Requirements Analyse

Geschreven door: groep 14

# Document Development

Als er wat veranderd staat dat hieronder beschreven.

|  |  |  |
| --- | --- | --- |
| Version | Description | Author |
| 0.1 | Initial start and creation of document | Veenstra B. |
| 0.2 |  |  |
|  |  |  |

In dit document worden de Requirements van de memorie game overzichtelijk op een rijtje getoond.

Dit word er besproken:

* Business requirements
* User requirements
* System requirements  
  1. Functional  
  2. Non-Functional

# 

### Business requirements

* Hoge tevredenheid over het spel van de klant

Het spel is makkelijk en overzichtelijk gemaakt. Alles in het spel spreekt voor zich en is daarom voor elke leeftijdscategorie te spelen. De spel is te spelen zonder handleiding of enkele uitleg en zodra je de game start kan je meteen beginnen met spelen. Dit zorgt voor hogere tevredenheid bij klanten.

* Meer klanten die het spel spelen

Door de Hoge tevredenheid over het spel van de klant zullen meer mensen het spel gaan spelen. Andere mensen zijn sneller geïnteresseerd en zullen sneller het spel gaan proberen door handige functies en meer uitdaging.

User requirements

* Het moet makkelijk te spelen zijn
* Overzichtelijk lay-out
* Handige en bruikbare functies
* Een leuk thema om mee te spelen en alle opmaak die erbij hoort

### System requirements

# Functional

Wat doet het systeem? Alles wat het systeem moet doen/moest hebben staat hieronder beschreven. Dit waren de vereisten. Alles wat hier word vermeld is terug te vinden in het memorie spel.

* Het speelveld moet bestaan uit 16 memory kaarten, deze zijn als 4 bij 4 kaarten  
  zichtbaar.
* Het scherm moet dynamisch opgebouwd worden (niet 16  
  kaarten vooraf op het windows form plaatsen), maar eerst een hoofdmenu.
* Memory moet met 2 spelers gespeeld kunnen worden, beide spelers zijn  
  ‘mens’ en spelen op hetzelfde bord/scherm (geen meerdere geopende  
  instanties van het memory spel).
* Bij het starten van het spel ziet de speler alle kaarten ‘op de kop’ in een   
  bij 4 matrix;
* De spelers kunnen hun namen ingeven in het scherm.
* Er moet een knop zijn om het spel te resetten (herstarten).
* Wanneer het spel is afgelopen, moeten de scores en namen van beide spelers worden opgeslagen en of de speler heeft verloren of gewonnen.
* Vanuit het hoofdmenu moet er genavigeerd kunnen worden naar een  
  pagina waar de highscores op te zien zijn
* De huidige status van het spel moet worden vastgelegd in het  
  tekstbestand ‘memory.sav’ (het spel wordt als het ware bevroren). De  
  volgende aspecten van het spel moeten worden vastgelegd: status van  
  de 16 kaarten, namen van de spelers, scores van de spelers, welke speler  
  aan de beurt is.
* Terug laden van de opgeslagen bevroren status van het spel. Bij het  
  terug laden van de bevroren status worden de kaarten teruggezet, de  
  namen ingevuld, de scores ingevuld en welke speler aan de beurt was.

# Non-Functional

Hoe werkt het systeem?

Alle punten die hierboven staan beschreven werken zoals het hoort. Het enige minpuntje aan de werking van het spel is dat de kaarten niet direct reageren. Zodra er op een kaart word gedrukt duurt het een kleine 0.5 seconden voordat de kaart reageert. Dit komt onder andere doordat alle zetten die worden gedaan worden opgeslagen in de memory.sav. hierdoor reageert het niet zo snel als de gebruiker misschien wil. Voor de rest werken de opties en de andere knoppen direct.