Testplan

TINPRJ0334 - Bank

N. Vreugdenhil-van Dormolen |Versie 1 | Eerste kans

Taylor Wernet(0948313)  
0948313@hr.nl

Marouan El Yachioui(1014860)  
1014860@hr.nl

Frank van Etten(1009269)  
1009269@hr.nl

Connor Cullens(1011010)  
1011010@hr.nl

Inhoudsopgave

[Testopstelling 3](#_Toc63116787)

[Test 1 3](#_Toc63116788)

[Test 2 3](#_Toc63116789)

[Test 3 3](#_Toc63116790)

[Test 4 4](#_Toc63116791)

[Uitvoer testen 4](#_Toc63116792)

[Testen 4](#_Toc63116793)

[Test 1.1 4](#_Toc63116794)

[Test 1.2 4](#_Toc63116795)

[Test 2.1 4](#_Toc63116796)

[Test 2.2 4](#_Toc63116797)

[Test 2.3 4](#_Toc63116798)

[Test 2.4 5](#_Toc63116799)

[Test 3.1 5](#_Toc63116800)

[Test 3.2 5](#_Toc63116801)

[Test 4.1 5](#_Toc63116802)

# Introductie

In deze testplan worden alle aspecten die getest gaan worden eerste geanalyseerd en vervolgens wordt de testmethode die gehandhaafd gaan worden gedocumenteerd. Tevens wordt van elke test een rapport opgesteld waar wordt weergegeven of de geteste onderdeel naar behoren werkt.

# Testopstelling

De volgende testen gaan behandeld worden:

|  |  |
| --- | --- |
| Test | Omschrijving |
| 1.1 | De kaartlezer moet de kaartnummer geven die wij vervolgens in de database kunnen gebruiken. |
| 1.2 | De bonprinter moet gegevens uitprinten dat kan illustreren welke gegevens er uitgeprint moet worden bij het pinnen. |
| 1.3 | Er moet met de keypad in combinatie met de kaartlezer een bepaalde pincode ingevoerd kunnen worden om te verifiëren of het correct werkt. |
| 2.1 | De gelddispenser moet fysiek sterk zijn, het mag niet kapot gaan als er tegenaan wordt gestoten of geld eruit vallen. |
| 2.2 | De gelddispenser moet 10 van de 10 keer werken bij het uitgeven van geld, het mag niet meerdere briefjes tegelijk geven. |
| 2.3 | De geldautomaat moet robuust zijn, het mag niet door een stoot stuk gaan. |
| 2.4 | De geldautomaat moet zo ontworpen zijn dat je het geld niet kan bereiken. |
| 3.1 | De user interface moet gebruiksvriendelijk zijn en duidelijk voor de gebruiker. |
| 3.2 | De user interface moet in samenwerking met de kaartlezer en keypad naar de volgende schermen kunnen gaan. |
| 3.3 | De user interface moet in samenwerking met de database en de API gegevens van de database kunnen halen. |
| 3.4 | De knoppen die in de user interface zijn geïmplementeerd moeten de juiste functie hebben en op de correcte manier werken. |
| 3.5 | De pas moet na 3 foute pincodes geblokkeerd worden in de database. |
| 3.6 | De taal kan gewijzigd worden bij het drukken van een knop. |
| 4.1 | De arduino moet input kunnen sturen naar de user interface en vervolgens moet de user interface gegevens kunnen versturen en ontvangen van de database. |
| 4.2 | Bij het intoetsen van de pincode moet de database controleren of het ingevoerd pincode correct is. |
| 4.3 | Bij het invoeren van een bedrag moet er gecheckt kunnen worden in de database of er genoeg saldo is. |
| 4.4 | Bij het invoeren van een bedrag moet er een logische keuze komen van de biljetten dat de gebruiker kan kiezen. |
| 4.5 | Bij het afronden van de transactie moet er een bon uitgeprint kunnen worden waar er informatie over de transactie staat. |
| 5.1 | Op het moment dat de database in de server is moet de user interface dezelfde werking behouden als in de lokale database. |
| 5.2 | Er moet gepind kunnen worden bij andere banken binnen ons land. |
| 5.3 | Er moet gepind kunnen worden bij andere banken in een ander land. |

# Testmethode

##### Test 1.1

De kaartlezer wordt verbonden met de arduino, om te zien hoe de arduino met de kaartlezer is verbonden kijk naar het bestand (bank elektrische schema.png). Vervolgens wordt er een code geupload in de arduino die ervoor zorgt dat de rfid de kaart kan lezen en er een bepaalde gegeven in de seriele monitor komt.   
dit is de code die er gebruikt gaat worden:

#include <SPI.h>  
#include <MFRC522.h>

#define SS\_PIN 10  
#define RST\_PIN 9

MFRC522 mfrc522(SS\_PIN, RST\_PIN);

void setup() {

Serial.begin(9600);

SPI.begin();

mfrc522.PCD\_Init();

}

void loop() {

if ( ! mfrc522.PICC\_IsNewCardPresent()) {

return;

}

if ( ! mfrc522.PICC\_ReadCardSerial()) {

return;

}

Serial.print("UID tag:");

String content = "";

byte letter;

for (byte i = 0; i < mfrc522.uid.size; i++) {

Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");

Serial.print(mfrc522.uid.uidByte[i], HEX);

content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));

content.concat(String(mfrc522.uid.uidByte[i], HEX));

}

Serial.println();

content.toUpperCase();

if (content.substring(1) == "E4 0D 2E 2A") { //Pas kan verandert worden naar wat jouw pas is

Serial.println("SUCCES");

}

else {

Serial.println("FAILED");

}

}

##### Test 1.2

De bonprinter wordt verbonden met de arduino om te zien hoe de arduino met de bonprinter is verbonden kijk naar het bestand (bank elektrische schema.png). Verder wordt er een code naar de arduino geupload om te kijken of de bonprinter de juiste waardes uitprint.

Dit is de code dat er gebruikt gaat worden:

#include "Adafruit\_Thermal.h"  
#include "SoftwareSerial.h"  
#define TX\_PIN 6  
#define RX\_PIN 5

SoftwareSerial mySerial(RX\_PIN, TX\_PIN);  
Adafruit\_Thermal printer(&mySerial);

void setup() {  
pinMode(7, OUTPUT); digitalWrite(7, LOW);

mySerial.begin(9600);  
printer.begin();

printer.justify('C');  
printer.setSize('L');  
printer.boldOn();  
printer.println("MoneyMakers");  
printer.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

printer.justify('L');  
printer.setSize('S');  
printer.setLineHeight(30);  
printer.println("Terminal: 3");  
printer.println("Account: XXXX XXXX XXXX ..");  
printer.println("Transactie: 00000001");

printer.setSize('M');  
printer.println("BETALING");  
printer.boldOff();  
printer.setSize('S');  
printer.println("'Datum & Tijd'");

printer.setSize('M');  
printer.boldOn();  
printer.println("Totaal: 'Prijs' EUR");  
printer.justify('C');  
printer.setSize('L');  
printer.println("SUCCES");  
printer.boldOff();  
printer.feed(3);  
  
printer.sleep();  
delay(3000L);  
printer.wake();  
printer.setDefault();  
}

void loop() {  
}

##### Test 1.3

De kaartlezer en de keypad wordt verbonden met de arduino om te zien of de pincode die geprogrammeerd is in de arduino kan worden ingevoerd en geverifieerd. Ook kan er vervolgens worden gekeken bij welke kaart de ingevoerde pincode hoort en of de pincode juist is ingevoerd. Om te zien hoe de arduino met de kaartlezer en de keypad is verbonden kijk naar het bestand (bank elektrische schema.png).

Dit is de code die gebruikt gaat worden:

#include <SPI.h>  
#include <MFRC522.h>  
#include <Keypad.h>  
#include <Wire.h>

typedef enum {  
State\_Card,  
State\_Pin  
}  
current\_state;  
current\_state state;

#define SS\_PIN 10  
#define RST\_PIN 9

MFRC522 mfrc522(SS\_PIN, RST\_PIN);

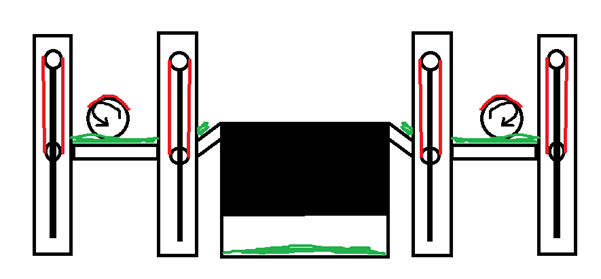
const byte ROWS = 4;  
const byte COLS = 4;  
char hexaKeys[ROWS][COLS] = {  
{'1', '2', '3', 'A'},  
{'4', '5', '6', 'B'},  
{'7', '8', '9', 'C'},  
{'\*', '0', '#', 'D'}  
};  
byte rowPins[ROWS] = {8, 7, 6, 5};  
byte colPins[COLS] = {4, 3, 2, A0};  
Keypad customKeypad = Keypad(makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);

const String password;  
String input\_password;  
int wrong = 0;  
int counter = 0;

void setup() {  
Serial.begin(9600);  
SPI.begin();  
mfrc522.PCD\_Init();  
input\_password.reserve(32); //Maximale input van 32 characters voordat je niks meer kan invullen.  
}

void loop() {  
switch (state) {  
case State\_Card: {  
if ( ! mfrc522.PICC\_IsNewCardPresent()) {  
return;  
}  
if ( ! mfrc522.PICC\_ReadCardSerial()) {  
return;  
}  
Serial.println("UID card: ");  
String content = "";  
byte letter;  
for (byte i = 0; i < mfrc522.uid.size; i++) {  
Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");  
Serial.print(mfrc522.uid.uidByte[i], HEX);  
content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));  
content.concat(String(mfrc522.uid.uidByte[i], HEX));  
}  
Serial.println();  
content.toUpperCase();  
if (content.substring(1) == "E4 0D 2E 2A" || content.substring(1) == "14 F1 D4 2B" || content.substring(1) == "C9 0C 72 C2" || content.substring(1) == "99 4A D1 B9") { // Marouan || Connor || Frank  
if (content.substring(1) == "E4 0D 2E 2A") {  
password = "1234"; //Wanneer deze pas wordt herkend geeft moet de wachtwoord 1234 worden ingevuld.  
}  
else if (content.substring(1) == "D3 3A 85 16") {  
password = "2580"; //Wanneer deze pas wordt herkend geeft moet de wachtwoord 0000 worden ingevuld.  
}  
state = State\_Pin; //Als de pas wordt geaccepteerd mag die zijn pincode invullen.  
Serial.println("Enter password: ");  
}  
else {  
Serial.println("Access denied"); //Bij een verkeerde pas geeft die dit aan. (Er kan geen pincode ingevoerd worden.)   
}  
}  
break;  
case State\_Pin: {  
char pin = customKeypad.getKey();  
if (pin) {  
if (pin != '#' && pin != '\*') {  
Serial.println("\*"); //Als er een pin wordt ingetoets komt er een \* in zijn plaats te staan. Verander ("\*") naar (pin) om de getallen te kunnen zien.  
}  
if (pin == '\*') {  
input\_password = "";  
}  
else if (pin == '#') {  
counter++; //Telt hoevaak er is geprobeerd om in te loggen.  
//Serial.print("COUNT: ");  
//Serial.println(counter);  
if (password == input\_password) {  
Serial.println("Correct password");  
wrong = 0; //Als het wachtwoord klopt wordt de teller van foute wachtwoorden terug naar 0 gezet.  
state = State\_Card;  
}  
else if (password != input\_password) {  
wrong++; //Telt hoevaak het wachtwoord verkeerd is ingevuld.  
}  
if (wrong == 1) {  
Serial.println("Incorrect password, 2 tries left!");  
}  
else if (wrong == 2) {  
Serial.println("Incorrect password, 1 try left!");  
}  
else if (wrong == 3) {  
Serial.println("Card is blocked"); //Bij 3 foutmeldingen wordt de pas 'geblokkeerd'. (Nu wordt je alleen uitgelogd.)  
state = State\_Card;  
}  
input\_password = "";  
}  
else {  
input\_password += pin;  
}  
}  
}  
break;  
default: {  
break;  
}  
}  
}

##### Test 2.1

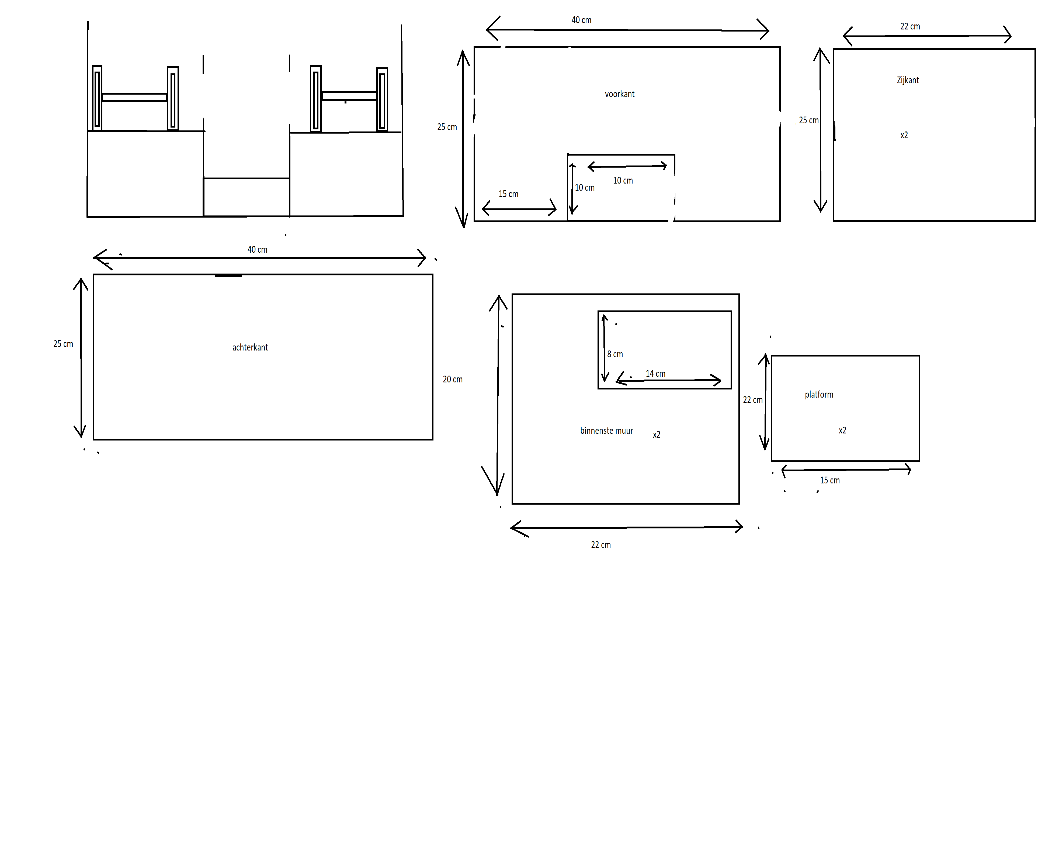
De gelddispenser moet op een manier gebouwd worden dat het niet kapot gaat of uit elkaar valt op het moment dat de geldautomaat een stoot krijgt als er iemand bijvoorbeeld er probeert in te komen. Daarvoor hebben wij een design gemaakt zoals te zien is in figuur 1. We gaan een gelddispenser maken die vier poten gaat hebben en vast gemaakt gaat zijn aan de onder en bovenkant, waardoor het stevig in het geldautomaat gaat zitten.

Figuur 1

##### Test 2.2

Het geld dat in de gelddispenser gaat vastgehouden worden door een motor die gaat draaien op het moment dat er geld moet worden gegeven. De motor gaat tegen een platform aanzitten die vastzit met elastieken. Die elastieken zorgt ervoor dat op het moment dat er geld ervan af gaat de platform omhoog wordt getrokken, dat zorgt voor de druk. Verder wordt er gekeken naar de aantal rotaties die de motor moet doen voordat er een geldbrief uit de gelddispenser valt. Verder kan het nog gecontroleerd worden met sensoren. De motoren en het platform wordt gemaakt zoals het in figuur 1 staat.

##### Test 2.3

De geldautomaat gaat balkvormig zijn, en in het geldautomaat worden twee verdiepingen gemaakt. Doordat er twee verdiepingen zijn wordt het geldautomaat steviger, ook wordt het geldautomaat van hout gemaakt dat was het sterkste materiaal dat we konden krijgen. De design van het geldautomaat is te zien in figuur 2.

##### Test 2.4

De geldautomaat gaat een kleine gat hebben in het midden waar de gebruiker het geld van kan pakken. Verder wordt er een helling in de geldautomaat gemaakt waar het geld op gaat komen en naar de gebruiker toe gaat voortschuiven, vervolgens kan de gebruiker dat het geld makkelijk uit het geldautomaat pakken. Dus er gaat geen mogelijkheid zijn om het geld te kunnen pakken omdat er constructies in het geldautomaat zijn die het belemmert.

##### Test 3.1

In de user interface worden er knoppen weergegeven die overeenkomen met de knoppen die op de keypad zijn. Verder wordt er bij elke scherm instructies gegeven van wat er moet gebeuren en wat de gebruiker in het scherm kan doen. In figuur 3 kan je een voorbeeld zien van hoe de user interface eruit moet komen te zien en wat voor mogelijkheden er moeten zijn.

##### Test 3.2

De user interface gaat met samenwerking van de kaartlezer en keypad verbonden worden. Vervolgens wordt een code opgemaakt dat op het moment dat de kaartlezer wordt gelezen het verder gaat naar de volgende scherm. Verder wordt er op het volgende scherm een code gemaakt dat ervoor zorgt bij het invoeren van de juiste pincode, de user interface naar de volgende scherm gaat. Zo gaan de kaartlezer en keypad getest worden op de user interface.

##### Test 3.3

Er wordt een connectie gemaakt van de user interface naar de database. Verder wordt er een API gemaakt die ervoor zorgt dat op het moment dat bijvoorbeeld een kaart gelezen wordt op de arduino, er wordt gekeken of de nummer van de kaart overeenkomt met de nummer die in de database staat. Vervolgens kan er ook gekeken worden of de pincode die ingevoerd wordt, overeenkomt met de pincode die bij de desbetreffende kaart hoort.

##### Test 3.4

Om te kijken of de keypad de juiste input geeft wordt er natuurlijk is de nodige geprogrammeerd per scherm om het de juiste input waarde te geven. Verder wordt er per scherm doorlopen of we echt de juiste resultaat krijgen bij het drukken op de desbetreffende knoppen en of we de code moeten wijzigen om het op de juiste knop toe te wijzen.

##### Test 3.5

Er komt een counter in de database die aan gaat geven of de pas geblokkeerd is of niet. Op het moment dat de persoon de pincode verkeerd invoert, gaat de counter 1 omhoog totdat de counter op 3 komt dan wordt de kaart geblokkeerd. Als de counter nog niet op 3 is en de gebruiker voert de pincode goed in, dan wordt de counter gereset en begint de telling opnieuw.

##### Test 3.6

In de user interface wordt er een knop gezet die bij elke panel zichtbaar gaat zijn waarop staat naar welke taal je de user interface kan veranderen. Dat wordt naar een knop op de keypad toegewezen, als de gebruiker op de desbetreffende knop drukt, dan wordt de taal van de user interface veranderd naar de gevraagde taal.

##### Test 4.1

Op het moment Het is gelukt om de hardware in samenwerking met de user interface en de database in samenwerking met de user interface is gelukt, kan het geheel bij elkaar gezet worden en per user panel getest worden.

Test 4.2

Er wordt een code gemaakt dat eerst de kaart verifieert, vervolgens wordt er aan de hand van de kaartnummer gekeken of de pincode van de kaart correct is. Ook wordt er getest als er een andere pincode wordt ingetoetst of er wordt aangegeven dat je pincode incorrect is. Verder worden er ook meerdere kaarten getest om te kijken of het de user interface de juiste pincode pakt.  
Test 4.3

Er wordt een code gemaakt die ervoor zorgt dat de correcte saldo van de juiste gebruiker wordt gekozen daarmee kan je de saldo informatie zien. Verder wordt er in de user interface een code gemaakt die ervoor zorgt dat er wordt gekeken naar de saldo die een bepaalde gebruiker heeft en kijk of het mogelijk is om de ingevoerde bedrag uit te geven. Ook wordt er gekeken als de ingevoerde bedrag, het maximum bedrag niet overschrijdt.  
Test 4.4

Op het moment dat een bedrag wordt ingevoerd wordt er eerst gekeken welke biljetten er beschikbaar zijn voor het bedrag, vervolgens wat voor combinaties er mogelijk zijn. Je krijgt dan twee opties, of grote of kleine biljetten. Bij grote biljetten krijg je bijv. alleen 20 en de rest 10 en bij kleine biljetten krijg je alleen 10.  
Test 4.5

Op het moment dat de transactie is afgerond en de gebruiker heeft zijn geld gekregen wordt er een transactie bon uitgeprint. Op de transactiebon moet de transactienummer, de datum, hoeveel er gepind is en de laatste nummer van de iban-nummer staan. Die gegevens worden allemaal van de database afgehaald op het moment dat de transactie is voltooid  
Test 5.1

De database wordt in de server geplaatst, daarna wordt de verbinding getest naar de database zelf en wordt er gekeken of er toegang wordt verleent naar de informatie dat in de database zitten en of we die gegevens kunnen gebruiken en aanpassen. Vervolgens wordt de gegevens van de lokale database wijzigt de remote database.  
Test 5.2

Er wordt een afspraak gemaakt met de andere banken over de communicatie protocol die we gaan aanhouden. Verder wordt het gedocumenteerd en daaruit wordt er een gezamenlijke server gemaakt waar elke bank naar toe gaat verbinden en zijn gegevens naartoe gaat sturen op het moment dat het nodig is.  
Test 5.3

.