Kyle (K.H.A.L.) Lukassen

Abstract

Een formulier dat detailleert hoe het escape vista spel werkt, hoe eraan gewerkt kan worden en welke aspecten nog moeten gebeuren.

Overdrachtsformulier escape vista

Escape vista spel

Inhoud

[Huidige situatie 2](#_Toc94256749)

[Opbouw van de website 2](#_Toc94256750)

[pijlen 2](#_Toc94256751)

[Afbeeldingen 2](#_Toc94256752)

[Werking van het spel 2](#_Toc94256753)

[Functies 3](#_Toc94256754)

[Travelpath(Destination) 3](#_Toc94256755)

[TaskQuestioning(QuestionTaskID) 3](#_Toc94256756)

[LocationVisit(LocationID) 3](#_Toc94256757)

[AnswerTask(InputTask, TrueAnswer) 4](#_Toc94256758)

[PointUpdate(Points) 4](#_Toc94256759)

[Opbouw van de data 4](#_Toc94256760)

[Bestemmingsdata 4](#_Toc94256761)

[Opdrachtdata 5](#_Toc94256762)

[Wat nog moet gebeuren 5](#_Toc94256763)

[Een eind punt 5](#_Toc94256764)

[“Inspect Element” preventeren 5](#_Toc94256765)

# Huidige situatie

De opdracht die gegeven werd, was om een website op te stellen die dient als een introductie voor studenten die de ICT opleiding op het VISTA college in Heerlen willen gaan volgen. Deze introductie website is in 2 onderdelen opgesplitst: een informatie pagina, waarop de gebruiker informatie kan vinden over vakken, docenten en schoolregels, enzovoorts. Het tweede onderdeel is een spel in de form van Point-and-click, waarbij de gebruiker een pad afloopt en daarmee het gebouw van binnen te zien krijgt. Dit pad is gelimiteerd tot de B toren en algemene belangrijke locaties binnen het gebouw aangezien de gebruiker tijdens de opleiding de meeste tijd in de B toren zal begeven.

# Opbouw van de website

Het pad wordt afgelopen door middel van afbeeldingen die in een SVG tag staan, deze tag houd ook de pijlen waarmee het pad belopen word. De website is verdeelt in een grid, waarbij ieder item zijn eigen grid-area heeft. Dit zorgt ervoor dat de items op de juiste plek blijven staan als het formaat van het venster verandert.

## pijlen

Dit pad wordt doorlopen door een maximum van 4 pijlen, waarvan een bepaalde aantal zichtbaar is, liggend aan de getoonde afbeelding. Pijlen die niet nodig zijn, worden buiten zicht geplaatst zodat deze niet zichtbaar zijn. De coordinaten van de pijlen zijn relatief aan de locatie van de SVG tag waar deze in staan.

## Afbeeldingen

De afbeeldingen hebben dezelfde naam als het ID van de bestemming, bijv. de bestemming “Hoofdingang” heeft een afbeelding met de naam “Hoofdingang.jpg”. Als een afbeelding verandert wordt, dan wordt het ID van de bestemming gepakt en een afbeelding met die naam + .jpg wordt dat neergezet op de webpagina.

# Werking van het spel

In de file: “EscapeVista.js” staat alle code die een functie uitvoert op de webpagina. Alle belangrijke data die gebruikt wordt in deze functies is terug te vinden in de file: “EscapeVistaImageInfo.js”.

Het pad van het spel wordt doorlopen door op de pijlen te klikken die op het scherm staan. Als op een pijl geklikt wordt worden de volgende stappen doorlopen in volgorde:

1. Doorloop de lijst ***ImageDataList*** totdat een item is gevonden waarvan ***ImageID*** overeen komt met de gevraagde bestemming. In dit item is alle informatie te vinden die nodig om de bestemming correct weer tegeven.
2. Van de vier pijlen op de webpagina, verander de “X” coördinaat.
3. Van de vier pijlen op de webpagina, verander de “Y” coördinaat.
4. Verander afbeelding die door ***Imageholder*** vastgehouden worden. Zie kopje “afbeeldingen” onder “Opbouw van website” voor naamgeving van de afbeeldingen.
5. Van iedere pijl wordt het attribute “onclick” aangepast zodat de bijbehorende pijl naar de juiste bestemming leidt als erop geklikt wordt.
6. De image die de pijl representeert wordt verandert, zodat de pijlen in de nieuwe bestemming de juiste richting aanwijzen.
7. Er wordt een check uitgevoerd om te kijken of de bestemming een opdracht heeft, als de bestemming een opdracht heeft, dan wordt een nieuwe functie aan geroepen die de opdracht op de webpagina toont. Als geen opdracht gevonden is, dan wordt de div waar de opdracht in getoond zou worden, onzichtbaar gemaakt zodat deze niet in beeld staat en afleidt.

# Functies

Hier onder worden de functies beschreven die op uitgevoerd kunnen worden en welke taken ze uitvoeren.

## Travelpath(Destination)

Beschreven in “Werking van het spel”.

## TaskQuestioning(QuestionTaskID)

Deze functie wordt aan geroepen wanneer een bestemming een opdracht heeft voor de gebruiker. Involgorde doet de functie de volgende taken uitvoeren:

1. Het blok wat de opdracht vraag vast houdt onzichtbaar maken zodat deze weggehaald wordt als deze is blijven staan
2. Een variable opstellen om de opdracht vragen in te zetten
3. Loop door de lijst van opdrachten totdat de opdracht is gevonden die gezocht wordt
4. Check of de opgehaalde opdracht al eens eerder voltooid is
5. Maak het block om de opdracht in te zetten zichtbaar
6. Als de opdracht een hint heeft, dan word de vraag gesteld. Als de opdracht geen hint heeft, dan is het een locatie om te bezoeken en wordt een volgende functie aan geroepen
7. Als de opdracht een hint heeft:
   1. Voor ieder antwoord in de opdracht, maak een input label van het type “radio” met daarin de vraag.
   2. Maak van alle antwoorden een string
   3. Verwijder de komma’s die tussen de antwoorden gezet waren
   4. Zet de opdracht en antwoorden in het block
   5. zet een onclick attribute voor de button
8. als de opdracht geen hint heeft, laat alleen de opdracht tekst zien, en roep de functie ***LocationVisit*** aan met het ID van de opdracht

## LocationVisit(LocationID)

Deze functie heeft als taak dat als een locatie bezocht wordt, dat opgeslagen wordt dat die locatie bezocht is en punten ervoor gegeven worden. In volgorde zijn de taken:

1. vindt de informatie van de locatie en sla deze op in een variable
2. als de locatie nog niet eerder bezocht is, ga door met de functie
3. als de lijst met bezocht locaties deze locatie nog niet heeft, ga door met de functie
4. zet in de lijst deze locatie met punten aangewezen
5. update de progess bar om de voortgang aan te geven
6. sla op dat de locatie bezocht is.
7. Roep functie aan om scoreboard up te daten.

## AnswerTask(InputTask, TrueAnswer)

Deze functie heeft als taak het afhandelen van het beantwoorden van een vraag. In volgorde zijn de taken van deze functie:

1. Haalt op de value van het gegeven antwoord
2. Haal op de informatie van de beantwoorde opdracht
3. Check of de vraag nog niet eerder beantwoord is.
4. Check of het gegeven antwoord ook het correcte antwoord is
5. Als het antwoord correct is, zet de opdracht met punten gegeven in een lijst en update de progress bar. Vervolgens roep functie aan om scoreboard up te daten
6. Als het antwoord niet correct is, zet opdracht in lijst zonder punten, en roep functie aan om scoreboard up te daten
7. Zet opdracht naar ‘beantwoord’.

## PointUpdate(Points)

Deze functie heeft als taak het bij werken van het scoreboard wanneer een opdracht voltooid is of locatie bezocht is. In volgorde heeft de functie de volgende taken:

1. Maak een lege variable aan
2. Bereken totale score
3. Voor ieder item in de lijst met voltooide opdracht, schrijf een lijn naar de variable met de naam van de opdracht en behaalde punten
4. Voor ieder item in de lisjt met bezochte locaties, schrijf een lijn naar de variable met de locatie en behaalde punten
5. Schrijf de variable naar het scoreboard.

# Opbouw van de data

In de file: “EscapeVistaImageInfo.js” staat alle informatie voor de foto’s en opdrachten die op de webpagina te zien zijn. In deze file staat alle data en zullen functies te vinden zijn.

Hieronder worden de verschillende items besproken die op geslagen zijn en wat ieder key voorstelt en de soort data die er in gebruikt.

## Bestemmingsdata

* ImageID
  + Het ID van de bestemming waar naar gezocht wordt.
* Arrow1CoordsX
  + De X coordinaat van pijl 1
* Arrow1CoordsY
  + De Y coordinaat van pijl 1
* Arrow1Direction
  + De richting waarin de pijl wijst.
* Destination1
  + De bestemming waarnaar toe gereist wordt als op de pijl geklikt wordt
* Task
  + Eventuele task die bij de bestemming hoort.

## Opdrachtdata

* TaskID
  + De ID voor de opdracht
* TaskQuestion
  + De vraag die gesteld wordt, of in het geval van locatie, de tekst die geplaatst wordt
* TaskAnswers
  + De antwoorden waar de gebruiker uit kan kiezen
* TaskCorrectAnswer
  + Het correcte antwoord
* TaskCompleted
  + Als de Opdracht nog niet beantwoord is, of de locatie is nog niet bezocht, dan staat deze op False. Deze key is om te checken of een locatie als een eerder bezocht is, of dat een vraag als eens eerder beantwoord is.
* TaskHint
  + Of de vraag een hint kan geven.

# Wat nog moet gebeuren

De volgende punten zijn langs gekomen die nog moeten gebeuren of waarvoor nog gevraagd werd.

## Een eind punt

Het project heeft nog geen eindpunt voor het escape-vista spel. Een suggestie voor dit eind punt was ruimte B.2.03. Het eind punt zou de totaal score laten zien die de gebruiker gehaald heeft, en geeft een link weer waarmee de informatie pagina bereikt kan worden.

## “Inspect Element” preventeren

Het voorkomen van Element inspecteren. In de huidige situatie kan de gebruiker de webpagina inspecteren en zo de antwoorden vinden voor de opdrachten als ze weten waar ze moeten kijken. Door dit te voorkomen, kunnen de gebruikers niet cheaten met de opdrachten. Om dit te doen moet de code: ***oncontextmenu="return false" onkeydown="return false;" onmousedown="return false;"*** in de body tag geplaatst worden van de webpagina. Deze staat er momenteel nog niet om development niet te hinderen.