

Project 4

YARAN VAN DAM 0955631

inhoudsopgave

- ▶ Inleiding
- ▶ Ontwerp
- ▶ Kwaliteitseisen
- ▶ aanbeveling

Doel van het project

- ▶ Het doel van het project is het ontwerpen en inrichten van de centrale bank

Mijn ontwerpproces

- ▶ In de volgende paar slide ga ik het hebben over mijn proces en aan welke punten ik heb gedacht tijdens het ontwerp van de centrale bank ook ga ik het hebben over dingen die ervoor konden zorgen dat het project niet goed verliep dit houdt in:
- ▶ Risicolog
- ▶ Issue log
- ▶ kwaliteitseisen



Risicolog

#	Risico Beschrijving	Kans	Impact	Risico*	Maatregel	😊	Status Omschrijving	Datum
R1	Tijdsnood	3	5	15	Goed plannen Niet slapen	:	Er is een planning gemaakt.	25-4-18
R2	Veel lessen vallen uit	4	3	12	Meer vrije tijd er in steken	:	Er is een planning gemaakt.	25-4-18
R3	Server valt uit.	1	5	5	Contact leggen met hosting bedrijf	N	Hij werkt nog.	25-4-18
R4	Componenten zijn kapot	1	2	2	Nieuwe componenten regelen	N	Alle componenten werken nog.	25-4-18

Kans: schaal 1 (klein) t/m 5 (zeer groot)
Impact: schaal 1 (zeer lage) t/m 5 (zeer hoge)
Risico: = kans * impact
😊: [status] :) opgelost, :| bezig, :(niet opgelost; N nieuw

Issuelog

- ▶ <https://github.com/YaranvDam/project4> link naar mijn github en hierin is ook mijn vooruitgang op het project te zien

#	<u>issue</u>	<u>datum</u>	<u>verantwoordelijk</u>	 <u>status</u>	<u>beschrijving</u>	<u>datum</u>
1	Opdracht is niet goed begrepen	9-5-2018	<u>ik</u>	 <u>opgelost</u>	Het was niet duidelijk dat we de centrale bank alleen moesten ontwerpen	14-5-2018

Kwaliteit voor de code

- ▶ Gekozen voor leesbaarheid en onderhoudbaarheid
 - ▶ Technologie verandert snel en als bank wil je mee kunnen gaan met de tijd. onderhoudbaarheid zorgt ervoor dat je makkelijk veranderingen kunt aanbrengen.
 - ▶ Leesbaarheid zorgt er ook voor dat de code makkelijk te onderhouden is

Andere alternatieven

- ▶ performance

- ▶ De code zou geen last moeten hebben van slechte performance en is dus geen punt waar we tijd in zouden moeten steken

Kwaliteit voor het product

- ▶ Gekozen voor security en privacy
 - ▶ Belangrijk als het gaat om een bank en geldzaken
 - ▶ De gegevens zijn nogal kwetsbaar en zouden nooit gehackt mogen worden

Andere alternatieven

- ▶ Uitbreidbaarheid betrouwbaarheid

- ▶ Voordelen

- ▶ Technologie groeit snel en als bank wil je meegaan met de tijd
 - ▶ Betrouwbaarheid is belangrijk voor het behouden van klanten

- ▶ Nadelen

- ▶ De groei is niet makkelijk te voorspellen
 - ▶ Betrouwbaarheid kan je moeilijk krijgen bij het maken voor het product. Betrouwbaarheid krijg je vooral door geen security fouten te maken

Kwaliteit voor het gebruik

- ▶ Gekozen voor het effectiviteit
 - ▶ Gebruikersvriendelijk
 - ▶ Makkelijk toe te passen door bijvoorbeeld een simpel design te hebben die makkelijk te is om te begrijpen.

Andere alternatieven

▶ Vertrouwen

▶ Voordelen

- ▶ Belangrijk als het draait om het succes van de bank. Als er meer mensen vertrouwen hebben in de bank krijg je meer klanten.

▶ nadelen

- ▶ Vertrouwen is moeilijk om te krijgen door het verbeteren van gebruik maar is beter te krijgen door goede security en niet gehackt te zijn

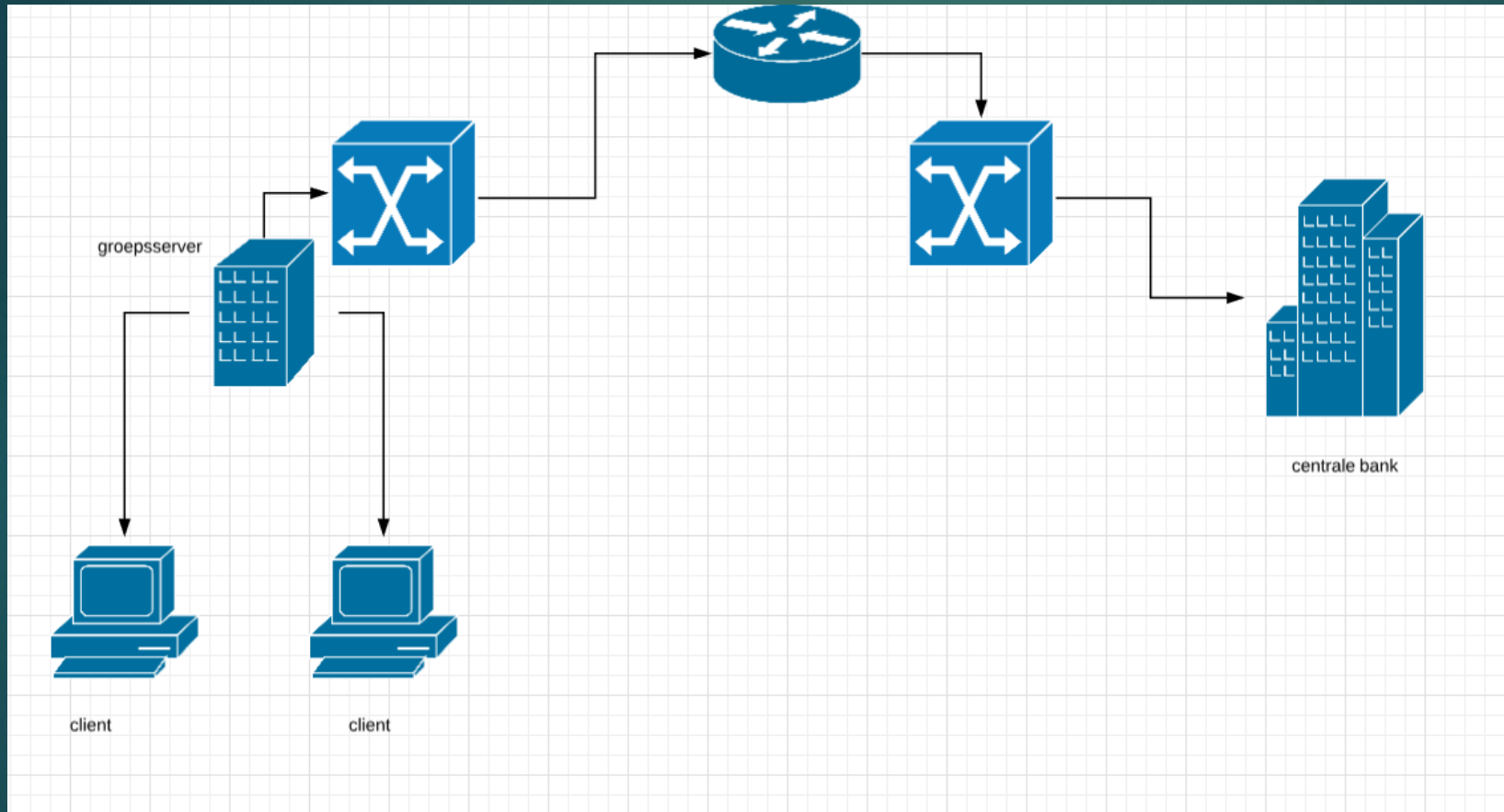
Niet functionele eisen

- ▶ De database is beveiligd met SSL
- ▶ Er moeten maatregelen worden genomen voor sql injecties
- ▶ De server en de client moet binnen 15 seconden zijn verbonden

ontwerp

- ▶ Gebruik van RSA, SSL en jdbc
 - ▶ RSA is een methode die veelgebruikt wordt door andere banken en heeft zichzelf al bewezen als een hele veilige methode
 - ▶ SSL wordt vaak gebruikt om te kijken of je met de goede server communiceert en is dus ook van belang als het gaat om communicatie tussen client en server
 - ▶ Voor de communicatie van de hardware en de client gebruik ik de RXTX library
 - ▶ Jdbc wordt gebruikt voor de communicatie met de database hiermee kan je communiceren in sql
 - ▶ Om dit veilig te maken, gebruik ik prepared statements.

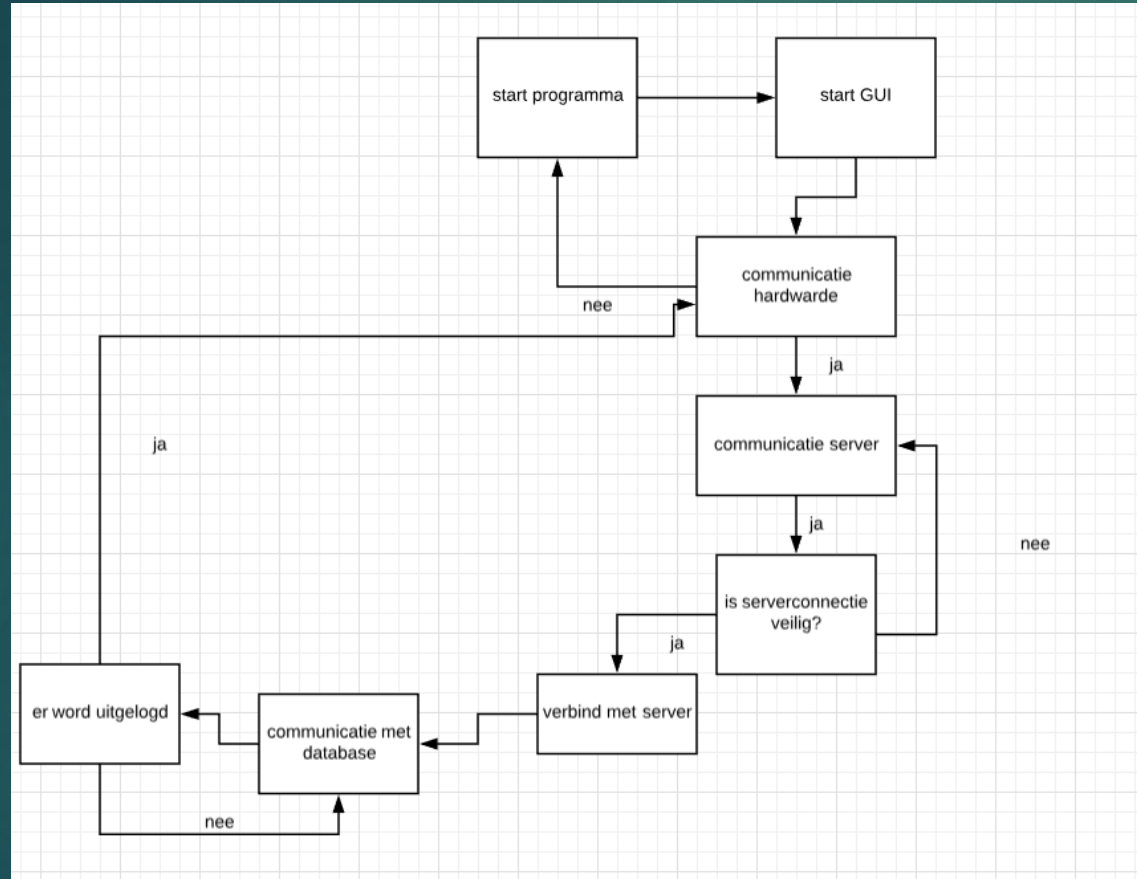
netwerkdigram



Een bank die
communiceert van client
naar groepsserver en van
groepsserver naar de
centrale bank

door het gebruik van
switches is dit op een
veilige manier gedaan

Dataflow diagram



Aanbeveling

- ▶ Mijn ontwerp is een zeer veilig ontwerp als we het hebben over een bank
- ▶ Het ontwerp is goed op het gebied van security en privacy en goed te realiseren
- ▶ Ik probeer ervoor te zorgen dat de code goed leesbaar is waardoor het communiceren over regels met andere groepen makkelijker gaat worden

bronvermelding

- ▶ [https://nl.wikipedia.org/wiki/RSA_\(cryptografie\)](https://nl.wikipedia.org/wiki/RSA_(cryptografie))
- ▶ https://nl.wikipedia.org/wiki/Transport_Layer_Security
- ▶ <https://playground.arduino.cc/Interfacing/Java>
- ▶ https://nl.wikipedia.org/wiki/Java_DataBase_Connectivity
- ▶ <https://docs.oracle.com/javase/tutorial/jdbc/basics/prepared.html>
- ▶ <https://www.digitalocean.com/community/tutorials/7-security-measures-to-protect-your-servers>