**FUNCTIONEEL ONTWERP**

MKB Digiwerkplaats

Luc van Koppen  
Tess Demir  
Riccardo van Dam

Gideon Bakker  
Finley Lamboo

19/02/2020

**Inhoudsopgave**

[1 Samenvatting 2](#_Toc401260718)

[1.1 Samenvatting voor de klant 2](#_Toc401260719)

[1.2 Versiebeheer 2](#_Toc401260720)

[1.3 Verzendlijst 2](#_Toc401260721)

[2 Plan van aanpak 3](#_Toc401260722)

[2.1 Op te leveren producten 3](#_Toc401260723)

[2.2 Planning 3](#_Toc401260724)

[3 Programma van eisen 4](#_Toc401260725)

[3.1 Doelstelling 4](#_Toc401260726)

[3.2 Huidige situatie 4](#_Toc401260727)

[3.3 Reden van verandering/aanpassing 4](#_Toc401260728)

[3.4 Nieuwe situatie 4](#_Toc401260729)

[3.5 Functionele eisen gesteld aan de nieuwe situatie 4](#_Toc401260730)

[3.5 Systeemeisen gesteld aan de nieuwe situatie 6](#_Toc401260731)

[4 Interface 7](#_Toc401260732)

[4.1 Interface 7](#_Toc401260733)

[4.2 Schema’s 7](#_Toc401260734)

[5 Use Case en Datamodel 8](#_Toc401260735)

[5.1.1 De Use Case 8](#_Toc401260736)

[5.1.2 Verklaring van de Use Case 8](#_Toc401260737)

[5.2.1 De database 8](#_Toc401260738)

[5.2.2 Het ER Model 8](#_Toc401260739)

[5.2.3 Inhoud van de database 8](#_Toc401260740)

[5.2.4 Beheer en beveiliging van de database 8](#_Toc401260741)

[6 Slotconclusie 9](#_Toc401260742)

[6.1 Verantwoording 9](#_Toc401260743)

[6.2 Bronvermelding 9](#_Toc401260744)

## 1 Samenvatting

### 1.1 Samenvatting voor de klant

Dit document omvat de rapportage over de fase functioneel ontwerp van MKB Digiwerkplaats. Alle hoofdonderdelen evenals conclusies en aanbevelingen zijn verwerkt in dit totaaloverzicht,

met eventuele verwijzingen naar separate uitgebreide documenten, die gedurende deze fase

zijn opgesteld.

### 1.2 Versiebeheer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Versie** | **Status** | **Datum** | **Auteur** | **Aanpassingen** |
| **1.0** | **AF** | **19/02/2020** | Luc, Tess, Riccardo, Finley, Gideon | **Functioneel ontwerp gemaakt** |
| **X.X** |  |  |  |  |
| **X.X** |  |  |  |  |
| **X.X** |  |  |  |  |

### 1.3 Verzendlijst

Dit document wordt ter beschikking gesteld aan:

Henry Robben – Docent

Tessa – Project Owner

Luc van Koppen – Developer

Riccardo van Dam – Developer

Tess Demir – Developer

Finley Lamboo – Developer

Gideon Bakker - Developer

## 2 Plan van aanpak

### 2.1 Op te leveren producten

Wat leveren wij op als developers: Functioneel Ontwerp, Informatiebehoefte en de webapplicatie.

### 2.2 Planning

Functioneel Ontwerp en Informatiebehoefte zal 19/02/2020 af zijn.  
De datum waar de webapplicatie zal worden gemaakt is nog onduidelijk aangezien er nog geen interview is geweest op de tijd van het schrijven van dit document.

## 3 Programma van eisen

### 3.1 Doelstelling

De doelstellingen van het functioneel ontwerp wordt hier omschreven. Deze worden opgesteld in samenspraak met de opdrachtgever Samengevat geef je hier aan wat je gaat ontwikkelen.

Het doel van deze webapplicatie is dat bedrijven die de werkplaats hebben gebruikt feedback kunnen geven en om dit zo makkelijk mogelijk te laten gebeuren.

### 3.2 Huidige situatie

Op het moment is er een webapplicatie waar een aantal functies werken maar een aantal ook nog niet. De styling is op veel plekken goed maar er zijn een aantal bugs die verbeterd moeten worden.

### 3.3 Reden van verandering/aanpassing

Er ontbreken een aantal onderdelen van de website dat de klant graag wil hebben. Hierbij gaat het om onder andere de admin panel waar rollen kan worden gegeven, beheerders de studenten kan koppelen aan klanten en de antwoorden in kan zien van de klanten. Daarbij wil de klant ook graag een completere vragen lijst met meer vragen, keuze uit meerdere soorten vragen ect.

### 3.4 Nieuwe situatie

Zoals hierboven genoemd zal de nieuwe situatie het volgende uit komen te zien.

De bezoeker van de website kan onder andere informatie opzoeken over de organisatie, inloggen op de website en een vragenlijst invullen. Ook als de bezoeker geen account heeft, kan hij/zij alsnog een vragenlijst invullen en de contact gegevens achterlaten.

Voor de beheerders komt er een admin panel. Op de admin panel moet er onder andere komen dat er rollen kan worden gegeven aan bijvoorbeeld docenten en studenten. En dat de taken vanuit daar worden onderverdeeld. Daarbij kan er ook een overzicht gevonden worden met de antwoorden van de bezoekers en door wie. Hierdoor kan MKBdigiwerkplaats op een eenvoudige manier contact maken met de desbetreffende.

### 3.5 Functionele eisen gesteld aan de nieuwe situatie

* Admin Panel
  + Alle ingevoerde antwoorden bekijken van de vragenlijst
  + Bekijken welke studenten zijn gekoppeld aan projecten
  + Lijst van alle studenten en beheerders
  + Rollen management
  + Contact gegevens van de accounts
  + Data van studenten zien
* Vragenlijst
  + Open vragen kunnen maken
  + Multiple choice vragen kunnen maken
  + Contact gegevens moeten invullen als je NIET bent ingelogd

### 

### 3.5 Systeemeisen gesteld aan de nieuwe situatie

De applicatie zou worden gehost op een medium sized server. De database wordt apart gedraaid op een MySQL Server.

Medium sized wordt onder gezegd dat er rekening wordt gehouden met redelijke grote data opslag en bezoekers. Maar niet dat de website zo veel bezocht en de dataopslag zo groot is dat er een dure premium server nodig is. En ook niet dat er weinig wordt bezocht met weinig opslag dat er een goedkope versie van de server moet komen waardoor dit hinder kwa performance kan zijn.

## 

## 4 Interface

### 4.1 Interface

Home pagina:

Vanaf hier kan je basisinformatie krijgen over de applicatie en een aantal videos kijken over MKB Digiwerkplaats. Ook kan je vanaf de homepagina aanmelden en zien met wie MKB Digiwerkplaats in samenwerking is.

Over pagina:

Op deze pagina vindt u alle informatie over de MKB Digiwerkplaats.

Contact pagina:

Op deze pagina vindt u de contact gegevens van MKB Digiwerkplaats en kan je direct contact opnemen met MKB Digiwerkplaats ook staat de locatie geïntegreerd met google maps.

Inlog pagina:

Op deze pagina kan je inloggen als je een account hebt aangemaakt. Als je nog geen account hebt kan je op registreren klikken om een account te maken.

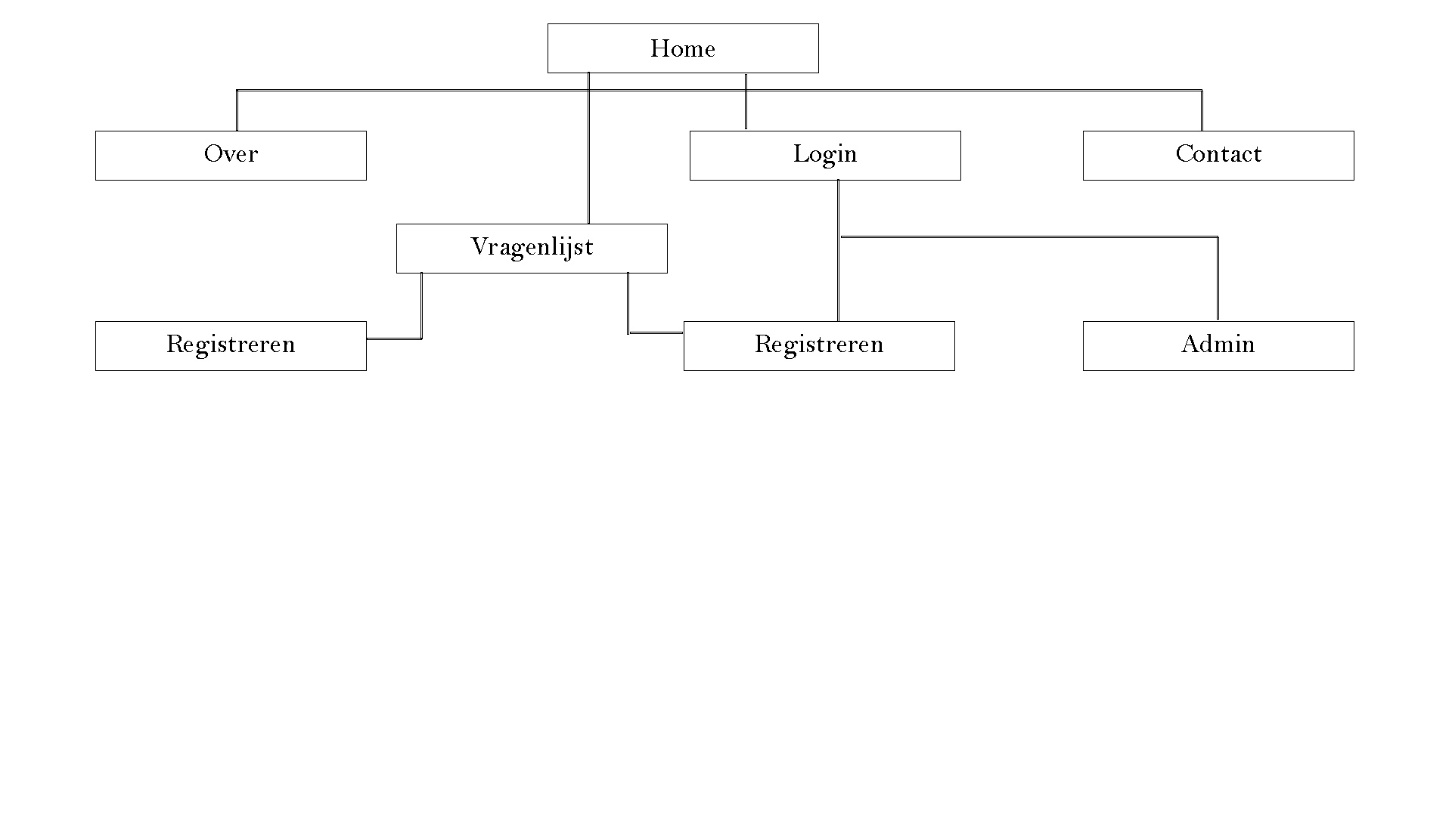
Registreren pagina:

Op deze pagina kan je een account maken.

Admin pagina:

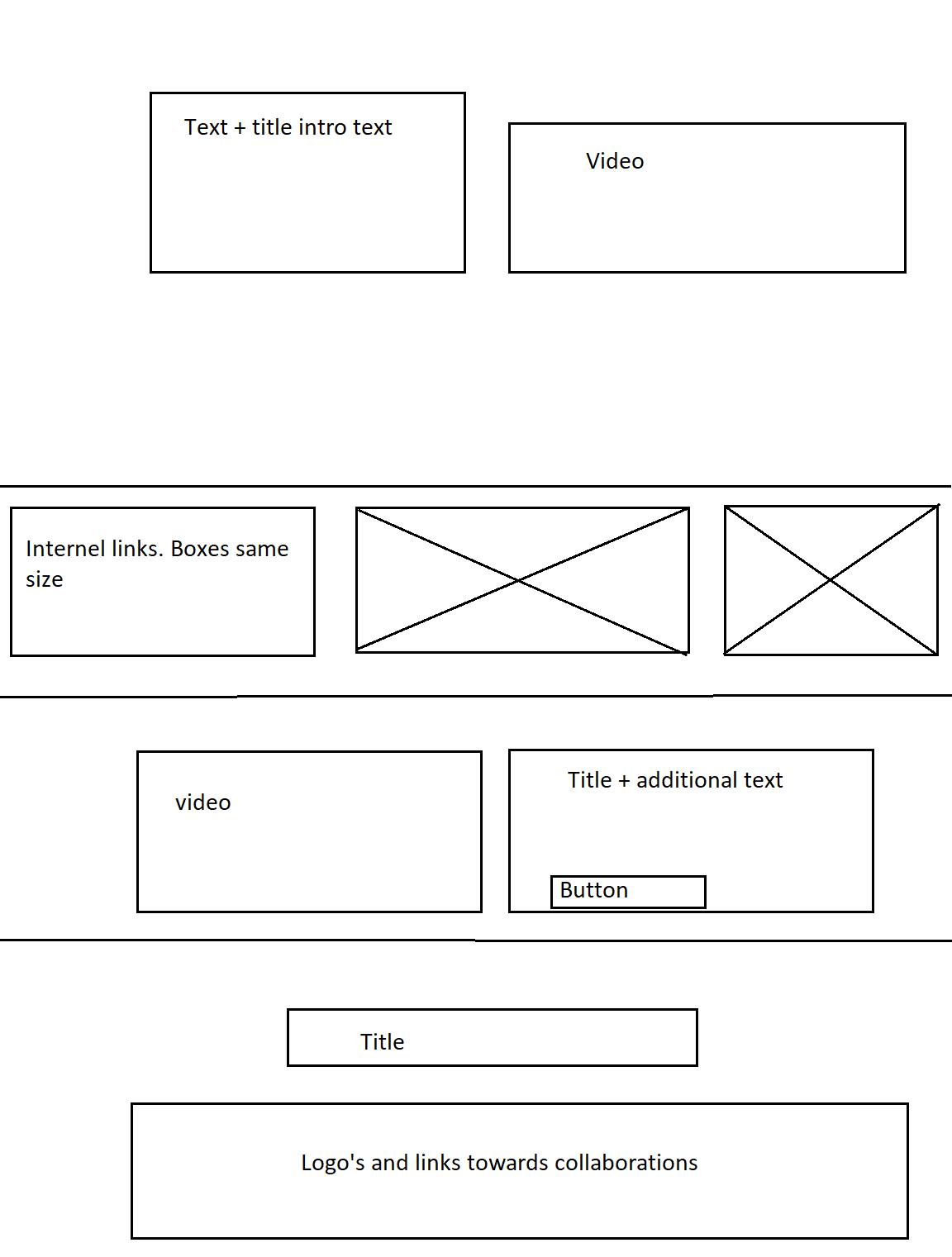
Hier kunnen alleen admins komen met de juiste inlog gegevens. Op deze pagina vindt u alle data van de ingevulde vragenlijsten. Ook kan je zien welke studenten zijn gekoppeld aan een project. Je kan een lijst van alle studenten en beheerders zien. Je kan rollen geven aan bepaalde gebruikers. Alle contact gegevens van de accounts staan hier ook.

### 4.2 Schema’s

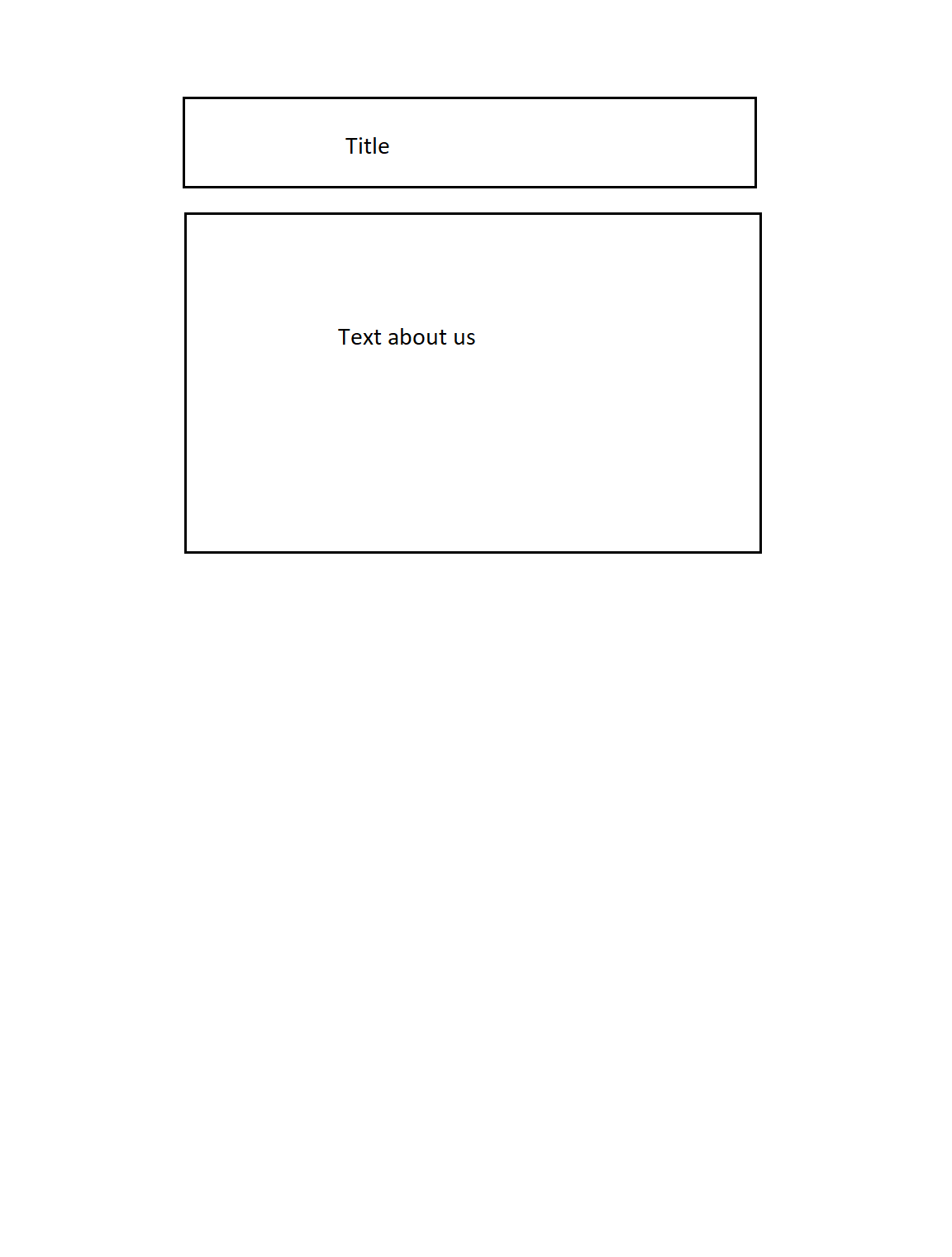


### 4.3 Wireframes

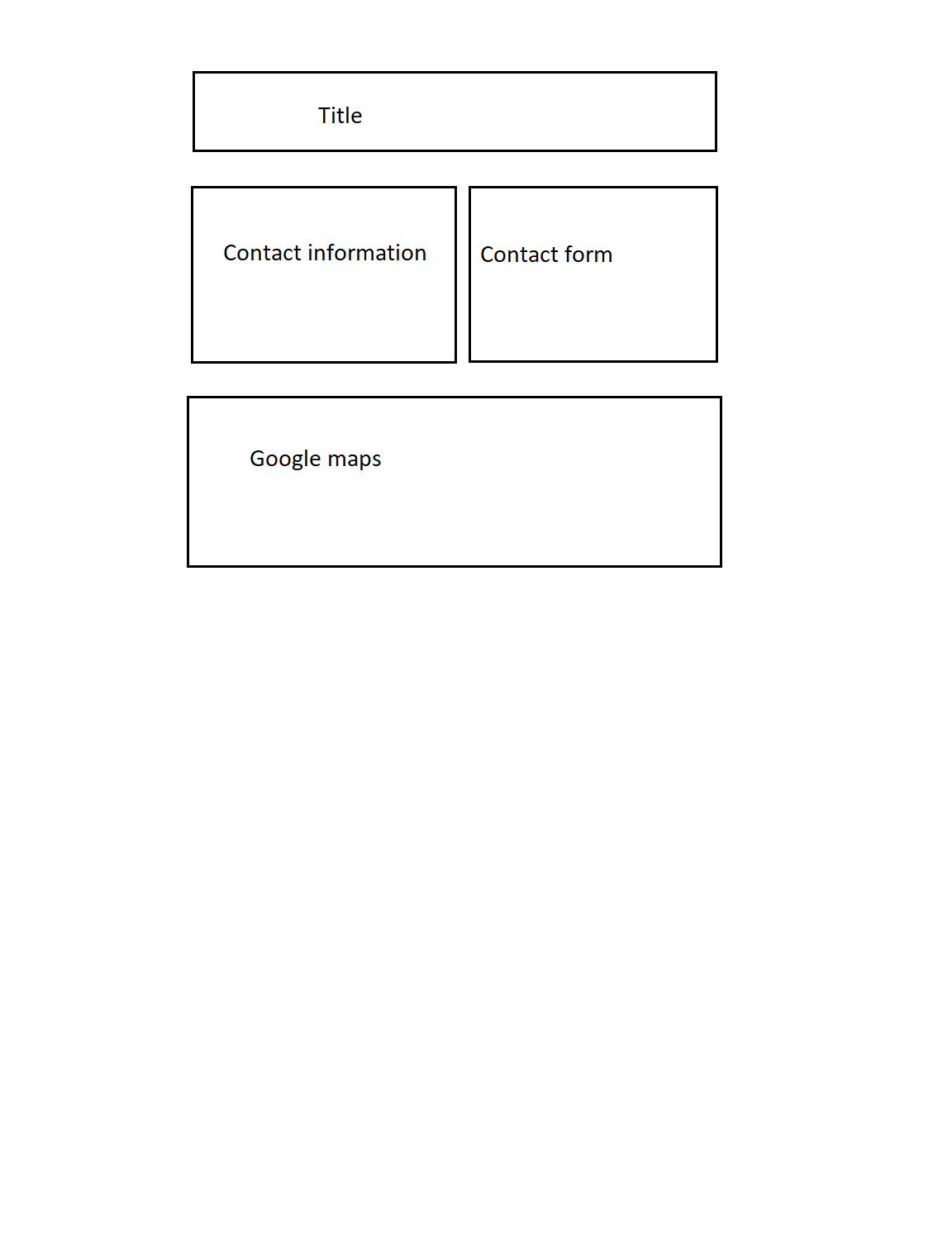
Home pagina:



Over pagina:

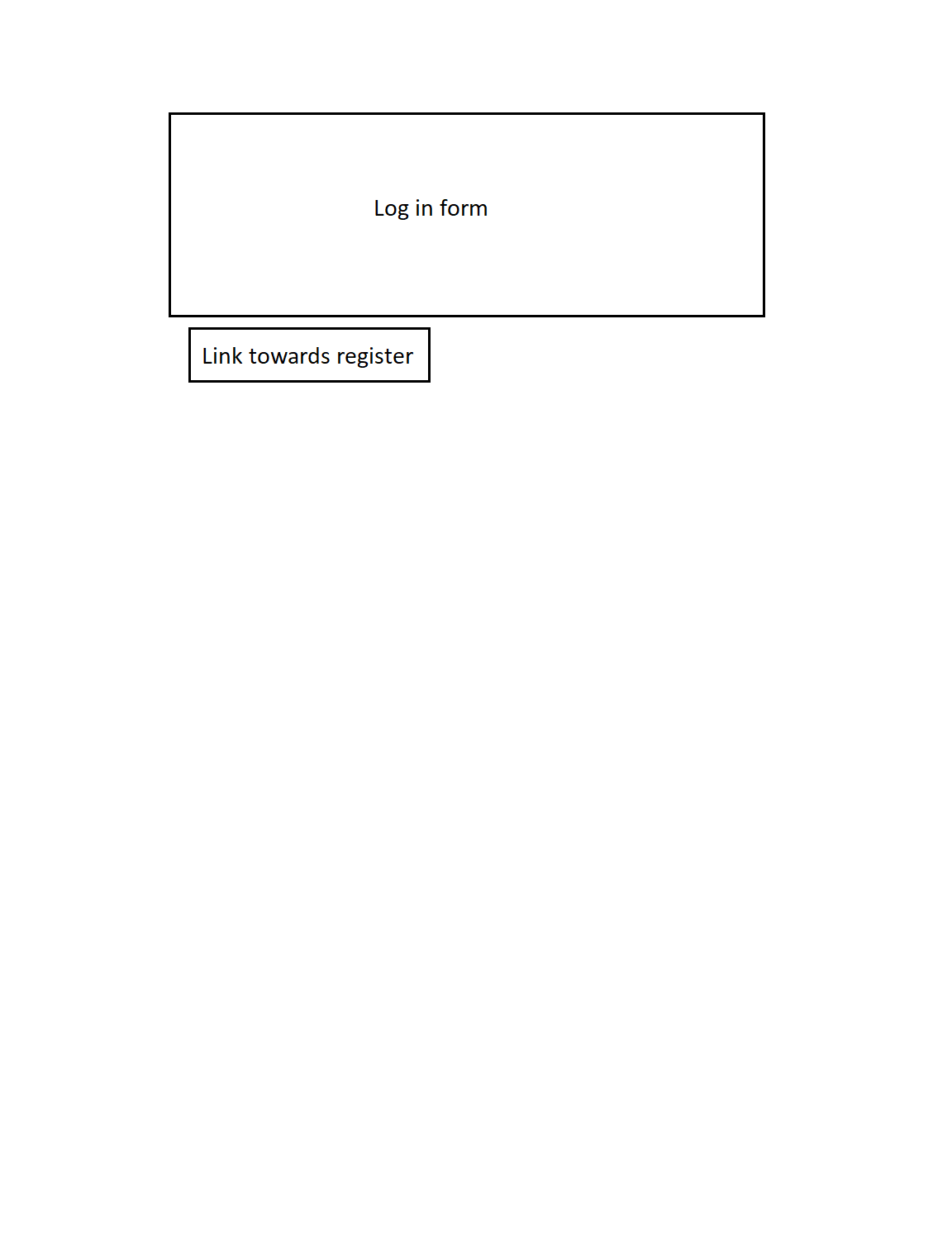


Contact pagina:

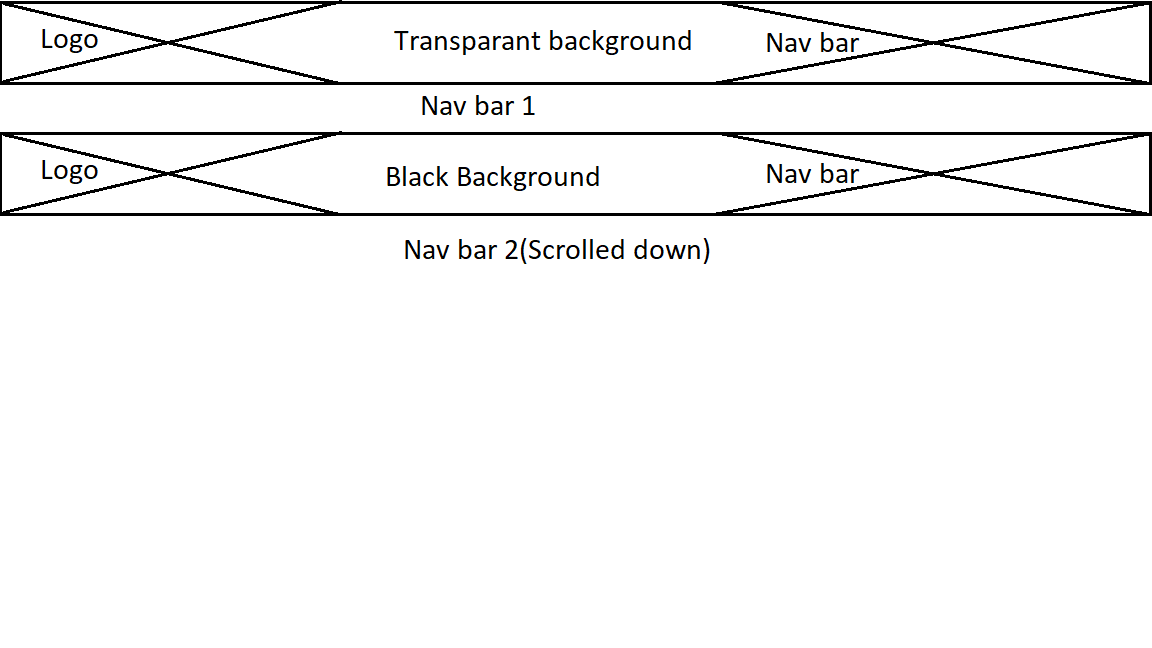
Footer:



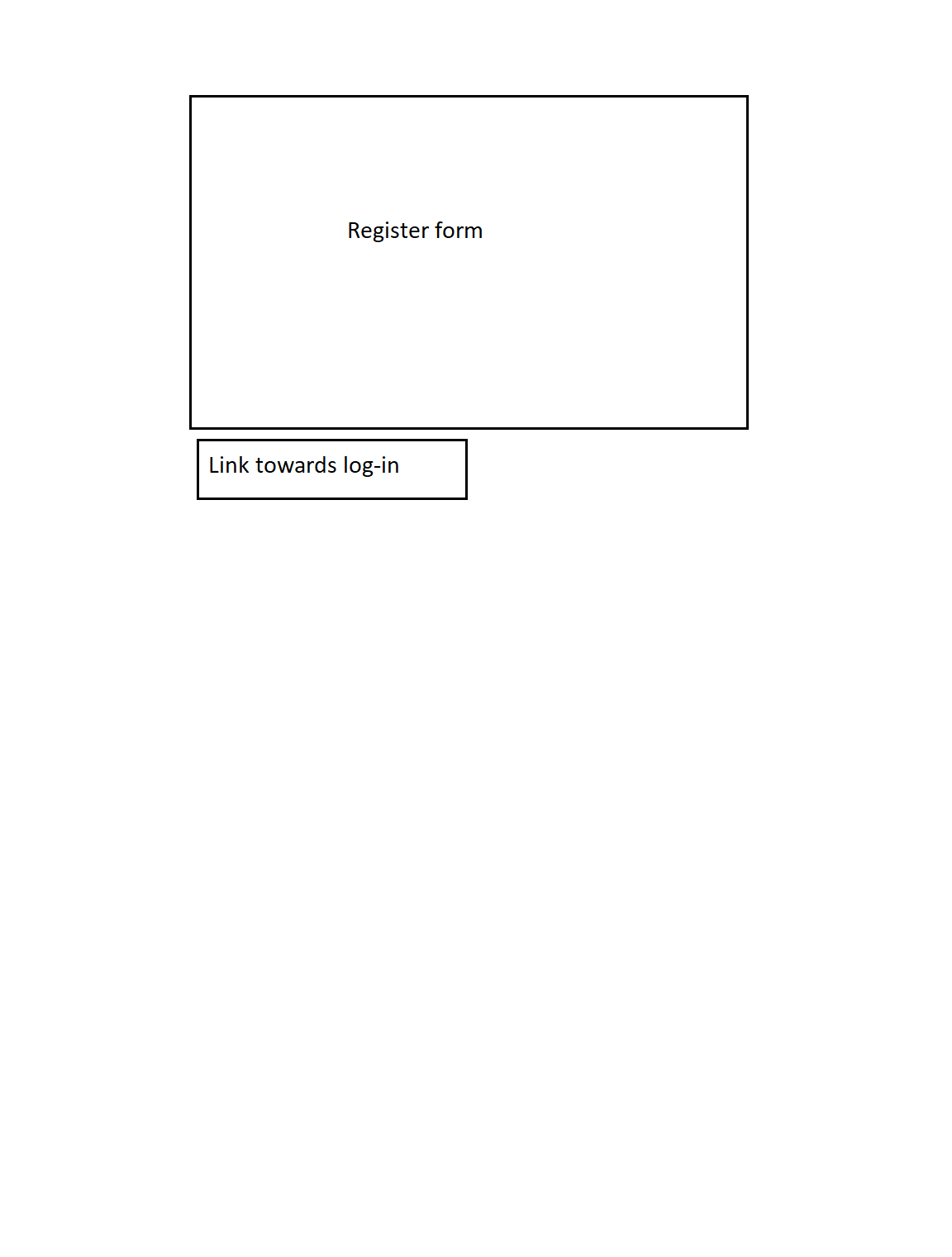
Log-in pagina:



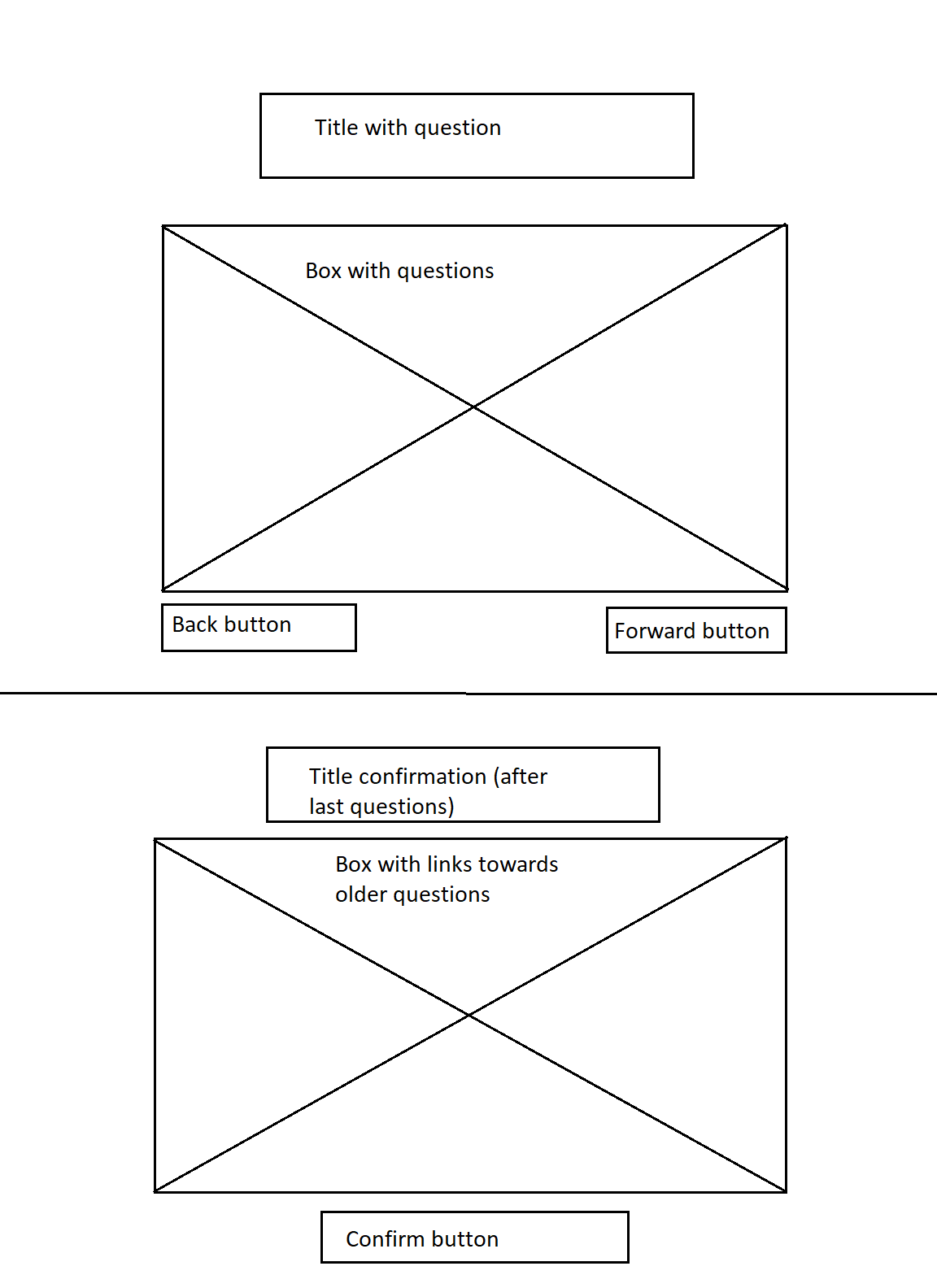
Nav-bar mockup:



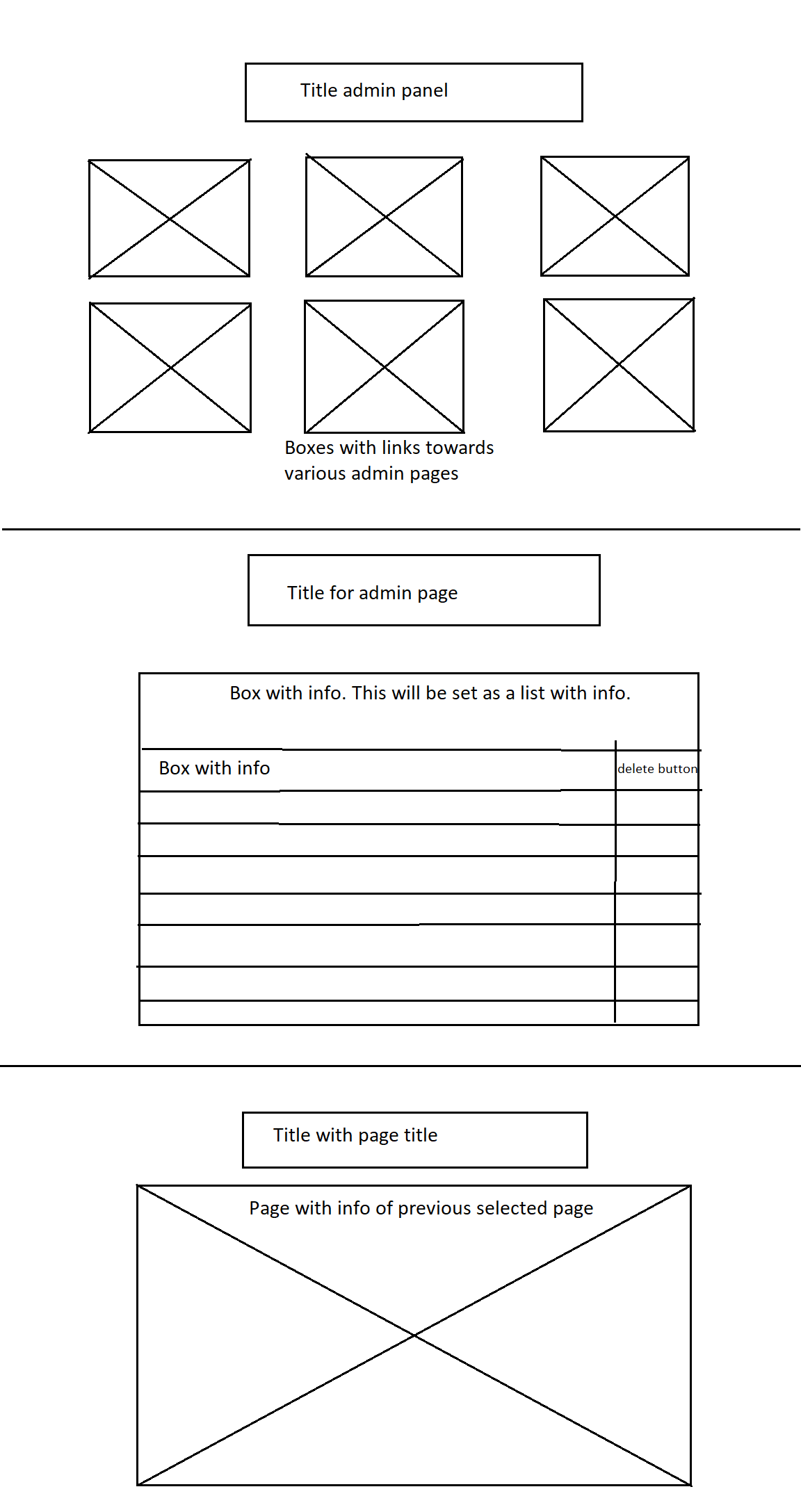
Register pagina:



Vragenlijst pagina:



Admin panel:

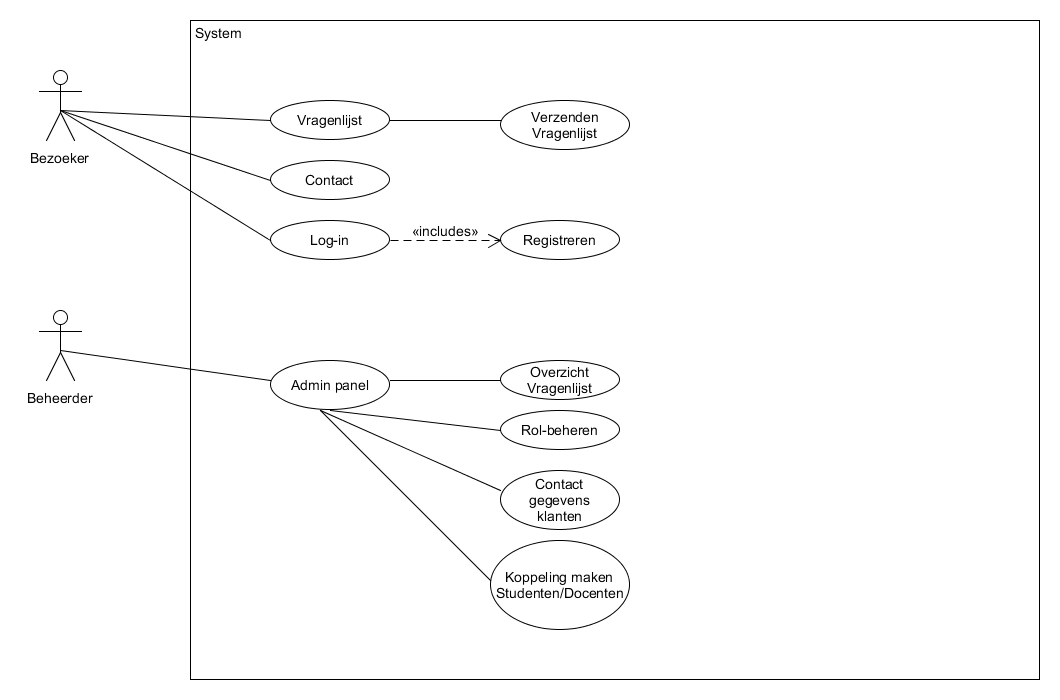


## **5 USE Case en Datamodel**

### 5.1.1 De USE Case

De use case beschrijft "wie" met het betreffende systeem "wat" kan doen

Een use case beschrijft een systeem vanuit het gebruikersperspectief. Het beschrijft de actor, de initiator van de interactie, en het systeem zelf als een opeenvolging van eenvoudige stappen. Hier zullen we onder andere zien hoe een bezoeker door de website heen gaat. Hierop kunt u dus zien wat voor pagina’s interactie vereisen en hoe de gebruiker door de website heen moet gaan.



### 5.1.2 Verklaring van de Use Case

Als eerste gaan we dus kijken bij de gebruiker. Deze kunnen we naar 3 onderdelen volgen. Als eerste log-in. Een gebruiker kan account aanmaken of inloggen. Als een gebruiker geen account heeft, kan de gebruiker naar de registreer pagina.

Als de gebruiker de vragenlijst wil invullen kan de bezoeker naar de vragenlijst pagina. Hier kan de bezoeker de hele vragenlijst doorlopen en beantwoorden. Als de bezoeker klaar is kan de bezoeker de lijst verzenden en word het opgeslagen in de database.

Dan gaan we naar de beheerder. Hier hebben we een aantal taken dat kan worden gedaan alleen door de beheerder en dat is op de admin panel. Op de admin panel heeft de beheerder mogelijkheden om onder andere vanuit daar de rollen beheering op zich te nemen en het overzicht van de vragenlijst op te vragen waar alle antwoorden van de bezoekers staan. Daaruit kunnen ze onder andere ook het weer koppelen aan studenten.

### 5.2.1 De database

De database is een complex systeem. Hier wordt alle data en gegevens van bezoekers opgeslagen en er moet goed worden mee omgegaan. Daarbij moet er ook worden bedacht hoe wij de juiste informatie krijgen uit de database op de juiste plek. Hier zal worden beschreven wat er nodig is, Hoe de database eruit komt te zien en hoe wij zorgen dat kwaadwilligen niet zomaar bij de data komen.

### 5.2.2 Het ER Model

### 5.2.3 Inhoud van de database

De database is zeer groot en bestaat uit 10 tabellen.

### 5.2.4 Beheer en beveiliging van de database

Het beheer van de database zal door ons worden gedaan. Dit betekent dat de data, verwijdering van de data door de beheerders worden gedaan. Dit kan zelf allemaal via de admin panel worden gedaan. Daarnaast als beveiliging zal er een fos user bundel worden gebruikt. Deze bundel dat bij symfony wordt gebruikt heeft een ingebouwde beveiliging en hierdoor zullen de user wachtwoorden standaard beveiligd zijn met een basis beveiliging.

## 6 Slotconclusie

### 6.1 Verantwoording

* Problemen die we hadden met dit document schrijven was vooral te weinig kennis hebben over de site aangezien er nog geen interview is geweest met de klant en we nog niet de volledige versie hadden van de website.
* We hebben het probleem aangepakt door veel te overleggen met elkaar en veel te kijken naar de website die al live staat zodat we genoeg informatie konden opdoen om het te maken

### 6.2 Bronvermelding

De bronnen gebruikt voor de informatie van dit document zijn: de live site, de site op de github en Henry Robben voor extra informatie.