Software Design Document (SDD)

Eindopdracht Nonogram – Software Development

A screenshot of a crossword puzzle

AI-generated content may be incorrect.

Geschreven door: Ismaël Winterman (1219656) en Brighton van Rouendal (1211945)

Beoordelaars: Dhr. J. Zuur, Dhr. S. Hoeksema, Dhr. B. Tukker.

Datum: 7 April 2025

Versie: 1.00

# Versies

|  |  |  |  |
| --- | --- | --- | --- |
| Versie | Auteur | Datum | Veranderingen |
| 1.00 | Ismaël, Brighton | 7-04-25 | Definitieve document opzetten en afronden |

Inhoudsopgave

[Versies 2](#_Toc194957514)

[1. Inleiding 4](#_Toc194957515)

[1.1. Doel 4](#_Toc194957516)

[1.2. Project definitie 4](#_Toc194957517)

[1.3. Product Visie 4](#_Toc194957518)

[1.4. Scope 4](#_Toc194957519)

[1.5. Stakeholders 5](#_Toc194957520)

[2. Gebruikerseisen 6](#_Toc194957521)

[2.1. Functionele eisen 6](#_Toc194957522)

[2.2. Niet-Functionele eisen 8](#_Toc194957523)

[3. Architectuuroverzicht 9](#_Toc194957524)

[3.1. Technische eisen 9](#_Toc194957525)

[3.2. Technologieën en frameworks 10](#_Toc194957526)

[4. Gedetailleerd Ontwerp 11](#_Toc194957527)

[4.1. Interface-ontwerp 11](#_Toc194957528)

[4.1.1. Mockups 11](#_Toc194957529)

[4.2. Datastromen 20](#_Toc194957530)

[4.3. Datamodel 21](#_Toc194957531)

[4.4. Module-overzicht 23](#_Toc194957532)

[4.4.1. Klassendiagram 23](#_Toc194957533)

[4.4.2. Opzet 23](#_Toc194957534)

[4.4.3. Autorisatie 23](#_Toc194957535)

[4.4.4. Database 24](#_Toc194957536)

[4.4.5. Game 24](#_Toc194957537)

[5. Testing en Validatie 25](#_Toc194957538)

[5.1. Teststrategie 25](#_Toc194957539)

[5.2. Prestatie-eisen 26](#_Toc194957540)

[6. Globale planning 27](#_Toc194957541)

[7. Bijlagen 28](#_Toc194957542)

# Inleiding

## **Doel**

Het doel van dit document is het ondersteunen en overzicht bieden tijdens het traject van dit project waaruit uiteindelijk een functionerend product moet komen. Dit product is een nonogram puzzel.

Dit document is bedoeld voor de stakeholders en toekomstige ontwikkelaars die het project willen voortzetten. Voor nu valt dit personeel hieronder: de ontwikkelaars, de docenten als beoordelaars en uiteindelijk gebruikers.

## Project definitie

Tijdens dit project is het product een WinForms applicatie waarin een gebruiker een account moet kunnen aanmaken, kan inloggen, tussen light & dark mode schakelen, een nonogram kunnen spelen, oplossen, later hervatten en het scoreboard kunnen bekijken. De applicatie houdt dus ook een scoreboard bij per moeilijkheidsgraad.

## Product Visie

Dit product wordt gemaakt via WinForms en zal de ingebouwde toolbox gebruiken om deze tot leven brengen. De uitzondering is bij het tekenen van het nonogram zelf, hierbij wordt een eigen geprogrammeerde functie gebruikt om deze weer te geven.

De waarde die deze software ons biedt is een leerproces waarin wij C#, .net 8.0 framework en WinForms op de technologie kant kunnen leren toepassen en op een agile/scrum matige wijzen werken. Deze software maken we voor de eindopdracht van periode 3 tijdens de module software development.

## Scope

Voor dit project zijn er bepaalde grenzen die wij voor onszelf hebben gelegd. Hieronder zullen wij hier dieper op in gaan.

Als eerste, na een voorafgaand onderzoek met betrekking tot het maken van een logische rendering solver voor hints en aanwijzingen en backtracking-oplosser hebben wij na een periode van pogingen om dit beide te begrijpen en zelf te implementeren ervoor gekozen om deze onderdelen niet toe te passen, omdat dit buiten ons begrip ligt. Hier kunnen wij later buiten het traject om op terugkomen.

Daarnaast hebben wij er ook voor gekozen om geen gebruik te maken van het loggen van data. De enige toepassing die wij hiervoor konden bedenken die nut bood voor de gebruiker was het loggen van elke zet die gemaakt is. Dit vonden wij niet relevant bij ons product aangezien het niet om een schaakspel ging.

Vervolgens het genereren van de nonogrammen. Een nonogram in onze applicatie wordt dynamische gegenereerd en krijgt daarbij een bijbehorend nummer(seed). Hiermee kan dezelfde puzzel, waar nodig, altijd opnieuw aangemaakt worden. Er is geen gebruik gemaakt van het draaien/spiegelen van de applicatie, omdat erop de achtergrond met de mogelijk van meerdere oplossingen rekening wordt gehouden.

Verder, multithreading, de applicatie maakt gebruik van een timer en een stopwatch om de speeltijd bij te houden. Deze draaien op hun eigen threads, waardoor de tijd nog steeds doorloopt wanneer de thread voor het tekenen van de nonogram gestopt wordt als er door de gebruiker op de afsluitknop in de titelbalk geklikt.

Qua standaardinstellingen draait de applicatie zonder het thema “dark-mode” te activeren. De gebruiker kan via een keuzevakje in de navigatiebalk aangeven dat zij tussen deze twee opties willen schakelen.

Wanneer het aankomt tot de inhoud van applicatie zal deze de gebruiker de volgende mogelijkheid bieden:

1. In de navigatiebalk knoppen om terug te gaan naar het hoofdmenu, het wisselen van thema’s, hun gebruikersnaam zien en uit te loggen.
2. Een account aan maken, inloggen en uitloggen.
3. Een nieuw spel starten, een onvoltooid spel automatisch opslaan en een oud spel hervatten.
4. De moeilijkheidsgraad aanpassen d.m.v. het veranderen van de grootte van het spel.
5. Tijdens het spel om hints vragen en zien hoeveel hints er zijn opgevraagd.
6. De oplossing, seed en speeltijd van het huidige spel inzien.
7. Het maken en ongedaan maken van zetten.
8. Het markeren van velden met een X als mentaal hulpmiddel.
9. Een scoreboard bijhouden van alle afgemaakte spellen en hiervan de top tien te laten zien gecategoriseerd op de spelgroottes.

Als laatste, de context waarin het system zich opereert. De applicatie wordt niet online gehost op een website en zal dus lokaal op de gebruikers computer zonder verbinding met het internet draaien.

## Stakeholders

Ontwikkelaars: Ismael Winterman, Brighton van Rouendal

Docenten: Dhr. J. Zuur, Dhr. S. Hoeksema, Dhr. B. Tukker.

Eindgebruikers

# Gebruikerseisen

## Functionele eisen

\* betekent dat je ingelogd moet zijn om deze pagina te bezoeken.

\*\* betekent dat je niet ingelogd moet zijn om deze pagina te bezoeken

1. **Globale navigatiebar**
   1. Een gebruiker moet een knop hebben om terug te kunnen gaan naar de menupagina.
   2. Een gebruiker moet via een selectievakje kunnen aangeven dat zij het thema “dark-mode” aan of uit willen schakelen.
   3. Een gebruiker moet zijn gebruikersnaam zien als deze is ingelogd.
   4. Een gebruiker moet een knop hebben om uit te kunnen loggen.
2. **Menupagina**
   1. Een gebruiker moet een knop hebben om aan een spel te beginnen.
      1. Een pop-up tonen als hierop geklikt wordt en er is niet ingelogd.
   2. Een gebruiker moet een knop hebben om te kunnen inloggen.
      1. Een pop-up tonen als hierop geklikt wordt en er is al ingelogd.
   3. Een gebruiker moet een knop hebben om naar het scoreboard toe te gaan.
3. **Loginpagina\*\***
   1. Een gebruiker moet een veld hebben om zijn gebruikersnaam in te vullen.
   2. Een gebruiker moet een veld hebben om zijn wachtwoord in te vullen.
   3. Een gebruiker moet een knop hebben om in te loggen.
      1. Een pop-up tonen als de gebruikersnaam niet bestaat.
      2. Een pop-up tonen als het wachtwoord onjuist is.
      3. Een pop-up tonen als de gebruiker succesvol is ingelogd.
      4. Een gebruiker wordt doorverwezen naar de menupagina na succesvol in te loggen.
   4. Een gebruiker moet een knop hebben om doorverwezen te worden naar de registratiepagina.
4. **Registratiepagina\*\***
   1. Een gebruiker moet een veld hebben om zijn gebruikersnaam in te vullen.
   2. Een gebruiker moet een veld hebben om zijn wachtwoord in te vullen.
   3. Een gebruiker moet een veld hebben om zijn wachtwoord te bevestigen.
   4. Een gebruiker moet een knop hebben om te registreren.
      1. Een pop-up tonen als de gebruikersnaam al bestaat.
      2. Een pop-up tonen als de ingevoerde wachtwoorden niet overeenkomen.
      3. Een pop-up tonen als de gebruiker succesvol is geregistreerd.
      4. Een gebruiker wordt doorverwezen naar de loginpagina na een succesvolle registratie.
   5. Een gebruiker moet een knop hebben om doorverwezen te worden naar de loginpagina.
5. **Pick your game pagina \***
   1. Een gebruiker moet ingelogd zijn om aan een spel te beginnen.
   2. Een gebruiker moet een knop hebben om een nieuw spel te starten.
   3. Een gebruiker moet een knop hebben om een onvoltooid spel te hervatten.
      1. Een pop-up tonen als er geen spellen zijn om te hervatten.
6. **Nonogrampagina \***
   1. Een gebruiker moet een veld hebben om de grootte(moeilijkheidsgraad) van het spel te kunnen selecteren.
   2. Een gebruiker moet een knop hebben om het spel te starten.
   3. Een gebruiker moet de mattrijs van het nonogram kunnen zien.
   4. Een gebruiker moet de hints die bij de rijen en kolommen horen kunnen zien.
   5. Een gebruiker moet binnen het spel verschillende zetten kunnen doen.
      1. Met linkermuisknop een vakje zwart maken om aan te geven dat deze correct is.
      2. Met rechtermuisknop een vakje kunnen markeren met een rode X als een mentaal hulpmiddel.
      3. Bovenstaande zetten ongedaan kunnen maken.
   6. Een gebruiker moet een knop hebben om een hint op te vragen
   7. Een gebruiker moet het aantal opgevraagde hints kunnen zien
   8. Een gebruiker moet kunnen zien hoe lang zij met het spel bezig zijn.
   9. Een gebruiker moet een knop hebben om de oplossen van het spel te tonen.
   10. Een pop-up tonen als de gebruiker het spel heeft opgelost.
7. **Scoreboardpagina**
   1. Een gebruiker moet door middel van een keuzemenu kunnen aangeven van welke moeilijkheidsgraad zij de top 10 scores willen bekijken.

## Niet-Functionele eisen

1. **Flexibel en Agile ontwerp**
   1. De applicatie wordt volgens agile principes met een flexibel ontwerp dat eenvoudig aanpasbaar is ontwikkeld.
2. **Codekwaliteit en consistentie**
   1. De applicatie voldoet aan Clean Code en bij voorkeur aan SOLID principes.
   2. De applicatie heeft een consistent design tussen de schermen en interacties.
3. **Performance en stabiliteit**
   1. De applicatie werkt vlot op alle puzzelgroottes.
   2. De applicatie is robuust opgezet tegen incorrecte invoer.
4. **Gebruikersvriendelijkheid**
   1. De applicatie heeft een intuïtieve indeling en soepele gebruikerservaring.

# Architectuuroverzicht

## Technische eisen

1. De applicatie wordt ontwikkeld in C# m.b.v. WinForms.
2. De applicatie volgt een modulaire opzet.
   1. Verantwoordelijk voor het hergebruik van systeemonderdelen zoals de database voor de gebruikers.
3. De applicatie maakt gebruik van JSON als een database.
   1. Verantwoordelijk voor het opslaan van de gebruikers en de staat van hun gespeelde spellen.
4. De applicatie moet wachtwoorden veilig opslaan.
   1. Verantwoordelijk voor het afschermen van wachtwoorden in blote tekst. Dit worden gedaan via PBKDF2 sha512 encryptie.
5. De applicatie moet via het versiebeheer van Git werken.
   1. Verantwoordelijk voor het gemakkelijk en overzichtelijk samenwerken en dient als een back-up van het project indien een rollback nodig is.
6. De applicatie moet makkelijk te navigeren zijn.
   1. Verantwoordelijk voor het intuïtieve gebruik van de applicatie.
7. De applicatie moet pagina’s kunnen afschermen
   1. De applicatie moet pagina’s die alleen als ingelogde gebruiker navigeerbaar zijn niet kunnen laten zien aan gast gebruikers.
   2. De applicatie moet pagina’s die alleen als gast gebruiker navigeerbaar zijn niet kunnen laten zien aan ingelogde gebruikers.
8. De applicatie moet de gebruikers input op het nonogram in overweging nemen bij het besluit of een spel opgelost is.
   1. Verantwoordelijk voor het nakijken of een gebruikers oplossing overeenkomt met de aangegeven hints i.p.v. de gegenereerde oplossing, zodat bij een spel met meerdere oplossingen deze allemaal correct zijn.
9. De applicatie wordt op meerdere manier getest.
   1. Unittests zullen verantwoordelijk zijn voor de back-end van de applicatie.
   2. Integratietesten zullen verantwoordelijk zijn voor de samenhang van de systemen binnen de applicatie.
   3. Acceptatietesten zullen verantwoordelijk zijn voor het testen van de functies die op de gebruikersinterface interactie uitvoeren.
10. De applicatie moet foutloos draaien.
    1. Verantwoordelijk voor een prettige gebruikerservaring.

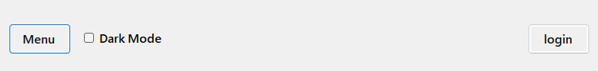
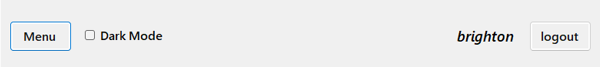
## Technologieën en frameworks

1. Programmeertalen
   * C# versie 12
2. Database
   * JSON
3. Frameworks
   * Microsoft .NET versie 8.0
   * WinForms
4. IDE
   * Visual Studio 2022

# Gedetailleerd Ontwerp

## Interface-ontwerp

### Mockups

**Navigatiebalk**

Elk scherm heeft dezelfde navigatiebalk met de volgende knoppen en teksten. Een knop met de tekst “Menu” die naar dit scherm lijdt, dark mode-selectie box om het thema aan te passen naar een donkere variant, een label met de naam van de ingelogde gebruiker (als deze is ingelogd) en een knop met de tekst “login” of “logout” om naar de inlog pagina te gaan of om uitloggen waardoor je naar dit scherm terug wordt verstuurd.

**Menu (Hoofdpagina)**A screenshot of a computer

AI-generated content may be incorrect.

Menu scherm inhoud: Een grote label met de tekst “Menu” om duidelijk aan te geven dat dit de menu view is, drie knoppen met de teksten “Play”, “Login” en “Scoreboard”. De knop play moet naar de view (pagina) van Spel kiezer (Game Chooser) verwijzen, de knop login moet naar de view voor het inlog scherm verwijzen en de knop scoreboard laat de view scoreboard zien.

**Scoreboard**A screenshot of a computer

AI-generated content may be incorrect.

Scoreboard scherm, dit scherm is altijd bereikbaar (beide als gast en gebruiker) De hoofdtekst van deze view is “Scoreboard”. Hieronder zit een keuzemenu om de scoreboard geschiedenis van een bepaalde grootte te selecteren. Hiervan worden alleen de top 10 weergegeven. De tekst voor een scoreboard regel heeft de volgende informatie: De index van die score, gebruikersnaam van de gezette score, de aantal hints dat de gebruiker heeft gebruikt en de datum van wanneer het nonogram was voltooid was.

**Login**  
A screenshot of a computer

AI-generated content may be incorrect.

Login scherm, dit scherm is alleen bereikbaar als gast en niet als je al bent ingelogd. De hoofdtekst voor deze view is “Login”. Vervolgens twee input velden waar tekst in geschreven kan worden, de eerste is voor de naam van de gebruiker en de tweede voor het wachtwoord. Deze twee invoervelden hebben de volgende placeholder teksten: “Name” en “Password”. Vervolgens een knop met de tekst “Login” om de gebruiker in te loggen (Verifiëren van gegevens) en een knop “Register” om naar de register view (account aanmaken) te gaan. Als na het invullen van gegevens de ingevulde naam bij “Name” niet overeenkomt met een naam in de database, dan komt er een pop-up dat deze gebruiker niet bestaat. Wanneer een wachtwoord is ingevoerd die niet pas bij de aangegeven gebruiker, dan komt er een pop-up dat het wachtwoord onjuist is.

**Register**A screenshot of a computer

AI-generated content may be incorrect.

Register scherm, het account aanmaak scherm, dit scherm is alleen bereikbaar als gast en niet als je al bent ingelogd. De hoofdtekst van deze view is “Register” met vervolgens drie invoervelden waar tekst in gevuld kan worden. Het eerste invoerveld “Name” is voor een unieke naam die gebruikt wordt voor het identificeren van de gebruiker. De volgende twee “Password” en “Confirm Password” zijn voor het wachtwoord dat deze gebruiker zal moet gebruiken bij het inloggen. Als er een account al bestaat met een naam die is ingevuld bij “Name”, dan zal er een pop-up komen die aangeeft dat deze naam als in gebruik is. Ook als de “Password” en “Confirm Password” niet overeenkomen, dan zal er een pop-up komen dat deze wachtwoorden niet overeenkomen. Vervolgens zijn er twee knoppen, knop één, “Register”, zal de verificatie en uiteindelijke registratie van de nieuwe account uitvoeren en zich doorsturen naar de view van Login, knop twee, “Login”, verwijst zichzelf naar de view van Login voor gebruikers die al een account hebben.

**Play (Kies modus)**A screenshot of a computer

AI-generated content may be incorrect.

Play scherm (modus kiezen), dit scherm is alleen bereikbaar als een gebruiker is ingelogd. De hoofdtekst is “Pick your game”, met twee knoppen: “New Game” verwijst naar het nonogram scherm met nog geen ingeladen nonogram, “Load Game” verwijst naar het “Load game” scherm waar een gebruiker een lijst van nonogrammen heeft die nog niet zijn opgelost kan kiezen. Als een gebruiker geen nonogrammen heeft om verder te spelen wordt een pop-up weergegeven die zegt dat de gebruiker geen nonogrammen heeft om te laden.

**Load Game (laad oude nonogrammen) TODO: new image**A screenshot of a computer

AI-generated content may be incorrect.

Load game scherm, dit scherm laat nonogrammen zien die een gebruiker nog niet heeft opgelost. Scherm is alleen bereikbaar als een gebruiker is ingelogd. De hoofdtekst voor dit scherm is “Load Game” met een keuzemenu van nog geen opgeloste nonogrammen. In het keuzemenu zijn de spellen aangegeven door: Grootte van nonogram, aantal tijd gespeeld op de nonogram en de datum van wanneer de nonogram is gestart. Daaronder staat een knop om vervolgens naar de game scherm te gaan met de gekozen ingeladen nonogram.

**Game scherm (Nonogram scherm)**A screenshot of a computer

AI-generated content may be incorrect. **Image 1**A screenshot of a game

AI-generated content may be incorrect. **Image 2**

Game scherm, view waarop de nonogram wordt gespeeld. Alleen bereikbaar als de gebruiker is ingelogd. Image 1 laat zien als de gebruiker “New Game” heeft gedrukt op de “**Load Game**” view waar de gebruiker nog geen spel ziet. Image 2 laat zien als een gebruiker of op de knop “Change” heeft gedrukt of een spel heeft ingeladen.

Op de onderste balk onder de game zelf zijn er meerde knoppen en acties die kunnen worden uitgevoerd. Sectie 1 (“Change Size”) heeft een label met tekst “Change Size” met daar onder een nummer input die gelimiteerd is met een minimum van 5 en een maximum van 20. Dit geeft aan welke grootte een nieuwe nonogram wordt als er op de knop “Change” wordt gedrukt. De knop “Change” herstart alle informatie en creëert een nieuw nonogram met het aangegeven formaat.

Sectie 2 (“Solution”/ “Hint”) heeft twee knoppen: “Solution” laat zien wat er verkeerd is op het scherm via een rood vakje, wat goed is op het scherm via een groen vakje en wat nog ingevuld moet worden via een blauw vakje. De knop “hint” geeft een pop-up dat aangeeft op welke rij en column een vakje moet worden gemarkeerd of on gemarkeerd.

Sectie 3 (informatie over het spel) “Hints” geeft aan hoeveel hints zijn gebruikt tijden het spel, meest rechter bovenste tekst is de “seed”, een herbruikbaar nummer om hetzelfde nonogram te generen en daar onder staat de speel tijd van het huidige nonogram.

## Datastromen

A diagram of a company

AI-generated content may be incorrect.

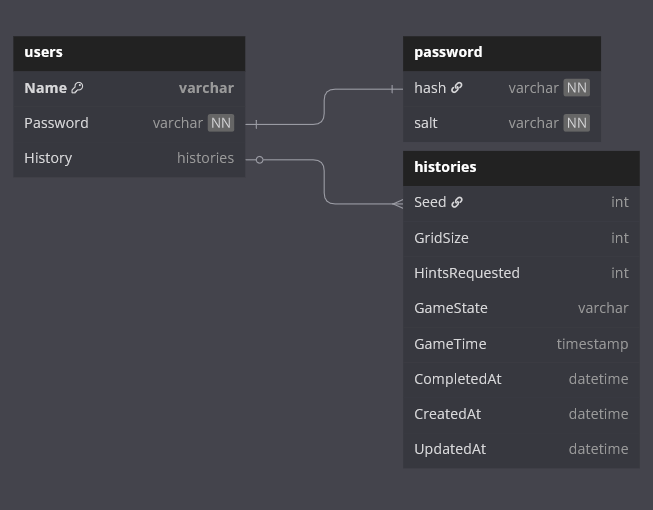
Het proces om een spel te spelen.

Start, wanneer er op de “Play” knop wordt gedrukt op de menu pagina moet een gebruiker zijn ingelogd ander krijgt deze een pop-up te zien dat de pagina een ingelogde gebruiker nodig heeft. Is een gebruiker niet ingelogd kan deze naar de inlog pagina om dat in te loggen op hun account, heeft een gebruiker nog geen account dan kan er genavigeerd worden naar de registratie pagina. Wanneer een gebruiker is ingelogd krijgen hun de mogelijk heit om op de “Play” knop te drukken en wordt dan doorverwezen naar de “Choose Mode” pagina waar de gebruiker een keuze heeft uit het start van een nieuw spel of het inladen van een nog niet afgerond spel. Wanneer de gebruiker heeft gekozen voor nieuw spel dan wordt hij doorverwezen naar de “Game” view waar hij een leeg vlak met knoppen te zien krijgt om een keuzen van nonogram grootte te kiezen, als de gebruiker een grootte heeft gekozen en op “change” drukt wordt er een spel gestart. Kiest de gebruiker voor “Load Game” (laad spel) en heeft de gebruiker geen onvoltooide spellen wordt krijgt hij een pop-up met de tekst dat er geen spellen om uit te kiezen zijn. Heeft de gebruiker wel spellen om uit te kiezen wordt de gebruiker dan naar een geschiedenis pagina verwezen waar onvoltooide spellen worden getoond in een lijst om dan een keuze van te maken. Heeft de gebruiker een keuze gemaakt wordt hij door verstuurd naar de “Game” pagina met zijn gekozen spel ingeladen en meteen gestart vanaf waar hij was.

Start, wanner er op de “Scoreboard” knop wordt gedrukt op de menu pagina wordt de gebruiker doorverwezen naar de “Scoreboard” pagina waar op de pagina een keuzemenu staat met nonogram groottes van 5x5 tot 20x20. De gebruiker kan een keuze maken van een van deze opties om de scoreboard te zien van de gekozen nonogram grootte.

## Datamodel

Definieer de database-structuur en de belangrijkste entiteiten met ER-diagrammen.



De gekozen database is JSON, maar hierboven is een beeld van wat de relatie zou zijn tussen de modellen.

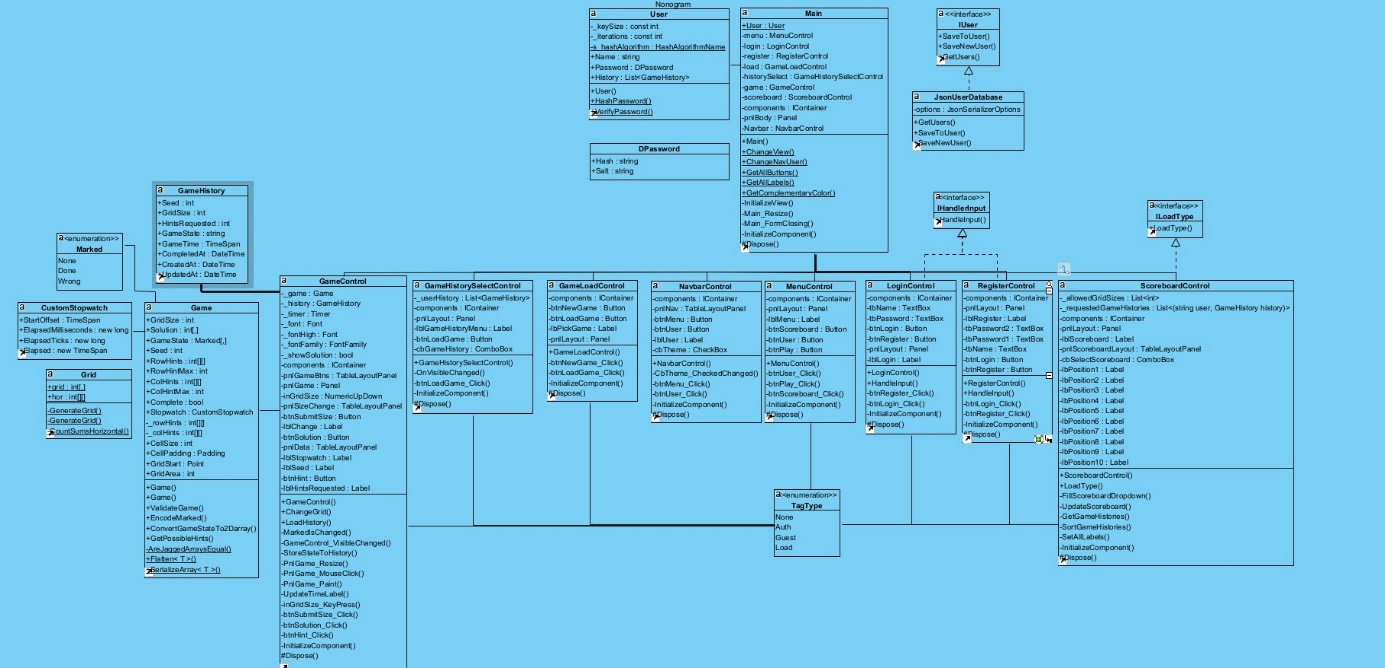
**JSON opmaak**

Het JSON bestand zal een lijst van users hebben elk met een password object die een hash en salt opslaat, het user object zal ook een naam die uniek moet zijn en een lijst van spel geschiedenis (“GameHistory”) dat de seed, gridsize (grootte van de nonogram), aantal hints dat zijn gevraagd (“HintsRequested”), de door de gebruiker gemarkeerde cellen(“GameState”), speeltijd (“GameTime”), wanneer het nonogram was voltooid (“CompletedAt”) dit geeft ook aan dat het spel was voltooid aan andere code, wanneer het nonogram was gecreëerd (“CreatedAt”) en wanner hij als laats was aangepast (“UpdatedAt”).

Dit komt zo er uit te zien in een JSON-bestand:  
A screenshot of a computer program

AI-generated content may be incorrect.

## **Module-overzicht**



### Klassendiagram

Dit is het klassendiagram van ons project. De lijnen geven aan welke klassen wat bevatten en moeten van boven naar beneden gelezen worden. Bijvoorbeeld “Main” bevat alle control die naar zichzelf toe lijden. Dit is gedaan omdat het onduidelijk was wie welke soortlijnen moest hebben en de class designer in Visual Studio 2022 geen enkele lijn trok tussen de klassen, excuses hiervoor.

In het project zelf en als bijlage staan beide de screenshot in zijn volle resolutie en het projectbestand voor Visual Paradigm waarin dit klassendiagram is gemaakt.

### Opzet

De applicatie bestaat uit één form (“Main” in het klassendiagram). Deze is opgebouwd uit twee onderdelen in de designer: De navigatiebalk die een gedeelte van het boven scherm inneemt en daaronder een panel die de rest van de form inneemt met de naam “pnlBody”. Het paneel wordt als een hoofdkader gebruikt waarin alle verschillende views (in het klassendiagram hebben die een naam die eindigt op Control) weergegeven worden door de zichtbaarheid van alleen de view die gezien moet worden aan te zetten.

### Autorisatie

Binnen de applicatie zijn de verschillende views getagd met verschillende toegangsrechten. Dit zijn geen niveaus. Buiten “none” om moet je daadwerkelijk dat niveau hebben om op de pagina te komen, i.e., een view met de tag “Guest” kan alleen maar bezocht worden door een gast gebruiker en niet door iemand die ingelogd is. Dit is gedaan om ervoor te zorgen dat de gebruiker niet opnieuw kan inloggen terwijl hij is ingelogd. De user wordt statisch bijgehouden in de form.

### Database

De applicatie gebruikt een JSON-bestand als database om de data van de gebruikers bij te houden. Hiervoor is een interface (“IUser aangemaakt”) die geïmplementeerd wordt door de JsonUserDatabase klasse. Hierin wordt ook aangegeven welke opties het JSON-bestand moet gebruiken om de data in een overzichtelijke manier op te slaan.

### Game

De game klasse maakt gebruikt van de grid klasse. Hierin zit de logica van het genereren van de grid en de hints. Er is een aangepaste stopwatch klasse waarin een override wordt gedaan op de ElapsedMilliseconds property, zodat de timer van het spel op de ingeladen speeltijd gezet kan worden. Dit was niet mogelijk met de standaard stopwatch klasse.

Verder wordt er in de game klasse de validatie van het spel, de mogelijke overige hints om naar gebruiker te tonen en de logica die nodig is om de gemarkeerde cellen van een mattrijs om te zetten naar een reeks van getallen als tekst.

# Testing en Validatie

## Teststrategie

1. **User tests** (Unit tests)
   1. Creatie van een gebruiker
   2. In lezen van gebruikers
   3. In lezen van gebruikers als er geen bestand is
   4. Updaten van een gebruiker met nieuw spel geschiedenis
   5. Updaten van een niet bestaande gebruiker met nieuw spel geschiedenis
2. **Game tests** (Integratietests)
   1. Creëer spel zonder seed\* waarden.
   2. Creëer spel met seed\* waarden.
   3. Komen hints overeen met verwachten resultaat.
   4. Invalide oplossing van game validatie.
   5. Valide oplossing game validatie.
   6. Gemarkeerde cellen opslaan in database (User – Unit test)
3. **Form user status** (Acceptatietests)
   1. Verander user op form scherm
   2. Geen user ingelogd
   3. Ingelogde user

\***seed** is een waarde dat hergebruikt kan worden om dezelfde logica te kunnen hergebruiken. Dit is voornamelijk gebruikt voor methoden of programma’s dat willekeurig (random) moet(en) gedragen. Met een seed waarden kan dit willekeurig gedrag opnieuw uitgespeeld worden.

## Prestatie-eisen

1. **User tests**
   1. Een gebruiker wordt opgeslagen in het JSON-bestand
   2. Alle opgeslagen gebruikers worden opgehaald.
   3. Als een bestand niet bestaat moet een legen lijst worden teruggegeven.
   4. In bestaand gebruiker die opgeslagen is in het JSON-bestand kan een spel geschiedenis aan zich toegevoegd krijgen.
   5. Als een gebruiker niet bestaat maar spel geschiedenis aan deze gebruiker wordt toegevoegd in het JSON-bestand dan wordt deze niet opgeslagen.
2. **Game Tests**
   1. Een willekeurige matrijs wordt gegenereerd voor het spel. Alle hints worden gemaakt voor dit matrijs (Rijen en kolommen krijgen de juiste hints).
   2. Een spel wordt gemaakt met dezelfde matrijs via een seed waarden.
   3. Verwachten hint resultaten worden na gecheckt dat deze hetzelfde zijn.
   4. Een niet opgelost spel dat wordt gevalideerd, wordt op niet afgerond gezet als deze onjuist is.
   5. Een opgelost matrijs die wordt gevalideerd wordt op afgerond gezet.
   6. Willekeurige cellen worden gemarkeerd en daarna ge- “encode” om deze als tekst op te slaan in een gebruikers spel geschiedenis in het JSON-bestand.
3. **Form user status**
   1. De label van de gebruikersnaam wordt gezet naar de huidige static user username variabel.
   2. De label is niet zichtbaar als er geen gebruiker in de static user variable zit (null value).
   3. User die wordt ingelogd krijgt de logout knop te zien in plaats van login

# Globale planning

**Week 1 – Week 5:** Oriëntatie & C# omgeving leren kennen. Probeer met de kennis die wordt opgedaan tijdens de lessen op verschillende manieren het nonogram op te bouwen. Denk hierbij voornamelijk aan het spel zelf spelend krijgen en onderzoek doen naar de logische rendering solver voor hints en aanwijzingen en backtracking-oplosser om deze proberen te begrijpen en implementeren.

**Week 6:**

* Features
  + Implementatie van die views (wireframes/mock-ups).
  + Registratie, inloggen, uitloggen van de user.
  + Creatie van het mattrijs en de hints voor de mattrijs.
  + Validatie van meerdere mogelijke oplossingen.
  + Teken logica van de mattrijs en hints.
  + Afhandeling van gebruikers input op cellen.
  + Veranderen van mattrijs grootte.
  + De speeltijd van een spel bijhouden.
  + Afbakenen van schermen op basis van gast of gebruiker.
  + Navigatiebalk.
* Tests schrijven voor de user handeling naar de database.

**Week 7:**

* Laatste features implementeren.
  + Hints opvragen van het spel om aan de gebruiker te tonen.
  + Uitladen en sla de staat van het spel op wanneer de view wordt veranderd.
  + Inladen van een spel met de laatste gamestate om deze te hervatten.
  + Sla het spel op na elke zet.
  + Dark-mode thema.
  + Een beter scoreboard.
  + Stopwatch in de game class.
  + UpdatedAt toevoegen aan GameHistory.
* Testen schrijven waar deze nog ontbreken.
* Definitieve SDD opzetten.
* Naming conventions en SOLID principes nalopen.
* Comments toevoegen waar nog nodig.
* Errors uit het programma halen.

# Bijlagen

Zie bijlagen folder voor het volgende:

1. Epics & Userstories document.
2. Feedback op de wireframes voor Damiën en Mischa.
3. Een mapje met het klassendiagram via Class Designer in Visual Studio 2022.
4. Een mapje met het klassendiagram via Visual Paradigm.
5. Last week’s sprint afbeelding.
6. Beoordelingsmatrijs.
7. Reflectiedocumenten.