**Verslag Case Study Windows Forms**

Inhoud

[Voorwoord 3](#_Toc122207538)

[Het project 3](#_Toc122207539)

[Voorbereiding 3](#_Toc122207540)

[Testen met windows form 3](#_Toc122207541)

[De uitwerking van de app 4](#_Toc122207542)

[Aanmaken Github-repo 6](#_Toc122207543)

[Aanmaken van de SQlite-database 7](#_Toc122207544)

[Conclusie 12](#_Toc122207545)

[Links 12](#_Toc122207546)

[Bibliografie 12](#_Toc122207547)

# Voorwoord

In dit project ga ik een app maken in Windows Forms met C#.

# Het project

Mijn idee voor dit project is om een handboek-appje voor minecraft te maken. Dan is het bijvoorbeeld mogelijk om info over verschillende monsters te raadplegen, die dan uit een database opgehaald word.

## Voorbereiding

De eerste stap die ik heb ondernomen is uitgebreid research doen. De bronnen die ik daarbij gevonden en geraadpleegd heb, zijn in de bibliografie terug te vinden.

## Testen met windows form

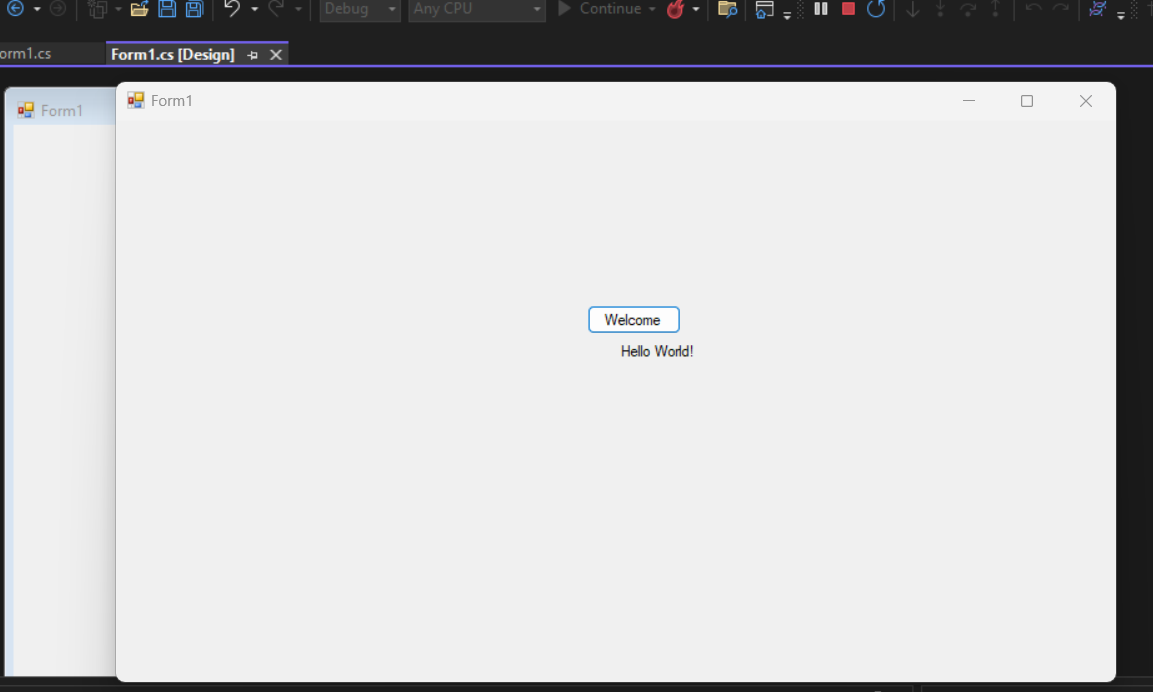
Als eerste ben ik begonnen met een klassiek hello-world experiment, om te testen. Hieronder de screenshots:

Afbeelding met tekst, schermafbeelding, monitor, scherm

Automatisch gegenereerde beschrijving

Afbeelding met tekst, schermafbeelding, monitor, scherm

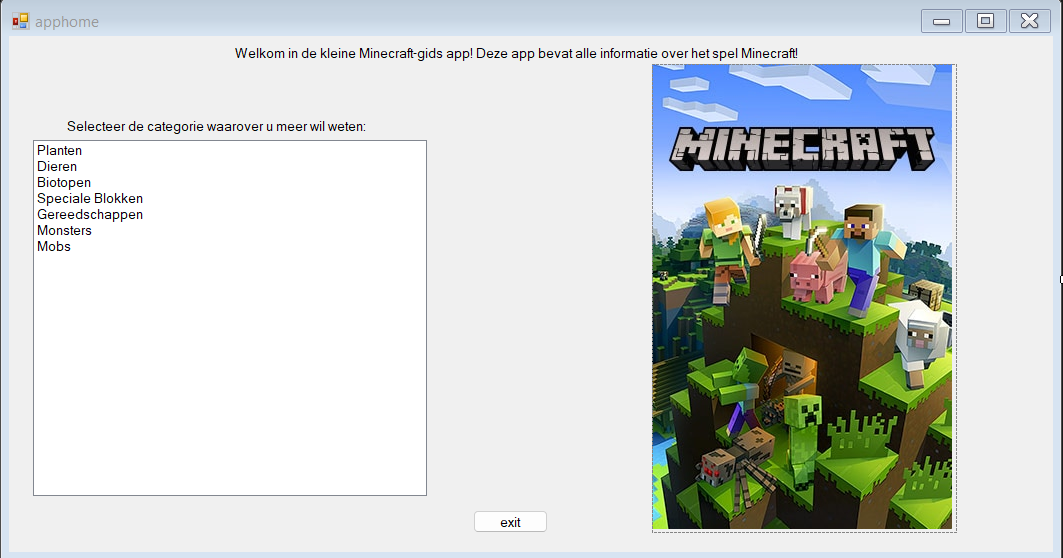
Automatisch gegenereerde beschrijving

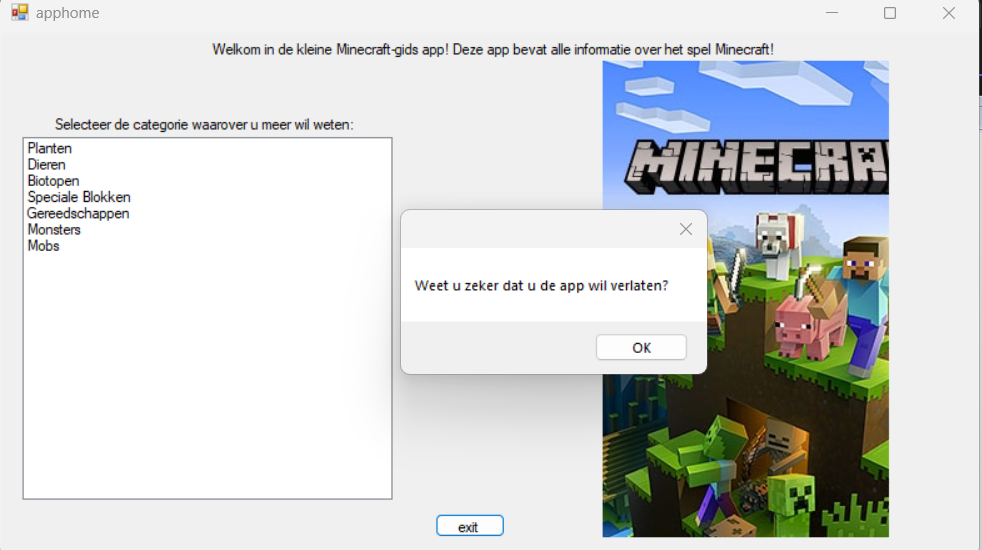


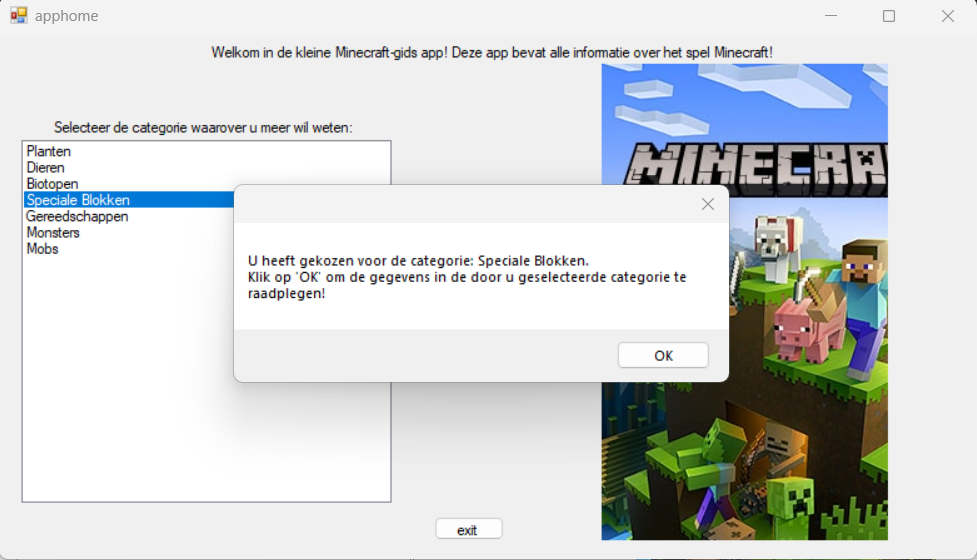
Een simpel experiment met een knop en een label. Als je op de knop drukt, verandert de tekst van het label naar “hello world!”.

## De uitwerking van de app

Nu tijd voor de echte app. Die bestaat in eerste instantie uit een simpel scherm met boodschappen en afbeelding, en een selectiescherm waar de informatie per categorie uit de database kan worden opgehaald.



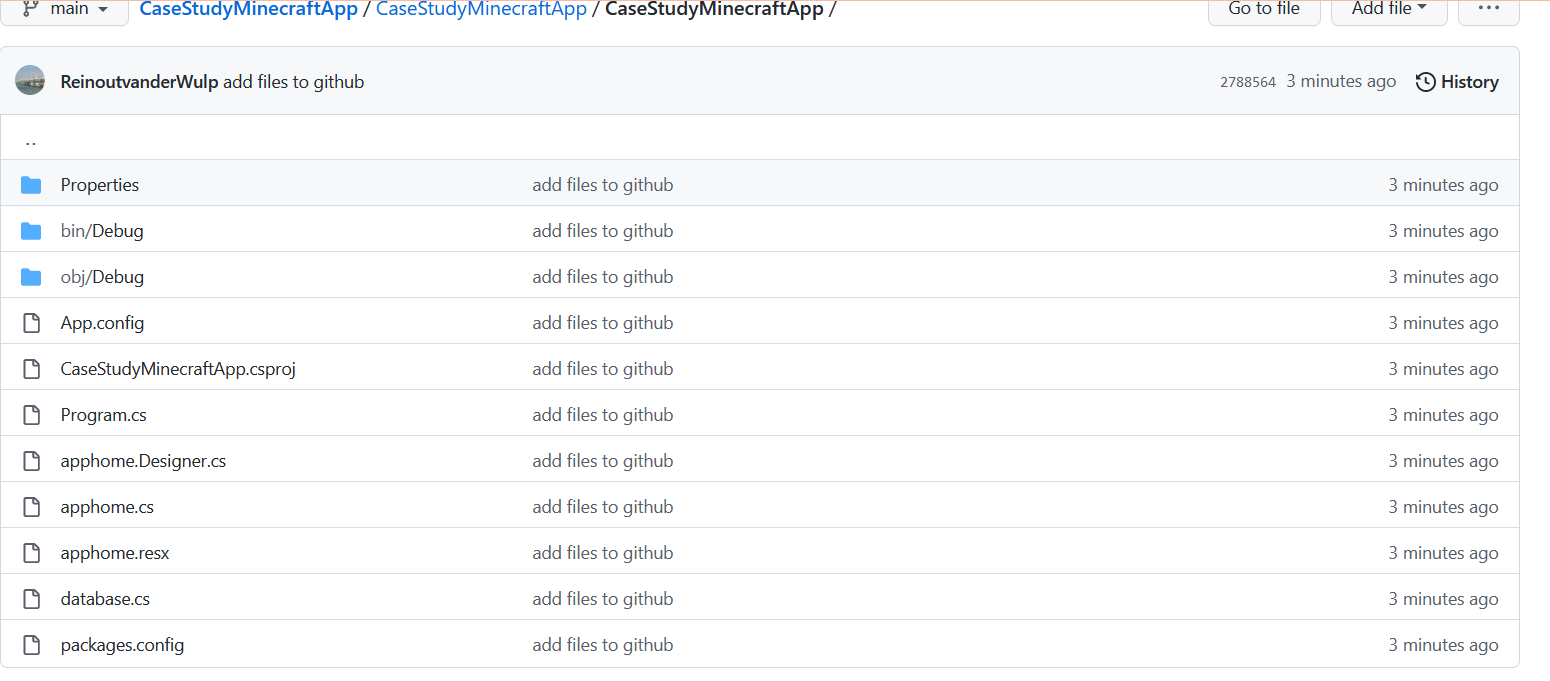




Als er op een van de categorieën geklikt word, is het de bedoeling dat nadat er op OK gedrukt word, de informatie opgehaald word uit de database. De database word in een volgende stap nog in orde gebracht. De knop exit onder in het scherm geeft een standaard melding om daarna de app te kunnen verlaten.

## Aanmaken Github-repo

Nu de app er staat, is het tijd om de files op GitHub te plaatsen, en er een action aan toe te voegen:



Afbeelding met tekst

Automatisch gegenereerde beschrijving

De workflow geeft echter een error, na het oplossen van deze error is dit het resultaat:

**OPLOSSING**

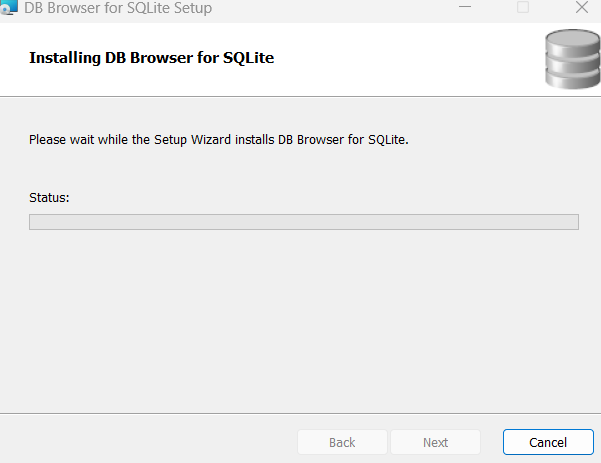
## Aanmaken van de SQlite-database

Nu is het tijd om de SQlite-database aan het project toe te voegen! Eerst de benodigde packages downloaden:

Afbeelding met tekst

Automatisch gegenereerde beschrijving

De volgende stap is de code schrijven om met de database in interactie te gaan, maar eerst maak ik de database aan met een tool die ik in een tutorial gevonden heb, Db browser voor SQLite. De screenshots:



Afbeelding met tekst

Automatisch gegenereerde beschrijving

Het aanmaken van de database met sqlite browser:

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Ik maak nu een C#-klasse voor de connectie, en plaats er een methode in om de waarden te raadplegen:

Afbeelding met tekst, schermafbeelding, monitor, scherm

Automatisch gegenereerde beschrijving

Volgende stap is de database vullen en integreren in de app. De gevulde DB:

**SCREENSHOT**

Nu ga ik methodes voorzien waarmee de app de database kan raadplegen, en kan aanpassen. Hiervoor moet ik soms het appdesign aanpassen, en code veranderen. Alle verschillende functies staan hieronder gedocumenteerd.

*App afsluiten:*

Voor deze functie hoef ik de layout niet aan te passen. De knop is ook al aanwezig, ik hoef alleen met de hulp van een tutorial de code zo aan te passen dat de app daadwerkelijk afgesloten word. Screenshots van de code en de werking:

private void exitbutton\_Click(object sender, EventArgs e)

{

try

{

DialogResult result = MessageBox.Show("Weet u zeker dat u de app wil verlaten?", "confirmation", MessageBoxButtons.OKCancel);

if (result == DialogResult.OK) {

Application.Exit();

}

}

catch (Exception ex) {

MessageBox.Show(ex.Message);

}

}

Afbeelding met tekst

Automatisch gegenereerde beschrijving

*Gegevens uit de database halen per categorie:*

Ook voor deze functie verandert het app-design niet. Verder maak ik een nieuwe klasse database aan, waar ik de mighty ORM importeer:

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Vervolgens haal ik de data uit de database met volgende code:

private void listbox\_SelectedIndexChanged(object sender, EventArgs e)

{

try

{

string text = listbox.GetItemText(listbox.SelectedItem);

DialogResult result = MessageBox.Show("U heeft gekozen voor de categorie: " + text, "confirmation", MessageBoxButtons.OKCancel);

if (result == DialogResult.OK) {

string connection = "Data Source=database.db";

database db = new database(connection);

dynamic query = db.Single("select \* from Categorie where categorieNaam = '" + text + "'");

}

}

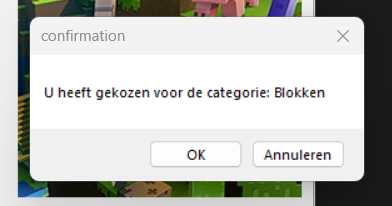
catch (Exception ex) {

MessageBox.Show(ex.Message);

}

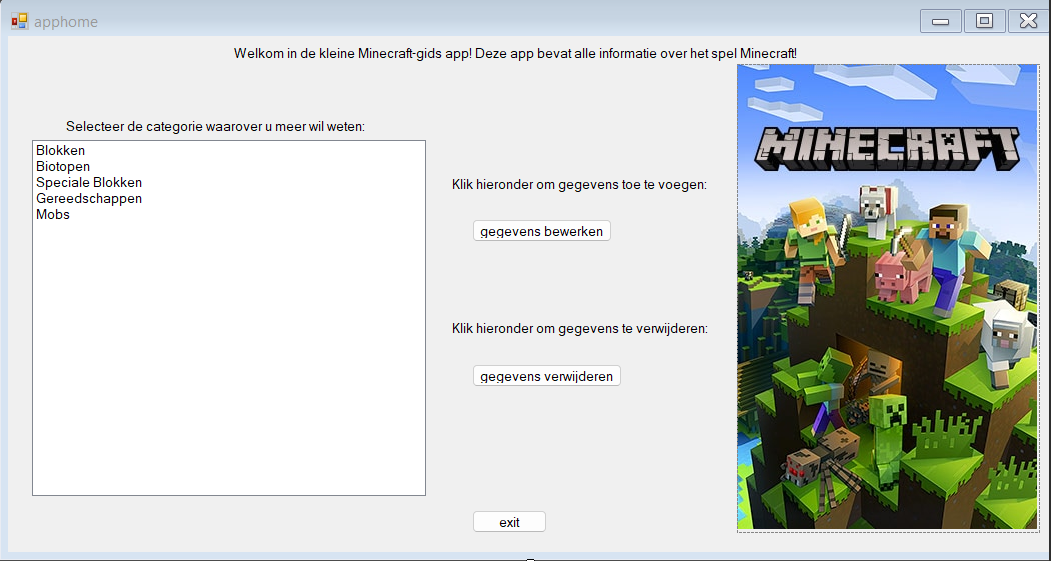
}

Screenshots van de werking:



*Design aanpassen:*

Voor de volgende twee functies pas ik het design aan, en voeg ik twee labels en knoppen toe. Wat deze knoppen doen, is te zien bij de uitleg van de volgende twee functies. Screenshot van de veranderde layout:



*Gegevens toevoegen aan de database:*

*Gegevens verwijderen uit de database:*

# Conclusie

Ik vond het een verassend project om aan te werken.

# Links

Link Github repository:

<https://github.com/ReinoutvanderWulp/CaseStudyMinecraftApp>

Link youtube video:

(link)

# Bibliografie

Hieronder de bronnen die ik gebruikt heb:

Windows form:

<https://learn.microsoft.com/en-us/visualstudio/ide/create-csharp-winform-visual-studio?view=vs-2022>

<https://learn.microsoft.com/en-us/visualstudio/get-started/csharp/tutorial-windows-forms-picture-viewer-layout?view=vs-2022>

<https://www.guru99.com/c-sharp-windows-forms-application.html>

<https://www.youtube.com/watch?v=8ioUk4RajpM>

<https://www.c-sharpcorner.com/UploadFile/c713c3/how-to-exit-in-C-Sharp/>

<https://stackoverflow.com/questions/26459103/how-to-make-exit-confirmation-yes-or-no-on-control-button-using-c-sharp-window-f>

<https://www.makeuseof.com/winforms-input-dialog-box-create-display/>

<https://learn.microsoft.com/en-us/dotnet/api/system.windows.forms.form.dialogresult?view=windowsdesktop-6.0>

Setting up GitHub pipeline:

<https://www.cbtnuggets.com/blog/certifications/microsoft/setting-up-a-ci-pipeline-with-github-actions-in-c-with-examples>

<https://github.blog/2022-02-02-build-ci-cd-pipeline-github-actions-four-steps/>

<https://stackoverflow.com/questions/61188284/github-action-dotnetcore-sln-and-project-in-same-folder>

<https://www.youtube.com/watch?v=VIlDni8-iWM>

<https://learn.microsoft.com/en-us/dotnet/devops/create-dotnet-github-action>

SQlite:

<https://www.c-sharpcorner.com/UploadFile/5d065a/how-to-use-and-connect-sqlite-in-a-window-application/>

<https://www.codeguru.com/dotnet/using-sqlite-in-a-c-application/>

<https://github.com/liviucotfas/ase-windows-applications-programming/blob/master/10%20-%20WinForms%20-%20Databases%20-%20SQLite.md>

<https://learn.microsoft.com/en-us/ef/core/get-started/winforms>

<https://zetcode.com/csharp/sqlite/>

<https://sqlitebrowser.org/>

Overerving:

<https://learn.microsoft.com/en-us/dotnet/desktop/winforms/advanced/how-to-inherit-windows-forms?view=netframeworkdesktop-4.8>

<https://learn.microsoft.com/en-us/dotnet/desktop/winforms/advanced/windows-forms-visual-inheritance?view=netframeworkdesktop-4.8>

<https://stackoverflow.com/questions/2094401/how-to-inherit-system-data-sqlite-in-c-sharp>

<https://medium.com/swlh/introducing-mighty-a-new-net-9bcc7bdd4814>

<https://www.geeksforgeeks.org/c-sharp-inheritance-in-constructors/>

ORM:

<https://stackoverflow.com/questions/3629761/best-orm-for-net-windows-forms-applications>

<https://foxlearn.com/windows-forms/data-access-sql-database-in-csharp-23.html>

<https://supportcenter.devexpress.com/ticket/details/q448226/how-do-i-use-orm-data-model-in-windows-forms>

<http://persianprogrammer.com/Education/Index/ORMs-for-NET-Core>

<https://mightyorm.github.io/Mighty/docs/getting-started>