Analyse document

Simplified School Portal

Tobias Sagis

S22

9-4-2018

Inhoud

[Productomschrijving 3](#_Toc507584460)

[Wireframe 3](#_Toc507584461)

[Functionele requirements 4](#_Toc507584462)

[Must 4](#_Toc507584463)

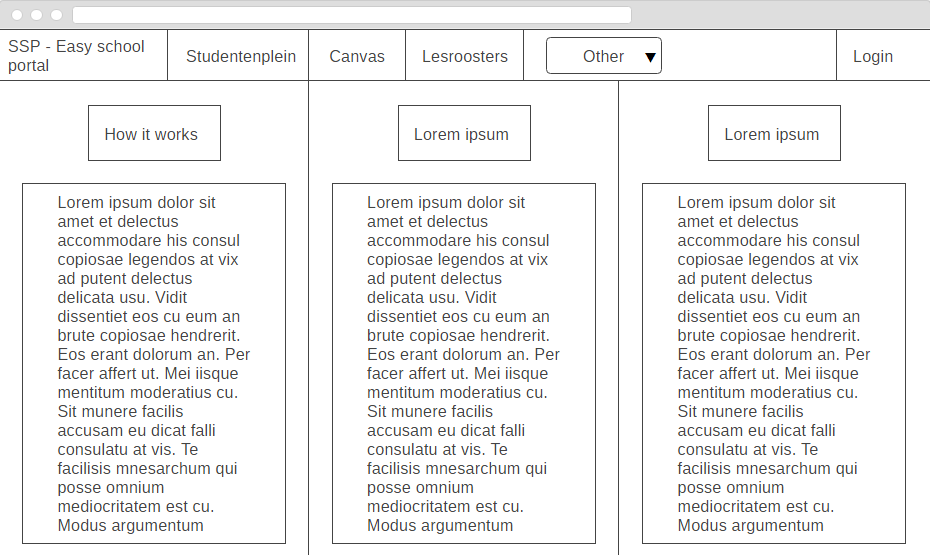
[Should 4](#_Toc507584464)

# Productomschrijving

Momenteel bestaat de digitale omgeving van school uit een groot aantal links, waarbij iedere link een aparte login-handeling vereist. Om bij informatie te komen zal de gebruiker dus 2 tot 3 maal moeten inloggen in een andere service. Simplified School Portal neemt de vorm van een ASP.net pagina aan, en werkt via het SSO (Single Sign-on) principe. Je logt een keer in via de ASP.net applicatie, waarna je informatie van vrijwel elke service (lesroosters, canvas, webmail) snel kan weergeven. Het doel van de applicatie is om de bestaande portal anders vorm te geven en authenticatie tussen services te vereenvoudigen, zodat SSO mogelijk wordt.

## Wireframe

<https://wireframe.cc/4v88XC>



De voorpagina van het product bevat weinig tot geen functionaliteiten. Zodra de gebruiker heeft ingelogd worden de navigatiebar-links actief gesteld. De gebruiker kan op dat moment gebruik maken van een overzichtelijk studentenplein, canvas courses, lesroosters en andere functionaliteiten zonder dat er een extra inlog is vereist.

# Functionele requirements

Aan de hand van de MoSCoW methode worden enkele requirements van de applicatie gecategoriseerd op basis van prioriteit.

## Must

* Na het inloggen verkrijgt de gebruiker toegang tot alle aangeboden services, zoals Canvas, SharePoint en de webmail, zonder dat hij/zij nogmaals moet inloggen.
* De functionaliteiten van de aangeboden services zijn identiek aan de originele services; De gebruiker mag geen functionaliteiten missen (zoals ontbrekende canvas courses) die wel te verkrijgen zijn door op de normale manier in te loggen.
* De gebruiker krijgt voldoende ondersteuning (informatie) over de werking van het systeem en de navigatie binnen het systeem. Het design van de UI helpt hierbij, door het overzichtelijk te maken.
* De gebruiker krijgt de mogelijkheid om contact op te nemen met de beheerder via een webformulier.

## Should

* Links die niet in werking zijn gesteld, ofwel weinig worden gebruikt/voor een langere tijd niet zijn geüpdatet, worden uit de representatie van het studentenplein gehaald.
* Het overzicht binnen Canvas wordt aangepast, zodat de gebruiker de mogelijkheid heeft om het overzicht van courses te personaliseren; Courses kunnen verwijderd worden of er kan gefilterd worden op actuele courses.
* De beheerder moet meerdere services kunnen toevoegen aan de bestaande portal. In dit proces kan de beheerder ook de pagina van de service vormgeven.
* De beheerder kan een uitdraai vragen van alle contactmomenten tussen beheerder-gebruiker, en alle aanvragen van nieuwe services (API requests).
* De beheerder kan nieuwe services toevoegen doormiddel van een grafische inrichting van de pagina. De beheerder krijgt een overzicht met informatie en kan deze op locaties in de webpagina plaatsen, en deze vervolgens genereren.

## Could

* Via de ASP.net pagina kunnen push-notificaties worden weergeven wanneer de gebruiker een mail ontvangt in zijn outlook postvak.