



**Rui Filipe Monteiro Queirós**

Licenciado em Ciências e Engenharia Informática

## **An Online Crowd-video service for collaborative event recording**

Dissertação para obtenção do Grau de Mestre em  
**Engenharia Informática**

Orientador: João Magalhães, Professor Auxiliar,  
Faculdade de Ciências e Tecnologia  
Universidade NOVA de Lisboa

Co-orientador: Nuno Correia, Professor Catedrático,  
Faculdade de Ciências e Tecnologia  
Universidade NOVA de Lisboa

Júri

Presidente: Professor Doutor Pedro Manuel Corrêa Calvente Barahona  
Arguente: Professora Doutora Maria Teresa Caeiro Chambel  
Vogal: Professor Doutor João Miguel da Costa Magalhães



FACULDADE DE  
CIÊNCIAS E TECNOLOGIA  
UNIVERSIDADE NOVA DE LISBOA

**Dezembro, 2016**



## **An Online Crowd-video service for collaborative event recording**

Copyright © Rui Filipe Monteiro Queirós, Faculdade de Ciências e Tecnologia, Universidade NOVA de Lisboa.

A Faculdade de Ciências e Tecnologia e a Universidade NOVA de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

Este documento foi gerado utilizando o processador (pdf)  $\text{\LaTeX}$ , com base no template "unlthesis" [1] desenvolvido no Dep. Informática da FCT-NOVA [2]. [1] <https://github.com/joaomlorenco/unlthesis> [2] <http://www.di.fct.unl.pt>



*To my parents.*









## ACKNOWLEDGEMENTS

I want to start to thank Faculdade de Ciências e Tecnologia, NOVA that helped me to grow individually and professionally since the first day I stepped the university. I also want to express my deep thanks to professors João Magalhães and Nuno Correia for their patience, help, advices and availability during the realization of this work.

I also want to thank my colleagues Rafael, Diogo, Ricardo Monteiro, Guilherme, Leonardo, Alexandre, Albert, Luís, André, Ricardo Pinto, Marta, Mário and Gonçalo for sharing with me all those amazing years during the course. I want to give a special thanks to Henrique which was not only my roommate, my partner in most of the graduation courses but which always gave me amazing help and support. I want to also thank all my friends who accompanied me before and during this journey with special thoughts to Rosa (you could have cooked more often), Rita Paulino, Rita Vieira, Daniel, João Silva, Diogo Santos, Inês Barbosa, Vítor, Pedro Alves, Tiago, Nuno Pereira, Nuno Rodrigues, Gonçalo, João Simões, Rita Pereira, Patrícia, Sofia and Leonor, I am very lucky and proud to have known you all. I want to thank Accenture and Ocidental clients where I worked during this year and KCS IT, where I am employed, for their opening and comprehension for my delays due to morning meetings in college.

Finally, I want to thank my family, all my uncles, cousins, my brother and my amazing parents, to the ones who know what I was doing during these years and to the ones who have absolutely no clue, you are all fantastic and one way or another helped me during this troubled times and of course my cat Salazar who made last months a lot more painful but also fun.



## ABSTRACT

---

Each day, technological evolution allows access to information in new ways. Traditional text-based newspapers were, in times past the main source of information, but now there is more and more options. First, with radio transmissions, then with the television, and more recently with the Internet. Nowadays, it is easy to get access to information and multimedia through images and video which can be accessed anytime, anywhere. This new source of information is gaining more and more fans, not only because it is easy to reach, but also because it offers the information needed in quick and effective ways.

This technological evolution does not stop, so now mankind wants to be more than mere receptors, they also want to take an active part in this process as a source, sharing their own information through those new technologies, with people from all over the world.

In this dissertation, we developed a collaborative system where users will share their own videos and collaborate between themselves to develop events related to a particular event. Our intention is to engage users into contributing to grow the available data regarding that event.

With this solution we want to offer, not only some features that similar and already existing systems like Youtube and Vimeo present, that we assume to be basic and necessary, but also some new like timelines and the possibility to every user contribute to them..

**Keywords:** Multimedia, Video Sharing, Event Recognition, UHD, 4K, Crowd-Video, Event Recording, Collaborative System, Video Gathering, Video, Classification System, Competitive System, Rewards

---



## RESUMO

---

A evolução tecnológica está a permitir a cada dia que passa o acesso a informação através de diferentes formas. Os tradicionais jornais à base de texto escrito que em tempos eram a principal fonte de informação foram ganhando cada vez mais novos concorrentes inicialmente com a rádio através da transmissão do som, mais tarde a televisão e agora a internet. Nos dias que correm há uma enorme facilidade de acesso à informação e a multimédia através de imagens e vídeos que podem ser consultados a qualquer altura começam a ganhar cada vez mais adeptos, não só porque causa da sua facilidade de acesso, mas também por oferecerem a informação pretendida de forma rápida e eficaz.

Mas essa mesma evolução tecnológica não para, e hoje em dia o ser humano não quer ser um mero receptor, mas também quer fazer parte ativa deste processo. Adquire a função de fonte, partilhando a sua própria informação através dessas novas tecnologias com pessoas de todo o mundo.

Nesta dissertação vamos desenvolver um sistema colaborativo onde os utilizadores poderão fazer a partilha de vídeos próprios e colaborando depois entre si para o desenvolvimento de eventos relacionados com um determinado evento. A intenção é incentivar os utilizadores a contribuírem positivamente para o crescimento desse mesmo evento de forma a que o conjunto de vídeos que o representa seja o mais vasto possível.

Com esta solução queremos oferecer não apenas algumas funcionalidades que são oferecidas por sistemas semelhantes e já existentes como Youtube e Vimeo e que assumimos que são básicas e necessárias, mas também algumas funcionalidades novas como a criação de timelines e a possibilidade de os utilizadores contribuírem para o crescimento das mesmas.

**Palavras-chave:** Multimédia, Partilha de Vídeo, Reconhecimento de Eventos, UHD, 4K, Crowd-Video, Gravação de Eventos, Sistema Colaborativo, Recolha de Vídeo, Vídeo, Sistema Classificativo, Sistema Competitivo, Recompensas

---



# CONTENTS

<b>List of Figures</b>	<b>xvii</b>
<b>List of Tables</b>	<b>xix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Scope and Objective . . . . .	1
1.2 Proposed Framework . . . . .	2
1.3 Challenges . . . . .	3
1.4 Contributions . . . . .	4
1.5 Document Organization . . . . .	4
<b>2 Related Work</b>	<b>5</b>
2.1 Collaborative Video Systems . . . . .	5
2.1.1 Event Detection In Social-media . . . . .	5
2.1.2 Events Visualization . . . . .	6
2.1.3 Collaborative Media and Video Annotation . . . . .	9
2.1.4 Quality Through Engagement . . . . .	11
2.2 Public Solutions . . . . .	12
2.2.1 Platforms With Video Visualization . . . . .	12
2.2.2 BBC Have Your Say . . . . .	19
2.2.3 Google Photos . . . . .	19
2.3 Summary . . . . .	20
<b>3 Livetime</b>	<b>23</b>
3.1 Requirement Analysis . . . . .	23
3.1.1 Use Cases . . . . .	23
3.1.2 Other Requirements . . . . .	27
3.1.3 Future Features . . . . .	27
3.2 Architecture . . . . .	28
3.3 Background Technologies . . . . .	30
3.3.1 User Interface . . . . .	30
3.3.2 Web Development Framework . . . . .	31
3.3.3 Database . . . . .	31

## CONTENTS

---

3.3.4	Video uploads . . . . .	32
3.3.5	Twitter API . . . . .	32
3.3.6	Leaderboards . . . . .	32
3.4	Development . . . . .	33
3.4.1	Domain Model . . . . .	33
3.4.2	Application . . . . .	34
<b>4</b>	<b>Building Better Timelines</b>	<b>41</b>
4.1	Timelines . . . . .	41
4.1.1	Event Storylines . . . . .	41
4.1.2	Microblog Contribution . . . . .	42
4.1.3	Visualizing Timelines . . . . .	43
4.1.4	Detecting Storygaps . . . . .	46
4.2	Collaborative-Competition . . . . .	47
4.2.1	Rewarding . . . . .	47
4.2.2	Involving End-Users . . . . .	48
<b>5</b>	<b>Evaluation</b>	<b>53</b>
5.1	Methodology . . . . .	53
5.2	Test Scenario . . . . .	55
5.3	Request Videos . . . . .	58
5.4	Conclusion . . . . .	59
<b>6</b>	<b>Conclusion and Future Work</b>	<b>61</b>
6.1	Conclusion . . . . .	61
6.2	Contributions . . . . .	62
6.3	Future Work . . . . .	62
	<b>Bibliography</b>	<b>65</b>
<b>A</b>	<b>Usability Tests Guide</b>	<b>69</b>
<b>B</b>	<b>Usability Tests Questionnaire</b>	<b>73</b>
<b>C</b>	<b>Usability Tests Questionnaire Answers</b>	<b>79</b>



## LIST OF FIGURES

2.1	Example of the interface created by Boreczky [4] (left) and Balabanovic [2] (right) . . . . .	7
2.2	A set of screenshots from Meerkat interface . . . . .	13
2.3	A set of Periscope screenshots that shows application interface . . . . .	14
2.4	A set of Snapchat screenshots that shows application interface . . . . .	17
2.5	A collage example using a selection of photos at Google Photos . . . . .	19
3.1	Livetime architecture model . . . . .	29
3.2	Current domain model already developed for Livetime application . . . . .	33
3.3	An Example of A Sequence of Actions in Livetime Application . . . . .	35
3.4	Livetime Homepage . . . . .	36
3.5	Livetime Page For Video Upload . . . . .	37
3.6	Video Page in Livetime Application . . . . .	38
3.7	Livetime Commentary Section . . . . .	39
3.8	Livetime Listings . . . . .	39
4.1	Livetime Horizontal Timeline . . . . .	42
4.2	Livetime Horizontal Videos Sequence . . . . .	43
4.3	A Twitter Contribution Made By a Livetime User To Twitter . . . . .	44
4.4	Livetime Vertical Timeline Inside A Contest . . . . .	45
4.5	An Example of A Storygap Found in Livetime . . . . .	46
4.6	Form Of A Contest About To Be Created . . . . .	47
4.7	Livetime Contest Leaderboard . . . . .	48
4.8	Livetime Global Leaderboards . . . . .	50
4.9	Livetime Twitter Account With Automatic Tweets . . . . .	51
5.1	System overall experience classification . . . . .	56
5.2	Usefulness of timelines . . . . .	57
5.3	Usefulness of contests . . . . .	57
5.4	Usefulness of competitions . . . . .	58
5.5	Usefulness of twitter . . . . .	58
5.6	Concerts in Portugal in 2016 Timeline Result . . . . .	60



## LIST OF TABLES

2.1	Comparison between described systems - Part 2/2 . . . . .	18
2.2	Comparison between described systems - Part 1/2 . . . . .	18
5.1	Results From SUS Questionnaire. . . . .	56



## INTRODUCTION

Each day, technology keeps evolving in different ways with new pieces of hardware and software providing users with new ways of interacting among themselves and with media. With our solution, we explore some of the existing work in the area of social-media collaborative UHD video and finding and filling the gaps where new features can be researched. This thesis focus on the work of a new web platform system called Livetime which focuses on user collaboration for sharing videos and event creation depicting real-world events, and at the same time exploring ways to encourage users to participate with quality contributions.

### 1.1 Scope and Objective

Thanks to the evolution of technology which made storage cheaper and cameras spread into almost every handheld device (e.g. laptops and smartphones), Internet is now being flooded with videos from everywhere and by everyone. Now, more than ever due to applications like Youtube, DailyMotion or VIMEO, companies and people in general are uploading videos to reach out larger communities of users. It is an easy way to try to promote a movie, a video game or a song, so in these web applications we can find almost everything and thanks to these applications there are also plenty of other growing web pages that are using content made mainly to these web applications. These applications are prime examples of Internet's importance and it is still spreading into other directions like video streaming which is getting widely adopted thanks to applications like Twitch, Twitter Periscope or Meerkat where people connect to real-world live videos broadcasts.

Finding the gaps in this scenario leads to a new web video application that fulfills a user need which is not only a big motivation but also a challenge. To achieve this, it is important to be clear about a set of basic features to depart from, features that other

applications normally have and also it is necessary to have a good, simple, clear and intuitive interface that point users right to what new features bring. In our solution we want to deliver a new framework with new features but maintaining some well known and more than proved basics to be easy and familiar for the users that are having their first contact with our framework. At the same time is also important to offer our experience in smartphones and tablets which represents nowadays a big slice of users across the web. In this thesis also have the goal to support 4K/UHD video, a recent digital video format that takes video quality to the next level with an aspect ratio of 16:9 and a minimum resolution of 3840x2160 pixels. This technology was first seen back in 2003 [1] with the first release of a cinematographic camera launched by Dalsa Origin but it was in 2010 when YouTube started to support 4K technology in their videos that it really started to "come out". YouTube support was fundamental for the spreading of this technology, but it was only four years later, back in 2014, that the first 4k television screen was launched. Nowadays it is getting more mainstream but it is still an expensive technology. However, we believe that in the future with the reduction of production costs and driven by market demand it will become the main technology used for video distribution.

**The objective of this thesis** is to research how to at the same time build a platform to deliver an amount of features that offer a collaborative experience to gather videos from a specific event, at the same time we aim to reduce video redundancy and to gather better contributions. To do it we explore new social experiences where users can share their videos and build video logs of real-world events collaboratively using all source of high-quality videos. Since Video gathering is the main goal of our goal, we want to encourage users to contribute with their videos with the best quality and correctness possible. To achieve that the system we need more than what we have just described and to do so we will explore a few already studied solutions like the reward mechanism by Rokicki et al. [26] to apply to our system.

## 1.2 Proposed Framework

The proposed solution is an on-line platform to be deployed to host User Generated Content of video with support for different devices.

We want to ensure collaboration from different users not only in basic features like video upload, but also adding video annotations and the creation of a timeline for a given event where other platform users can contribute to with their own videos. This is our solution main focus, to have timelines from different events with different videos that were uploaded from different persons to make a more complete timeline for a given event. That is the main reason why we decided to call our application Livetime, because our timelines will tell a story from a given time in life and it can be a story shared by only one user using only their own videos but also a story shared for plenty of others who participated for example in the exactly same event that other user did and who has different videos of it to share.

Apart from that, we also want to offer some social and more basic features like the possibility to comment and rate timelines and videos or to search videos and timelines and find related information to it.

We want the solution to also support different types of video with a special interest to Ultra High Definition video which is now growing, , although we will not treat it directly during our work. We will also host a competition system based on leaderboards to try to keep users interested and challenge them to participate with more quality contributions.

Finally, we also want to have a good Twitter integration. Twitter will be another important element of our work and we want to allow our users to use this application to interact with Livetime.

## 1.3 Challenges

The main and first challenge of a solution like this one is to deliver a new way to users collaborate among themselves and at the same time trying to find the missing piece of the puzzle to offer in our system and which would be different from other already available systems. It is an extremely competitive world with thousands of developments being made each day. Is pretty hard to find a feature that others do not yet have. But there are other challenges. In a collaboration system it is important to maintain coherence in the whole platform. Livetime is a web application and by that it can be accessed by multiple users at the same time, and at that exactly same time they can do a lot of things like editing some video information, commenting or upload a video. By that reason, avoiding data loses and data incoherence is naturally important.

Also since we are talking about video files, we are talking about big data files. It is true that they are being made smaller each day and like it was said in section 1.1 data storage is not that expensive nowadays. It is important to have a simple database that responds fast to its queries so it will not take to much time to access videos which would eventually frustrate the users.

Also important is to have a simple, clear and user friendly interface. Since what we are building is completely direct to users interaction, it is important to have a simple platform where anyone can easily look and understand how to use. At the same time we want to guarantee support to different devices and to retrieve data from different sources.

Finally it is important to have members registered and using Livetime. We want them to be active and to share their videos, to create their timelines and to make an active part of the process of growing our database, preferably with high quality data. With that in mind, we will try to achieve ways to create a competitive environment with a general leaderboard classification system but also giving the users the possibility of hosting their own competitions to engage them in different ways.

## 1.4 Contributions

Throughout this work we made a series of contributions:

**Collaborative storylines** a new way to create storylines with different people contributing to them. We let our users create their own events where everyone can contribute with their videos to gather a bigger amount of user generated content for that same event.

**Reward contributions** a competition system with rewards associated to engage users into making better contributions among with individual competitions for specific events with the goal to fill some event gaps.

**Livetime** the full development of the system that makes all these ideas possible with an experience different from what we can find in other existing platforms with new forms of presenting information and where users can create their events and making their collaborative contributions to grow the information regarding them.

**Short-Paper** was also written but it was not yet published.

## 1.5 Document Organization

The presented document follows the following structure:

**Chapter 1 - Introduction.** In this chapter, we provided an introduction to the problem we are addressing in this thesis and what are our goals.

**Chapter 2 - Related Work.** In this chapter, we discuss concrete examples of Web applications that fall into the scope of collaborative video and identify their main contributions and gaps. The chapter finishes with a critical summary listing the best approaches used by the related work and where novel contributions can be made.

**Chapter 3 - Livetime.** This chapter presents our foundational implementation that will serve as the base system where novel contributions will be implemented, tested and evaluated. We cover the system architecture and also the background of technologies used in the development process.

**Chapter 4 - Building Better Storylines.** This chapter describes the main contribution of our work. We will discuss our timelines, what they represent and how we engage users into making better and more contributions.

**Chapter 5 - User Tests.** In this Chapter, we will provide information from the feedback received by the users interaction with Livetime.

**Chapter 6 - Conclusion.** Finally, this ending chapter will make a conclusion of all the work done and we identify work that could yet be achieved.



## RELATED WORK

In this chapter, we will study a few academical investigations and solutions in section 2.1, and we will also present a few interesting commercial solutions in section 2.2, ending this chapter with a summary in section 2.3.

### 2.1 Collaborative Video Systems

Building a platform for viewing and sharing video content is not something particularly innovative. In fact, there are plenty of services already available like Youtube, DailyMotion or Vimeo and surely more to come in the near future. There is continuous investigation in the multimedia field to keep building new services to help day to day life or to simply find new innovate and fun ways of interacting. That is why to build Livetime we investigated about some of the work already done in the area and what they do. We studied different researchers and ideas from event detection to video annotation to grab different ideas that we could use to make Livetime different from what already exists.

#### 2.1.1 Event Detection In Social-media

Event detection has the goal to detect a specific event using the available data like titles or descriptions and to build a story or to use information from different sources to compose an improved and more complete story. In 2014, Habibian et al. [13] purposed a new format to make video representation, recognition and translation of events which was called VideoStory. This new event recognition algorithm used videos available on Youtube using their title, content and description to provide the semantic interpretation of a video and not only a collection of words used to summarize it. The collection was made available on-line in a storage called VideoStory46k. Our main goal in Livetime is not to develop a

new event detection algorithm, but we will use the idea of comparing information like video titles and genre to find related content within the videos and timelines.

In the work of Liu et al. [21] they tried to solve the problem of the rapidly growing social media data posted on the web. To do it they purposed a new framework that extracts and illustrates social media events automatically for a given query. They use a natural language algorithm which parses the input query to extract related news from a social news web service. With the given query they try to extract some relevant information like event location, topic or time information to search directly through a social news service provided by Digg<sup>1</sup>. After they divide the query and they use Twitter API to retrieve some relevant tweets regarding the topic cleverly filtering the search from the date that the news were posted on Digg to the current date to get more accurate and relevant tweets. They also use Google to retrieve related pictures to illustrate the content. Their big advantage using a services provided by others, in this case Google, is that they save a lot of resources in data storage or processing time. It is another example of how event detection is being used to facilitate the process of present information. Another work regarding this problem was studied by Bian et al. [3] which applied event detection and summarize techniques to answer the quick growth of microblogging services. This type of services are changing the way that information is exchanged and this work introduces a new framework to generate visualized summaries for trending topics. Basically what they do is to analyse the context of the topic and to divide it into sub-topics, then the system uses a set of images to automatically select representative images to illustrate the given topic.

Also interesting and regarding the same problem is the work of Kennedy et al. [15]. In their work, they build a species of a timeline focusing on video concerts. Their goal is to ease the search of concert videos since usually after a concert there will be plenty of videos regarding that same concert across the web. With their work, they gather a collection of videos regarding that concert uploaded by different users, and then they run an algorithm which compares them to find points in common. With that, they can achieve things like reducing the redundancy and they can even go further choosing the best quality videos instead of the poorest ones. With our solution we also aim to reduce video redundancy since the creation of events will help to store in the same place all videos regarding that same event. With that, users can check which videos are already uploaded or not. The automation of this process is a really interesting but that will not be addressed in the context of this work.

### 2.1.2 Events Visualization

Delgado et al. [10] studied similar problems like the ones that we just discussed, but with a bigger focus on visualization and on ways to present information to the users. In their work, they give another example of news interaction with assisted news which tries to

---

<sup>1</sup><http://digg.com/>



Figure 2.1: Example of the interface created by Boreczky [4] (left) and Balabanovic [2] (right)

solve the problem of people's distraction who forget for instance what they just read a few moments ago. With their mechanism they try to illustrate news in a way to facilitate what is being read. The application developed provided an interface where users can select through a series of dialog boxes to fill the required information like news subject and localization. Then a new article will be created and it will be presented like a slideshow presentation where the news sentences will be associated with a picture. To finalize their work, they also have conducted a study to see if users prefer to read news without the help of illustrations or with it and they come to the conclusion that most of them prefer this new system they used. Our timelines will also use a similar system since a timeline will represent an event and will gather different information from different sources to present it into one place. Also they use dialog box and forms to fill the information regarding the news. That is the kind of interface presentation that we will also use.

Newsmap<sup>2</sup> is another great example of how information presentation can be achieved and was developed by Li et al. [19] as a new social-media news browsing system where users can search for the news using a browsing map or retrieving news using a location in the query. The goal was to develop a location-based organized news system. Along with it, they also added a similar system very similar to the one by Delgado et al. [10], since they also associate pictures to their news. To infer about the location they use text processing answering three fundamental questions: Who, Where and What. The answer of this questions help them to find other related news to facilitate the process of discovering news location. That way of visualizing information shows one possibility that we could adopt. In Livetime we will also use approximate locations for videos uploaded. Although, at least for now, we will keep it simple and it is just thought as query or sort filter.

<sup>2</sup><http://newsmap.jp/>

Throughout this chapter, the word timeline already came up a few times, that is not a coincidence since timelines will be the very important in Livetime. In Livetime, Timelines will be a way to tell a story of an event or a memory and will use video to compose it. However, the idea of using multimedia information to build this kind of stories is not new since in 2000, Balabanovic et al. [2] started working with a similar idea using digital photographs. Since technology back then was not as much evolved as nowadays, their work included a device similar to what we call today a tablet but with a few buttons and no touch screen options. Their main focus on developing system's interface was to keep it simple to use and with a very easy learnability. With this device they want to approach two different directions, first the possibility of sharing stories locally in the presence of other persons and after to share it remotely. Their device allows pictures upload which are automatically rearranged by date and can be seen as slideshow presentation with the very important feature that users could record their own voice to narrate the pictures that are currently being presented. So in the first scenario users basically slide through the album and tell the picture story face to face with other persons watching. The second is the possibility to record users own voice into the device and send it to someone elsewhere that could then also see the pictures while listening to the recorded voice. For each story they upload a set of pictures, this pictures can make part of different stories and also other pictures can be uploaded later on by other person. The right picture on the figure 2.1 shows the created device among with a few pictures uploaded into it. Our idea with the timelines is almost the same, but now using videos from many different sources to also build a story/Livetime event. The idea of recording a narrator voice to accompany the slide show is also great, but we think it is unnecessary in our solution since the users can talk directly to their uploaded video.

Continuing in the video visualization, when we go to a web video player we always can see an image representative of the video before we click to start watching it. That image, called thumbnail is absolutely essential to give users a first impression of what he is about to watch and also to catch their attention. The impression given by a thumbnail is sometimes decisive for the user to make the decision if he is going to watch the video or not. So making a good decision during the selection of thumbnails is very important. There are a few algorithms already available to automatically generate thumbnails which can be based on content or user interaction. Leftheriotis et al. [18] purposed a new way to select the best thumbnail for a given video through a new web video player called VideoSkip. This web-player uses the user interaction to chose the correct thumbnail for each video. During their work they also studied user interaction to check which buttons do the users click more often while watching a video, and how it is related to video content. Their web player has Youtube as basis, in fact they use Youtube Api to support almost the entire application with a few additional features. Their main addiction is the addition of a GoBack button. When a user is watching a video he can click GoBack button to go back to the last thirty seconds of the video from their current time. Each second of the video corresponds to an array position, when the button is clicked the

position of this thirty seconds in the array are incremented by one since they consider that the click of the button may correspond to the favourite parts of the video. In the end their algorithm chooses the three seconds/array positions with the highest counter to generate the corresponding thumbnails. As the second part of their work, whenever a button is pressed that interaction is saved to VideoSkip database along with video ID, time and which button was clicked. With this information they came to the conclusion that the most used buttons were play and fast forward while GoBack button was one of the fewer clicked, which does not help the process of generating the best thumbnails based on user preferences. After VideoSkip they continued their work [8], this time not focusing on thumbnails generation but more on the second part of their work that is analysing user activity in a web video player. In this solution called SocialSkip<sup>3</sup> they reduced the number of buttons available, maintaining VideoSkip architecture. The goal was to explore more the relation between video content and user interaction and to allow collaborative contributions to analyse user activity data.

But there are other examples regarding visualization that we found interesting like the work done by Boreczky et al. [4] which uses frames from video to summarize them. The goal is to facilitate the process of finding specific parts in videos, specially when we are before a large set of videos. Their system automatically selects relevant frames from video and combines them into one comic book like presentation. Then the user can select one of that frames, which are different in size (bigger frames are more relevant than small ones), to a video and start playing a video right from that frame. It is a clever approach that allows us to have an idea of what we can find on it before we start actually watching a video. The left image on the figure 2.1 shows a picture summary of a video from the developed system which gives a good idea of what was created and how it was presented.

### 2.1.3 Collaborative Media and Video Annotation

Like it was previously mentioned, collaboration is key in our work. It is very important to have a collaborative system where different users can contribute in different ways to make it evolve and to deliver a better experience overall. Following that idea we decided to study a few systems that use collaboration between users like Vannotea which is a collaborative video indexing, annotation and discussion system for broadband networks developed by Schroeter et al.[27]. The project main goal is to allow real time and collaborative annotations. This solution uses MPEG-2<sup>4</sup> content and users are able to make editions to description templates that are automatically generated for each video in the system. The generated template uses a mechanism to easily convert it to other types of information like MPEG-7. The main challenge but also the goal of this work was to guarantee that every user using the application and annotation would not affect other users work and the system would maintain a correct state for all accesses. But there is

---

<sup>3</sup><http://www.socialskip.org/home>

<sup>4</sup><https://en.wikipedia.org/wiki/MPEG-2>

more work done in the collaborative area for video annotation like the one developed by Kavasisidis et al. [14] called Multimedia Tools Api, a web-based collaborative platform similar to Vannotea but that differs in the way that annotations are built. For each video available, the platform offers a set of utilities through its interface to make a set of annotations using Microsoft technologies. These annotations can be applied directly to a frame and can be used in different ways like a piece of text or drawings. Using the available toolbox users can pick up a pencil or an eraser and start drawing the contours around the frame and select parts to be annotated. One of the examples used during test phase was a video with fish where users had the goal to draw the contour. They also came to the conclusion that contours precision would increment proportionally to the number of users annotating. That is because they use an algorithm with a few math functions like euclidean distance to automatically approximate the best contour for each element of the picture having as base all the contours made by all members for a given video or frame. This also proved that users collaboration can improve data quality.

IBM Efficient Video Annotation (EVA) developed by Volkmer et. al [30] and Informedia Image Classifier developed by Christel et al. [9] are two more examples of applications for video annotation. EVA like Vannotea or Multimedia Tools Api also uses the power of collaboration to annotate a set of videos. A user who is going to make annotations will initially configure a set of parameters to set how much frames he wants to see for each page. Then there will be available four quick label options to label the whole page or individual frames with positive, negative, skip or ignore. The goal is to facilitate the process of making video annotations. EVA is also an available web application optimized for collaborative work. Informedia Image Classifier <sup>5</sup> is different since it is a windows application and by that reason, it works offline and all the users need to have the complete set of data to work, the main advantage is the fact that it does not need Internet connection to be accessed.

All the previous examples regarding collaboration have common points which is video annotation. In our solution we will also make use of simple video annotations, not so elaborated as Multimedia Api Tool for instance, but text annotations to a frame, a set of frames or a specific part of the frame. These annotations will be made collaboratively and will also be made available through the interface. But these are not the only way that users can collaborate among themselves in Livetime, in fact the main source of collaboration will be content uploaded to fill our timelines.

Moving away from video annotation we found a system that uses user crowd-sourced sensing and Twitter to deliver a collaborative system with many applications [11]. Their work is divided in three parts which includes Asktweet, Twitter and Sensweet. Basically Twitter acts as a middleware between all the systems, and users can create a question using Asktweet. They gave an example which is to infer about the current and future weather prevision to other users will respond to it through Sensweet. Sensweet works

---

<sup>5</sup><http://www.informedia.cs.cmu.edu>

using sensors to infer about the conditions in question. Crowd-sourced sensing allows the application to receive information from different users and with that reducing the errors of the results.

#### 2.1.4 Quality Through Engagement

Finally, our solution needs to have users and even more important we want to have users participating and contributing to Livetime with quality to facilitate our goal of data gathering. To do it we decided to study a few ideas to improve users participation like rewarding users, a method studied by Rokicki et al. [26]. This rewards mechanism is a large-scale experimental evaluation and they use different approaches like paying for each task completed, competitive strategies where they use ranks rewarding users in higher positions or just the winner in a "Winner-Takes-It-All" competition. They have also used random scenarios inspired by lotteries where users receive lottery tickets where winners would be drawn. They also studied questions like what information should they make available for instance in a competition which reveals to play a crucial role in keep users interested and how that affects their performance and motivation. For instance, in a ranking system some users will start to gain more points than others. Generally the users with more points are also the users with better or at least most contributions to the systems which will reduce the noise and consequently it will improve data quality. Their work also showed us that users generally enjoy and strive to win well balanced competitions especially when exponential rewards are involved.

Larson et al.[16, 29] are other main investigators in this area. They have been present in many ACM Multimedia Conferences editions building workshops about crowd sourcing with the goal to improve systems data quality using people contributions. So they study how to formulate tasks, but also received input so then they could improve their methods and crowd sourcing techniques that would eventually result in a better quality data contributions made by users.

In multimedia, crowd sourcing is just recently being explored so there is still a lot to grow and explore. Although there are already a few applications and platforms that use this type of rewards systems to make users collaborate among themselves to improve the quality of the media available, like for instance the m-Dvara project [23] that was applied to museums where users can leave their media and comments to the art pieces and new visitors will be able to read that information left by other visitors to get for instance museum recommendations. To develop this platform, they inferred about a few use cases that they wanted to offer and started from there, a system similar to what we use to develop our platform. We have an example from India where the government opened a contest called INVITE [28]. In this contest, students from all over India were engaged in the development of a few applications and interface design of their e-Government platform and then the winner would win a prize. These kind of initiatives are becoming more popular because a single prize cost much less than the development of a whole

system and since participants will be participating in a contest against other users they will try to improve their work as much as they can to win.

The popularity of crowd sourcing gained investigators interest and there are already some work made regarding it, like Blevis et al.[24] who defines Crowd Sourcing as *a new paradigm that relies on the intelligence of "crowds" of people to solve a specific problem or complete a task-sometimes with a monetary reward*. In their work they divided crowdsourcing into multiple sectors as crowdsourcing for enterprises, for academics and for social networks who vary in the type of challenges and motivations that should be presented to the users, since for instance an enterprise crowdsourcing environment should be more about making profit while an academic one is more innovation related. They also define a few roles through different variables to use in crowdsourcing like role-oriented crowdsourcing which is more task oriented, behavior-oriented crowdsourcing which focus more on users reputation and quality, and media-oriented crowdsourcing. Lastly they also define a set of motivation models that goes from social connection, fun, self-value, altruism, learning or profit for instance. In our work our the application of crowdsourcing will be focused on both behavior-oriented and role-oriented techniques and we offer as motivation aspects like a funny system classified by Blevis et al. as personal incentives, social connection classified as social incentives, and also monetary incentives. In Chapter 4 we will discuss more about the ideas adopted like the hosting of competitions to keep users interested and to try to improve their participation's quantity and quality and other forms of engaging to receive better contributions.

## 2.2 Public Solutions

Video sharing is growing every day and it is now becoming more and more mainstream as a way to share information. People from all over the world are now sharing and streaming video to other people, videos of entertainment, news or simply day-to-day life. Allied to this growth are new technologies that facilitate the process like Meerkat, YouTube or Twitter Periscope. We studied some of these technologies to learn more about them and to try to figure out if it can be combined with Livetime. We will present some of those solutions next.

### 2.2.1 Platforms With Video Visualization

#### 2.2.1.1 MeerKat

Meerkat<sup>6</sup> was launched during the year of 2015 and can be found through Google Play or AppStore available for the Android and iOS. It acts as a social network where registered users can live stream video through twitter to all their followers spread all around the globe at the distance of a click of a button by just using their mobile application and

---

<sup>6</sup><https://meerkatapp.co/>



mobile camera [20]. The following picture 2.2 has some screen captures of Meerkat which demonstrates some parts of the interface.

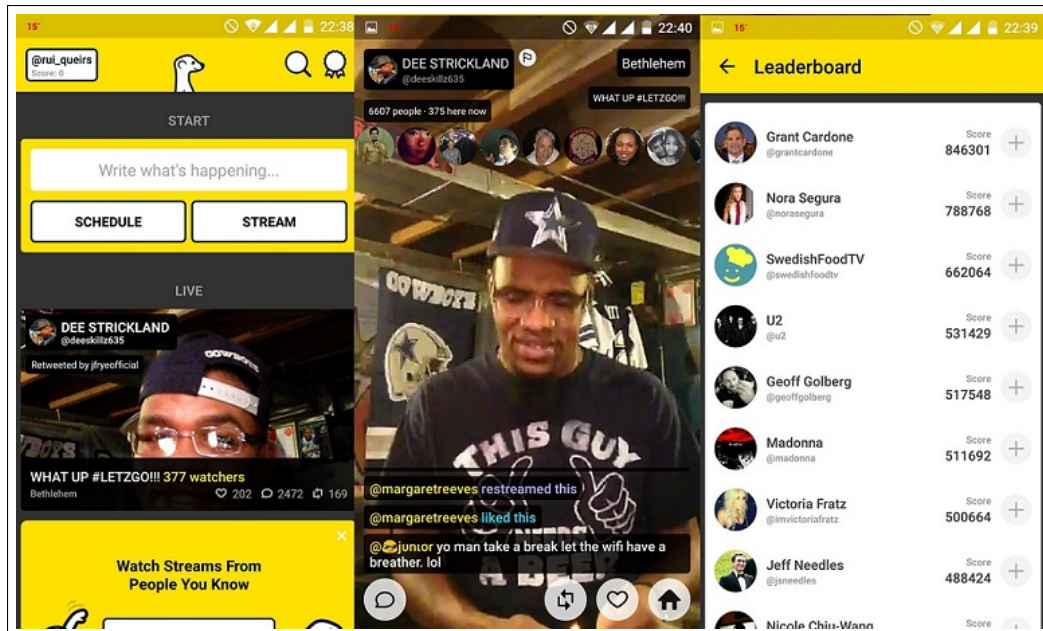


Figure 2.2: A set of screenshots from Meerkat interface

The first screenshot from the left shows the application when we enter it with a couple of information and buttons to users schedule or start their streams or to watch a currently live and trending stream. It is a very simply designed interface with the most important information popping out right at the first contact so users can get used to it in a simple and fast way. The screenshot in the middle shows a live stream. As we can see in the picture, there are plenty of buttons and information on the screen. At the top we can get info about the stream and right below it users who are currently watching it. Then at the bottom we can see comments made live through the click of the button with the balloon form icon, but we can also like or share the stream. The application also offers other features like a leaderboard which we can see in the last screenshot at the right. Streamers gain points not only by streaming videos, but also by having viewers watching their streams. Leaderboards shows the user with most points and also have a few options for us to start watching their streams or following them.

Finally, Meerkat has an API<sup>7</sup> available through JSON format, this API that can be integrated with any system and it is possible to access all the currently live streams, a list of scheduled streams and other information regarding the viewers or comments made. Also, through the Meerkat website it is possible to embed our own stream. We just need to add our user name to the Meerkat Embed page and with a couple of some customizable options, the website will generate an HTML5 code to use in other webpages.

<sup>7</sup><http://developers.meerkatapp.co/api/>

### 2.2.1.2 Periscope

Periscope <sup>8</sup> is another example of a live streaming application. Like Meerkat, it was also developed to Android and iOS and allows users to stream video from their mobile phone or tablet to Twitter instantly. The following picture 2.3 has some screen captures of Periscope.

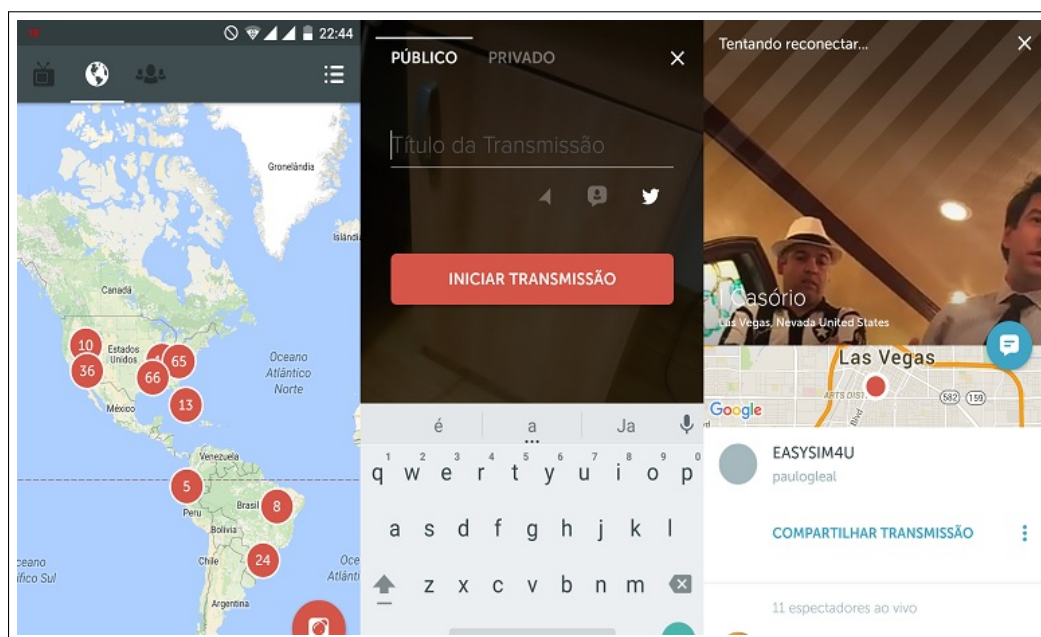


Figure 2.3: A set of Periscope screenshots that shows application interface

The first picture from the left shows the application menu when it starts. As it is possible to see there are four buttons at the top. The first one with television icon is to see the users that we follow that are currently streaming. The icon in the middle send us to the world map that aggregate streamers by location. In this map, we can zoom in and out to start watching their streams. Finally, there is the last icon which gives us trending streams. At the bottom right there is the button to start our own stream which leads us to the picture in the middle in figure 2.3. We can pick a name, choose if it is a public or private stream, if we want to share our currently location, and then we will start streaming live across the world. Periscope, like Meerkat also has plenty of features like liking and commenting a stream. However, they work in a different way since when we are watching a stream, the screen is almost completely clear, which can be good because we have a screen with much less information. But also is a bit harder to share or comment, since we need to return to the previous menu like the one we show on the last picture counting from the left in the figure 2.3. In this picture we can see some more information about the streamer like the exact point from where he is streaming. Other feature that Periscope offers is the easy way share our own stream to our Twitter account which will allow our followers to watch live every move that our camera makes.

<sup>8</sup><https://www.periscope.tv/>

Regarding Periscope API we found a major difference between this one to Meerkat. Like it was presented in 2.2.1.1 Meerkat has a public API<sup>9</sup> available. The API permits developers to have full access to Meerkat features using them to build new products and forms of interaction. Although, there have been some independent advancements around Periscope by Gabriel Girona [25] who is building a Ruby on Rails Client using Periscope, there is not yet available an API for other users to use and integrate in their own applications which can also explain why Meerkat is currently leading by far in the mobile department of live streaming.

### 2.2.1.3 YouTube

YouTube<sup>10</sup> was founded in February 2005 and is currently part of the Google Group. YouTube is also the largest video shared repository available on-line in the whole world with millions and millions of videos from every corner of the planet, sent from many different people and companies and it keeps growing everyday with more and more videos being uploaded to its database. [31]

Since the beginning that Youtube has been evolving to offer their users new ways of sharing their videos, like for instance the creation of channels for each of the YouTube users. That way, YouTube users have their own channel with all their videos. But that is not all, because after that came video annotations which we discussed in section 2.1.3 and that gave to video new information to supplement the one that it already had. Youtube never stops and to remain on top needs to embrace the changes of the world. Changes like the ones we just discussed in sections 2.2.1.1 and 2.2.1.2: streaming. So now, Youtube also has available streaming through its platform, allowing users to start live streaming from their channels to all over the world just like Meerkat and Twitter Periscope. The big difference is their API<sup>11</sup> which can be easily integrated in many different applications thanks to its amazing support which explains all the requests that we can do with it. In the Youtube API, we can also find some links for some API samples<sup>12</sup> in different programming languages like Java, PHP, Python or Javascript. This YouTube Live API is for instance being used with the Playstation 4 system which allows users to live stream their gaming sessions directly to their Youtube channels. But there are many other examples of other companies like the Euro Hockey League or some universities that live stream to Youtube or even Hangouts which can also be used to live stream directly to Youtube. And since we are only in a BETA phase for this new Youtube Service, it is probable that the service will keep growing more and more. Besides the Youtube Live API, Youtube also offers Youtube API<sup>13</sup> which is used for instance in VideoSkip system discussed in section 2.1.2. This API allows the access to Youtube videos and some of its features.

---

<sup>9</sup><http://developers.meerkatapp.co/api/>

<sup>10</sup><https://www.youtube.com/>

<sup>11</sup><https://developers.google.com/youtube/v3/live/getting-started>

<sup>12</sup><https://github.com/youtube/api-samples>

<sup>13</sup><https://developers.google.com/youtube/>

#### 2.2.1.4 Vimeo

Vimeo <sup>14</sup> is another web application for video sharing very similar to Youtube. It is an application where users can upload, share and view videos. The biggest difference is that Vimeo is more restricted than Youtube in the accepted content and it is more focused in high quality videos. In fact, Vimeo was the first video sharing site to support High Definition video back in 2007. The fact that Vimeo has also a smaller and more focused community allows that the contributions made have more quality in general and less uninteresting videos.

We think the solution we want to build and we will present in Chapter 3 is closer to Vimeo, since we will work with a smaller group of members and we want our solution to focus more on high quality content.

#### 2.2.1.5 Facebook

According to Statista [17] Facebook <sup>15</sup> is the most popular existing social network. It has been this way for years and it will probably continue to be for the next years to come. It all started with an application from Harvard University but it quickly spread worldwide. The main reason is because they just did not stop evolving from a very simple platform with easy interaction to a platform where we can find videogames, create groups, chat with other users and share all kind of posts from text to photos to our friends or worldwide. In Facebook we can do almost anything and more recently video is also becoming a big part of the social network since users can now start recording and sharing video directly to their Facebook accounts and even launch live streaming in the social network.

From Facebook we can learn a lot like the importance of interaction simplicity that makes users enjoy using the platform and also that users really like to share all kinds of information across the web, something that our platform will contribute to expanding this type of information sharing and presentation.

#### 2.2.1.6 WhatsApp

WhatsApp <sup>16</sup> is another social network where users can contact with other users individually or in groups. In this application, users can send text, pictures and of course video.

It is really easy to start conversations with other users in WhatsApp and that is probably the main reason why this application is getting so popular. Statistics from Statista [17] shows that it is already the second most popular social network. WhatsApp main goal is to facilitate communication between people and it is mainly used in smartphones.

---

<sup>14</sup><https://vimeo.com/>

<sup>15</sup><https://www.facebook.com/>

<sup>16</sup><https://www.whatsapp.com/>

### 2.2.1.7 Snapchat

Launched in 2009, Snapchat had one main goal: send a picture to another user that would be deleted after read. The idea gained success and now it is not only possible to send pictures, but it also incorporated a very simple chat in screenshot number one from figure 2.4 and also the possibility to send small videos up to ten seconds like we show.

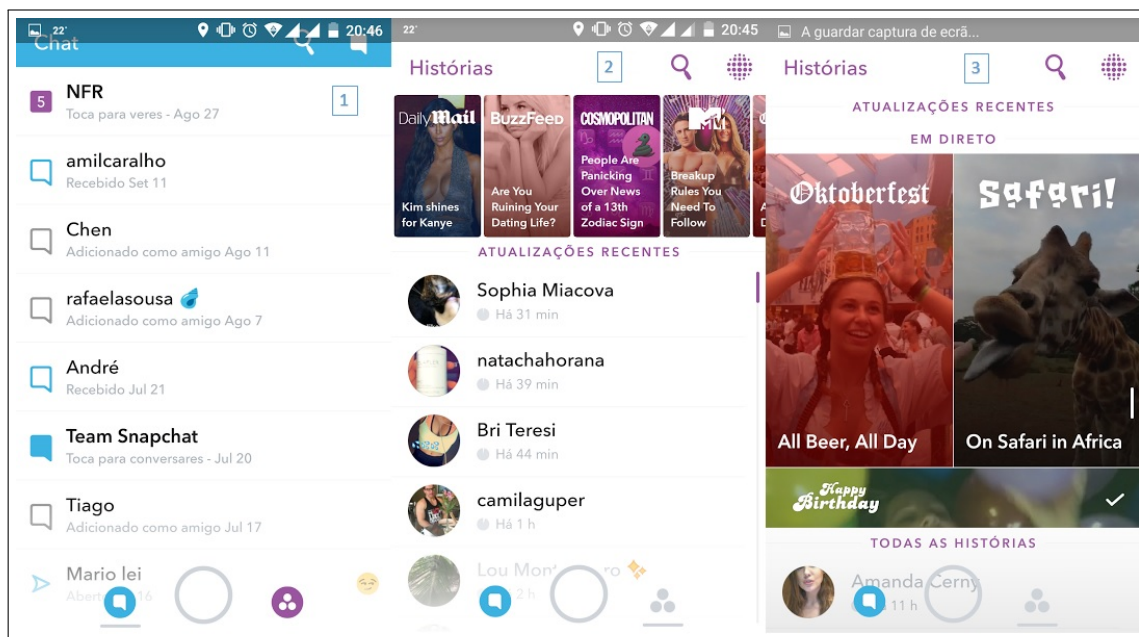


Figure 2.4: A set of Snapchat screenshots that shows application interface

Apart from that, Snapchapp also offers stories that are very similar to the timelines that we use in our application. A story in Snapchat is a collection of snaps (videos and pictures). Each user can create his own story and only he can contribute to it. In picture number two from figure 2.4, we have a set of stories from Snapchat users. In Livetime a story is a collection of videos. The difference remains, apart from the presentation form, in two aspects: In Snapchat each snap that composes a story is only available for twenty four hours, and in Livetime it will be always available, also a user story in Snapchat will only have contributions from the story creator while in Livetime everyone can contribute to.

But Snapchat has another feature which is called Live Stories. This feature allows, like Livetime, users to contribute to the same the Live Stories with their snaps. To do it the Story needs to exist and users need to be in the location of that same Live Story. That way they will contribute with their snaps to it building bigger a story, in screenshot number three from figure 2.4, we have an example of some existing live stories.

### 2.2.1.8 Instagram

Another popular social network is Instagram <sup>17</sup> with the goal of sharing video and photos to other users. The main differentiating feature that Instagram brought was the possibility to add filters to the pictures taken which leads to more artistic photos.

Instagram is a very simple application with very few features but it also conquered the audience. Recently like Snapchat in 2.2.1.7, they also added the possibility to create time limited stories with videos or photos that added a new feature to the application. Apart from that it offers connectivity with others social networks, the possibility to add locations using mobile GPS, to like or to comment the posts made to the social network.

### 2.2.1.9 Comparison Between Solutions

Feature	Meerkat	Periscope	Youtube	Vimeo
Streaming	X	X	X	-
Public API	X	X	X	X
Video Storage	Temporary	-	X	X
Quality Filter	Small	Small	Small	High
Web Application	-	-	X	X
Mobile Support	X	X	X	X
Timelines Support	-	-	-	-

Table 2.1: Comparison between described systems - Part 2/2

Feature	Facebook	WhatsApp	Snapchat	Instagram
Streaming	X	X	-	-
Public API	X	-	-	X
Video Storage	X	X	Temporary	X
Quality Filter	Small	Small	Small	Small
Web Application	X	X	-	X
Mobile Support	X	X	X	X
Timelines Support	X	-	X	X

Table 2.2: Comparison between described systems - Part 1/2

The following tables 2.1 and 2.2 makes a comparison between the systems described in subsection 2.2. This comparison is done between elements that we think that are relevant to compare between all the applications like public Api or video storage and allow us to have a more concrete idea of what they have and do not have to offer. To facilitate the reading process an "X" means that it has the feature and an "-" means that it does not have.

---

<sup>17</sup><https://www.instagram.com/>

### 2.2.2 BBC Have Your Say

BBC <sup>18</sup> is one of the biggest and oldest radio and television public broadcasters of the world, broadcasting for all over the planet. But even companies like BBC are trying to find better ways to reach the audience and to involve users. Internet expanded circumstantially the ways to communicate and like we have previously discussed users want to take an active part of this new types of communication.

With that in mind, BBC News launched a new service called **BBC Have Your Say** <sup>19</sup>, a service that calls for users to contribute with their own stories, news and events. BBC made available a few means of communication through messaging, e-mail or Twitter specialized to **BBC Have Your Say** giving people the power to contribute to the news world. With people's shares BBC can create stories that they were not aware about or they can complete their own stories with information brought from people that were at the events in question.

Similarly to Livetime, BBC launched a service to ask for people collaboration to improve their stories. Clearly, all contributions made by people will be reviewed before being published, something that we will not do.

### 2.2.3 Google Photos



Figure 2.5: A collage example using a selection of photos at Google Photos

Another application that we consider important to discuss is Google Photos <sup>20</sup>. Google Photos is not new, but some of its recent features are, and it is in these new features that

<sup>18</sup><http://www.bbc.com/>

<sup>19</sup>[http://www.bbc.com/news/have\\_your\\_say](http://www.bbc.com/news/have_your_say)

<sup>20</sup><https://photos.google.com/?hl=pt-PT>

we will focus on.

Despite being able to be used as a web application, Google Photos Application main advantage is its mobile application [12] which allows us to upload photos or videos to Google Cloud Storage and to easily create some really great stuff with our own generated content.

Figure 2.5 shows a simple example of what we can do with it. In the mobile application or through the web application we can access our data stored in Google Photos and then we can create a new collage that will automatically generate an image like the one we just presented. But we have more features available since we can create for instance an animation using once again a set of pictures where we can add a few effects. Also we can use the application to build a story using a set of pictures the story will be automatically organized into an album by date. Google Photos also has a feature that automatically creates a new collage or presentation using the photos available in our account and the users does not need to do anything to do it except having a set of content in the gallery.

Despite all of that, the best feature is the possibility of building our own video using that same uploaded data. We just need to select all the videos and photos that we want to add and it will automatically generate a video with the selected set, also automatically adding an adequate song regarding the content used.

## 2.3 Summary

As has been discussed previously, there are already a lot of work and solutions developed in this area. We have studied a very wide set of scenarios like event recognition, multimedia presentation or ways to guarantee quality content. Since we want to build a very wide platform, our goal with these readings was in general to drink a bit of all that ideas to build our own ideas and not necessarily to fix or improve something that is already done.

That is why we also explored some of the already existing solutions not only to see what they offer and how we could use them, but also to try to figure out what we could do different.

The section 2.1.1 helped us with the idea of building events or timelines as we call it in Livetime. Also since there is already a lot of work in event detection, we will use the parsing of natural language and keywords retrieved from videos and timelines title, description and tags to add an automatic system to notify users of existing timelines related to the video that he just uploaded, so that way users can easily add their video to the timeline. Our timeline presentation is very similar to the one that we can find in section 2.1.2 with the work from Balabanovic, only in our scenario we have videos in a web application and not photos in a fixed device. In that section we also learned more about what users use most in a video interface and how to make use of user interaction to improve our system. That brought us the idea of using manual tags to make a Timeline Summary that we will present later on. Finally to improve the quality of users contribution we studied in section 2.1.4 ways to improve their contributions. Because we think it



is important for us to have high quality contributions. Specially because we aim for Ultra High Definition videos, but also because we want to gather good samples of data to our system, we studied some ways to do it that we will integrate in our platform and at the same time that may help to maintain user interest among our system and contributing.

We have explored a few commercial solutions in section 2.2. Our first goal was to try to learn a way to integrate them with our system, but we came to the conclusion that or they do not offer what we are looking for like 2.2.1.1 or 2.2.1.2 or that they will not add that much great value to our solution like 2.2.1.4 or 2.2.1.3 so we just studied them to learn what they offer, what they do and what we can we can do that they already do and what we can do different focusing on features that are not already explored by them. With this study we also come to the conclusion that new applications that are emerging have two main focus, first a good mobile adaptation since applications like 2.2.1.7 and 2.2.1.8 where developed entirely with mobile users in mind, second simplicity. New applications offer one or two features different from what we can find in other applications and then they focus the entire application's interface around that particular and innovative feature.

Finally, we also explored Google Photos referenced in section 2.2.3. It is a very interesting case study, since it uses sets of images or videos to build a new video regarding an event. This idea is very similar to the one we want to build, we will use videos to build timelines using multiple and different videos.



CHAPTER



## LIVETIME

In this chapter we will introduce our solution. First in section 3.1, we discuss about all the requirements using concrete use cases. In the section 3.2, we present and discuss the solution architecture and how the technologies referenced in section 3.3 are used. Finally we have section 3.4 to present the solution.

### 3.1 Requirement Analysis

Livetime is the proposed name to the solution we have built. Our solution main goal is to become a new video visualization service which offers features to create life stories and different ways of interacting. These life stories are called timelines and basically they represent an event in time. Besides that we also offer a set of features from the most basic ones and which we consider essential to an application of this kind, to some other new and more complex features.

#### 3.1.1 Use Cases

In the following subsections, we will make use of some use cases to extract Livetime requirements. These are requirements that we have already available in Livetime and some that are thought for the future.

##### 3.1.1.1 Use Case 1: New User Creates Account, Creates Channel, Uploads Video and Adds Video Tag

**Primary actor:** User, Livetime

Basic Flow:

1. A new user arrives at Livetime and chooses Create Account option.

2. The user creates the account. User will be automatically logged in
3. Livetime stores the new account information in the data store
4. User creates video channel
5. User fills the form to create his channel and the channel is created
6. Livetime stores the new channel information in the data store
7. User uploads a new video to Livetime
8. User fills the video form with all the information and uploads it to Livetime
9. Livetime stores the new video in Livetime store
10. The user goes to an uploaded video location and starts watching the video
11. At a given time, the user manually adds a new tag to the video associated with the current video time
12. Livetime stores the new video tag information in the data store

**Requirements** With the basic flow described in this section we can find a few requirements that are available in Livetime:

**Create Account** Users have the possibility to create their own accounts. For that we will ask through the filling of a form for some basic information like e-mail and password to make authentication, birth date, name and country of origin. It is required to have an account to access some of Livetime features like directing uploading videos, comment or participate in the timelines.

**Create Channel** To upload their videos to Livetime a user needs to create his channel. This option will be made available the first time that a user tries to upload a video. The channel creation is very simple and again it is through the filling of a form where the user can choose the channel name and also give it a description. Channel and videos are connected since all users have their own channel with their videos.

**Upload Video** Every registered user also has the possibility to make a direct upload of a new video to Livetime. Once again, to do it we present a form that needs to be filled with some information like title, description, genre selection, date, a set of tags and of course the video file location.

**Video Tag** Users also have the opportunity to add new and relevant tags to videos. When watching a video a user can add a new video tag to some part of the video, for instance in a music video add tag to when a guitar solo starts. Then that tag can make the video jumping right to it.

### 3.1.1.2 Use Case 2: A User Creates a Timeline and Adds Videos To It, Another User Uploads a Video And Also Adds It to the Created Timeline

**Primary actor:** User A, User B, Livetime

Basic Flow:

1. User A decides to create a new timeline in Livetime.
2. User A then adds videos from his channel to the newly created timeline
3. Livetime stores all the information regarding the timeline
4. User B enters Livetime and decides to upload a video
5. Livetime checks its database to see if there are already existing related timelines
6. Livetime suggests the User B to add his new video to the timeline created by User A
7. User B adds his video to the timeline
8. Livetime stores the new video in the User's A timeline
9. User's A timeline now has videos from User A and User B

**Requirements** With the basic flow described in this section, we find another set of requirements available in Livetime:

**Create timeline** The user have the opportunity to create their own timeline. This timelines will have a genre like sports, education, politics for instance and can be created through the filling of a form. Every timeline will also have a set of tags associated with it and also an hashtag.

**Collaborative timeline** Users can not only create timelines but they can also help them build it. Imagine the following scenario: User A went to a concert of the band X and recorded three songs which after uploaded to his Livetime account and created a new timeline with the songs he recorded. But Livetime has also another user, User B who also went to the same concert and recorded three different videos and also upload them to his Livetime account. User B will also have the opportunity to share his videos to timeline created by User A and now that timeline will not have just three concert videos from User A, but six from both User A and B which will create a bigger and more complete timeline. This can be achieved using Livetime.

**Timeline event detection** When a video is uploaded there is an automatic search made by Livetime to find similar timelines. If there are similar timelines found, the uploader has the opportunity to add the video to the existing timeline.

### 3.1.1.3 Use Case 3: A User Will Search, Comment and Rate a Timeline and a Video

**Primary actor:** User A, Livetime

Basic Flow:

1. User A searches for a timeline
2. User A gives timeline a classification
3. User A also comments the timeline
4. Livetime stores new timeline classification and commentary
5. User A searches for a video
6. User A gives video a classification
7. User A also comments the video
8. Livetime stores new video classification and commentary

**Requirements** With the basic flow described in this section we can find another set of available features:

**Commentary sections** For each video and timeline is available a commentary section. In this section users can leave a text comment that will be associated with their account and which everyone will be able to read. Only registered users have access to this possibility.

**Rate timeline and videos** For each video and timeline, a rating section is available . In this section users can rate the timeline or the video with a 1 to 5 star classification. This information will be stored in Livetime application.

**Search Video and Timeline** Users can search videos and timelines using keywords. After the search, it will be retrieved a list of related videos and timelines.

### 3.1.1.4 Use Case 4: A User Searches a Timeline and Opens a Contest, Another User Uploads a Video to Contest Using Twitter

**Primary actor:** User A, Livetime, User B, Twitter

Basic Flow:

1. User A searches for a timeline
2. User A opens a contest for an already existing timeline
3. Livetime stores new contest information
4. User B tweets with a new video to Twitter

5. Twitter stores the tweet
6. Livetime grabs the tweet and stores it in the contest

**Requirements** With the basic flow described above, we present another set of requirements:

**Contests** Contests are a very important feature of Livetime. Every user has the possibility to open a contest. In this scenario, imagine the following: User A arrives to a searched timeline, he finds that there are a part of the timeline that has very low contributions and he decides he wants more, so he opens a contest for that timeline portion. Basically, a contest is just another timeline with a reward system. Much more about it will be discussed in chapter 4.

**Twitter Upload** This is one of the features that uses Twitter that we have available in Livetime. Like the previous requirement, it will be much more detailed in chapter 4, for now we leave the basic idea: Since each timeline as an hashtag associated, when a user Tweets using that hashtag Livetime uses the Twitter API referenced in subsection 3.3.5 to grab the video and add it to the timeline.

### 3.1.2 Other Requirements

**Related videos and Timeline** When users are inside a timeline or a video, they can see a set of related timelines and videos.

**Leaderboards** Solution has leaderboards available which are divided in different categories like top video uploaders, top timeline creators, best rated timelines and videos and a combination of all of them. They are also absolutely essential to the contests available since they will determine the winners.

**Timeline Presentation** There are two ways to present timelines. The first form works like a slideshow where videos will be disposed horizontally. The other is vertical with all videos that compose a timeline will be shown at the same time.

### 3.1.3 Future Features

The following requirements have been thought but they are not yet available we will present them but they will be more discussed in Chapter 6 where Future Work will be approached.

**Add Video Portion to Timeline** Users can choose between uploading a complete video into the timeline or just a portion of it. When adding a video to a timeline a user can choose to add just a time interval of the video.

**Timeline Intelligence** Timelines will also have some intelligence on its data. Imagine again a scenario where Livetime User A goes to a concert of the band X and he records three videos and starts a timeline for that concert. With the timeline name and description plus video name, description and tags, we will have plenty of information regarding the timeline. Imagine now that User B wants to add videos to the timeline, our application will also be able to compare the new video information to the current timeline and its videos information to make a validation to check if the video makes or it does not make sense to the existing timeline. If a user adds the video to the timeline, but Livetime does not think the video is appropriated to the Livetime in question, the user will be alerted to remove it and will also alert the timeline creator.

**Timeline Summary** For each timeline we will also provide a video summary of all the videos that compose it. This video summary will be generated automatically.

**Closed, Protected and Open Timeline categories** Timeline categories are responsible to add restriction to timeline. Each timeline will be divided into one of these categories that will be chosen by timeline creator. Closed category is a the most restricted one and will only allow users with access from the creator to upload videos to it. Protected category which will allow everyone to upload their videos to the timeline but only its creator will be able to accept them. Finally there will be an Open category which will allow every user to edit and add new videos directly into the timeline.

## 3.2 Architecture

The proposed solution consists of a web application based in a client-server architecture which uses Bootstrap for the client presentation and Ruby on Rails and its MVC (Model, View, Controller) pattern.

This pattern separates data representation from user interaction and consists in three components:

**Model** is responsible for data storage and it has PostgreSQL as its bases. It notifies all associated views and controllers when a state change occurs, so that way we will always have up to date data in our storage.

**View** is the layer responsible to generate results to clients through model solicitations

**Controller** is responsible to send model commands to update data and also to the view to change presented information.

Based in this pattern we have built our application architecture scheme which is presented in figure 3.1.



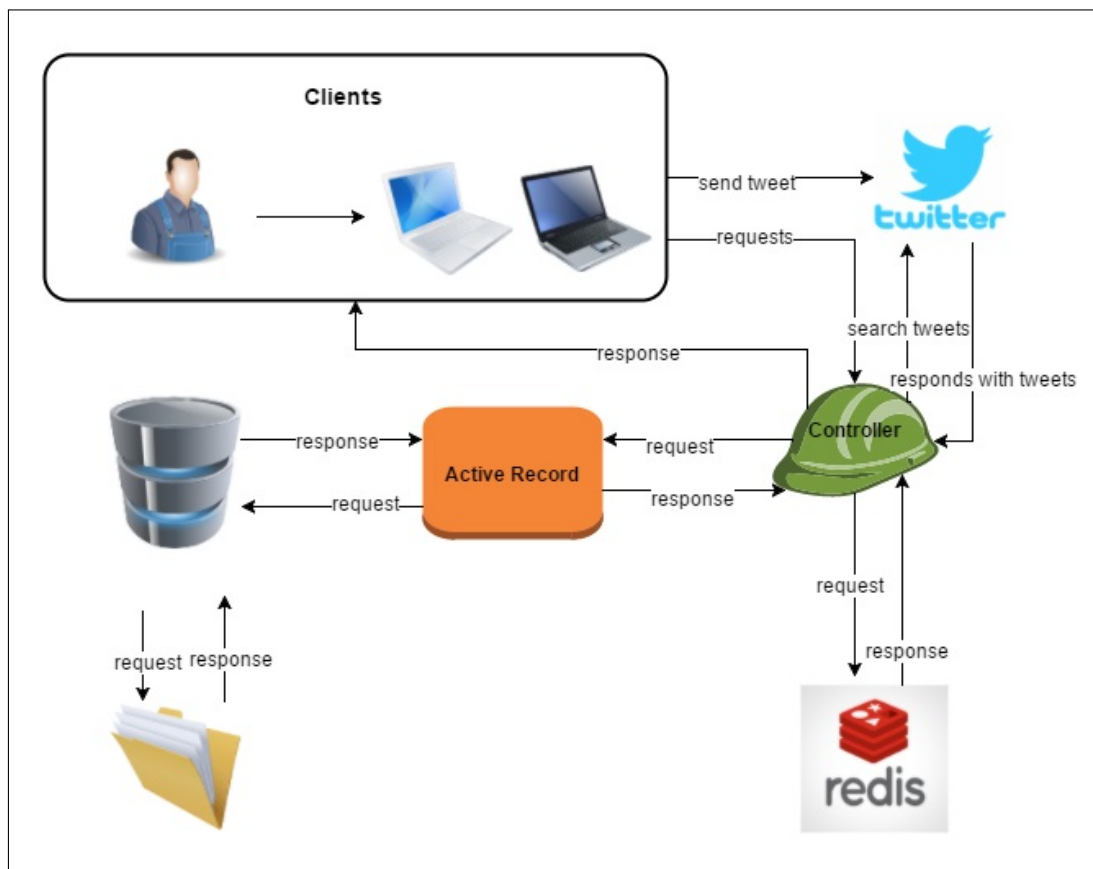


Figure 3.1: Livetime architecture model

Our model is basically divided into seven components. In figure 3.1 we can find all of them. First we have the clients. The clients are the users that will interact with Livetime application using their devices like mobile phones, computers or laptops. It is there where the users are able to perform all the functions referenced in the previous section 3.1 like create an account, upload or watch a video, create a timeline, watch a timeline, comment or rate.

Bootstrap described in the section 3.3.1 is the technology chosen to build user interface. That way, we hope to deliver the users a modern looking and dynamic application which can be used in every device like if it was native.

Ruby on Rails referenced in section 3.3.2 uses MVC pattern and it is responsible to link all the layers in our architecture making sure that not only the data shown is consistent and up to date but also that it is context based. For instance, we want who is logged in to have some features that users that are not logged in do not have, like the possibility to create a channel, timelines, comment or rate videos. These are some of the features that only registered and logged users have access to. It is also Ruby on Rails that allow us to make sure that sessions are respected so that way, a user will not see his information violated. Of course it is not everything that Ruby on Rails does. As it is possible to see in the figure there are controllers and an Active Record, two more of

the seven components spoken about this architecture and that Ruby on Rails brings to us. Controllers are responsible to make the first connection to user interaction and the rest of the application. All user requests are made to the controllers and the controller will redirect them to the Active Record. The Active Record is then responsible to retrieve or to store the data in the database like a simple request reply system. Active record facilitates the communication between our controller and database since it automatically adds features as insert, update or remove new data with proprieties that correspond to the database table.

Our database is another one of the seven architecture components. It uses PostgreSQL referenced in section 3.3.3 to store data information. That data information includes user accounts information, channels information, videos information and basically every information necessary to fulfil our system requirements and also the videos location. Videos will not be stored in our database but in a file system, another of the seven components.

Finally there are two other components. First the Redis Server discussed in section 3.3.6. Our controller is responsible to send notifications to the Redis Server everytime we want to make a change to the rankings. Controller is also responsible to retrieve data from the Redis Server to show clients the rankings. Second there is Twitter. Clients can interact directly with Twitter. Clients can post their tweets to Twitter and our application is responsible to gather Twitter information to store in our database to be shown to every client that uses Livetime.

### 3.3 Background Technologies

The following subsections introduce the used technologies in the development of our solution applied to our system components.

#### 3.3.1 User Interface

The User Interface is a key point in a web based application. Having a clean, simple and user-friendly interface is essential and nowadays, since the growing of mobile devices in the web, building a responsive web page is also a top priority to facilitate mobile users interaction. To do it, there are a few guidelines to be followed which can be applied for instance with pure HTML5 technology. Fortunately and to facilitate this developing process, there are already plenty of open-source frameworks like Zebra <sup>1</sup>, Skeleton <sup>2</sup>, Foundation <sup>3</sup> and Twitter Bootstrap <sup>4</sup>.

We decided to use Twitter Bootstrap because of previously experience with the framework and also because it is a very popular framework which can be easily adapted to many

---

<sup>1</sup><http://zebra.gravitysoft.org/>

<sup>2</sup><http://getskeleton.com/>

<sup>3</sup><http://foundation.zurb.com/>

<sup>4</sup><http://getbootstrap.com/>

different technologies and devices. Also, there are plenty of examples of how to use it, examples available on the web created by other users with a very strong documentation.

Twitter Bootstrap is used for developing HTML5 applications. It uses HTML, the standard markup language used to create web pages, CSS used to format HTML web pages and Javascript Library jQuery designed to simplify client-side scripting of HTML. The combination of these technologies is responsible for Livetime presentation.

#### 3.3.2 Web Development Framework

Another component of our system is the back end. Back end is responsible for the possibility of creating members, uploading videos or creating timelines that will be shared and stored in the application. Like in User Interface there are a lot of available options like Spring <sup>5</sup>, .NET <sup>6</sup>, Django <sup>7</sup> or Ruby on Rails <sup>8</sup>.

We decided to go for Ruby on Rails, again because of previous experience and also because of its MVC (Model, View, Controller) integration that facilitates a lot of work. Ruby on Rails is a fast growing technology with multiple distributions already developed. Its success is not only due to the use of MVC but also because of its active record pattern which offers basic features to add, update or delete data from relational databases easily.

Apart from that, Ruby on Rails is an open source framework with a very good community and plenty of gems (add-ons build by other members) that we can easily use, install and integrate with our application, which can make us avoid plenty of unnecessary work.

#### 3.3.3 Database

Regarding persistence, since we have chosen Ruby on Rails to implement our Back End, and since Ruby on Rails already has a native support for many database management systems like MySQL <sup>9</sup>, SQLite <sup>10</sup>, SQL Server <sup>11</sup> or PostgreSQL <sup>12</sup>, to develop our project we decided for PostgreSQL. Again because PostgreSQL is an open-source technology but at the same time there are already plenty of extensions available to use with it that allows us to grow in different directions, if we choose to in the future. Also, PostgreSQL is a very good balanced database service which responds fast, so it can help with Livetime processing times.

---

<sup>5</sup><https://spring.io/>

<sup>6</sup><http://www.microsoft.com/net>

<sup>7</sup><https://www.djangoproject.com/>

<sup>8</sup><http://rubyonrails.org/>

<sup>9</sup><https://www.mysql.com/>

<sup>10</sup><https://www.sqlite.org/>

<sup>11</sup><http://www.microsoft.com/pt-pt/server-cloud/products/sql-server/>

<sup>12</sup><http://www.postgresql.org/>

### 3.3.4 Video uploads

Video storage is one of our biggest problems, since videos take up a lot of space, we need plenty of storage space available. Throughout the investigation period we have studied many different cloud storage systems like Dropbox <sup>13</sup> or Mega <sup>14</sup> and we made an actual move to use Google Drive<sup>15</sup>. We decided to put that idea on pause for now since we had a few compatibilities problems with the existing Drive gems, and the Ruby on Rails version used (gems can be very helpful in Ruby on Rails but sometimes finding the right versions can be hard). So for now we are using a local database to store all videos uploaded directly to the platform.

### 3.3.5 Twitter API

Apart for all the technologies referenced, we are also working with Twitter API <sup>16</sup> which is used in many different ways that will be presented and discussed later in our work, so for now here is a quick briefing:

Twitter is a really big social network with millions of tweets being made each day [22] and with millions of users familiar with it. We wanted to use the power of Twitter in our application so we have studied its API to try to find the best way to use it. One of the biggest features that we added using this Twitter API is the possibility to contribute to Livetime uploading videos directly to Twitter and not to our application. With this, we can not only reduce the size of the storage needed to store the uploaded videos, but we also give our users a very easy way to contribute to Livetime.

### 3.3.6 Leaderboards

Leaderboards are one important feature in Livetime since it is with them that we make users compete against each other. Since there are already a few servers developed that help us in the building of this rankings, and since using them would be much more efficient and faster than building our own, we made an investigation to decide which one to use and we picked Redis <sup>17</sup>.

Redis is an open source data structure server with an excellent performance and with very good compatibility with Ruby on Rails having all the necessary features required to the development of our leaderboards.

---

<sup>13</sup><https://www.dropbox.com/home>

<sup>14</sup><https://mega.nz/>

<sup>15</sup><http://drive.google.com/>

<sup>16</sup><https://dev.twitter.com/>

<sup>17</sup><http://redis.io/>

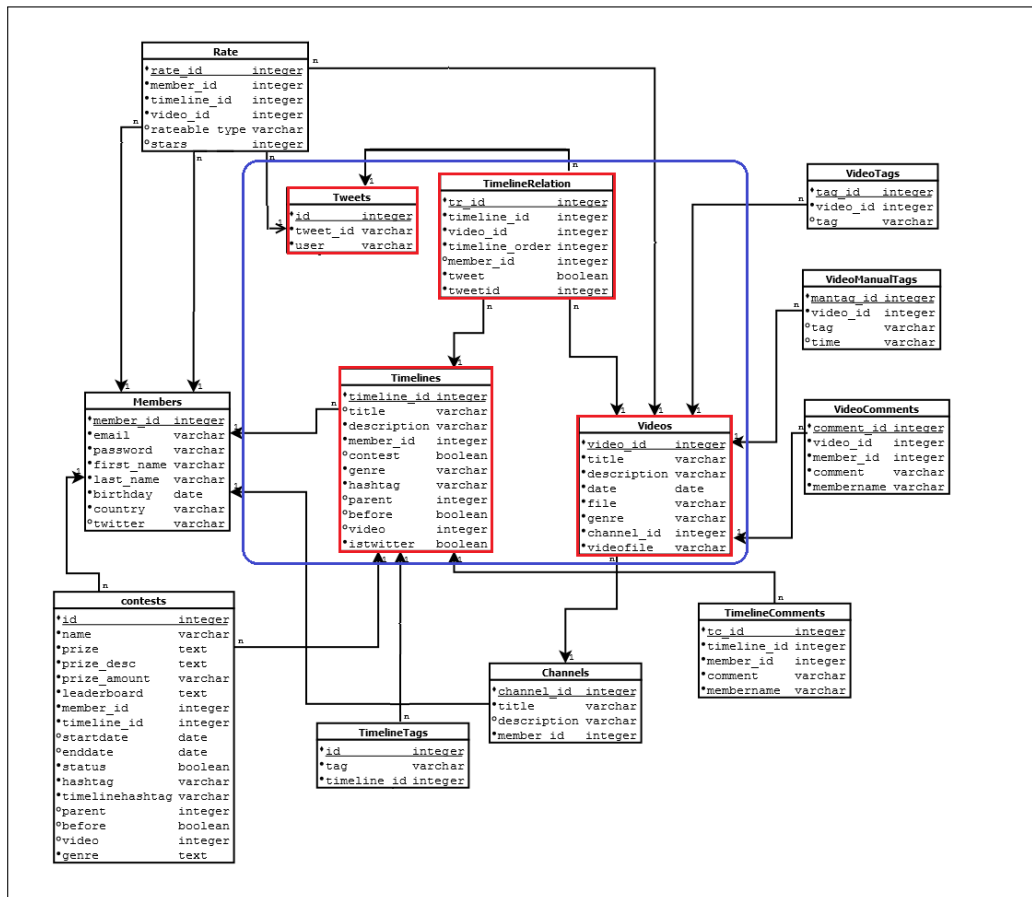


Figure 3.2: Current domain model already developed for Livetime application

## 3.4 Development

In this section we detail more about the development process and features available in Livetime application. We will present Livetime domain model, some Livetime features and interface screenshots.

### 3.4.1 Domain Model

The domain model represented in figure 3.2 is the current model for Livetime application. In the left side, we have Members. Members are essential to the whole system since they are the key to the collaborative experience that Livetime offers. As we can see in the figure, Members have almost a relationship to every other entity available in our system. Each member has its own channel and can have a multiple collection of videos. They also can create multiple timelines and contests and make multiple comments to Videos or Timelines. There is also an association between rating and members and rating and Tweets, Videos and Timelines. This means that every member can rate any video, any tweet or any timeline.

In Livetime, we also have a few entities responsible to store tags. Timelines and

Videos have their own tags that work as key values for features like search. Associated with videos there is another entity called `VideoManualTags`. This is the entity responsible to store the tags that are added by any user to a given time of the video. We will detail more about this in the next subsection 3.4.2.

At the center of the figure you can also see an entity called `TimelineRelation`. This entity is responsible to associate a created timeline to all the videos or tweets that belong to it. Also there is the `contest` entity which is an extension to the timeline. It has all the features that timeline has and a few more added. When we create a contest we create a timeline.

The figure also has a rectangle aggregating four entities, this happens because they are the main focus of our work. Videos, Tweets, Timelines and the relation between them all. Timelines are the events that we have in Livetime while the other entities are the contributions made by users to fill those events with information provided by them.

### 3.4.2 Application

In this subsection, we present some Livetime screenshots along with their description associated with the Livetime features already available. Like we described in section 3.3, our application is developed using Ruby on Rails framework, a PostgreSQL server as database and HTML5 with jQuery.

#### 3.4.2.1 Workflow

Before going into specific Livetime details, we will show a small preview of Livetime. In picture 3.3 we show some examples of Livetime interaction and although we skipped a few Livetime features throughout this workflow, like rating, leaderboards, timeline and contest creation, account registration and Twitter contributions. Most of these features we will be detailed in the following sections and chapters so for now we just want to give a quick hint of what we developed.

To begin, in the first picture (marked with red number 1) we are already logged in and we want to upload a new video, to do it we click the option marked by number 1 inside the square on the menu bar. We will then move on to the picture that we have a page (red number 3) to add the new video **Lifetime Presentation 1**. After we click the button marked with the number 2. Inside the square we will start video upload and we will move on to picture where we can see the Livetime video page (red number 3) for the video that was just added. Then we decided to click on the element marked with number 3 to go the list of Livetime available timelines (red number 4) and we select the timeline marked with number 4 **Lifetime Presentation 2**. When we reach the timeline (red number 5) we want to add a new video, and to do it we click the button marked with number 5 **Add Video** which will move the application to the logged user video channel (red number 6) with all his videos. We selected the video that we have just added clicking the button marked with number 6 and then we will jump back to the previous page of the timeline

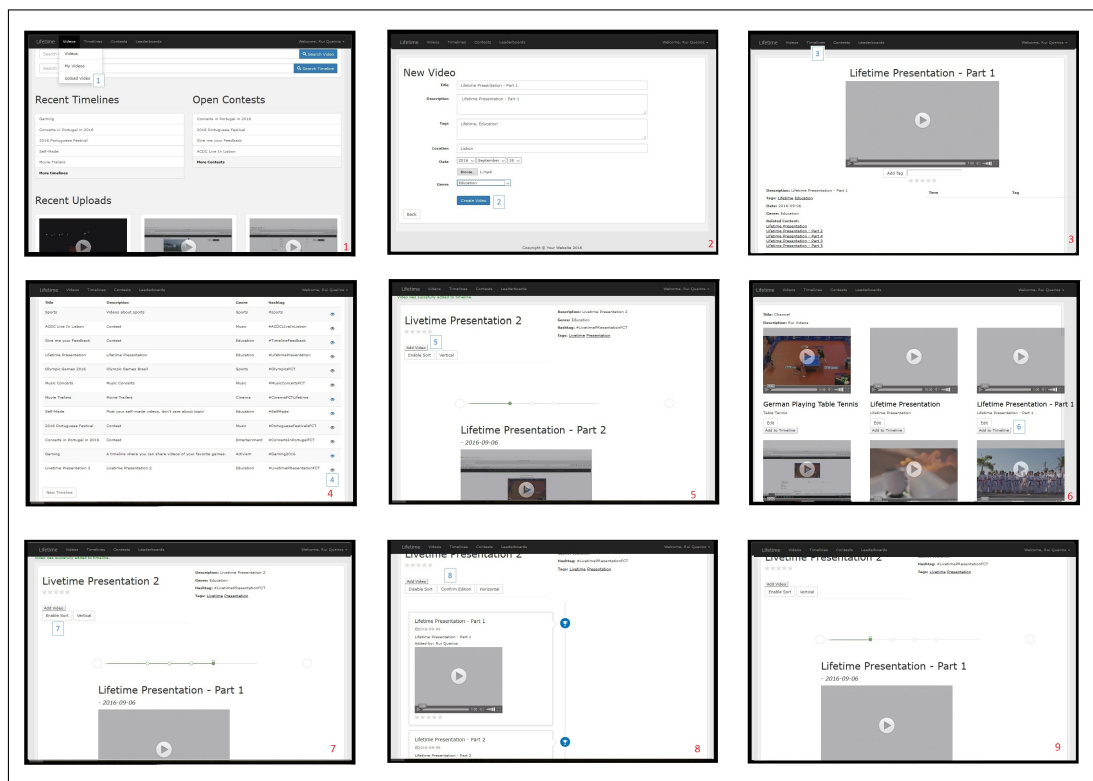


Figure 3.3: An Example of A Sequence of Actions in Livetime Application

(red number 7) with the new video already added to the last position of it. We will then click **Sort Timeline** button marked with number 7 to change videos order manually. We will move to the vertical disposition of timeline presentation (red number 8) where we can drag and drop users contributions to the timeline switching their position. After we will click the **Confirm Edition** button marked with number 8 to confirm new timeline order. In the last picture of this storyboard (red number 9) we can see that a new order is already available with the video that we just added **Lifetime Presentation 1** moved to the first position.

As we mentioned in the beginning of this section, these are just some of the actions available in our application and the goal was to show a sequence of actions that we can do in it and to have a small preview of what we can find. In the follow sections we will go into more detail about Livetime features.

### 3.4.2.2 Presentation

To start Livetime presentation we have figure 3.4. In this figure we can see the homepage of Livetime, it is in here where everything starts. In the figure you can see numbers from one to five. We will use it throughout the presentation to facilitate the reading process when we are referring to some page parts. So to begin, the squares marked with number 1, 2 and 3 corresponds to recent creations, for instance, number 1 shows a table where we can see the last five created timelines and we have a link to check the other created

timelines. Number 2 is the same, but this time it refers to contests. In the mark number 3 is where is possible to see the most recent video uploads. In number 4, we have the login system. In this work we skipped the pages with the forms to login or register an account since they are similar to other pages login systems and it is not that important in the context of our work. Still to implement the login system we used a Ruby on Rails gem called Devise<sup>18</sup>. The login section varies from situation to situation, since a not logged in User will have the option to create an account or to Login. When the user is logged in he can logout or edit his profile, to do it we offer a dropdown list. Actually all the content available in the black bar with number 5 are all elements with dropdowns list associated, we can see the example of the dropdown associated with timelines. Close to number 5 it is possible to see elements such as Videos, Timelines, Contests or Leaderboards. There, users can go to the upload video section, check Livetime videos, create new contests, timelines or checking Livetime leaderboards. All the elements are clickable and offer different options.

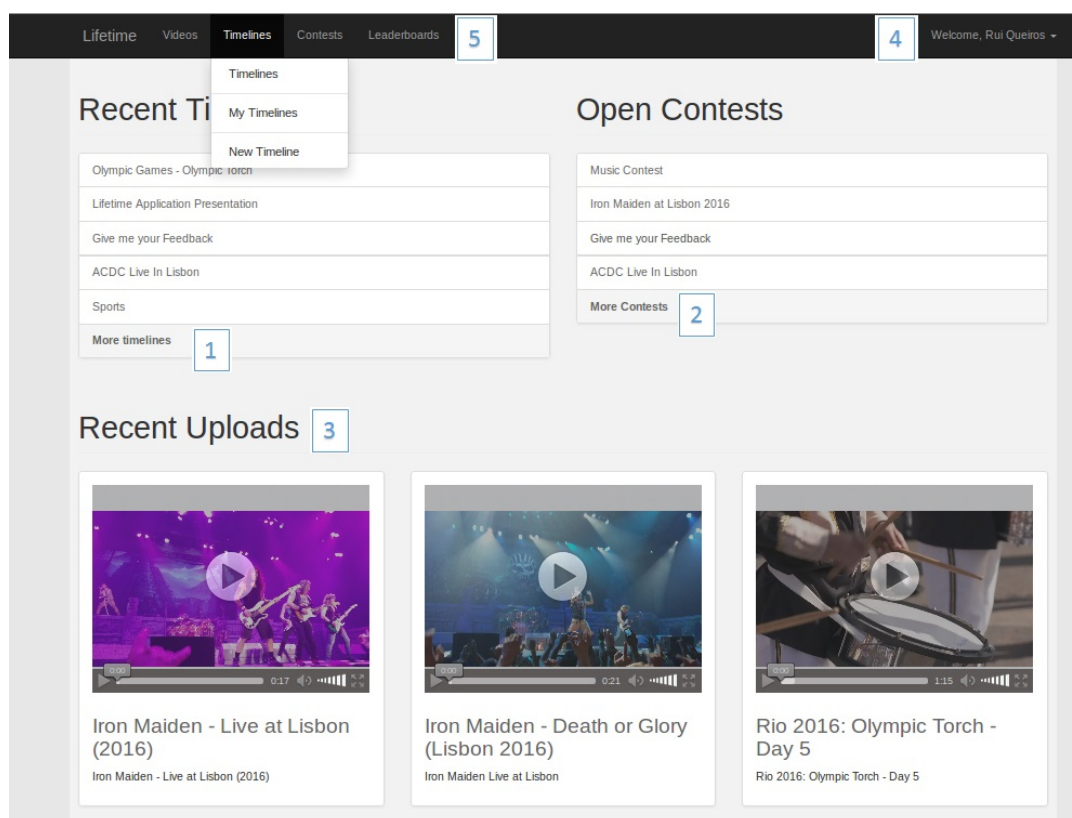


Figure 3.4: Livetime Homepage

In figure 3.5, we present the page that corresponds to the upload video feature. When a user wants to upload a video he needs to write a title, give it a description, add an amount of tags, a video location, a date, the video file and a genre. In figure 3.6, we present the page of an already uploaded video. As it is possible to see in the video page

<sup>18</sup><https://github.com/plataformatec/devise>



there is information added during the video upload and a few more things like in the mark number 1 where it possible to see the current video rating. Ratings are available in videos and timelines and were added using Letsrate Rating Gem <sup>19</sup> a jQuery plugin for Ruby on Rails developed by Murat Guzel. Bellow the video and above the rating you can see an Add Tag option. Basically users can add tags to any part of the video writing the context in the box and then clicking Add Tag button. The created tag will be added to the table associated with the mark number 2 in the figure, the table has the tag time and the tag content. Finally regarding this figure we have marks 3 and 4 which we can see related content both for videos and timelines. Tags added to the video, video genre and title keywords are responsible for the search.

The screenshot shows a video upload form with the following fields and values:

- Title:** Rio 2016: Olympic Torch - Day 1
- Description:** Rio 2016: Olympic Torch - Day 1
- Tags:** Olympics, Rio 2016
- Location:** Brasil
- Date:** 2016, May, 5
- File:** Dia 1 - Revezamento da Tocha Olimpica Rio 2016[via torchbrowser.com].mp4
- Genre:** Sports
- Buttons:** Uploading, Back

Figure 3.5: Livetime Page For Video Upload

To find related videos we start to search inside our database by genre and we find all the videos corresponding to the genre in question, then we count similar tags of each video and finally similar keywords. For each match the value of the found content is incremented by one. In the end, the content will be sorted by counter from maximum to minimum and the top three will be added to the related content. Related timelines work the same way. This system is basically a combination of Liu et al. [21] and Leftheriotis et al. [18] work referenced in chapter 2 since we use a natural language search [21] and also an array that is incremented every time we find a match [18] to make the most appropriated selection.

In figure 3.7, we have the representation of the commentary section which is available not only in the videos but also in timelines and contests. A commentary section is not that important for what we want to achieve with our work, but we think that is very important to involve the users within the system and the content giving them the opportunity to

<sup>19</sup><https://github.com/muratguzel/letsrate>

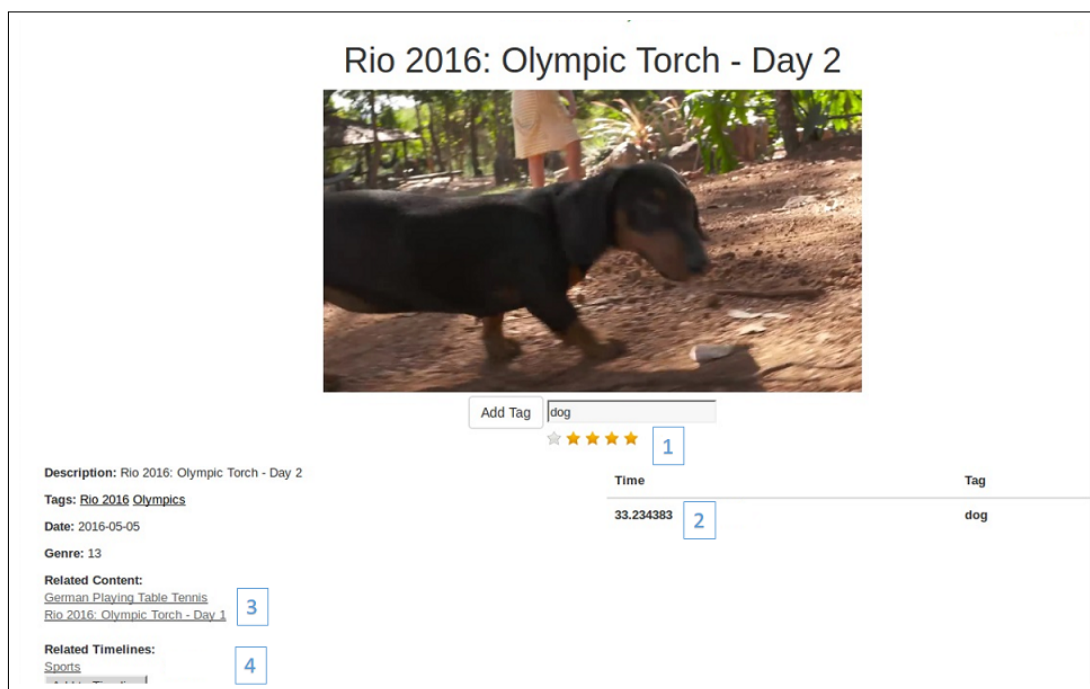


Figure 3.6: Video Page in Livetime Application

communicate not only with the creators but also with other users in a way to share ideas or similar stuff. To leave a comment is very simple, users just need to be logged in and then write a comment.

In figure 3.8, we have a listing of Timelines with all the timelines current available in the system. All our listings, except videos listing where we show video thumbnails, looks like this one. So if a user does a search, if a user wants to check his timelines, all Livetime timelines, contests and similar stuff will be presented in a page similar to the number two in the figure.

These are not the only features that we have available in Livetime, we have many more to present and to show, but we will leave them for the next chapter. The reason is because the features that we are missing to present are features that are more specific to what we want to achieve with our work instead of features that can easily be found in other existing applications. So in the next section we will focus more on our Timelines, Contests, our Twitter integration and of course our crowdsourcing collaboration.

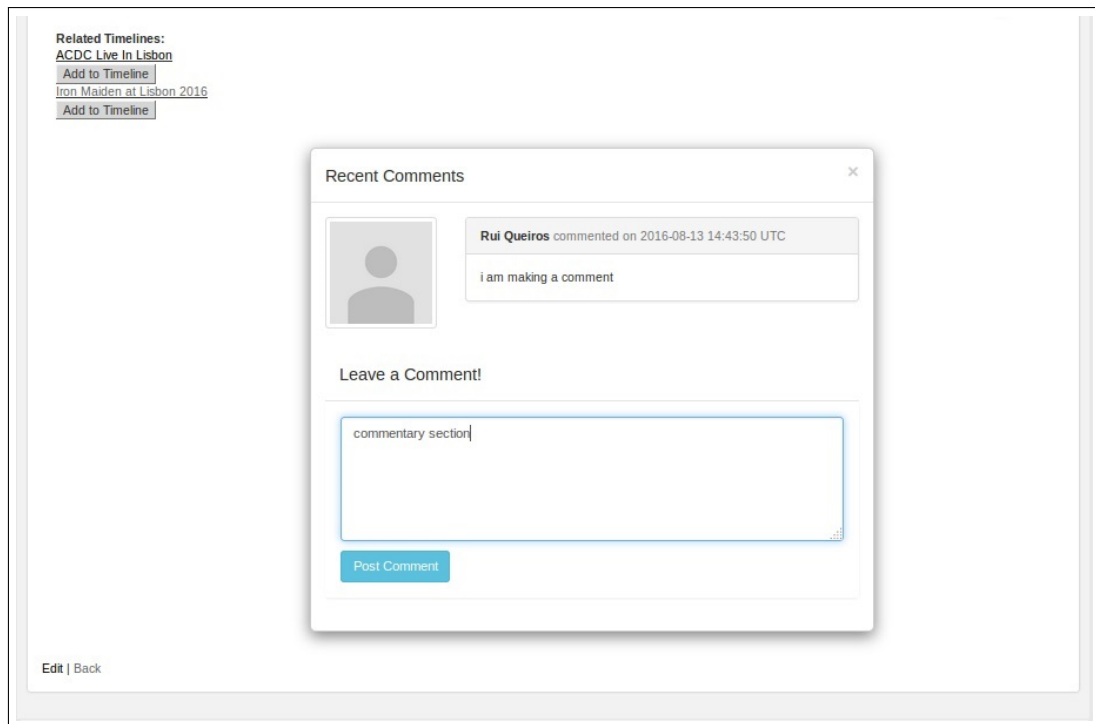


Figure 3.7: Livetime Commentary Section

### Listing Timelines

Title	Description	Genre	Hashtag
Sports	Videos about sports	13	#sports
ACDC Live In Lisbon	Contest	8	#ACDCLiveInLisbon
Give me your Feedback	Contest	5	#TimelineFeedback
Lifetime Application Presentation	Lifetime Application Presentation	5	#LifetimePresentation
Olympic Games - Olympic Torch	This timeline has videos from the Olympic Torch walk.	13	#OlympicTorch
Iron Maiden at Lisbon 2016	Contest	8	#MaidenLisbon

New Timeline

Figure 3.8: Livetime Listings



## BUILDING BETTER TIMELINES

With Livetime we are aiming for two main goals: First we have the goal of building better storylines. Like we have already previously discussed, in Livetime we have timelines. Timelines can represent a real life event or a real life story through a collection of videos. With this in mind we reach the second goal that is to try to achieve a way to make the user contributions to be the best possible having good quality videos that makes sense to the context of the event.

In this chapter, we will discuss how we achieve this and we will present timelines and contest creation and visualization, the filling of story gaps and the reward mechanism that we use to encourage users to make their Livetime contributions.

### 4.1 Timelines

Timelines are absolutely essential and key to Livetime application. In Livetime we want users to make the best and the most contributions possible to our timelines to help us growing the content available for a given event which will help to tell its story.

#### 4.1.1 Event Storylines

In our system, everyone can contribute to any timeline available. The goal is to the most complete timeline possible. To help timelines building, there is the possibility of hosting competitions to engage users into making their contributions with the possibility to be rewarded by them. However this is just one way to reach the goal of building better timelines, since we are confident that most contributions will come spontaneously which we call a **spontaneous contribution**. A spontaneous contribution is no more than a contribution to an existing timeline with no contest associated only with the goal to help it grow and to help build it. To be a spontaneous contributor is very simple since for each

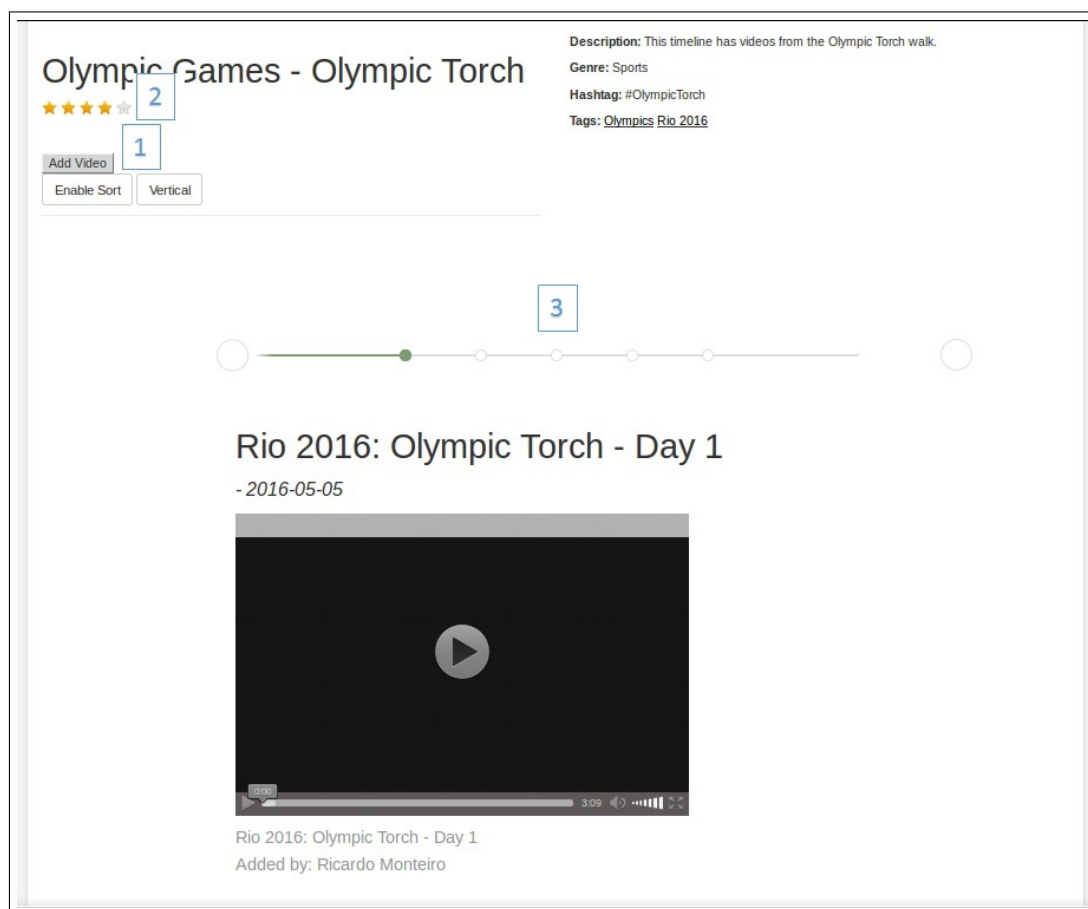


Figure 4.1: Livetime Horizontal Timeline

timeline available there will be a button called **Add Video** marked by the number 1 in figure 4.2. By clicking this button the user will be redirected to his video channel and he will be able to add any of his videos to the timeline story. After the video has been added a new spontaneous contribution will be made.

Now a timeline can have a contest associated or not, but even if there is no contest associated with it in Livetime, we have global competitions running all the time. This means that we have leaderboards associated with video uploads, video ratings, timelines creation and timelines ratings and everything will count towards this "contest". We will discuss more about it in section 4.2.2.

#### 4.1.2 Microblog Contribution

In addition to the **spontaneous contribution** concept we also add two more concepts since a spontaneous contribution can be the result of a **direct contribution**, which is really simple and it is exactly what we have described in subsection 4.1.1, basically is when an user uses Livetime platform to upload and host his video and then adds it to an existing timeline, and there is the **indirect contribution** which is possible using Twitter.

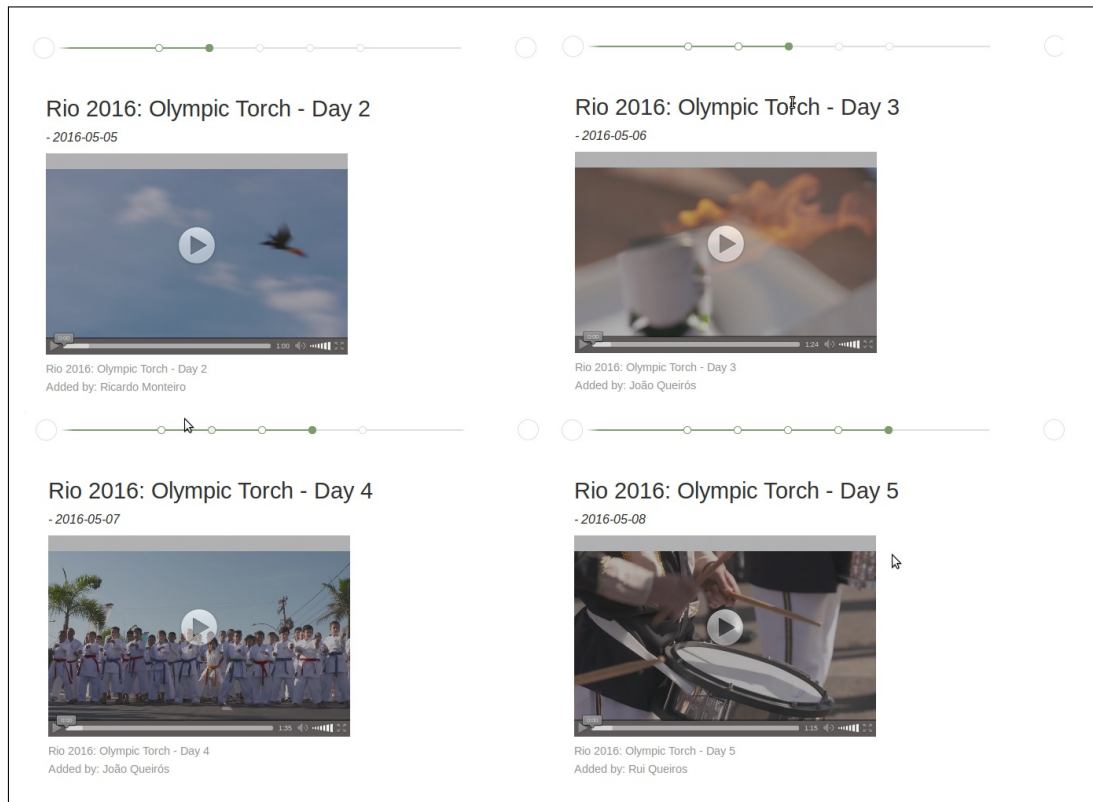


Figure 4.2: Livetime Horizontal Videos Sequence

Using Twitter API <sup>1</sup> we allow our users to contribute to a timeline or a contest both using Livetime or Twitter. To achieve it each timeline has an hashtag. In figure 4.4 we have the page of a contest and an hashtag marked with number one. To facilitate the process of a Twitter Contribution we also created a Livetime Twitter Account. In figure 4.3 we present an example of a Tweet that will be used by Livetime. The twitter posted by the user henrigarces has two important things that make his contribution possible. First it is posted to our Twitter Account and the hashtag of an existing timeline. With both things Livetime will add henrigarces contribution to Livetime and a new **indirect contribution** will be made.

In figure 4.1 we can see both examples of **direct contributions** and **indirect contributions** in the same timeline since **Iron Maiden at Lisbon 2016** is both using videos directly uploaded to Livetime and videos uploaded through Twitter.

### 4.1.3 Visualizing Timelines

Throughout this work we emphasize the importance of our timelines as one of the biggest contributions of our work. That contribution is the possibility to represent the story of an event which will at the same time help to reduce redundancy and to have a better organized content. These timelines are a composition of videos uploaded for one or more

<sup>1</sup><https://dev.twitter.com/>



Figure 4.3: A Twitter Contribution Made By a Livetime User To Twitter

members and to visualize them, we have two options which we call horizontal like in figure 4.4 and vertical timeline like in figure 4.4.

Horizontal timeline works like a slideshow as the work of Delgado et al. [10]. In the figure, below the mark number 3 we have a line where each contribution made to the timeline is represented by a circle. All the circles are clickable and when we click one of the circles we will change video. In figure 4.2 we have the rest of the videos that composes **Olympic Games - Olympic Torch** timeline. Each video can be reached clicking existing circles. Horizontal timelines allows us to better compress the content, which requires much less visual space to present the information in the screen. Of course the information that is showed is also much less from what we can see in vertical timeline in figure 4.4.

In vertical timeline, the form of navigation is very different since to see all of the content that composes the timeline it is required to scroll up or down instead of slide right or left, but it is also much faster to see all all videos that composes the it.

An important point of our timelines is video organization. As reported in Chapter 3 the focus of our work is not the automatic sort of videos, so when a video is added to a timeline it will be added to its last position. However its position can be changed if users think there is a different and better organization to it. With that in mind, we added as it possible to see in figure 4.4 a button called *Enable Sort* which is signed with 1.1. When a user clicks that button users get the possibility to rearrange the timeline order dragging



## Iron Maiden at Lisbon 2016

Add Video
Enable Sort
Disable Sort
Confirm Edition

2

**Competition Type:** Winner Takes-It-All

**Contest Description:**

**Prize amount:** 1000

**Leaderboard:** Most Contributions

**Hashtag:** #MaidenLisbonFCT 1

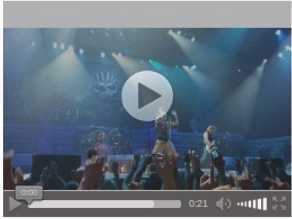
**Member:** jmq@sapo.pt

**Timeline:** Iron Maiden at Lisbon 2016

**Status:** Open

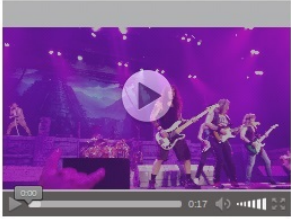
**Iron Maiden - Death or Glory (Lisbon 2016)**

Vid:19  
©2016-07-11  
Iron Maiden Live at Lisbon  
Added by: Rui Queiros



**Iron Maiden - Live at Lisbon (2016)**

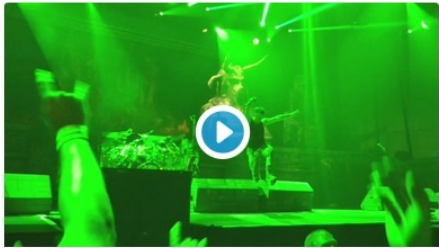
Vid:20  
©2016-07-11




Iron Maiden - Live at Lisbon (2016)  
Added by: Rui Queiros

**Twitter Contribution**

Tweet:3





**Henrique Garcês**  
@henrigarces

Follow

@lifetime\_r here is my contribution #MaidenLisbonFCT  
10:36 PM · 11 Aug 2016

1

2

3

4

Figure 4.4: Livetime Vertical Timeline Inside A Contest

and dropping the videos where they want. Then they just need to confirm the edition and a new order will be arranged.

But this is not the ending regarding our timelines since for each of them there is the possibility to create a new contest. There is a button signed with number 3 in figure 4.4 that gives the possibility for the user to create a new contest for that specific timeline position. We will discuss more about that right in the next subsection 4.1.4.

#### 4.1.4 Detecting Storygaps

We introduced this section in the end of the last one, since they are both connected. One of the main purposes of Livetime as we already repeated a few times is to have our events/timelines with the most videos possible to also be the most complete possible, since each video will fill gaps that exist in it.

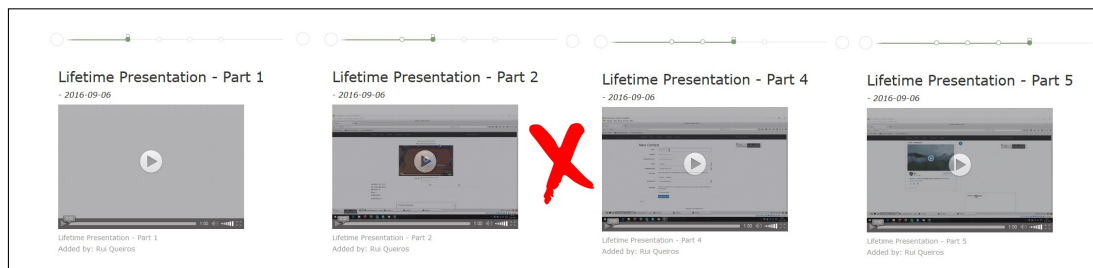


Figure 4.5: An Example of A Storygap Found in Livetime

In figure 4.5 we have an example of a simple to detect **Storygap**. Basically we can divide a storygap in two parts, the first part which is represented in the picture is where we can clearly see that something is missing from an event. In the presented figure, we can check that the timeline is composed with four videos and we can also clearly see that the videos are titled **Lifetime Presentation - Part 1**, **Lifetime Presentation - Part 2**, **Lifetime Presentation - Part 4** and **Lifetime Presentation - Part 5**, so we are clearly missing **Lifetime Presentation - Part 3**. This is a **basic storygap** that someone with the missing video can fill.

Another and more complex storygap is for instance when we have an event of a summer music festival with multiple days. We have some videos from each day but we would like to have more videos to a specific day. With that in mind we introduced contests. A contest is something that every user create and then Livetime users will compete against each other to win it. The contest main goal is to gather videos to specific events or to a specific day of a festival as exemplified. Contests can be created from scratch with no timeline association or to already existing timelines 3.1.1.3. To create a contest for an already existing timeline, in figure 4.4 there is the mark number three which points to a button that when clicked allows users to create a contest to that specific timeline part. To create a contest users will find a form like the one in figure 4.7. For each contest created there are a few options that user will need to select like the competition type and

**New Contest**

Name \*

Hashtag \*  1

Hashtag for event \*  2

Genre

Competition type:  3

Descrição: Winner Takes-It-All is a challenge where only the leader of the leaderboard wins the prize.

Money  Reward

Pot:

Leaderboard  4

Descrição: A type of competition where all contributions equal the same and where making contributions is the main goal.

Schedule Date?

Start date: 2016 July 3

End date: 2016 August 15

Figure 4.6: Form Of A Contest About To Be Created

leaderboard type. In Livetime we offer four types of competitions. First there is Winner Takes-It-All which is nothing more than a contest where only the leader of the contest will win the described prize. We also have an equal distribution competition where the prizes will be distributed in equal parts to all the contributors, then we also offer a top contribution system where the prize will be offered only to a few contributors in the top of the table and finally the possibility to create a contest with no reward at all. We choose this type of contests because we found them to be the most popular with the work of Rokicki et al. [26] and the ones will give us the best results.

After selecting the competition type, contest creator also has to select a type of leaderboard. We will discuss more about our leaderboard system later in this chapter. For now and in the contest context we offer three types of leaderboards called Most Contributions which counts the number of contributions of each user to the timelines. Best classification which counts the best rated contributions of each user to the contest and finally a mixed of both systems.

In conclusion, our contests are an addition to Livetime as way to fill storygaps.

## 4.2 Collaborative-Competition

### 4.2.1 Rewarding

Crowdsourcing and rewarding are nowadays hand to hand. As we showed in chapter 2 with a concrete example [28] these type of initiatives are growing since companies in

general are saving money and at the same time having the job done with much more options to choose and sometimes with better quality.

Since we want to gather videos we also want to give our users the opportunity to *pay-to-get* which basically means that they can open a contest and reward the contributors. Rewards will be chosen by the contest creator and it can be a payment or a prize that will be described by users. In future work we aim to develop this system adding an integrated payment system, an automatic prize distribution and some safety policies regarding these trades. As we studied in section 2.1.4 this rewarding system exists for a reason and the studies presented there show us that they work.

In use case 3.1.1.4, we gave an example of how to use contests and we from what we studied came to the conclusion that it can be really helpful to achieve for instance the storygaps discussed in section 4.1.4. Our contests are supposed to work together with Leaderboards, for instance in figure 4.7 it is possible to check the current ranking of the Iron Maiden at Lisbon 2016 contest presented in figure 4.4. As reminder, when we created that contest we decided to use a Winner Takes-It-All competition type and leaderboard where the user with most contributions wins. In the picture we can see that there are two contributions from the user Rui Queiros and one twitter contribution from user @henrigarces. Since Rui Queiros is a registered user in the contest leaderboard, we will see his registration e-mail, while concerning twitter contribution we will see the twitter account of the participant.



Rank	Member	Score
1	ruiqueiros@gmail.com	2.0
2	henrigarces	1.0

Figure 4.7: Livetime Contest Leaderboard

Although we believe that rewarding is a great way to achieve video gathering, we also believe that this possibility is more interesting for companies responsible for the real life events creation since they are the ones who could be more interested in this type of contributions and who also have the possibility to pay to receive as example of what Indian government did with Invite [28] referenced in section 2.1.4.

But in Livetime there are other types of competition that we offer that were already slightly addressed and that will be discussed in the next section.

### 4.2.2 Involving End-Users

We referenced in previous chapters that it is very important for Livetime to keep users interested and involved with our product and we have defined a set of features that we

would need to approach like interface simplicity and some innovative features that would make users interested.

We have also studied in chapter 2 that users like to be challenged and that they like to compete among themselves. Contests are just one way to achieve that but in Livetime we offer more. As we presented in the previous section 4.2.1 our contests offer leaderboards which are the way to determine users classification when they are participating in it and it is in leaderboards that we will focus on this section.

Besides offering contest specific leaderboards Livetime also has global leaderboards that are updated every time a video is uploaded or a timeline created or rated. In figure 4.8 we have all type of leaderboards that we have available in Livetime and we will now present them all. Two of them are very simple to explain which are **Top Videos** and **Top Timelines**. In these leaderboards each timeline created or video uploaded will add one point to the Livetime member responsible to its creation which is like the competition presented in section 4.1.4. Then we have Videos By Rating and Timelines By Rating leaderboards that count the rating given to the video and to timelines by users. Basically each star corresponds to a value and everytime the rating of a video changes so will change the leaderboard classification increasing or decreasing the user score accordingly. Imagine a video rated with one, it gives the user five points in the score. A video rated with five stars gives the user twenty points in the score. The same goes for timelines. We use this classification system to engage users for better contributions. The Best Classification Ranking we can choose in Contests works like this. Every video can be classified and it will influence the score. We learned from [7] that ratings in applications of this genre are not that much used, in fact only 0.22% of the total video views generate ratings, but with our system we believe it can change since there are many benefits associated with them.

Finally we have the **Global** which is nothing more than a sum of all the timelines. In contests a leaderboard of this genre will be called Mixed since it will count not only video contributions but also its ratings.

Despite of leaderboards importance they are not the only way that we find to get users more involved. We have read about Twitter and that is why we decided to integrate it with Livetime for instance in the building of storylines 4.1.2. But that is not the whole Twitter features that we have in Livetime. As we announced before, we decided to create a Livetime user account. This account is used to facilitate the process of filtering users contributions to our solution, but it has a few other features. Everytime a new contest or timeline is created Livetime will automatically make a tweet and will post it to its Twitter account announcing that a new contest or event was created like we show in picture 4.9 that way all the users that follows Livetime on Twitter will instantly know of the creation and can start helping and participating in the timeline building process.

In Livetime's homepage, we also added a widget with Livetime feed stories. In this widget users can not only check the latest Livetime tweets but they can also start following Livetime on Twitter.

Last, a user when is creating his Livetime account can associate his Twitter Account.



Figure 4.8: Livetime Global Leaderboards

Basically when an user fills the Twitter Account form field presented to new users when all the contributions made direct or indirect to Livetime will count points towards the same user in the leaderboards rankings.

With all these features associated with all the other features, like visualization events with specific videos and even having the opportunity to win prizes, we offer a system with different ways of interacting for a new User Generated Content Video System.



Figure 4.9: Livetime Twitter Account With Automatic Tweets





## EVALUATION

This chapter is set to evaluate our work. In the following sections we will present the used methodology to evaluate the application and we will present the results of the tests gathered.

## 5.1 Methodology

After the development of the application we decided that it was necessary to validate the developed application to see, if it in fact fulfils the requirements and its initial purpose to deliver users a new way of interacting with a new video service platform that uses collaboration between users and crowdsourcing to help video gathering.

Our evaluation will focus on the usability and utility of the features that we developed in Livetime. To do it we gave our users a guide [A](#) with some Livetime tasks to be performed and questionnaire [B](#) to be answered in the end related to the tasks that were performed. The task list goes through most of the features currently available in Livetime platform with a special focus in timelines creation with contributions coming both direct and indirect. The questionnaire has the goal is divided in three sections. The first section allow us to determine the user profile, the second is based on System Usability Scale by Brooke et al. [[5](#), [6](#)] and the third section consists in a series of questions to evaluate the system utility, interface and also the global appreciation of our platform.

Regarding SUS Evaluation System we are using it since it is a very trustworthy tool to evaluate interfaces and can be used in multiple systems like our website achieving great results even with small samples. This tool consists in ten questions with a linear scale from 1 - *Strongly disagree* to 5 - *Strongly Agree*:

1. I think that I would like to use this system frequently.

2. I found the system unnecessarily complex.
3. I thought the system was easy to use
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

The value is achieved summing all the results from each question but not in a linear way since for each odd number the number subtracts 1 for each result chosen by the participant and for even number the result is obtained subtracting 5 to each of the values chosen. The final result will vary between 0 to 40.

Apart from these questions we also add a few of our own to know what users think about the main features that we added to our application in particular and that focus on the general system performance. All the following quotes except the last two which are questions of yes or no answer have a linear scale from one to five similar to the one used previously.

1. How do you classify the overall system's overall experience.
2. It is easy to upload videos.
3. It is easy to create timelines.
4. It is easy to create contests.
5. It is easy to contribute to timelines or contests.
6. Timelines are useful.
7. Contests are useful.
8. Competing among other users will lead to better contributions.
9. Twitter features are useful.
10. Leaderboards are useful.
11. Contests are useful.

12. It is easy to view the content in general?
13. Did you experience any difficulty using the application?

Next we will focus on the results that we achieved, in **C** is possible to find all the answers from the subjects.

## 5.2 Test Scenario

During our test scenario we had twenty one participants from ages from eighteen to fifty six years old with an average of 26,38 years old. The percentage of male and female subjects were divided in 23,38 % female and 76,2 % male. Finally regarding the subject information we also know that fifty seven percent of the twenty participants are related to computers science or computers in general, while the other 43 are from other areas.

All the test subjects were presented with a series of tasks **A** to perform with a few options and paths that they can take. In the test scenario we created a few timelines to see to which ones the users will contribute to. Our main goal is to see the evolution of the timeline called **Concerts in Portugal in 2016**. We started the timeline completely without any video and in the tasks guide we recommend our users to contribute to it. We will infer about how many contributions were made to that timeline and we will present the timeline that resulted from users contributions. We will also infer about other contributions and tasks performed and the results of our questionnaire **B**.

In table 5.1 we have the results for the usability using the SUS system. The range varies from 60% of classification to 97,5% with an average of 76,31%. Regarding learnability we achieved better result with a minimum of 62,5% and a maximum of 90,63% and an average of 79,77%. The standard deviation for this two components were bellow 10 and 5 respectively. Regarding usability we achieved the best results with a minimum of 50 and a maximum of 100. To calculate the usability using SUS only questions number four and ten matter. That way the standard deviation is a little high since each result influences the final usability calculation a lot.

Still we think we achieved some good results which demonstrates that the interface is easy to use and also that different formations area does not have a that big influence in the final result.

As we said previously apart from the SUS questions we also added a few of our own more focused in the work we developed. To start, we have in table 5.1 the results that classify the system overall experience. The options vary from 1 to 5 and 1 corresponds to a Very Bad Classification and 5 to a Very Good Classification. The most common answer was 4 that classifies the system as Good with 66,7% but we also got 23,8% that classified the system as very good and only 9.5% as average which shows that users in general enjoyed the system.

After, we made a few questions about the usefulness of some of our features. To start in figure 5.2 we have the results regarding the usefulness of timelines. Like in the

Table 5.1: Results From SUS Questionnaire.

#	Age	Sex	Formation	SUS	<i>Learn.</i>	<i>Usab.</i>
1	25	M	Computer Sciences	87,5	84,38	100
2	18	M	IT Technician	80	78,13	87,5
3	24	M	Computer Sciences	72,5	62,5	87,5
4	26	M	Computer Sciences	92,5	84,38	100
5	23	M	Computer Sciences	75	78,13	87,5
6	23	M	Computer Sciences	97,5	84,38	100
7	24	M	Computer Sciences	60	81,25	50
8	24	M	Computer Sciences	70	84,38	87,5
9	28	M	Forest Engineer	77,5	78,13	100
10	30	M	Computer Sciences	62,5	75	62,5
11	23	F	Arts	67,5	71,88	75
12	24	M	Computer Sciences	87,5	81,25	87,5
13	25	F	Arts	65	71,88	62,5
14	24	M	Electronics Engineer	62,5	75	62,5
15	23	M	Computer Sciences	82,5	71,88	100
16	53	F	Accounting	87,5	90,63	50
17	24	M	Computer Sciences	72,5	78,13	100
18	24	F	Architecture	65	87,5	50
19	25	F	Pharmaceutical Sciences	77,5	81,25	87,5
20	26	M	Psychology	70	87,5	75
21	38	M	Engineering	90	87,5	100
<b>Average</b>				76,31	79,77	81,55
<b>Maximum</b>				97,50	90,63	100
<b>Minimum</b>				60	62,50	50
<b>Standart Deviation</b>				8,81	5,48	17,78

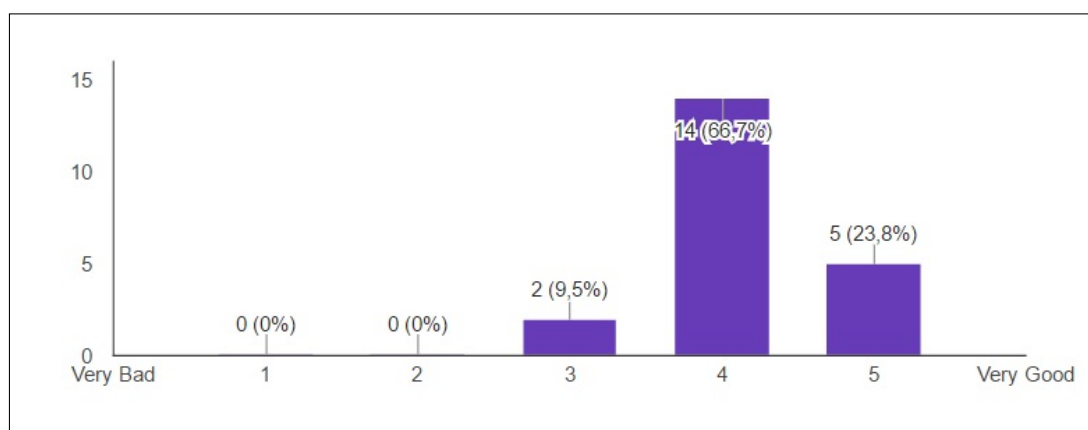


Figure 5.1: System overall experience classification

previous figure, the results vary from 1 to 5, where 1 is not usefulness and 5 is very useful, and again the most common answer was 4 with 47,6 %. Regarding this question, every value has been chosen but only two which corresponds to 9,6% are negative and 33,33%

saying that is a very useful feature which demonstrates that in general the feature is very welcome. The same type of question goes to figure 5.3, but this time is regarding the usefulness of contests. The response was not so good as the previous one but still the most common answer was 4 and again with 47,6 %, but only 19% answered that is a very useful feature. We believe we could achieve much better results regarding this question if we offered a real contest were the users could be in touch with a big competition and prizes.

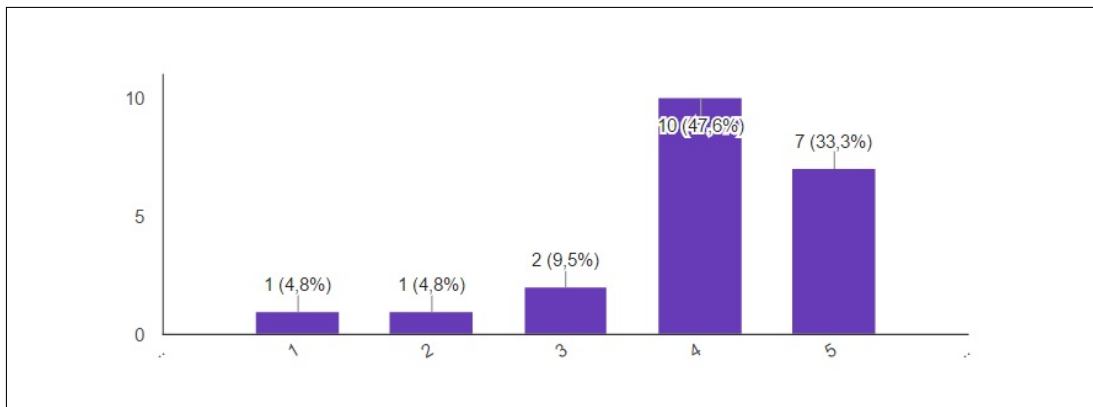


Figure 5.2: Usefulness of timelines

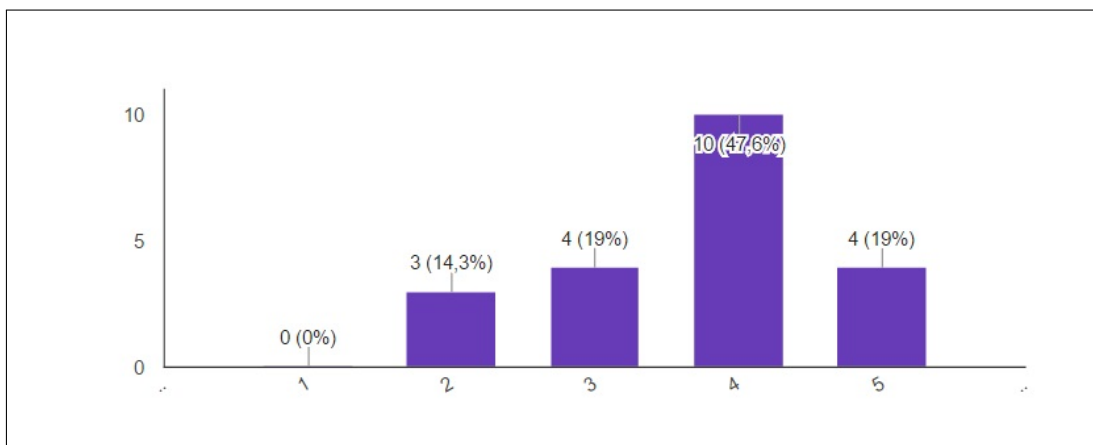


Figure 5.3: Usefulness of contests

Still in the question regarding the Usefulness of competitions 81% answered that they are useful or very useful with very similar distribution between values 4 and 5.

Finally we asked the usefulness of Twitter since we have a few Twitter features present in Lifetime. Again the feedback received was very good since 85,8% answered useful or very useful to our question. We believe that our Twitter features were a great improvement to our platform and although not all users in our test scenario tested all that features they also seem to believe it.

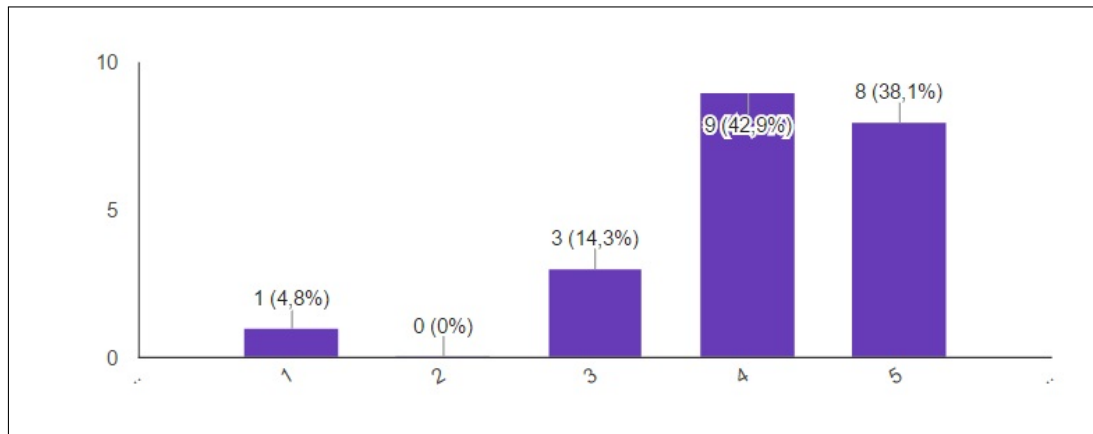


Figure 5.4: Usefulness of competitions

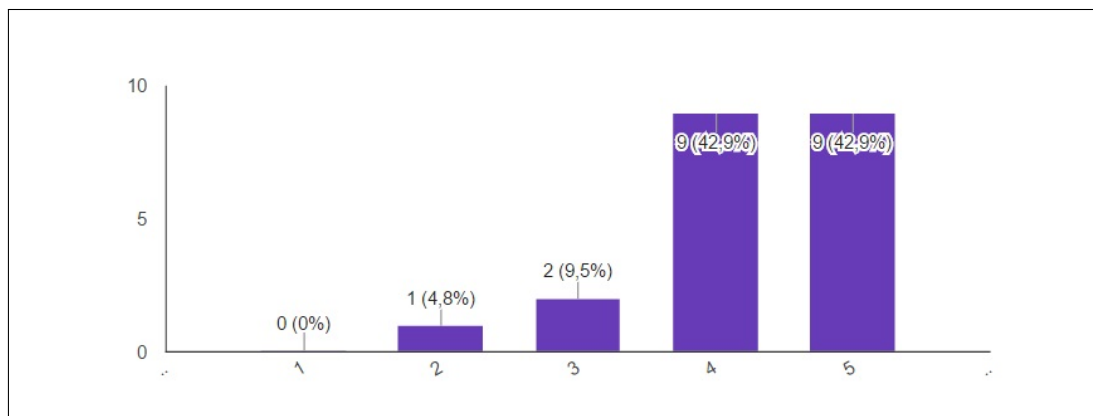


Figure 5.5: Usefulness of twitter

### 5.3 Request Videos

As we mentioned previously, during our test section, we launched a small timeline and asked users to help us build it. The timeline is called **Concerts in Portugal in 2016** and the goal was for the users to contribute with videos from concerts, where they went during this year in our country.

The timeline had also a little contest associated to rank the number of contributions made by each user and started completely empty. Not all the users contributed to it and we believe the main reason from what we talked with them, was the lack of videos for the subject in matter. Still we gathered a total of eight contributions to it with two coming from the same user and the others coming from individual users and Twitter. Since all the twitter contributions were made using the Twitter account that we provided for our test scenario, we do not know how many of them are individual. Five of the contributions were directly to the contest created and four of them were made using the Twitter Account provided.

In figure 5.6 we can check the final result of the timeline that was created with users

contributions during this test phase. To present the whole timeline in one picture we decided to compress all the contributions in the picture. It does not represent the real timeline presentation that can be found in figure 4.4.

Apart from that there were made twenty three direct videos uploads to Lifetime from eighteen individual users and there were also made eight tweets uploads through Twitter, six of them using the Twitter Account that we provided. Finally there were created ten timelines and three contests from thirteen different users.

## 5.4 Conclusion

With all our results presented we believe that in general people enjoyed our system and the idea behind it. Of course it would be better for a system like ours to have a larger amount of subjects to test it, making their contributions and to give their opinion, but we also believe that our results are already a good sample for a first phase. Also, to test our system and perform all the supposed tasks users need to spend a few amount of time since there is a lot to do and to try so that is probably one of the main reasons why not all of them performed all the tasks.

There are still some improvements that need to be made in the responses we received there were even a few suggestions made by the testers. Ideas like changing the buttons colour to better suit reality and at the same time facilitate user interaction is an easy addition to be made that we will think about in future work. A few dropdowns to order videos by genre is another of the suggestions that were made. An important suggestion that was made was the possibility to direct upload a video directly into a timeline or a contest instead of what is happening now were users need to first upload their video into Lifetime and only then add it to the timeline.

Finally there is a few performance issues that need to be fixed that were reported but that we were aware of them. It was also important to have a sample of people from different formation areas. Despite the main formation area of responses being Computer Science, almost half of them were from other different areas and did not seem to have significant difficulties performing the tasks or understanding the system.

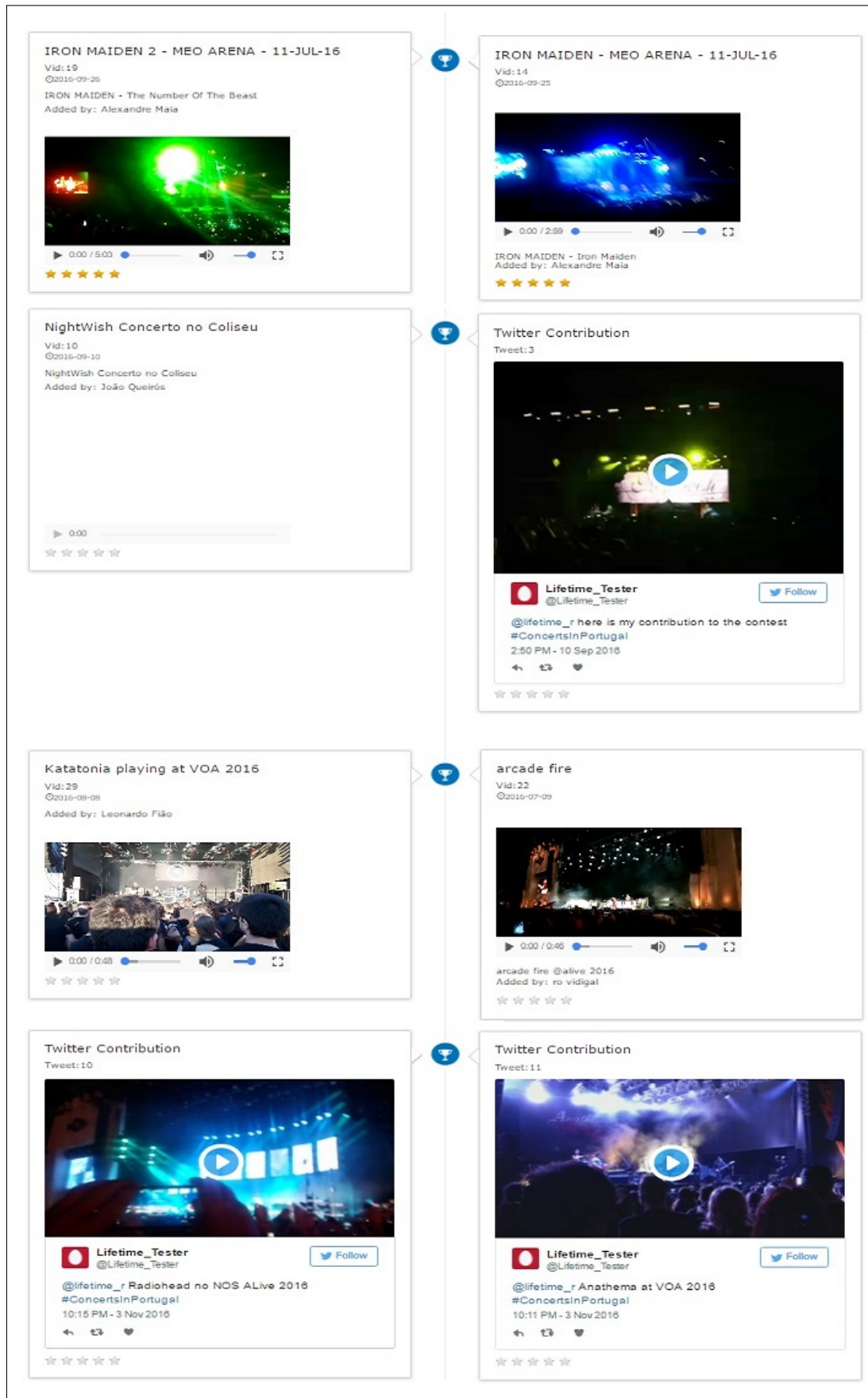


Figure 5.6: Concerts in Portugal in 2016 Timeline Result



## CONCLUSION AND FUTURE WORK

In this chapter we summarize the work done during this dissertation, and what we accomplished as a final result. We will also discuss some of Livetime future work to evolve our application.

### 6.1 Conclusion

We started our work with the idea that Internet is giant and there are plenty of different and similar applications fighting to gain their share. In the context of our work, we presented a few existing applications like [2.2.1.3](#) or [2.2.1.4](#) that are very similar with only a few differences that give them a purpose to exist.

With our work we developed a solution to focus on video gathering to specific topics aggregation the videos together at the same localization with a timeline presentation. We also give the users the opportunity to ask for videos to other users always with a competition system in mind to engage them for better contributions and to keep them interested.

Apart from that, we added Twitter integration to facilitate user interaction. That way we allow our users to directly contribute to our application, or to make indirect contributions through Twitter.

In addition, we want to highlight some points after the test phase that we consider important:

**Timelines and Contests** were well received by our testers with more than 80 % of the users respondents answering that timelines are useful or very useful and more than 65 % saying that contests are useful or very useful, which demonstrates that the core of our work made sense.

**Twitter Interaction** was also a well received feature with almost 90 % of positive responses.

**SUS Score** also must be highlighted since we scored 76.31 points which shows that our system is easy to be learned and used by users.

In general, our system offers a solution to share and visualize information which at the same time can help reduce video redundancy and that concentrates related information which facilitates its access and at the same helps to grow the media information available regarding a topic.

## 6.2 Contributions

In chapter 1, we have a section with contributions 1.4 that we wanted to achieve in our work and we achieved them all.

Our work delivers **Livetime** fully implemented. Livetime is an on-line system that aggregates all the ideas described in chapters 3 and 4. Our solution is available on focus on **collaborative storylines** where users can share their videos into existing stories to help them grow and create their own stories with the possibility to request for videos to them with a series of possibilities like creating contests that permit to **reward** users contributions.

Finally we have also worked on a **short-paper** called **Livetime** where we focused on our created solution and what it offers and also all the results that we achieved that can be found in chapter 5.

## 6.3 Future Work

Livetime still has a lot to grow and that is way we already have a set of features defined. We believe this work was only the beginning and we can explore much more features that will make the platform richer and more interesting. Some of the features that are thought to be added were already addressed in section 3.1.3 and our main goal with it is to evolve to different directions.

The first goal is to have a better notification system. We want users to receive notifications based on a subscription system inside our platform everytime a new timeline or contest is created and opened, and we also want to notify the users responsible for the event and contest creation everytime a new contribution is made. This notification system will also be used for the feature **Timeline Intelligence** that was described in chapter 3 and which has the goal to automatically evaluate contributions. The automatic evaluation of contributions is also something that we aim for future work to try to avoid contributions that does not make sense in the context of a timeline which will help the building of better storylines.

The possibility of adding video portions instead of a complete video is also in study. That way we want to allow users to directly inside our platform cut some video parts and add them to the timeline instead of the complete video. Picking up from this idea we have the **Timeline Resume** feature that we believe it will be very helpful when we are presented before a large amount of videos in a timeline. With this idea we want to pick the tags that users can manually add to each video individually to try to extract a small part of all the videos that compose the timeline. The goal is to create one unique video with pieces of all the videos that composes that timeline. This way users will have a faster and easy way to have a quickly preview of each video composition and will have the opportunity to jump right into it.

Apart from these features we also want to extend our members information. With that we want to build a safer environment for the contests creators and participants. We want to implement a wallet system that will make sure the contests participants will automatically receive their prize (if we are talking about money for instance) when the contest ends. In the contest area there is still plenty of work to do since in a final platform we will also need to ensure that contests creation with rewards will in fact reward the users contribution so to do it we will need to establish a few policies to guarantee it before allowing everyone to create contests and that we will need to study how to achieve it. We also want to add an easy feature that allows the contest creator to download the videos added to his contest.

We also want to keep evolving our system gamification with the introduction of monthly competitions and an achievement system that users will unlock with their contributions. Our system social aspect is also important and the integration of new forms of contribution and other social networks is also something that will be need to address. We would also to develop our own Livetime API so other users can use and adapt into their applications.

Our system interface will also need to keep evolving to become simpler and modern and to facilitate user interaction like the example that was given during our test phase 5.4. Finally we want to keep involving our system architecture to guarantee for instance a cloud system that was previously thought but it is not yet available.

At the moment we already have plenty of work done and a stable platform running, but as it possible to see from this section we have already a lot of work planned that we really believe that it will contribute to have a better product in the end that users will enjoy to use and to reach larger audiences.



## BIBLIOGRAPHY

- [1] *4K Resolution*. [Online; accessed 27-November-2015]. 2015. URL: [https://en.wikipedia.org/wiki/4K\\_resolution](https://en.wikipedia.org/wiki/4K_resolution).
- [2] M. Balabanović, L. L. Chu, and G. J. Wolff. “Storytelling with Digital Photographs”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’00. The Hague, The Netherlands: ACM, 2000, pp. 564–571. ISBN: 1-58113-216-6. DOI: 10.1145/332040.332505. URL: <http://doi.acm.org/10.1145/332040.332505>.
- [3] J. Bian, Y. Yang, and T.-S. Chua. “Multimedia Summarization for Trending Topics in Microblogs”. In: *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*. CIKM ’13. San Francisco, California, USA: ACM, 2013, pp. 1807–1812. ISBN: 978-1-4503-2263-8. DOI: 10.1145/2505515.2505652. URL: <http://doi.acm.org/10.1145/2505515.2505652>.
- [4] J. Boreczky, A. Girgensohn, G. Golovchinsky, and S. Uchihashi. “An Interactive Comic Book Presentation for Exploring Video”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’00. The Hague, The Netherlands: ACM, 2000, pp. 185–192. ISBN: 1-58113-216-6. DOI: 10.1145/332040.332428. URL: <http://doi.acm.org/10.1145/332040.332428>.
- [5] J. Brooke. “SUS: a retrospective”. In: *Journal of usability studies* 8.2 (2013), pp. 29–40.
- [6] J. Brooke et al. “SUS-A quick and dirty usability scale”. In: *Usability evaluation in industry* 189.194 (1996), pp. 4–7.
- [7] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon. “I Tube, You Tube, Everybody Tubes: Analyzing the World’s Largest User Generated Content Video System”. In: *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*. IMC ’07. San Diego, California, USA: ACM, 2007, pp. 1–14. ISBN: 978-1-59593-908-1. DOI: 10.1145/1298306.1298309. URL: <http://doi.acm.org/10.1145/1298306.1298309>.
- [8] K. Chorianopoulos, I. Leftheriotis, and C. Gkonela. “SocialSkip: pragmatic understanding within web video”. In: *9th International Interactive TV&Video Conference, EuroITV ’11, Lisbon, Portugal, June 29 - July 01, 2011*. 2011, pp. 25–28. DOI: 10.1145/2000119.2000124. URL: <http://doi.acm.org/10.1145/2000119.2000124>.

## BIBLIOGRAPHY

---

- [9] M. Christel, T. Kanade, M. Mauldin, R. Reddy, M. Sirbu, S. Stevens, and H. Wactlar. “Informedia Digital Video Library”. In: *Commun. ACM* 38.4 (Apr. 1995), pp. 57–58. ISSN: 0001-0782. DOI: [10.1145/205323.205337](https://doi.org/10.1145/205323.205337). URL: <http://doi.acm.org/10.1145/205323.205337>.
- [10] D. Delgado, J. Magalhaes, and N. Correia. “Assisted News Reading with Automated Illustration”. In: *Proceedings of the 18th ACM International Conference on Multimedia. MM ’10*. Firenze, Italy: ACM, 2010, pp. 1647–1650. ISBN: 978-1-60558-933-6. DOI: [10.1145/1873951.1874311](https://doi.org/10.1145/1873951.1874311). URL: <http://doi.acm.org/10.1145/1873951.1874311>.
- [11] M. Demirbas, M. A. Bayir, C. G. Akcora, Y. S. Yilmaz, and H. Ferhatosmanoglu. “Crowd-sourced sensing and collaboration using Twitter”. In: *Proc. of WoWMoM*. 2010.
- [12] *Google Photos*. [Online; accessed 06-February-2016]. 2015. URL: <https://play.google.com/store/apps/details?id=com.google.android.apps.photos>.
- [13] A. Habibian, T. Mensink, and C. G. Snoek. “VideoStory: A New Multimedia Embedding for Few-Example Recognition and Translation of Events”. In: *Proceedings of the 22Nd ACM International Conference on Multimedia. MM ’14*. Orlando, Florida, USA: ACM, 2014, pp. 17–26. ISBN: 978-1-4503-3063-3. DOI: [10.1145/2647868.2654913](https://doi.org/10.1145/2647868.2654913). URL: <http://doi.acm.org/10.1145/2647868.2654913>.
- [14] I. Kavasidis, S. Palazzo, R. D. Salvo, D. Giordano, and C. Spampinato. “An Innovative Web-based Collaborative Platform for Video Annotation”. In: *Multimedia Tools Appl.* 70.1 (May 2014), pp. 413–432. ISSN: 1380-7501. DOI: [10.1007/s11042-013-1419-7](https://doi.org/10.1007/s11042-013-1419-7). URL: <http://dx.doi.org/10.1007/s11042-013-1419-7>.
- [15] L. Kennedy and M. Naaman. “Less Talk, More Rock: Automated Organization of Community-contributed Collections of Concert Videos”. In: *Proceedings of the 18th International Conference on World Wide Web. WWW ’09*. Madrid, Spain: ACM, 2009, pp. 311–320. ISBN: 978-1-60558-487-4. DOI: [10.1145/1526709.1526752](https://doi.org/10.1145/1526709.1526752). URL: <http://doi.acm.org/10.1145/1526709.1526752>.
- [16] M. Larson, P. Cremonesi, and A. Karatzoglou. “Overview of ACM RecSys CrowdRec 2014 Workshop: Crowdsourcing and Human Computation for Recommender Systems”. In: *Proceedings of the 8th ACM Conference on Recommender Systems. RecSys ’14*. Foster City, Silicon Valley, California, USA: ACM, 2014, pp. 381–382. ISBN: 978-1-4503-2668-1. DOI: [10.1145/2645710.2645783](https://doi.org/10.1145/2645710.2645783). URL: <http://doi.acm.org/10.1145/2645710.2645783>.
- [17] *Leading social networks worldwide as of September 2016, ranked by number of active users*. [Online; accessed 17-09-2016]. 2016. URL: <http://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/>.

- [18] I. Leftheriotis, C. Gkonela, and K. Chorianopoulos. “Efficient Video Indexing on the Web: A System that Leverages User Interactions with a Video Player”. In: *CoRR* abs/1204.6321 (2012). URL: <http://arxiv.org/abs/1204.6321>.
- [19] Z. Li, M. Wang, J. Liu, C. Xu, and H. Lu. “News Contextualization with Geographic and Visual Information”. In: *Proceedings of the 19th ACM International Conference on Multimedia*. MM ’11. Scottsdale, Arizona, USA: ACM, 2011, pp. 133–142. ISBN: 978-1-4503-0616-4. DOI: 10.1145/2072298.2072317. URL: <http://doi.acm.org/10.1145/2072298.2072317>.
- [20] I. Life On Air. *Meerkat - Android Apps on Google Play*. [Online; accessed 06-October-2015]. 2015. URL: <https://play.google.com/store/apps/details?id=com.getair.meerkat>.
- [21] X. Liu and B. Huet. “Event representation and visualization from social media”. In: *PCM 2013, 14th Pacific-Rim Conference on Multimedia, 13-16 December 2013, Nanjing, China / Also published in LNCS, Volume 8294/2013*. Nanjing, CHINE, Dec. 2013. DOI: [http://dx.doi.org/10.1007/978-3-319-03731-8\\_69](http://dx.doi.org/10.1007/978-3-319-03731-8_69). URL: <http://www.eurecom.fr/publication/4159>.
- [22] *Number of Monthly Active Twitter Users*. [Online; accessed 18-08-2016]. 2016. URL: <http://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/>.
- [23] S. M. P. Coppola R. Lomuscio and E. Nazzi. “m-dvara 2.0: Mobile web 2.0 services integration for cultural heritage”. In: *SWKM*. 2008.
- [24] B. E. Pan Y. “A Survey of Crowdsourcing as a means of Collaboration and the Implications of Crowdsourcing of Interaction Design”. In: *Proceedings of International Conference on Collaboration Technologies and Systems*. 2011, pp. 397–403.
- [25] *Periscope Api*. [Online; accessed 06-February-2016]. 2015. URL: [https://github.com/gabrielg/periscope\\_api](https://github.com/gabrielg/periscope_api).
- [26] M. Rokicki, S. Chelaru, S. Zerr, and S. Siersdorfer. “Competitive Game Designs for Improving the Cost Effectiveness of Crowdsourcing”. In: *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. CIKM ’14. Shanghai, China: ACM, 2014, pp. 1469–1478. ISBN: 978-1-4503-2598-1. DOI: 10.1145/2661829.2661946. URL: <http://doi.acm.org/10.1145/2661829.2661946>.
- [27] D. K. Ronald Schroeter Jane Hunter. “Vannotea - A Collaborative Video Indexing, Annotation and Discussion System for Broadband Networks”. In: (2003).

- [28] N. Shah, A. Dhanesha, and D. Seetharam. "Crowdsourcing for e-Governance: Case Study". In: *Proceedings of the 3rd International Conference on Theory and Practice of Electronic Governance*. ICEGOV '09. Bogota, Colombia: ACM, 2009, pp. 253–258. ISBN: 978-1-60558-663-2. DOI: 10.1145/1693042.1693093. URL: <http://doi.acm.org/10.1145/1693042.1693093>.
- [29] M. Soleymani and M. Larson. "Crowdsourcing for Multimedia Research". In: *Proceedings of the 21st ACM International Conference on Multimedia*. MM '13. Barcelona, Spain: ACM, 2013, pp. 1111–1112. ISBN: 978-1-4503-2404-5. DOI: 10.1145/2502081.2502234. URL: <http://doi.acm.org/10.1145/2502081.2502234>.
- [30] T. Volkmer, J. R. Smith, and A. P. Natsev. "A Web-based System for Collaborative Annotation of Large Image and Video Collections: An Evaluation and User Study". In: *Proceedings of the 13th Annual ACM International Conference on Multimedia*. MULTIMEDIA '05. Hilton, Singapore: ACM, 2005, pp. 892–901. ISBN: 1-59593-044-2. DOI: 10.1145/1101149.1101341. URL: <http://doi.acm.org/10.1145/1101149.1101341>.
- [31] *YouTube*. [Online; accessed 10-October-2015]. 2015. URL: <https://en.wikipedia.org/wiki/YouTube>.



A P P E N D I X



## USABILITY TESTS GUIDE

# Livetime

## Usability Tests

This test section has the goal to evaluate the Livetime application. Livetime is an application that allows video upload and event creation in a collaborative environment. It is an application addressed to every common user who wants to share their videos across the Internet and to view videos related to specific events.

In Livetime, users can create their own events, their contests and can contribute with videos using both direct uploads to the application and indirect uploads through Twitter. The application also allows users to create contests to fill some gaps in the already existing timelines.

This test consists in going through some of the application main features and also we invite you to make part of the first official Livetime video gathering **Concerts in Portugal in 2016**.

We recommend you to read carefully the tasks below. If you have any doubt performing any of the tasks please contact the supervisor. In the end we will provide a questionnaire to evaluate your performance using our system.

### Notes

- In our application you will find some timelines already available. We want you to help filling them. We would also like to invite you participate in our contest called: **Concerts in Portugal in 2016**. We would appreciate you to contribute with any video from any concert you attended to during this year. If you do not have any feel free to upload other video to other available timeline.
- Timelines and Contests have hashtags. These hashtags are used for contributions through Twitter. On contest creation an hashtag will be needed to identify the contest and a different hashtag will be needed to identify only the timeline.
- To contribute using Twitter you can use your own account Twitter Account or the test account created for our testing environment:  
User: **Lifetime\_Tester**  
Password: **TesterFCT2016**
- To contribute use Twitter you need to attach a video with the following text: **"@Lifetime\_r my contribution to #TimelineOrContestHashtag**.
- We recommend the use of Mozilla Firefox.

### Tasks

1. Go to the menu and create a new account using the necessary information.
2. In the Menu select the option to Upload a New Video.
3. Access the list of all the timelines and then choose one timeline of your choice. After the result presentation please add your video to the timeline.

4. Change the timeline view to Vertical Timeline and Enable Timeline Sort. Change the order of the videos dragging them to change position and save (confirm edition) if you think your order is better.
5. **Optional:** Open a contest clicking the **trophy button** to fill a specific gap in the current timeline.
6. Go to the homepage and search a video. Select one of the videos and open it leaving a comment and a rating.
7. Play a video and add a Tag to it.
8. This task can be skipped if **task 5** was completed: Create a new timeline or a contest.
9. **Optional:** Contribute to the newly created timeline or contest.
10. Go to a contest or a timeline, it can be the one that you just created, and use its hashtag to tweet with a new contribution. After refresh the page.

## End of Tasks

Now please fill the questionnaire about the application and the tasks that you have just performed. You can find it on the application's navigation bar or you can use the direct address: <https://goo.gl/forms/cZaTiq2vrp39OPOh2>.

Thank you for your collaboration!



APPENDIX



## USABILITY TESTS QUESTIONNAIRE

# Livetime Questionnaire

This questionnaire is anonymous, so your personal information will just be used for statistical analysis. The questionnaire is divided in three sections and it pretends to evaluate Livetime's usability. In the next pages we will make a few questions about the tasks that you have just performed. We appreciate you to be as honest as possible to obtain the best results information about the work done.

\*Obrigatório

1. Age \*

---

2. Sex \*

Marcar apenas uma oval.

Male

Female

3. Formation Area \*

---

## System Usability Scale

---

SUS is a very used scale to evaluate systems. We want you to be the most instinctive possible answering the following questions.

4. I think that I would like to use this system frequently. \* Marcar apenas uma oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

5. I found the system unnecessarily complex. \* Marcar apenas uma oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

6. I thought the system was easy to use. \* Marcar apenas uma oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

7. I think that I would need the support of a technical person to be able to use this system. \*

Marcar apenas uma oval.

1	2	3	4	5
---	---	---	---	---

---

Strongly Disagree      Strongly Agree

---

8. I found the various functions in this system were well integrated. \* Marcar apenas uma oval.

1 2 3 4 5

---

Strongly Disagree      Strongly Agree

---

9. I thought there was too much inconsistency in this system. \* Marcar apenas uma oval.

1 2 3 4 5

---

Strongly Disagree      Strongly Agree

---

10. I would imagine that most people would learn to use this system very quickly. \* Marcar apenas uma oval.

1 2 3 4 5

---

Strongly Disagree      Strongly Agree

---

11. I found the system very cumbersome to use. \* Marcar apenas uma oval.

1 2 3 4 5

---

Strongly Disagree      Strongly Agree

---

12. I felt very confident using the system. \* Marcar apenas uma oval.

1 2 3 4 5

---

Strongly Disagree      Strongly Agree

---

13. I needed to learn a lot of things before I could get going with this system. \* Marcar apenas uma oval.

1 2 3 4 5

---

Strongly Disagree      Strongly Agree

---

## System Usability and General Performance

This is the last section of our questionnaire and focus more on the overall Livetime experience and Interface

14. How do you classify the overall system's overall experience \* Marcar apenas uma oval.

1 2 3 4 5

---

Very Bad      Very Good

---

15. It is easy to upload videos. \* Marcar apenas uma oval.

	1	2	3	4	5	
Very Easy	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Very Hard

16. It is easy to create timelines.

Marcar apenas uma oval.

	1	2	3	4	5	
Very Easy	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Very Hard

17. It is easy to create contests.

Marcar apenas uma oval.

	1	2	3	4	5	
Very Easy	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Very Hard

18. It is easy to contribute to timelines or contests. \* Marcar apenas uma oval.

	1	2	3	4	5	
Very Easy	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Very Hard

19. Timelines are useful. \*

Marcar apenas uma oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

20. Contests are useful. \*

Marcar apenas uma oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

21. Competing among other users will lead to better contributions. \* Marcar apenas uma oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

22. Twitter features are useful. \* Marcar apenas uma oval.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

23. Leaderboards are useful. \* Marcar apenas uma oval.



---

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

---

24. It is easy to view the content in general. \* Marcar apenas uma oval.

- Yes  
 No

25. Did you experience any difficulty using the application? \* Marcar apenas uma oval.

- Yes  
 No

26. If yes please detail them here

---

---

---

---

---

27. Any suggestions? Leave them now

---

---

---

---

---



A P P E N D I X



## U S A B I L I T Y   T E S T S   Q U E S T I O N N A I R E   A N S W E R S

# 21 respostas

---

## Resumo

### Age

24

23

25

26

18

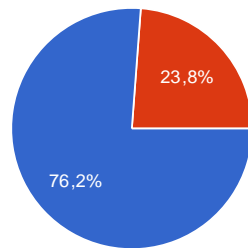
28

30

53

38

### Sex



Male 16 76.2%

Female 5 23.8%

### Formation Area

Computer Science

Engenharia Informática

Informatic

Técnico de Gestão de Equipamentos Informáticos

informática

Computer Science

Forest Engineering

IT

Conservação-Restauro

Informática

Restauro

Electronics

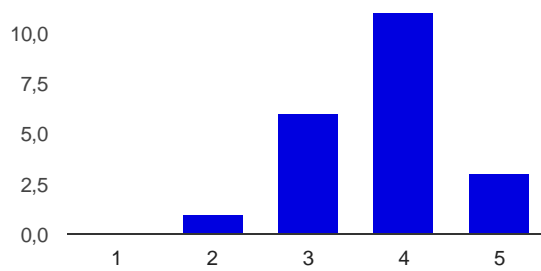
Computer Engineering

Contabilidade

Architecture  
 Pharmaceutical Sciences  
 Psicologia  
 Engineering

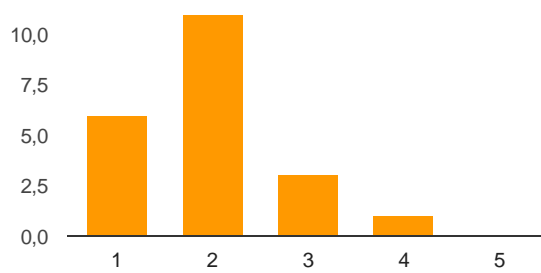
## System Usability Scale

I think that I would like to use this system frequently.



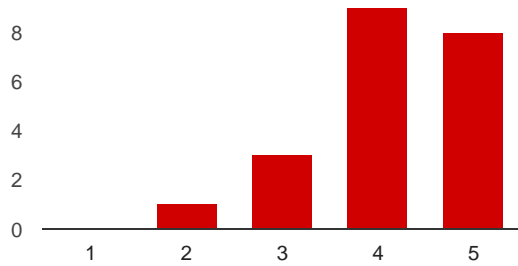
Strongly Disagree: 1	0	0%
2	1	4.8%
3	6	28.6%
4	11	52.4%
Strongly Agree: 5	3	14.3%

I found the system unnecessarily complex.



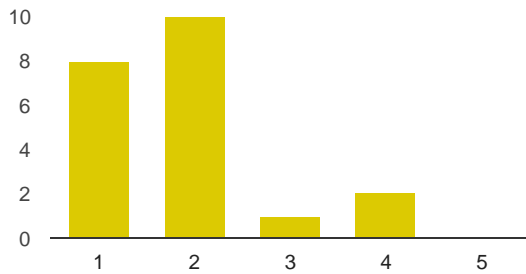
Strongly Disagree: 1	6	28.6%
2	11	52.4%
3	3	14.3%
4	1	4.8%
Strongly Agree: 5	0	0%

I thought the system was easy to use.



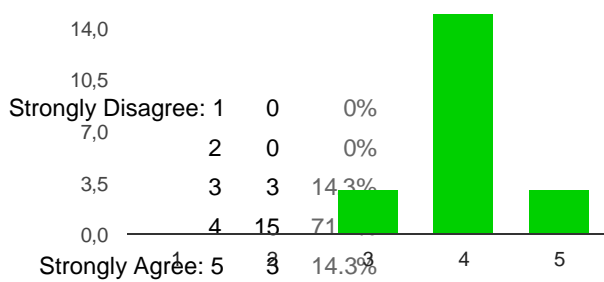
Strongly Disagree: 1 0 0%  
 2 1 4.8%  
 3 3 14.3%  
 4 9 42.9%  
 Strongly Agree: 5 8 38.1%

I think that I would need the support of a technical person to be able to use this system.

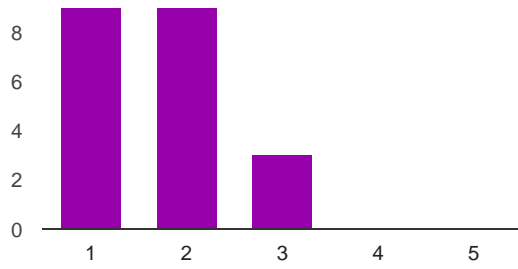


Strongly Disagree: 1 8 38.1%  
 2 10 47.6%  
 3 1 4.8%  
 4 2 9.5%  
 Strongly Agree: 5 0 0%

I found the various functions in this system were well integrated.

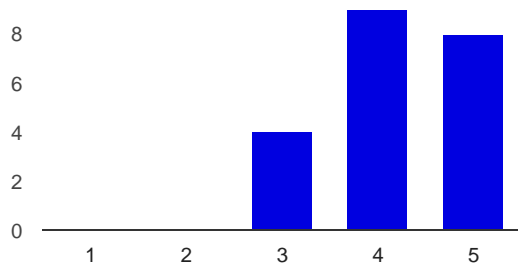


I thought there was too much inconsistency in this system.



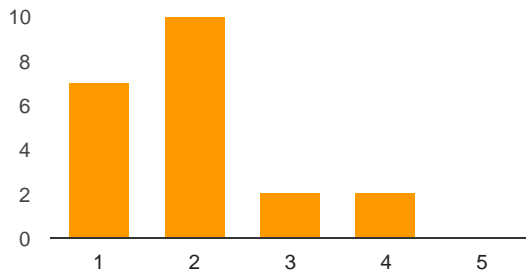
Strongly Disagree: 1 9 42.9%  
 2 9 42.9%  
 3 3 14.3%  
 4 0 0%  
 Strongly Agree: 5 0 0%

I would imagine that most people would learn to use this system very quickly.



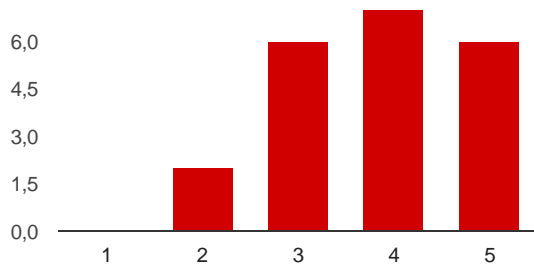
Strongly Disagree: 1 0 0%  
 2 0 0%  
 3 4 19%  
 4 9 42.9%  
 Strongly Agree: 5 8 38.1%

I found the system very cumbersome to use.



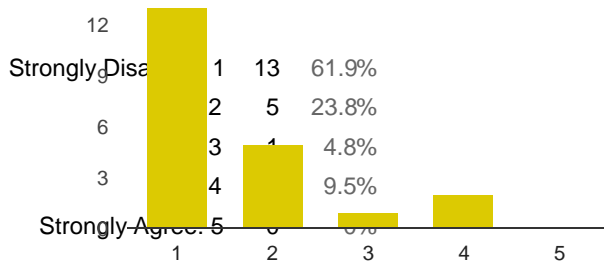
Strongly Disagree: 1 7 33.3%  
 2 10 47.6%  
 3 2 9.5%  
 4 2 9.5%  
 Strongly Agree: 5 0 0%

I felt very confident using the system.



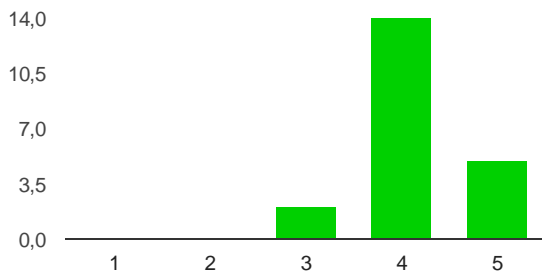
Strongly Disagree: 1 0 0%  
 2 2 9.5%  
 3 6 28.6%  
 4 7 33.3%  
 Strongly Agree: 5 6 28.6%

I needed to learn a lot of things before I could get going with this system.



## System Usability and General Performance

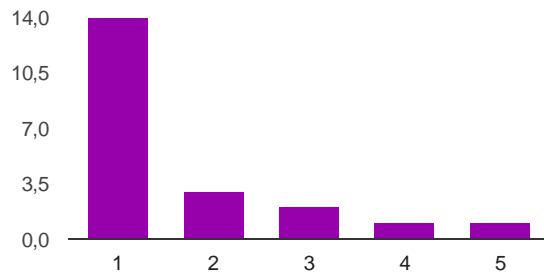
How do you classify the overall system's overall experience



Very Bad: 1 0 0%  
 2 0 0%  
 3 2 9.5%  
 4 14 66.7%  
 Very Good: 5 5 23.8%

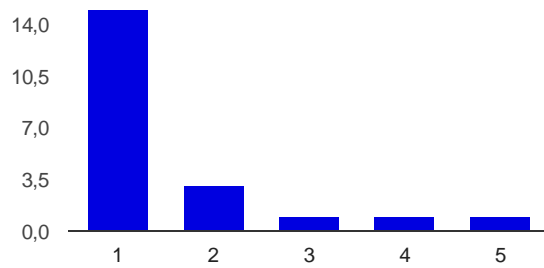
It is easy to upload videos.





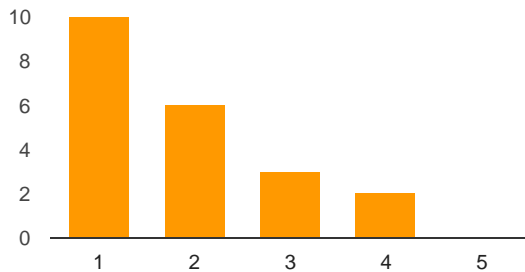
Very Easy:	1	14	66.7%
	2	3	14.3%
	3	2	9.5%
	4	1	4.8%
Very Hard:	5	1	4.8%

It is easy to create timelines.



Very Easy:	1	15	71.4%
	2	3	14.3%
	3	1	4.8%
	4	1	4.8%
Very Hard:	5	1	4.8%

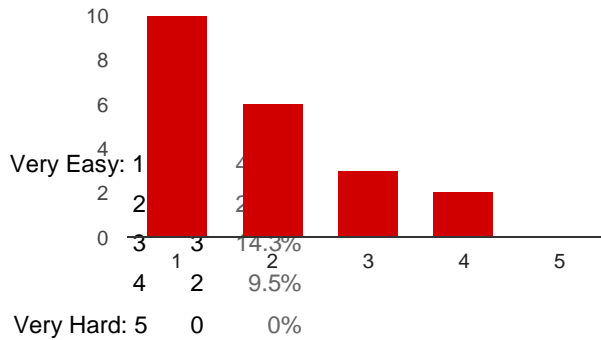
It is easy to create contests.



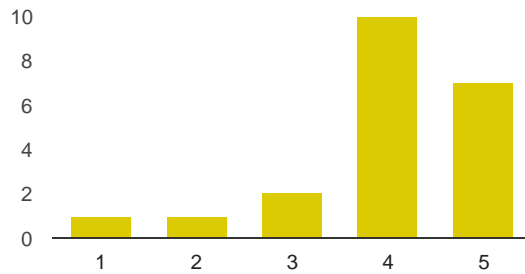
Very Easy:	1	10	47.6%
	2	6	28.6%
	3	3	14.3%
	4	2	9.5%

Very Hard: 5 0 0%

It is easy to contribute to timelines or contests.

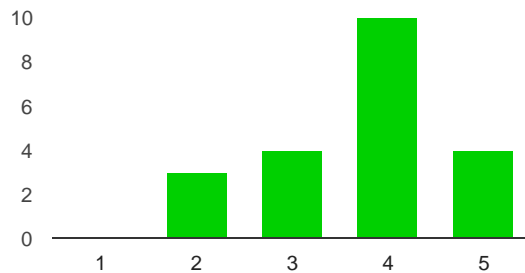


Timelines are useful.



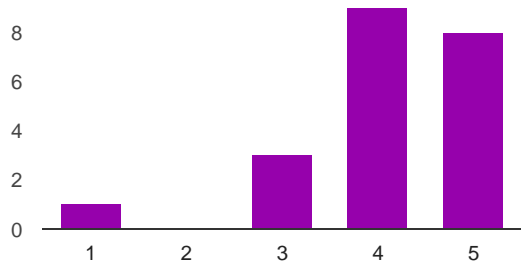
Strongly Disagree: 1 1 4.8%  
 2 1 4.8%  
 3 2 9.5%  
 4 10 47.6%  
 Strongly Agree: 5 7 33.3%

Contests are useful.



Strongly Disagree: 1 0 0%  
 2 3 14.3%  
 3 4 19%  
 4 10 47.6%  
 Strongly Agree: 5 4 19%

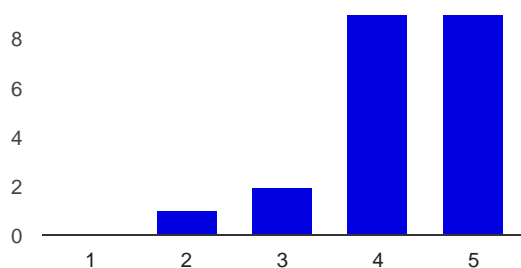
Competing among other users will lead to better contributions.



Strongly Disagree: 1 1 4.8%  
 2 0 0%  
 3 3 14.3%  
 4 9 42.9%

Strongly Agree: 5 8 38.1%

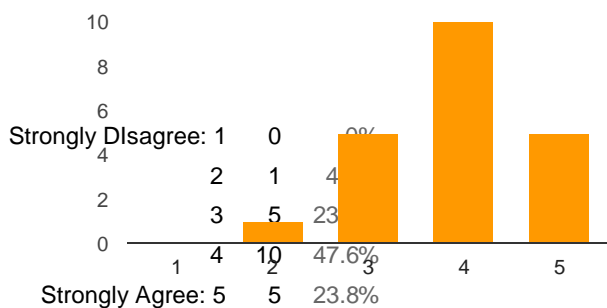
Twitter features are useful.



Strongly Disagree: 1 0 0%  
 2 1 4.8%  
 3 2 9.5%  
 4 9 42.9%

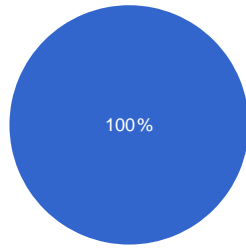
Strongly Agree: 5 9 42.9%

Leaderboards are useful.



It is easy to view the content in general.

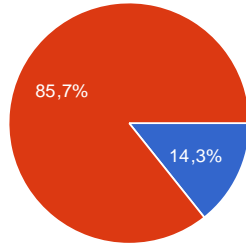
Yes 21 100%



No 0 0%

Did you experience any difficulty using the application?

Yes 3 14.3%



No 18 85.7%

If yes please detail them here

Dragging videos in the timeline in edit mode didn't work.

Há alguns videos que dizem "Formato de vídeo nao suportado ou tipo MIME não suportado".

Não consegui entrar com o Chrome.

the website is rather slow, sometimes

slow in Google Chrome

Any suggestions? Leave them now

1 - Os botões "Post Comment", "Create Video", "Add Video", "Add to Timeline", "Create Timeline" e outros que sejam para criar/adicionar coisas podiam ser verdes. É apenas um pequeno detalhe de interface, dado o simbolismo que o verde tem. E sempre adicionava mais cor à interface e aos botões que ou são azuis ou não tem cor. 2 - Não sei se é importante ou não mas podia ser bom acrescentar uma dropdown para ordenar os vídeos por categoria na página dos vídeos. Ou então ter checkboxes para as categorias e o utilizador seleccionava quais as categorias que queria ver.

Gostei muito do vídeo dos chinocas, a jogar pingue-pongue. Boa contribuição, muito útil!

Fill the video title in the same page of the upload window. Upload a new video directly to a timeline.

it would be more practical to be able to upload a video directly to a timeline, instead of having to upload it first and then add it to the timeline.

The videos in general, in my PC take a while to appear.I would like that to be faster.