

Klasserna

Game klasen

Detta är klassen som sätter igång spelet, samtidigt som den har en metod som skriver ut en välkommen text. Här finns även metod som tar in statiskt och omvandlar dem till ett resultat för användaren.

UserInteraction

Detta kan man säga är självaste huvudelen i klassen, här hämtas frågorna in för att kunna ge dem till användaren, tar in användarens svar och jämför dem med de rätta svaren.

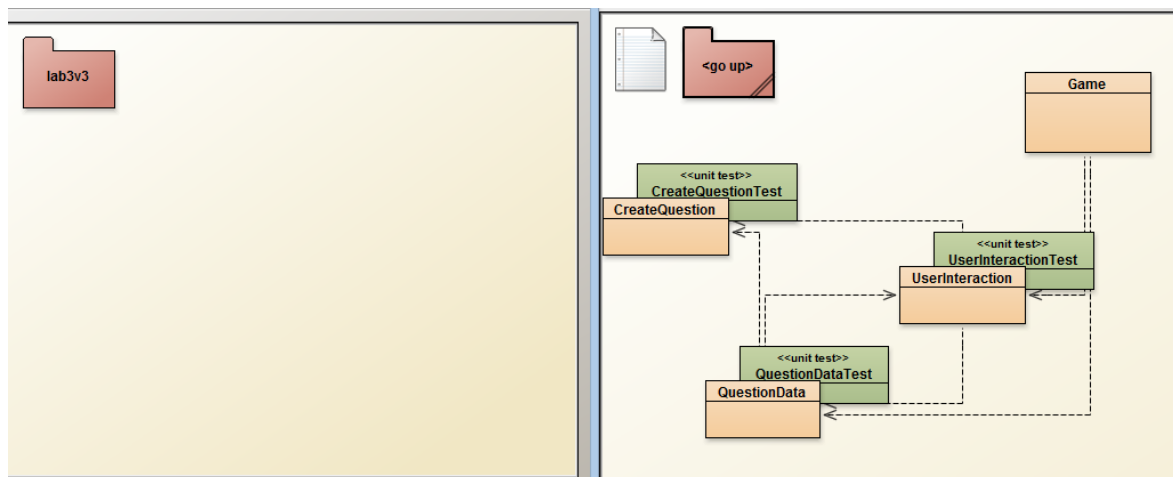
QuestionsData

Denna klass skapar frågorna till spelet, här finns även metod för att slumpa om elementen i en Lista.

CreateQuestion

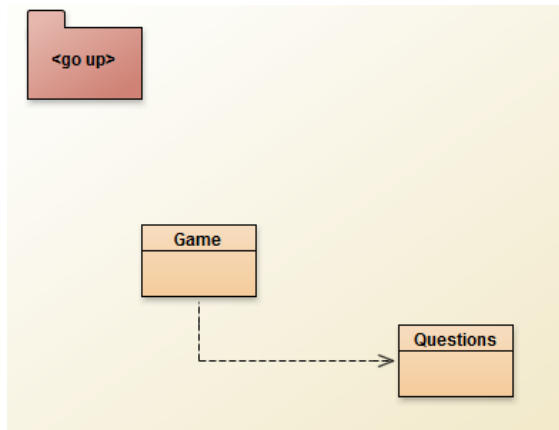
Detta är ett så kallad skelett klass är till för att definiera hur ett fråge objekt skall se ut.

All detta har lagts in ett paket.



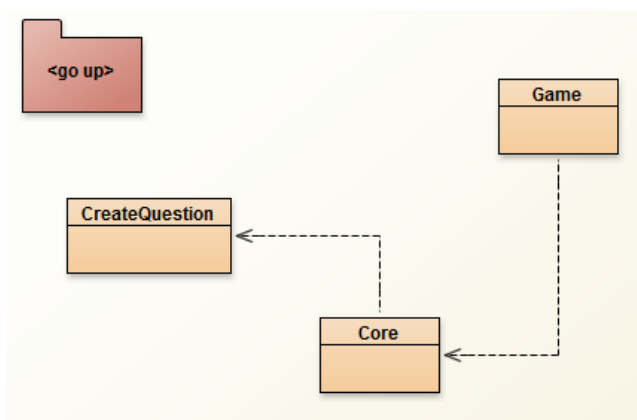
Första Bygget utav spelet LAB 3V1

Mitt mål här var att bara få till något som fungerar och sedan därefter förbättra den. Här hade jag byggt två listor där jag hade frågor i en och svar i den andra och med hjälp utav random generade jag fram ett nummer som jag använde båda till frågor och svår och på så sätt kopplades alltid rätt svar till frågan, det kändes verkligen dåligt men det funkade. Jag kände att det var för mycket ansvar för två klasser och valde att refaktorera till nästa lösning.



Andra Bygget utav spelet LAB 3V2

Det jag var mest missnöjt med utav bygge 1 vara hur frågorna och svaren var sammankopplades, det kändes riktigt dåligt, så jag bestämde mig först att försöka hitta en bättre lösning till detta. Jag kom fram till att det vore bästa att skapa ett fråge objekt där jag kunde binda svaren till frågorna och med hjälp utav en skelett klass kan jag själv bestämma hur detta skall se ut. Jag väljer att göra om min ArrayList till en HashMap istället där jag har en Integer som key och fråge objektet som value. Tanken här var att jag skulle försöka skapa en metod där jag kunde använda Integern i key som någon sorts ID så att random funktionen inte skulle kunna kalla på samma fråga, men tyvärr fick jag det aldrig att fungera så jag bestämde mig för att ändra om bygget.



Tredje och sista bygget LAB 3V3

Jag tyckte att Core klassen fortfarande hade för mycket ansvar och skapade en klass till QuestionData som endast är till att lägga till frågor, med tanke på att spelet går ut att ställa frågor till användaren så borde det vara enkelt att utöka andelen frågor.

Mitt problem med att hur jag skulle hantera att samma fråga inte dök upp fler än en gång under varje spelomgång ledde mig till att ändra om hur jag lagrade frågorna igen, det blev skippa HashMappen och köra på arrayList igen för att kunna använda shuffle metoden för att slumpa om elementen, detta gjorde även att när användaren valde att spela om ett spel med samma antal frågor så fick andra frågor.

Jag tar hjälp utav variabeln noQuestions för att bestämma hur många list element (frågor) som skall skrivas ut, Detta kan du se på metoden userNumberOfQuestions() där jag använder int i som parameter som in sin tur använder det till list elementen.

Jag kan tycka att jag eventuellt borde ha skapat en klass till och ta ut userInput hanteringen i UserInteraction till en egen klass men programmet är inte så stort så jag nöjde mig med 4 klasser. Jag försköte med att använda inheritance lösningar men fick hela tiden nullpoint error exception fel, så jag beslöt mig för att skippa det. Jag gjorde en del sökningar på interface samt abstrakta klasser men ansåg att jag inte hade tillräckligt med kunskap för att utnyttja dessa än.

