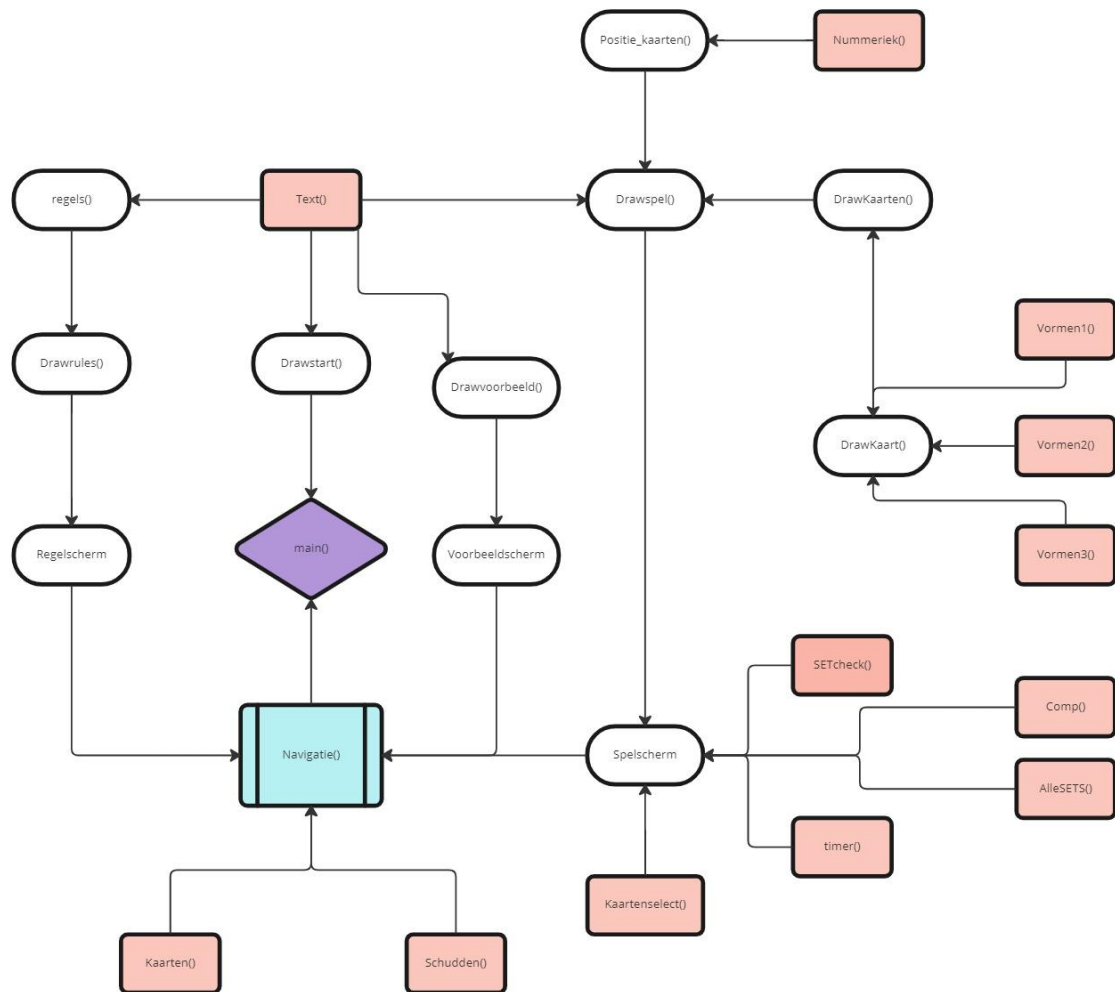


Datastructuur



In deze flowchart is te zien hoe de communicatie tussen alle functies werkt. Er is gekozen om zoveel mogelijk losse functies te schrijven die samengevoegd konden worden, zodat er makkelijker een probleem opgelost kon worden zodra deze ontstond. Vanwege de meeste lineaire afkomst van de functies was het makkelijk te achterhalen in welke functie het probleem zat, en dit helpt al best veel voor het fixen van een bug. Ook is er bijvoorbeeld een tekst functie gemaakt, omdat dit veel gebruikt werd en daardoor veel tijd heeft bespaard.

Handleiding

Code

De code werkt met gebruik van de datastructuur hierboven. Voor het startscherm wordt er gebruik gemaakt van een functie die alle tekst weergeeft op het scherm, waaronder onder

andere de toetsen die worden gebruikt voor de navigatie voor verdere dingen. Vervolgens wordt de rest gecontroleerd via de Navigatie() functie; deze functie voegt vele functies samen. Terwijl de speler zich bevindt op het startscherm zorgt de Navigatie() functie ervoor dat de speler een moeilijkheidsgraad kan kiezen; namelijk 1, 2 of 3. Bij 1 vindt de computer elke 45 seconden een SET, bij 2 elke 30 seconden en bij 3 elke 15 seconden. Ook geeft de Navigatie() functie vanaf het startscherm de mogelijkheid om naar het regelscherm te gaan of naar het spel zelf.

Als de speler naar het regelscherm gaat, wordt er opnieuw gebruik gemaakt van de tekst functie. Ook geeft de Navigatie() functie hier de mogelijkheid om terug te gaan naar het startscherm of door te gaan naar het voorbeeldscherm. In het voorbeeldscherm wordt opnieuw gebruik gemaakt van de tekst functie. De Navigatie() functie geeft de mogelijkheid terug te gaan naar het startscherm.

Als de speler naar het spelscherm gaat, worden er vele functies geactiveerd.

De Navigatie() functie geeft dan aan vele variabelen een waarde of reset de waarden als de speler zich al eerder in het spelscherm heeft bevonden. Ook activeert het de functies die alle 81 kaarten generen en schudden. Het kaarten generen wordt gedaan door 4 loops in elkaar, waarbij die steeds een eigenschap uit een lijst neemt en voegt hier aan alle

overige mogelijkheden van combinaties van andere lijsten aan toe. Het schudden gebeurt door middel van een functie die door middel van een loop steeds de kaart omwisselt met een random andere kaart; deze functie doet twee keer dezelfde loop, zodat de kaarten beter geschud zijn. Deze geschudde lijst kaarten is dus ook een van de variabelen die gecreëerd wordt. Vervolgens komt de Navigatie() functie in een while loop waarin deze de hele tijd door meerdere functies heen gaat.

De functie activeert het tekenen van het spelscherm. Dit maakt opnieuw gebruik van de tekst functie maar ook van de Positie_kaarten() functie en van de Drawkaarten() functie. De Drawkaarten() functie voert op basis van de lengte van de lijst met kaarten, kaarten en posities in de Drawkaart() functie die deze moet tekenen. De Drawkaart() functie maakt gebruik van de eigenschappen die in de lijst van de kaart staan. Met gebruik van if-statements achterhaalt de functie welke afbeelding deze moet gebruiken en welke van de vormen functies deze moet gebruiken. De vormen functie plaatst en verdubbelt indien nodig de afbeelding om de hoeveelheid van de kaart te matchen. De kaarten worden vervolgens weergegeven op het scherm.

De Positie_kaarten() functie is een functie die kijkt hoeveel kaarten er op het scherm komen/zijn en op basis hiervan met gebruik van de tekst functie toont de functie welke toetsen de speler moet gebruiken voor welke kaarten.

Een van de functies is de timer() functie; deze functie checkt of er 30 seconden voorbij zijn of niet. Zo niet, dan voegt de timer() functie een extra telling toe. Deze telling houdt rekening met dat de code op 60 FPS runt; dus met 30 seconden is dat een telling van 1800. Als er wel 30 seconden voorbij zijn, zoekt de functie drie random kaarten die niet op het scherm staan en vervangt de eerste drie kaarten op het scherm.

Er wordt een Kaarteselect() functie geactiveerd; dit is een functie die op basis van nummers van kaarten die de speler invoert, registreert en onthoudt welke dit zijn. Als deze functie terug geeft dat er 3 kaarten gekozen zijn, activeert de SETcheck() functie. Deze functie kijkt op basis van de eigenschappen van de geselecteerde kaarten of het een SET is of niet. Als het een CapSET is dan de-selecteren de kaarten en gebeurt er verder niks. Als er een SET is, krijgt de speler 3 punten, worden de kaarten uit de lijst weggehaald en wordt de timer op 0 gezet. Doordat de kaarten uit de lijst met kaarten worden weggehaald, voert de Drawkaarten() functie dus andere kaarten in Drawkaart() en veranderen ook de kaarten op het scherm. Deze functie bevat ook de reset toets en de toets om terug te gaan naar het startscherm. Als de reset toets geactiveerd wordt, doet de functie het volgende; hij reset alle variabelen behalve de hoogst scoren. Tevens runt hij opnieuw de Kaarten() en Schudden() functies zodat het dek kaarten wordt vervangen. De terug toets stopt alle functies in deze while loop van de Navigatie() functie.

Naast de speler input is er ook een input van de computer; de snelheid hiervan hangt af van de moeilijkheidsgraad gekozen door de speler (bij het startscherm). Hiervoor worden er twee functies gebruikt. De ene functie is de functie AlleSETS(); dit is een functie die door middel van 3 for loops langs alle kaartcombinaties op het scherm gaat en kijkt of deze een SET zijn. Zo ja is dan voegt de functie ze toe aan een lijst. Zo nee dan gaat de functie door met zoeken. De andere functie heet Comp(); dit is een functie die ook tijd bijhoudt. Deze tijd hangt af van de moeilijkheidsgraad gekozen bij de speler: 1 - 45 seconden, 2 - 30 seconden, 3 - 15 seconden. Deze functie checkt of deze tijd voorbij is. Zo nee dan telt de tijd door. Zo ja, en er staan SETS op het scherm dan kiest de computer een random SET en deze wordt uit de lijst van kaarten gehaald. Ook krijgt de computer dan 3 punten erbij.

Spelen

Elke kaart heeft 4 eigenschappen, te weten een hoeveelheid, een kleur, een vorm en een vulling. Er zijn drie varianten van elke eigenschap. Een SET bestaat uit 3 kaarten waarvan voor alle 4 eigenschappen afzonderlijk geldt dat de variant ervan precies gelijk of volledig verschillend moeten zijn. Bij elke combinatie van 3 kaarten stel je jezelf steeds voor elke eigenschap apart de vraag: is de variant van deze

eigenschap op alle kaarten precies gelijk of volledig verschillend? Kan je deze vraag voor alle eigenschappen met JA beantwoorden, dan heb je een set gevonden.

Uit handleiding SET, Marsha J. Falco

Je speelt het spel door gebruik te maken van de navigatie met de toetsen van het keyboard. Op de meeste plekken staat wat een bepaalde toets doet als je erop klikt. Op alle kaarten staat linksboven in de hoek welke knop nodig is om deze kaart te kiezen. Wanneer er drie kaarten gekozen zijn, checkt het spel of het een SET is en haalt het programma de set weg.