**FUNCTIONEEL ONTWERP**

Rekentuin

Merlijn Metman

28-02-2019

Versie 0.1

**Inhoudsopgave**

[1 Samenvatting 2](#_Toc401260718)

[1.1 Samenvatting voor de klant 2](#_Toc401260719)

[1.2 Versiebeheer 2](#_Toc401260720)

[1.3 Verzendlijst 2](#_Toc401260721)

[2 Plan van aanpak 3](#_Toc401260722)

[2.1 Op te leveren producten 3](#_Toc401260723)

[2.2 Planning 3](#_Toc401260724)

[3 Programma van eisen 4](#_Toc401260725)

[3.1 Doelstelling 4](#_Toc401260726)

[3.2 Huidige situatie 4](#_Toc401260727)

[3.3 Reden van verandering/aanpassing 4](#_Toc401260728)

[3.4 Nieuwe situatie 4](#_Toc401260729)

[3.5 Functionele eisen gesteld aan de nieuwe situatie 4](#_Toc401260730)

[3.5 Systeemeisen gesteld aan de nieuwe situatie 6](#_Toc401260731)

[4 Interface 7](#_Toc401260732)

[4.1 Interface 7](#_Toc401260733)

[4.2 Schema’s 7](#_Toc401260734)

[5 Use Case en Datamodel 8](#_Toc401260735)

[5.1.1 De Use Case 8](#_Toc401260736)

[5.1.2 Verklaring van de Use Case 8](#_Toc401260737)

[5.2.1 De database 8](#_Toc401260738)

[5.2.2 Het ER Model 8](#_Toc401260739)

[5.2.3 Inhoud van de database 8](#_Toc401260740)

[5.2.4 Beheer en beveiliging van de database 8](#_Toc401260741)

[6 Slotconclusie 9](#_Toc401260742)

[6.1 Verantwoording 9](#_Toc401260743)

[6.2 Bronvermelding 9](#_Toc401260744)

## 1 Samenvatting

### 1.1 Samenvatting voor de klant

Dit document omvat de rapportage over de fase functioneel ontwerp van rekentuin. Alle hoofdonderdelen evenals conclusies en aanbevelingen zijn verwerkt in dit totaaloverzicht,

met eventuele verwijzingen naar separate uitgebreide documenten, die gedurende deze fase

zijn opgesteld.

### 1.2 Versiebeheer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Versie** | **Status** | **Datum** | **Auteur** | **Aanpassingen** |
| **0.1** | **Niet definitief** | **28-02-2019** | **Merlijn Metman** | **Eerste versie** |
| **X.X** |  |  |  |  |
| **X.X** |  |  |  |  |
| **X.X** |  |  |  |  |

### 1.3 Verzendlijst

Dit document wordt ter beschikking gesteld aan:

* Jeffrey Grüne – Docent

## 2 Plan van aanpak

### 2.1 Op te leveren producten

De producten die gaan worden opgeleverd:

* Het rapport informatiebehoefte
* Het functioneel ontwerp
* Het technisch ontwerp
* De realisatie/bouw van het systeem
* Het testrapport

### 2.2 Planning

**Activiteit Einddatum**

Rapport Informatiebehoefte 21-02-2019

Functioneel Ontwerp 28-02-2019

Technisch Ontwerp 07-03-2019

Backend Ontwikkeling 14-03-2019

Front-end Ontwikkeling 14-03-2019

Testen + verslag 21-03-2019

## 3 Programma van eisen

### 3.1 Doelstelling

De doelstellingen van het functioneel ontwerp wordt hier omschreven. Deze worden opgesteld in samenspraak met de opdrachtgever Samengevat geef je hier aan wat je gaat ontwikkelen.

Basisschool rekenwonders heeft een website waar je de reken tafels kan oefenen en toetsen. Maar hier willen ze nu ook een app van maken. Deze app is bedoelt voor 6 t/m 10 jarige.

### 3.2 Huidige situatie

Er is op dit moment wel een website alleen die zal apart zijn van de app. Dus op dit moment hebben ze nog geen app alleen een website en wij gaan de app maken hij zal voor android users zijn. Dus het is helaas niet mogelijk om op een apparaat van apple deze app te gebruiken.

### 3.3 Reden van verandering/aanpassing

In het verleden heeft basisschool de Rekenwonders een programma laten ontwikkelen om de leerlingen interactief te leren rekenen. Om te voldoen aan de huidige technische vooruitgangen is besloten om het voorgaande project in een nieuw jasje te steken. Uit een intern onderzoek is gebleken dat er vraag is naar een mobiele app versie van de rekentuin.

### 3.4 Nieuwe situatie

Zodra de app gerealiseerd is kan iedereen waar ze dan ook zijn de tafels 1 t/m 10 oefenen/toetsen. En uiteraard wanner alle toetsen van 1 t/m 10 met een voldoende zijn afgerond verder gaan met de tafel van 11 t/m 20. En zullen de leerlingen het veel leuker vinden om te leren met de app in plaats van een boek of website met als gevolg betere cijfers.

### 3.5 Functionele eisen gesteld aan de nieuwe situatie

* Tafels kunnen toetsen.
* Tafels kunnen oefenen.
* Mogelijkheid om je cijferlijst te zien.
* Mogelijkheid om je gemiddelde te zien.
* Mogelijkheid om het aantal toetsen die je gemaakt hebt te kunnen zien
* Mogelijkheid om de tafel van 11 t/m 20 te ontgrendelen.

**Realisatiechecklist**

|  |  |  |
| --- | --- | --- |
| Eisen | werkt (ja/nee) | verdere acties |
|  |  |  |
| Tafels kunnen getoetst worden | Nee | n.v.t. |
| Tafels kunnen geoefend worden | Nee | n.v.t. |
| Cijfergemiddelde laten zien | Nee | n.v.t. |
| Cijferlijst laten zien | Nee | n.v.t. |
| Aantal toetsen gemaakt laten zien | Nee | n.v.t. |
| De tafel van 11 t/m 20 zijn 0ntgrendelbaar | Nee | n.v.t. |
|  |  |  |
|  |  |  |
|  |  |  |

### 3.6 Systeemeisen gesteld aan de nieuwe situatie

Wat er benodigd is om de webapplicatie te kunnen beheren:

* Windows 7 of hoger
* Wifi of LAN
* Een laptop/computer die goed een website kunnen laden etc.
* Een browser (Bijvoorbeeld Google Chrome).

## 4 Interface

Hierin beschrijf je de interface tussen programma en gebruiker.

### 4.1 Interface

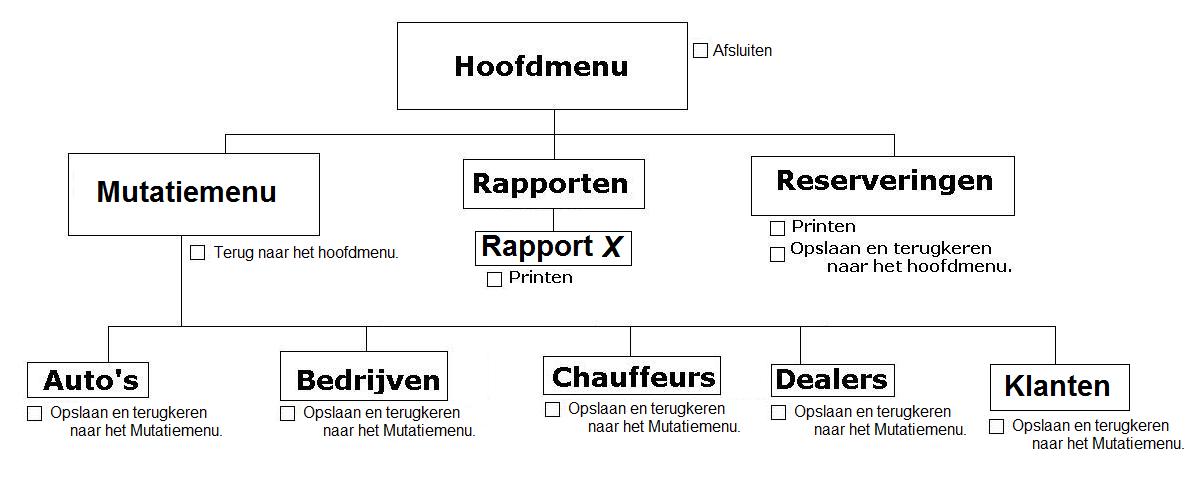
Alle schermen, inclusief vorm en lay-out van rapporten, formulieren, pop-up windows, foutboodschappen, meldingen, knoppen, menu’s, submenu’s, etc.

Je geeft aan hoe het eruit ziet en wat het doet (vanuit het standpunt van de gebruiker)

### 4.2 Schema’s

Hierin beschrijf je in welke volgorde en wanneer de verschillende schermen worden aangeroepen. Van belang is dat de klant ziet wanneer welk scherm komt en wat de mogelijkheden in zo’n scherm zijn.

Hieronder zie je een voorbeeld:



Als je via het OO model gaat werken komen hier de diagrammen die bij het OO model gebruikt worden, zoals use-case diagram en klassendiagram.

**Ook andere schema’s en diagrammen die inzicht geven in de werking van de applicatie kun je hier opnemen of in hoofdstuk 5 of in het technisch ontwerp.**

## 

## 5 USE Case en Datamodel

### 5.1.1 De USE Case

De use case beschrijft "wie" met het betreffende systeem "wat" kan doen

Een use case beschrijft een systeem vanuit het gebruikersperspectief. Het beschrijft de actor, de initiator van de interactie, en het systeem zelf als een opeenvolging van eenvoudige stappen. Actoren kunnen iets of iemand zijn, die bestaat buiten het te bestuderen systeem, en die deelneemt in de opeenvolgende activiteiten in een dialoog met het systeem om een bepaald doel te bereiken. Actoren kunnen eindgebruikers, andere systemen of hardware (apparatuur) zijn. Elke use case is een complete serie van zogenaamde "events", beschreven vanuit het standpunt van de actor.

### 5.1.2 Verklaring van de Use Case

USE Cases zijn eenvoudigweg een beschrijving van een reeks gebeurtenissen, die met elkaar het systeem in staat stellen iets bruikbaars te doen.[2] Elke use case beschrijft, hoe de actor zich zal gedragen ten opzichte van het systeem om een doel te bereiken.

### 5.2.1 De database

In de meeste gevallen wordt een database gebruikt om gegevens op te halen of op te slaan.

Het is van belang hoe deze database eruit ziet.

Daarom moet je vanuit de functionele kant alles beschrijven.

* Hoe zien de tabellen eruit
* Welke velden bevat een tabel en waarom
* welke typering heeft een veld en waarom
* Zijn er sleutels gedefinieerd (m.a.w. is een veld uniek)
* Is er een format op het veld gelegd, bv. een postcode

<<code en specifieke formats horen bij het technisch ontwerp, tenzij de klant dat moet weten>>

### 5.2.2 Het ER Model

Welke relatie is er tussen de tabellen. Geef dit weer in een schema. Wat betekent deze relatie voor het muteren van gegevens. Waarom maak je deze keuzes.

Zorg ervoor dat de database genormaliseerd is.

### 5.2.3 Inhoud van de database

Ook de grootte is van belang voor de applicatie. Hoe groter de database, hoe sneller het systeem moet zijn i.v.m. performance. Ook de hoeveelheid mutaties die verwacht gaan worden zijn van belang evenals de mogelijkheid tot het maken van back-up.

### 5.2.4 Beheer en beveiliging van de database

Beschrijf hier op welke manier hoe de klant het beheer wenst en hoe de beveiliging er uit moet zien. Wie mag wat?

## 6 Slotconclusie

### 6.1 Verantwoording

* Hier beschrijf je tegen welke problemen je bent opgelopen.
* Welke problemen heb je wel aangepakt, welke niet en waarom
* Welke consequenties gaat de nieuwe situatie veroorzaken en hoe moet daar mee worden omgegaan.

### 6.2 Bronvermelding

Documentatie: welke bronnen heb je gebruikt voor je functioneel ontwerp