FUNCTIONEEL ONTWERP

Rekentuin

Sam vd Greft

1032961

Inhoudsopgave

[1 Samenvatting 4](#_Toc3382341)

[1.1 Samenvatting voor de klant 4](#_Toc3382342)

[1.2 Versiebeheer 4](#_Toc3382343)

[1.3 Verzendlijst 4](#_Toc3382344)

[2 Plan van aanpak 5](#_Toc3382345)

[2.1 Op te leveren producten 5](#_Toc3382346)

[2.2 Planning 5](#_Toc3382347)

[3 Programma van eisen 6](#_Toc3382348)

[3.1 Doelstelling 6](#_Toc3382349)

[3.2 Huidige situatie 6](#_Toc3382350)

[3.3 Reden van verandering/aanpassing 6](#_Toc3382351)

[3.4 Nieuwe situatie 6](#_Toc3382352)

[3.5 Functionele eisen gesteld aan de nieuwe situatie 7](#_Toc3382353)

[3.5 Systeemeisen gesteld aan de nieuwe situatie 7](#_Toc3382354)

[4 Interface 8](#_Toc3382355)

[4.1 Interface 8](#_Toc3382356)

[4.2 Schema’s 12](#_Toc3382357)

[5 USE Case en Datamodel 13](#_Toc3382358)

[5.1.1 De USE Case 13](#_Toc3382359)

[5.1.2 Verklaring van de Use Case 13](#_Toc3382361)

[5.2.1 De database 15](#_Toc3382362)

[5.2.2 Inhoud van de database 15](#_Toc3382363)

[5.2.3 Beheer en beveiliging van de database 15](#_Toc3382364)

[6 Slotconclusie 16](#_Toc3382365)

[6.1 Verantwoording 16](#_Toc3382366)

[6.2 Bronvermelding 16](#_Toc3382367)

## 1 Samenvatting

### 1.1 Samenvatting voor de klant

Dit document omvat de rapportage over de fase functioneel ontwerp van mobiele applicatie <project>. 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** | **Klaar** | **10/03/19** | **Sam** | **Begin van document** |
| **0.2** | **Klaar** | **11/03/19** | **Sam** | **Designs afgemaakt** |
| **1.0** | **Klaar** | **13/03/19** | **Sam** | **Document afgerond** |
| **X.X** |  |  |  |  |

### 1.3 Verzendlijst

Dit document wordt ter beschikking gesteld aan:

Jeffrey Grüne

Anton Prajo

## 2 Plan van aanpak

### 2.1 Op te leveren producten

De op te leveren producten zijn:

* functionele eisen
* systeemeisen
* datamodel
* schema’s
* schermen
* informatiebehoefte rapport

### 2.2 Planning

De mijlpaalproducten in combinatie met de data wanneer ze opgeleverd worden vormen je planning. Uiteindelijk moet je hierin kunnen lezen wanneer je je Functioneel Ontwerp oplevert en hoe je dit doet. Via een presentatie of via een rapport of anders.

|  |  |  |  |
| --- | --- | --- | --- |
| Fase | Deadline | Oorspronkelijk ingeleverd |  |
| Informatie behoefte rapport | Zondag 17 Februari 2019 |  |  |
| functioneel ontwerp | Zondag 24 Februari 2019 |  |  |
| Technisch ontwerp | Zondag 10 Maart 2019 |  |  |
| Product | Zondag 17 Maart 2019 |  |  |
| Testen en opleveren | Zondag 24 Maart 2019 |  |  |

## 3 Programma van eisen

### 3.1 Doelstelling

De doelstelling van dit project is om een applicatie te creëren waar basisschool studenten de tafels 1 t/m 10 op kunnen oefenen en toetsen. Zodra deze tafels perfect zijn getoetst kan een student de tafels 11 t/m 20 oefenen en toetsen.

### 3.2 Huidige situatie

Op dit moment is er een website waar de studenten kunnen oefenen en toetsen. Dit is een simpele web applicatie met html/css/javascript. Deze is verouderd en daarom word er nu een applicatie gecreëerd voor android.

### 3.3 Reden van verandering/aanpassing

Tegenwoordig hebben de studenten allemaal een android telefoon en is een mobiele applicatie beter in de omgang dan een website. De website kan niet hoger dan tafel 10 en is verouderd

### 3.4 Nieuwe situatie

Het nieuwe systeem zal een mobiele applicatie worden met een simpel menu met een mooie titel en achtergrond en 2 knoppen voor toetsen en oefenen. Het oefen en toets menu zullen er redelijk hetzelfde uit komen te zien. Er zijn 10 knoppen met cijfers, 3 rijen van 3 en nummer 10 in het midden onderaan het scherm. Zodra de student tafels 11 t/m 20 heeft ontgrendelt komen er 2 navigatie pijltjes links en rechts van de 10.

Als de student op 1 van de cijfers klikt krijgt deze een scherm met overzicht te zien met het vorige cijfer dat behaalt is en een grote start knop aan de onderkant om te beginnen. Boven in de hoek zit een rode kruis om terug te gaan naar het menu

In het design zal oefenen er vrolijker uitzien en toetsen iets serieuzer. Daarbij word gedacht aan kleur verandering en verschillende layouts tussen de 2 modes.

Als een student aan het oefenen of toetsen is zal de som verschijnen aan de bovenkant van het scherm en is er een toetsenbord geïmplementeerd met de cijfers 1 t/m 9. Dit kan eventueel uitgebreid worden naar een modus met meer keuze antwoorden.

Bij oefenen krijgt de student gelijk te weten of hij het goed of fout heeft doormiddel van een groene of rode notificatie. Bij het oefenen krijgt de student niks te weten tot het eind.

Bij het toetsen en oefenen krijgt de student de antwoorden te zien aan het eind van de toets met een cijfer. Dit cijfer zal blijven staan

### 3.5 Functionele eisen gesteld aan de nieuwe situatie

Wat moet de reken applicatie kunnen:

* Tafels 1 t/m 10 kunnen worden geoefend
* Tafels 1 t/m 10 kunnen worden getoetst
* Tafels 11 t/m 20 kunnen worden geofend en getoets binnen de volgende voorwaarden:
  + Tafels 1 t/m 10 zijn perfect getoetst
* Overzicht van alle cijfers kunnen tonen

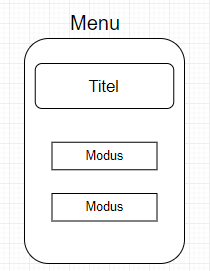
### 3.5 Systeemeisen gesteld aan de nieuwe situatie

Hard- en software benodigdheden:

* Een android telefoon met android versie 4.0 Nougat of hoger
* Minimaal 20MB vrije ruimte

## 4 Interface

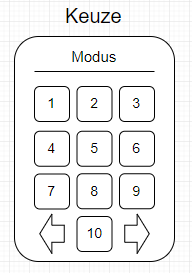
### 4.1 Interface



Menu

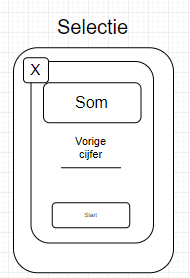
Dit is het scherm dat de student zal zien zodra deze de app opent

Hier staat de titel ‘Rekentuin’ en 2 knoppen met oefen modus en toets modus



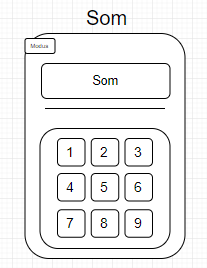
Keuze menu

Dit is het menu dat opent zodra de student op oefenen of toetsen drukt. Hier kiest de student welke som deze wilt doen



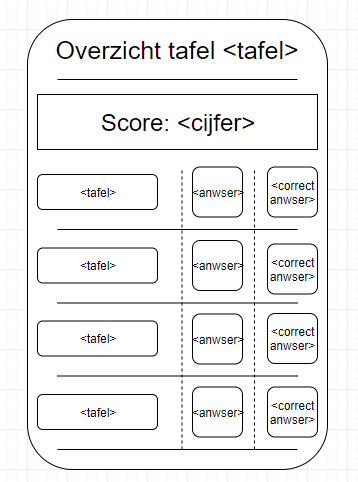
Selectie scherm

Dit scherm opent zodra de student op een som klikt om te bevestigen of ze deze som willen doen. Ook krijgen ze hier het cijfer te zien wat ze vorige keer hebben gehaald. Dit cijfer is het hoogste cijfer, deze wordt pas vervangen als er een hoger cijfer wordt gehaald



Som maken

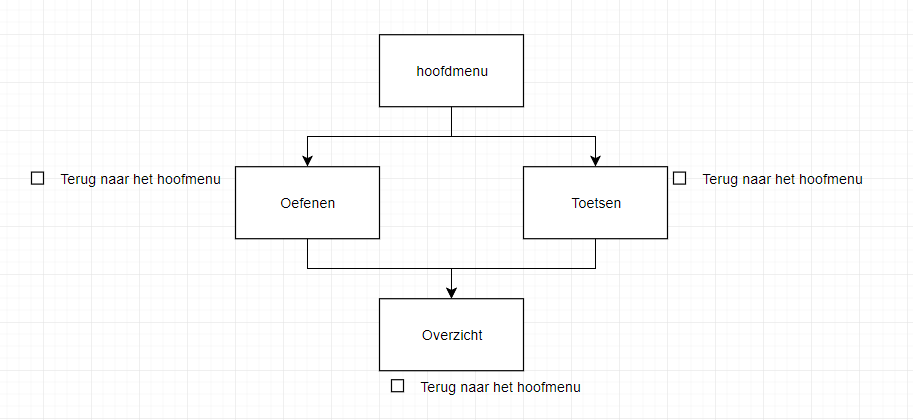
Dit scherm wordt wergeven zodra de student een som heeft gestart. Dit zelfde scherm herhaalt zich door de opdracht met alle sommen. Zodra de student een antwoord heeft ingevoerd krijgt deze (bij het oefenen wel) niks te zien. Pas op het eind bij het overzicht (het volgende scherm) krijgt de student zijn cijfer te zien met de juiste antwoorden en de antwoorden die zij gegeven hebben.



Het overzicht

Dit is het overzicht. Dit scherm ziet de student aan het eind van een toets of oefening. Hier staat de student zijn cijfer en alle tafels die gemaakt zijn. Het correcte en gegeven antwoord staan naast elkaar voor referentie

### 4.2 Schema’s



Zodra een student de app opent begint deze op het hoofdmenu. Hier vind de student 2 knoppen die leiden naar oefenen of toetsen. Vanaf daar kiezen ze een tafel, wanneer ze op een tafel drukken krijgen ze hun vorige cijfer te zien (deze word opgeslagen van de laatste toets die is gemaakt) op dar scherm drukken ze op start en maken de sommen, zodra ze de sommen hebben gedaan komen ze bij het overzicht waar ze hun resultaten kunnen bekijken

## 5 USE Case en Datamodel

### 5.1.1 De USE Case

### 

In de rekentuin app kan de student een toets maken of oefenen en zijn cijfers bekijken. Dat is het eigenlijk wel

### 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.

|  |  |
| --- | --- |
| **Use-case naam:** | Toetsen |
| **Versie:** | **1.0** |
| **Actor(en):** | **Student** |
| **Preconditie:** | **App is ge-opent** |
| **Main scenario:** | 1. Student klikt op toetsen knop 2. Student kiest een tafel 3. Daarna drukt deze op start 4. De student maakt de sommen 5. En bekijkt het resultaat met zijn cijfer |
| **Postconditie:** |  |
|  |  |
| **Use-case naam:** | Oefenen |
| **Versie:** | **1.0** |
| **Actor(en):** | **Student** |
| **Preconditie:** | **App is ge-opent** |
| **Main scenario:** | 1. Student klikt op oefenen knop 2. Student kiest een tafel 3. Daarna drukt deze op start 4. De student maakt de sommen 5. En bekijkt het resultaat met zijn cijfer |
| **Postconditie:** |  |

### 5.2.1 De database

Er zal gebruik gemaakt worden van bestanden om gegevens op te slaan inplaats van een database. Hierdoor zal de student niet continu een internet verbinding nodig hebben

### 5.2.2 Inhoud van de database

Alle data zal lokaal worden opgeslagen in een json bestand. Hierin staat de hoeveelheid tafels zodat er makkelijk tafels kunnen worden toegevoegd en ingeladen. Hier staan ook de behaalde cijfers van de student.

### 5.2.3 Beheer en beveiliging van de database

Er zal niet veel beveiliging zijn aangezien het een simpele studenten app is waarbij de gegeven data niet erg waardevol is.

## 6 Slotconclusie

### Verantwoording

* Uitzoeken hoe ik precies data ga opslaan is een werkend proces
* Het opstarten van de app gaat waarschijnlijk moeizaam aangezien dit mijn eerste app word
* Een admin panel is niet aanwezig aangezien daar niet om is gevraagd, dit kan eventueel wel geïmplementeerd worden. Om alle resultaten van alle studenten bij te kunnen houden zal de overstap naar een database waarschijnlijk nodig zijn met een inlog/registreer systeem

### 6.2 Bronvermelding

Ik heb bronnen gebruikt die niet van toepassing zijn voor de klant. Bijvoorbeeld coding tutorials om te besluiten wat ik ga doen binnen de code