|  |
| --- |
|  |
| Paperless Office |
|  |
| Cloud Applications |

September 30, 2016

Opgesteld door: Levi Gillis, Hannelore Peeters & Sverre Oste

Paperless Office

Cloud Applications

# Project Omschrijving

## Probleemstelling

De klant wil een manier om de grote stapel facturen en documenten die hij ontvangt, te verkleinen. Het aantal zorgt ervoor dat er een onoverzichtelijke papierberg ontstaat waarin, wanneer nodig, de juiste documenten niet snel gevonden kunnen worden. Ook neemt het klasseren van deze documenten veel plaats in en kan er niet altijd eenvoudig gesorteerd worden. Het kwijtspelen van een belangrijk document gebeurt ook te pas en te onpas en zou verholpen moeten worden. Wanneer er een document door meerdere personen geraadpleegd moet worden, wordt er veel tijd verloren met het zoeken en doorgeven of kopiëren van het desbetreffende document.

## Doel van het project

Het doel van het project is het maken van een digitaal platform waar de gebruiker facturen en andere documenten naar kan uploaden. Hierdoor zal er efficiënter gewerkt kunnen worden, aangezien er geen tijd verloren gaat aan het fysiek zoeken van een bepaald document. De gebruiker zal een account kunnen maken, zodat zijn bestanden enkel voor hem beschikbaar zijn. De bestanden moeten online beschikbaar zijn en moeten gesorteerd kunnen worden. Het systeem moet ervoor zorgen dat een geüpload bestand tags krijgt aan de hand van de inhoud van het document, maar de gebruiker moet ook zelf tags kunnen meegeven. De gebruiker zal de documenten dan kunnen filteren op allerhande waarden, zoals de inhoud, de datum, de naam, etc. Het digitale platform zal de vorm hebben van een website en er zal gebruik gemaakt worden van een database om de documenten bij te houden. De gebruiker zal documenten kunnen uploaden nadat hij ze gescand heeft, maar ook via een smartphone zullen bestanden geüpload kunnen worden. Er zal een extra app ontwikkeld worden die het mogelijk maakt om een foto te maken met een smartphone en die te uploaden.

## Methodologie

Dit project zal ontwikkeld worden met de scrum methode, hiervoor maken we gebruik van GitHub met de ZenHub plug-in. Dit houdt in dat er gewerkt wordt in korte sprints van ongeveer twee weken. Elke sprint zullen er enkele user stories onderhanden genomen worden, deze user stories zullen opgedeeld zijn in een aantal kleinere issues. De issues worden opgesteld na de analyse van de desbetreffende user story. De bedoeling is om minstens twee user stories per sprint af te werken. Wanneer dit niet lukt, of misschien zelfs sneller dan normaal lukt, verplaatsen we issues om ze respectievelijk later of eerder te doen. We willen tijdens de sprints ook telkens kort even overlopen, met elkaar en de klant, of we op het goede spoor zitten.

Telkens er een sprint afgewerkt is, willen we een werkende demo hebben. Deze kunnen we dan aan de klant laten zien, zodat die tijdig feedback kan geven en er eventueel snel aanpassingen doorgevoerd kunnen worden.

## Verwachte resultaten

Het uiteindelijke resultaat zal een digitaal platform zijn in de vorm van een website. De website zal toegankelijk zijn via pc en smartphone. Er wordt ook een database gemaakt. De bedoeling is om te werken met verschillende datapakketten, omdat er maar een beperkt aantal documenten geüpload kan worden per maand.

Op de homepagina van de website zal de gebruiker algemene informatie over het product kunnen vinden. Ook zal er de mogelijkheid zijn om zich te registreren, met een gratis datapakket of een betalend. Het voordeel van een betalend account te nemen is uiteraard een grotere hoeveelheid beschikbare data, dus meer uploads per maand.

Wanneer de gebruiker ingelogd is zal hij bestanden kunnen uploaden. De API’s en cognitive services die we gebruiken, zullen automatisch enkele tags genereren. De gebruiker zal zelf een titel en extra tags kunnen toevoegen aan het bestand. De bestanden die geüpload kunnen worden zijn PDF bestanden en foto’s die gemaakt zijn met een smartphone.

De gebruiker zal elk geüpload bestand individueel kunnen raadplegen. Er zal extra informatie over het documenten te zien zijn, zoals bijvoorbeeld een deel van de inhoud zonder het te moeten openen. Er zullen ook suggesties gegeven worden die gebaseerd zijn op overeenkomstige tags (van de gebruiker zelf of van het systeem) en titels. Wanneer de gebruiker dan het verkeerde document heeft, kan hij snel het juiste vinden.

# Technologie

## Microsoft Cognitive Services – Computer Vision API + OCR

We gaan gebruik maken van Microsoft Cognitive Services om de inhoud van de geüploade bestanden om te zetten in tekst. Deze tekst wordt gebruikt om de content tags te genereren.

## Bootstrap

Voor de front-end op de site gaan we bootstrap gebruiken om de lay-out van de site te ontwerpen.

## Angular.js

Om ervoor te zorgen dat de juiste data weergeven wordt, wordt Angular.js gebruikt. Dit zorgt ook voor de client requests.

## Node.js backend

De backend server wordt met node.js gemaakt hiermee kan gemakkelijk een API call beantwoord worden (bv. inloggen). Dit zorgt ervoor dat de server met javascript geprogrammeerd kan worden.

## Xamarin

De mobile app wordt gemaakt met behulp van xamarin dit zorgt ervoor dat de app in C# kan geschreven worden en gemakkelijk naar andere platformen geport kan worden.

## Cloudconvert API

Deze API gaan we gebruiken om de geüploade PDF bestanden om te zetten naar JPG bestanden. Dit omdat de cognitive services waar we gebruik van maken enkel werken op JPG bestanden.

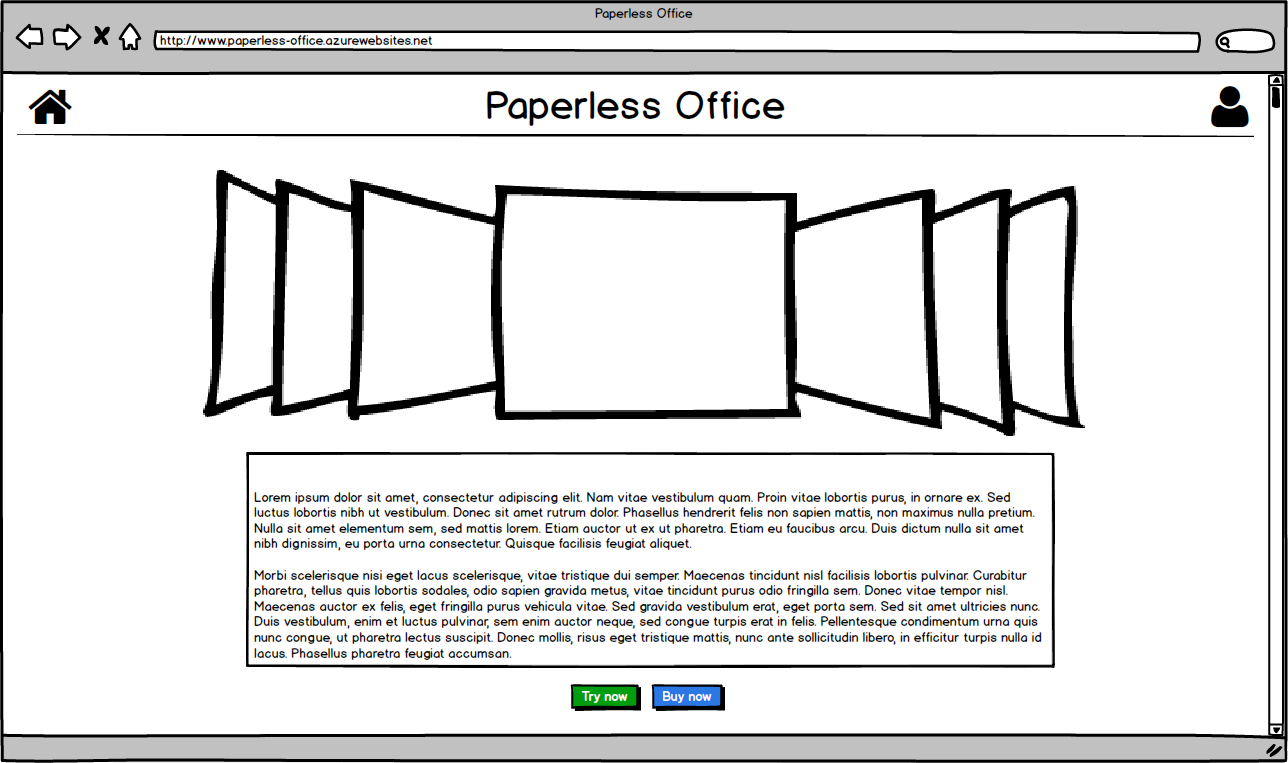
# Uitkomst voor de maatschappij

Het gebruik van ons digitale platform zal in vele kantoren het papierverbruik sterk verminderen en er zal geen ruimte vrijgemaakt moeten worden om al de documenten te klasseren. Ook zal er efficiënter gewerkt kunnen worden, omdat er minder tijd verspild wordt met het zoeken naar documenten. De interne communicatie van het kantoor zal ook vaker al digitaal gebeuren, omdat de documenten dan niet meer ingescand moeten worden en sneller geüpload kunnen worden. Documenten zullen ook sneller gedeeld kunnen worden.

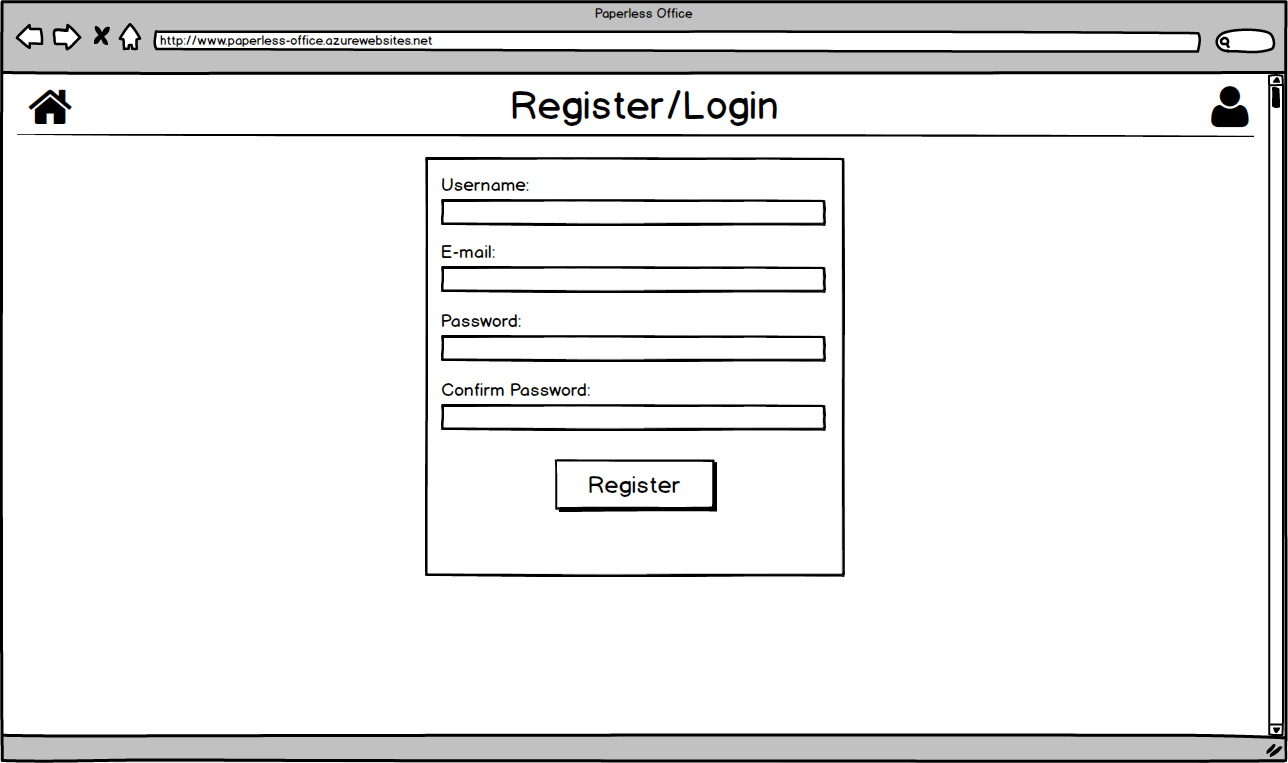
# User stories / Actoren

|  |
| --- |
| Als gebruiker wil ik een homepagina om informatie over het product te vinden. |
| Als gebruiker wil ik kunnen inloggen. |
| Als gebruiker wil ik bestanden kunnen uploaden. |
| Als gebruiker wil ik tags aan mijn geüploade bestanden kunnen toevoegen. |
| Als gebruiker wil ik de geuploade bestanden kunnen filteren en sorteren op inhoud, datum, tags en naam. |
| Als gebruiker wil ik een optie om de beschikbare data uit te breiden. |
| Als app gebruiker wil ik kunnen inloggen. |
| Als app gebruiker wil ik een foto kunnen nemen met gsm en uploaden. |
| Als gebruiker wil ik mijn documenten online kunnen bekijken. |
| Als gebruiker wil ik elk document apart kunnen raadplegen en downloaden. |
| Als gebruiker wil ik suggesties krijgen bij het geopende document. |
| Als gebruiker wil ik mijn dataverbruik kunnen checken. |
| Als systeem wil ik PDF-bestanden kunnen omzetten naar afbeeldingen. |
| Als systeem wil ik thumbnails kunnen maken van afbeeldingen. |
| Als systeem wil ik controleren of de gebruiker nog beschikbare opslag heeft. |
| Als systeem wil ik tekst kunnen herkennen in afbeeldingen. |
| Als systeem wil ik content tags kunnen genereren uit tekst. |

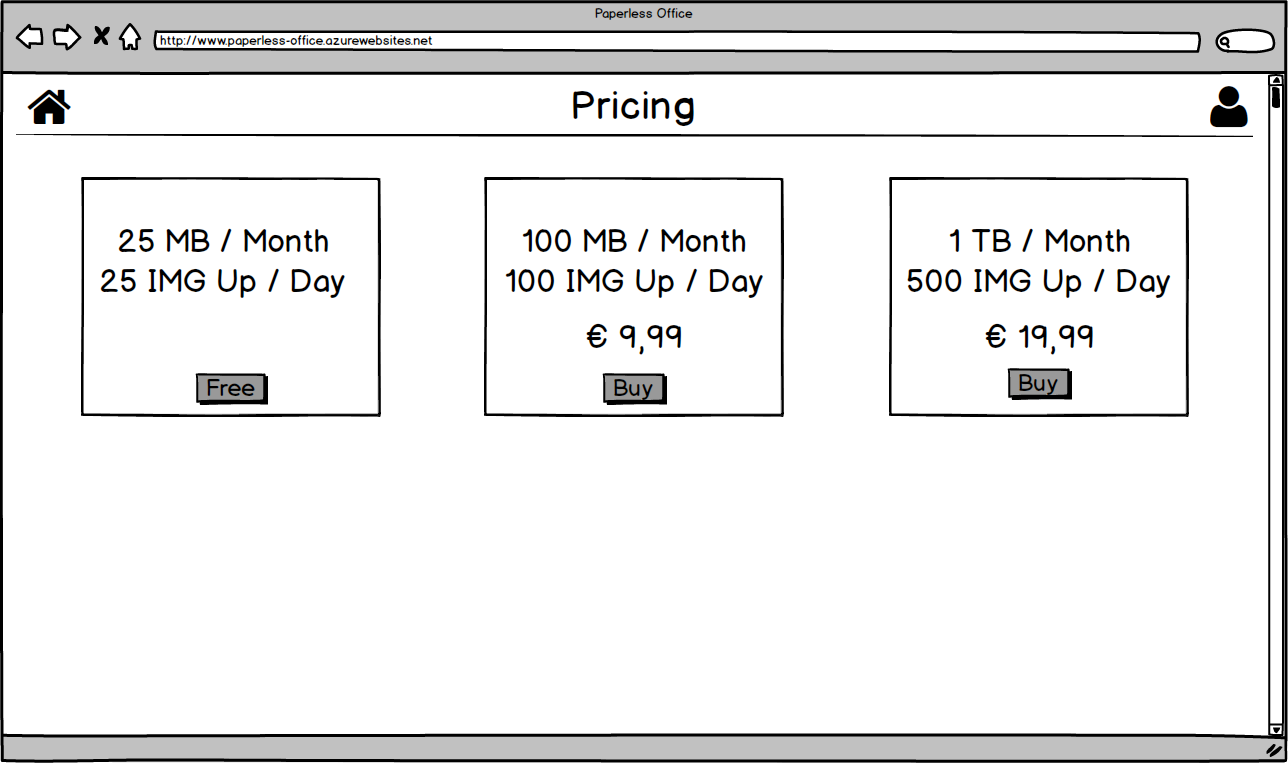
# Mockups



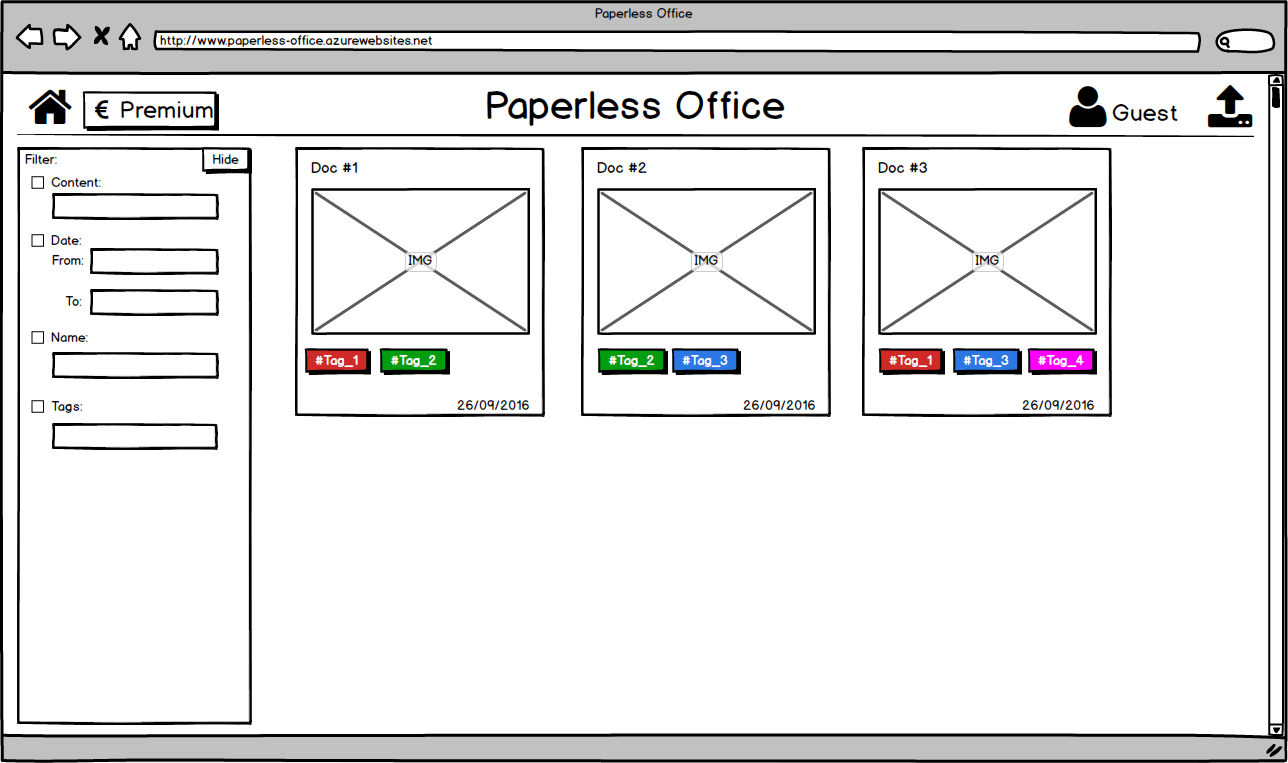
Figuur Homepagina mockup



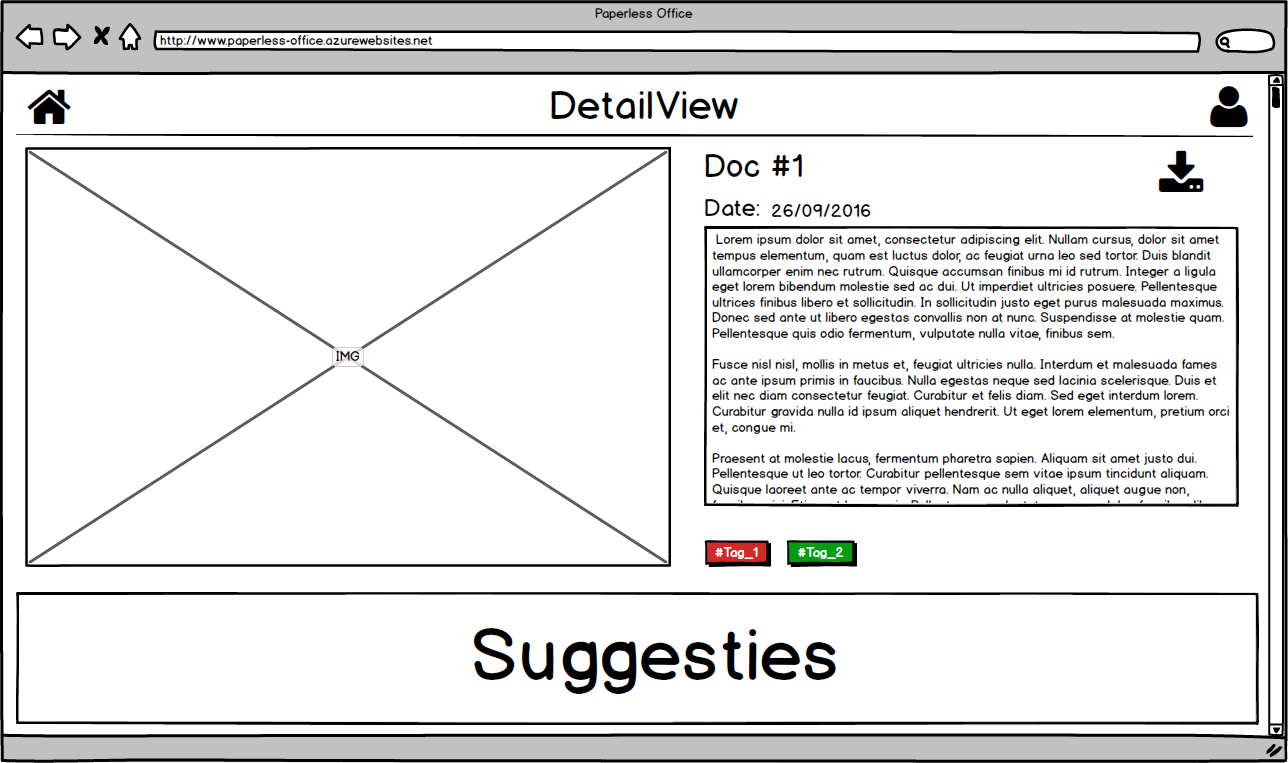
Figuur Register mockup



Figuur Pricing mockup



Figuur UI mockup



Figuur Detailview mockup

# Links

* *Github*

<https://github.com/Levi--G/CA-Paperless-Office-LHS.git>

* *Github projects*

<https://github.com/Levi--G/CA-Paperless-Office-LHS/projects#boards?repos=69014963>