Elevator Design Document

Tiemon Steeghs

Inhoud

[1. Inleiding 3](#_Toc126741171)

[2. Use case definitie 4](#_Toc126741172)

[2.1. Use cases 5](#_Toc126741173)

[2.2. Use case diagram 6](#_Toc126741174)

# Inleiding

In deze opdracht ga ik de verschillende functies bepalen van een Elevator om vervolgens verschillende diagrammen op te stellen. Met behulp van de diagrammen kan ik een duidelijk beeld creëren van hoe het uiteindelijk systeem eruit gaat zien.

# Use case definitie

• What is the primary function of the system?

Het primaire functie van de lift is het vervoeren van passagiers van verdieping na verdieping.

• What are the secondary functions of the system?

1. Op het moment dat er een object tussen de deuren bevindt moeten de deuren van de lift niet dichtgaan de deuren gaan dan weer open. Na vijf seconden gaan de lift deuren dan weer dicht.
2. De potentiële passenger kan een “floor request button” indrukken om de lift naar een andere verdieping toe te laten gaan.
3. De Passenger kan de “Elevator request button” indrukken om een nabije lift op te vragen.
4. Elke verdieping beschikt over twee halve schuifdeuren die elkaar in het midden ontmoeten op het moment dat ze sluiten.
5. Als de lift aankomt bij een verdieping moeten de schuifdeuren van de lift en de desbetreffende verdieping tegelijk opengaan.
6. Speakers op elke verdieping die een geluid afspelen bij de aankomst van een lift.
7. Elke lift beschikt over een indicator waarop de verdieping staat waar de lift momenteel is
8. Elke verdieping beschikt over een indicator waarop te zien is op welke verdieping de lift momenteel is
9. Elke verdieping beschikt over een indicator waarop staat welke richting de lift op gaat.
10. Op het moment dat de “request buttons” worden ingedrukt zal er een led achter de knop oplichten.
11. Wanneer er teveel druk op de kabel van een lift komt zal de lift worden stopgezet

• Why is this system being built? What is it replacing and why?

Het systeem wordt gebouwd als vervanging voor de trap. Met de lift is het veel gemakkelijker om goederen omhoog te brengen in een gebouw en de lift zorgt er ook voor dat invaliden het gebouw kunnen betreden.

• What are the actors in this system?

Passenger, Potentiële Passenger, Service personeel

## Use cases

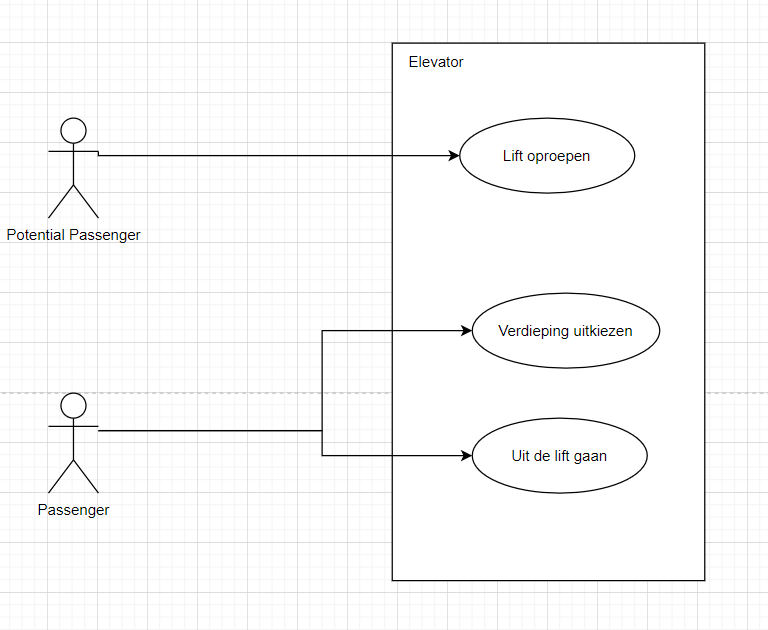
|  |  |
| --- | --- |
| Naam | Lift oproepen |
| Samenvatting | De Actor drukt op een “Elevator request button” om een lift op te roepen. |
| Actor | Potentiele Passenger |
| Aanname | N.v.t. |
| Beschrijving | 1. De Actor drukt op de “Elevator request knop” 2. Het systeem laat de dichtbijzijnste lift naar de verdieping van de gebruiker toegaan 3. De actor wacht totdat de lift arriveerd 4. Het systeem opent de deuren van de lift op het moment dat hij arriveerd |
| Uitzondering | 2) Er is al een lift onderweg – use case word afgebroken |
| Resultaat | De Lift is gearriveerd bij de bestemming |

|  |  |
| --- | --- |
| Naam | Verdieping uitkiezen |
| Samenvatting | De Actor kiest een verdieping uit met de “Floor request button” waar de lift vervolgens naartoe gaat. |
| Actor | Passenger |
| Aanname | De Actor bevindt zich in de lift |
| Beschrijving | 1. De Actor drukt op de “Floor request knop” 2. Het systeem sluit de deuren van de lift 3. Het systeem stuurt de lift naar de opgevraagde verdieping 4. Het systeem opent de deuren van de lift op het moment dat hij arriveerd |
| Uitzondering | 2) De deuren zijn al gesloten, er is al een verdieping opgevraagd - usecase wordt afgebroken. |
| Resultaat | De lift is gearriveert bij de gevraagde verdieping |

|  |  |
| --- | --- |
| Naam | Uit de lift gaan |
| Samenvatting | De gebruiker verlaat de lift nadat hij is gearriveerd bij een verdieping |
| Actor | Passenger |
| Aanname | De Passenger heeft een verdieping uitgekozen |
| Beschrijving | 1. De Actor wacht totdat de lift is aangekomen bij de opgevraagde bestemming 2. Het systeem opent de deuren van de lift 3. De gebruiker stapt uit de lift 4. Het systeem sluit de deuren na een delay van vijf seconden |
| Uitzondering | 4) Er bevindt zich een object tussen de deuren – de use case start opnieuw |
| Resultaat | De Actor is aangekomen bij zijn verdieping en is uitgestapt |

## Use case diagram

Dit is de use case diagram die ik heb gemaakt voor de lift. Hierin zie je dus dat je twee actors hebt: de passenger en de potentiële passenger. De passenger is de gebruiker die al in de lift zit en de potentiële passenger is een gebruiker die de lift kan gaan gebruiken en dus nog buiten de lift staat. De passenger kan dan vanuit de lift een verdieping uitkiezen om naartoe te gaan en de passenger kan dan ook weer uit de lift stappen.



# Werken tot een klassendiagram

*Object discovery: read the user requirements and list the objects of relevance*

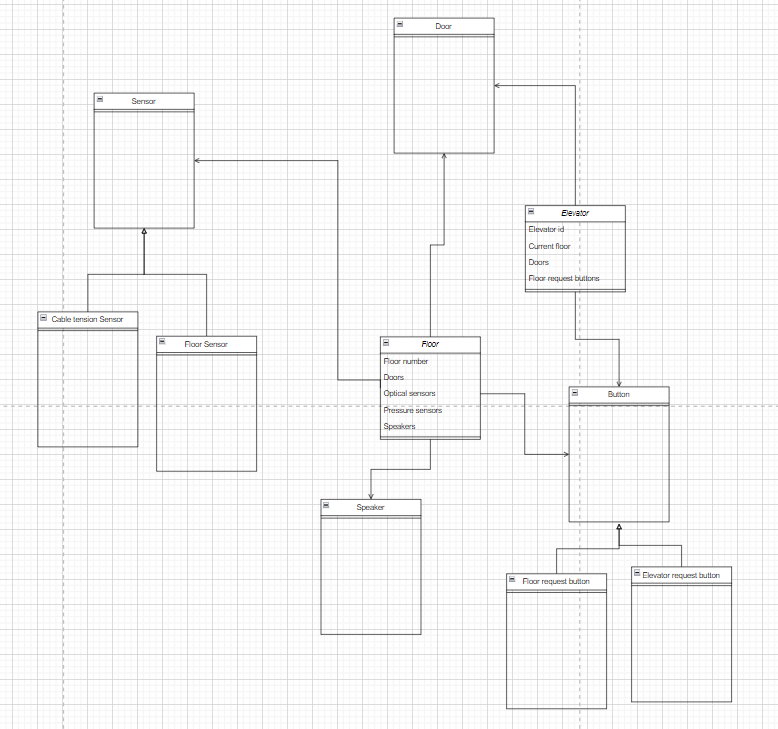
1. Elevator
2. Button
3. Elevator request button
4. Floor request button
5. Floor
6. Door
7. Sensor
8. Floor sensors
9. Cable tension Sensor
10. Speaker
11. Indicator
12. Floor Indicator
13. Elevator indicator
14. Direction Indicator

## Object diagram

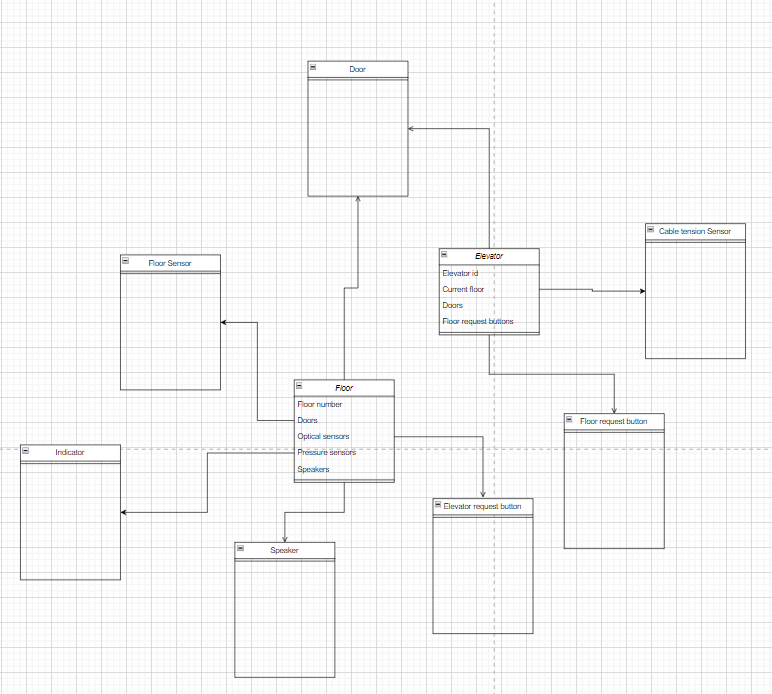
*Association discovery: draw an object diagram which lists the objects and their  
associations, use sequence diagrams where appropriate to find associations*

De object diagram van de lift. Je ziet hier alle objecten die voorkomen in de lift zoals de sensoren en deuren.

**V1**

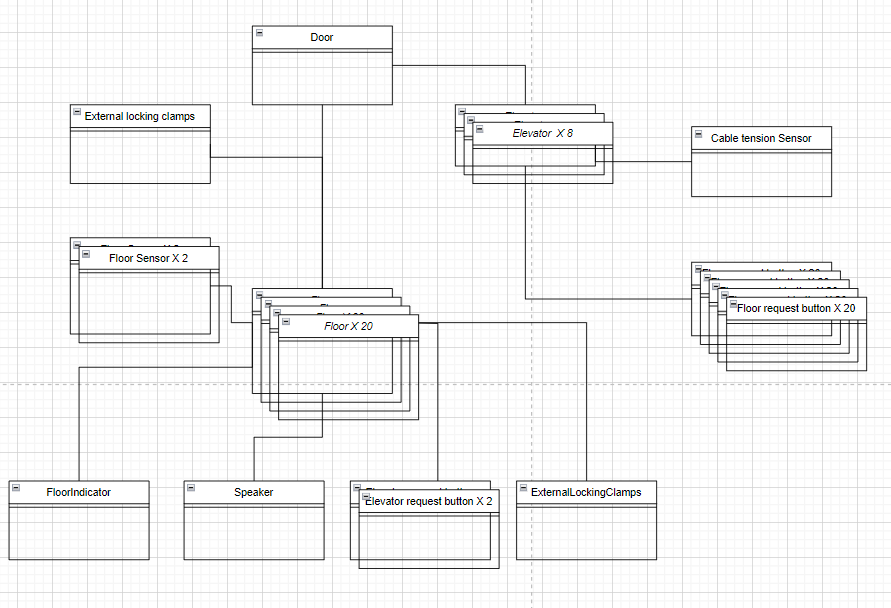


**V2**



**V3**

De laatste versie van de objecten diagram. Deze versie laat alle objecten zien en de relaties tussen de objecten. Naast dit kan je ook nog het aantal zien dat van elk object aanwezig is in het systeem.



## Sequence diagrams

De eerste sequence diagram zie je hieronder, deze is nog niet goed:

Afbeelding met tekst, binnen

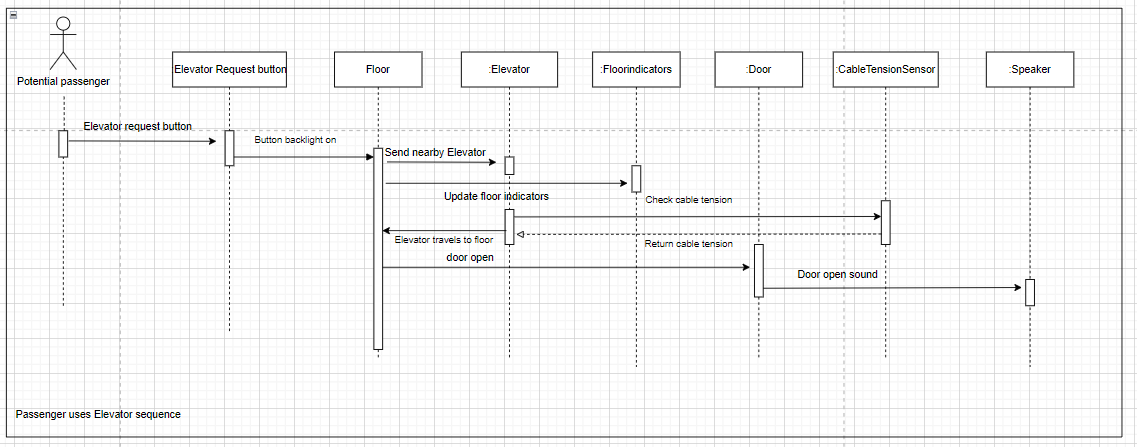
Automatisch gegenereerde beschrijving

De nieuwere versie van de sequencediagram zie je hier. Deze sequencediagram is van een passenger die de lift gebruikt. De passenger drukt als eerst op een knop om de floor te kiezen waarna hij toe wilt gaan. Na het drukken op de knop zal de lift naar de juiste verdieping toe gaan en zal er ondertussen ook worden gecontroleerd op dingen zoals objecten tussen de deur en druk op de liftkabel. Als de lift uiteindelijk aankomt op zijn bestemming, zullen de speakers een geluid maken.

Afbeelding met tekst, binnen

Automatisch gegenereerde beschrijving

De volgende sequence diagram is die van een potential passenger. Deze passenger roept als eerste een lift op waarna de lift dus naar de desbetreffende verdieping toe zal gaan, de liftindicatoren zullen dit dan ook aangeven. Tijdens het liftproces zullen er ook weer controles worden gedaan voor of er teveel druk op de kabels staat. Als tenslotte de lift aankomt bij zijn bestemming, zullen de speakers een geluid maken.



## Klassendiagram

*Use the objects to define the classes you need, draw a first class diagram and define the  
responsibility of each class (usually defined in a table)*

*Complete your first class diagram from step 3. You don’t need to use the appropriate  
relationships yet, however if you use inheritance you must indicate that correctly.*

*Define the required operations for each class.*

*Use sequence diagrams to verify the interactions between classes. Important: do not try to  
be complete in your sequence diagram: make a sequence diagram for a specific situation.  
E.g.: sequence diagram for trying to send a file in which the server does not accept the file.*

|  |  |
| --- | --- |
| Class 1 Floor |  |
| Elevator |  |
| Door |  |
| FloorIndicator |  |
| Speaker |  |
| External locking clamps |  |
| Button |  |
| Elevator request button |  |
| Floor request button |  |
| Sensor |  |
| CableTensionSensor |  |
| FloorSensor |  |

De klassendiagram bestaat uit in totaal 12 klassen. Onder deze 12 klassen zijn twee abstracte klassen om de verschillende soorten sensoren en buttons van elkaar te onderscheiden. Voorderest zijn de twee belangrijkste klassen de Floor en Elevator class, deze beschikken over alle verschillende objecten die je verspreid over de diagram ziet.

