



Faculty of Engineering and Architecture  
iMinds - Multimedia Lab - University of Ghent

# **Advanced Multimedia Applications**

## ***Project: AVS Mobile Reporter App***

### ***Progress report***

Glenn DECOCK

Steven DE RIDDER

Brecht HANSSENS

Olivier JANSSENS

Lionel PIGOU

Academic year 2012-2013



# Contents

<b>1 General Information</b>	<b>4</b>
<b>2 General project decisions</b>	<b>5</b>
2.1 Decisions . . . . .	5
2.2 Responsibilities . . . . .	6
2.3 Requirements . . . . .	6
2.3.1 Functional requirements . . . . .	6
2.4 Risk List . . . . .	8
2.5 APIs & Frameworks . . . . .	9
<b>3 Prototype</b>	<b>10</b>
3.1 Index page . . . . .	11
3.2 Making news . . . . .	11
3.3 Tasks . . . . .	12
3.4 View news . . . . .	12
3.5 Profile . . . . .	13
<b>4 Application</b>	<b>14</b>
4.1 Front-end Mobile Application . . . . .	14
4.1.1 Cycle 1 (25/03-10/04) . . . . .	15
4.1.2 Cycle 2 (10/04 - 22/04) . . . . .	16
4.1.3 Cycle 3 (22/04 - 21/05) . . . . .	19
4.2 Back-end . . . . .	20
4.2.1 Cycle 1 (25/03-10/04) . . . . .	21
4.2.2 Cycle 2 (10/04 - 22/04) . . . . .	22
4.2.3 Cycle 3 (22/04 - 21/05) . . . . .	23
4.3 Content Management System . . . . .	26
4.3.1 Cycle 1 (25/03-10/04) . . . . .	26
4.3.2 Cycle 2 (10/04 - 22/04) . . . . .	26
4.3.3 Cycle 3 (22/04 - 21/05) . . . . .	26
<b>5 Overview of the different meetings</b>	<b>29</b>
5.1 Co-creation session at iMinds (27/02/2013) . . . . .	29
5.2 Meeting with Christophe Sevenoo at AVS (01/03/2013) . . . . .	29
5.3 Meeting with the team (05/03/2013) . . . . .	31
5.4 Meeting with the professors (08/03/2013) . . . . .	31
5.5 Meeting with the team (13/03/2013) . . . . .	32
5.6 Meeting with the team (21/03/2013) . . . . .	32

5.7	Meeting with Christophe Sevenoo at AVS (22/03/2013) . . . . .	32
5.8	Meeting with team on Skype (23/03/2013) . . . . .	33
5.9	Meeting with the professors (25/03/2013) . . . . .	33
5.10	Meeting with with team on Skype (07/04/2013) . . . . .	33
5.11	Meeting with the professors(18/04/2013) . . . . .	34
5.12	Meeting with Christophe Sevenoo at AVS (19/04/2013) . . . . .	34
5.13	Meeting with Christophe Sevenoo at AVS (29/04/2013) . . . . .	34
5.14	Meeting with the team (06/05/2013) . . . . .	35
5.15	Meeting with Christophe Sevenoo at AVS (10/05/2013) . . . . .	35

# List of Figures

3.1	First overview of some functionality of the prototype. . . . .	10
3.2	Second overview of some functionality of the prototype. . . . .	11
3.3	Third overview of some functionality of the prototype. . . . .	12
4.1	Overview of some test functionality. . . . .	14
4.2	First overview of some functionality of the application. . . . .	15
4.3	Overview of the upload functionality of the mobile application. . . . .	16
4.4	Overview of the news functionality of the mobile application. . . . .	17
4.5	Overview of the task functionality of the mobile application. . . . .	18
4.6	Overview of the mobile application. . . . .	19
4.7	Project architecture. . . . .	20
4.8	Comparison of requests per second of a Nginx versus an Apache web server. .	21
4.9	Database schemas. . . . .	22
4.10	Overview of how the Vimeo logic works. . . . .	24
4.11	Overview of some functionality of the Content Management System. . . . .	27
4.12	Second overview of some functionality of the Content Management System. .	28

# CHAPTER 1

---

## GENERAL INFORMATION

AVS, a regional TV station from East-Flanders Belgium is in search of new audience. The mobile market is booming and AVS sees the opportunity to step in this market with the idea to gain new audience.

At the moment they have a website, a Facebook page and they are broadcasting there regional content via broadcast DV pal H264 mov files. The idea is to develop a cross media interactive platform that is offering regional content that is generated and consumed by young people.

The idea is to develop a mobile cross-platform reporter application for iOS and Android. The application should provide the user with the possibility to view AVS news items through the use of filters and the possibility to act as a reporter through uploading user generated news content to AVS. This UGC can consist of videos, stills and text and it should be made possible to share news items through social media as well.

## CHAPTER 2

### GENERAL PROJECT DECISIONS

Here, we will list a number of decisions on different aspects of the project that we have all agreed upon. These decisions range from task allocation to concepts used to manage our project.

#### 2.1 Decisions

- Git with Github will be used as distributed versioning control system for the code. (5/03)
- Documents are shared via Google Drive (5/03)
- Tasks and planning will be distributed via Zoho. (5/03)
- Steven is responsible for the progress report throughout the project. (i.e. Make sure it is uploaded in time and correct)(5/03)
- Steven is in charge of the communication with the external parties (5/03)
- Olivier is responsible for Git throughout the project. (i.e. owner of the repository)(5/03)
- Glenn is in charge of the planning throughout the project. (5/03)
- AVS decided not to have Lodestar to work with us (23/03)
- AVS decided to use VIMEO for video hosting(23/03)
- We decided to put the video editing tools as nice to have. Video editing requires us to build native apps which will require too much time. (23/03)
- AVS wanted to send video fragments of a big file to the back-end to make sure at least some footage is sent. We decided to not implement this, because this requires a native app. (23/03)
- The decision was made to extend the first cycle to wednesday the 10th of april. This is because some were not able to complete their tasks, because of other deadlines. (07/04)

Decision for 2nd cyclus:

- Olivier: Develop WS to update the news.
- Glenn: Get news. (Has to work with the app and moderation site). Filter on Categories, filter on moderated or not, etc ...
- Lio: Display news.
- Steven: Make views for both cycle 1 and 2.
- Brecht: Make part to update the news in the moderation site.

Decision for 3th cycle: All the rest.

- Steven will set up a meeting with AVS to discuss the cycles.
- Steven will set up a meeting in the week of the 22th of April with AVS to discuss the moderation site and deployment options. (22/04)
- The decision was made to let the users upload the video to the back-end which automatically uploads it to vimeo. This will probably result in other work which wont be done. (29/04)

## 2.2 Responsibilities

The division of the different responsibilities can be found in Table 2.1.

Module	Responsible
Frontend Mobile App	Steven De Ridder & Lionel Pigou
Backend	Glenn Decock & Olivier Janssens
backend Content Management	Brecht Hanssens

Table 2.1: Division of the different responsibilities.

## 2.3 Requirements

In the next sections, some enumerations of various requirements are written down.

### 2.3.1 Functional requirements

#### Must-haves

##### Frontend:

- Capture video and photos via the app.
- Filtered AVS news: categories, location, most viewed.
- Share news through social media.

- Upload UGC: metadata (text, phononenumber, geolocation, etc), photos, video.
- Uploading can happen immediately after capturing the video or photo, but also from the users library. The news content to be uploaded cannot be older than 48h. Make the phononenumber field and geolocation field in the fill in form mandatory.
- Tag UGC with a UGC tag. If the UGC was for a reward, add an extra tag.
- Optional user profile: login through Facebook, Google or fill in a login form.
- Display tasks, rewards, task submission count, task due time.
- Submission of content for tasks should be with user information (from profile or filled in on the moment of submission).
- Data needs to be provided to the back-end in a standard format.
- Bread crumbs. (i.e.: The user should always be aware where he is in the app).
- Content should not get lost when the app has no internet connection or gps signal.

### **Back-end**

- UGC database: text, stills, video.
- UGC moderation (content management)
- Vimeo video hosting
- Set task with rewards and due time.

### **Nice-to-haves**

- News map: news locations on Google Maps
- Advertisements
- Add ranking to profiles based on their submitted content. (e.g.: kabelsleper, stagiaire reporter, reporter, redacteur, hoofdredacteur)
- The ability to edit and manipulate videos, depending on the possibilities of smartphone.

### **Quality attributes**

- **Performance:** responsive mobile application, large video uploads in background
- **Usability:** user friendly, minimal buttons and text, usable offline and without login
- **Portability:** Android and iOS devices

## 2.4 Risk List

1. Authentication is not implemented on time.

Impact: M

This is NOT a requirement given by the client. Though we see it as self-evident that it has to be implemented. Otherwise anyone would be able to access the data from AVS.

2. Upload access to vimeo is not granted by vimeo. (Request is pending. Started on 20/04/2013)

Impact: H (Risk eliminated)

Use the backup option and do not use Vimeo. The back-end is perfectly capable to replace vimeo. Status: <https://developer.vimeo.com/apps/26175>

3. There are missing detailed requirements.

Impact: M

Implement missing requirements if time allows it.

4. Lack of experience in certain technologies.

Impact: M

Consult tutorials or other team members

5. Implementation of a module takes longer than expected.

Impact: M

Adjust planning and resources

6. Unable to complete all requirements.

Impact: M

Well work agile so different functionalities will be working every cycle. Even if some requirements are not met, there will still be a working version of the app.

7. UI is not responsive enough.

Impact: L

Test on as many devices as possible.

8. Some team members have not much time due to thesis.

Impact: M

Other team members will do some more work

## 2.5 APIs & Frameworks

Mobile application:

- PhoneGap
- jQuery Mobile
- Vimeo API

Back-end:

- Nodejs, MongoDB, Passportjs, Mongoosejs, Express, formidable, imagemagick, request, oa
- Restful webservice
- Vimeo API

## CHAPTER 3

### PROTOTYPE

In a first phase of the project, a prototype was developed to show AVS different possibilities in lay-out and functionality. This prototype was developed with Axure<sup>1</sup>. The software is available for a trial period of 30 days. It offers the user the possibility to build, in a convenient and fast way, dynamic and functional prototypes. These prototypes are browser based application thereby making it possible to show them to a large and divers crowd. The great advantages of such an application are the interaction possibilities with the intended users.



Figure 3.1: First overview of some functionality of the prototype.

<sup>1</sup><http://www.axure.com/>

### 3.1 Index page

The index page of the application consists of four main buttons who make up the general functionalities the application provides. These are nieuws maken, opdracht, nieuws bekijken and profiel. These four buttons are very intuitive, and large enough to be easily clickable.

### 3.2 Making news

The functionality of this button is to start generating news content. Next on in the “Maak zelf je nieuws item” page, the user has to decide if he wants to make a new video, a new photo, if he wants to select content from his library or if he wants to write a news article. Both the “nieuwe video” and “nieuwe foto” button make the smart phone change view to that of the camera, where the user can start recording a video or take a photo. After this, the user has to decide if he is satisfied with the video or the photo he shot. If not, he can start recording again. If he is satisfied with the content he generated, the application takes the user to a screen where he has to fill in some contact information. This makes it easy for AVS to contact this user, in order to verify the newsitem he has sent them. The contact information consists of a name, an e-mail address, a phone number and the location of where the news was reported. Not all this information has to be filled in, but the field where the user has to fill in his phone number is mandatory, making it easy for AVS to contact this user. The user can also select the assignment linked to this news item, if there is one linked to it. Next, he has to accept the terms and conditions, after which he can finally send in the news to AVS. The user also sees an upload bar, indicating how much of his news item is already sent. The bar will be visible during the further use of the application. If the user should lose connection before the item was fully sent, the video, photo or article he generated will be stored in the library of the application. This to make sure no content is permanently lost.



Figure 3.2: Second overview of some functionality of the prototype.

### 3.3 Tasks

User also have the opportunity to generate news content in function of an assignment written by AVS. This also makes it more appealing for users to use the application to generate news on their own. When the user clicks “Opdrachten” in the index page of the application, he is led to a screen where the most popular assignments are displayed. Each such assignment in this page consists of a small description of the assignment and the reward, the popularity of this assignment and the deadline to send it in. He can choose to read more about one of these assignments, or make a search for all the available assignments. When he chooses to read more about an assignment, he is led to a screen containing more information about the assignment and the reward. On this page he can also “like” or “share” this assignment with friends, if he is for example connected using a Facebook account. When he likes to hand in this assignment, he is led again to the screen “Maak zelf je nieuws item”. If he already has content concerning this assignment, he can select it from the library of his smart phone. If not, he can start generating new content by making a video, a photo or write an article. When he has his news content, he is led again to the screen to fill in some contact information where he can also submit the assignment.



Figure 3.3: Third overview of some functionality of the prototype.

### 3.4 View news

When the user wants to look at some of the actual news provided by AVS, he will click on “Nieuws bekijken” on the index page. The application then shows three main tab pages, in this example those are “Actueel”, “Sport” and “Cultuur”. On each of these pages, one main article is being shown, followed by some smaller articles. Each article is provided with a photo and a headline. When the user clicks on a news item, he will be led to the full article containing more information about the news item, and some photos or videos. There, he can also see if the news was generated by AVS or by a specific user, along with the date and time of the news. It is also possible to provide a link to the AVS website, where the user can view

even more information, or some related items. When the user is watching news, he can also make use of some filter settings to customize what he is being presented with. The user can filter on a certain municipality, or a specific news interest.

### **3.5 Profile**

On the profile page, the user is able to login with his Facebook or his Gmail account. These two options will fill in some information in the subsequent fields, and make it possible to “like” or “share” some of the content on the pages supporting these functions. Each user can of course fill in this information himself. The main functionality of the profile page is that a specific user does not have to fill in his contact information each time he wants to submit a news item.

# CHAPTER 4

## APPLICATION

### 4.1 Front-end Mobile Application

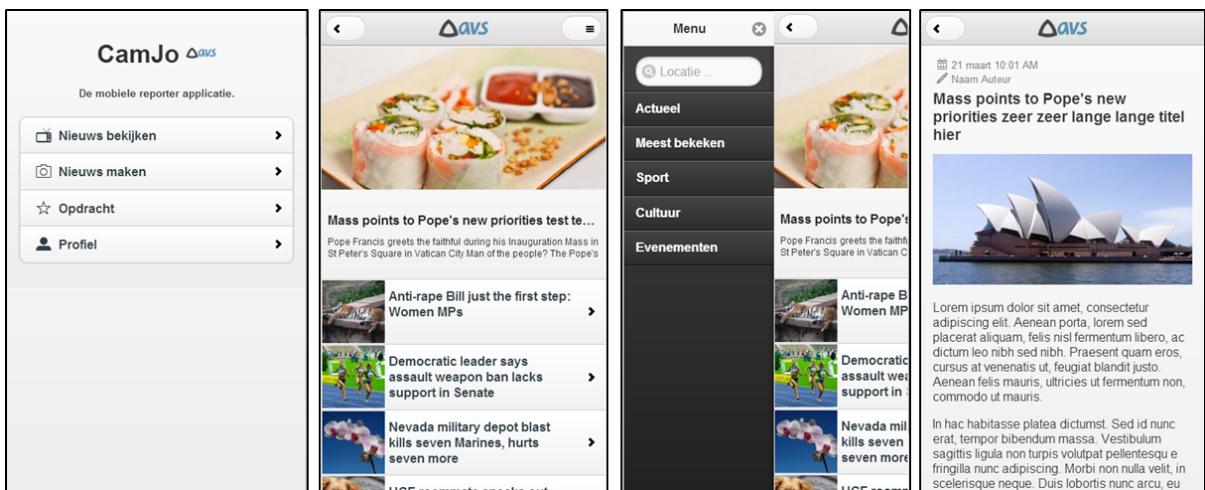


Figure 4.1: Overview of some test functionality.

A first look and feel is created using the jQuery Mobile framework. In Figure 4.1, a placeholder news overview is shown. With a toggle-able side panel the user can choose categories and news location. Tapping one of the news headlines will display a more detailed view of the article.

The application is tested on a real iPhone 4 (iOS 6.1.2) device and a cheap Android 2.x (version of 2010) phone with a small screen. The application runs responsive on the iPhone and acceptable on the Android. There is however a noticeable difference in performance between a native application and a PhoneGap application. Buttons, scrollings and transitions are, as to be expected, less responsive than native ones.

#### 4.1.1 Cycle 1 (25/03-10/04)

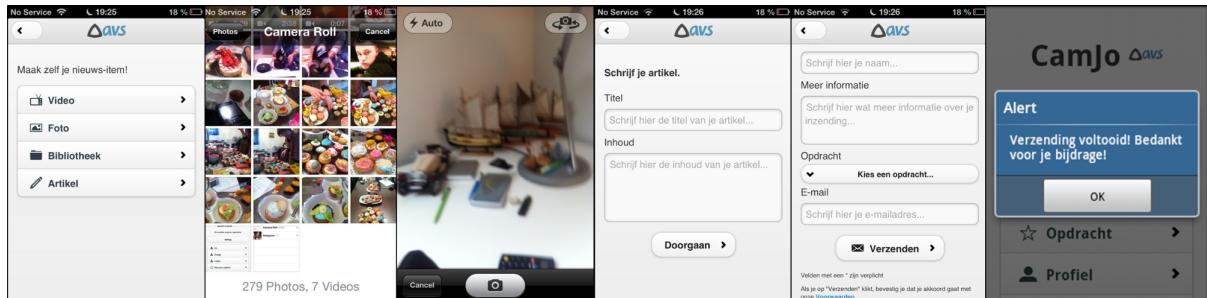


Figure 4.2: First overview of some functionality of the application.

The user can send videos or stills from their camera or from the users local library. It is also possible to write an article. After the file is selected, the user fills in information about the news. This information is also stored locally so the user wont have to re-enter everything. The date of the file is extracted and send to the server. Only one file at a time can be uploaded.

**iOS:** When choosing a file from the library the user is able to edit the video or photo. The video will be compressed automatically, which takes a relatively long time.

**Android:** Videos and images cannot be edited.

#### To do:

- Log in with Facebook and Google account integration
- Tasks
- News parsing

#### 4.1.2 Cycle 2 (10/04 - 22/04)

##### Upload

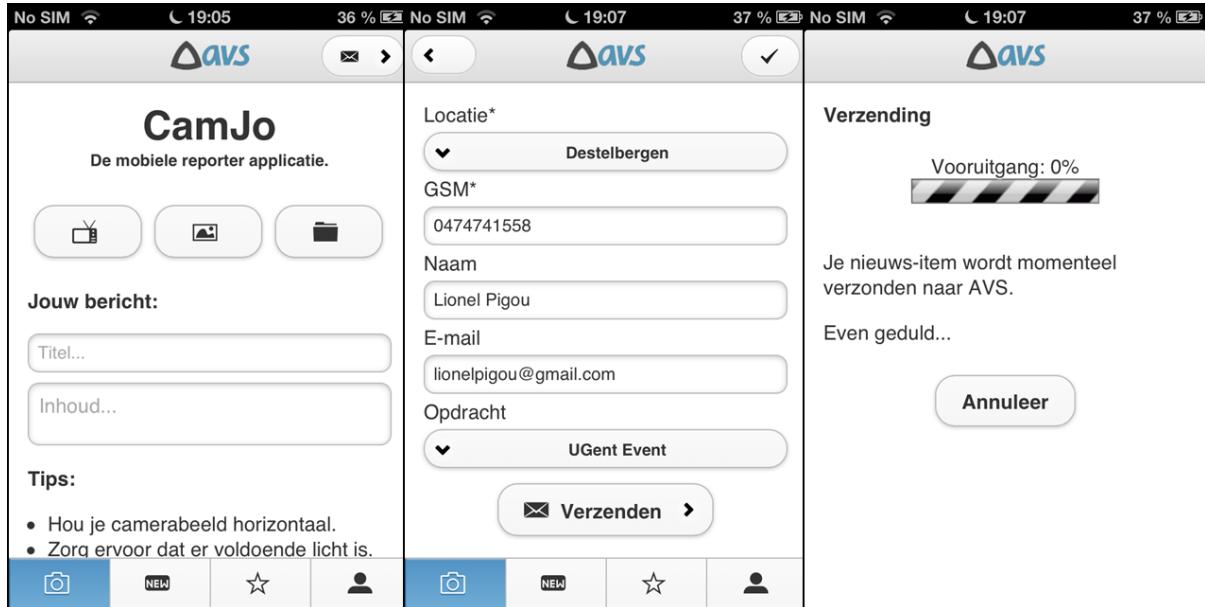


Figure 4.3: Overview of the upload functionality of the mobile application.

The **front-page** has been changed as requested by AVS. The navigation menu is a footer navbar which is displayed on almost every page. The user should also be able to quickly write a message about the uploaded content, therefore it was put on the front-page together with the 3 buttons to capture news. A few tips are displayed to guide the user.

Once the user has captured news, a page will appear that asks for information about the user and if the news submission is for a specific task. The upload progress page is shown if at least the location and the phone number is completed. This page is also made more user friendly.

## News

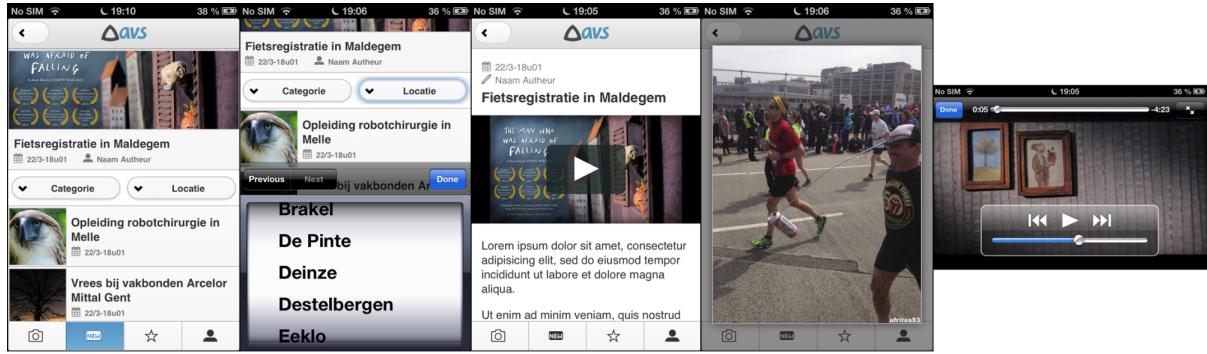


Figure 4.4: Overview of the news functionality of the mobile application.

The news can be displayed dynamically with a given array of news items in JSON format. The news items are categorized and have a specific location. The categories are currently: latest news, popular, culture and sports. There is also a distinction between user generated content and content made by the editors of AVS. The name of the author will be displayed in the news overview if it is UGC. This news overview uses thumbnails that are specified in the received news item object. Thumbnails are important, because it allows smooth scrolling and fast loading.

For videos, the Vimeo API is used to request large (for the headline) or small thumbnails. All media can be viewed full screen and the Vimeo video player is embedded using the `<iframe>` tag.

## Tasks

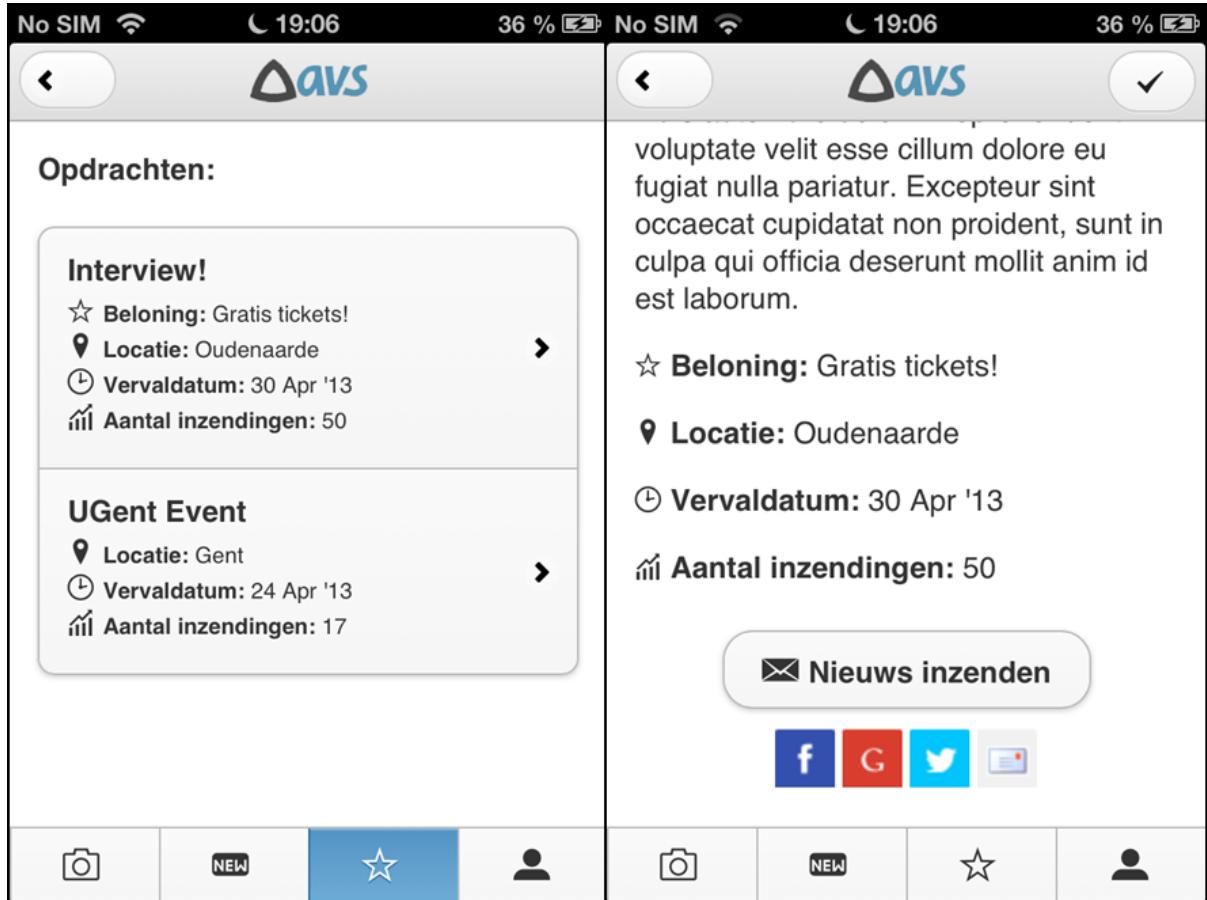


Figure 4.5: Overview of the task functionality of the mobile application.

Tasks are made by AVS, they describe a mission in which the user can participate. On a first page, an overview is shown with the location, title, reward, due date and submission count of the tasks. If the user clicks on a task, a more detailed description will be presented and the possibility to share it with social media is available.

### To do:

- Sharing news and tasks through social media
- Social media profile integration
- Testing application with test-news on the back-end database

### 4.1.3 Cycle 3 (22/04 - 21/05)

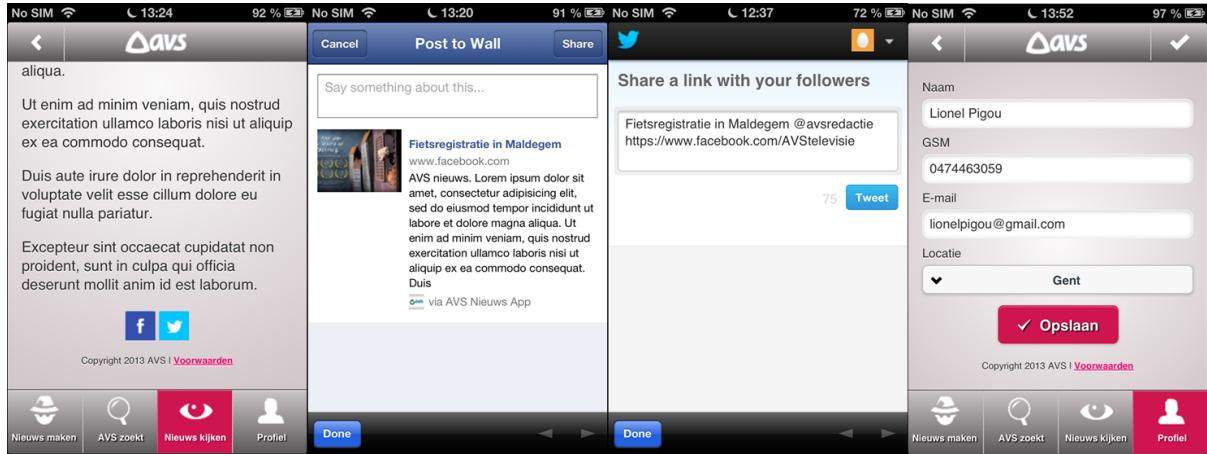


Figure 4.6: Overview of the mobile application.

In the third cycle, integration with social media is added: Facebook and twitter. We used the JQuery-share plugin (<http://plugins.in1.com/share>) to realize this. The user can share the app, tasks and news. On Facebook: the title, the description and the thumbnail will be shared. The link of the article will redirect to the Facebook page of AVS, because there is no corresponding website yet. On twitter, the user shares the title, the AVS twitter account and the AVS facebook page. There can always be added some text by the user. Phonegap uses the “InAppBrowser module to show the Facebook and twitter websites. On Android, this module did not work for some versions, so therefore we used the “ChildBrowser plugin only for Android.

The purpose of the profile is to fill in the information once and forget about it. This information is saved locally and will automatically fill in the send-form before uploading content.

The app has been tested to work properly with the backend. We uploaded photos, videos and metadata to the backend. In the database of the backend, we checked if everything is uploaded, if thumbnails are being generated and if videos are being uploaded to Vimeo. The news is then published together with some extra test-news. On the app we then see that all the news is displayed in the news-tab under the right category and location.

The design and layout developed during this third and final cycle is the result of an ongoing evolution, where the initial prototype provided a solid base which could be built upon. The customer played a crucial role in defining the final layout and overall look-and-feel of the application. Through lots of back-and-forth communication between the customer and the team, the application’s design was drawn up to suit the customer’s need and correspond with the AVS brand’s native style. The most eye-catching features are: the use of four main buttons on the “Nieuws maken” page in order to simplify the use of the app, radial gradient background in order to give the app an extra edge instead of the regular white to grey gradient background, new “Back” and “OK” buttons in the header to correspond with the styling

of the footer bar,..

## 4.2 Back-end

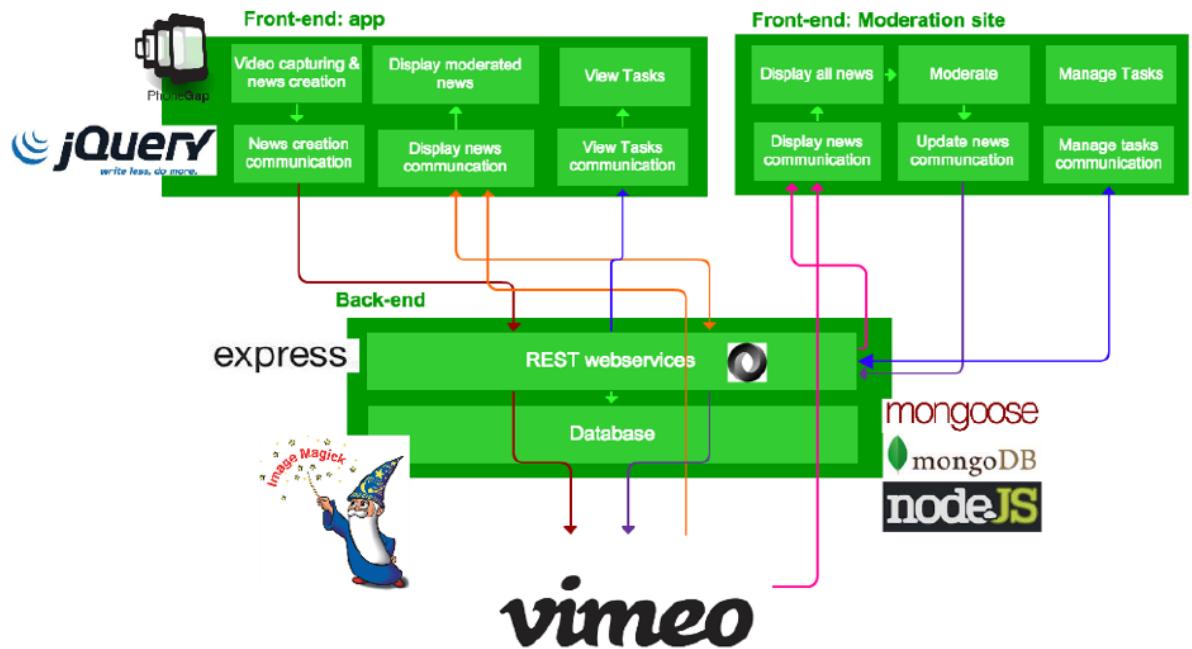


Figure 4.7: Project architecture.

Node.js is chosen because it focusses on responsiveness and scalability. This is because nodejs works with an event loop with asynchronous I/O instead of multi-threading with blocking I/O. An example of the performance can be seen here (NGINX also works with a asynchronous I/O like nodejs).

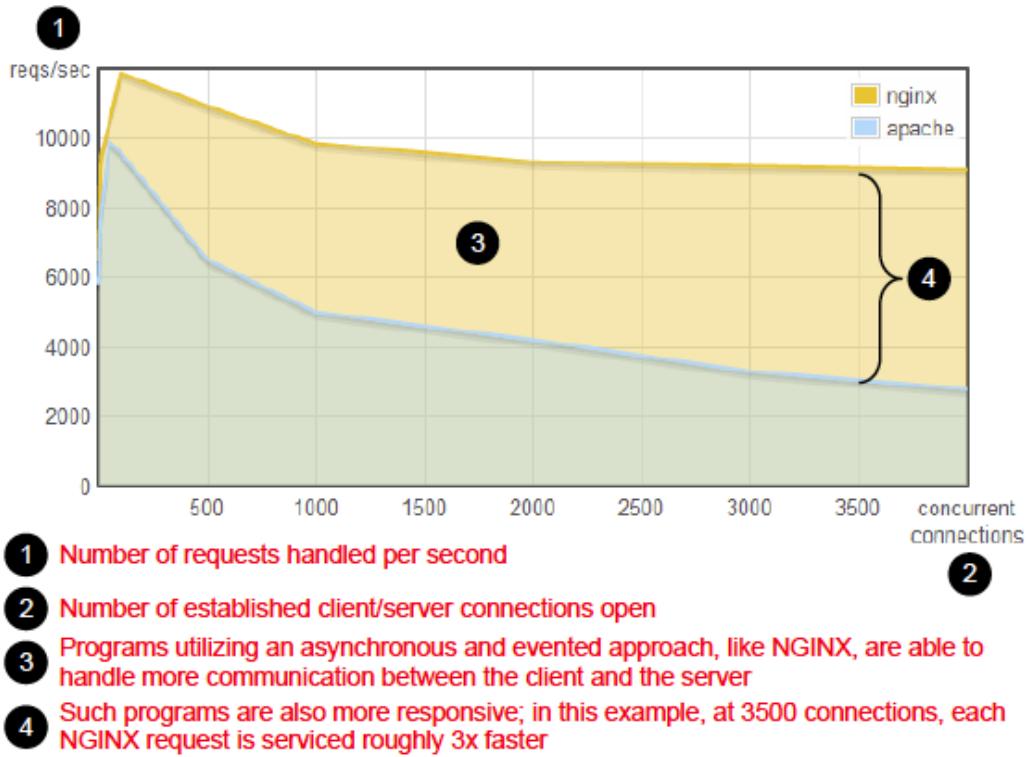


Figure 4.8: Comparison of requests per second of a Nginx versus an Apache web server.

In nodejs, like in ruby, it is possible to use 3rd party modules. In this project passportjs will be considered for the authentication and authorization; mongoDB as database; mongoose as entity framework and expressjs as web framework. Additional modules form nodejs were added during the project, such as OA to authenticate call to vimeo ( oauth2), imagemagick to rescale the image, and request to make easy request. The decision to use these technologies is based on the amount of documentation available and usability.

The decision to use Restful web services is because the focus of restful web services is on loosely coupling of different components. This means it will allow us to re-use several components from the back-end in the two different front-ends, such as the authentication service.

#### 4.2.1 Cycle 1 (25/03-10/04)

In this first cycle a Restful webservice was made to post Newsitems to the webserver. A newsitem includes metadata, pictures and videos. The webservice supports multi-file upload. The newsitems which are posted are saved in the database which has the schema of Figure 4.9

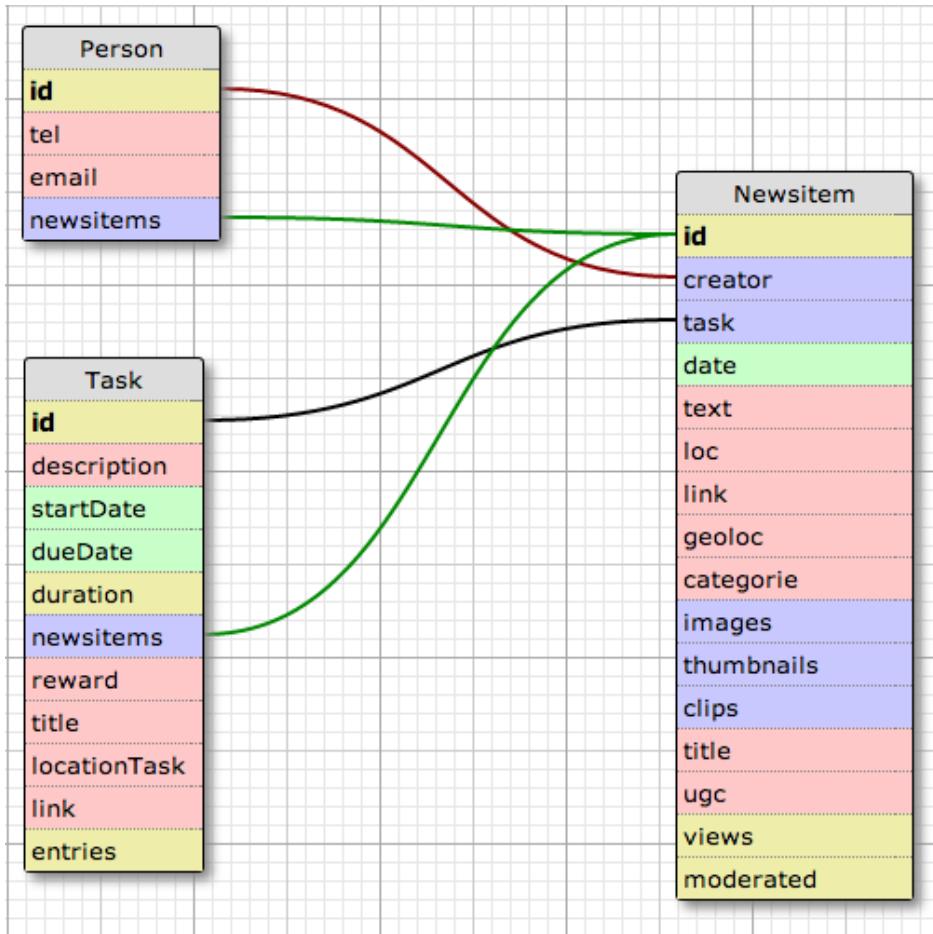


Figure 4.9: Database schemas.

As can be seen here, a newsitem belongs to a user and to a Task (though the task is not compulsory). A user and task can have many newsitems.

The files are saved in the file system and the path to the files are kept in the database. The schema was made with mongoose, the database is made with mongodb and the multi-file upload is done with formidable a nodejs module.

Another Restful web service (GET) was created in order to expose the available task. Together with this web service, two get web services were made to get news from the back-end. These two web services are: `getNewsitems` which will return all the possible news items for the front-end and the second web service is `getNewsitemById` where the possibility exists to get one news item by an unique id.

#### 4.2.2 Cycle 2 (10/04 - 22/04)

In cycle two all the CRUD (create, update, delete) Rest webservice were created for the tasks. This was also done for the newsitems. Important when managing the newsitems is the deletion of the media files which belong to the newsitems since they can take up a lot of space.

The current webservice that returns news was updated. Now it is possible to get news based on a query string. The two parameters that can be added are the location of the news and the category. The possibility exists to add some more parameters if necessary. Example: localhost:4242/news?loc=Gent&cat=Actueel.

#### 4.2.3 Cycle 3 (22/04 - 21/05)

##### Thumbnail creation

All the CRUD webservices for the newsitems have been update to work with thumbnails. This means that when a newsitem is submitted the original picture is saved and a thumbnail is generated and saved. This is also done in the update web service when a new picture is send. Thumbnails are also deleted (just like the original images) when the delete web service is called. Thumbnail creation is done with imagemagick and node-imagemagick.

##### VIMEO connection

During a meeting with AVS the decision was made to store the uploaded videos on VIMEO even if they are not moderated. This meant that extra work had to be done. The web services related to the newsitems was updated to work with vimeo.

In node.js it is possible to pipe the output of one module to the input of another. Attempts were done to pipe the output of the multi-file upload module to the vimeo module. This was in order to save temporary storage space. Though this did not work. This is because VIMEO upload API works in several steps. Firstly authentication has to happen in order to make authenticated calls to the VIMEO api. Secondly the quota has to be requested which indicates how much space is left on VIMEO for your account. Next a ticket has to be requested in order to upload. When this ticket is returned, video can be uploaded.

Since there are multiple steps before uploading to VIMEO, it is not possible to pipe the output of the upload module to the input module for VIMEO. This also means that uploading takes twice the normal time (time to upload to the back-end + time to upload to VIMEO from the back-end). Though since node.js is chosen, files are uploaded in parallel since node.js works asynchronously.

At the moment the vimeo upload works as Figure 4.10.

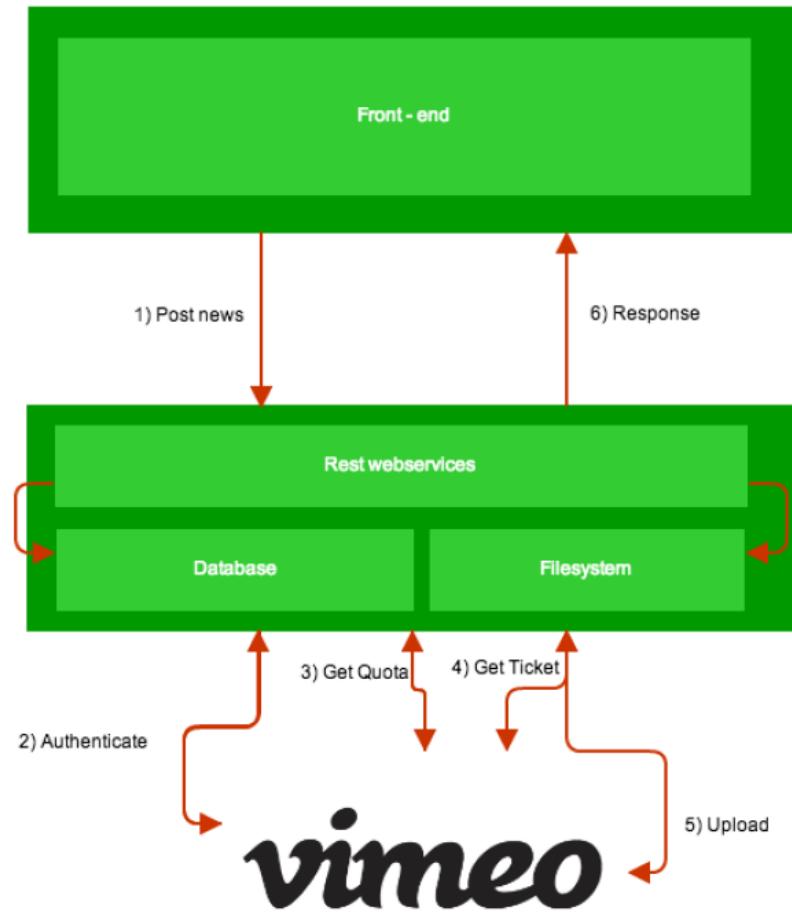


Figure 4.10: Overview of how the Vimeo logic works.

Videos are first set as private so that no one can access them except for AVS. Only when the newsitem is updated and the access rights are changed, then the video becomes public. This can be done via the moderator site.

It is also possible to delete a newsitem which will automatically result in all videos related to that newsitem to be deleted on vimeo.

AVS made some remarks about the news web service. This web service returned all the news that are available in the database. This has been changed with a date check. This means that news that is older than 14 days as of today will no longer be returned. An extra parameter has been added so that it is possible to ask news based on the fact that it is moderated or not. A sort parameter has been added to the web service as well so that the news item that was most viewed by the users will be returned as the first item and the rest of them in descending order.

AVS demanded to add two new schemas to the database so that it will be possible to add “gemeentes” and to add news “categorien”. The gemeenteSchema was added which has one element, a name of a town. The categorieSchema has also one element, a category. Based on these two schemas it will be possible for AVS to add new “gemeentes” and to add new

“categorien” which will automatically be available in the mobile application. Otherwise the list of “gemeentes” and “categorien” has to be hard coded in the mobile app and it would be impossible to change this item which would be a major flaw.

The list of web services created can be found below.

- WS to create newsitems (multi-file upload)
- WS to update newsitem (multi-file upload)
- WS to delete newsitems
- WS to create Tasks
- WS to display Tasks
- WS to update Tasks
- WS to delete Tasks
- WS to display individual Task
- WS to display news items
- WS to display individual news item
- WS to create a “gemeente”
- WS to update a “gemeente”
- WS to delete a “gemeente”
- WS to display a list of all “gemeente”
- WS to create a “categorie”
- WS to update a “categorie”
- WS to delete a “categorie”
- WS to display a list of all “categorie”

The goal was to add the authentication module to the back-end in this third cycle. This was not a requirement by AVS but our idea was that it would be nice that only authenticated users could access the moderation site. The technology that we wanted to use is Passportjs. Due to problems with this technology, not knowing where to place the appropriate code, not finding any good tutorials and examples, spending time to other extra requirements and updates by AVS, this authentication module was not added to the final code. More information about the Passportjs technology can be found on the website<sup>1</sup>.

---

<sup>1</sup><http://passportjs.org/>

## **4.3 Content Management System**

### **4.3.1 Cycle 1 (25/03-10/04)**

In this first cycle a simple Content Management System (CMS) was made which shows the news items that were being submitted. This CMS has the intent to select news items in order to properly moderate them. In an index page, a quick summary of all the different news items is given. Here, moderators can select a specific news item to take a broader look at it, which takes them to a moderating page. On this page, they are given additional information about the news item, including who has sent in the item, personal information about this user, what the additional photos and videos are, etcetera.

### **4.3.2 Cycle 2 (10/04 - 22/04)**

In a second cycle, the CMS was enriched with the functionality to moderate the news items. These moderation aspects are editing the content of news items, or even deleting the item as a whole. This moderating occurs on the web page where one can view the full details of a news item. Here, one can edit certain aspects of this item (e.g., correct typographical errors), or remove certain malicious content. Analogous to the first cycle, a web page was also developed which shows the created tasks. First on a quick summary of all the tasks in an index page, and later on in full detail on a moderating page. In order to make the CMS site more visually appealing, the necessary CSS styling was applied in correspondence with the AVS brand's native style. The new design and layout provides the user with a cleaner and more structured way to make alterations in the database and perform the different moderation tasks. Next to that, the pages where moderators can make their own news item, or create a task, were further developed and finalised. One can also sort the news items for moderated or unmoderated.

### **4.3.3 Cycle 3 (22/04 - 21/05)**

In a third cycle, the CMS was finalised on different fronts. The displaying of thumbnails for photos or videos on the news index page was added to the management system. The moderation of news items was enhanced with the functionality to add news items to a specific task, or remove them from it. Therefore, the moderator is given a list with all the available tasks on the page where he can moderate the contents of an item. In this development cycle, the moderator was also given the functionality to set the type of the news item to unmoderated, or set it as moderated. He's also given the option to make the video in this news item private or public on the AVS Vimeo page. It's also possible to add or delete media in each news item. As such, one can add certain photos or videos to the item, or delete a few of the original media files.

The task-part of the CMS was also enriched with the functionality to take a look at all the news items that were submitted for each task.

 NIEUWS

AL HET NIEUWS

NOG NIET GEMODEREERD

Nieuws toevoegen



## #1: BEWERKTE TITEL MET FOTO

Locatie: Wetteren

Datum: 20/05/2013

Inzender: Brecht | GSM: 7

Bewerk



## #2: TITEL VAN EEN TWEEDE NIEUWSBERICHT

Locatie: Gent

Datum: 21/05/2013

Inzender: Brecht | GSM: 7

Bewerk



## #3: TITEL VAN EEN DERDE NIEUWSBERICHT

Locatie: Gent

Datum: 21/05/2013

Inzender: Brecht | GSM: 7

Bewerk

Figure 4.11: Overview of some functionality of the Content Management System.

**Q TAAK BEWERKEN**

**Annuleren** **Verwijderen**

**Titel\***  
Interview met Herman Brusselmans

**Locatie**  
Gent

**Start datum**  
20/05/2013

**Duur van de taak (#dagen)\***  
15

**Beschrijving\***  
Beschrijving van het interview.

**Beloning**  
Een reis naar Ibiza

De nieuwsitem die tot deze taak behoren:  
[Nieuwsitem #1: Bekijk](#)

**Opslaan**

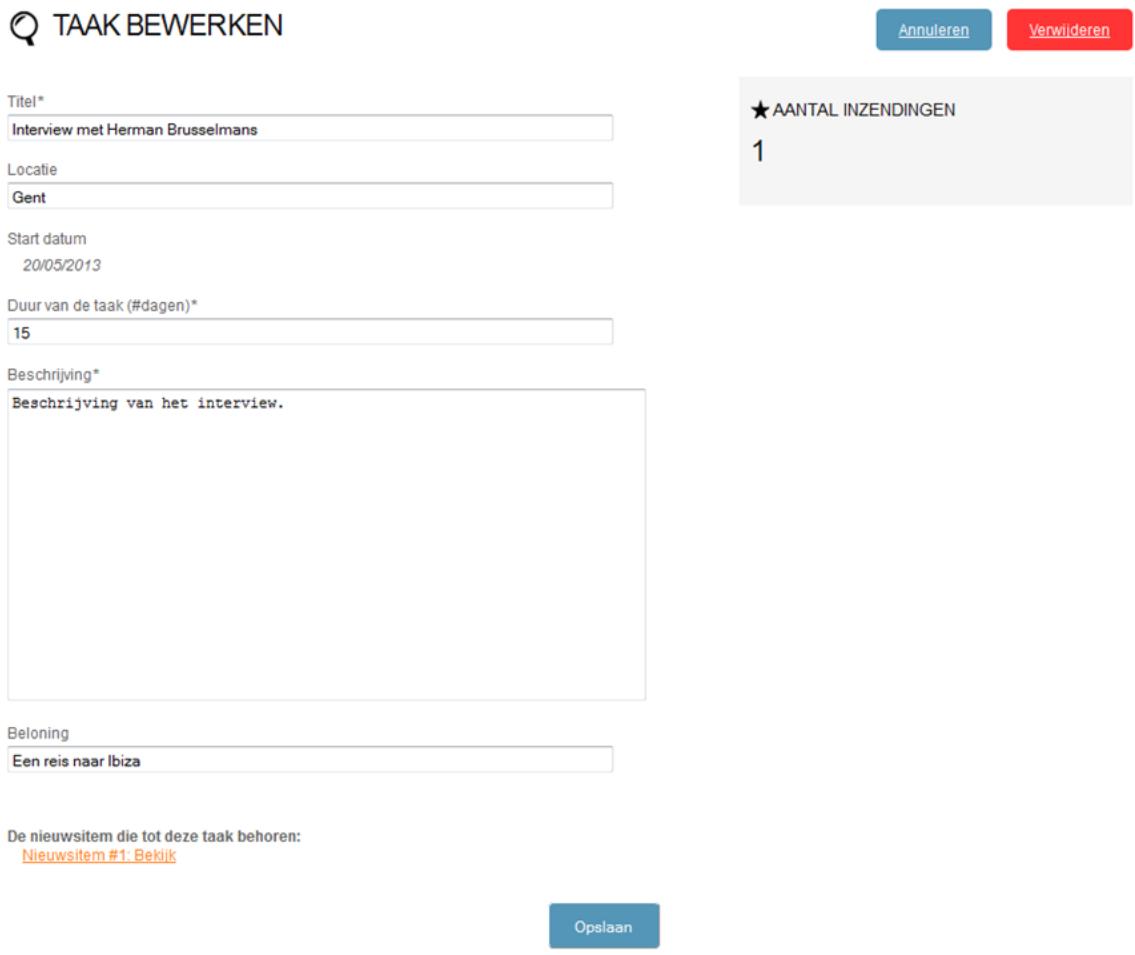


Figure 4.12: Second overview of some functionality of the Content Management System.

## CHAPTER 5

### OVERVIEW OF THE DIFFERENT MEETINGS

#### 5.1 Co-creation session at iMinds (27/02/2013)

- **Present:** Steven De Ridder, Lionel Pigou, Christophe Sevennoo, AVS spokesperson, several volunteers from the target audience
- **Goal:** Obtain requirements and ideas for the AVS mobile reporter app directly from the target audience through playful exercises
- **Results:** The co-creation session was well organised and a steady pace was kept in order to make sure everybody was actively taking part in the exercises and a maximum amount of thought was given to the different questions. Several interesting ideas were presented and the session seemed to confirm the general trend in results from the earlier intake survey. We were able to establish a concrete idea on how people would like to see the app being presented and what functionalities it should contain. However, in contrast to what was labeled as a nice-to-have feature in the results of the intake survey, being the challenge/reward system, this functionality was conceived as being critical to the success of the app. Nearly everybody present during the session responded positively to the idea and agreed that it would stimulate the interest of the end user to actually actively generate content. In order to even further increase the level of commitment, some sort of ranking system should be implemented. Here, a user whose content has effectively been used to create a AVS news system would step up in the ranking system. This would lead to numerous new possibilities.

#### 5.2 Meeting with Christophe Sevennoo at AVS (01/03/2013)

- **Present:** Steven De Ridder, Olivier Janssens, Christophe Sevennoo.
- **Goal:** Requirements discussion, limit the scope of the project to the most important requirements of the project, get an insight into the technical environment present at AVS with which our system needs to cooperate with, schedule further appointments according to our agile project planning

- **Results:** During this meeting we were able to pinpoint the most important desired aspects of the app: the ability to view AVS news through the use of filters on a mobile platform and the ability to create user generated news content and upload this to the AVS news server. The app, in order to boost the number of potential users, should also contain the possibility to show the different challenges that AVS has created together with their respective reward. Users should be able to log in using their social media account, in order to help speed up creating a user profile. A crucial element within the user information is ones cell phone number, in order to provide the editorial staff at AVS the possibility to contact the reporter and double check the validity of the UGC before using it in an AVS news item. When one is using the app without a profile or without logging in, it is crucial that the UGC can only be uploaded upon entering the cell phone number. AVS is currently negotiating with Lodestar about the development of a central news server. The idea is that all the news items generated by AVS will be stored in this central location, from where these news items can be pushed to different locations being television, social media, the mobile app,... Lodestar will also provide software to moderate the content on this server. Since our system will be strongly connected to this central server, it is important that we get more information on how this system will be implemented. Christophe has provided us with contact details for the Lodestar company so we can directly contact Lodestar. An e-mail has already been sent. Another crucial aspect that came to mind is the way in which compression for the UGC will be implemented. Again, we first of all need to establish contact with Lodestar to further get an idea on how this can be realised. Christophe responded positively to our proposed agile project planning, this way he can be strongly involved in the development process and he can steer us in the right direction after every development cycle if our vision on a certain aspect of the project would differ from what AVS expected. Christophe also pointed out that for the visual design of the app, we can contact the graphical department within AVS who can create visual element corresponding to the typical AVS style.

### **5.3 Meeting with the team (05/03/2013)**

**Present:** Everyone from the team

**Purpose:** Requirements discussion, initial division of tasks, discussion workflow

**Results:**

- Steven will contact lodestar ASAP.
- Olivier and Steven will work on the requirements. (see below)
- Lionel will develop the workflow which was discussed. (see below)
- Glenn will concretise the planning which is due friday.
- Brecht will conduct further research in web frameworks.
- Olivier will be the spokesperson on friday.

### **5.4 Meeting with the professors (08/03/2013)**

**Present:** Everyone from the team + all the professors.

**Purpose:** Elaborate the progress made in the project

**Results:**

- We need a planning with exact dates and tasks. These tasks need to be clear and need an allocation of who will do it.
- We need to adjust the nice to haves in our last progress report.
- When a meeting with Lodestar is made, a report of this needs to be present.
- We need a complete technology stack until the next meeting with the professors.
- We need clear API's
- If we go Agile or Scrum, a progress report every week or every two weeks need to be send to the professors.
- When making a prototype, use Axure software. This is interactive software used to build such prototypes.
- When the design is finished, AVS needs to sign this off.
- Make two plans. One for the back end and one for the front end.

## 5.5 Meeting with the team (13/03/2013)

**Present:** Everyone from the team

**Purpose:** Discuss the design, discuss the modules, make clear agreements, make initial division of the modules

**Results:**

- An initial prototype was build on paper. Brecht will transform this to a prototype in Axure.
- A first devision of modules has been made. Brecht will do the Transcode module, Steven will do the database, Olivier will do the webservices, Lionel will work on the front end and Glenn will look for the HTTP live streaming module.
- An overview of the project had been made to send to Lodestar. This list contains our ideas and questions about their part in this project.
- Steven will send this document to Lodestar. We want to make asap a meeting with them.
- A discussion has been made about the problems that can occur with Lodestar and possible sollutions.

## 5.6 Meeting with the team (21/03/2013)

**Present:** Everyone from the team

**Purpose:** Discuss progress, planning, agile development cycles

**Results:**

- Initial cycle
- Meetings with AVS and lodestar

## 5.7 Meeting with Christophe Sevenoo at AVS (22/03/2013)

**Present:** Lionel Pigou, Brecht Hanssens, Steven De Ridder, Christophe Sevenoo

**Purpose:** Requirements approval, prototype

**Results:**

- Adjustments to the prototype:
  - A bigger Nieuws maken button
  - Users must be able to choose a task after capturing something
  - A date must be added to the news detail page
- Approval and adjustment of requirements:
  - Profile ranking is a nice-to-have instead of a must-have

- Video editing is a must-have
- Captured news is a must-have
- Vimeo will be used to host videos

## 5.8 Meeting with team on Skype (23/03/2013)

**Present:** Everyone from the team.

**Purpose:** Restructure planning, updating progress report.

**Results:**

- Http Live Streaming and transcoding modules are removed, because video will be uploaded on Vimeo
- Content management front-end and back-end modules are added.
- Updated progress report and planning
- Decisions added to decision list.

## 5.9 Meeting with the professors (25/03/2013)

**Present:** Everyone from the team, Prof. Rik Van de Walle, Erikmannens.

**Purpose:** Update.

**Results:**

- Keep track of graphical designer, it is our responsibility.
- Adjust timing with a forecast of the cycles.
- Keep a rough estimate of the time we have spent on the project.

## 5.10 Meeting with the team on Skype (07/04/2013)

**Present:** Everyone from the team

**Purpose:** Getting up to date, refine planning cycle 2&3, update progress report

**Results:**

- Planning has been updated
- Ideas were exchanged

## 5.11 Meeting with the professors(18/04/2013)

**Present:** Everyone

**Purpose:** Discuss progress

**Results:**

- We have to update our risk list.
- Update planning.
- Ask AVS about deployment options.

## 5.12 Meeting with Christophe Sevenoo at AVS (19/04/2013)

**Present:** Lionel Pigou, Christophe Sevenoo, AVS artists

**Purpose:** Demonstrating progress made so far

**Results:**

- Requirements are signed.
- Christophe expects us in 1 or 2 weeks with a working version of the moderator website.
- Layout, logos and icons will be sent to Steven on 23/04
- Olivier and Glenn must contact Christophe to discuss back-end deployment.

## 5.13 Meeting with Christophe Sevenoo at AVS (29/04/2013)

**Present:** The whole team, Christophe, Thinline person, AVS intern

**Purpose:** Demonstrating progress on the moderator side. Discuss deployment options. Discuss data access for the central system which might be developed by thinline.

**Results:**

- Christophe was very satisfied with the moderator side. He only had one question, “What is displayed on the moderator site when a video is sent instead of a picture” (regarding the thumbnail) This issue will be solved by Brecht.
- A url attribute should be added to a news item.
- The moderator attribute should be changed to an integer so that several levels can be indicated.
  - 0 = Unmoderated
  - 1 = Moderated
  - 2 = Published

This was a request made by thinline

- Only news items which are older than now - 2 weeks should be displayed in the moderator side. This will be solved by Glenn

- For the deployment option a long discussion took place together with an internal administrator of their servers. The decision is made to upload all videos to Vimeo (moderated and unmoderated), thereby decreasing the pressure put on the web server. This will require extra time, which will have to be taken from another element.
- Further deployment details will be provided for the webservice.
- AVS will put the apps on the correct appstores (together with thinline)
- Christophe will provide us with the “algemene gebruiksvoorwaarden”
- Christophe will provide a small manual in order to use the app.

## 5.14 Meeting with the team (06/05/2013)

**Present:** The whole team

**Purpose:** Discuss progress

**Results:**

- Steven has implemented the design given by AVS.
- Glenn has contacted Combell in order to get the possibilities for the hosting
- Steven will contact AVS to set-up the acceptance testing.
- Glenn will mail Combell with the webserver requirements.
- Glenn will continue to work on some web services and if there is still time the authentication
- Olivier changed the upload WS for the news item so that it works with VIMEO
- Olivier changed the CRUD newsitem WSes to work with thumbnails.
- Olivier changed the delete WS to remove videos from VIMEO.
- Olivier will continue to work on the VIMEO part.
- Lionel & Brecht continue to work on their part.

## 5.15 Meeting with Christophe Sevenoo at AVS (10/05/2013)

**Present:** Steven De Ridder, Christophe Sevenoo, Philip from AVS's graphical department

**Purpose:** Discuss the adapted design

**Results:**

- The overall design was examined in full detail to indicate its strong points and weaknesses.
- Based on this detailed examination, alternative designs and layouts were discussed.
- Both parties reached an agreement on how the design should be altered.

- Steven will correspondingly make the necessary changes and show the result to Christophe in order to obtain feedback.