

# Wat is uw risico?

---

Bereken het risico op een arbeidsongeval



Jeroen van de Laar

4-6-2012

Eindhoven

## Afstudeerverslag voor Fontys Hogeschool ICT

### Gegevens student:

Naam + voorletters student	JPA van de Laar
Studentnummer	2127717
Afstudeerrichting	S (Veltijd)
Afstudeerperiode datum van	20-2-2012 t/m 15-6-2012
Aantal Werkdagen	85 werkdagen

### Gegevens bedrijf:

Naam bedrijf/instelling	Roop group
Plaats	Eindhoven
Naam + voorletters+ functie van de bedrijfsbegeleid(st)er	R van Gassel

### Gegevens docentbegeleider:

Naam + voorletters docentbegeleid(st)er	MJM Lamers
---	------------

### Gegevens verslag:

Titel afstudeerverslag	Wat is uw risico?
Datum uitgifte afstudeerverslag	4-6-2012

---

Getekend voor gezien door bedrijfsbegeleider:

Datum:

De bedrijfsbegeleider,

## Voorwoord

Dit document heb ik geschreven als verslag van de afstudeerstage bij Roop. De opdracht heb ik bij Roop uitgevoerd als onderdeel van de opleiding software engineer bij de Fontys hogescholen te Eindhoven.

Ik wil graag een aantal personen bedanken die hebben bijgedragen aan het tot een succes maken van mijn afstudeerstage. Allereerst wil ik mijn bedrijfsbegeleider Rob van Gassel bedanken voor de steun die hij mij gegeven heeft gedurende het project. Verder wil ik ook Martijn Lamers bedanken voor de ondersteuning tijdens het project.

Eindhoven, Juni 2012

Jeroen van de Laar

## Inhoudsopgave

Samenvatting.....	5
Summary .....	6
Verklarende woordenlijst.....	7
1 Inleiding .....	9
2 Het bedrijf.....	10
2.1 Ontstaan van Roop .....	10
2.2 Filosofie .....	10
2.4 Organigram.....	11
3 Opdracht.....	12
3.1 Beginsituatie.....	12
3.2 Doel van de opdracht .....	12
3.3 Initiële Opdracht.....	13
3.4 Aanvullingen opdracht .....	14
3.4.1 Uitbreidingen opdracht .....	14
3.4.2 Aparte onderdeel opdracht.....	14
3.4.3 Losstaande opdracht .....	14
4 Projectinitiatie .....	15
4.1 Projectorganisatie .....	15
4.2 Initiële planning.....	16
4.3 Aangepaste planning.....	17
5 Onderzoek .....	19
5.1 Nationale gegevens .....	19
5.2 Formules.....	20
5.2.1 Stappen formules specificeren.....	20
5.2.2 Huidige formules .....	20
5.2.3 Wiskundige notaties .....	20
5.3 Programmeertaal .....	21
5.3.1 Java + JSP/J2EE .....	21
5.3.2 C# + ASP.NET .....	21
5.3.3 Keuze .....	21
5.4 Grafiek .....	22
5.4.1 ZedGraph .....	22
5.4.2 Webchart.....	22

5.4.3 Nplot.....	22
5.4.4 Keuze .....	22
5.5 Android.....	23
5.5.1 Programmeertaal .....	23
5.5.2 GUI ontwerp .....	24
5.5.3 Database connectie .....	25
5.5.4 Remote database & Webservice .....	26
6 Ontwerp.....	27
6.1 Light versie .....	27
6.2 Full versie.....	29
6.3 Voorraad.....	30
6.4 Android .....	31
7 Realiseren .....	32
7.1 Risico Calculator .....	32
7.1.1 Berekeningen.....	32
7.1.2 Light versie.....	32
7.1.3 Full versie.....	32
7.3 Android.....	33
7.3.1 List Layout.....	33
8 Tools & Technieken .....	34
8.1 Tools .....	34
8.1.1 Microsoft Visual studio.....	34
8.1.2 Navicat.....	34
8.1.3 PhpMyAdmin.....	34
8.1.4 Microsoft office Visio.....	34
8.2 Technieken .....	35
8.2.1 MySQL.....	35
8.2.2 UML .....	35
9 Conclusie(s) en Aanbeveling(en) .....	36
Evaluatie .....	37
Literatuurlijst .....	38
Bijlage A: PID .....	39
Bijlage B: Onderzoeksdocument .....	53
Bijlage C: Ontwerpdokument .....	61

## Samenvatting

De opdracht "Risico calculator" is uitgevoerd bij het bedrijf Roop. Roop wil met de risico calculator risico's voor bedrijven berekenen. Met behulp van een analyse van deze risico's wil Roop de bedrijven in laten zien dat veiligheid op de werkvloer een erg belangrijk onderwerp is. Roop is een bedrijf dat zich inzet om de veiligheid op de werkvloer te verhogen.

In de afstudeerperiode van 85 dagen zijn de onderdelen uitgevoerd die men tegenkomt bij het bouwen van een nieuwe applicatie. Zo is in het begin de project initiatie uitgevoerd. Tijdens deze fase is het Project Initiatie Document opgesteld. In dit document worden de opdracht en de planning beschreven. Een ander onderdeel van deze fase is een gesprek met de bedrijfsbegeleider en docentbegeleider.

Na de project initiatie fase is de onderzoeksfase uitgevoerd. Tijdens de onderzoeksfase is er naar verschillende onderwerpen onderzoek gedaan. Allereerst is er gekeken naar de gegevens die beschikbaar waren en hoe deze gegevens gespecificeerd waren. Daarna is er onderzoek gedaan naar de formules die de risico's moesten uitrekenen. Tijdens het onderzoek naar de formules bleek dat deze eenvoudiger waren dan gedacht en is de planning aangepast. Na het onderzoek naar de formules is er gekeken naar welke programmeertaal het beste was om te gebruiken. Hieruit bleek dat C# en ASP.NET het beste waren. Later in het project is er nog onderzoek gedaan naar de library die gebruikt moest worden voor het maken van een grafiek en er is nog gekeken naar hoe een Android applicatie ontwikkeld moest worden.

Het maken van ontwerpen van de te ontwikkelen applicaties was de volgende stap in het project. Tijdens deze fase zijn er verschillende diagrammen gemaakt van de applicaties. Ook zijn er ontwerpen gemaakt van de GUI. Naast de ontwerpen voor de applicaties zijn er ook ontwerpen gemaakt voor de database.

Na de ontwerpfase is er overgegaan naar de ontwikkelfase. Tijdens deze fase zijn de applicaties gerealiseerd. Tijdens deze fase zijn de ontwerpen en de formules omgezet in code en in een GUI. De realisatie fase heeft geleid tot een diverse applicaties zoals die beschreven zijn tijdens de project initiatie fase.

De aanbeveling die gedaan is aan Roop is om van de Android applicatie een Apple IOS versie te maken.

## Summary

The assignment "Risk calculator" was conducted at Roop. Roop wants to calculate risks, for companies, using the risk calculator. With the analysis of these risks Roop wants to convince companies that security in a working environment is a very important subject. Roop is a company whose effort is to improve the security in a working environment.

In a period of 85 days a number of subjects have been conducted that are normally found during the build of a new application. In the beginning of the project, the project initiation phase started. During this phase the Project Initiation Document was formed. In this document the assignment and planning are described. Another part of this phase was a meeting with the company supervisor and the school supervisor.

After the project initiation phase the research phase was executed. During this phase research was conducted towards several subjects. At first there was research conducted towards the data that was available and towards the specification of this data. After that there was looked into the formulas needed to calculate the risks. During the research towards the formulas it turned out that the formulas were much easier to write and because of that the planning got changed. After the formulas research was done to find the best programming language. It turned out to be C# in combination with ASP.NET. During the later stages of the project there was some research towards libraries for graphs and some towards the development of Android applications.

Making the designs of the applications was the next step during the project. During this phase all sorts of diagrams were made. As well as designs for the GUI of the application. Besides the designs for the applications designs were made for the database.

After the design phase the development phase was conducted. During this phase the applications were realized. During this phase the designs and formulas were transformed into code and a GUI. The development phase resulted into multiple application as described in the project initiation phase.

There was also a recommendation towards Roop. This was to make an Apple IOS version of the Android application.

## Verklarende woordenlijst

Naam	Uitleg
Android	Besturingssysteem van Google voor mobiele telefoons.
Apache	Webserver.
Apple OS	Besturingssysteem, Apple variant van Windows
Apple IOS	Besturingssysteem van Apple voor mobiele telefoons.
ASP.NET	Webprogrammeertaal van Microsoft
C#	Desktop programmeertaal van Microsoft
CBS	Centraal bureau voor de statistiek
Database Viewer	Een programma wat gebruikt wordt om een database te kunnen beheren. Met dit programma kunnen alle acties op een database gedaan worden die normaal met SQL statements gedaan worden.
DML	Data Manipulation language, Alle activiteiten op een database die de gegevens kunnen ophalen of aanpassen. select, insert, update en delete
ERD	Entity-relationship diagram, Diagram wat op schematische manier een database weergeeft met de bijbehorende relaties tussen tabellen.
Fysiek device	Een apparaat wat echt bestaat.
HTML	Hypertext Markup Language, Taal die gebruikt wordt voor de opmaak van webpagina's.
IDE	Integrated development environment, Programma wat gebruikt wordt om applicatie in te bouwen.
J2EE	Webprogrammeertaal van Java
Java	Programmeertaal
JSON	JavaScript Object Notation, Speciale notatie die gebruikt wordt voor het versturen van gegevens over het internet.
JSP	Webprogrammeertaal van Java
Klassendiagram	Diagram wat op een schematische manier de opbouw van een applicatie weergeeft.
Library	Code die extern van een applicatie gemaakt is, maar die extra functionaliteit kan geven aan een applicatie.
Linux	Besturingssysteem, Variant van Windows.
Open source	Programma wat gratis is en door iedereen verbeterd kan worden.
PHP	Webprogrammeertaal.
PHPMyAdmin	Database viewer
PID	Projectinitiatie document, Document wat aan het begin van het project gemaakt wordt. In dit document staat alles wat, aan het begin, belangrijk is voor het project.
Requirements	Lijst van functionaliteiten die een applicatie moet kunnen nadat hij gebouwd is.
Scrum	Software ontwikkelmethode, Bij scrum zijn er veel losse deelprojecten die gemiddeld 1 tot 4 weken duren. Ook kunnen deelprojecten tegelijkertijd uitgevoerd worden en wordt er aan het einde van een deelproject gedemonstreerd wat er in de deelproject is uitgevoerd.
Sequentie diagram	Diagram wat op een schematische manier de volgorde van aanvragen in een applicatie weergeeft.
TNO	Een onderzoeksinstituut met als doelstelling het toepassen van wetenschappelijke kennis in de praktijk.
UML	Unified Modeling Language, Standaard die gebruikt wordt voor het maken van ontwerpen van applicatie.



Unit test	Geprogrammeerde tests die een applicatie testen om te zien of deze doet wat vooraf gespecificeerd is dat hij moet doen.
Usecases	Diagram wat een stap voor stap specificatie geeft van de gewenste functionaliteit voor een applicatie.
Visual Basic	Eerste programmeertaal van Microsoft. Voorloper van C#.
Webserver	Server die via internet verzoeken ontvangt van applicaties en een passend antwoord terugstuurt.
XML	Extensible Markup Language, Taal die gebruikt wordt voor de opmaak van documenten die door zowel machines als mensen leesbaar is.

## 1 Inleiding

Weet u wat uw risico is op een bedrijfsongeval? Dit zou namelijk veel hoger kunnen zijn dan u verwacht. Jaarlijks gebeuren er meer dan 200.000 bedrijfsongevallen en vallen er 85 doden door ongelukken op de werkvloer. Vaak zijn er eenvoudige maatregelen om deze ongelukken te voorkomen, maar veiligheid is bij de meeste bedrijven geen punt van aandacht omdat ze niet weten welk risico er is op een ongeluk. Daar komt nu verandering in.

Deze verandering komt er door de inzet van Roop. Roop is een bedrijf wat zich inzet om de veiligheid op de werkvloer te verhogen. Daarom hebben ze de risico calculator ontwikkeld. Dit is een programma waarmee het risico op een bedrijfsongeval is te berekenen. Doordat dit risico vaak hoger is dan gedacht worden bedrijven wakker geschut om nogmaals over veiligheid na te denken. De risico's worden berekent op basis van de gegevens die door de bedrijven invoeren en op basis van nationale gegevens verkregen door TNO en CBS. Het project is uitgevoerd door eerst een onderzoek te doen, vervolgens zijn er ontwerpen gemaakt en daarna is het programma gebouwd.

In deze alinea kunt u informatie vinden over de indeling van de rest van dit document. In hoofdstuk 2 wordt informatie gegeven over het bedrijf waar het project is uitgevoerd. In hoofdstuk 3 wordt uitgebreid beschreven wat de opdracht was. Hoofdstuk 4 beschrijft de start van het project. In hoofdstuk 5 kunt u informatie vinden over het onderzoek dat verricht is. Hoofdstuk 6 bevat alle informatie met betrekking tot ontwerpen. Hoofdstuk 7 beschrijft de keuzes die gemaakt zijn tijdens het realiseren. In hoofdstuk 8 is er informatie te vinden over de verschillende tools en technieken die gebruikt zijn bij dit project en in hoofdstuk 9 vind de afsluiting van dit document en aanbevelingen aan Roop.

## 2 Het bedrijf

### 2.1 Ontstaan van Roop

Roop is een doorstart van een in het verleden failliet gegaan bedrijf, waarbij de activiteiten, voorraad en personeel zijn overgenomen uit de failliete boedel. Roop is daarna verder gegaan als een bedrijf wat zich vooral focust op veiligheid op bouwplaatsen en overige veiligheid voor bouwbedrijven. De veranderende visie van de investeerder heeft ervoor gezorgd dat de directeur ontslag heeft genomen en vanaf dat moment heeft Roop zich meer ontwikkeld als algemeen veiligheid bedrijf. Twee nieuwe directeuren worden aangetrokken en deze hebben beide een sterke focus op dienstverlening en service naar klanten, niet alleen maar het verkopen van producten. Roop breidt het klanten aanbod uit en splitst de bouwactiviteiten af in een eigen eenheid: valnetten.nl. Daardoor ontstaat een uniformere benadering naar de markt, waarbij de klanten ook meer divers worden en uit allerlei sectoren komen.

### 2.2 Filosofie

Roop heeft een duidelijke missie en visie waar het gaat om veilig werken. De missie en visie van Roop zijn beschreven door Rob van Gassel en Willem de Vocht.

#### 2.2.1 Visie

“Wij geloven dat veiligheid een beleving is die je kunt delen, waarmee je mensen motiveert en inspireert”

Veiligheid is een gevoel en niet het volgen van regels en wetten. Inspireer en motiveer mensen om veilig te werken en win daarmee op alle punten in de organisatie. Door veilig werken te stimuleren de organisatie laten groeien naar het nieuwe werken: Veilig werken = prettig werken = productief werken! Veel organisaties laten zich in hun veiligheidsbeleid door angst voor ongelukken en sancties sturen, terwijl het vooral een sociale betrokkenheid en intrinsieke waarde van een bedrijf moet zijn.

#### 2.2.2 Missie

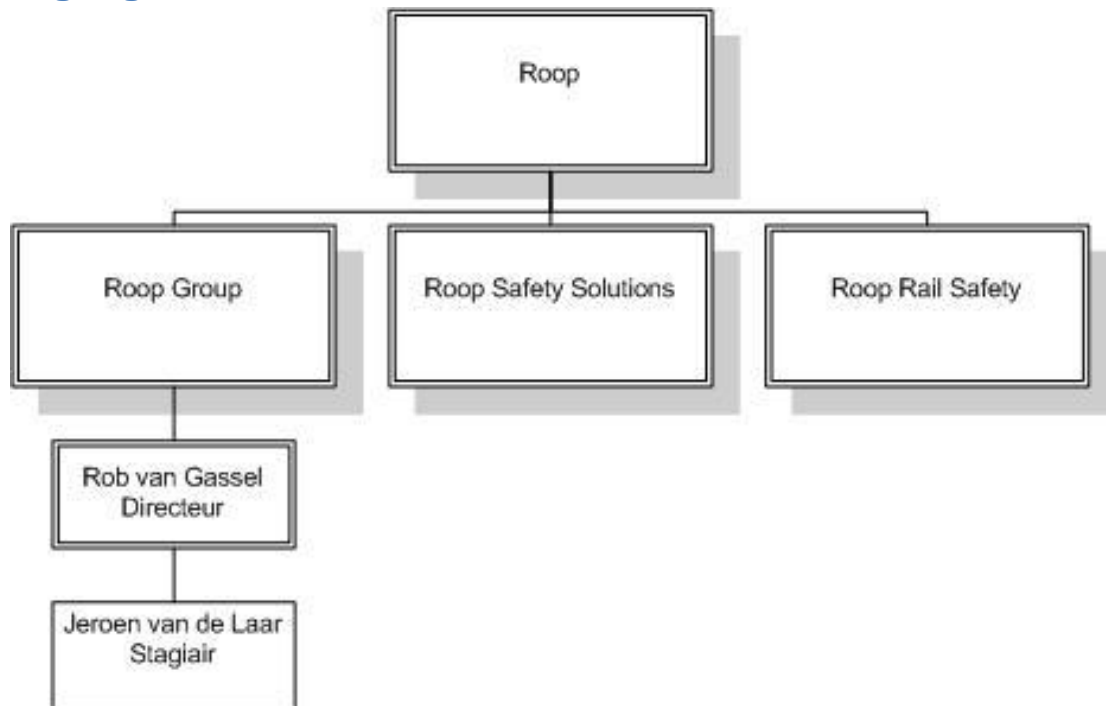
Alle diensten en producten van Roop moeten het gevoel van veiligheid voor klanten en hun werknemers versterken, veilig werken omdat het beter is en niet omdat het “moet”.  
Samenvattend: “Meer veiligheid, minder zorgen!”

### 2.3 Organisatie

Roop Group bestaat uit een drietal werkmaatschappijen die gezamenlijk werken onder de naam Roop. De twee werkmaatschappijen “Roop Rail Safety” en “Roop Safety Solutions” bevatten oplossingen en producten voor onder andere constructie georiënteerde branches. De werkmaatschappij Roop Group houdt zich bezig met dienstverlening en consultancy op het gebied van veilig werken en veiligheid. Roop Group richt zich vooral op het MKB en niet zozeer op een bepaalde sector of branche. Veilig werken is immers niet gebonden aan een bepaalde beroepsgroep, maar een universeel arbeidsrecht.

Roop Group gaat voor totale ontzorging en ondersteuning op het gebied van arbeidsrisico's en veiligheidsvraagstukken. De focus ligt op het overnemen van de zorgen en vragen rondom veilig werken in de meest brede zin van het woord. Veiligheid wordt door Roop als een dienst gezien en niet alleen maar als producten en maatregelen die u in kunt zetten. De uitgebreide kennis en opgedane ervaring maakt van Roop een allround specialist op het gebied van veiligheid. Een groot netwerk van specialisten en veiligheidsdeskundigen maakt het mogelijk om voor iedere branche of sector de juiste personen aan te bieden.

## 2.4 Organigram



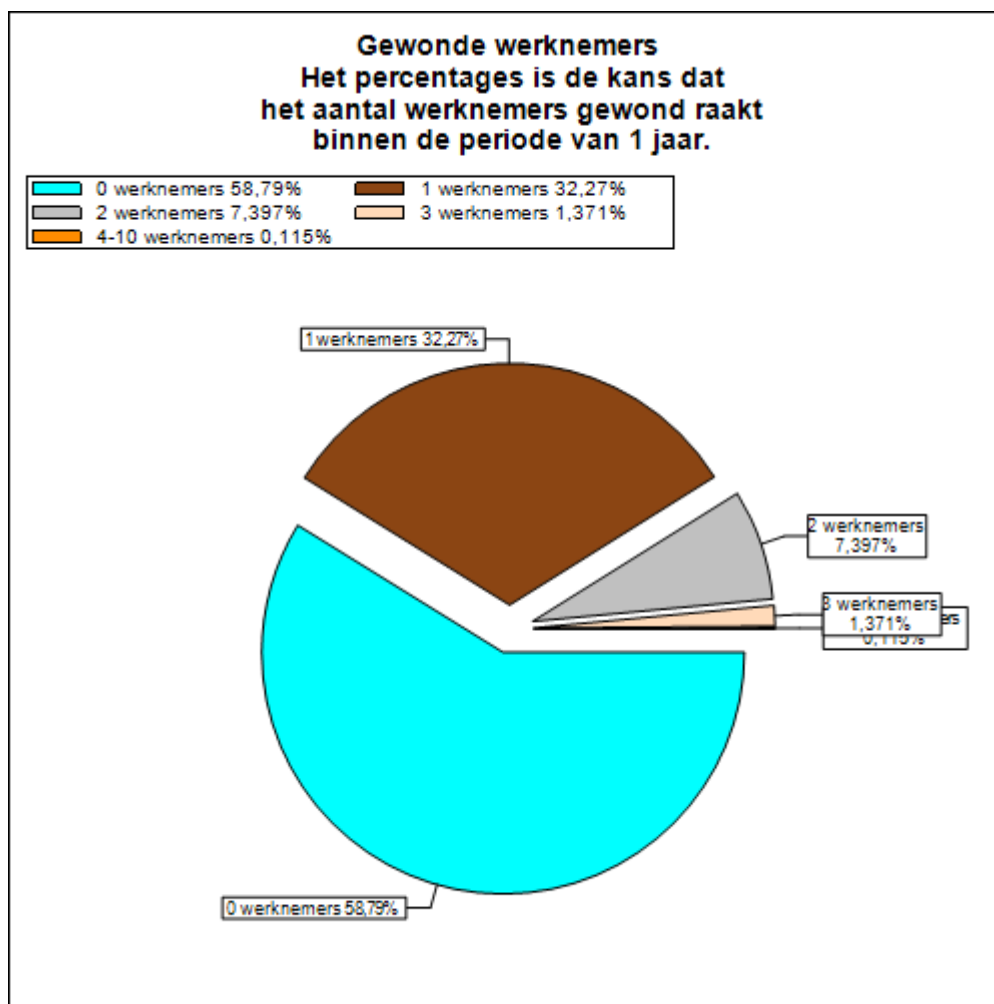
## 3 Opdracht

### 3.1 Beginsituatie

Voor aanvang van het project bestond er nog geen applicatie om op voort te borduren, er waren geen formules en de nationale gegevens, van TNO en het CBS, waren nog niet beschikbaar. De opdracht was erop gericht om de kans op een ongeluk, op de werkvloer, in kaart te brengen. Doordat bedrijven niet weten welk risico ze lopen nemen ze geen adequate maatregelen. Hierdoor wordt de kans op een ongeluk nog groter. Verder was de opdracht erop gericht om bedrijven duidelijk te maken wat de consequenties zijn van een ongeluk.

### 3.2 Doel van de opdracht

Het doel van de opdracht is om een systeem te ontwikkelen waarmee veiligheidsrisico's berekent kunnen worden. De analyse van de risico's heeft als doel om bedrijven wakker te schudden en in te laten zien dat veiligheid erg belangrijk is en dat dit dan ook veel aandacht verdient. De systemen moeten gebruikersvriendelijk zijn om ervoor te zorgen dat ze ook daadwerkelijk gebruikt zullen worden.



Afbeelding 3.1

### 3.3 Initiële Opdracht

De initiële opdracht was het maken van een risicocalculator. Een risicocalculator is een applicatie die, voor een bedrijf, risico's berekent op basis van nationale cijfers en gegevens van het bedrijf. De gegevens worden verkregen via nationale onderzoeken en de gegevens van het bedrijf worden verkregen door het invullen van deze gegevens in de applicatie.

De risicocalculator die gemaakt moest worden bestaat uit twee versies. Een light versie en een full versie. De light versie is een webapplicatie. Dit betekent dat bij de light versie de gegevens in de applicatie ingevuld moeten worden in invoervelden, die geplaatst zijn op de website, of door de locatie van een XML file in te voeren. Bij de light versie krijgen de bedrijven slechts enkele risico's als resultaat. De resultaten worden getoond op het scherm in de vorm van tekst en grafieken. Als bedrijven alle risico's willen inzien zal er contact opgenomen moeten worden met Roop. De full versie is een desktop applicatie. Bij de full versie wordt een beschrijving van alle risico's getoond. De risico's worden getoond d.m.v. een normale tekst file, een XML file of een HTML file. Ook wordt er een grafiek gemaakt van de resultaten. De invoer gebeurt d.m.v. een database, XML file of invoervelden.

De risico's die berekend moesten worden zijn de kans op een ongeluk voor werknemers binnen een bedrijf, de kans dat meerdere ongelukken zich voordoen en het gemiddelde risico van een bedrijf. Om de risico's te berekenen zullen er allereerst formules geschreven moeten worden die, op basis van de landelijke cijfers en de cijfers van het bedrijf, de kans op letsel berekenen. Aan de hand van de resultaten kan het bedrijf dan zien welk risico zij lopen. Een voorbeeld hiervan is te vinden in afbeelding 3.1.

Voor het schrijven van de formules moest er eerst onderzoek gedaan worden naar de gegevens die beschikbaar zijn. Dit betrof de type gegevens maar ook hoe specifiek de gegevens zijn. Bijvoorbeeld een groep van 120 mensen waarvan 60 mensen 15-30 jaar zijn en 60 mensen 31-45 jaar zijn. Van de groep van 120 mensen zijn 12 mensen schilder. Of een groep van 120 mensen waarvan 9 mensen 15-30 jaar zijn en schilder en waarvan 3 mensen 31-45 jaar zijn en schilder. De tweede situatie is veel specifiek. Ook moest er onderzoek gedaan worden naar het schrijven van formules zelf. Dit betrof onderzoek naar welke stappen de formules moesten doen, of er al bestaande formules zijn en wat de verschillende wiskundige notaties binnen formules betekenen.

Er was geen eis opgelegd aan een specifieke programmeertaal die gebruikt moest worden. Om de geschiktste programmeertaal te vinden was het wenselijk om hier onderzoek naar te doen. Het onderzoek was nodig voor zowel een desktop taal als voor een web taal. Ook voor de ontwikkelmethode die gebruikt moest worden was geen eis opgelegd. Er is hier gekozen voor scrum omdat deze methode veel deelprojecten definieert waardoor het makkelijker in te plannen is en ook beter teruggekoppeld kan worden.

### 3.4 Aanvullingen opdracht

De oorspronkelijke projectdoelstellingen waren eerder bereikt dan verwacht. Doordat de formules minder gecompliceerd waren dan vooraf ingeschat, hebben de meeste onderdelen van de opdracht minder tijd gekost. Dit gaat voornamelijk over het ontwerpen van de applicatie, maar ook over het programmeren van de formules zelf. Naast het sneller afhebben van de formules bleek ook de webapplicatie eenvoudiger te zijn. Deze was dus ook eerder klaar dan vooraf gepland. Naast deze onderdelen was ook het verkrijgen en verwerken van de benodigde gegevens sneller klaar.

Doordat er na het afronden van de oorspronkelijke projectdoelen nog tijd over bleef binnen dit project, zijn er een aantal aanvullingen geweest op de opdracht. Deze aanvullingen omvatten zowel uitbreidingen op de bestaande opdracht in de vorm van nieuwe onderdelen, maar er is ook een opdracht bijgekomen die los staat van de initiële opdracht.

#### 3.4.1 Uitbreidingen opdracht

De uitbreidingen op de opdracht zijn in de vorm van extra informatie die aan de bedrijven getoond wordt. Denk bijvoorbeeld aan sterftcijfers of aan hoe lang werknemers niet kunnen werken na een ongeval. Het doel van deze gegevens is om bedrijven nog meer te laten realiseren dat veiligheid iets is waarop gelet moet worden.

#### 3.4.2 Aparte onderdeel opdracht

Het aparte onderdeel binnen het onderwerp van de opdracht is een Android. Het doel van de applicatie is opnieuw om mensen duidelijk te maken dat veiligheid belangrijk is. De applicatie zal een lijst van activiteiten bevatten die gebruikers kunnen uitvoeren. Denk bijvoorbeeld aan het werken op een dak.

De gebruikers selecteren een activiteit uit deze lijst en dan toont de applicatie maatregelen die de gebruikers zouden kunnen treffen om ongelukken te voorkomen. Denk hierbij bijvoorbeeld aan randbeveiliging (metalen rand om de werkplaats) of val beveiliging (valnetten). Als men op de maatregelen klikt verschijnt een scherm met uitgebreide uitleg over de maatregel. Denk aan wat voor type maatregel of een uitleg van hoe het materiaal bevestigd moet worden. De gegevens in de applicatie worden geladen uit een remote database via een webservice en in een lokale database opgeslagen. De gebruiker kan deze gegevens zelf updaten.

#### 3.4.3 Losstaande opdracht

De losstaande opdracht betreft een voorraad applicatie. Dit is een applicatie die de voorraad van valnetten, die gebruikt worden in de bouw, beheerd voor Roop. De applicatie moet op verschillende attributen netten kunnen zoeken uit een database. De applicatie moet het mogelijk maken dat de status en de locatie van de netten gewijzigd wordt. Verder moet de applicatie het mogelijk maken dat er een rapport gegenereerd wordt van alle netten in het magazijn.

## 4 Projectinitiatie

In de eerste week van het project is de projectinitiatie uitgevoerd. Het doel van de projectinitiatie is het verduidelijken van de opdracht. De projectinitiatie is uitgevoerd d.m.v. het schrijven van een PID (Projectinitiatie document).

Aan het begin van de projectinitiatie is er een oriënterend gesprek geweest tussen de opdrachtgever en de stagiair. Dit is gedaan om de motivatie voor het starten van het project duidelijk te krijgen bij de stagiair. Ook krijgt de stagiair op deze manier een eerste beeld bij de opdracht.

Aan de hand van dit gesprek is er nagedacht over mogelijke oplossingen en zijn er vragen opgesteld. Op deze manier kunnen, tijdens een tweede gesprek, de specifieke eisen en wensen van de opdrachtgever vastgesteld worden.

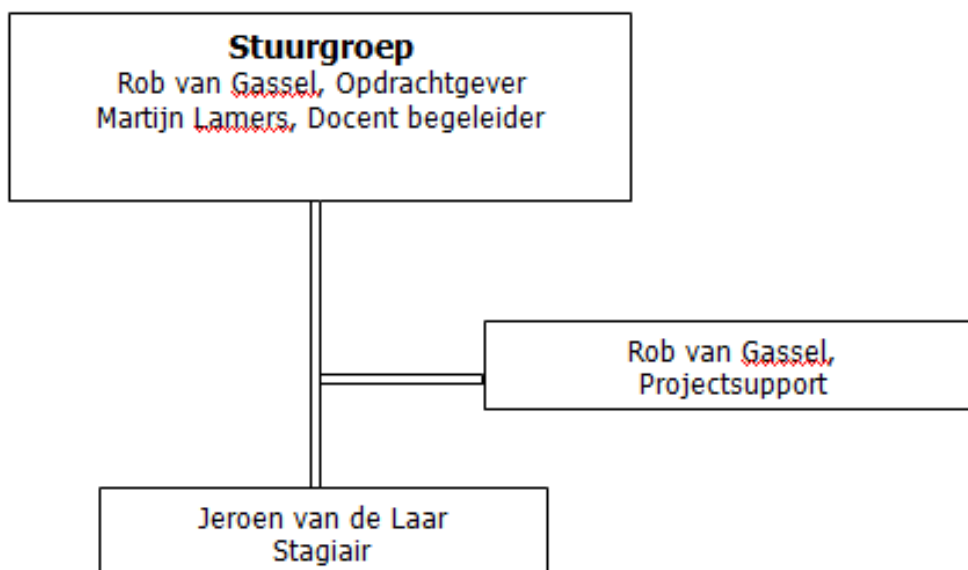
Na het tweede gesprek is er onderzoek gedaan naar de beschikbare gegevens, mogelijke oplossingen en eventuele bestaande oplossingen. Dit is gedaan om een duidelijk beeld te krijgen van wat er mogelijk is en op deze manier de opdracht nog verder te verduidelijken. Verder is dit gedaan om een eigen mening te vormen t.a.v. de opdrachtspecificatie.

Na het verduidelijken van de opdracht is er begonnen met het schrijven van het PID. Hierin is de definitieve opdrachtomschrijving opgenomen en een globale planning. Dit wordt gedaan omdat er dan altijd teruggevallen kan worden op de opdrachtomschrijving en zodat er aan het einde van het project beoordeeld kan worden of het project succesvol is geweest en of het project binnen de geplande periode is uitgevoerd.

Nadat het PID opgesteld is, is deze door de stuurgroep beoordeeld en is er akkoord gegaan met de start van het project. Dit is gebeurd om ervoor te zorgen dat de visie die de stuurgroep heeft dezelfde is als de visie die de stagiair heeft. Verder gebeurd dit om ervoor te zorgen dat de stuurgroep gedurende het project kan controleren of de student nog wel op de juiste koers ligt.

### 4.1 Projectorganisatie

De projectorganisatiestructuur is weergegeven in afbeelding 4.1. Een uitleg van de taken van de rollen is te vinden in het PID. Deze is te vinden in Bijlage A.



Afbeelding 4.1



## 4.2 Initiële planning

Wanneer?	Wat wordt er gedaan?	Wat is er af?
Week 1	<b>Fase 1 formules</b> Opdracht verduidelijken  PID maken  Afspraken maken  Onderzoek doen naar de landelijke cijfers  Onderzoek doen naar formules	1ste versie Requirements  PID
Week 2	Onderzoek doen naar de landelijke cijfers  Onderzoek doen naar formules  Schrijven van formules	Gegevens voor uiteindelijke database
Week 3	Schrijven van formules	Formules
Week 4	<b>Fase 2 ontwerpen</b> Requirements verfijnen  Onderzoek programmeer taal  Ontwerpen Usecases	Requirements  Keuze programmeertaal  Usecases
Week 5	Ontwerpen klassendiagrammen	
Week 6	Ontwerpen klassendiagrammen  Ontwerpen sequentie diagrammen	Klassendiagrammen  Sequentie diagrammen
Week 7	Ontwerpen database  Opzetten database	Database ontwerp  Database
Week 8	<b>Fase 3 light versie</b> Programmeren Light versie tool	GUI ontwerp light versie
Week 9	Programmeren Light versie tool	
Week 10	Programmeren Light versie tool  Testen light versie tool	Light versie tool  Unit tests light versie, Rapportage over het testen
Week 11	<b>Fase 4 full versie</b> Programmeren Full versie tool	GUI ontwerp full versie
Week 12	Programmeren Full versie tool	
Week 13	Programmeren Full versie tool	
Week 14	Programmeren Full versie tool  Testen full versie tool	Full versie tool
Week 15	Testen Full versie tool	Unit tests full versie, Rapportage over het testen
Week 16	Uitloop	
Week 17	Uitloop	

### 4.3 Aangepaste planning

Doordat het project sneller af was dan gedacht zijn er een aantal onderdelen toegevoegd. Hierdoor is de planning in de loop van het project meerdere malen verandert. Hieronder vind u de uiteindelijke planning zoals deze gevolgd is tijdens het project.

Wanneer?	Wat wordt er gedaan?	Wat is er af?
Week 1	<b>Fase 1 formules</b> Opdracht verduidelijken  PID maken  Afspraken maken  Onderzoek doen naar de landelijke cijfers  Onderzoek doen naar formules  Schrijven van formules	1ste versie Requirements  PID  Gegevens voor uiteindelijke database  Formules
Week 2	Onderzoek doen naar de landelijke cijfers  Schrijven van formules <b>Fase 2 ontwerpen</b> Requirements verfijnen  Onderzoek programmeertaal  Ontwerpen Usecases  Ontwerpen database  Opzetten database  Ontwerpen klassendiagrammen <b>Fase 3 light versie</b> Programmeren Light versie	Definitieve cijfers voor database  Definitieve formules  Requirements  Keuze programmeertaal  Usecases  Erd  Database + SQL files  Klassen diagram Light versie  GUI light versie + grote delen light versie ingevuld
Week 3	Programmeren Light versie  <b>Fase 4 voorraad programma</b> Programmeren Voorraad programma	Light versie voor het testen formules.  Voorraad programma
Week 4	Programmeren Light versie  Testen Light versie  Ontwerpen Sequentie diagrammen <b>Fase 5 ontwerpen full versie</b> Ontwerpen klassen diagrammen <b>Fase 6 full versie</b> Programmeren Full versie	Definitieve Light versie  Unit tests light versie, Rapportage over het testen  Sequentie diagrammen Light versie  Klassen diagrammen Full versie
Week 5	Programmeren Full versie	
Week 6	Programmeren Full versie  Testen Full versie  Ontwerpen sequentie diagrammen	Definitieve Full versie  Unit tests Full versie, Rapportage over het testen Sequentie diagrammen Full versie

Week 7	<b>Fase 7 uitbreiden full &amp; light versie</b> Toevoegingen Full & Light versie	XML input en output HTML Output
Week 8	Toevoegingen Full & Light versie	Grafiek voor gegevens Gegevens over de branche
Week 9	Toevoegingen Full & Light versie	Optie gegevens wel/niet tonen
Week 10	<b>Fase 8 Android</b> Android programmeren uitgezocht	Keuzes Android programmeren
Week 11	Programmeren Android tool	GUI ontwerp Android Database koppeling Android
Week 12	Programmeren Android tool  Webservice installeren	GUI uitgebreid Functionaliteit geïmplementeerd Getest op fysiek device  Wamp server geïnstalleerd
Week 13	Android database vullen via remote database implementeren	Remote connectie via webservice
Week 14	<b>Fase 9 fine tune</b> Fine tune	
Week 15	Fine tune	
Week 16	<b>Fase 10 implementeren</b> Implementeren	
Week 17	Implementeren	

Tijdens **Fase 2** is er besloten om eerst te beginnen met het realiseren van de light versie. Dit is gebeurd omdat er al snel bleek dat er veel tijd over zou zijn. Door eerst te beginnen met het realiseren van de light versie was er de mogelijkheid om de formules aan te passen mochten deze niet voldoen. Als er ook al direct een ontwerp voor de full versie gemaakt zou worden, zou deze ook herzien moeten worden als de formules niet voldeden.

**Fase 4** is uitgevoerd tijdens **Fase 3** dit is gedaan nadat er gebleken was dat er genoeg tijd was om extra onderdelen toe te voegen aan de opdracht. **Fase 4** was een korte fase waarin een applicatie ontwikkeld is die direct in gebruik genomen kon worden. Het feit dat deze applicatie direct in gebruik genomen kon worden is de reden dat **Fase 4** tijdens **Fase 3** is uitgevoerd.

Na het uitvoeren van **Fase 6** is er besloten om de full en light versie uit te breiden. Dit is gebeurd in **Fase 7**. Deze fase is geplaatst na **Fase 6** omdat er dan de meeste zekerheid is dat de uitbreidingen ook daadwerkelijk gerealiseerd worden.

## 5 Onderzoek

Het onderzoek dat uitgevoerd is voor dit project bestaat uit vijf onderdelen. De eerste drie onderdelen zijn aan het begin van het project uitgevoerd. De laatste twee zijn tijdens het project uitgevoerd. De eerste twee onderzoeken zijn uitgevoerd om een beeld te krijgen hoe het project uitgevoerd zou moeten worden. Het derde onderzoek is uitgevoerd om te bepalen welke programmeertaal er gebruikt zou worden. Het vierde onderzoek is uitgevoerd om een grafiek te kunnen maken op basis van de risico's. Het vijfde onderzoek is uitgevoerd om inzicht te krijgen in Android en zodoende een Android applicatie te kunnen ontwikkelen.

### 5.1 Nationale gegevens

Aan het begin van het project is er onderzoek gedaan naar de beschikbare gegevens. Er is tijdens het onderzoek naar gegevens vooral gekeken naar welke gegevens zijn er en hoe specifiek zijn die gegevens. Zijn de gegevens bijvoorbeeld van een branche of zijn ze gespecificeerd op beroepen. Er is onderzoek gedaan naar deze twee punten omdat er bij verschillende gegevens, verschillende oplossingen horen.

#### 5.1.1 Type gegevens

Allereerst is er onderzoek gedaan naar de verschillende soorten gegevens. Uit dit onderzoek is naar voren gekomen dat er 27 verschillende soorten gegevens beschikbaar zijn. Dit loopt uiteen van ongelukken met dodelijke afloop tot wat voor soort letsel er opgelopen is. De volledige lijst van gegevens kunt u vinden in het onderzoeksdocument.

Van deze 27 worden er in de applicaties 13 gebruikt. Deze worden gebruikt omdat ze nodig zijn om de risico's te berekenen, omdat ze het shock effect versterken of omdat ze punten blootleggen waar de veiligheid verbeterd kan worden.

De gegevens die niet gebruikt zijn, zijn niet gebruikt omdat ze niet voor een shock effect zorgen, omdat ze niet in de goede vorm beschikbaar waren, omdat het om algemene kennis gaat of omdat het invullen van de gegevens voor bedrijven erg langdurig zou worden.

#### 5.1.1 Specifiekheid gegevens

Na het onderzoek naar de verschillende soorten gegevens is er onderzoek gedaan naar de specifiekheid van gegevens. Er is hierbij gekeken naar hoe de gegevens opgesplitst zijn. Er zijn hiervoor twee verschillende manieren gevonden.

- Niet opgesplitst
- Opgesplitst per branche

##### 5.1.1.1 Niet opgesplitst

Het voordeel van het gebruik van deze gegevens is dat de gegevens in meer categorieën opgesplitst zijn. De gegevens van leeftijd zijn bijvoorbeeld opgesplitst in 12 categorieën terwijl bij gegevens per branche leeftijd is opgesplitst in 3 categorieën. Een andere reden om deze gegevens te gebruiken is omdat er gegevens in staan die bij splitsing op branche niet aanwezig zijn.

##### 5.1.1.2 Opgesplitst per branche

Het voordeel van het gebruik van deze gegevens is dat de gegevens gespecificeerd zijn op branche en dat het risico, wat berekend wordt, beter overeenkomt met de werkelijkheid. Een bijkomend voordeel is dat de gegevens van beroep op deze manier beter gespecificeerd zijn en er ook meerdere beroepen zijn.

#### 5.1.1.4 Keuze

Op basis van deze voordelen is er gekozen om primair de gegevens opgesplitst per branche te gaan gebruiken. De reden hiervoor is dat het risico dan beter overeenkomt met het werkelijke risico. De gegevens die niet beschikbaar zijn per branche zijn ondersteunende gegevens. Deze gegevens kunnen daarom wel meegenomen worden. Dit omdat deze gegevens los staan van andere gegevens en ze niet gebruikt worden bij het berekenen van het risico. Door voor deze oplossing te kiezen zijn alle gegevens beschikbaar en komt het berekende risico beter overeen met de werkelijkheid.

## 5.2 Formules

Na het onderzoek naar de gegevens is er onderzoek gedaan naar de formules. Dit begon met het specificeren van de stappen die de formules moeten doen. Daarna is er onderzoek gedaan naar eventuele gelijksoortige formules die al bestaan. Het onderzoek naar de wiskundige notaties is tijdens het onderzoek naar de bestaande formules en tijdens het schrijven van de formules uitgevoerd. Deze onderdelen zijn onderzocht zodat, mochten er al formules bestaan, er geen nieuwe formules geschreven hoeven te worden. Het onderzoek naar de wiskundige notaties is gedaan omdat verschillende notaties onbekend waren.

### 5.2.1 Stappen formules specificeren

Het onderzoek naar de formules is begonnen met het specificeren van wat de formules moeten doen. Dit omdat er anders niet bekend is waar er naar gezocht moet worden. Tijdens het opstellen van de stappen die de formules moeten doen, bleek dat de formules minder complex waren dan vooraf gedacht. De stappen die de formules doorlopen kunt u vinden in het onderzoeksdocument.

### 5.2.2 Huidige formules

Doordat de formules minder complex bleken te zijn is het onderzoek naar de huidige formules niet erg uitgebreid geweest, omdat het schrijven van nieuwe formules zeer waarschijnlijk niet lang zou duren. Het onderzoek is gedurende één dag uitgevoerd en het enige bruikbare wat het heeft opgeleverd is kennismaking met twee sites met een wiskundig forum. Bij deze site zijn ook veel wiskundigen actief die hulp kunnen bieden bij eventuele problemen.

### 5.2.3 Wiskundige notaties

Het onderzoek naar de wiskundige notaties is uitgevoerd zodat de wiskundige notaties die niet bekend waren begrepen worden. Het onderzoek is uitgevoerd tijdens de overige onderzoeken en tijdens het schrijven van de formules. Het onderzoek werd pas actief als er een wiskundige notatie was die niet bekend was. Zodra deze notatie bekend was werd het onderzoek ook weer gestopt. Dit zodat er niet onnodig veel tijd besteed werd aan het uitzoeken van notaties die niet van toepassing zijn op de formules. Hierdoor is het onderzoek niet erg uitgebreid geweest. Doordat bestaande formules, die gevonden zijn, niet van toepassing waren op dit project zijn de onderzoeken naar de wiskundige notatie in die formules weggelaten. Hierdoor heeft dit onderzoek geen eigen stuk in het onderzoeksdocument.

## 5.3 Programmeertaal

Het onderzoek naar de te gebruiken programmeertaal is uitgevoerd omdat de keuze voor de programmeertaal een vrije keuze was. Er is tijdens het onderzoek gekeken naar 2 programmeertalen. Dit omdat de programmeertalen zowel een web als een desktop applicatie moesten kunnen bouwen. Verder heeft meegespeeld dat enkel de talen Java en C# bekend waren. Dit zijn dan ook de talen die onderzocht zijn. Uit het onderzoek kwamen de volgende punten naar voren

### 5.3.1 Java + JSP/J2EE

#### 5.3.1.1 Voordelen

Java bekend en mee geprogrammeerd  
JSP/J2EE bekend en mee geprogrammeerd  
Platform onafhankelijk  
Open source

#### 5.3.1.1 Nadelen

JSP/J2EE is lastig in gebruik  
Mindere performance

### 5.3.2 C# + ASP.NET

#### 5.3.2.1 Voordelen

C# bekend en mee geprogrammeerd  
Betere performance  
Bedrijfsbegeleider heeft ervaring met Visual basic  
ASP.NET lijkt makkelijk te gebruiken

#### 5.3.2.2 Nadelen

Extern programma nodig om te draaien op Linux en Apple OS  
ASP.NET onbekend en niet mee geprogrammeerd

#### 5.3.2.3 Persoonlijk voordeel

Leren ASP.NET programmeren

### 5.3.3 Keuze

Na het afwegen van de voor- en nadelen is gebleken dat C# en ASP.NET de geschiktste programmeertaal is. Dit is vanwege de volgende punten:

- Betere performance dan Java. Dit was een belangrijk punt voor de opdrachtgever
- ASP.NET, ondanks onbekend, sneller te programmeren dan JSP/J2EE
- Server draait op een Windows omgeving dus geen extern programma nodig
- ASP.NET is een nieuwe taal en dus uitbreiding van kwaliteiten

## 5.4 Grafiek

Het onderzoek naar een grafiek heeft betrekking op de library die gebruikt wordt bij de Full versie. De library wordt gebruikt om een grafiek te maken en deze op te slaan in een file. Hier is onderzoek naar gedaan omdat er geen standaard library in C# zit om een grafiek te maken. Er zijn verschillende libraries te vinden op internet. Er is enkel gekeken naar gratis libraries. Dit omdat er geen kosten met het project gemoeid zijn. Er zijn in totaal drie libraries onderzocht. ZedGraph, WebChart en Nplot.

### 5.4.1 ZedGraph

#### 5.4.1.1 Voordelen

- Eenvoudig in gebruik
- Veel tutorials
- Alles is aan te passen
- Goede support
- Meest gebruikt
- Open source

#### 5.4.1.2 Nadelen

- Lastig te doorgronden

### 5.4.2 Webchart

#### 5.4.2.1 Voordelen

- Tutorials
- Open source

#### 5.4.2.2 Nadelen

- Simpele grafieken
- Weinig aanpasbaar
- Bedoeld voor ASP.NET

### 5.4.3 Nplot

#### 5.4.3.1 Voordelen

- Open source

#### 5.4.3.2 Nadelen

- Weinig tutorials
- Simpele grafieken
- Weinig gebruikt

### 5.4.4 Keuze

Na het bekijken van de drie libraries is de afweging gemaakt tussen de voor- en nadelen. Hieruit kwam naar voren dat ZedGraph de beste optie is. Dit omdat, ondanks dat het lastig te doorgronden is, het de meeste functionaliteit biedt, de grootste community heeft en dus de meeste support en tutorials heeft.

## 5.5 Android

Het onderzoek naar Android was nodig omdat Android een tot nu toe onbekende taal was. Het onderzoek omvat dan ook de meest vitale en basis aspecten van Android. Allereerst te beginnen met de programmeertaal zelf, vervolgens wordt er gekeken naar hoe een GUI ontworpen moet worden, vervolgens wordt er gekeken naar een lokale database connectie en als laatste zal er gekeken worden naar de mogelijkheid voor een remote database. Deze onderwerpen worden onderzocht om allereerst een begin te kunnen maken met programmeren en een GUI te kunnen bouwen. Er wordt gekeken naar de lokale database connectie om de gegevens in een database op te slaan en zo de performance hoog te houden. De remote database wordt onderzocht om te kijken of het op die manier mogelijk is een lokale database periodiek te vullen zodat er geen internet connectie nodig is voor de applicatie.

### 5.5.1 Programmeertaal

Het onderzoek naar de programmeertaal Android was vooral een onderzoek naar hoe wordt Android geprogrammeerd. In welke taal wordt het geprogrammeerd, met welke IDE kan het geprogrammeerd worden en hoe kan het getest worden.

#### 5.5.1.1 Taal

Als eerste is er gekeken naar de taal waar Android in geprogrammeerd wordt. Dit bleek Java te zijn voor de achterkant en XML voor het maken van een GUI. Ten tweede is er gekeken naar in welke IDE er dan geprogrammeerd kan worden. Bij Java programmeren komen twee IDE naar voren, Eclipse en Netbeans. Eclipse is de officiële IDE van Android. Deze wordt ook ondersteund door Android. Netbeans kan ook gebruikt worden omdat er een 3rd party programma is om Android programmeren mogelijk te maken.

#### 5.5.1.2 IDE

De keuze voor de te gebruiken IDE is gevallen op Netbeans. Dat de voorkeur naar Netbeans gaat is omdat dit een IDE is waar al eerder mee gewerkt is en waar de functionaliteit van begrepen wordt. Een andere reden is dat Eclipse erg lastig is om goed te leren gebruiken en aangezien de Android applicatie pas later is toegevoegd is er geen tijd om uitgebreid onderzoek te doen naar het werken met Eclipse.

#### 5.5.1.3 Testen

Het onderzoek naar het testen van Android applicaties was een kort onderzoek. Dit bleek na kort zoeken te kunnen met het 3rd party programma wat nodig is om Android te kunnen programmeren in Netbeans.



### 5.5.2 GUI ontwerp

Het onderzoek naar het maken van een GUI was een erg belangrijk onderzoek. Dit was zo belangrijk omdat de GUI naast het opslaan van gegevens het enige deel is van de Android applicatie. Het was ook erg belangrijk omdat het maken van een GUI via XML gebeurt. Waar vooraf nog geen ervaring mee was.

#### 5.5.2.1 Theorie GUI

Als eerste is er onderzoek gedaan naar hoe het maken van een GUI in zijn werk gaat. Alle stappen van het maken van de componenten tot het gebruiken van de componenten. Dit is gedaan om duidelijk te krijgen wat de theorie achter het GUI maken is. Dit was ook erg belangrijk om een goed beeld te krijgen van welke stappen er bij Android allemaal ondernomen moeten worden.

#### 5.5.2.2 Componenten

Als tweede is er onderzoek gedaan naar de verschillende componenten. Waarbij voornamelijk gekeken is naar welke functionaliteit deze componenten allemaal hebben. Dit is gedaan om ervoor te zorgen dat de componenten die gebruikt worden ook de componenten zijn die de juiste functionaliteit hebben. Zodat er niet opnieuw begonnen hoeft te worden als blijkt dat de gewenste functionaliteit niet beschikbaar is.

#### 5.5.2.3 Declareren

Tegelijkertijd met het onderzoek naar de functionaliteit van de componenten is er ook gekeken naar hoe de componenten gedeclareerd en gedefinieerd moeten worden. Dit is gedaan zodat de gewenste functionaliteit van de componenten aangeroepen kan worden op de juiste momenten.

#### 5.5.2.4 Events

Als laatste is er gekeken naar het afhandelen van de events die afgevuurd worden door de componenten. Dit is erg belangrijk om te onderzoeken omdat dit vaak op verschillende manieren gebeurt en je voor elk event ook moeten weten welke handler je nodig hebt.

### 5.5.3 Database connectie

Het onderzoek naar de database connectie is uitgevoerd omdat het opslaan van gegevens in een database zorgt voor een betere performance dan opslag in files of in de code zelf. Het onderzoek bestond uit een onderzoek naar de te gebruiken database, het inrichten van de database en het gebruik van de database.

#### 5.5.3.1 Database

Bij het onderzoek naar de te gebruiken database bleek dat SQLite inbegrepen is bij Android devices. Dit is om die reden de database die gebruikt wordt.

#### 5.5.3.2 Inrichten

Bij het onderzoek naar het inrichten van de database bleek dat dit op twee verschillende manieren kan.

- Het inrichten van een SQLite database op een pc en deze kopiëren naar het device
- Het inrichten van een SQLite database via SQL statements in de code

Het voordeel van het inrichten op een pc is dat dit vaak makkelijker is omdat er gewoon aangegeven kan worden via knoppen waar de database uit moet bestaan. Dit is bij SQL statements vaak lastig zeker als er een vooraf onbekende database wordt gebruikt. Er is echter toch gekozen om het via SQL statements te doen. Dit omdat het uiteindelijke product gedownload wordt en het dan niet mogelijk is om een database mee te downloaden. Hierdoor moet er bij het eindproduct gebruikt gemaakt worden van SQL statements en is het handig om dit meteen te doen zodat er geen dubbel werk is.

#### 5.5.3.3 Gebruik

Het onderzoek naar het gebruik van de database is gedaan om erachter te komen hoe de database aangeroepen kan worden voor alle DML activiteiten. Uit het onderzoek bleek dat dit op twee verschillende manieren kan.

- Het gebruik van SQL statements
- Het gebruik van methodes

Bij het gebruik van SQL statements worden de statements geschreven en vervolgens uitgevoerd. Bij het gebruik van methodes worden, aan vooraf gedefinieerde methodes, gegevens meegegeven die men normaal via SQL statements aangeeft. Deze methodes maken hier dan SQL statements van en voeren deze uit. Er is gekozen om de DML activiteiten te doen via methodes. Dit omdat het niet veel minder is voor de performance, maar dat de moeilijkheidsgraad wel een stuk lager ligt.

#### 5.5.4 Remote database & Webservice

Bij het onderzoek naar de remote database zijn er vier punten waar naar gekeken moest worden.

##### 5.5.4.1 Web server

Het onderzoek naar de web server is enkel van belang voor de test omgeving. Dit omdat de server waar het uiteindelijk op zal draaien een web server is. Voor het onderzoek is er gekeken naar twee servers.

- Xampp
- Wamp

Bij beide web servers zijn Apache, MySQL, PHP en PHPMyAdmin standaard meegeleverd. Beide zijn open source en beide zijn te installeren op Windows. Het verschil tussen de twee is niet erg groot. Het enige verschil is dat Wamp eenvoudiger werkend te krijgen is. Aangezien de server niet veel gebruikt zal worden en enkel en alleen voor test doeleinden is er gekozen om met Wamp te werken.

##### 5.5.4.2 Versturen

Het onderzoek naar het versturen van de gegevens heeft geresulteerd in twee mogelijkheden HTTP en HTTPS. Omdat het niet om bedrijfsgevoelige informatie of persoonsgegevens gaat is er gekozen om gebruik te maken van HTTP Post en HTTP Get.

##### 5.5.4.3 Server applicatie

Voor het onderzoek naar de applicatie die de HTTP verzoeken afhandelt op de server moest eerst worden gekeken wat deze applicatie allemaal zou moeten kunnen. Hieruit bleek dat het een eenvoudige applicatie moest zijn die enkel een verzoek binnen krijgt, de gegevens uit de database haalt en deze terug stuurt naar de aanvrager.

Als programmeertalen waarin deze applicatie geschreven zou kunnen worden is er gekeken naar drie talen.

- JSP/J2EE
- ASP.NET
- PHP

Omdat het programma eenvoudig is en niet veel activiteiten hoeft uit te voeren is er gekozen voor PHP. Dit omdat JSP/J2EE en ASP.NET erg uitgebreid zijn en veel meer tijd zouden kosten voor deze eenvoudige activiteiten. PHP is hier veel beter voor te gebruiken.

##### 5.5.4.4 Formaat

Doordat PHP de gekozen taal is was er weinig onderzoek nodig naar het formaat waarin de data verstuurd zou worden. Dit omdat bij zowel PHP als JAVA/Android het versturen van gegevens via JSON mogelijk is. Er is dus ook gekozen voor het gebruik van JSON voor het versturen van de data.

## 6 Ontwerp

In dit hoofdstuk zal uitleg worden gegeven waarom bepaalde keuzes zijn gemaakt in het ontwerp van de applicaties. Dit zal voornamelijk gaan over het ontwerp van de GUI omdat de overige onderdelen voor zichzelf spreken of omdat ze niet uitgebreid behandeld zijn.

### 6.1 Light versie

Bij de light versie zal er uitleg gegeven worden over de keuzes van de GUI en over een aantal keuzes in het klassendiagram.

#### 6.1.1 GUI ontwerp

Bij het GUI ontwerp van de light versie is er rekening gehouden met twee verschillende soorten invoer. Invoer voor totaal bedrijf en invoer per werknemer (Afbeelding 6.1).

Mocht er door de gebruiker gegevens niet of foutief ingevuld worden dan wordt er boven aan het scherm in het rood getoond wat er fout of niet ingevuld is en worden de invoervelden rood gekleurd. Dit is gedaan om ervoor te zorgen dat de gegevens die ingevuld zijn, correct zijn en zodat als er iets fout gaat de gebruiker snel kan zien waar dat het fout gegaan is zodat dit makkelijk en snel aan te passen is.

##### 6.1.1.1 Invoer per werknemer

Bij de invoer per werknemer is er een tabel geplaatst om de werknemers te tonen. Hier is voor een tabel gekozen omdat deze het duidelijkst de informatie weer geeft. Dat is hier belangrijk omdat de persoon die de gegevens invult na moet kunnen lopen of de gegevens correct zijn.

De gegevens worden ingevuld door middel van listboxen. Bij een listbox kan een gebruiker een vooraf ingestelde waarde selecteren. Er is hier gekozen voor listboxen omdat er enkel opties ingevuld kunnen worden die ook bestaan in de brongegevens, dit omdat het risico anders niet berekent kan worden.

Zodra een gebruiker de werknemer toevoegt verschijnt deze in de tabel en blijven de listboxen hun geselecteerde waarde houden. Dit gebeurt zodat gelijksoortige werknemers snel ingevuld kunnen worden zonder weer alle listboxen te moeten selecteren.

##### 6.1.1.2 Invoer totaal bedrijf

Bij invoer totaal bedrijf is er gekozen om twee rijen van labels en tekstvelden naast elkaar te plaatsen. Dit is gedaan omdat door de lange tekst van sommige beroepen er op een aantal plaatsen niet meer dan twee naast elkaar passen. De tekstvelden zijn op een vaste plaats, allemaal recht onder elkaar, neer gezet. Dit is gedaan zodat het overzichtelijk blijft welke gegevens allemaal ingevuld zijn en welke gegevens nog ingevuld moet worden. Ook helpt dit om ervoor te zorgen dat duidelijk is welke gegevens waar zijn ingevuld.

### 6.1.2 Klassendiagram


Het Klassendiagram van de light versie is opgebouwd uit zeven folders.

- GUI
- Communicatie
- Beheer
- Reken
- Domain
- Database
- Reader


De folders Reader, Database, Domain en Reken zitten alleen aan Beheer gekoppeld. Dit omdat beheer alle aanvragen tussen de klassen beheert. Communicatie zit tussen Beheer en de GUI.

Communicatie zorgt voor de communicatie tussen deze klassen. Verder zorgt Communicatie ervoor dat Domain niet aan GUI gekoppeld zit. Dit is gedaan zodat domain en GUI onafhankelijk van elkaar aangepast kunnen worden. Dit heeft als voordeel dat de applicatie veel beter te beheren en verbeteren is nadat hij in gebruik is genomen.

Een schematische weergave van het klassendiagram is terug te vinden in het ontwerpdocument.

**RISICO CALCULATOR**


Home
Help
Beheer

Database Connectie 

**RISICO CALCULATOR**

Dit is de risico calculator van RoOp.

Bedrijfsnaam 
Adres 
Plaats 
Postcode

Telefoonnummer 
Email 
Branche

Beroep	Geslacht	Leeftijd	Opleiding	Herkomst	Contract	Werktijden	Wijzig
Machine bankwerkers, monteurs instrumentenmakers, reparateurs	Man	15-29 jaar	Basis/MAVO/VMBO	Autochtoon	Vast	0-24 uur	<input type="button" value="Wijzig"/>
Machine bankwerkers, monteurs instrumentenmakers, reparateurs	Man	15-29 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>
Elektromonteurs, reparateurs van elektrische apparaten	Man	15-29 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>
Elektromonteurs, reparateurs van elektrische apparaten	Vrouw	15-29 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>
Elektromonteurs, reparateurs van elektrische apparaten	Vrouw	30-49 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>
Elektromonteurs, reparateurs van elektrische apparaten	Man	50-64 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>
Elektromonteurs, reparateurs van elektrische apparaten	Man	50-64 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>

Beroep 
Geslacht 
Leeftijd

Opleiding 
Herkomst 
Contract 
Werktijden

Als u alle werknemers heeft ingevuld drukt up de knop "Gereed"

Afbeelding 6.1

## 6.2 Full versie

### 6.2.1 GUI ontwerp

#### 6.2.1.1 Database

Bij de invoer via de database kan er gezocht worden op alle aspecten van een bedrijf. Dit is op deze manier gerealiseerd zodat zelfs met minimale informatie het bedrijf gevonden kan worden. Voldoen er meer bedrijven aan de gespecificeerde attributen dan volgt er een pop up met een lijst van bedrijven waaruit de gebruiker kan kiezen. Er is hier voor een pop up gekozen omdat deze omhoog springt waardoor het voor de gebruiker meteen duidelijk is dat hij een bedrijf moet kiezen.

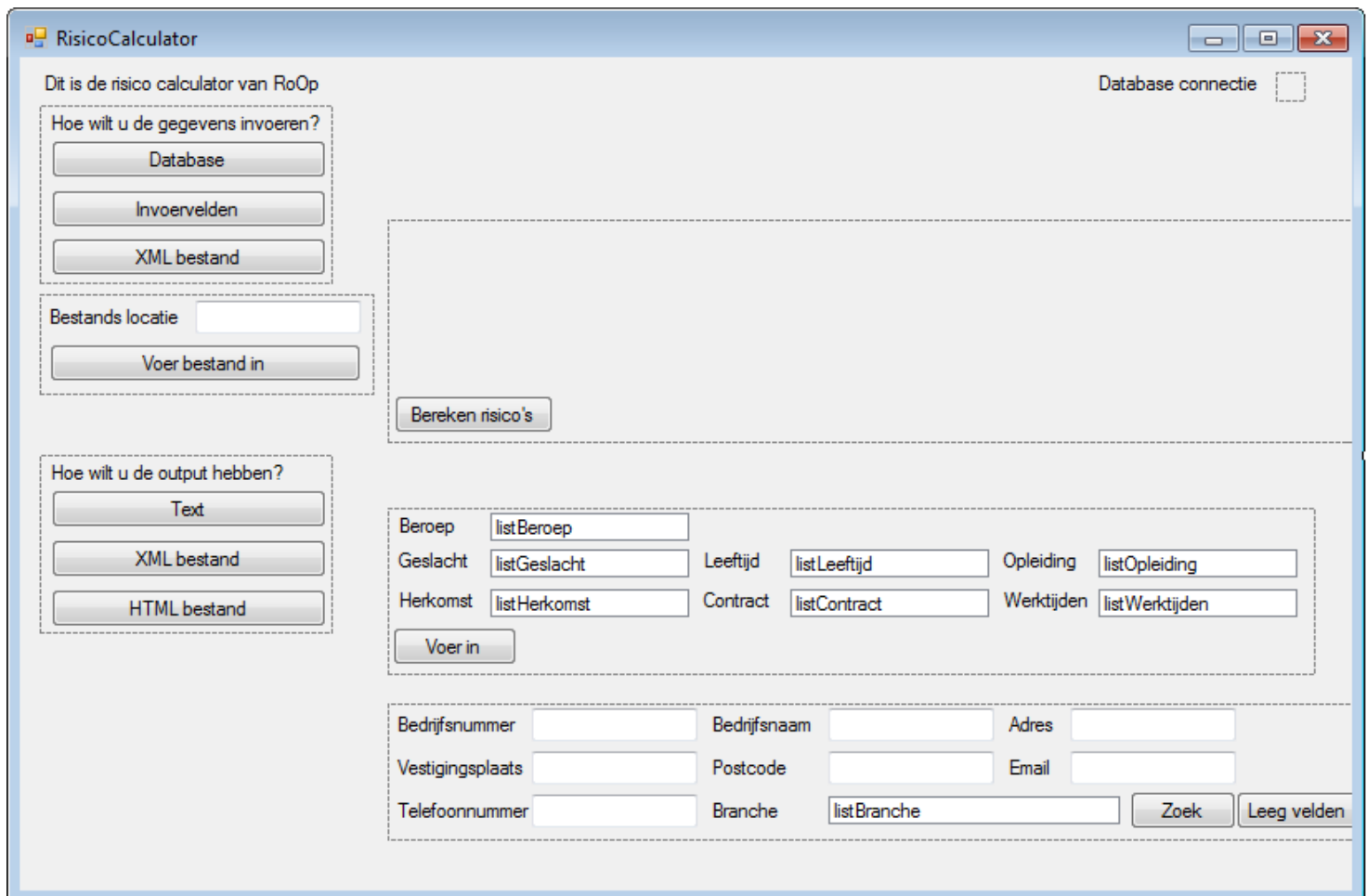
#### 6.2.1.2 Invoervelden

Bij de invoer via invoervelden is er voor dezelfde manier gekozen als bij de invoer per werknemer van de light versie.

### 6.2.2 Klassendiagram

Het klassendiagram van de full versie komt, op de GUI en reader/writer klassen na, volledig overeen met die van de light versie. Het verschil in de GUI is enkel en alleen het verschil tussen web en desktop, de functionaliteit is grotendeels gelijk.

Het verschil tussen de reader/writer klassen is dat er bij de full versie meerdere reader/writer klassen zijn. Dit is zo gedaan zodat het duidelijk is dat er in verschillende formaten kan worden gelezen en geprint. Verder is dit gedaan omdat er verschillende functionaliteit in deze klassen zit. Sommige kunnen alleen lezen, andere kunnen ook schrijven.



The screenshot shows the 'RisicoCalculator' application window. The title bar reads 'RisicoCalculator'. Inside the window, there is a message: 'Dit is de risico calculator van RoOp'. In the top right corner, there is a 'Database connectie' checkbox. The main interface is divided into several sections:

- Hoe wilt u de gegevens invoeren?**: Contains three buttons: 'Database', 'Invoervelden', and 'XML bestand'.
- Bestands locatie**: A text input field followed by a 'Voer bestand in' button.
- Hoe wilt u de output hebben?**: Contains three buttons: 'Text', 'XML bestand', and 'HTML bestand'.
- Bereken risico's**: A button located below the input sections.
- Input fields for personal data**: A group of fields including 'Beroep' (with a dropdown 'listBeroep'), 'Geslacht' (with a dropdown 'listGeslacht'), 'Leeftijd' (with a dropdown 'listLeeftijd'), 'Opleiding' (with a dropdown 'listOpleiding'), 'Herkomst' (with a dropdown 'listHerkomst'), 'Contract' (with a dropdown 'listContract'), and 'Werktijden' (with a dropdown 'listWerktijden'). Below these is a 'Voer in' button.
- Company information fields**: A group of fields including 'Bedrijfsnummer', 'Bedrijfsnaam', 'Adres', 'Vestigingsplaats', 'Postcode', 'Email', 'Telefoonnummer', and 'Branche' (with a dropdown 'listBranche').
- Buttons**: 'Zoek' and 'Leeg velden' buttons are located at the bottom right of the company information section.

Afbeelding 6.2

## 6.3 Voorraad

### 6.3.1 GUI ontwerp

#### 6.3.1.1 Zoeken

Het zoeken van de netten kan, net als bij de full versie, op alle attributen. Als er meer netten gevonden worden, wordt de lijst, links onderin, gevuld met de id en afmeting van de netten. Deze worden niet in een pop up getoond omdat er bij deze applicatie meestal alleen gezocht wordt op Tag nummer en deze geeft maar één resultaat. Een andere reden is dat er op deze manier makkelijker door de netten gezocht kan worden, omdat als je een ander net selecteert de gegevens wijzigen.

#### 6.3.1.2 Wijzig status

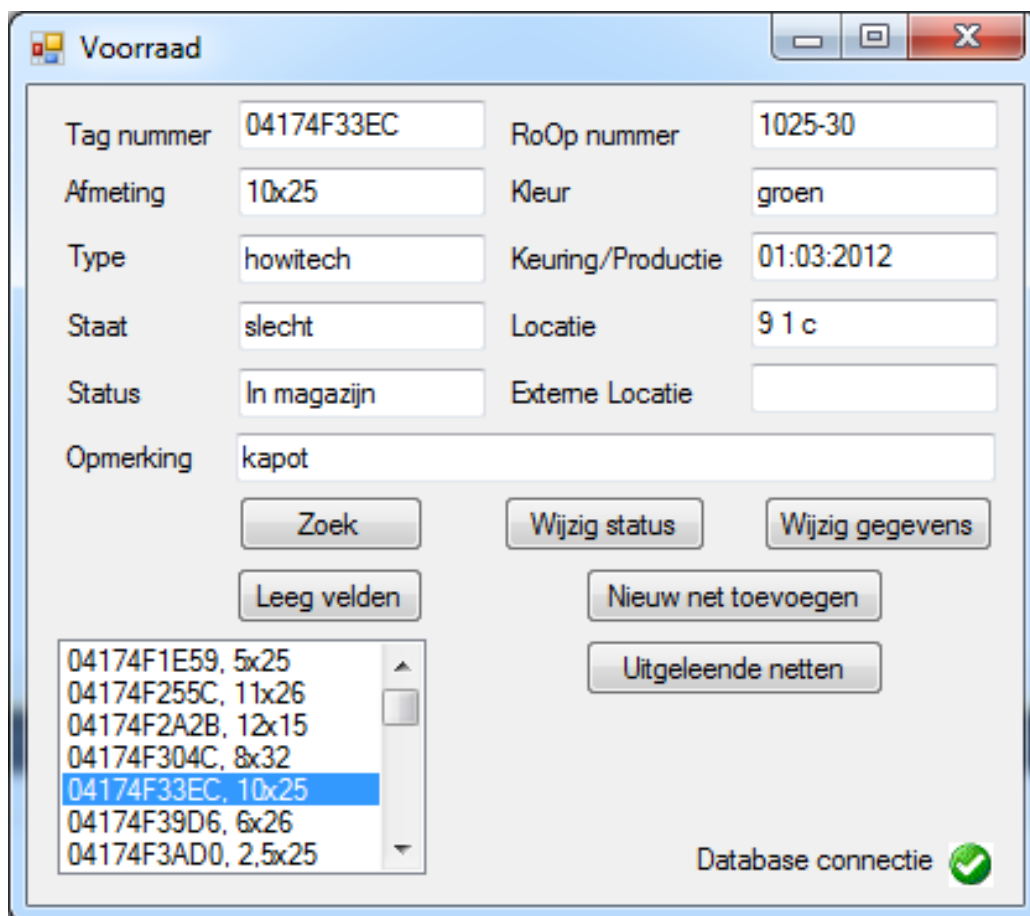
Bij het wijzigen van de status wordt er wel gebruik gemaakt van een pop up. In dit geval is een pop up handiger om meteen de aandacht te vragen en omdat het wijzigen slechts één keer hoeft.

#### 6.3.1.3 Nieuw net toevoegen

Het toevoegen van een nieuw net gebeurt door de velden in te vullen die gebruikt worden om gegevens te tonen. Dit omdat de GUI dan minder groot hoeft te zijn en hij overzichtelijker blijft. Er wordt tijdens het toevoegen van nieuwe netten ook gecontroleerd of de gegevens niet al bestaan zodat er niet per ongeluk een nieuw net toegevoegd kan worden.

#### 6.3.1.4 Uitgeleende netten


De knop Uitgeleende netten geeft een pop up met daarin een tabel van alle uitgeleende netten. Op deze manier is het makkelijk een goed overzicht te krijgen van welke netten er zijn uitgeleend. De reden dat er een tabel gebruikt wordt is omdat de gegevens dan beter zichtbaar zijn.



**Voorraad**

Tag nummer	04174F33EC	RoOp nummer	1025-30
Afmeting	10x25	Kleur	groen
Type	howitech	Keuring/Productie	01:03:2012
Staat	slecht	Locatie	9 1 c
Status	In magazijn	Externe Locatie	
Opmerking	kapot		

04174F1E59, 5x25  
 04174F255C, 11x26  
 04174F2A2B, 12x15  
 04174F304C, 8x32  
**04174F33EC, 10x25**  
 04174F39D6, 6x26  
 04174F3AD0, 2,5x25

Database connectie 

Afbeelding 6.3

## 6.4 Android

### 6.4.1 GUI ontwerp

#### 6.4.1.1 Activiteit selecteren

Het selecteren van een activiteit gebeurt via een spinner. Een spinner werkt door erop te klikken en dan verschijnt er een pop up met de opties waaruit gekozen kan worden. Dit is in dit geval gewenst omdat de gebruiker wel alle opties wil kunnen zien, maar zodra hij gekozen heeft moet het scherm gevuld kunnen worden met andere informatie.

#### 6.4.1.2 Maatregel selecteren

Het selecteren van een maatregel gebeurt via een listview. Een listview werkt door alle opties ten alle tijden te tonen en wanneer er op een optie geklikt wordt volgt er een actie. Er wordt hier gebruikt gemaakt van een listview omdat de gebruiker alle maatregelen wil kunnen zien en er moet ook op geklikt worden als er extra informatie over de maatregel gewenst is.

#### 6.4.1.3 Update

De knop "Update" is er zodat de gebruiker de lokale database kan bijwerken via de remote database. Dit kan alleen als er internet is. Er is gekozen om dit door de gebruiker te laten doen omdat er maar zeer zelden een update in de data komt. Als het programma periodiek zou gaan kijken zou dit performance kosten en tevens zou het programma dan alleen kunnen updaten als er een internetverbinding beschikbaar is. Een gebruiker kan als hij een update wil internet activeren en zo de update aanvragen. Het programma kan dit niet.



Afbeelding 6.4



## 7 Realiseren

### 7.1 Risico Calculator

#### 7.1.1 Berekeningen

Bij het berekenen van het risico voor een werknemer is er gekozen om eerst in de database te zoeken of er al een werknemer bestaat met exact dezelfde eigenschappen. Dit gebeurt omdat het doorzoeken van de database sneller is dan het berekenen van het risico. Op deze manier is het op den duur ook niet meer nodig om te berekenen omdat dan alle mogelijke werknemers berekent zijn.

Bij het berekenen van het risico op meerdere gewonde werknemers wordt er gebruik gemaakt van een BigInteger. Dit is een object wat extreem grote getallen kan bevatten. Dit is nodig omdat bij het rekenen met faculteiten de getallen al te groot zijn voor een Long op het moment dat een bedrijf 20 werknemers heeft. Om ervoor te zorgen dat de getallen ook decimale waarden kunnen weergeven worden de getallen die ingevoerd worden vermenigvuldigd door een groot getal en later weer gedeeld als de waarde in een tekst wordt gezet.

#### 7.1.2 Light versie

Bij de light versie zijn er verschillende variabelen die voor langere tijd en voor meerderen klassen nodig zijn. Er zijn 3 verschillende variabelen.

- Application
- ViewState
- Session

De application variabelen worden gebruikt om ervoor te zorgen dat een medewerker van Roop verschillende onderdelen van de website aan en uit kan zetten. Dit gebeurt via application variabelen omdat deze voor iedereen die de applicatie gebruikt dezelfde zijn. Hierdoor hoeft er niet gelezen en geschreven te worden naar een database of file om de instellingen van de website goed te krijgen.

De viewState variabelen worden gebruikt voor gegevens die gedurende de levenscyclus van een pagina moeten bestaan. Dit zijn viewState variabelen omdat bij asp.net elk event voor een nieuwe pagina aanroep zorgt. Hierdoor kunnen gegevens niet in de klasse opgeslagen worden. De viewState variabelen verdwijnen pas als er naar een andere pagina genavigeerd wordt.

De session variabelen worden gebruikt voor gegevens die gedurende het gebruik van de website, voor één gebruiker, nodig zijn, of voor gegevens die nodig zijn voor één pagina maar die niet in een viewState kunnen. De session variabelen die voor één gebruiker zijn, zijn nodig om de instellingen van die ene gebruiker op te slaan.

#### 7.1.3 Full versie

Bij het maken van de grafiek van risico's is er gekozen om een taart grafiek te maken. Dit omdat deze het duidelijkst weergeeft wat de grootste risico's zijn. Omdat er voor de meest uiterste gevallen risico's worden berekent zijn er ook erg kleine getallen in de grafiek. Dit maakt het allemaal weer onduidelijker. Er is dan ook gekozen om de risico's onder de 1,0 samen te voegen en hier één groter taart stuk van te maken. Er wordt dan bij beschreven om welke risico's het gaat zodat het nog steeds mogelijk is om conclusies te trekken.

## 7.3 Android

### 7.3.1 List Layout

Er is bij de Android applicatie gekozen om een eigen list layout te maken. Deze wordt gebruikt om de gegevens weer te geven in de listview. Er is gekozen om een eigen layout te maken omdat de beschikbare layouts niet voldoen. De layouts die bestonden en uiterlijk goed waren, konden niet gecentreerd worden. Hierdoor werd de applicatie onbruikbaar voor verschillende scherm groottes. Bij de layouts die wel te centreren waren was de tekst niet goed leesbaar of was de tekst juist te groot. De gemaakte layout is een combinatie van de verschillende layouts en dus te centreren en de tekst is leesbaar.

## 8 Tools & Technieken

### 8.1 Tools

Er zijn voor het ontwikkelen van het product verschillende tools gebruikt. De meeste van deze tools zijn open source. Sommige waren al in bezit en er is ook een trial gebruikt van één tool. Dit is gedaan om de kosten voor het project op 0 euro te houden zodat er geen risico aan het project verbonden is. In dit hoofdstuk staan enkel de tools beschreven die nog niet eerder uitgelegd zijn.

#### 8.1.1 Microsoft Visual studio

De tool die het meeste gebruikt is, is Microsoft Visual studio. Dit is een IDE die gebruikt wordt om onder andere C# en ASP.NET te programmeren. Er is gekozen om deze IDE te gebruiken, omdat deze al eerder gebruikt is en er geen tijd geïnvesteerd hoeft te worden om een andere IDE te leren gebruiken. Wat ook heeft meegespeeld is dat deze IDE de officiële IDE is voor het programmeren van C# en ASP.NET. Microsoft visual studio is een programma waar een licentie voor nodig is, maar dit programma was al in bezit dus is er geen licentie aangeschaft.

#### 8.1.2 Navicat

Navicat is een database viewer. Hiermee kun je een database beheren. Er is bij dit project gekozen om Navicat te gebruiken, omdat deze erg uitgebreid is en bijvoorbeeld tabellen automatisch kan vullen aan de hand van Excel files. Aangezien de bron gegevens in Excel files staan was dit een snelle manier van de gegevens verwerken. Voor het gebruik van Navicat is een licentie nodig, maar er kon ook gewerkt worden met een 30 dagen trial versie. Dit is in dit geval gekozen omdat de database alleen aangemaakt en gevuld hoefden te worden en dit niet lang in beslag nam.

#### 8.1.3 PHPMyAdmin

PHPMyAdmin is net als Navicat een database viewer. PHPMyAdmin is echter een stuk minder uitgebreid en daarom ook alleen gebruikt om de eenvoudige database op de web server in te richten. De reden dat hier PHPMyAdmin is gebruikt, is omdat deze standaard meegeleverd is met de Wamp server en omdat de database niet ingewikkeld is.

#### 8.1.4 Microsoft office Visio

Microsoft office Visio is een programma waarmee ontwerpen voor software gemaakt kunnen worden. Dit programma is gebruikt voor het maken van ERD's, klassendiagrammen en sequentie diagrammen. Er is gekozen om gebruik te maken van Visio omdat dit programma afgestemd is op C#. Bij het ontwerpen van de diagrammen kunnen dan ook de gegevenstypes die in C# bestaan getoond worden. Dit heeft als voordeel dat het precies overeenkomt met de werkelijkheid.

## 8.2 Technieken

Er zijn tijdens het project ook verschillende technieken gebruikt. Deze lopen uiteen van programmeertalen tot ontwerpstandaards. De technieken die gebruikt zijn worden vaak gebruikt. Op deze manier kan er verzekerd worden dat er altijd hulp gezocht kan worden. In dit hoofdstuk staan enkel de technieken beschreven die nog niet eerder uitgelegd zijn.

### 8.2.1 MySQL

MySQL is een databasetaal. Dit is de taal waarin databases geschreven worden. Er is bij dit project gebruik gemaakt van MySQL omdat het open source is en Roop al met MySQL werkt. Doordat Roop al met MySQL werkt hoeft er geen nieuwe database geïnstalleerd te worden er hoeft alleen een nieuwe aangemaakt en gevuld worden.

### 8.2.2 UML

UML is een standaard die gebruikt wordt voor het ontwerpen van klassendiagrammen. Bij dit project is er gekozen om UML te gebruiken omdat deze veel gebruikt wordt. De meeste software engineers kennen UML en kunnen UML diagrammen gebruiken. Dit is handig omdat het bij dit project om een afstudeerstage gaat en er na de afstudeerstage door andere mensen aan de applicatie gewerkt wordt mocht dit nodig zijn.

## 9 Conclusies en Aanbeveling

Om het succes van het project te beoordelen is er gekeken naar de opdracht zoals deze in het PID is beschreven. Er zijn enkele verschillen met de opdracht specificatie en het uiteindelijk opgeleverde product. Het grootste verschil is dat er in de opdracht specificatie stond dat er bij de light versie alleen basis gegevens ingevuld hoefden te worden. Achteraf is er gekozen om bij de light versie zowel de mogelijkheid te geven om basis gegevens in te vullen, als de mogelijkheid om alle gegevens in te vullen. Dit is gedaan omdat dan later niet overlegd hoeft te worden om de correcte gegevens te krijgen. Een ander verschil is dat de resultaten allemaal berekent worden bij de light versie en dat niet alle risico's getoond worden.

Het project is sneller dan verwacht afgerond en hierdoor was er tijd om uitbreidingen toe te voegen aan het project. Dit betreft zowel uitbreidingen voor het product, uitbreidingen op het onderwerp als losstaande uitbreidingen. Deze uitbreidingen zijn vooraf niet gedefinieerd en kunnen dus ook niet aan de hand van vooraf gemaakte definities beoordeeld worden. De losstaande uitbreiding kan echter wel beoordeeld worden aan de hand van de verwachtingen van de gebruiker. Over deze applicatie, de voorraad applicatie, is te zeggen dat hij voldoet aan de verwachtingen van de gebruikers.

Als er gekeken wordt naar het totale project kan er gezegd worden dat dit project een succes is geweest. De initiële opdracht is met succes afgerond en er zijn aan de initiële applicatie uitbreidingen toegevoegd. Er zijn ook nog extra applicaties gebouwd die zijn ook met succes afgerond. Dit succes is afgemeten aan de ervaringen van de gebruikers of aan de functionaliteit van de applicatie.

Als aanbeveling zou ik aan Roop doen om, als de Android applicatie aanslaat, deze applicatie ook te ontwikkelen in Apple IOS omdat dit een veel gebruikt besturingssysteem is en dit meer klandizie op zou kunnen leveren.

## Evaluatie

### Waarom deze opdracht

In het laatste semester van de opleiding informatica had ik de mogelijkheid om een afstudeeropdracht te kiezen uit een door Fontys beschikbaar gestelde lijst van afstudeeropdrachten. Bij het bekijken van deze opdrachten sprong er voor mij één opdracht uit. Deze sprong eruit omdat het een combinatie was van programmeren en wiskunde. Twee taken die ik erg leuk vind om te doen. Het onderwerp van de opdracht, het berekenen van een risico, leek mij dan ook erg leuk. Het feit dat het de bedoeling was om een webapplicatie te maken was voor mij ook een reden om contact te zoeken met RoOp. Dit omdat ik mezelf verder wilde ontwikkelen in het programmeren van webapplicaties.

### Ervaring met de opdracht

Zoals ik verwacht had voor het begin van de stage is het een erg leuke opdracht geweest. Een bijkomend voordeel is dat de opdracht sneller af was dan gedacht en er hierdoor een aantal opdrachten bijgekomen zijn. Met de toevoeging van deze opdrachten is de opdracht ook erg divers geworden. Hierdoor is de opdracht nog leuker geweest dan hij al was.

### Wat heb ik geleerd

Het belangrijkste wat ik geleerd heb zijn de nieuwe programmeertalen. Dit betreft ASP.NET en Android. Dit zijn twee talen die veel gebruikt worden en ook nog veel groeien. Hier ga ik dus nog veel plezier van hebben. Door het programmeren in ASP.NET heb ik ook geleerd om een website te bouwen en heb ik geleerd wat er belangrijk is bij de GUI van een website. Ik heb ook mijn programmeer kwaliteiten bijgeschaafd en dan voornamelijk in C#. Ik heb C# lange tijd niet gebruikt dus was het goed om deze een keer bij te schaven. Ook heb ik bij C# bijgeleerd hoe ik XML en HTML kan lezen en schrijven. Dit is erg handig omdat er vaak met XML en HTML documenten gewerkt wordt.

Ik heb voor mezelf ook gekeken waar het nuttig zou zijn als er PLSQL geïmplementeerd zou worden. Omdat er als database met MySQL gewerkt werd was het niet mogelijk om dit ook daadwerkelijk te implementeren, maar ik heb wel geleerd dat PLSQL erg handig zou kunnen zijn. Zeker bij internet applicatie waarbij de gegevens door gebruikers ingevuld worden. Ik heb ook het een en het ander geleerd over de wiskundige formules die nodig zijn voor het berekenen van risico's en mogelijkheden.

Een ander belangrijk aspect wat ik geleerd heb is dat plannen erg belangrijk is. Meestal ontstaat er een probleem omdat projecten langer duren dan verwacht, maar in dit geval was het zo dat het project te snel af was. Dit heeft niet zozeer tot problemen geleid, maar af en toe wel tot situaties waarbij ik niet wist wat ik na het weekend zou moeten gaan doen. Verder heb ik geleerd om te werken in een bedrijfssituatie en heb ik mijn zelfstandig werken verbeterd.

## Literatuurlijst

### Internet bronnen

- [msdn.microsoft.com](https://msdn.microsoft.com)
- [www.asp.net](https://www.asp.net)
- [www.w3schools.com](https://www.w3schools.com)
- [nl.wikipedia.org](https://nl.wikipedia.org)
- [stackoverflow.com](https://stackoverflow.com)
- [developer.android.com](https://developer.android.com)
- [zedgraph.sourceforge.net](https://zedgraph.sourceforge.net)
- [sourceforge.net](https://sourceforge.net)
- [wiskundeforum.nl](https://wiskundeforum.nl)
- [wetenschapsforum.nl](https://wetenschapsforum.nl)

## Bijlage A: PID



# PROJECT: Risico Calculator

---

( Project Initiation Document )



## Documenthistorie

### Revisies

Versie	Status	Datum	Wijzigingen
0.1	Concept	21-2-2012	Eerste opzet van het document
0.2	Work in progress	22-2-2012	In alle hoofdstukken wijzigingen doorgevoerd.
0.3	Feedback	23-2-2012	Planning toegevoegd en wijzigingen in H3
1.0	Final	29-5-2012	Feedback verwerkt

### Goedkeuring

Dit document heeft de volgende goedkeuringen:

Versie	Datum goedkeuring	Naam	Functie	Paraaf

### Distributie

Dit document is verstuurd aan:

Versie	Datum verzending	Naam	Functie
0.3	23-2-2012	Rob van Gassel	Opdrachtgever/bedrijfsbegeleider
0.3	23-2-2012	Martijn Lamers	Docentbegeleider
1.0	29-5-2012	Martijn Lamers	Docentbegeleider

## Managementsamenvatting

### Doel van dit document

Dit document heeft tot doel het project te definiëren, als basis te dienen voor het beheer ervan en de beoordeling van het succes van het project mogelijk te maken.

De twee belangrijkste redenen voor gebruik van dit document zijn:

- Om het project te definiëren.
- Om het succes van het project te kunnen beoordelen.

### Aanleiding

De aanleiding voor dit project is dat bedrijven vaak niet weten wat de kans is dat een werknemer gewond raakt. Hierdoor nemen ze geen adequate maatregelen om dit te voorkomen. Als is een werknemer gewond raakt zitten er voor het bedrijf naast een hoop ongemak ook financiële kosten aan.

Om ervoor te zorgen dat de bedrijven de risico's goed kunnen inschatten moet er een tool ontwikkeld worden die dit voor het bedrijf berekent. Dit moet gebeuren op basis van de nationale cijfers. De nationale cijfers zijn verzameld door TNO in samenwerking met het CBS. De cijfers hebben betrekking op de arbeidsongelukken van een periode.

### Globale aanpak

Om de risico's te berekenen zullen er allereerst formules geschreven moeten worden die, op basis van de landelijke cijfers en de cijfers van het bedrijf, de kans op letsel berekenen. Aan de hand van de resultaten kan het bedrijf dan zien welk risico zij lopen en hoe zij dit het beste kunnen verbeteren.

Na het schrijven van de formules zal de tool ontwikkeld worden. De tool zal bestaan uit twee verschillende versies. Een light versie en een full versie.

De light versie zal een gratis variant worden. Bij de light versie kunnen bedrijven na het invullen van enkel een paar gegevens hun risico berekenen. De light versie zal op internet komen.

De full versie zal een betaalde variant worden. Bij de full versie moeten er meer gegevens ingevuld worden dan bij de light versie. Hierdoor kan er een gedetailleerde risico analyse gegeven worden voor het specifieke bedrijf. De full versie zal anders dan de light versie een desktop applicatie worden.

### Globale kosten en doorlooptijd

Er zijn bij dit project in principe geen kosten van toepassing, enkel bij een goed resultaat omdat er wordt gewerkt met een eindbonus. Het project brengt wat dat betreft ook geen risico's met zich mee. De doorlooptijd van het project is 85 werkdagen en de deadline is 15 juni 2012.

## Inhoudsopgave

1	Inleiding.....	44
1.1	Doel van dit document .....	44
1.2	Opbouw van dit document .....	44
2	Achtergrond .....	45
3	Projectdefinitie.....	46
3.1	Projectdoelstellingen .....	46
3.2	Gekozen oplossing of aanpak .....	46
3.3	Scope van het project .....	46
3.4	Producten c.q. eindresultaat .....	47
3.5	Uitsluitingen .....	47
3.6	Beperkingen .....	47
3.7	Afhankelijkheden .....	47
3.8	Randvoorwaarden .....	47
3.9	Aannames .....	47
4	Projectorganisatiestructuur .....	48
4.1	Opdrachtgever .....	48
4.2	Docent begeleider .....	48
4.3	Stagiair .....	49
4.4	Projectsupport .....	49
Bijlage A:	Communicatieplan .....	50
Bijlage B:	Planning.....	52

## Inleiding

### Doel van dit document

Dit document is opgesteld om alle relevante basisinformatie en uitgangspunten van het project vast te leggen om het project op de juiste wijze te kunnen besturen. Het heeft tot doel het project te definiëren, als basis te dienen voor het beheren van het project en ter beoordeling van het succes van het project.

Dit Project initiatiedocument (of PID) behandelt de volgende fundamentele aspecten van het project:

- Wat hoopt men met het project te bereiken?
- Waarom is het belangrijk om deze doelstellingen te bereiken?
- Wie zijn er betrokken bij het managen van het project en wat zijn hun rollen en verantwoordelijkheden?
- Hoe en wanneer zullen de afspraken, die in dit PID worden besproken, gerealiseerd worden?

Het document wordt gebruikt:

- Om het project te definiëren.
- Om het project te kunnen beheren.
- Om het succes van het project te kunnen beoordelen.

### Opbouw van dit document

Om aan te geven welke onderdelen worden bijgewerkt en dus nieuwe versies zullen krijgen tijdens de voortgang van het project is dit Project initiatiedocument verdeeld in twee secties: een statisch gedeelte en een dynamisch gedeelte:

Het "statische" deel bestaat uit de volgende hoofdstukken en bijlage:

- Achtergrond (Hoofdstuk 2)
- Projectdefinitie (Hoofdstuk 3)
- Projectorganisatiestructuur (Hoofdstuk 4)
- Communicatieplan (Bijlage A)

Het "dynamische" deel bestaat uit een bijlage:

- Planning (Bijlage B)

## Achtergrond

RoOp is een bedrijf dat zich bezig houdt met de veiligheid op de werkvloer. Ze leveren materialen die de veiligheid verhogen voor de werknemers. Doordat bedrijven vaak niet weten wat de kans is dat een werknemer gewond raakt, nemen ze geen adequate maatregelen om dit te voorkomen. Als is een werknemer gewond raakt zitten er voor het bedrijf naast een hoop ongemak ook financiële kosten aan.

Om ervoor te zorgen dat de bedrijven de risico's goed kunnen inschatten moet er een tool ontwikkeld worden die dit voor het bedrijf berekent. Dit moet gebeuren op basis van de landelijke cijfers.

Om de risico's te berekenen zullen er allereerst formules geschreven moeten worden die, op basis van de landelijke cijfers en de cijfers van het bedrijf zelf, de kans op letsel berekenen. Aan de hand van de resultaten kan het bedrijf dan zien welk risico zij lopen en hoe zij dit het beste kunnen verminderen.

Na het schrijven van de formules zal de tool ontwikkeld worden. De tool zal bestaan uit twee verschillende versies. Een light versie en een full versie.

De light versie zal een gratis variant worden. Bij de light versie kunnen bedrijven na het invullen van alleen enkele basis gegevens het globale risico berekenen. Bij de light versie zullen enkel een paar gegevens getoond worden. De light versie zal op internet komen.

De full versie zal een betaalde variant worden. Bij de full versie zullen alle risico's berekend worden. Hierdoor kan er een gedetailleerde risico analyse gegeven worden voor het specifieke bedrijf. De full versie zal anders dan de light versie een desktop applicatie worden.

## Projectdefinitie

### Projectdoelstellingen

Op de werkvloer raken veel mensen gewond. Dit kost bedrijven elk jaar handen vol met geld. Als de bedrijven de veiligheid zouden verhogen zou dit een hoop geld besparen, maar bedrijven zien dit vaak niet in omdat ze alleen naar de directe kosten kijken. Het doel van dit project is om bedrijven in te laten zien dat het verhogen van de veiligheid op de werkvloer hen een hoop kosten en rompslomp bespaart. Als de bedrijven namelijk de veiligheid verhogen hebben hun werknemers minder kans op letsel. Hierdoor zullen de bedrijfskosten voor letsel omlaag gaan.

### Gekozen oplossing of aanpak

Het project is opgedeeld in 4 fases die hieronder verder gespecificeerd zullen worden.

In **Fase 1** zullen de statische formules ontwikkeld worden. Om deze te kunnen ontwikkelen zal er eerst onderzocht moeten worden welke gegevens er allemaal zijn. Van deze gegevens wordt bekeken welke nuttig zijn en welke gebruikt kunnen worden. Na het bekijken van de gegevens zal er onderzoek gedaan worden naar hoe de formules opgesteld moeten worden. Ook zal er gekeken worden of er al formules bestaan die gelijkenissen bevatten met de formules die nu ontwikkeld worden.

In **Fase 2** zal de tool en de database ontworpen worden. Dit omvat zowel de light als de full versie. Het ontwerpen zal gebeuren aan de hand van eerder opgestelde requirements. Allereerst zal er onderzoek gedaan worden naar welke programmeertaal er gebruikt zal worden. Vervolgens zullen er use cases gemaakt worden. Daarna worden er klassendiagrammen gemaakt en als laatste zullen er ook nog sequentie diagrammen gemaakt worden. Na het ontwerpen van de tool zal de database ontworpen worden. Ook worden de scripts voor de database geschreven.

In **Fase 3** zal de light versie van de tool ontwikkeld worden. Het ontwikkelen van de light versie zal beginnen met het maken van de GUI. Daarna zal stap voor stap de functionaliteit toegevoegd worden. Als laatste zullen er Unit tests geschreven worden en zal de light versie handmatig getest worden aan de hand van de use cases.

In **Fase 4** zal de full versie van de tool ontwikkeld worden. Deze ontwikkeling is vergelijkbaar met die van de light versie. Dus, er zal eerst een GUI gemaakt worden. Waarna stap voor stap de functionaliteit toegevoegd zal worden. Als laatste zullen er Unit tests geschreven worden en zal de full versie handmatig getest worden aan de hand van de use cases.

### Scope van het project

Binnen de scope van het project valt het ontwikkelen van de formules en de tool, maar ook het kiezen van de programmeertaal waarin de tool ontwikkeld zal worden, het opzetten van een database en het verwerken van de landelijke risico-gegevens. Het maken van goede documentatie is ook erg belangrijk omdat de applicatie in de toekomst door iemand anders beheerd zal worden.

## Producten c.q. eindresultaat

Aan het einde van het project zullen de volgende producten opgeleverd worden/zijn.

- Documentatie m.b.t. het verwerken van de landelijke gegevens. Dit bevat de keuzes over welke gegevens gebruikt zullen worden en hoe ze gebruikt kunnen worden.
- Documentatie m.b.t. de formules. Dit bevat de formules zelf, uitleg hoe de formules werken, resultaten van het onderzoek naar de formules.
- Documentatie m.b.t. de keuze over programmeertalen.
- Een light versie (webbased) van de tool. Hierbij zitten ook de Unit tests.
- Een full (desktop) versie van de tool. Hierbij zitten ook de Unit tests.
- Documentatie m.b.t. de tool. Dit bevat de use cases, de klassendiagrammen, sequentie diagrammen en rapportage over de tests.
- Een database inclusief de bijbehorende documentatie en scripts.
- Eindpresentatie.
- Eindrapportage.

## Uitsluitingen

De hardware om de tools op te draaien zal aanwezig zijn en hier hoeft niet op gelet te worden.

## Beperkingen

Beschikbaar Budget A	Beschikbare resources	Gewenste opleverdata	Beschikbare Doorlooptijd
680 uur	Landelijke gegevens	16-6-2012	85 werkdagen

## Afhankelijkheden

De kwaliteit van de risicoanalyse voor de bedrijven is afhankelijk van de kwaliteit van de landelijke gegevens en de kwaliteit van de gegevens het bedrijf zelf. Verder zijn er geen afhankelijkheden.

## Randvoorwaarden

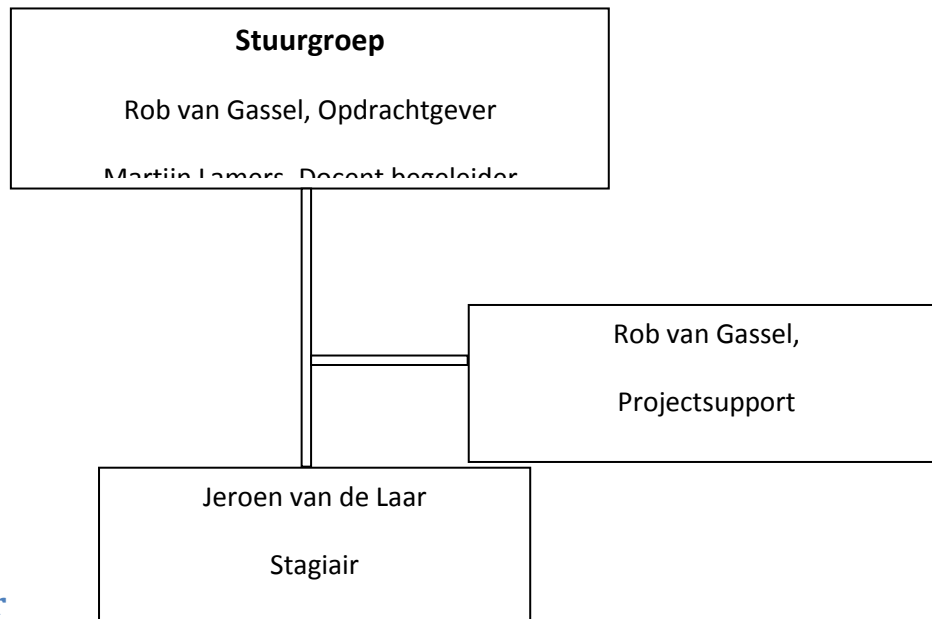
Er hoeft buiten een functioneel werkstation niets aanwezig te zijn voor het project.

## Aannames

Er zijn geen aannames gemaakt.



## Projectorganisatiestructuur



### Opdrachtgever

#### Rolbeschrijving

De opdrachtgever geeft een mondelinge briefing van de opdracht, en leest het PID. Als er verder nog vragen zijn kan er contact met hem worden opgenomen. De opdrachtgever moet duidelijk zijn wensen en eisen overbrengen.

#### Projectgerelateerde taken

Beantwoorden van vragen die in directe relatie zijn met het project.  
Tussentijdse controle.  
Verantwoordelijk voor de goedkeuring van het eindresultaat.

#### Specifieke verantwoordelijkheden

De opdrachtgever is verantwoordelijk voor aanleveren van eventuele benodigde informatie en documentatie met betrekking tot de scope van het project op verzoek van de stagiair.  
Verantwoordelijk voor de goedkeuring aan het einde van het project.

### Docent begeleider

#### Rolbeschrijving

De docent begeleider moet op basis van de eisen die door de school opgelegd zijn controleren of het project hieraan voldoet.

#### Projectgerelateerde taken

Controle van het PID.  
Controle van het eindverslag.

#### Specifieke verantwoordelijkheden

De docent begeleider is verantwoordelijk voor het meten van het project aan de door de school opgelegde eisen.

## **Stagiair**

### **Rolbeschrijving**

De stagiair is degene die het project uitvoert.

### **Projectgerelateerde taken**

Afspraken over doelstellingen en randvoorwaarden.

Een PID opstellen.

Beheersen van de voortgang van het project.

Maken van de producten.

### **Specifieke verantwoordelijkheden**

De stagiair is verantwoordelijk voor het op tijd en goed afkrijgen van het project.

## **Projectsupport**

### **Rolbeschrijving**

Projectsupport biedt hulp aan de stagiair wanneer nodig. Mocht de stagiair een probleem tegenkomen waar hij zelf niet uit komt, een mening over een onderwerp zoeken of zelf niet genoeg expertise hebben, dan wordt projectsupport ingeschakeld.

### **Projectgerelateerde taken**

Ondersteunen van de stagiair wanneer hij problemen tegenkomt die hij zelf niet kan oplossen. Het bieden van een second opinion.

### **Specifieke verantwoordelijkheden**

De projectsupport is verantwoordelijk voor ondersteuning van de stagiair wanneer hij daar om vraagt.

## Bijlage A: Communicatieplan

### Inleiding

Dit communicatieplan benoemt alle partijen die een (positief of negatief) belang hebben bij het project en de wijze waarop zij bij het project zullen worden betrokken en welke communicatievormen daarbij gebruikt worden. Het gaat hierbij om partijen en communicatie buiten de formele projectmanagementstructuur zoals beschreven in het PID.

### Belanghebbenden bij het project

Wie	Namens	Belang	Communicatievorm
<i>persoon</i>	<i>groep, afdeling</i>	<i>Welk belang bij project?</i>	<i>Hoe wordt deze persoon betrokken bij het project?</i>
Rob van Gassel	Opdrachtgever	Controle/Rapporteren	Email, mondeling
Rob van Gassel	Projectsupport	Advies	Mondeling
Martijn Lamers	Docent begeleider	Controle/Rapporteren	Email, mondeling
Jeroen van de Laar	Stagiair	Uitvoeren	Email, mondeling

## Communicatiekanalen

Van	Naar	Informatie	Medium	Frequentie of data
<i>Persoon of groep</i>	<i>Persoon of groep</i>	<i>Soort informatie</i>	<i>Email, telefoon, memo, rapport</i>	
Opdrachtgever	Stagiair	Opdracht geven	Mondeling	Meerdere malen
Stagiar	Opdrachtgever	Rapportage	Mondeling, email	Wekelijks
Projectsupport	Stagiair	Advies	Mondeling	Wekelijks/Dagelijks
Stagiair	Docent begeleider	Rapportage	Email, mondeling	Meerdere malen
Stagiair	Alle Betrokkenen	Informereren	Email	Bij onvoorziene omstandigheden

Partij:	Docent-begeleider	Opdrachtgever	Project-support	Stagiair
<b>Rapport:</b>				
<b>PID</b>	G/T	G/T	A	O
<b>Voortgangsrapport</b>		G		O
<b>Fase-eindrapport</b>	A	I		O
<b>Projectissues</b>			A	O
<b>Project documentatie</b>		G	A/T	O
<b>Leerpunten rapport</b>	I	I		O
<b>Projecteindrapport</b>	G/T	G		O

### Legenda:

<b>O</b> Opstellen	<b>A</b> Adviseren	<b>I</b> Ontvangen ter informatie
<b>T</b> Toetsen	<b>D</b> Distribueren/archiveren	<b>G</b> Goedkeuren

## Bijlage B: Planning

Wanneer?	Wat wordt er gedaan?	Wat is er af?
Week 1	<b>Fase 1 formules</b> Opdracht verduidelijken  PID maken  Afspraken maken  Onderzoek doen naar de landelijke cijfers  Onderzoek doen naar formules	1ste versie requirements  PID
Week 2	Onderzoek doen naar de landelijke cijfers  Onderzoek doen naar formules  Schrijven van formules	Gegevens voor uiteindelijke database
Week 3	Schrijven van formules	Formules
Week 4	<b>Fase 2 ontwerpen</b> Requirements verfijnen  Onderzoek programmeer taal  Ontwerpen usecases	Requirements  Keuze programmeertaal  Usecases
Week 5	Ontwerpen klassendiagrammen	
Week 6	Ontwerpen klassendiagrammen  Ontwerpen sequentie diagrammen	Klassendiagrammen  Sequentie diagrammen
Week 7	Ontwerpen database  Opzetten database	Database ontwerp  Database
Week 8	<b>Fase 3 light versie</b> Programmeren Light versie tool	GUI ontwerp light versie
Week 9	Programmeren Light versie tool	
Week 10	Programmeren Light versie tool  Testen light versie tool	Light versie tool  Unit tests light versie, Rapportage over het testen
Week 11	<b>Fase 4 full versie</b> Programmeren Full versie tool	GUI ontwerp full versie
Week 12	Programmeren Full versie tool	
Week 13	Programmeren Full versie tool	
Week 14	Programmeren Full versie tool  Testen full versie tool	Full versie tool
Week 15	Testen Full versie tool	Unit tests full versie, Rapportage over het testen
Week 16	Uitloop	
Week 17	Uitloop	

## Bijlage B: Onderzoeksdocument

# Onderzoeksdocument

---

## Inhoudsopgave

1 Gegevens .....	55
1.1 lijst van gegevens .....	55
1.2 Specifiekheid gegevens .....	55
2 Formules .....	56
2.1 Stappen formules .....	56
3 Programmeertaal .....	57
3.1 Java + JSP/J2EE .....	57
3.2 C# + ASP.NET .....	57
4 Grafiek .....	58
4.1 ZedGraph .....	58
4.2 Webchart .....	58
4.3 Nplot .....	58
5 Android .....	59
5.1 Taal .....	59
5.2 IDE .....	59
5.3 Test .....	59
5.4 GUI .....	59
5.5 Database .....	59
5.6 Remote database .....	60

## 1 Gegevens

### 1.1 lijst van gegevens

#### 1.1.1 Gebruikte gegevens

Beroep  
Dodelijk  
Verzuim  
Geslacht  
Leeftijd  
Opleiding  
Herkomst  
Bedrijfsomvang  
Contractvorm  
Werkuren  
Oorzaak ongeluk  
Type letsel  
Locatie letsel

#### 1.1.2 Niet gebruikte gegevens

Ploegendienst  
Autonomie  
Tijdsdruk  
Gevaarlijk werk  
Kracht zetten  
Lawaai  
Ongewenst gedrag(Extern)  
Ongewenst gedrag(Intern)  
Ongeval in verkeer  
Locatie ongeluk  
Voorwerp ongeluk  
Overwerk  
Avond/Nachtwerk  
Weekendwerk

### 1.2 Specifiekheid gegevens

#### 1.2.1 Opgesplitst per branche

Risico reëler  
Meer beroepen

#### 1.2.2 Niet opgesplitst

Gegevens opgedeeld in meer categorieën  
Meer verschillende gegevens



## 2 Formules

### 2.1 Stappen formules

#### 2.1.1 Formule invoer per werknemer

##### Stap 1

Neem het aantal werknemers van bedrijfsomvang.

--> werknemers

Neem het aantal gewonden van bedrijfsomvang.

--> gewonden

##### Stap 2

Uit beroep neem je het percentage van de werknemers(beroep) van de totale bevolking(totaal).

--> %werknemers

Uit beroep neem je het percentage van de gewonden(beroep) van alle gewonden(totaal).

--> % gewonden

##### Stap 3

$[\% \text{werknemers}] * ([\text{werknemers}] / 100) = [\text{werknemers}]$

$[\% \text{gewonden}] * ([\text{gewonden}] / 100) = [\text{gewonden}]$

##### Stap 4

Herhaal stap 2 en 3 voor alle verschillende gegevens.

##### Stap 5

Uitrekenen wat het risico voor de persoon is.

$([\text{gewonden}] / [\text{werknemers}]) * 100 = \text{RisicoPersoon}$

#### 2.1.2 Formule gemiddeld risico bedrijf per werknemer

##### Stap 1

$\text{Risico} = \text{Risico} + \text{Risico werknemer}$

##### Stap 2

Herhaal **Stap 1** voor alle werknemers.

##### Stap 3

$\text{Risico bedrijf} = \text{Risico} / \text{Aantal werknemers}$

#### 2.1.3 Formule gemiddeld risico bedrijf totaal bedrijf

##### Stap 1

$\text{Risico} = \text{Risico} + (\text{Aantal met dit risico} * \text{Risico groep})$

$\text{Totaal} = \text{Totaal} + \text{Aantal in de groep}$

##### Stap 2

Herhaal **Stap 1** voor alle groepen.

##### Stap 3

$\text{Risico bedrijf} = \text{Risico} / \text{Totaal}$

#### 2.1.4 Formule kans op meerdere gewonde werknemers

##### Stap 1

Bereken het aantal mogelijkheden voor het gewond raken van werknemers.

(1 van de 3 gewond is 3 mogelijkheden, g-ng-ng, ng-g-ng, ng-ng-g)

(2 van de 3 gewond is 3 mogelijkheden, g-g-ng, g-ng-g, ng-g-g)

(3 van de 3 gewond is 1 mogelijkheid, g-g-g)

##### Stap 2

Bereken de kans op het voordoen van de situatie.

## **3 Programmeertaal**

### **3.1 Java + JSP/J2EE**

#### **3.1.1 Voordelen**

Java bekend en mee geprogrammeerd  
JSP/J2EE bekend en mee geprogrammeerd  
Platform onafhankelijk  
Open source

#### **3.1.1 Nadelen**

JSP/J2EE is lastig in gebruik  
Mindere performance

### **3.2 C# + ASP.NET**

#### **3.2.1 Voordelen**

C# bekend en mee geprogrammeerd  
Betere performance  
Bedrijfsbegeleider heeft ervaring met Visual basic  
ASP.NET lijkt makkelijk te gebruiken

#### **3.2.2 Nadelen**

Extern programma nodig om te draaien op Linux en apple OS  
ASP.NET onbekend en niet mee geprogrammeerd

#### **3.2.3 Persoonlijk voordeel**

Leren ASP.NET programmeren

## **4 Grafiek**

### **4.1 ZedGraph**

#### **4.1.1 Voordelen**

Eenvoudig in gebruik  
Veel tutorials  
Alles is aan te passen  
Goede support  
Meest gebruikt  
Open source

#### **4.1.2 Nadelen**

Lastig te doorgronden

### **4.2 Webchart**

#### **4.2.1 Voordelen**

Tutorials  
Open source

#### **4.2.2 Nadelen**

Simpele grafieken  
Weinig aanpasbaar  
Bedoeld voor ASP.NET

### **4.3 Nplot**

#### **4.3.1 Voordelen**

Open source

#### **4.3.2 Nadelen**

Weinig tutorials  
Simpele grafieken  
Weinig gebruikt

## 5 Android

### 5.1 Taal

Background gebeurt met Java.

GUI gebeurt met XML

### 5.2 IDE

#### 5.2.1 Eclipse

Officieel gebruikt door Android

Veel functionaliteit

Lastig te leren

#### 5.2.2 Netbeans

3rd party programma voor gebruik van Android

Bekend met werking Netbeans

### 5.3 Test

Mee geleverde Android Virtual device programma met 3rd party programma.

Met AVD worden Android devices opgestart om de applicatie te testen.

### 5.4 GUI

#### 5.4.1 Lists

##### 5.4.1.1 Listview

Toont alle waardes

Geen andere views nodig om waardes in te voeren

Waardes te selecteren

##### 5.4.1.2 Spinner

Toont geselecteerde waarde

Erop klikken om alle waardes te tonen

Waardes te selecteren

Klein formaat

##### 5.4.1.3 Scrollview

Toont alle waardes

Views nodig om waardes in te voeren

Aparte view nodig om waarde te selecteren(bijv. button)

### 5.5 Database

SQLite is standaard aanwezig op Android devices.

Het inrichten van de database kan op twee verschillende manieren.

- Het inrichten van een SQLite database op een pc en deze kopiëren naar het device
- Het inrichten van een SQLite database via SQL statements in de code

DML acties kan op twee verschillende manieren

- Het gebruik van SQL statements
- Het gebruik van methodes

## 5.6 Remote database

### 5.6.1 Web server

#### 5.6.1.1 Xampp

Apache  
MySQL  
PHP  
PHPMyAdmin  
Open source  
Uitgebreid

#### 5.6.1.2 Wamp

Apache  
MySQL  
PHP  
PHPMyAdmin  
Open source  
Eenvoudig te configureren

### 5.6.2 Versturen

HTTP  
Eenvoudig

HTTPS

Veilig voor bedrijfsgevoelige informatie of persoonsgegevens.

### 5.6.3 Server applicatie

#### 5.6.3.1 JSP/J2EE

Lastig te doorgronden  
Uitgebreid

#### 5.6.3.2 ASP.NET

Eenvoudig  
Uitgebreid

#### 5.6.3.3 PHP

Eenvoudig

### 5.6.4 Formaat

#### 5.6.4.1 Tekst

Eenvoudig te versturen  
Lastig te parsen

#### 5.6.4.2 JSON

Eenvoudig te versturen  
Eenvoudig te parsen  
Geïmplementeerd in zowel Android als PHP

## Bijlage C: Ontwerpdocument

# Ontwerpdocument

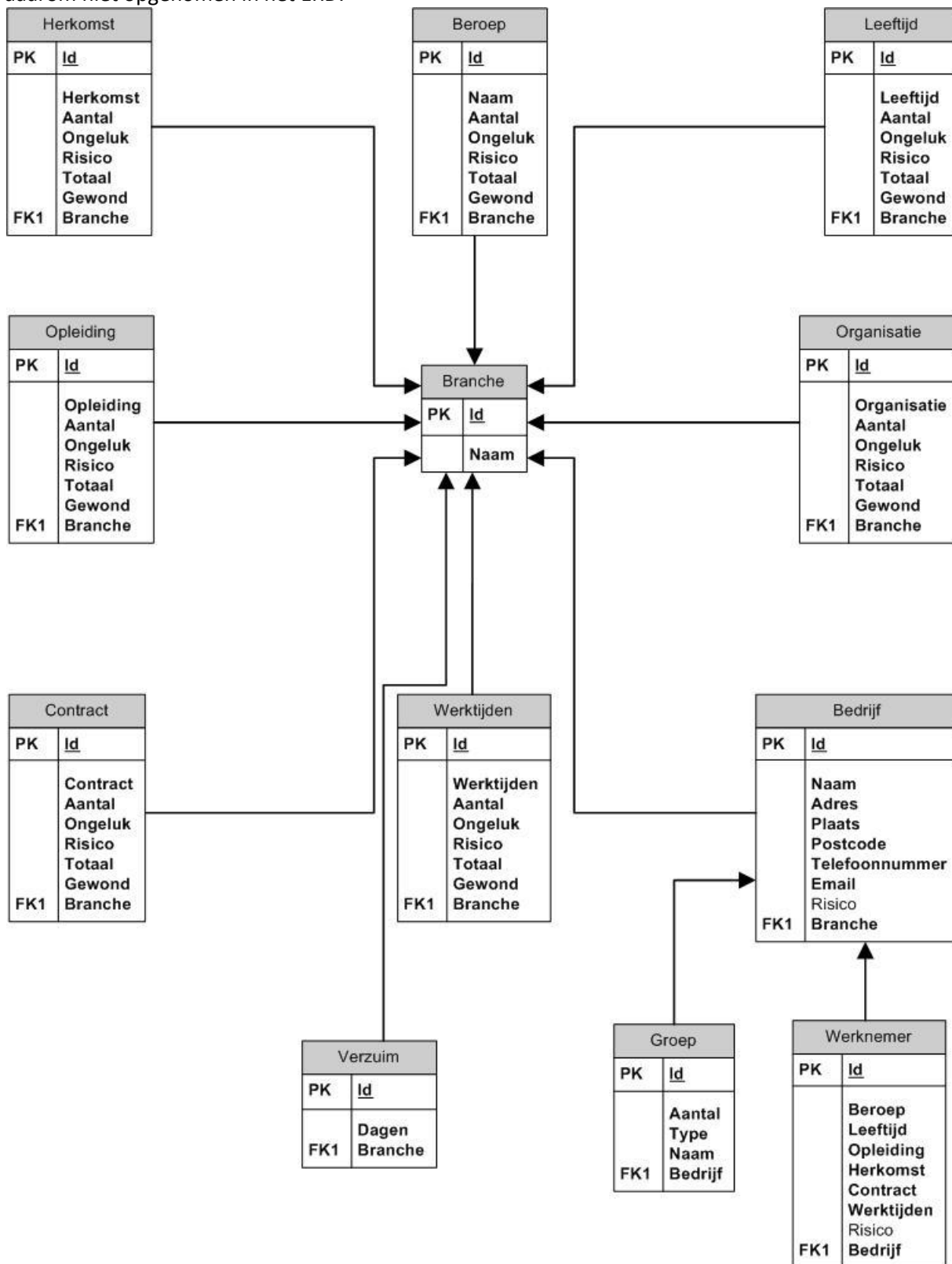
---

## Inhoudsopgave

1 Database ontwerp .....	63
2 Klassendiagrammen .....	64
2.1 Light versie .....	64
2.2 Full versie.....	65
3 Use cases .....	66
3.1 Light versie .....	66
3.1.1 Berekenen risico bedrijf per werknemer .....	66
3.1.2 Berekenen risico bedrijf .....	67
3.2 Full versie.....	68
3.2.1 Berekenen risico bedrijf per werknemer .....	68
4 Sequentie diagrammen .....	69
4.1 Light versie .....	69
4.1.1 Invoer per groep.....	69
4.1.2 Invoer per werknemer.....	70
4.2 Full versie.....	71
4.2.1 Invoer via database .....	71
4.2.2 Invoer via XML.....	72
4.2.3 Invoer via invoervelden .....	73
5 GUI.....	74
5.1 Light versie .....	74
5.1.1 Beheer .....	74
5.1.2 Invoer per werknemer.....	74
5.1.3 Invoer per groep.....	75
5.1.4 Resultaten.....	76
5.1.5 Foutmelding.....	77
5.2 Full versie.....	77
5.3 Voorraad.....	78
5.4 Android.....	78

## 1 Database ontwerp

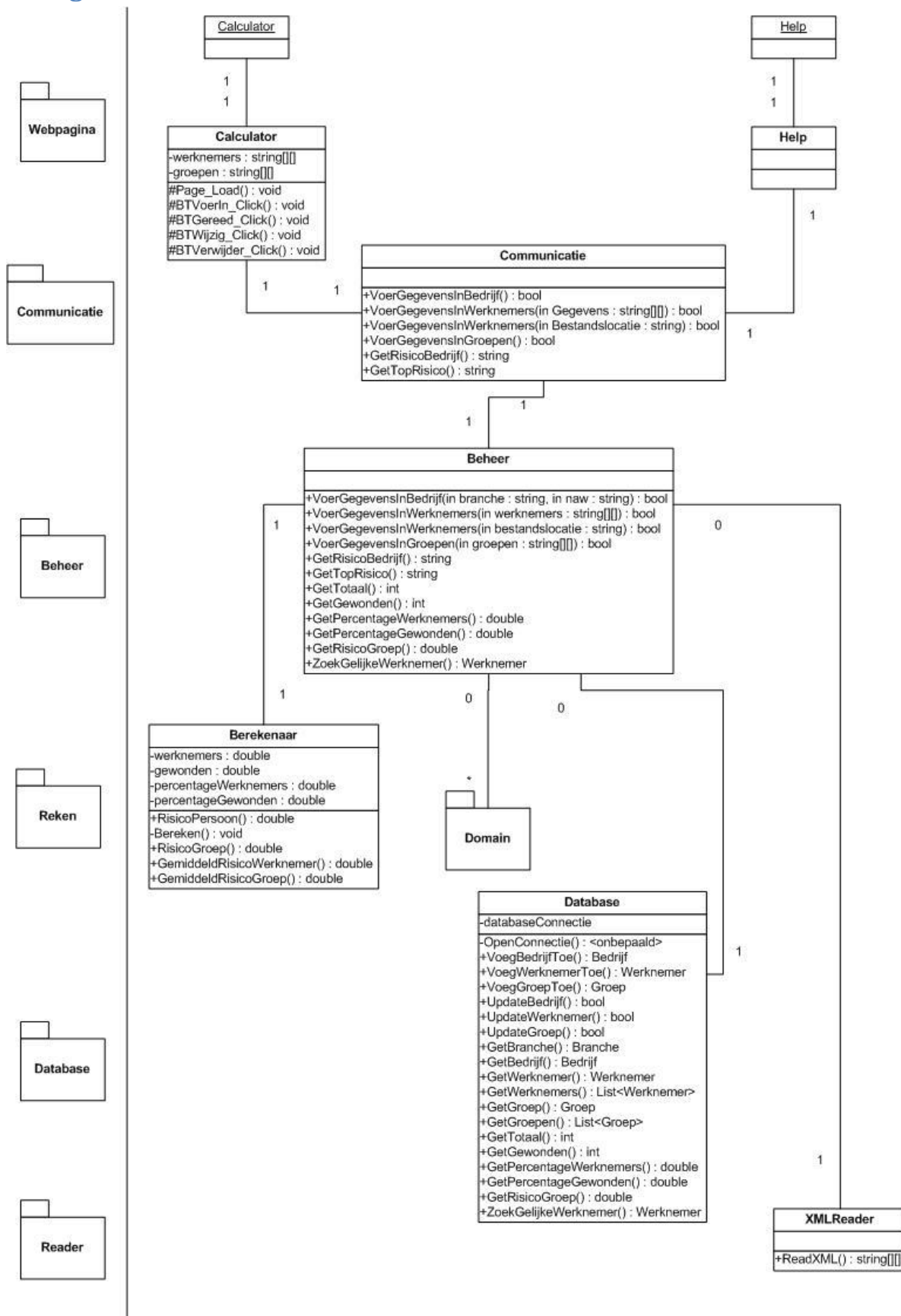
De tabellen Dodelijk, Oorzaak, LetselLocatie en LetselType staan los van alle andere tabellen en zijn daarom niet opgenomen in het ERD.



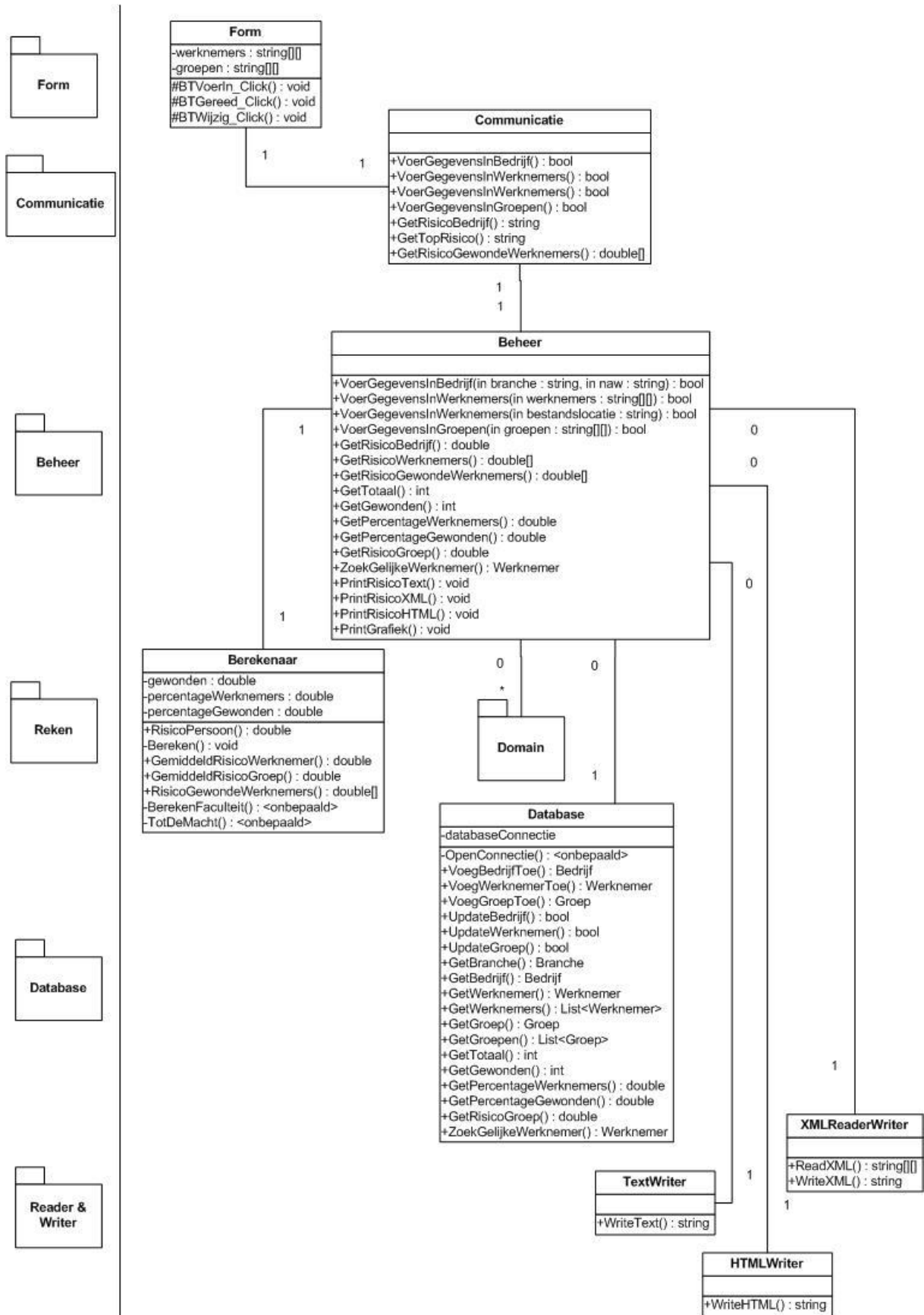


## 2 Klassendiagrammen

### 2.1 Light versie



## 2.2 Full versie



## 3 Use cases

### 3.1 Light versie

#### 3.1.1 Berekenen risico bedrijf per werknemer

##### Goal

De risico's voor het bedrijf per werknemer worden berekent.

##### Rationale

Deze use case is er om het berekenen van de risico's voor het bedrijf per werknemer mogelijk te maken.

##### Actors

Programma, gebruiker.

##### Preconditions

Gebruiker moet verbinding hebben met internet, de gebruiker moet op de website zijn.

##### Post Conditions

##### Successful

De risico's zijn berekent en gedeeltelijk getoond.

##### Unsuccessful

De risico's zijn niet berekent of niet gedeeltelijk getoond.

##### Flow of Events

##### Basic Flow

1. Gebruiker gaat naar de website.
2. Gebruiker klikt op invoer per werknemer.
3. Gebruiker klikt op invoer d.m.v. invoervelden.
4. Programma genereert een form waar de gegevens ingevoerd kunnen worden.
5. Gebruiker vult de contact gegevens in.
6. Gebruiker vult de gegevens van de eerste werknemer in.
7. Gebruiker klikt op voer in.
8. Programma zet de gegevens over in labels en schuift de invoervelden 1 regel naar beneden.
9. Stap 6, 7 en 8 worden herhaalt tot alle werknemers ingevuld zijn.
10. Gebruiker klikt op gereed.
11. Programma controleert de gegevens en slaat ze op in de database.
12. Programma berekent de risico's per ingevoerde werknemer en slaat ze op in de database.
13. Programma berekent het gemiddelde risico voor het bedrijf en slaat deze op in de database.
14. Programma toont het gemiddelde risico en het hoogste risico van een werknemer op het scherm.

##### Alternate Flow

- 3.1 Gebruiker klikt op invoer d.m.v. xml.
  - 3.1.1 Programma genereert een form waar de gegevens ingevoerd kunnen worden.
  - 3.1.2 Gebruiker vult de contact xml gegevens in.
  - 3.1.3 Gebruiker vult de locatie van de file in.
  - 3.1.4 Ga verder bij stap 10.
- 11.1 Contact gegevens zijn niet/ niet correct ingevuld.
  - 11.1.1 Gebruiker wijzigt de contact gegevens.
  - 11.1.2 Ga verder bij stap 10.
- 11.2 Gegevens over werknemer zijn niet/ niet correct ingevuld.
  - 11.2.1 Gebruiker wijzigt de gegevens.
  - 11.2.2 Ga verder bij stap 10.

### 3.1.2 Berekenen risico bedrijf

#### **Goal**

De risico's voor het bedrijf worden berekent.

#### **Rationale**

Deze use case is er om het berekenen van de risico's voor het bedrijf mogelijk te maken.

#### **Actors**

Programma, gebruiker.

#### **Preconditions**

Gebruiker moet verbinding hebben met internet, de gebruiker moet op de website zijn.

#### **Post Conditions**

##### **Successful**

De risico's zijn berekent en gedeeltelijk getoond.

##### **Unsuccessful**

De risico's zijn niet berekent of niet gedeeltelijk getoond.

#### **Flow of Events**

##### **Basic Flow**

1. Gebruiker gaat naar de website.
2. Gebruiker klikt op invoer totaal bedrijf.
3. Gebruiker klikt op invoer d.m.v. invoervelden.
4. Programma genereert een form waar de gegevens ingevoerd kunnen worden.
5. Gebruiker vult de contact gegevens in.
6. Gebruiker vult de gegevens van het bedrijf in.
7. Gebruiker klikt op gereed.
8. Programma controleert de gegevens en slaat ze op in de database.
9. Programma berekent de risico's per ingevoerde groep en slaat ze op in de database.
10. Programma berekent het gemiddelde risico voor het bedrijf en slaat deze op in de database.
11. Programma toont het gemiddelde risico op het scherm.

##### **Alternate Flow**

- 8.1 Contact gegevens zijn niet/ niet correct ingevuld.
  - 8.1.1 Gebruiker wijzigt de contact gegevens.
  - 8.1.2 Ga verder bij stap 7.
- 8.2 Gegevens over groepen zijn niet/ niet correct ingevuld.
  - 8.2.1 Gebruiker wijzigt de gegevens.
  - 8.2.2 Ga verder bij stap 7.

## 3.2 Full versie

### 3.2.1 Berekenen risico bedrijf per werknemer

#### Goal

De risico's voor het bedrijf per werknemer worden berekent.

#### Rationale

Deze use case is er om het berekenen van de risico's per werknemer mogelijk te maken.

#### Actors

Programma, medewerker.

#### Preconditions

Medewerker moet verbinding hebben met de database, het programma moet draaien.

#### Post Conditions

#### Successful

De risico's zijn berekent en getoond.

#### Unsuccessful

De risico's zijn niet berekent of niet getoond.

#### Flow of Events

##### Basic Flow

1. Medewerker gaat naar het programma.
2. Medewerker klikt op invoer d.m.v. invoervelden.
3. Medewerker klikt op output in plain text file.
4. Programma genereert een form waar de gegevens ingevoerd kunnen worden.
5. Medewerker vult de contact gegevens in.
6. Medewerker vult de gegevens van de eerste werknemer in.
7. Medewerker klikt op voer in.
8. Programma zet de gegevens over in labels en schuift de invoervelden 1 regel naar beneden.
9. Stap 7, 8 en 9 worden herhaalt tot alle werknemers ingevuld zijn.
10. Medewerker klikt op gereed.
11. Programma controleert de gegevens en slaat ze op in de database.
12. Programma berekent de risico's per ingevoerde werknemer en slaat ze op in de database.
13. Programma berekent het gemiddelde risico voor het bedrijf en slaat deze op in de database.
14. Programma zet de risico's in een plain text file en maakt een grafiek van de risico's.

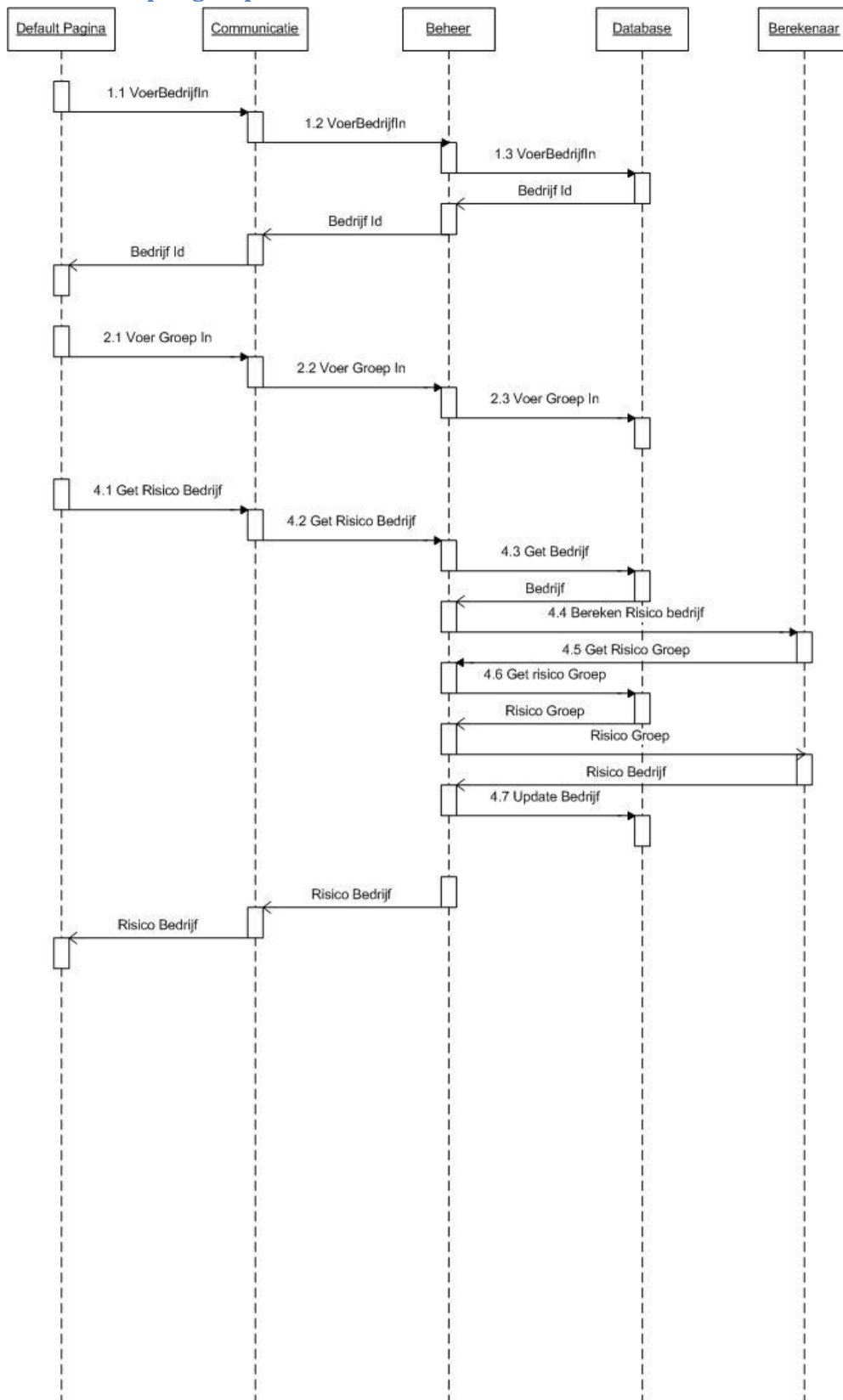
##### Alternate Flow

- 2.1 Medewerker klikt op invoer d.m.v. xml.
  - 2.1.1 Programma genereert een form waar de gegevens ingevoerd kunnen worden.
  - 2.1.2 Medewerker vult de contact gegevens in.
  - 2.1.3 Medewerker vult de locatie van de xml file in.
  - 2.1.4 Ga verder bij stap 11.
- 2.2 Medewerker klikt op invoer d.m.v. database.
  - 2.2.1 Programma genereert een form waar de gegevens ingevoerd kunnen worden.
  - 2.2.2 Medewerker vult het bedrijfsnummer in of andere gegevens voor het bedrijf.
  - 2.2.4 Ga verder bij stap 12.
- 14.1 Medewerker klikt op output in xml file.
  - 14.1.1 Programma zet de risico's in een xml file en maakt een grafiek van de risico's.
- 14.2 Medewerker klikt op output in html file.
  - 14.2.1 Programma zet de risico's in een html file en maakt een grafiek van de risico's.
- 12.1 Contact gegevens zijn niet/ niet correct ingevuld.
  - 12.1.1 Medewerker wijzigt de contact gegevens.
  - 12.1.2 Ga verder bij stap 11.
- 12.2 Gegevens over werknemer zijn niet/ niet correct ingevuld.
  - 12.2.1 Medewerker wijzigt de gegevens.
  - 12.2.2 Ga verder bij stap 11.

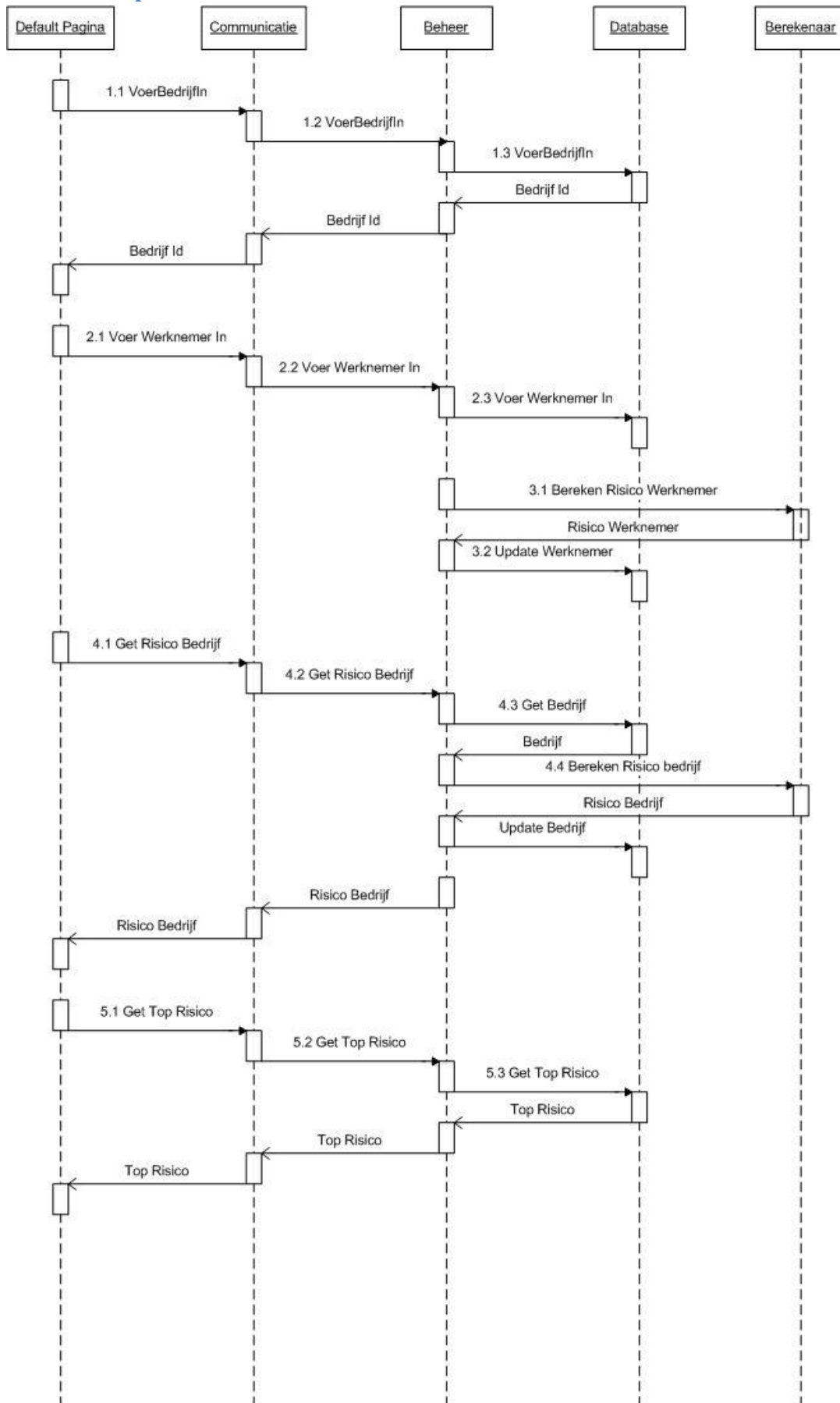
## 4 Sequentie diagrammen

### 4.1 Light versie

#### 4.1.1 Invoer per groep

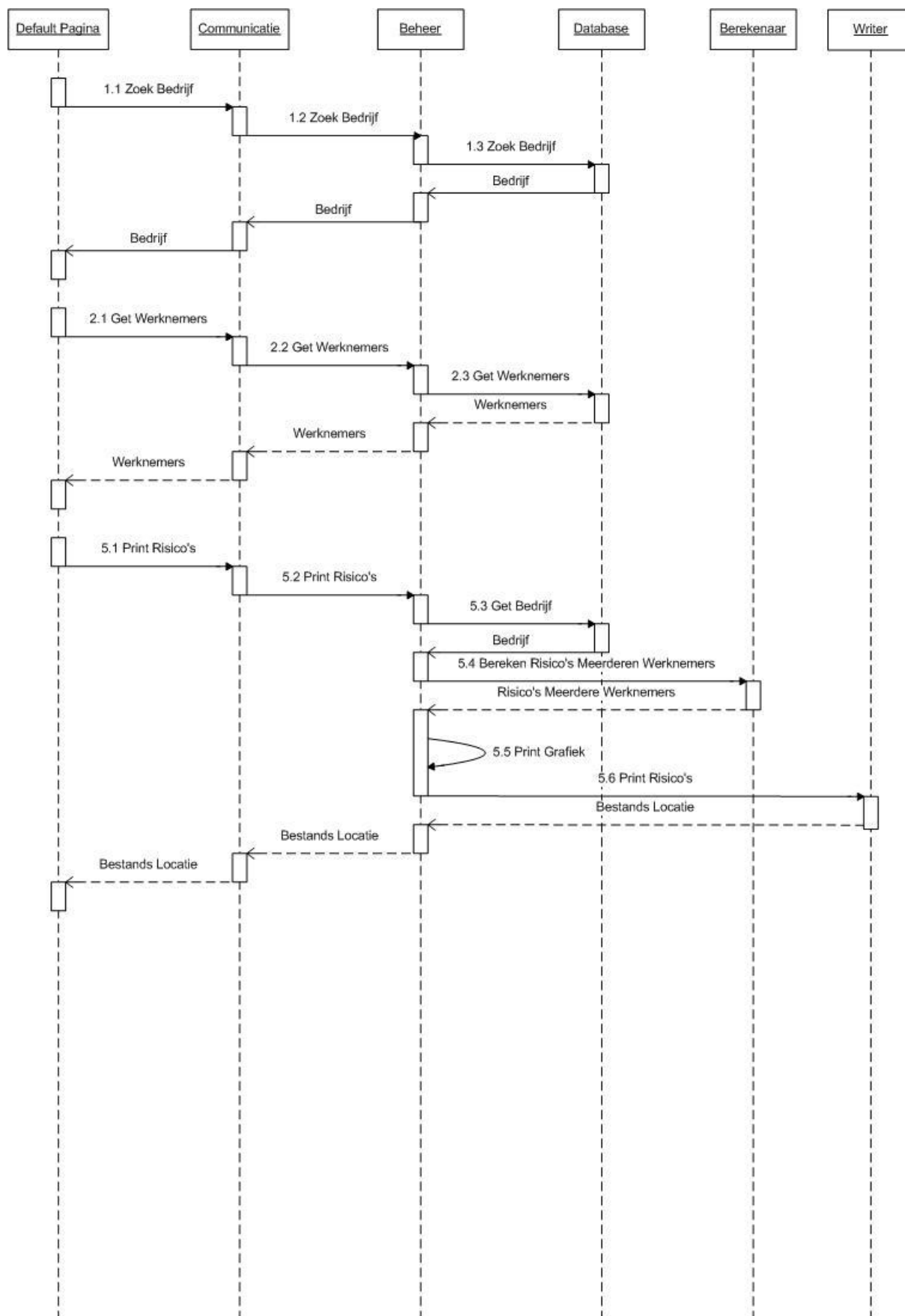


#### 4.1.2 Invoer per werknemer



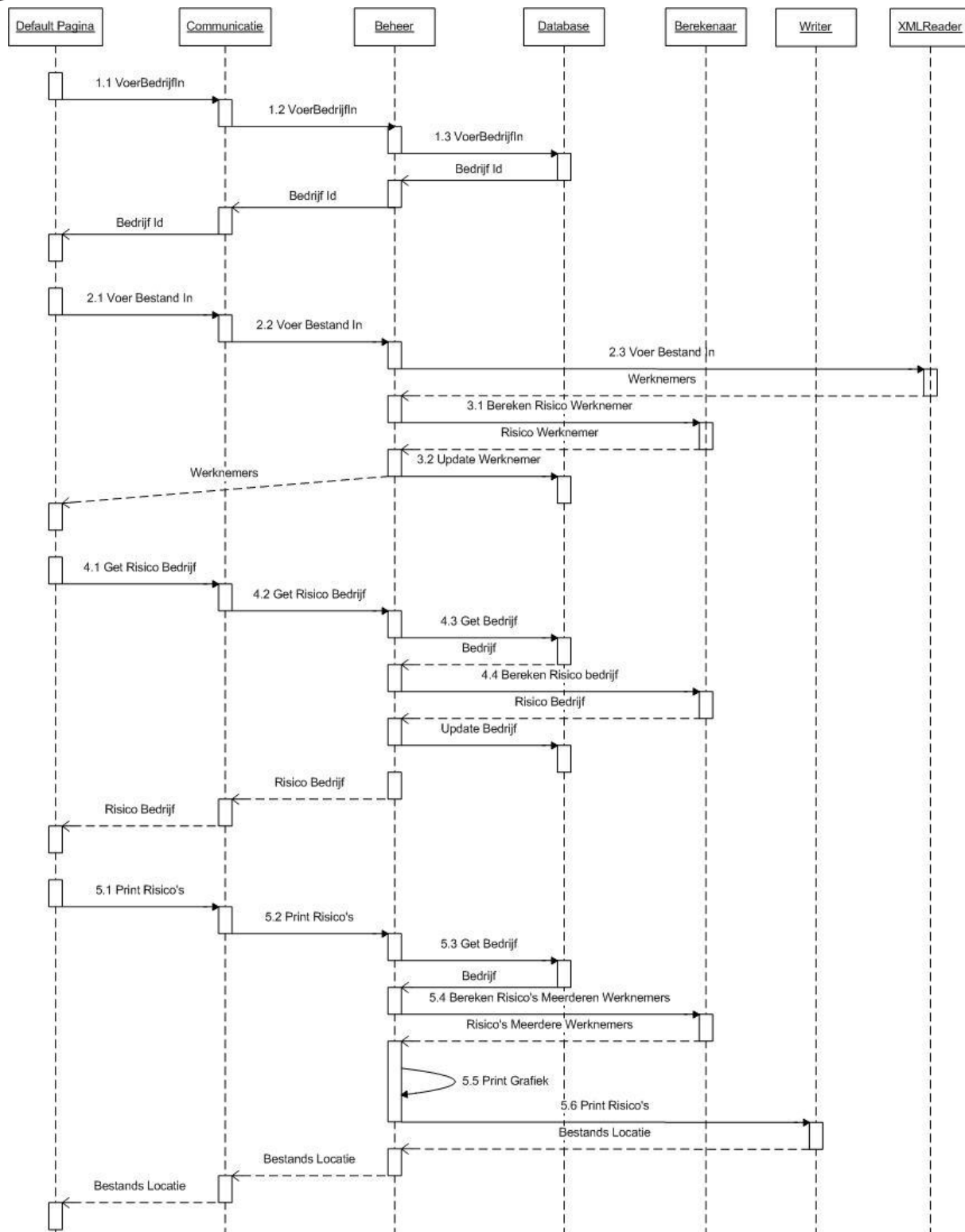
## 4.2 Full versie

### 4.2.1 Invoer via database

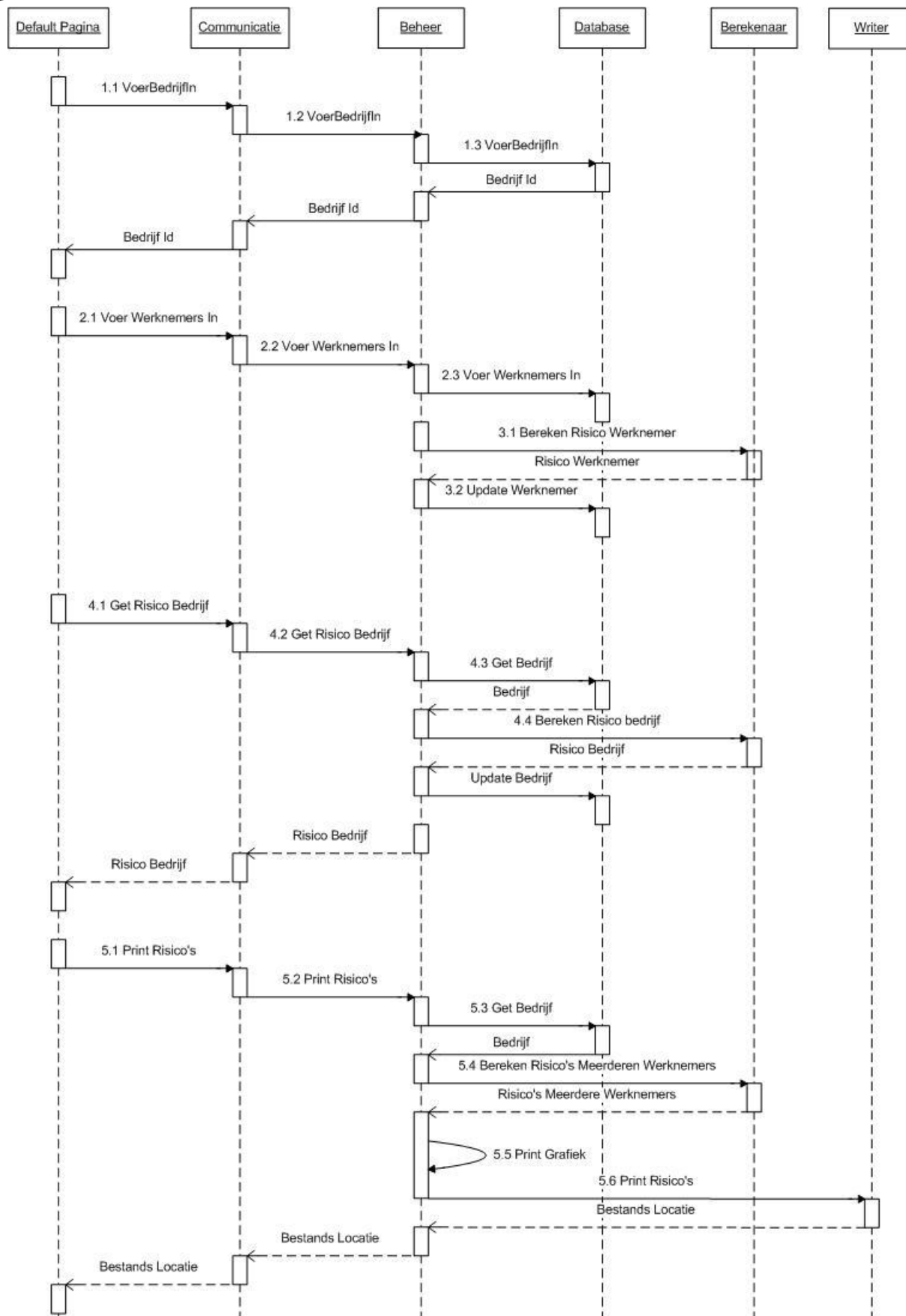




## 4.2.2 Invoer via XML




### 4.2.3 Invoer via invoervelden



## 5 GUI

### 5.1 Light versie


#### 5.1.1 Beheer

**RISICO CALCULATOR**


Home


Help

Beheer

Database Connectie 

Wachtwoord  
  
Toon verzuim ☒
  
Toon oorzaak ☒
  
Toon dodelijk ☒
  
Toon letsel type ☒
  
Toon letsel locatie ☒


#### 5.1.2 Invoer per werknemer

**RISICO CALCULATOR**


Home

Help

Beheer

Database Connectie 

**RISICO CALCULATOR**

Dit is de risico calculator van RoOp.


Bedrijfsnaam  Adres  Plaats  Postcode

Telefoonnummer  Email  Branche

Beroep	Geslacht	Leeftijd	Opleiding	Herkomst	Contract	Werktijden	Wijzig
Machine bankwerkers, monteurs instrumentenmakers, reparateurs	Man	15-29 jaar	Basis/MAVO/VMBO	Autochtoon	Vast	0-24 uur	<input type="button" value="Wijzig"/>
Machine bankwerkers, monteurs instrumentenmakers, reparateurs	Man	15-29 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>
Elektromonteurs, reparateurs van elektrische apparaten	Man	15-29 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>
Elektromonteurs, reparateurs van elektrische apparaten	Vrouw	15-29 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>
Elektromonteurs, reparateurs van elektrische apparaten	Vrouw	30-49 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>
Elektromonteurs, reparateurs van elektrische apparaten	Man	50-64 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>
Elektromonteurs, reparateurs van elektrische apparaten	Man	50-64 jaar	Basis/MAVO/VMBO	Autochtoon	Flexibel	0-24 uur	<input type="button" value="Wijzig"/>

Beroep  Geslacht  Leeftijd 
  
Opleiding  Herkomst  Contract  Werktijden  
  
Als u alle werknemers heeft ingevuld drukt u op de knop "Gereed"

### 5.1.3 Invoer per groep

**RISICO CALCULATOR**


Home
Help
Beheer

Database Connectie ✔

**RISICO CALCULATOR**

Dit is de risico calculator van RoOp.

Bedrijfsnaam

Adres

Plaats

Postcode

Telefoonnummer

Email

Branche

<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>Schilder</p> <p>Loodgieters, fitter, lassers, plaat- en constructie werkers</p> <p>Metselaars, timmerlieden en andere bouwvakkers</p> <p>Elektromonteurs, reparateurs van elektrische apparaten</p> <p>Machine bankwerkers, monteurs instrumentenmakers, reparateurs</p> <p>Overige ambachtelijke en industriële beroepen</p> <p>Overige beroepen</p> <p>HBO/WO</p> <p>Onbekende opleiding</p> <p>Autochtoon</p> <p>Allochtoon</p> <p>Vast</p> <p>Flexibel</p> </div> <div style="width: 45%; border: 1px solid #ccc; padding: 2px;"> <p>3</p><p>2</p><p>3</p><p>1</p><p>2</p><p>0</p><p>2</p><p>0</p><p>0</p><p>11</p><p>2</p><p>2</p><p>11</p> </div> </div>	<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>Man</p> <p>Vrouw</p> <p>15-29 jaar</p> <p>30-49 jaar</p> <p>50-64 jaar</p> <p>Basis/MAVO/VMBO</p> <p>HAVO/VWO/MBO</p> <p>Onbekend contract</p> <p>0-24 uur</p> <p>25-36 uur</p> <p>36+ uur</p> <p>Onbekende werktijden</p> </div> <div style="width: 45%; border: 1px solid #ccc; padding: 2px;"> <p>10</p><p>3</p><p>3</p><p>8</p><p>2</p><p>10</p><p>3</p><p>0</p><p>0</p><p>9</p><p>4</p><p>0</p> </div> </div>
--	--


Als u alle werknemers heeft ingevuld drukt up de knop "Gereed"

## 5.1.4 Resultaten

### RISICO CALCULATOR



Home Help Beheer

Database Connectie 

#### RISICO CALCULATOR

Dit is de risico calculator van RoOp.

Het risico voor uw bedrijf is uitgekomen op Hoog

Het hoogste risico voor een van uw werknemers is Extreem hoog

Er zijn het afgelopen jaar 23 mensen overleden binnen uw branche, dit is 27,058823529% van de totale dodelijke ongelukken.

Oorzaak	Aantal	Procent
Snijden aan object	3900	28,953229399
Geraakt door bewegend object	3100	23,014105419
Beknelling	870	6,4587973274
Stoten tegen stilstaand object	590	4,3801039347
Val van hoogte (excl. trap en ladder)	1100	8,1662954714
Val van trap of ladder	640	4,7512991834
Zwikken	540	4,008908686
Struikelen	180	1,3363028953
Uitglijden	150	1,1135857461
Val door sprong	100	0,7423904974
Vreemd voorwerp in lichaam	1800	13,363028953
Acute fysieke belasting	260	1,9302152932
Verbranding	140	1,0393466964
Chemische inwerking	100	0,7423904974


Verzuim	Aantal/Dagen
1, 2 of 3 dagen	5100
4 dagen tot 2 weken	8000
2 weken tot 1 maand	3700
1 tot 6 maanden	6100
Meer dan 6 maanden	1500
Gemiddeld	44,143442623

Type Letsel	Aantal	Procent
Oppervlakkig letsel en open wond	860	20
Botbreuk/Fractuur	1800	41,860465116
Luxatie en distorsie	80	1,8604651163
Traumatische amputatie	160	3,7209302326
Organletsel en hersenletsel	500	11,627906977
Brandwond en bevriezing	170	3,9534883721
Vergiftiging en infecties	40	0,93023255814
Overig	580	13,488372093
Onbekend	70	1,6279069767

Locatie van Letsel	Aantal	Procent
Hoofd	700	16,279069767
Nek	30	0,6976744186
Rug	160	3,7209302326
Romp en organen	610	14,186046512
Bovenste ledematen	1300	30,23255814
Onderste ledematen	1200	27,906976744
Meerdere lichaamsdelen	130	3,023255814
Overig	120	2,7906976744
Onbekend	20	0,46511627907

## 5.1.5 Foutmelding

# Risico CALCULATOR



Home
Help
Beheer

Database Connectie ✔

### Risico CALCULATOR

Dit is de risico calculator van RoOp.

Bij alle lijsten moet een waarde geselecteerd zijn

Bedrijfsnaam

Adres

Plaats

Postcode

Telefoonnummer

Email

Branche

Voer in

Beroep

Geslacht

Leeftijd

Opleiding

Herkomst

Contract

Werktijden

Voer werknemer in

Als u alle werknemers heeft ingevuld drukt up de knop "Gereed"

## 5.2 Full versie

RisicoCalculator
Database connectie ☐

Dit is de risico calculator van RoOp

Hoe wilt u de gegevens invoeren?

Bestands locatie

Hoe wilt u de output hebben?

Beroep

Geslacht

Leeftijd

Opleiding

Herkomst

Contract

Werktijden

Bedrijfsnummer

Bedrijfsnaam

Adres

Vestigingsplaats

Postcode

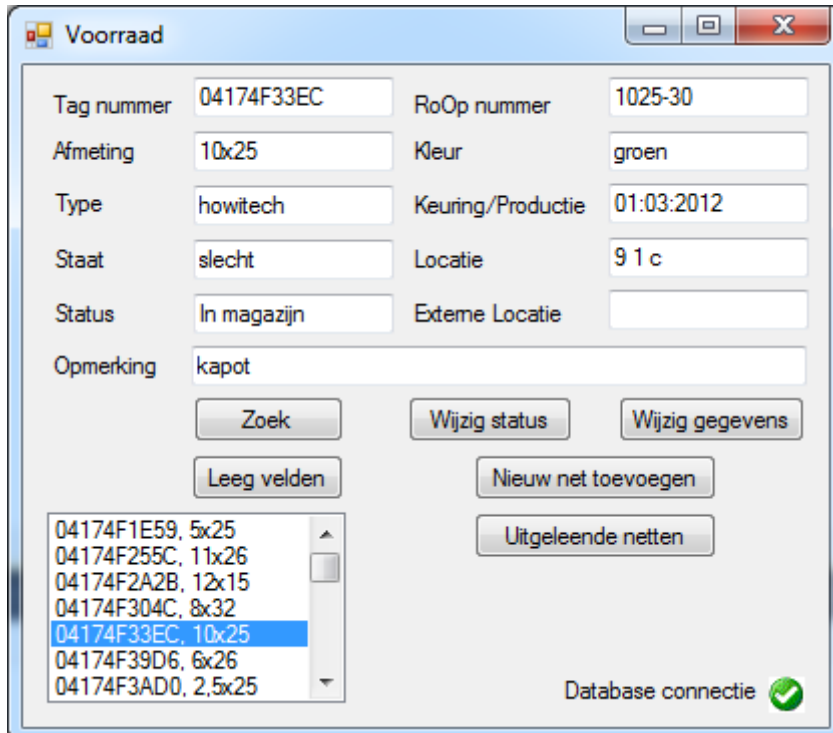
Email

Telefoonnummer

Branche

77

## 5.3 Voorraad



**Voorraad**

Tag nummer: 04174F33EC      RoOp nummer: 1025-30

Afmeting: 10x25      Kleur: groen


Type: howitech      Keuring/Productie: 01:03:2012

Staat: slecht      Locatie: 9 1 c

Status: In magazijn      Externe Locatie:

Opmerking: kapot

Buttons: Zoek, Wijzig status, Wijzig gegevens, Leeg velden, Nieuw net toevoegen, Uitgeleende netten

Database connectie: 

Inventory list (selected item highlighted):

- 04174F1E59, 5x25
- 04174F255C, 11x26
- 04174F2A2B, 12x15
- 04174F304C, 8x32
- 04174F33EC, 10x25**
- 04174F39D6, 6x26
- 04174F3AD0, 2,5x25

## 5.4 Android



3G   2:35 PM

**MainActivity**



Selecteer de activiteit die u gaat uitvoeren

activiteit 2 ▼

Update de database

Update

Dit zijn de punten waar u op moet letten:

**Maatregel 1 Activiteit 2**

**Maatregel 2 Activiteit 2**