget interaktion mellem en "bruger" og systemet, så har vi ikke lavet så mange SSD diagrammer.

Vi har også lavet et SD diagram. Til dette har vi valgt "Create swimmer" som udgangspunkt. Vi har beskrevet processen omkring "create swimmer" i de forskellige klasser.

Sidste diagram vi har lavet inden koden er klassediagrammet. Her har vi tilføjet attributter og metoder, så vi kunne kode ud fra det.

Undervejs i vores projekt, har vi redigeret i vores klassediagram og domain model, da vi kom længere frem i vores kode og fandt ud af vi kunne tilføje/fjerne nogle metoder og klasser.

Use cases:

Title: Register swimmer

Actor: Admin

Scenario: The chairman enter swimmers personal information
System approves and create swimmer
System saves swimmer

Alternative Scenario:

Title: Save swimmers in arrears/debt

Actor: System

Scenario: System register swimmers in debt
System shows a list of people in debt

Alternative Scenario:

Title: Add swimmers to coach

Actor: System/Admin

Scenario: System register the swimmer
System add the swimmer to the coach

Alternative Scenario:

Title: Save swimmer in relevant group

Actor: System

Scenario: System check personal information on swimmer
System then add the swimmer in relevant group

Alternative Scenario:

Title: Save swimmers best result continually in each competition

Actor: System

Scenario: System compares swimmers new result with all previous results
System saves the new result, if better than the previous ones

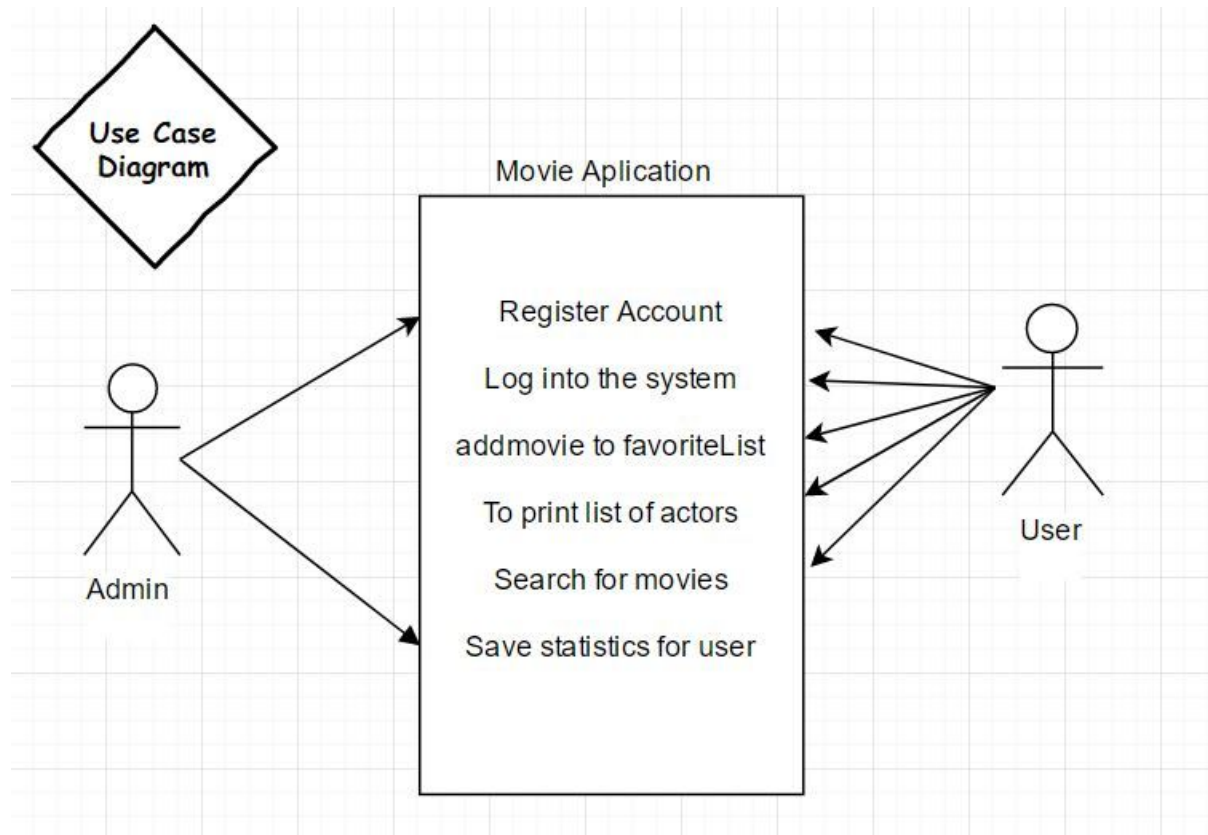
Alternative Scenario:

Title: Show top 5 swimmers

Actor: System

Scenario: System runs through all swimmers
System show a list of top 5

Alternative Scenario:



Noun and verb list:

<u>NounList:</u>	<u>VerbList:</u>
Admin	Register
Swimmer	Enter
Chairman	Approve
Personal Information	Create
Debt	Save
Coach	Show
Group	Add
Result	Check
Competition	Compare
Top 5	Runs trough

