

# CardCase

## A Collection Game

Door Sven Wilbrink en Nathan Versteeg

Eindproject web development VW05



# Inleiding

Bij het krijgen van de opdracht wisten wij nog niet zo goed wat we wilde gaan maken. Na wat brainstormen hadden we een vaag idee. Iets met kaarten en het verkopen van deze kaarten. Uiteindelijk kwamen op een concreet idee. Vervolgens hebben we een aantal versies van de site bedacht:

v0. De gebruikers kunnen inloggen en kaarten online zetten om te kopen of verkopen met online geld. Kaarten worden online niet doorgegeven.

v1. Gebruikers hebben een inventaris met kaarten, kaarten kunnen nu wel online doorgegeven worden.

v2. Er is een shop waar gebruikers kaarten kunnen krijgen uit packs.

v3. De shop heeft nu ook kaarten die per uur resetten om te kopen. Ook online geld is te verkrijgen.

Na dat we doelen gesteld hadden gingen we aan de slag. We begonnen natuurlijk met het idee 'trading cards' uitwerken. Sven heeft hier een prachtig spreadsheetje van gemaakt die te vinden is in de drive.

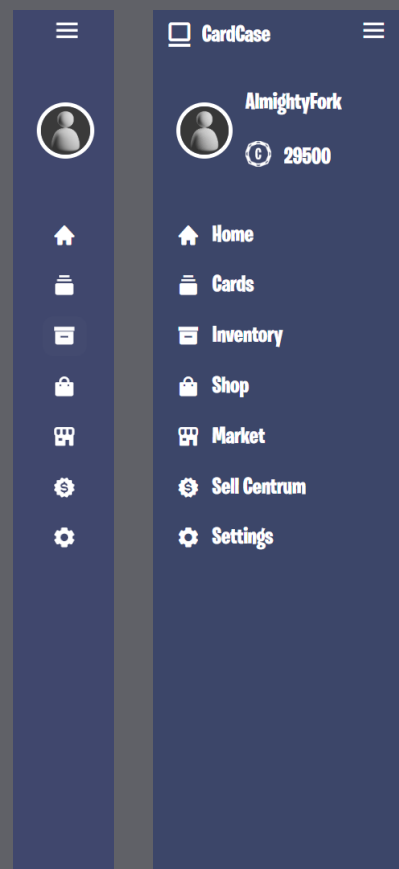
# Usability

Dit onderdeel is verdeeld in 6 stukken, deze stukken zijn:

1. appropriateness recognisability
2. User error protection
3. Learnability
4. Operability
5. User interface aesthetics
6. Accessibility

## 1. Appropriateness recognisability

Wij hebben bewust ons logo simpel gehouden zodat het mensen snel bij blijft. Het logo stelt een stapel kaarten voor, dat is natuurlijk erg toepasselijk voor onze website. Zo weten mensen ook snel of onze site is wat ze zoeken. Verder hebben wij op dit gebied ervoor gezorgd dat er voor alles duidelijke te herkennen is door icoontjes te gebruiken. Zo is in ons menu met duidelijke icoontjes aangeven wat wat is, ook is het nog eens in- en uitklapbaar om meer informatie weer te geven wanneer dit nodig is. Daarnaast als je op een kaart drukt heeft deze ook nog overal icoontjes staan om makkelijk de eigenschappen van de kaart weer te geven en te herkennen.



## 2. User error protection

Natuurlijk wil je niet dat de gebruiker ergens op klik of iets doet wat hij/zij niet wil of wat leidt tot een error. Daarom hebben we ten eerste een 404-pagina gemaakt zodat de gebruiker na een typefout gewoon weer terug kan naar een andere pagina. Een plek waar veel fout zou kunnen gaan op onze website is natuurlijk het verkoop van kaarten. Dat is de rede dat je na het selecteren van een kaart een klein overzichtje krijgt. Niemand wil perongeluk zijn mythische kaart verkopen, toch? Bovendien kunnen we door het gebruiken van een dropdown menu makkelijk de mogelijke kaarten aanpassen die beschikbaar zijn om te verkopen. Op deze manier voorkomen we dat mensen kaarten kunnen verkopen die ze niet bezitten of die helemaal niet bestaan. Tot slot willen dat meerdere accounts tegelijkertijd bezig kunnen zijn, dat is een deel van de rede dat we sessies gebruiken in flask.

## 3. Learnability

Op onze site CardCase is alles vrij makkelijk te herkennen en zijn er geen ingewikkelde stappen. Dit geeft dus geen overweldigend gevoel aan nieuwkomers. Een aantal van onze vrienden snapten het al snel, zelfs Svens oma begreep het nadat sven de website had laten zien, je dus kunt wel zeggen dat de website makkelijk te (leren) gebruiken is.

## 4. Operability

Al vanaf de start wisten we hoe gebruiker zich in de site moesten gaan navigeren, een sidebar. Wij kwamen bij dit idee omdat wij beide op onze 1e website een bar aan de bovenkant hadden. Toen wij met die topbar bezig waren merkte we dat je al snel te weinig ruimte hebt voor nieuwe links, we moesten dan alsnog een dropdown gebruiken. Dat vonden we allebei een te ingewikkelde manier om snel naar de gewenste pagina te navigeren. Om de website makkelijker gebruikelijk te maken hebben we dus een sidebar gebruikt. Bovendien kan je daar makkelijk heel veel data kwijt omdat het mogelijk is om deze in en uit te klappen. Om informatie niet meerdere keren te laten zien verdwijnt bijvoorbeeld het aantal C-BUCKS wat de gebruiker bezit van de site op het moment van het uitklappen van de sidebar. Ook hebben we onze gebruik van elementen en kleuren zeer consistent gehouden. Zo weet de gebruiker altijd wat wat is.

## 5. User interface aesthetics

We hebben voor de site een consistent kleurenpalet gebruikt om het niet te druk te maken en we hebben ervoor gezorgd dat er niet te veel op het scherm is. We hebben dit bereikt door relatief donkere kleuren te gebruiken en alleen waar nodig was lichtere te gebruiken maar wel met de gedachte dat het niet in strijd moet zijn met de donkere kleuren. Elke form of button staat in een div met een blauwe achtergrond, buttons zijn allemaal licht blauw en inputs paars. We kozen hiervoor zodat de website als 1 geheel voelt, alles hangt nu samen. wat over pixel art en kaarten



Form



Button

## 6. Accessibility

Onze website is erg toegankelijk en gebruiksvriendelijk. We hebben de gedaan door bijvoorbeeld de home- en kaartenpagina toegankelijk te maken voor iedereen of je nou wel of niet bent ingelogd. Waarom? Dit is gedaan zodat mensen eerst rustig kunnen verkennen voordat ze besluiten een account te maken. Als ze inderdaad een account willen is dit niet moeilijk te maken. Ga naar elke willekeurige pagina of die nou bestaat of niet en je wordt meteen naar de login doorgewezen. Verder maakt de dunne maar handige sidebar het erg makkelijk om de gewenste pagina te navigeren.

# Database

We hebben voor onze website 3 databases gebruikt. Om het kort te vertellen hebben we er een voor de gebruikers, eentje voor de kaarten die iedereen bezit en eentje voor de market. Natuurlijk kan het veel simpeler en in 1 database met 3 linked tabellen. Helaas hebben we niet kunnen vinden hoe dit werkt met SQLAlchemy. Wel is dit iets wat we graag willen aanpassen als we besluiten verder te werken aan de website.

## Users

Deze database is waarschijnlijk de belangrijkste. Deze database zorgt er allereerst voor dat gebruikers kunnen inloggen door eerst te kijken of de ingevulde username in de database zit en daarna wordt er gechecked of het wachtwoord ook klopt bij de username. Alhoewel dit niet erg veilig is is een een goeie optie voor ons. Wij zijn natuurlijk nog niet zo ervaren met websites en python vooral als het gaat om beveiliging. Deze database heeft ook de hoeveelheid C-BUCKS van iedere gebruiker opgeslagen, dit is standaard 4000.

## Inventory

Deze database is vrij simpel. Iedere keer dat er gebruiker bij komt wordt er een nieuwe rij gemaakt met het aantal per kaart in de kolomen, achteraf gezien is dit niet de beste keuze omdat dit uiteindelijk zorgt voor een niet efficiënte manier van het opslaan van veel data. Zoals eerder vermeld hadden we dit pas later door en hadden dit graag anders gehad. Helaas weten wij nog steeds niet helemaal hoe.

## Market

In deze database worden de kaarten die te koop staan tijdelijk opgeslagen samen met de prijs en de aanbieder. Op het moment dat een kaart te koop wordt gezet wordt de kaart weggehaald van de verkoper, als de kaart wordt gekocht gaat deze uit de market database en naar de inventaris van de koper. Ook wordt het geld van de koper verminderd en wordt het bedrag bij de verkoper gestort.

## Reflectie

Deze website kostte veel tijd om te maken. We hebben veel gebruik moeten maken van online bronnen om dingen die we nog niet snapten alsnog toe te kunnen voegen. Dit maakte het project wel een stuk leuker om aan te werken. We hebben tussendoor ook de plannen een beetje aangepast vanwege de tijd dat het zou kosten en de mogelijkheid ervan. We hebben bijvoorbeeld het plan van het bieden op de kaarten moeten schrappen omdat het helaas niet haalbaar was in de tijd die we hadden gekregen. Tijdens het maken van de website kwamen we er achter dat het erg moeilijk is om data van python naar javascript en andersom te krijgen. In 1 instantie hebben wij hier dan ook een library voor gebruikt. De personages op de kaarten was ook nog een leuk extraatje. We hebben ieder onderdeel van de characters apart ontworpen en vervolgens via een programma alles laten combineren tot unieke plaatjes (de resultaten hiervan staan in een drive toegevoegd bij het bestand). Dit was een erg leuk proces, uiteindelijk hebben we zelfs alle onderdelen opnieuw ontworpen zodat het beter bij de site paste en zodat het allemaal de zelfde colorpalette gebruikte. Bovendien ziet het er een stuk beter uit.

Drive met extra bestanden [https://drive.google.com/drive/folders/1vGibwPfS0\\_6MqW70zU3Yvtj4KpWg9oU?usp=drive\\_link](https://drive.google.com/drive/folders/1vGibwPfS0_6MqW70zU3Yvtj4KpWg9oU?usp=drive_link)

# Logboek

We hebben tijdens het maken van de site geen logboek bijgehouden. Wel hebben we een aardig idee van wie wat heeft gedaan. We komen uit op ongeveer een verhouding van 60% door Nathan en 40% door Sven. Dit komt omdat we allebei er thuis erg veel aan hebben gewerkt, uiteindelijk gingen we gewoon thuis aan werken als we er zin in hadden, de percentages zijn dus niet gelijk gebleven.

## Code:

Nathan heeft zo'n 80% van de code geschreven, sven heeft de rest gedaan. Wel hebben Sven veel geholpen met pseudocode voor sommige functies.

## Assets:

Sven heeft de originele layers voor de characters ontworpen, later heeft Nathan deze layers een 'kleine' upgrade gegeven. Nathan heeft in blender de card packs en de C-BUCKS assets gemaakt voor de shop.

## Overig:

Sven heeft de gehele spreadsheet gemaakt waarin hij de de kaarten en het systeem voor de kaarten heeft ontworpen. Daarnaast heeft hij klad versie van ons verslag gemaakt en daarna uitgewerkt zodat Nathan het vervolgens in html en css kon schrijven als een extraatje.

## Extraatje

Ja u las het goed. Wij vonden het een leuk idee om het verslag in html te schrijven, past wel bij de opdracht. We hebben dit bereikt door een nieuw project aan te maken. In dit project hebben we het verslag geschreven. Elke pagina is een aparte div, door de class van deze div precies zo hoog te maken als een pagina van een pdf hebben perfect controle over de pagina en de style.