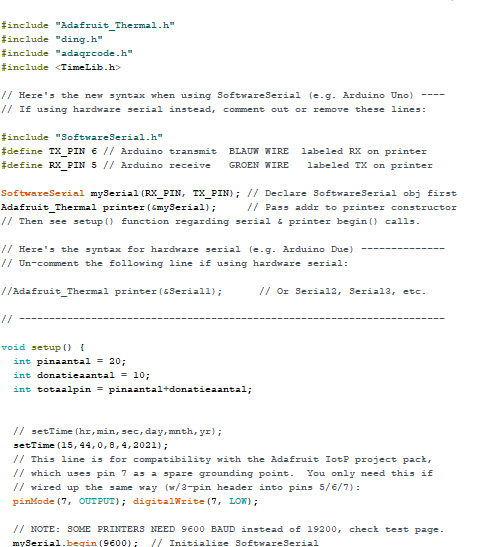
# Test plan en -rapport

1. Testen voor de bon printer.

*Verwachte resultaat:* Dat we de bon printer zonder fouten kunnen gebruiken.

*Waarneming:* Tijdens het opstellen van de bon printer hebben we via Arduino IDE een voorbeeld code kunnen pakken. We hebben getest met normaal printen en wat de geven waardes waren die we terug kregen. We gingen ook testen hoe we onze merk gingen afbeeldingen. Hiervan zijn dan de waardes aangepast op de variables die wij gebruiken voor het info wat we hadden gebruikt.

Hier is de code van de printer test.[[1]](#footnote-1)





Dit is oftewel de voorbeeld code van Arduino IDE. Om de rest te kunnen zien moet u naar onze GitHub gaan en dan code > alles > losse onderdelen. Hier staan alle codes.

*Conclusie:* De test is behaald. We kregen de waardes terug op het bonnetje wat eruit was geprint. Onze logo was ook met een paar aanpassingen duidelijk te zien.

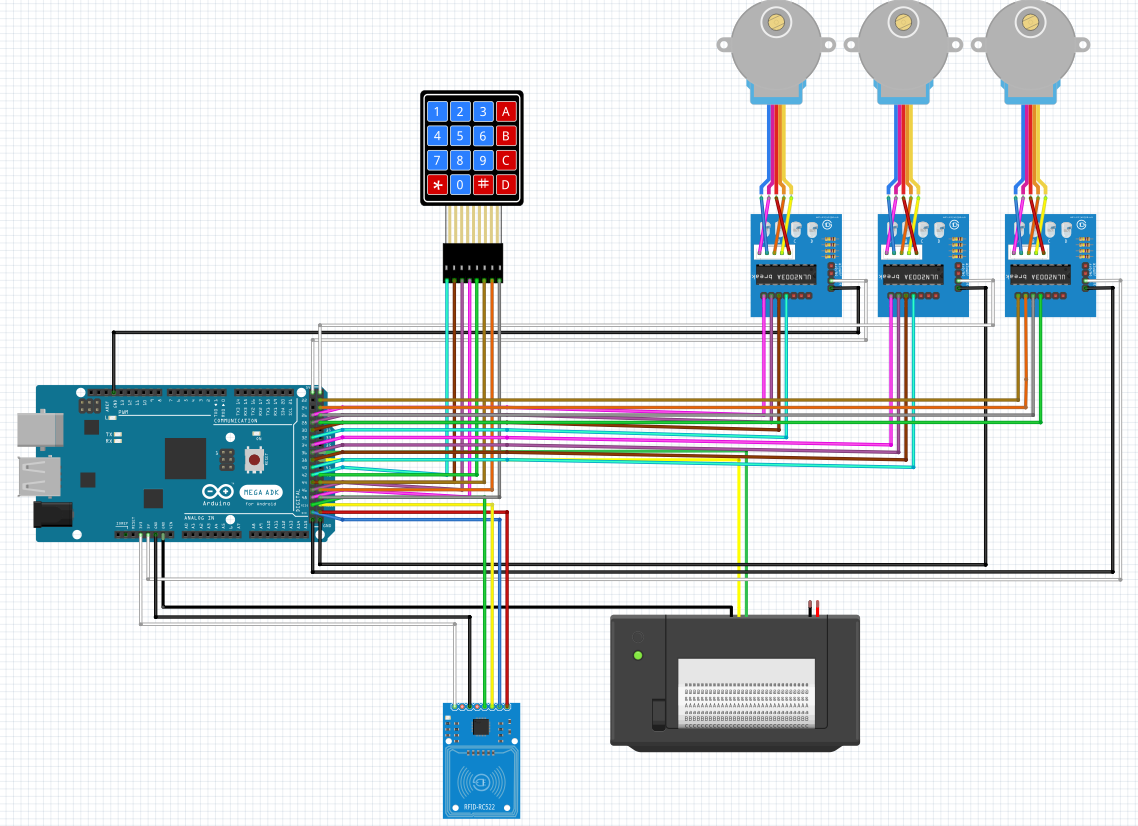
1. Testen op het samenvoeging van alle componenten.

*Verwachte resultaat:* Dat alle componenten vloeiend met elkaar kan werken.

*Waarneming:* We zijn vanuit gegaan dat we een Arduino Mega gaan gebruiken als bron. Daarna hebben we 3 dispenser bakken gemaakt, daarvan zijn 3 stappenmotoren in gebruik. De numpad is aangesloten, met de bon printer en de RIFD. Hiervan kregen we eerst data problemen. De data die er werd doorgestuurd via de Mega naar de printer was niet als te soepel. We kregen meestal errores. Dit was opgelost door Wouter en Jurgen die samen aan de componenten aan het testen waren.

We hadden ook problemen met de stappenmotoren gekregen, waardoor de verbinding niet perfect overkwam. Dit was opgelost door een vervanging te bestellen. Dit kwam namelijk door een defecte product.

De code is te groot om te delen via word. Dus we hebben dit in onze GitHub bij code>Alles staan. Hier is een voorbeeld hoe het eruit ziet in de fritzing.



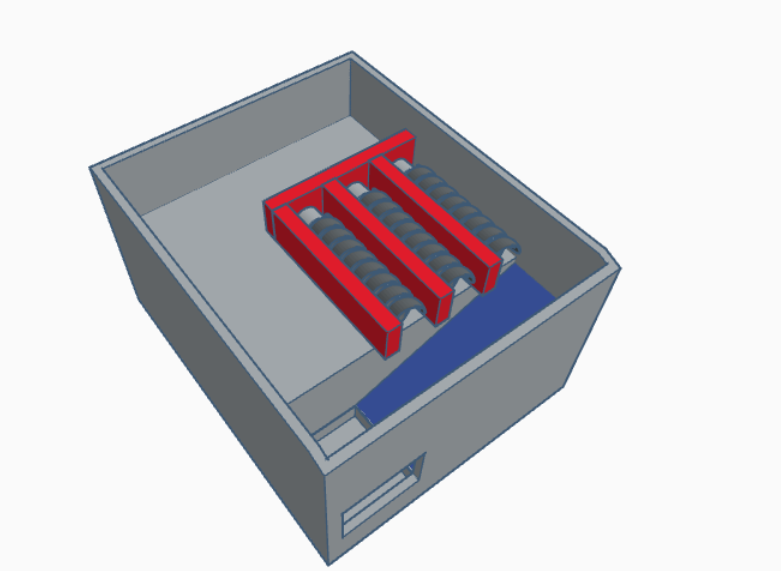
De numpad en de RIFD hadden geen problemen gemaakt, waardoor we hier niet hoefden te kijken.

*Conclusie:* De test is behaald. Alle componenten kunnen met elkaar communiceren. De communicatie levert geen problemen op. Hierdoor kunnen we het verbinden met de GUI.

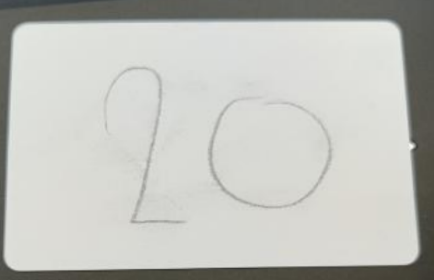
1. Testen voor de gelddispenser.

*Verwachte resultaat:* Dat 3 soorten biljetten uit onze dispenser kan worden gehaald.

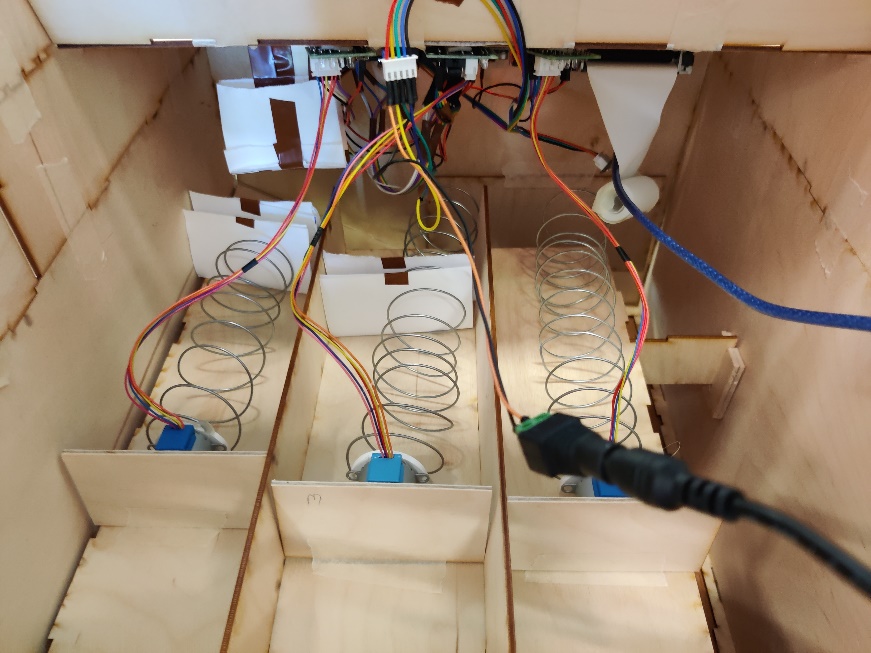
*Waarneming:* Eerst hadden we een proef ontwerp gemaakt en hoe we het gingen maken.



Als eerst hadden we een snoep automaat in gedachte. Hiervan gingen we vanuit dat biljetten de zelfde manier kunnen gebruiken. Dus we hebben de biljetten afgemeten. We gebruikte hierbij biljetten van 10, 20 en 50 euro. De afmetingen waren 9.9 x 5.1 cm.



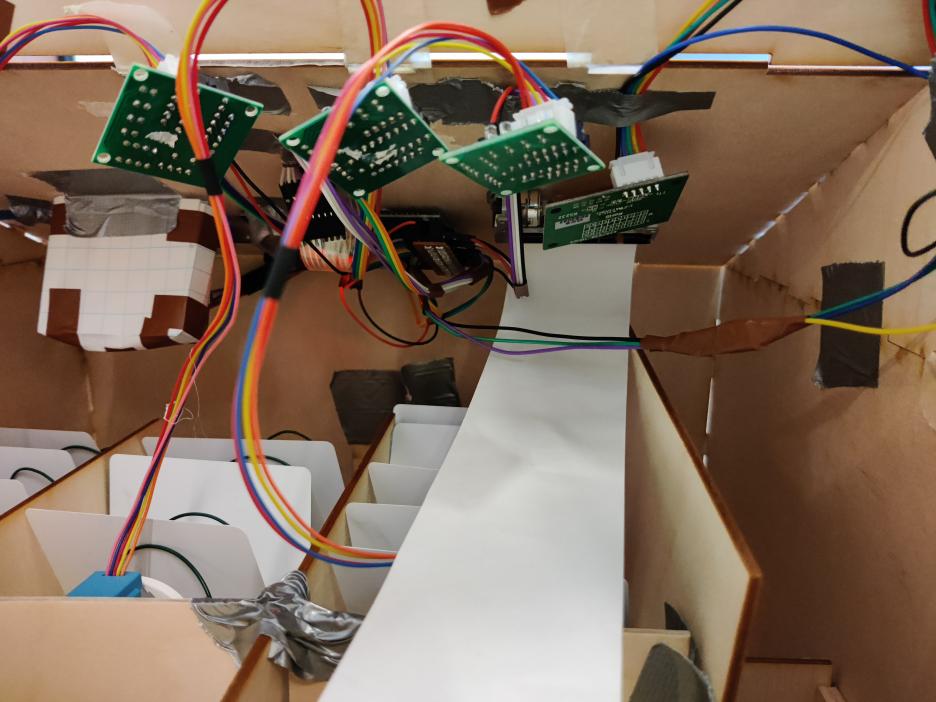
De 3 geldispensers hebben we in het onderste deel van onze ATM gestopt. Ze bevinden zich naast elkaar.



De biljetten vallen via een schuine baan naar het uitgang van het ATM behuizing.

Tijdens de 6e scrum meeting hadden we een nieuw ontwerp gemaakt voor de dispenser behuizing, want wat er nu boven in de afbeelding wordt weergeven is een prototype.

We zijn dan gaan testen om de biljetten soepel te laten vallen. De biljetten zou dan de schuine baan naar het uitgang toe glijden.

We hebben hier een nieuw inhoud.

We hebben ook een filmpje in onze GitHub staan, waarvan de werking van de dispenser zit.

*Conclusie:* Test is behaald. De dispenser kan nu soepel de biljetten verwijderen. Hiervan is ook het uitgangspunt beveiligd door een snoep automaat klep als voorbeeld te nemen.

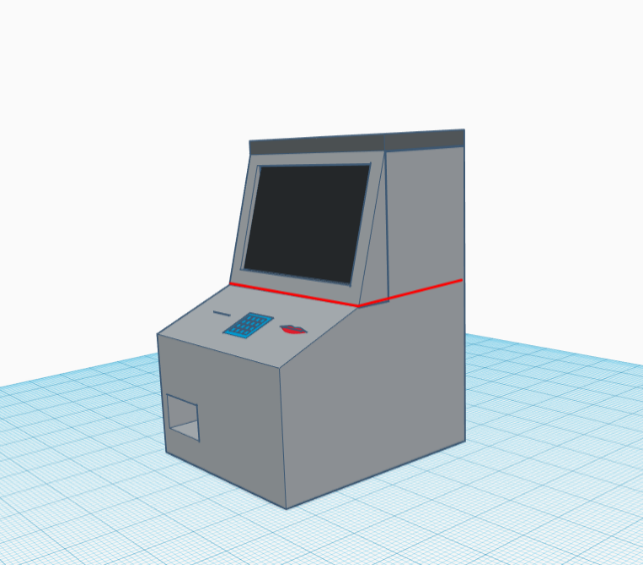
1. Testen voor het ATM behuizing.

*Verwachte resultaat:* Dat we een behuizing hebben voor onze ATM die aan alles voldoet.

*Waarneming:* Bij het maken van de ATM zijn we eerst met afmetingen vanuit gegaan van Jurgens laptop. We gebruiken hierbij Jurgens laptop scherm als het scherm van de ATM. Hier baseren wij de werking van de GIU. De afmetingen hebben we in ons GitHub documenten staan via tinkerscad.

Daarna is Wouter via adobe editor de afmetingen van de ATM aan het maken. Hierbij was het dikte van de plank minstens 3mm de rest zijn ook in de documenten te vinden. We hebben uiteindelijk de ATM in 2 delen gesplist. Het was anders echter te groot voor ons. Hiervan is het onderste deel van de dispenser en het bovenste deel de rest van de componenten.

Uiteindelijk is Jia gaan laser cutten. Waarvan we een fout tegen kwamen. Dit waren de bovenste zijkanten van onze ATM. Ze waren echter te klein voor het formaat. Dit moest opnieuw gesneden worden.

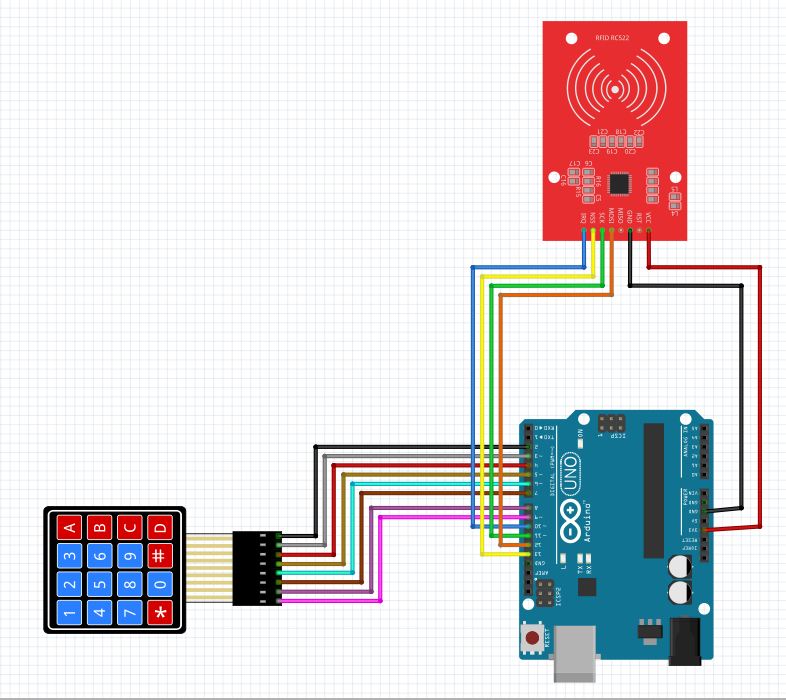
De afmetingen staan op tinkercad, die is gelinkt via GitHub. Verder zitten de laser cut afmetingen in de map documenten > ATM laser cut. Hier is het ontwerp van onze ATM in tinkerscad: 

*Conclusie:* Test is behaald. Het ATM is eindelijk een complete set geworden, waarvan alle componenten erin zitten.

1. Afhankelijke testen voor de keypad en RIFD

*Verwachte resultaat:* Testen op defecte componenten en dat er een code gemaakt wordt.

*Waarneming:* We gingen vooral kijken of er geen defecte keypad of een stappenmotor hadden. Dit deden we door een voorbeeld code op Arduino IDE te pakken.

De code staat in onze GitHub onder code > alles > losse onderdelen > keypaden en RIFD.

Uiteindelijk waren de componenten goed. Dus we hoefden geen nieuwe te halen. De werking voor beide componenten waren via Arduino al makkelijk te maken. Dit is dan via de waardes aangepast.

*Conclusie:* Test is geslaagd. Er waren geen fouten tijdens het verbinden van 2 componenten. Er waren ook geen defecte componenten langs gekomen. Dus een succesvolle test.

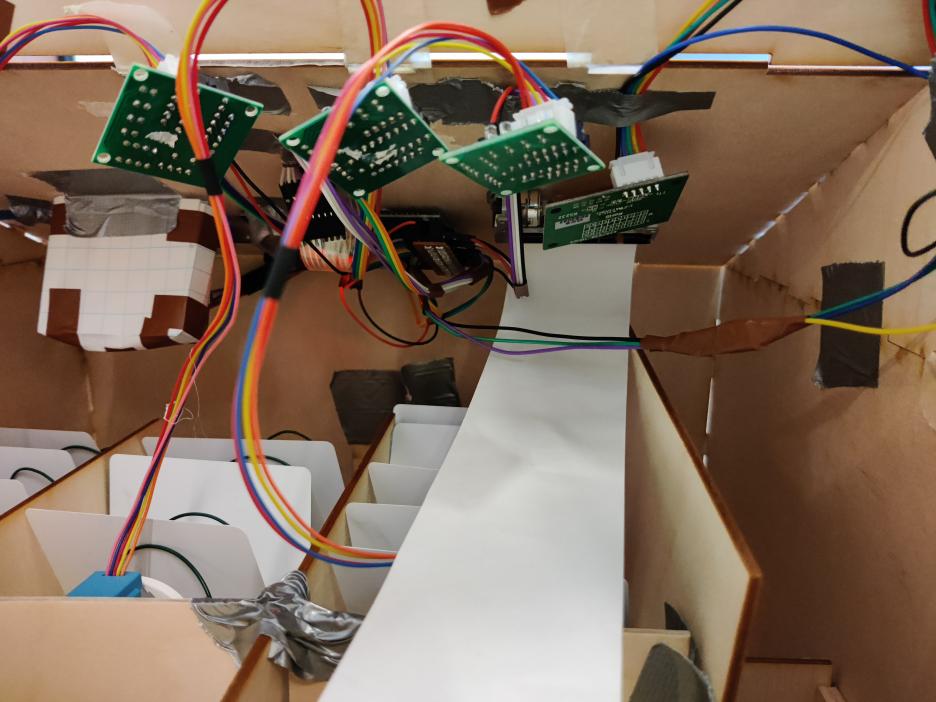
1. Testen op veiligheid voor de ATM.

*Verwachte resultaat:* Zorgen dat onze ATM beveiligd is op aanvallen van een dief.

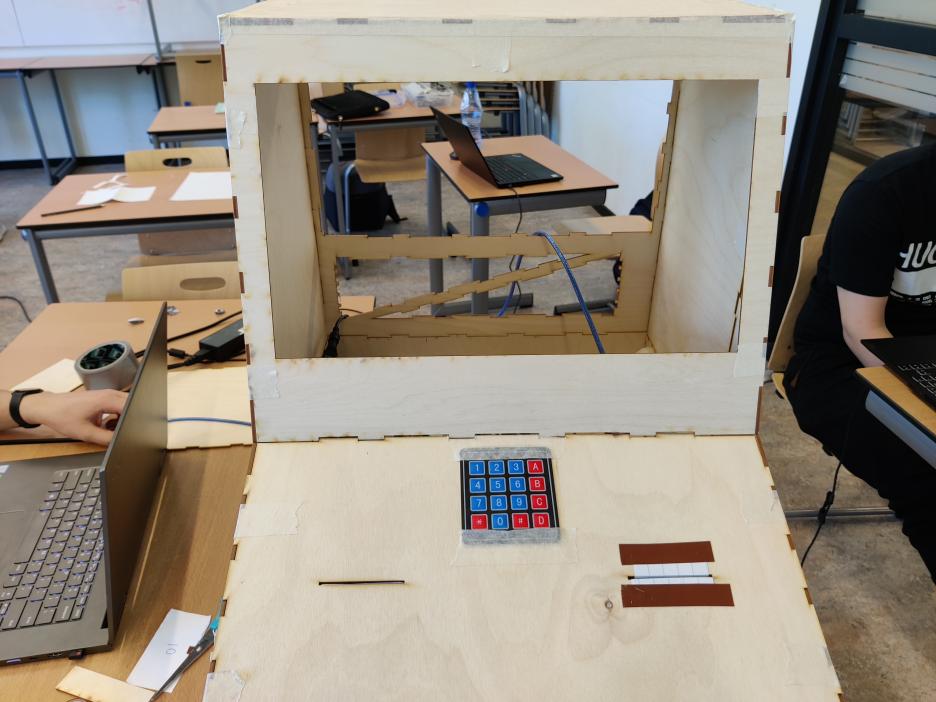
*Waarneming:* Voor deze test moesten we onze ATM zo veilig mogelijk maken. Dieven mogen dus niet direct in aanraking komen met onze bron componenten van het systeem. Dus voor het eerste component gaat het om de dispenser. De dief mag niet met zijn hand bij de biljetten komen. We hebben hiervoor een voorbeeld genomen van een snoep automaat. Een snoep automaat kan dus de aanvaller tegen houden om een uitgang klepje toe te voegen. Het is alleen via het opening naar binnen te gaan. Dus voor onze dispenser hebben we een houten klepje toegevoegd.



Daarna zijn alle componenten goed verbonden met elkaar. Er is met kabel management een duidelijk inzicht.



De buitenkant is niet bereikbaar voor de aanvaller. Hier is een proef voorkant foto.



Daarna hebben we nog een robuust video in onze GitHub folders.

*Conclusie:* De test is behaald. Onze ATM is nu wel veilig tegen de aanvallers. Hierdoor zal de klant veilig kunnen pinnen.

1. Testen op een complete database.

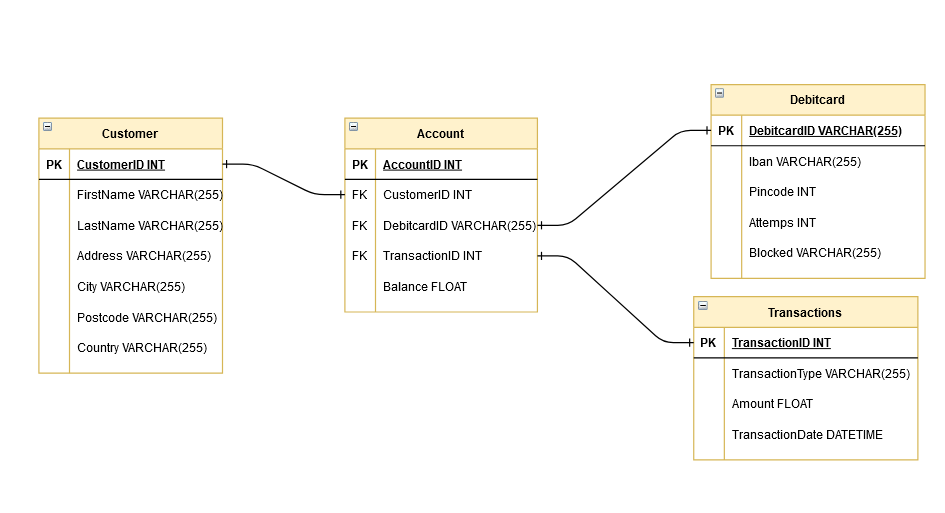
*Verwachte resultaat*: De database werken op de bank server.

*Waarneming*: In de eerste week hadden we een test database opgericht. Hiervan hadden we een gratis database op internet gevonden die we konden gebruiken. Om het inhoud te checken staat het op GitHub onder documenten.

Hiervan hadden we een API in elkaar gezet die dan met onze GUI communiceert.

We kunnen dus met de GUI de waardes aanpassen.

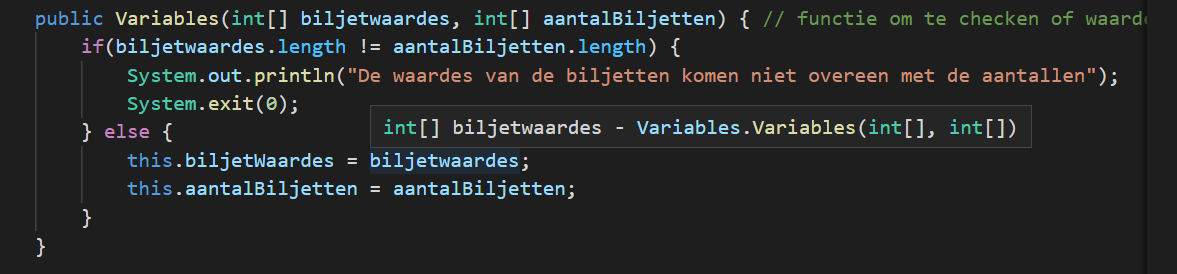
De waardes staan op GitHub en we hebben onze foto’s en filmpjes laten zien.



*Conclusie*: Test is behaald. Er is ook een filmpje in de documenten.

1. Testen om saldo checken.

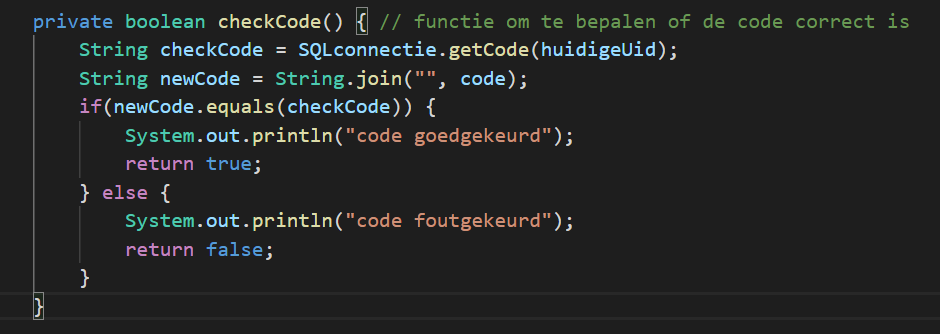
*Verwachte resultaat*: Dat de gebruiker zijn saldo kan zien wanneer hij/zij inlogt.

*Waarneming*: We hadden alles Java gemaakt. Hiervan hebben we de code om de waarde van alles te kunnen zien. Dit is direct verbonden met de server.

*Conclusie*: Deze test is behaald. De saldo kan gecheckt worden.

1. Testen op een voor pincode.

*Verwachte resultaat*: Dat de aantal pogingen worden onthouden en geblokkeerd.

*Waarneming*: In onze code houden we vast hoeveel pogingen er zijn geweest en of het goed is. De rest is verbonden met de database.

*Conclusie*: De test is behaald. Er is ook een filmpje hiervan.

1. Testen dat de beveiliging van onze GUI.

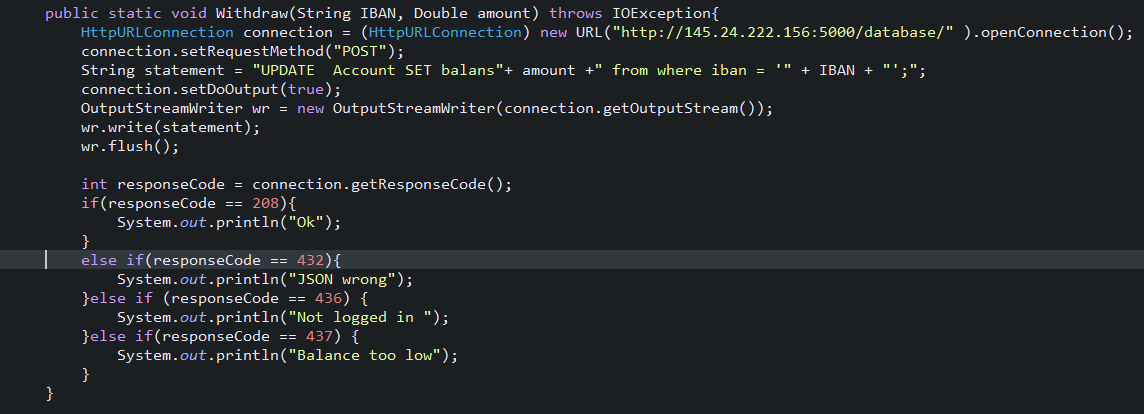
*Verwachte resultaat*: De GUI is beveiligd door een SSH en encryptie.

*Waarneming*: Onze bank weergeeft geen Iban of pincode. Dus dit is beveiligd dat de aanvaller geen zicht heeft op de gebruiker gegevens.

*Conclusie*: Deze test is behaald en is goed beveiligd.

1. Testen op de API

We kunnen de API testen met een de functies. Bijvoorbeeld wij willen 50 halen en die stuurt een request aan de landserver en we krijgen een get requests. We kunnen dus communiceren met de GUI via de API en kunnen pinnen bij andere banken.



Output van onze console er werd een get request terug gekregen via de API.



*Verwachte resultaat*: Wij onze API stuurt de correcte response en krijgen de correcte response.

*Waarneming*: Er werd een correcte response gekregen via de API en dus de correcte reponse werd gestuurd via onze API.

*Conclusie*: Deze test is behaald de API werkt correct.

1. [↑](#footnote-ref-1)