**Praktische opdracht Woordzoeker**

**Globale beschrijving:**

Je ontwerpt en bouwt samen in Visual Basic een elektronische woordzoeker.

**Doel van de opdracht:**

* Leren plannen en uitvoeren van een praktische opdracht.
* Leren samenwerken in het plannen en uitvoeren van een praktische opdracht.
* Inzicht verwerven in het proces van ontwerpen en bouwen van een programma.
* Ervaring opdoen met het werken met een moderne hogere programmeertaal.
* Leren “zelfstandig” methoden te leren kennen en gebruiken.

**Eindproducten:**

1. Logboeken van het gehele ontwerp- en bouwproces.
2. Alle solution-bestanden.

**Werkvorm:**

Deze opdracht moet worden gedaan in groepjes van 2 personen, niet meer, niet minder.

**Tijdplanning:**

De totale tijd die je voor dit project nodig hebt schat ik op 8 a 10 klokuren per persoon. Ik ga er dan wel van uit dat je het lesboek programmeren hebt doorgewerkt en redelijk beheerst.

**Omschrijving - probleemstelling**

Jullie bouwen een programma waarbij de 2 spelers om beurten elk binnen een bepaalde tijd van aansluitende letters in een raster van letters woorden maken en het programma controleert of dat woord wel bestaat. Ek goed woord levert een punt op. Het programma houdt de scores van de beide spelers en hun speeltijd bij. Het werkt als volgt:

Bij aanvang van het spel verschijnen in een raster van 4 bij 4 letters. Deze worden ad random gegenereerd door het programma. Net als bij Scrabble komen bepaalde letters vaker voor dan andere. Geef de letters i, e, a en o 3 keer zoveel kans om te verschijnen als de andere letters. De speler klikt op een letter. De aangeklikte letter wordt toegevoegd aan de letters op een letterplankje. Nadat een letter is aangeklikt mogen alleen aanliggende, nog niet aangeklikte letters aanklikbaar zijn. Wanneer de speler denkt een goed woord te hebben gemaakt klikt de speler op een knop en het programma controleert of het woord voorkomt in het standaard woordenboek van Word voor Windows. De speler ontvangt een passend bericht. Als het woord juist is, wordt het woord in een lijst met goede woorden geplaatst en ontvangt de speler 1 punt. Dit wordt toegevoegd aan de score van de speler. In beide gevallen klinkt een passend geluid. De score van beide spelers is zichtbaar. Een goed woord mag natuurlijk maar een keer worden beloond met een punt. Er wordt ook getest op woorden van 0 of 1 letter. Deze zijn niet toegestaan. Reageer passend op foute woorden e.d. Nadat een speler een woord heeft getest gaat de beurt naar de andere speler. Al zijn letters zijn (weer) aanklikbaar. Zichtbaar is wie aan de beurt is. De code moet worden voorzien van ruim voldoende commentaar.

Een voorbeeld van wat ik in gedachten heb is beschikbaar op de pagina po’s van de informatica-site van onze school.

**TIPS**

* Maak voor de verschillende event-handlers eerst PSD’s. Dit helpt. ECHT!!!
* Een werkend voorbeeld van wat ik in gedachten heb is beschikbaar op de informatica site van onze school.
* Gebruik voor het controleren van het woord een instantie van Word voor Windows.
* Gebruik globale variabelen voor de score en de letters. Waar nodig ook voor andere zaken.
* Gebruik eventueel een MultiMedia Control voor het geluid. Een geluid moet eerst worden teruggespoeld voor het opnieuw kan worden afgespeeld.
* Een instantie van Word gebruiken.

'Maak in de public class Form1 het object spellchekr.

 Dim spellChekr As New Object()

'Maak in het Load event van Form1 de variabele spellChekr tot een Word object

 spellChekr = CreateObject("Word.Application")

'Gebruik het spellChecr object om de inhoud van strCK te laten controleren door Word.

If spellChekr.CheckSpelling(strCK) = True Then ...

'Vernietig in het Form1\_Formclosing event het spellChekr object

spellChekr = Nothing

## Logboeken

Jullie houden individueel en als groep logboeken bij van je werkzaamheden. Gebruik hiervoor de (eventueel iets aangepaste) standaard-logboekformulieren. Deze logboeken moeten een goed inzicht geven van het gehele proces. De verzameling logboeken vormt onderdeel van het eindproduct en wordt betrokken bij de beoordeling van de opdracht.

**Nabespreking**

De opdracht wordt **NIET** afgesloten met een nabespreking. Je wordt beoordeeld op de ingeleverde producten.

**Beoordelingsmodel PO Woordzoeker**

* Logboek geeft een goed beeld van het proces. 0 1
* Genereren van letters werkt goed. 0 1
* Vormen van woord werkt goed 0 1
* Controle van woord werkt goed. 0 1
* Bijhouden van scores werkt goed 0 1
* Bijhouden van beurt werkt goed 0 1
* Geluid werkt goed. 0 1
* Voldoende commentaar zodat duidelijk is wat er gebeurt. 0 1
* Efficiënte, mooie en/of slimme implementatie 0 1 2

In principe krijgen leerlingen van dezelfde groep hetzelfde cijfer, tenzij er redenen zijn hiervan af te wijken.

Naam:

Totaal aantal punten:

Ingeleverd op:

Puntenaftrek wegens te laat ingeleverd 0 1

Eindcijfer: