# Examen .Net Advanced 2014-2015

Theorie gedeelte

bestaat uit redelijke kutvragen , er wordt een stuk code gegeven waar dingen aan schelen + een groot deel opengelaten. In dit opengelaten deel moet je zelf code schrijven zodat het programma werkt. In dit geval kwam de code rechtstreeks uit een oefening , nl. wpfVendorMaintanance ,Je moest hier ook via een messagebox er voor zorgen dat de geselecteerde naam vanuit een tabel via ID ofzoget afgebeeld werd. De tweede Theorie vraag was in verband met een andere oefening maar dit weet ik niet meer, k’weet enkel da et fucking gay was. De derde oefening bestond uit een soort multiple choice vraag , het was eerder verbind met wat he past stijl. Deze was redelijk makkelijk op te lossen met logisch nadenken, het ging over iets met Dirty reads en ghost reads en unreredebeale reads ofzoget. De 4de vraag was hoe dat deze problemen opgelost kunnen worden in .net. DE 5de vraag was iets van wat betekend dit: “the perfect code blablablabla” leg uit liefst via een voorbeeld.

## Praktijk gedeelte

**Omschrijving van de applicatie**

Je gaat een toepassing schrijven die de peerevaluaties (studenten die elkaar binnen een projectgroep evalueren) van een project kan uitvoeren. Het is hierbij erg belangrijk dat je de implementatierichtlijnen (zie verder) strikt volgt. Er is ook heel wat code gegeven. Door middel van commentaar is aangegeven waar je code moet toevoegen:

“TODO: Dit is een voorbeeld hoe de code gemarkeerd wordt die jij moet aanvullen”

Laat al deze commentaren staan en verwijder ze dus niet; dit helpt bij het verbeteren. Start nu Visual Studio en open de solution genaamd *PeerEvalTool*. Je zal merken dat er twee projecten geopend worden: *PeerEvalDB*  en *PeerEvalTool*.  
Daarnaast heb je een 2de solution, genaamd Kubus die je nodig hebt voor het oplossen van de laatste vraag.  
Lees eerst de ganse opgave volledig door, eer je start met het oplossen van het examen.

**Database ontwerp**

Voer het script CreatePeerEvalDatabase.sql uit. (PRO TIP: Die fuckers hebben dit ni zo genoemd, da noemt gewoon script.sql) Dit script creëert de PeerEval Database met daarin de volgende tabellen:

(NORMAAL IS HIER NE FOTO MAAR IK WEET NI HOE IK DIT KAN SCREENCAPPEN DUS IK SCHRIJF DE 4 TABELLEN MAAR FF)

Students  
Studentgroups  
Peerevaluations  
workpackages

Studenten (*students*) participeren in groepen (*studentgroups*) waarbij elke groep uit 3 of 4 studenten bestaat. Dit wordt ecchter niet door het model afgedwongen. Voor elke deadline van een *workpackage* moeten studenten uit een groep mekaar evalueren. Elke evaluatie is een rij in de tabel *peerevalutions*. Deze tabel heeft de volgende kolommen:

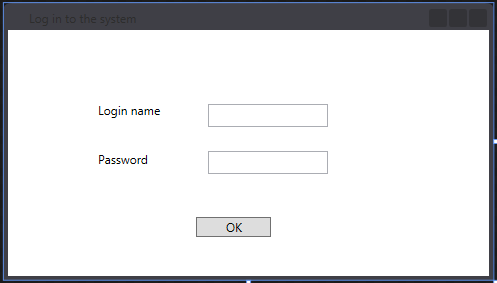
* *Wpid*: id van het werkpakket
* *From\_student*: student die het cijfer geeft
* *To\_student*: student die het cijfer krijgt
* *Amount*: het cijfer
* *Comment*: eventuele opmerkingen
* *Timestamp*: de datum en tijd wanneer het cijfer is gegeven

We verlopen nu de verschillende schermen.

**Schermen**

**LoginWindow**

Bij het starten van de applicatie verschijnt het volgende loginscherm:



(titel = Log in to the system)

Volgende foutmeldingen zijn mogelijk: (IK HEB DIT NIET WANT IK BEN EEN KIEKEN IN .NET DUS IK TYP MAAR WAT ER STAAT)

Ik denk dat het messaboxen zijn :

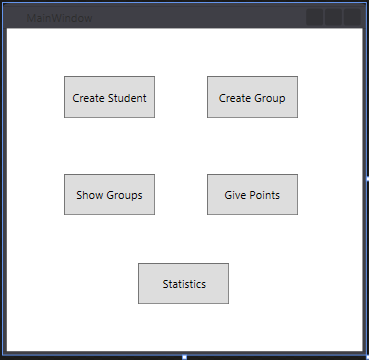
Messagbox 1  
Titel = Login Exception  
Tekst erin = Login Failed: wrong password

Messagbox 2  
Titel = Login Exception  
Tekst erin in dit voorbeeld = Unknown user: pipo

ALS HET INLOGGEN LUKT VERSCHIJNT HET VOLGENDE SCHERM. (ppienter/ppienter en bbibber/bbibber zijn succesvole inlognamen ! )

**MainWindow**

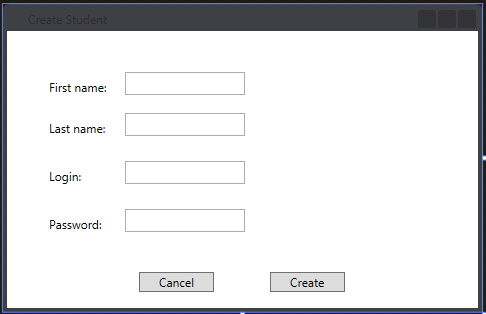
Dit is het hoofdscherm van de applicatie. Bemerk dat in de titelbalk de volledige naam van de ingelogde gebruiker wordt getoon.



Titel = Student: Piet Pienter

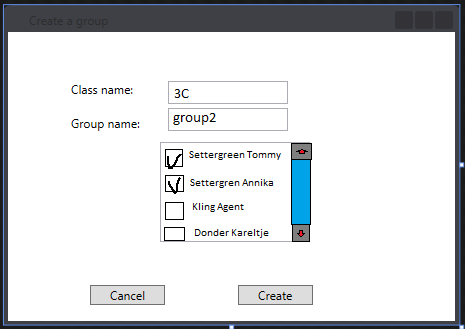
Van hieruit zijn er 5 mogelijke functionaliteiten die kunnen opgeroepen worden.

**CreateStudentWindow**Hiermee kunnen nieuwe studenten aangemaakt worden. Deze studenten hebben nog geen groep. De gebruiker dient alle velden in te vullen.

  
Foto van een erorr message: Titel = Entry Error , Tekst = Last name is a required field.

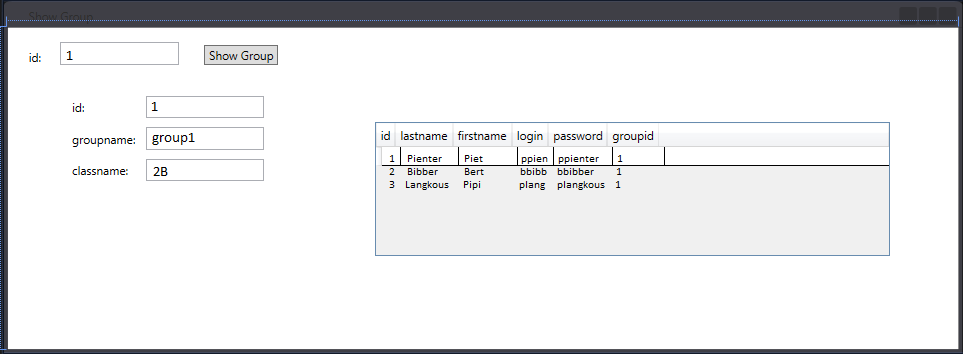
**CreateGroupWindow**

Hiermee kunnen nieuwe groepen samengesteld worden. De lijst met een checkboxitems presenteerd enkel studenten die nog **geen** groep hebben. De gebruiker dient exact 3 of 4 studenten te selecteren en dient alle overige velden in te vullen.

  
Titel = Create a group  
Error message : Titel = Entry Error, tekst = Please select 3 or 4 students

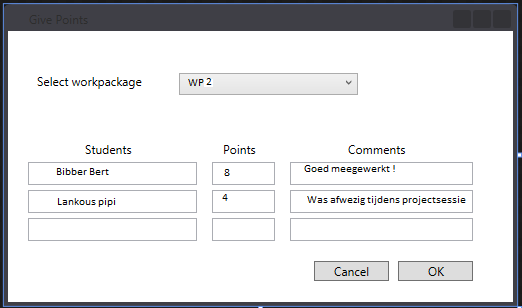
**ShowGroupsWindow**

Hiermee kan de gebruiker een id ingeven van een groep, en na een klik op de knop “Show Group” worden de details van deze groep getoond.

  
Titel = Show Group

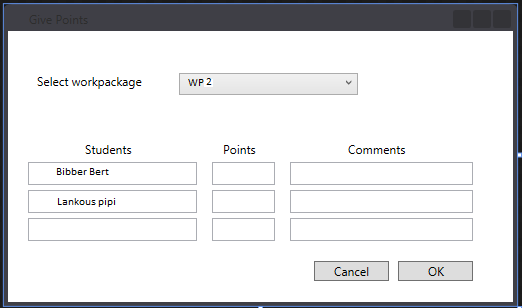
**GivePointsWindow**

Hiermee kan de ingelogde student zijn groepsleden beoordelen. Er zijn twee mogelijkheden:   
Ofwel selecteerd de student een workpackage waarvoor hij voorheen reeds een beoordeling had ingevuld. In dat geval kan hij enkel nog deze waarden bekijken. Onderstaand scherm ttont dit voor gebruiker “ppienter”:



Titel = Give Points

Ofwel selecteert de student een workpackage waarvoor hij nog geen evaluaties had ingevuld. Nu kan hij de punten ingeven en bewaren. Onderstaand scherm toont dit voor gebruiker “ppienter”:

  
Titel = Give Points  
Message na op OK klikken: Titel = NIKS = “”, tekst = Peerevaluations saved !  
MERK OP DAT DE LAASTE RIJ TEXTBOXES DISABLED BLIJFT OMDAT DIT EEN GROEP VAN SLECHTS 3 PERSONEN IS

Voeg 2 animaties toe aan het GivePointsWindow:

Zorg ervoor dat de OK knop van een lichte groene kleur evolueert naar een donker groene kleur wanneer je met de muis in de zone van de drukknop komt. Laat de kleur evolueren tussen de 2 kleuren in 5 seconden. Vervolgens wordt het effect omgekeerd. Buiten de zone van de drukknop heeft de drukknop een lichte groene kleur.

Kies 1 textbox en zorg ervoor dat de achtergrondkleur evolueert van oranje naar geel in 10 seconden en omgekeerd.  
De animatie pauzeert wanneer de muis in de zonde van de textbox komt en gaat verder wanneer de muis de zone verlaat.

(Hier staat een foto maar die heeft geen nut van te painten want da is animatie -\_- )

**StaticsWindow**

PRO TIP: DIT IS GEEN FUCKING WINDOW , DA’S GEWOON EEN MESSAGEBOX ALS GE OP U MAINMENU OP STATISTICS KLIKT, ECHT EEN UUR OP ZITTEN ZOEKEN #OVERDREVEN

Hiermee kan de ingelogde student (in de screensot van Bert Bibber die is ingelogd) zijn scores worden getoond, maar wel het gemiddelde, de minimumscore en de maximumscore.

Foto van een messagebox: Titel = Statistics, Tekst =  
Results for Ber Bibber  
 Average: 8,000000  
 Minimum: 8,0  
 Maximum: 8,0

**Richtlijnen bij het coderen**

Tracht onderstaande richtlijnen zo goed mogelijk te volgen. Het is aanbevole om deze volgorde aan te houden. Als er iets niet helemaal werkt, tracht dan in commentaar aan te geven wat je bedoeling was. Zorg er steeds voor dat je programma compileert, dit vereenvoudigt het verbeteren.

IN DE SKELET CODE STAAT HEEL WAT NUTTIGE INFO IN ‘TODO COMMENTAAR’. LEES DIT!

**Student en StudentDB**

Schrijf de methodes *CreateStudent* en *Findstudent* in de klasse *StudentDB* van het project *PeerEvalDB.* Gebruik hiervoor **het lagenmodel**  (DAFUQ IS DIT ???)  
Een PK mag je niet expliciet meegeven, vermits deze automatisch voor students records worden aangemaakt.

**LoginWindow**

Schrijf de event handler voor de *okButton*. Deze moet volgens **het lagenmodel** geschreven zijn, (dus *Student* en *StudentDB* gebruiken).

OM LATERE VERWARRING MET DE STUDENT KLASSE VAN HET ENTITY FRAMEWORK TE VERIJMIJDEN, SCHRIJF JE BEST STEEDS PeerEvalDB.Student IPV GEWOON Student.

**Mainwindow**

Implementeer code zodat de naam van de student in de titelbalk terecht komt.

**CreateStudentWindow**

Schrijf de event handler voor de *createButton*. Deze moet volgens **het lagenmodel** geschreven zijn, (dus *Student* en *StudentDB* gebruiken).

**ShowGroupWindow**

Schrijf deze code door gebruik te maken van **een DataSet** genaamd peerEvalDataSet.xsd die er als volgt uitziet:

Aanpassen 🡪 StudentsTableAdapter 🡪 FillByGroupId

Nu hebt ge zo ne gigantische foto van zo tabellen, ik vermoed da ge een nieuw SQL statement hebt gemaakt waarbij ge kunt filteren op ID ofzoget, buh weet ik et.

DE DATATABLE PEEREVALUATIONS MET BIJHORENDE TABLEADAPTER ZAL JE NODIG HEBBEN VOOR DE STATISTIEKEN (ZIE VERDER), NIET VOOR HET TONEN VAN DE GROEPEN.

**StaticsButton in MainWindow**

Als je deze DataSet hebt gemaakt, kan je voor de ingelogde student al zijn gekregen peerevaluations ophalen via de FillByStudentId methode uit de peerEvaluationsTableAdapter. Vervolgens schrijf je een LINQ query die hiervan hetgemiddelde, het maximum en het minimum berekent. Dit alles doe je in de event handler voor de statics button in MainWindow.

**CreateGroupWindow**

Maak een Entity Model van de gegeven database en hanteer hierbij volgend schema (PeerEval.edmx)

Hier hebt ge weer een gigantische foto van een tabel maar kweet begot ni hoe ge daar geraakt. Iets met linken plaatsen en shit buh

Voeg ook een propert *FullName* toe aan de entiteit *Student* door een Partial Class te schrijven (FullName = lastname & “ “ & firstname).  
Vervolledig nu de constructor CreateGroupWindow en de methode createButton\_Click zodat je niewe groepen kan aanmaken. Dit dient te gebeuren gebruik makende van het entity framework (het model dat je zojuist hebt aangemaakt)

**GivePointsWindow**

Vervolledig het window om punten te geven aan je groepsgenoten. Ook dit dient te gebeuren volgens het model van het Entity Framework.

## MAAK DEZE OPDRACHT IN HET PROJECT KUBUS

Voor de class Kubus gelden de volgende eisen:

* Een kubus heeft gelijke zijden, dit wil dus zeggen dat breedte, hoogte en lengte aan elkaar gelijk moeten zijn. Als dit niet zo is dan moet de correcte ArgumentOutOfRangeException opgegooid worden.
* De oppervlakte van een kubus wordt berekend door de oppervlakte van 1 vlak te berekenen en dit te vermenigvuldigen met 6. Bij een zijde van 5cm is de oppervlakte dus 150 cm²
* De inhoud van een kubus wordt berekend met de formule (lengte x breedte x hoogte). Bij een zijde van 5 is de inhoud dus 125 cm³

Schrijf voor de class Kubus de vereise unit tests. Pas de code van de class Kubus aan, indien nodig, om ze te optimaliseren.

EN TOT SLOT…

VEEL SUCCES!!!!!

Goed , ik heb ongeveer 2 uur en een half gedaan om dees tekst te typen, ik ben 30 uur wakker, vies moe, et fucking heeft gesneeuwdd, k’ga afgeven en pitten. yuuuuu