



Hogeschool PXL

Departement IT
Academiejaar 2017-2018

EXAMEN

Vak	
Resultaat	/20
Periode	Semester 2
Datum	08/06/2018
Tijdstip	8u30
Klassen	1TIN A – B – C – D – E – F – G – H – I – J – K – L
Lectoren	R. Lambrechts – C. Derkoningen – N. Custers – W. Hendrikx – M.Willems – K. Nys

Studentengegevens	
Naam student	
Voornaam student	
Klas	
Lector	
Lokaal	

Samenstelling bundel		
Onderdelen (*)	Deel 1	Deel 2
Inhoud	Theorie	Praktijk
Pagina's	8 p	13 p
Puntenverdeling /3p (15%) / 17p (85%)
Digitaal beginbestand		X
Digitale indiening		X
Toegelaten hulpmiddelen:		
* rekenmachine		
* laptop		X
* internet		
* cursusmateriaal		X
Opmerkingen:	Gesloten boek zonder laptop (deel 2 wordt dadelijk ontvangen na afgeven deel 1)	- Open boek laptop FileZilla: enkel handboek en gebundelde slides; geen losse bladen, geen usb-sticks, ... - Solution moet op standaard VS 2017 geopend kunnen worden - Startpunt: solution <i>DotNetExamenJuni</i> waarin verder gewerkt moet worden en 3 bestanden: <i>Main Stage.txt</i> , <i>The Barn.txt</i> en <i>The Slope.txt</i>
Elke student(e) is verantwoordelijk voor de correcte samenstelling van zijn/haar bundeltje. Eventuele afwijkingen moeten onmiddellijk aan de toezichthouderesignaleerd worden.		
MD5Hash:		
<div style="border: 1px solid black; height: 20px; width: 100%;"></div>		
Aanvangsuur examen: 8u30		
Einde examen: 12u00		

EXAMEN 41TIN1140 .Net Essentials

Puntenverdeling:

- Theoretisch deel (gesloten boek zonder laptop): 15%
- Praktijkvraag (open boek met laptop): 85%

Bij het verbeteren van de praktijkvraag wordt er rekening gehouden met de onderstaande criteria.

1001 0110	10%
--------------	-----

Niet compileerbare code of niet uitvoerbare code leidt tot een malus van 10% procent op het resultaat van de opgave.

Je haalt 14/20 op een opgave, maar je project compileert niet: 10% van 14/20 is 1,4/20 → Je behaalt 12,6/20.

{ CC }	1% max 10%
--------	---------------

Het niet volgen van de gangbare codeconventies leidt tot een malus van 1% per fout met een maximum van 10% op het resultaat van de opgave.

Je haalt 14/20 op een opgave, maar 5 klassen beginnen niet met een hoofdletter → 5% van 14/20 is 0,7/20 aftrek → Je behaalt 13,3/20.

✓ SPEC	1% max 10%
--------	---------------

Het niet volgen van de specificatie (naamgeving van bestanden, klassen, variabelen, ...) zoals gevraagd in de onderstaande implementatiedetails, leidt tot een malus van 1% per fout met een maximum van 10%.

De percentages worden eerst opgeteld vooraleer de malus berekend wordt.

Je behaalt 14/20, maar het compileert niet (10%), er zijn 5% codeconventiefouten en 5% specificatiefouten → Dit geeft een malus van 20%.

Elk maluspercentage wordt PER VRAAG berekend, oftewel per IDE-project.

Praktijkgedeelte

Lees eerst aandachtig de volledige opgave.

STAPPEN UIT TE VOEREN 'VOORALEER' JE MET HET EXAMEN BEGINT

Verander de directory Festival van naam. Deze moet bestaan uit Festival + Achternaam + Voornaam. Dit alles zonder spaties en in deze volgorde.

Bijvoorbeeld: FestivalDerkoningenCarine

Kladblad: Dit mag los gemaakt worden van de bundel tijdens het examen. Je moet hier dan wel **eerst** je naam op vermelden en het blad steeds voor je laten liggen. Ook ben je **verplicht** om dit mee af te geven met de examenbundel!!!

STAPPEN UIT TE VOEREN WANNEER JE KLAAR BENT MET HET EXAMEN

Maak een zip-file van FestivalNaamVoornaam. Je zip-file moet als naam FestivalNaamVoornaam.zip hebben (bv: FestivalDerkoningenCarine.zip).

Vooraleer je je zip-file gaat uploaden, dien je eerst de md5 op te vragen en te noteren op je examenkopij.

Vergeet je kladblad niet toe te voegen aan je examenbundel!

Festival (20 pt → 17pt)

1001 0110	10%	{ CC }	1% max 10%	✓ SPEC	1% max 10%
--------------	-----	--------	---------------	--------	---------------

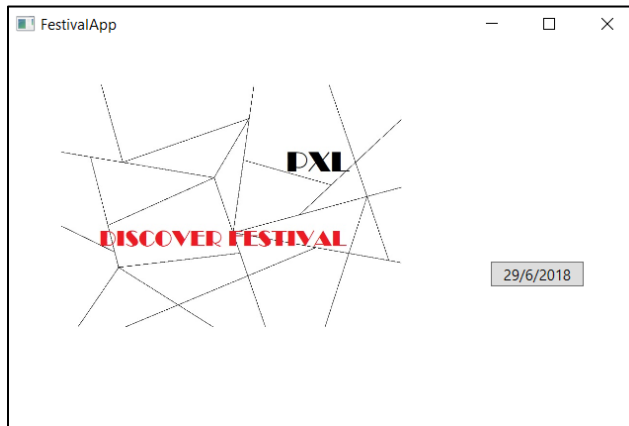
Je maakt een applicatie om de line up per podium van het PXL Discover Festival te tonen en aan te passen.

Implementatiedetails

Bij deze opgave dien je de voorgestelde aanpak **strikt** op te volgen!

De volgende zaken krijg je reeds cadeau in het project:

- De nodige XAML voor de user interface voor het hoofdvenster. (screenshot pagina 5)
Het scherm bevat reeds de volgende gui-componenten:
 - Een Image van het festival
 - 1 knop voor 1 festival dag
- De beperkte XAML voor de user interface voor het StageLineUpWindow, hier moet je nog een menu aan toevoegen. (zie verder in de opgave)
- De nodige XAML voor de user interface voor het NewActWindow.
 - Formulier met de nodige controls om een nieuwe act (band/zanger) toe te voegen aan de affiche van het festival (zie screenshot op p10)
- Een aantal klassen die je nog moet vervolledigen. Gelieve de namen van deze klassen niet te wijzigen.
- 3 tekstbestanden: *Main Stage.txt*, *The Barn.txt* en *The Slope.txt*. Zet deze documenten op je bureaublad.

MainWindow

Zorg ervoor dat het StageLineUpWindow geopend wordt wanneer je op de knop met de datum klikt. Tegelijkertijd zorg je er ook voor dat je het MainWindow verbergt.

Abstracte klasse 'Performer'

Maak een abstracte klasse Performer die ervoor zorgt dat van elke act volgende gegevens (Properties! Zowel get als set!) bijgehouden worden:

- Name (string)
- ReservationNumber (int)
- StartTime (TimeSpan)
- EndTime (TimeSpan)
- TechnicalSupplies (array van int) Het eerste element in array geeft het vereiste aantal stopcontacten weer en het tweede element de minimum oppervlakte op het podium (in m²).
- Rider (generische lijst van strings) (hierin worden de extra niet-financiële eisen van een performer bij gehouden: drank, eten)

Volgende methode moet er zeker inzitten!

- Methode ToString() overschrijven: In de methode ToString() geef je terug: starttijd, eindtijd en de naam van de act volgens volgend formaat "hh:mm-hh:mm actnaam", vb. "13:00-13:45 Rival Sons"

Vul verder aan indien je dit nodig acht in het kader van goed coderen.

Klasse 'Band' en 'Solo'

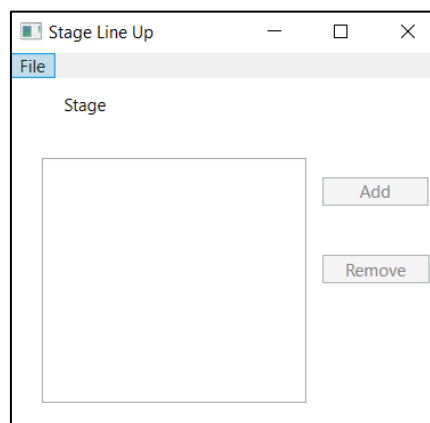
- Vul de 2 bestanden Band.cs en Solo.cs verder aan, zodat de 2 klassen van de abstracte klasse Performer overerven.
 - Band heeft als extra eigenschap
 - MemberCount: het aantal groepsleden (int)
 - Solo heeft als extra eigenschap
 - Type: (“dj”, “zang”, “instrument” ...) (string)
- Maak in beide klassen slechts 1 constructor die voldoende parameters heeft, om alle properties van een waarde te voorzien. Starttijd en Eindtijd worden als string (hh:mm) in de constructor parameters gezet.

Exception klasse FestivalException

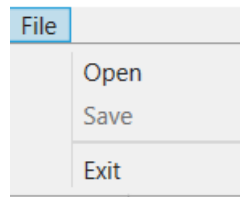
- Maak een correcte exception klasse: FestivalException

StageLineUpWindow

- In het 2^{de} venster StageLineUpWindow (dit verschijnt als je op de datumknop in MainWindow klikt) staat reeds XAML code :
 - een TextBlock om de naam van het podium (stage) te tonen.
 - een Listbox om de optredens te tonen.
 - 2 knoppen: Add en Remove: zorg dat deze knoppen niet actief zijn.

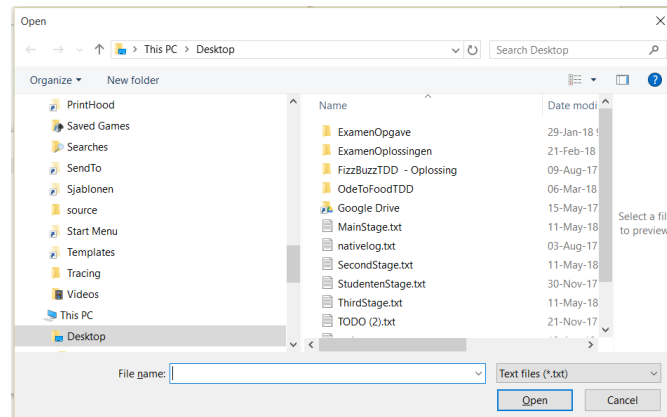


- Menu ‘File’ ontbreekt nog in de code, maak onderstaand menu aan, identiek volgens onderstaand screenshot.



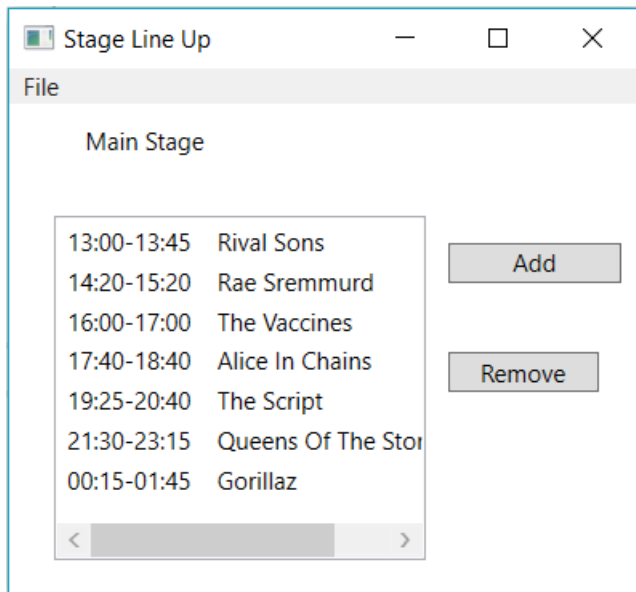
Zorg voor volgende acties:

- Open:
 - Volgend OpenFileDialog venster wordt getoond



- Initiële directory = Desktop
- Enkel txt bestanden worden getoond
- De gebruiker kan een .txt bestand kiezen met de naam van het podium. Toon dus enkel .txt bestanden vertrekkend vanuit het bureaublad van de gebruiker.
- Zet de gegeven .txt documenten op je bureaublad, indien je dat nog niet gedaan hebt.
- Elk gegeven .txt bestand bevat gegevens van alle optredens per podium en ziet er als volgt uit:
 - Letter B of S, die aangeeft of het om een band of solo artiest gaat.
 - Indien B, volgt er een getal, indien S, volgt er een soort. (zie verder)
 - Naam performer
 - Boekingsnummer
 - Startuur optreden

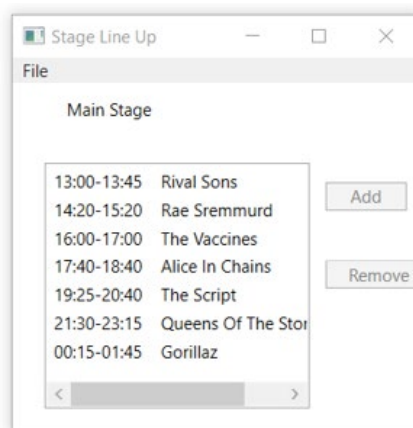
- Einduur optreden
 - 2 technische benodigdheden (aantal stopcontacten, minimum grootte podium in m2)
 - Eisen van de artiest om optreden mogelijk te maken (=Riderlist) (voeding, drank, eigen kleedkamer, extra ruimte, een douche.....) Deze lijst is variabel in lengte.
- Als het bestand wordt ingelezen, zorg je ervoor dat je van de gegevens een juist object maakt (Band of Solo).
 - Deze gegevens zet je in een lijst, die je als basis gebruikt om te tonen.
 - Pas als dit gelukt is mag je de knoppen Add en Remove, alsook het saveMenuItem actief maken.
- Save:
 - Dit mag pas actief worden, nadat gebruiker een bestand geopend heeft.
 - Je schrijft alle gegevens uit de lijst weg naar het laatst ingelezen bestand. (Overschrijven, zo worden alle eventuele wijzigingen zeker opgenomen in het document) Vanaf dat moment zijn de wijzigingen in de lijst definitief.
 - Exit:
 - Hier zorg je ervoor dat het programma op een propere manier gesloten wordt.
- Zorg dat het startvenster getoond wordt na het sluiten van dit venster met het kruisje. Gebruik hiervoor het Window_Closing event.
 - Wanneer het bestand is ingelezen, worden de gegevens als volgt getoond in het StageLineUpWindow.



- Naam van het podium (Stage)
- Naam van elke artiest/band met start- en einduur van optreden

Je kan nu optredens toevoegen en/of verwijderen door middel van de knoppen.

- Door een optreden te selecteren en op de *Remove* knop te klikken, verdwijnt het optreden uit de getoonde lijst.
Opgelet! Wil je het optreden definitief verwijderen, dan gebeurt dit wanneer er hierna op File→Save wordt geklikt. Anders staat het er de volgende keer terug tussen, als je de gegevens van dit podium terug ophaalt.
- Bij het klikken op de *Add* knop wordt het NewActWindow geopend:



- De knop Add (mag pas actief worden, als je al een bestand hebt geopend), opent het NewActWindow, maar het andere venster blijft zichtbaar. Open dit venster niet-modaal. Je zorgt door middel van de methode DisableEvents() dat de knoppen en menuitems geen event kunnen triggeren in je StageLineUpWindow. Zorg er wel voor dat je nog aan de lijst kan.
- Na ingave van de gegevens van het nieuwe optreden, klikt de gebruiker op een knop: Add. Van de gegevens moet een juist object gecreëerd worden (band of solo). Indien er een gegeven niet is ingevuld, ontstaat er een FestivalException met de boodschap “Gelieve alle velden in te vullen!”.
- Je mag ervan uitgaan dat gegevens hier juist worden ingevuld. Controle is dus niet nodig. De 2 getallen bij technical stuff moeten ingegeven worden, met een spatie ertussen. De items van de rider geef je in, gescheiden door een komma. (zie screenshots vorige pagina)
- Je voegt dit object toe aan de lijst van optredens:
 - Je zorgt dat de lijst geordend blijft volgens starttijd van optreden.
 - Er mag geen optreden starten, als het vorige optreden nog niet beëindigd is. Ook dan ontstaat er een FestivalException met de boodschap(“Vorig optreden is nog niet beëindigd”)
 - Ook moet het nieuwe optreden beëindigd zijn, voordat het volgende optreden in de lijst start. Indien het optreden te lang duurt, zorg je voor een FestivalException met de

boodschap("Optreden duurt te lang, het volgende optreden start voor het einde van dit optreden").

- Zorg ervoor dat de bijgewerkte lijst getoond wordt.
 - Als er zich geen probleem heeft voorgedaan en de lijst is bijgewerkt:
 - Je zorgt dat alle knoppen/menulitems in je StageLineUpWindow terug events kunnen triggeren. Dit doe je in de methode `EnableEvents()`.
 - Je sluit dit venster.
 - Denk er ook aan, dat als je deze toevoeging aan dit podium definitief wil maken, je dit ook via `File>Save` moet doen.
-
- Zorg voor het opvangen van `FileNotFoundException` en `IO exceptions` waar deze zich kunnen voordoen.
 - Als er geen controle gevraagd wordt op juiste ingaven, moet je deze controle ook niet doen. (Verlies daar dus geen tijd aan, want dat brengt je geen extra punten op). Je mag ervan uit gaan, dat de gebruiker enkel juiste gegevens invult.

Veel succes !!!

Naam en voornaam:

Kladblad (verplicht mee af te geven met het examen).