Projectdocument

PocketTopo

Team Topo

Laurens Castelijns

Denzel Feurich

Machteld Hamers

Felix Hinlopen

Mark Jan van Lieburg

Daniël Lin

November 2015

**Inhoudsopgave**

Inhoudsopgave 1

Inleiding 2

Productanalyse 3

Handleiding 7

Projectaanpak 8

Praktische informatie 11

**Inleiding**

Dit document beschrijft hoe ons team te werk is gegaan met het informatica introductieproject voor eerstejaars bachelor studenten, onder begeleiding van professor doctor Arno Siebes. Ons team is op het idee gekomen om een applicatie te ontwikkelen voor topografie, met als doelgroep basisscholieren.

Waarom het ontwikkelen van een applicatie? Vaak kan topografie leren gezien worden als een saaie taak om alles gewoon in je hoofd te stampen. Door middel van leren met een applicatie op een mobiel apparaat kan het leren van topografie op een moderne manier, maar het kan ook als leuk ervaren worden.

Ons doel met het ontwikkelen van deze applicatie is dan ook dat scholieren die topografie moeten leren niet meer als een obstakel zien, maar het spelenderwijs leren meer motivatie zal geven aan de scholieren

**Product analyse**

In dit stuk wordt er over het product dat er geleverd moet worden gesproken. Zoals in de inleiding beschreven staat, gaat het team een applicatie ontwikkelen waarmee gebruikers, voornamelijk basisscholieren, topografie mee kunnen leren. Het enige wat nodig is voor deze app is een Android apparaat en een eventuele internetconnectie om de app te downloaden. De code wordt geschreven in Java , met behulp van AndroidStudio wordt de applicatie gerealiseerd.

**Interface**

Na het opstarten van de applicatie krijgt de gebruiker het menu. Op het startscherm kan de gebruiker kiezen voor de volgende opties: oefenen, instellingen en info. Als er wordt gekozen voor oefenen moet de gebruiker selecteren wat er geoefend moet worden. Nadat de gebruiker klaar is met het selecteren kan er geoefend worden met topografie. In figuur 1 en 2 zijn er voorbeelden te zien van het concept van de interface.

**

*fig.1 beginscherm*

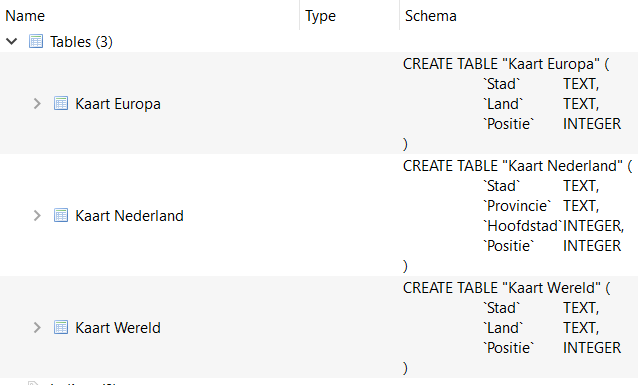


*fig.2 voorbeeldvragen*

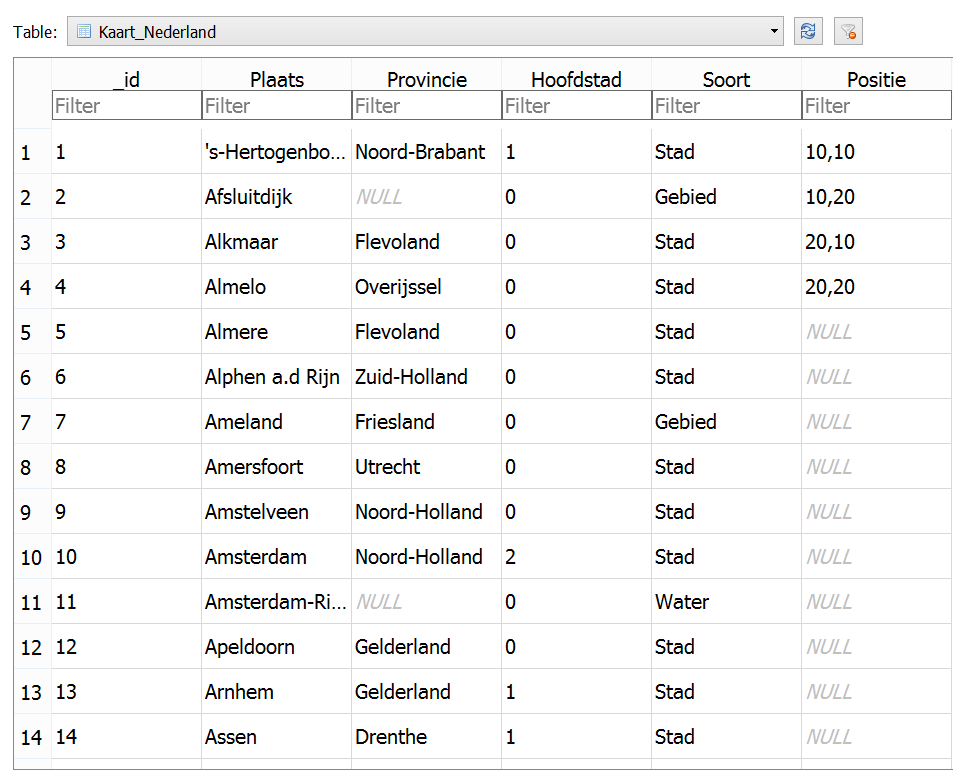
**Vragenstelling**

Het team maakt gebruik van een database voor de vragen, antwoorden, het controleren van antwoorden en distractors bij meerkeuze vragen. Distractors zijn foutieve antwoorden bij meerkeuze vragen .Verder is de keuze gemaakt dat er gebruik gemaakt gaat worden van SQLite. SQLite is een veelgebruikte database voor android apparaten, dit is een van de redenen waarom er is gekozen voor SQLite.

In figuur 3 is een concept gegeven van een database. De juistheid van het antwoord wordt gecontroleerd door middel van de juiste combinatie in de database.



*fig.3 voorbeeld database*



*fig.4 Voorbeeld tabel in database*

De gegevens die uit de database worden gehaald wordt bepaalt door middel van de input van de gebruiker die deze gegevens tijdens de selectionactivity meegeeft. Daarnaast wordt er een lijst gemaakt met distractors , deze lijst wordt gevuld door elementen uit de database. Verder zorgt een klasse, TopoHelper, ervoor dat er een lijst wordt aangemaakt met vragen. Daarnaast zorgt de klasse TopoHelper ervoor dat er een lijst met vragen wordt aangemaakt.

Daarnaast wordt er op het beeld altijd een kaart afgebeeld die bij de vragen hoort. Dit wordt met behulp van een bitmap gedaan.

**Functionaliteit**

Als het gaat om functionaliteit moet er worden voor gezorgd dat de applicatie op niveau is. Het is niet de bedoeling dat de gebruiker alleen steden krijgt die hij/zij al kent. Het zou erg handig zijn voor de gebruiker als diegene feedback terug kan krijgen, bijvoorbeeld een percentage en herhalingen van vragen. Dit laatste werkt door middel van een foute beantwoorde vraag weer achteraan de lijst toe te voegen. Verder moet de applicatie meerdere toetsing mogelijkheden bevatten. Bijvoorbeeld multiple choice of openvragen.

Om een beter beeld te geven welke functies de applicatie allemaal moet hebben, hebben wij een lijst opgesteld met de functies die deze applicatie moet, zou moeten, had kunnen en zeker niet zal bevatten.

**Must have**

* Het volledige spel voor Nederland.
* Selectie maken van wat er getoetst moet worden (provincies, hoofdsteden, steden, rivieren)
* Oefeningen op basis van stadsnaam
* Oefening (Multiple choice, aanwijzing)
* Herhalingen van foute vragen

**Should have**

* Evaluatie aan het eind van de oefening
* Hint button
* Het volledige spel voor andere landen / de hele wereld

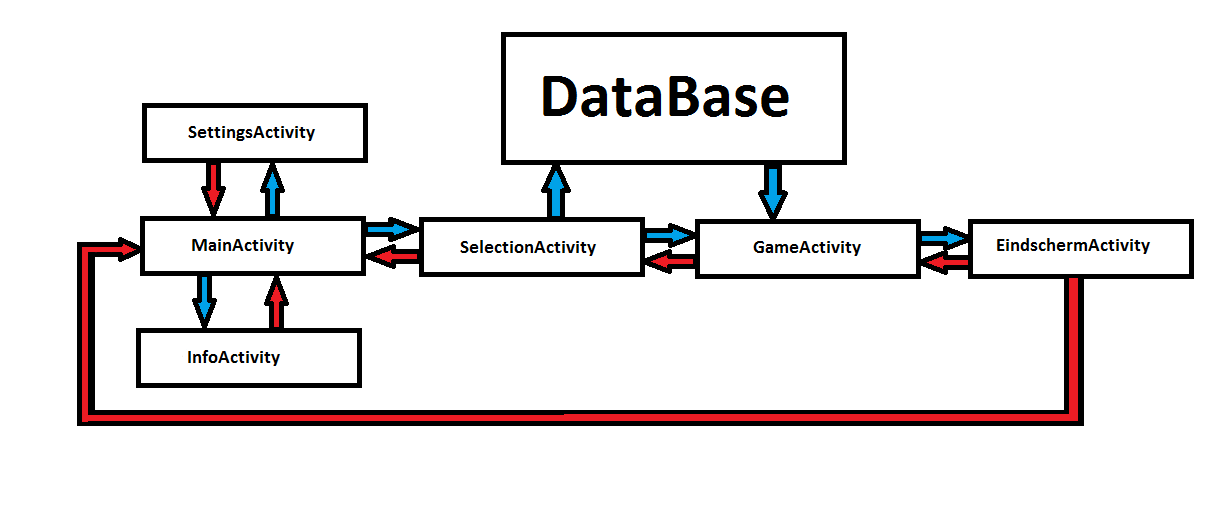
**Could have**

* Meest recente resultaten
* Kleine weetjes na het goed hebben van het antwoord
* Oefeningen over rivieren / zeeën / gebergtes / andere gebieden
* Oefeningen op basis van vlaggen / beschrijvingen over land / stad
* Dorpjes in Nederland (per provincie)
* Geluidseffecten / Animaties
* Vertaling (Engels)
* Een spel

**Won’t have**

* Kleine dorpen buiten Nederland
* 3D
* Advertenties
* Sociale aspecten

In figuur 5 is afgebeeld vanaf welk scherm de gebruiker naar een ander scherm kan springen. Zoals te zien in figuur 5 kan de gebruiker vanaf het startscherm naar het niveauselectie scherm. Het niveauselectie houdt als het ware in dat de gebruiker kan kiezen wat die van een bepaald land wil oefenen. De gebruiker kan kiezen uit: wateren, steden, hoofdsteden en provincies. Na het niveauselectie wordt het spelselectiescherm getoond. Met spelselectie kan de gebruiker kiezen wat voor soort vragen hij/zij wil hebben, met de keuze uit de volgende opties: meerkeuze, aanwijzen, of beantwoorden welk blokje er op het scherm wordt aangewezen. Na alle keuzes te hebben gemaakt kan het spel begonnen worden. Tijdens het spel kan de gebruiker naar een pauzescherm. Vanuit het pauzescherm kan de gebruiker nog naar de instelling, mocht de gebruiker geluiden die in de applicatie verwerkt zitten niet fijn vinden, kan hij die vanuit daar uitzetten.



*fig.5 UI*

**Gebruikshandleiding**

In de applicatie kunnen er verschillende gebruikers scenario voorkomen. Om deze allemaal op een rijtje te zetten zullen de scenario’s op papier komen te staan. Bekijk figuur 5 op de vorige bladzijde voor meer informatie

**Startscherm**

Vanuit het startscherm kan de gebruiker kiezen om naar welk volgscherm te springen. De gebruiker heeft de keuze uit : Spel (selectiescherm), Instellingen en een infoscherm.

**Selectiescherm**

Als de gebruiker op de button “spel” heeft gedrukt komt diegene als eerst uit op een selectiescherm. Hier dient de gebruiker zijn selectie te maken bijvoorbeeld : wat er geoefend moet worden en hoe. Met wat er geoefend moet worden , wordt bedoelt of de gebruiker liever steden, provincies en hoofdsteden of wateren enzovoorts wil oefenen. Hoe het geoefend kan worden is ook geheel aan de gebruiker. De gebruiker kan ervoor kiezen voor multiple-choice, openvragen, aanwijsvragen. Bij openvragen is het zo dat de gebruiker niet meer dan een woord hoeft in te vullen. Aanwijsvragen zijn vragen waar er een bepaalde plaats gevraagd wordt en de gebruiker die kan aanwijzen op de kaart.

**Spelscherm**

In dit scherm is waar het eigenlijke overhoren/ spelen gaat gebeuren. Hierin wordt alle input van de gebruiker verwerkt, nadat het aantal vragen zijn beantwoordt, wordt de gebruiker naar een eindscherm toegestuurd.

**Eindscherm**

Dit scherm bevat feedback voor de gebruiker zodat die weet waaraan hij of zij aan toe is. Het eindscherm toont dan ook het aantal vragen , het aantal goede. Op basis van onze eigen indicatie krijgt de gebruiker ook advies .Vanuit het eindscherm kan er gekozen worden om het spel opnieuw te spelen, andere selecties te maken, of helemaal te stoppen met het spel.

**Instellingenscherm**

In het instellingenscherm kan de gebruiker ervoor kiezen om het spel zoveel mogelijk naar zijn of haar voorkeur in te stellen. Bijvoorbeeld wil de gebruiker de geluiden die erin zijn verwerkt aan of uit hebben. In het instelingenscherm is tevens ook de functie voor het herhalen van vragen te vinden.

**Infoscherm**

Hierin vindt de gebruiker een verkorte versie van de gebruikshandleiding. Hierin staat uitgelegd hoe het spel als het ware in elkaar zit en mocht de gebruiker geïnteresseerd zijn , kan hij of zij naar de website met een enkele knop.

**Projectaanpak**

Ten eerste heeft het team globaal een interface op papier gevisualiseerd. Daarna is er bekeken vanuit welke pagina we naar de ander kunnen springen, bijvoorbeeld vanuit het hoofdscherm kunnen we naar de oefenpagina, instellingenpagina, spelpagina en de infopagina. Dit is voor alle pagina’s die in de applicatie komen gedaan. Ten tweede is de vragenstelling en de functionaliteit aan bod gekomen, hierover is al het een en ander behandelt in de productanalyse. Ten slotte hebben we alle klassen die we nodig zullen hebben op papier gezet en die onderverdeeld over de teamgenoten. Hieruit is het volgende tot stand gekomen.

**Taakverdeling**

|  |  |  |
| --- | --- | --- |
|  | Wat | wie |
| - | MainActivity met buttons die naar pagina’s  Springen | Daniël |
| - | SettingsActivity met shared preferences | Machteld |
| - | SelectionActivity met input van de gebruiker | Felix |
| - | GameActivity met functie voor het uitvoeren  van de vraag, een lijst met <vragen> ,  quiz uitvoeren , de score bijhouden die uiteindelijk  wordt verwerkt in de EindschermActivity | Mark Jan |
| - | EindschermActivity met de score/ feedback  over wat de gebruiker goed deed en wat minder  goed ging. Een button om de game te restarten  en een button om de game te stoppen | Daniël |
| - | Database creëren | Laurens |
| - | De vraag genereren , parameters die worden  meegegeven door de gebruiker bij de selectionActivity.  Functie om gegevens uit de database te halen  die gebaseerd zijn op de parameters die  de gebruiker heeft meegegeven | Laurens |
| - | VraagFragmenten , fragmenten die de vraag aanduiden  deze vragen kunnen de vragen kunnen zichzelf tekenen  en zorgen voor de afhandeling van een vraag. | Mark jan |
| - | InfoActivity, informatie over de app en een button  die verwijst naar onze website | Daniël |
| - | Kaartplaatjes maken | Felix & Denzel |
| - | Database vullen met de benodigde informatie | Machteld |
| - | Website maken | Machteld |
| - | Flyer maken | Denzel |
| - | Gebruikshandleiding maken | Daniël |
| - | Projectdocument maken en bijhouden | Daniël |
|  |  |  |

Verder is er gekozen voor het ontwikkelprincipe Scrum .Vanwege tijdgebrek zijn de sprints niet dertig dagen maar zeven dagen. Als de sprints niet behaald worden zal er in de week die daarop volgt extra aandacht worden besteedt aan wat er niet is gehaald. Nieuwe sprints worden gebaseerd op de vorige sprints wat er wel is behaalt en wat niet. Er wordt van iedere teamgenoot verwacht dat deze een logboek bijhoudt. Dit logboek moet informatie bevatten met wat deze desbetreffende persoon heeft gedaan en hoeveel uur er aan besteedt is. Verder moet de persoon ook het resultaat ervan kunnen aantonen ook al werkt het nog niet. Mocht er weinig tot geen vorderingen in een week zijn gemaakt, dan vindt er midden in de week nog een bespreking plaats.

De werkwijze bij het samenwerken aan programmacode is als volgt gegaan. Er is in de groep afgesproken dat alleen werkende componenten op github komen te staan. Zodra iets nog niet werkt wordt er tijdens de eerste wekelijkse bijeenkomst van de week bekeken welke vorderingen er wel zijn gemaakt. Stel dat er weinig vordering of geen vordering is gemaakt, dan zal een andere teamgenoot meehelpen met de andere twee.

In figuur 6, op de volgende pagina, is er voorbeeld ontwerp gemaakt voor de leden om een richtlijn te creëren om misverstanden zoveel mogelijk te voorkomen. Hierin is er te zien dat in de database de vragen, antwoorden, en distractors komen. De database controleert de juistheid van de antwoorden. Verder staan er in figuur 6 een aantal voorbeeldnamen van methoden die er gebruikt worden in onze code.

 *fig.6 voor beeldontwerp*

**Planning**

Tijdens het project wordt er verwacht dat er een aantal documenten geleverd worden en zijn er ook een aantal deadlines. Hieronder worden die op een rijtje gezet.

Logboek 10 November

Projectdocument 2 December

Website 2 December

Tussenproduct 9 December

Tussenproduct presentatie 16 December

Flyer 13 Januari

Eindproduct , Website, Eindverslag,Flyer 22 Januari

**Praktische informatie**

Laurens Castelijns

l.a.castelijns@students.uu.nl

Denzel Feurich

d.feurich@students.uu.nl

Machteld Hamers

m.l.s.hamers@students.uu.nl

Felix Hinlopen

f.a.hinlopen@students.uu.nl

Mark Jan van Lieburg

m.j.lieburg@students.uu.nl

Daniël Lin

d.lin@students.uu.nl