

ClipIt Reader

G. Mulder & L. van Dijk

16-5-2012

Inhoudsopgave

1 Inleiding	1
1.1 Lijst met gebruikte afkortingen en termen	1
2 Requirements	2
2.1 Functional requirements	2
2.2 Non-functional requirements	2
2.3 Use-cases	3
3 Ontwerp	5
3.1 Gebruikers	5
3.2 Componenten	5
3.3 Interface	6
3.4 Gegevens	6
4 Planning	6

1 Inleiding

Veel moderne bedrijven willen graag weten wat er in digitale media over hen gezegd wordt. ClipIt is een bedrijf dat in deze behoefte voorziet door nieuwsberichten te verzamelen en te verwerken. Dit verwerken gebeurt iets versimpeld als volgt: Een klant meldt zich eerst aan bij ClipIt voor een bepaald zogenaamd analysepakket bij de marketing-afdeling. Op basis van dit analysepakket mag een klant dan een zeker aantal zoektermen doorgeven aan de operations-afdeling, waarna operations de zoektermen koppelt aan het profiel van de klant. Vervolgens worden alle binnenkomende nieuwsberichten getoetst aan deze zoektermen, waarna de gevonden berichten dan weer in een portal te bekijken zijn door de klant.

Hoewel deze werkwijze op zich prima is, hebben meerdere klanten al aangegeven dat de portal onhandig is in het gebruik op smartphones en tablets. Het doel van ons project is dan ook het ontwikkelen van een applicatie waarmee klanten makkelijk voor hen interessante berichten uit hun ClipIt-profiel kunnen bekijken op een smartphone, terwijl toch zo veel mogelijk informatie uit de portal behouden wordt.

1.1 Lijst met gebruikte afkortingen en termen

- API: Applications Program Interface
- App: Applicatie, en specifiek, de ClipIt reader.
- ClipIt: ClipIt BV, een online media monitoring bedrijf dat opereert vanuit het mercatorgebouw. zie <http://www.clipit.nl> voor meer informatie over het bedrijf.
- Reader: De ClipIt Reader-applicatie.

- Sentiment: De 'toon' van een nieuwsbericht, positief ("goed product!"), neutraal ("pas gekocht") of negatief ("wat een rommel!")

2 Requirements

In dit hoofdstuk bespreken wij de requirements waaraan onze applicatie zal moeten voldoen. Omdat wij nog niet hebben overlegd met alle stakeholders binnen ClipIt is de kans groot dat de requirements nog deels zullen veranderen.

2.1 Functional requirements

Hieronder staan per component de functionele requirements van de applicatie zoals ze nu staan.

System

- De applicatie kan informatie opvragen van de servers van ClipIt.
- Communicatie tussen de applicatie en de ClipIt-servers verloopt met behulp van een API.
- De applicatie kan nieuwsberichten sorteren en tonen op volgorde van datum, sentiment en mediawaarde.
- De applicatie kan onderscheid maken tussen klanten met- en zonder lopend abonnement bij ClipIt.

End-users

- Gebruikers van de app moeten via de applicatie kunnen inloggen op hun ClipIt-account.
- Via de reader kunnen klanten nieuwsberichten die aan hun profiel gekoppeld zijn lezen.
- Gebruikers kunnen in een opties-menu instellen welke nieuwselectie ze willen lezen.
- In de applicatie moeten gebruikers door kunnen klikken naar de volledige versie van een artikel.
- Gebruikers moeten de artikelen kunnen analyseren, als zij hier rechten toe hebben.
- Via de applicatie moeten gebruikers op telefonische- of andere wijze contact op kunnen nemen met de operations-afdeling van ClipIt.

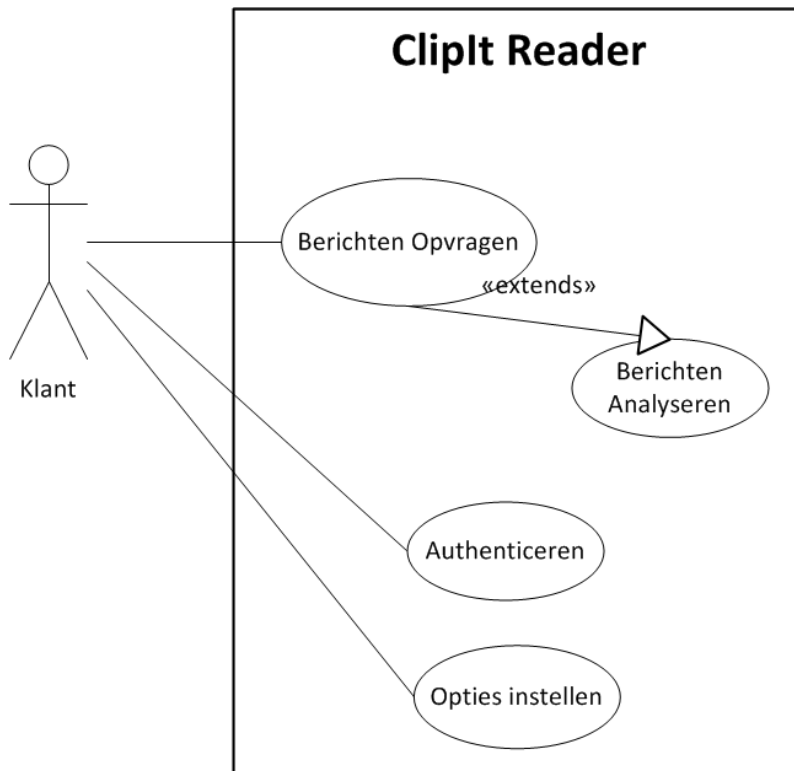
2.2 Non-functional requirements

System

- In eerste instantie zal de applicatie gebruikers geen geld kosten.
- De applicatie moet schaalbaar zijn voor verschillende displays.
- De applicatie moet veilig zijn: klanten moeten alleen op hun eigen profiel in kunnen loggen.
- De applicatie moet te onderhouden zijn door de development-afdeling van ClipIt.
- Nieuwsberichten moeten binnen vijf minuten te lezen zijn in de applicatie.

2.3 Use-cases

Integrated use-case diagram



Specifieke use-cases

Usecase 01: Authenticeren

Use case diagram:	Authenticeren
Description:	Een gebruiker authenticereert zich via de applicatie bij de ClipIt-server.
Preconditions:	De gebruiker is nog niet ingelogd in zijn of haar profiel.
Postconditions:	De gebruiker is ingelogd in zijn of haar profiel.
Actors:	Klant, Medewerker
BCoE:	<ol style="list-style-type: none">1. De applicatie vraagt om de credentials van de gebruiker.2. De gebruiker vult zijn of haar credentials in en signaleert in te willen loggen.3. De applicatie zendt de gegevens naar de server.4. De server zendt een session ID en geeft aan dat de login correct is.5. De applicatie laat het profiel van de gebruiker zien en geeft aan dat de gebruiker is ingelogd.
Alternative Paths:	<p>Er is een verkeerd wachtwoord ingevoerd.</p> <ol style="list-style-type: none">4a. De server geeft aan dat de login mislukt is omdat er een verkeerd wachtwoord is ingevuld.1a. De deels ingevulde credentials worden weergegeven, en de klant krijgt te zien welke credentials foutief zijn.
Exceptions:	<p>Gebruiker heeft geen profiel bij ClipIt</p> <ol style="list-style-type: none">4a. De server zendt terug dat de login mislukt is omdat het profiel niet bestaat.6. De gebruiker sluit de applicatie.

Usecase 02: Berichten lezen

Use case diagram:	Berichten lezen
Description:	Een gebruiker leest een overzicht van gevonden nieuwsberichten.
Preconditions:	De gebruiker is ingelogd in zijn of haar profiel.
Postconditions:	.
Actors:	Klant
BCoE:	<ol style="list-style-type: none">1. De klant navigeert naar het 'nieuws'-component.2. De applicatie zendt een verzoek naar de server om nieuws op te halen, samen met de credentials van de gebruiker.3. De server zendt een serie nieuwsberichten terug.4. De applicatie toont een overzicht van deze nieuwsberichten.5. De gebruiker sluit het nieuws-component.
Alternative Paths:	<ol style="list-style-type: none">5a. De klant selecteert een nieuwsbericht om te analyseren.6. De applicatie initialiseert de use-case "berichten analyseren" met het gekozen nieuwsbericht.7. Het nieuws-component wordt gesloten.
Exceptions:	<ol style="list-style-type: none">3a. De server reageert niet.8. De applicatie geeft een foutmelding.9. Het nieuws-component wordt gesloten met een foutmelding.

Usecase 03: Berichten Analyseren

Use case diagram:	Berichten Analyseren
Description:	Een gebruiker analyseert een bericht uit zijn of haar nieuwsselectie.
Preconditions:	De gebruiker is ingelogd in zijn of haar profiel, en heeft een bericht geselecteerd.
Postconditions:	Het geanalyseerde bericht is als geanalyseerd aangemerkt.
Actors:	Klant
BCoE:	<ol style="list-style-type: none">1. De applicatie zendt een verzoek naar de server om meer informatie over het gekozen bericht.2. De server stuurt alle toepasselijke informatie terug.3. De applicatie toont het analysemenu voor het gekozen bericht aan de gebruiker4. De gebruiker sluit het analysemenu.
Alternative Paths:	<ol style="list-style-type: none">4a. De klant klikt op een optie van het analysemenu.3. De applicatie toont het analysemenu voor het gekozen bericht aan de gebruiker
Exceptions:	<ol style="list-style-type: none">2a. De server stuurt geen informatie terug.5. De applicatie geeft een foutmelding.6. Het analysemenu wordt gesloten met een foutmelding.

Usecase 04: Opties instellen

Use case diagram:	Opties instellen
Description:	Een gebruiker past de opties van de applicatie aan.
Preconditions:	De gebruiker is ingelogd in zijn of haar profiel.
Postconditions:	Aangepaste opties zijn opgeslagen.
Actors:	Klant
BCoE:	<ol style="list-style-type: none">1. De klant opent het optiesmenu.2. De applicatie toont een lijst met instellingen.3. De klant sluit het optiesmenu.
Alternative Paths:	<ol style="list-style-type: none">3a. De klant past een optie aan.4. De applicatie toont de aanpassingen aan de gebruiker, en vraagt om bevestiging.5. De gebruiker bevestigt de wijziging.6. De applicatie slaat de wijzigingen op.2. De applicatie toont een lijst met instellingen.
Exceptions:	<ol style="list-style-type: none">6a. De applicatie kan de wijzigingen niet opslaan.2a. De applicatie toont de lijst met instellingen en geeft een foutmelding.

3 Ontwerp

3.1 Gebruikers

De reader zal hoofdzakelijk gebruikt worden door twee soorten gebruikers, die hieronder summier worden beschreven:

Mediaspecialisten

De klanten van ClipIt: Mediaspecialisten, marketingpersoneel, online media monitoring workers, of wat voor andere naam deze functie heeft. Concreet zijn dit de mensen die contact hebben met ClipIt, en dus ook de beoogde eindgebruikers van onze applicatie.

De mediaspecialisten moeten de applicatie kunnen gebruiken om in te loggen op hun ClipIt-account, en via dit ClipIt-account nieuwsberichten kunnen bekijken die gekoppeld zijn aan hun profiel. Ook moeten ze de berichten kunnen analyseren aan de hand van vooraf berekende waarden. Als laatste moeten ze via de applicatie kleine aanpassingen aan hun nieuwsselectie kunnen maken.

ClipIt-medewerkers

Het personeel van de afdeling marketing bij ClipIt moet de reader ook kunnen gebruiken: de applicatie moet bekend gemaakt worden bij klanten van ClipIt door middel van demonstraties. Voor het gemak beschouwen wij medewerkers in deze als 'klanten', met dezelfde rechten en mogelijkheden.

3.2 Componenten

ClipIt API-connector

De data die vanuit ClipIt komt is niet meteen geschikt om te tonen in de applicatie. Er zal een vertaalslag gemaakt moeten worden van de data uit de API naar de applicatie. Hiervoor is het nodig om een API-connector op te nemen in de componenten.

APIConnector
- hostname: String - port: int - authenticated: boolean - lastRetrieve: int - apikey: String - newsitems: ClipItNewsItem[]
+ APIConnector(host: String, port: int, key: String) + authenticate(): boolean + deauthenticate(): boolean + retrieveNewsItems(): int + getNewsItems(): ClipItNewsItem[]

ClipItNewsItem

Het nieuws wordt in een datastructuur gegoten, zoals onder in het klassediagram afgebeeld. Deze lijst met variabelen en getters is wellicht nog onvolledig.

ClipItNewsItem
- id: int - title: String - content: String[] - url: String - category: String
+ ClipItNewsItem(id: int, title: String, content: String[], url: String) + getID(): int + getTitle(): String + getContent(): String[] + getURL(): String + getCategory(): String

Sorteer- en filtermechaniek

De data moet gesorteerd en gefilterd kunnen worden door de gebruiker. Dit kan op meerdere manieren, en hiervoor zal een mechaniek gebouwd moeten worden die dit kan doen.

SortFilter
+ <u>SortFilter()</u> + <u>sortNewsItems(items: ClipItNewsItem[], criteria: String[], order: boolean[])</u> + <u>filterNewsItems(items: ClipItNewsItem[], criteria: String[], action: boolean[])</u>

3.3 Interface

Hoofdmenu

Een simpel menu: als de gebruiker niet is ingelogd worden hier de credentials van de klant gevraagd. Als de gebruiker wel is ingelogd staan hier buttons naar de nieuwslijst en opties-menu, en een button met het telefoonnummer van ClipIt. Ook zal er een knop verschijnen waarmee de gebruiker weer uit kan loggen.

Nieuwslijst

Hierin staan links met een korte beschrijven van gevonden nieuwsartikelen. Een gebruiker kan op deze links klikken om het analysemenu te zien. Verder is er een button naar het hoofdmenu en een button naar het opties-menu.

Analysemenu

In dit menu wordt alle relevante analyseinformatie voor één bericht gegeven. Hieronder vallen vooralsnog in ieder geval het sentiment van een artikel, de mediawaarde en een link naar het originele artikel. Verder zijn in dit menu knoppen om terug naar de nieuwslijst en naar het opties-menu te gaan.

Opties-menu

In het opties-menu staan alle instellingen die van toepassing zijn op de applicatie. Als een gebruiker is ingelogd kan hij of zij in ieder geval de nieuwselectie aanpassen, het aantal weergegeven items in de nieuwslijst bepalen en aangeven of de applicatie de login-gegevens van de gebruiker moet opslaan.

3.4 Gegevens

Logingegevens van de gebruiker.

Bestaat uit een unieke key die gegenereerd wordt via de API, een gebruikersnaam en een wachtwoord. Er kan slechts één gebruiker tegelijk gebruik maken van de applicatie, dus er wordt geen database van logins in de applicatie zelf bijgehouden.

Zoekgegevens van een gebruiker (nieuwselectie+profiel)

Bestaat uit gegevens die via de API opgehaald worden bij de ClipIt-servers, in de vorm van XML-blokken met daarin de zoekwoorden waar een gebruiker op zoekt. Dit wordt zo snel mogelijk nog overlegd met het hoofd IT van ClipIt.

Nieuwsberichten

Deze worden ook weer opgehaald (pull) van de server. XML-blokken in de vorm van een `jnewsitemj` met attributen zoals mediawaarde, naam, titel, auteur, enzovoorts.

4 Planning

Hieronder staat een ruwe planning van onze werkzaamheden per week. Een precieze planning met een wekelijks rapport zal op de R&D-wiki te vinden zijn op onze projectpagina.

Projectweek 4

Deliverables: Ontwerpdocument

Actions: Ontwerpdocument schrijven, overleggen met domeinexpert & stakeholders.

Projectweek 5

Deliverables: Ruwe opzet van de applicatie.

Actions: Ruwe opzet voorleggen aan developmentafdeling ClipIt

Projectweek 6

Deliverables: Prototype-applicatie zonder interactie klantgegevens.

Actions: Overleg met domeinexpert ClipIt over prototype.

Projectweek 7

Deliverables: Versie twee van de prototype-applicatie.

Actions: Presentatie prototype

Projectweek 8

Deliverables: Bijgeschaafde versie prototype.

Actions: Interview domeinexpert

Projectweek 9

Deliverables: Productieversie ClipIt Reader.

Actions: Laatste overleg stakeholders bij ClipIt.

Projectweek 10

Deliverables: Eindrapport, live-versie ClipIt Reader.

Actions: Presentatie applicatie! Bij R&D en bij ClipIt op kantoor.