



Senior Design Project

Project short-name: YouTalkWeSign

Low-Level Design Report

Abdurrezak Efe, Yasin Erdoğan, Enes Kavak, Cihangir Mercan

Supervisor: Hamdi Dibeklioglu

Jury Members: Varol Akman, Mustafa Özdal

Contents

1. Introduction

- 1.1 Object design trade-offs
- 1.2 Interface documentation guidelines
- 1.3 Engineering standards (e.g., UML and IEEE)
- 1.4 Definitions, acronyms, and abbreviations

2. Packages

- 2.1 GUI Package
- 2.2 Application Logic Package
- 2.3 Storage Package

3. Class Interfaces

- 3.1 Application Logic Package
- 3.2 Storage Package

1. Introduction

YouTalkWeSign will be a web application which is designed to serve people with hearing impairments or deafness. In today's world, there are many people who suffers from hearing loss and deafness. According to the last records of World Health Organization (WHO), there are 360 million people worldwide, corresponds over %5 of the world population who have disabling hearing loss. Moreover, they also claim that 1.1 billion young people, who have age between 12 and 35 years, are at the risk of hearing loss because of the exposure to the noise in recreational settings. The general reasons of the hearing loss are genetic causes, complications at birth, chronic ear infections, the use of certain drugs and growing old. People who have a significant hearing loss which also called deaf frequently use sign language to communicate with other people.

Sign language is a language which primarily uses manual communication to have a meaning, as opposed to spoken language. It is basically combination of simultaneous hand shapes, orientation, facial expressions and movement of the hands, arms or body to express ideas and communicate with other people. Sign Language helps to build bridge between people who can hear and who cannot. In addition to this, Sign language is not only used by deaf people, but it also can be used at some circumstances among people who have suffering from hearing and talking to each other such as very crowded or very quiet areas. Despite the significant role and 2 widespread usage of the sign language in human lives, it is not cared sufficiently and there are no enough resources.

With YouTalkWeSign, we aim to provide easily accessible source to create sign language of a video from YouTube. Thanks to the YouTalkWeSign, we are hoping that people who have disabling hearing loss can watch videos they like with understanding what kind of conversations and sounds are in the videos. It would also be a good resource for other people who do not have a hearing loss, but want to learn sign language.

In this report, general overview of the low-level design and architecture of our system is provided. We begin with describing object design trade-offs and the engineering standards. In addition, the interface documentation guidelines are listed. After the first part, we describe the packages of our system and their functionalities along with detailed class diagram views. Moreover, each software component descriptions and class interfaces are indicated in this report.

1.1. Object Design Trade-offs

1.1.1. Usability vs. Functionality

Usability and functionality are the main two goals of our system. On the one hand, we are providing functionality requirements of system such as speech recognition and transforming text to sign language. On the other hand, we are trying to keep our system as usable and user-friendly in order to make it easy to use for users. Some specified functionalities will be inaccessible, if they cause the system to provide an easy-to-use interface to users. Thus, we consider usability more important than functionality. We will be trying to provide more functionalities to users as possible as while keeping the system usable.

1.1.2. Cost vs. Robustness

Youtalkwesign fetches video from youtube and tries to convert text to sign language. If there are automatic subtitles of given youtube video, youtalkwesign focuses on only transforming to sign language, however, there are no subtitles of some videos. To handle this situation, we are using some free speech API for speech recognition. We are utilizing public shared speech APIs to convert speech to text. We aim to keep cost as small as possible. Due to using costless sources, our system may not be able to overcome some issues, since third party costless sources are not developed enough. Therefore, youtalkwesign may not be robust enough against some potential errors. We will try to handle these errors as much as we can.

1.1.2. Portability vs. Efficiency

Portability is a crucial issue for a software, since it provides that the software can be reached for a wide range of users. In this respect, youtalkwesign will be a web-based project and each person around the world who has internet connection can access youtalkwesign. Also, we are trying to develop youtalkwesign as much as possible as efficient for each user. There are lots of technological devices which have different performance and capabilities. Therefore, we aim to give more importance to portability than efficiency.

1.2. Interface documentation guidelines

In this documentation, the description of classes in our system includes class name and short information about class and its functionalities and then, properties and methods of that class are listed. For representing classes, properties and methods, we used `className`, `propertyName` and `methodName()` tags respectively. The sample outline is like below:

Class Name

-Description of class

Attributes

- Attribute name
- Type of attribute

Methods

- Method name
- Parameters
- Return value

1.3. Engineering standards

This report includes UML(1) design principles for description of class interfaces, subsystem compositions, scenarios, diagrams and use cases. UML is a common language for specifying, visualizing, constructing, and documenting the components of software systems. We represent software components, functionalities and system structure with using UML. This report also constructed with using IEEE citation guidelines for the references.

1.4 Definitions, acronyms, and abbreviations

API: Application Programming Interface

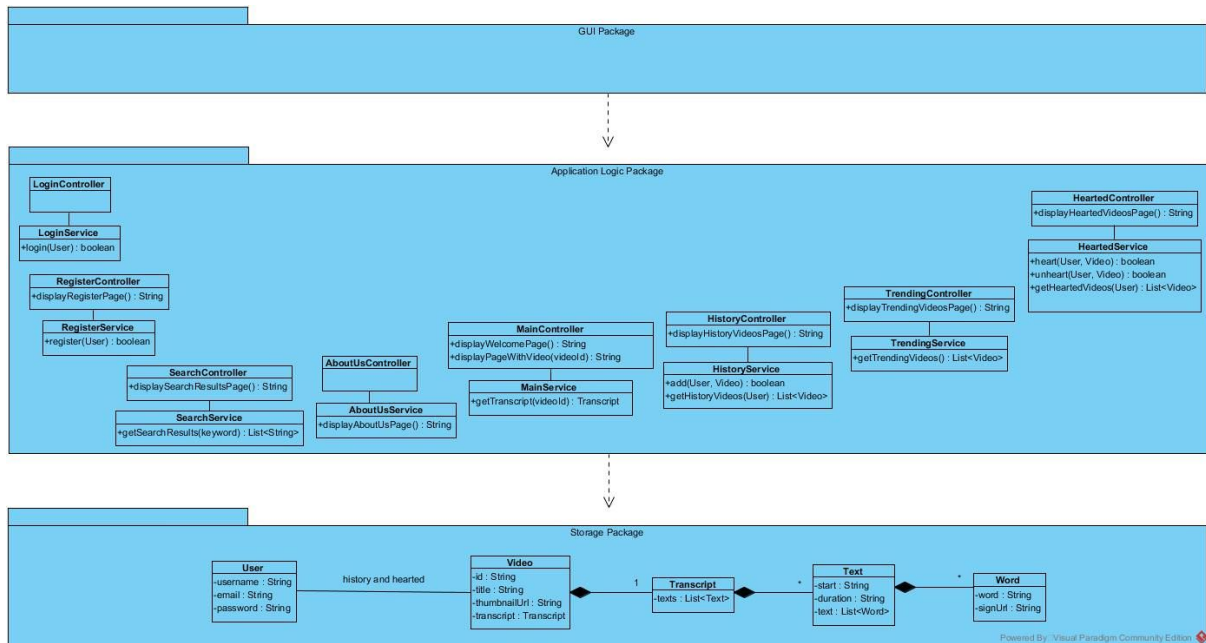
Client: The part of the system the users interact with

Server: The part of the system responsible from logical operations, scheduling, and data management.

UI: User Interface

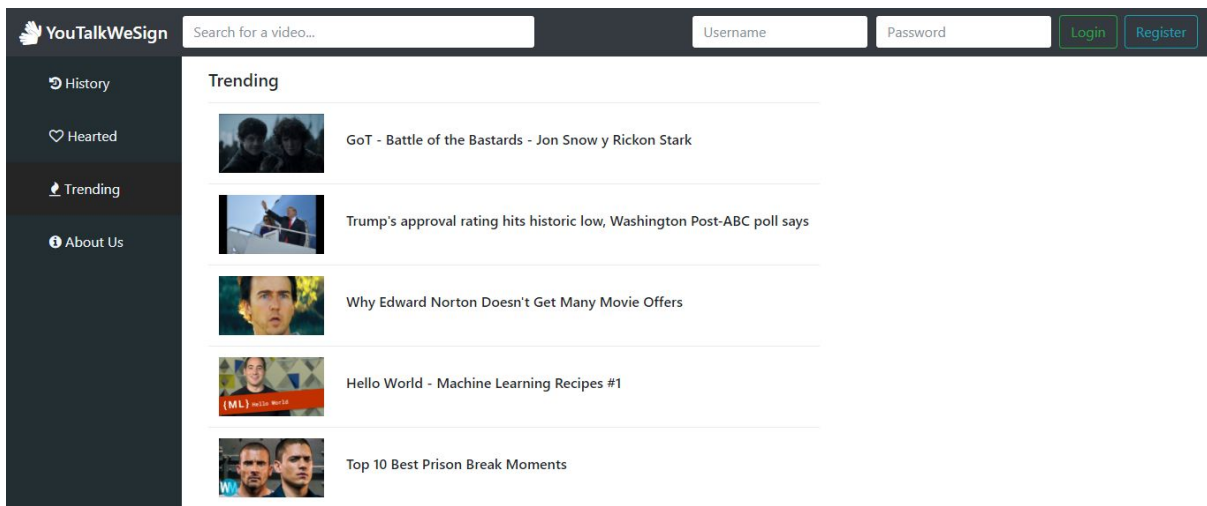
2. Packages

YouTalkWeSign's 3-package architecture is below.
(Full resolution: <https://i.imgur.com/KrmbqID.jpg>)

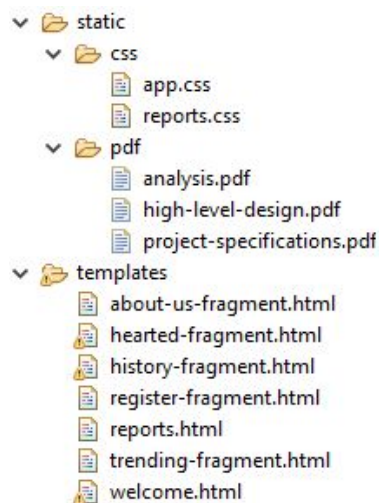


2.1 GUI Package

The GUI package consists of HTML/CSS pages that are powered by Javascript and Thymeleaf Java template engine. For the design of the web pages, Bootstrap v4 beta is used. Therefore, our website is full responsive and has a consistent design. In addition, for the purpose of being a dynamic webpage, we have benefited from jQuery a lot. Also, the icons that are used in our website is from the Font Awesome library. Below is an example view from our website:



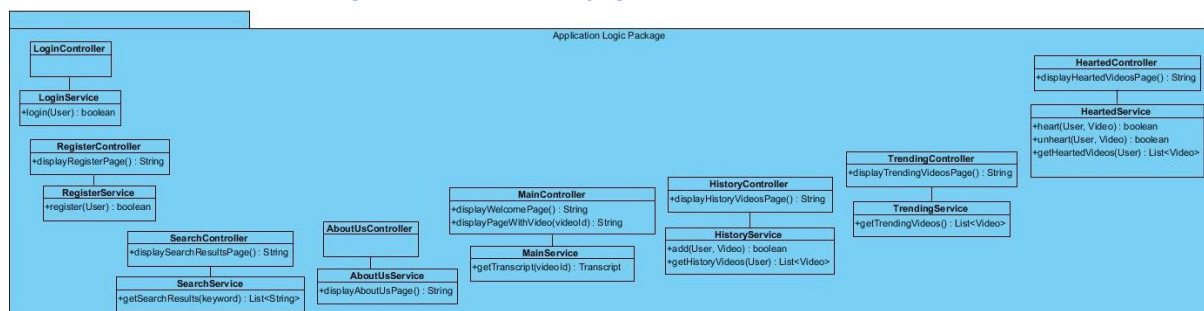
For each link at our website, we have a corresponding html fragment.



2.2 Application Logic Package

Our application logic package consists of Controller/Service pairs. In this logic, first, request (GET or POST) from the user comes to the related controller and controller catches it. Then, controller calls its service class to get the job done.

(Full resolution: <https://i.imgur.com/rLHEuPd.jpg>)



At this package,

AboutUsController is responsible for showing our project's information to the user when clicked to 'About Us'.

HistoryController is responsible for keeping user's watched videos at the database.

LoginController is responsible for authentication.

RegisterController is responsible for authorization.

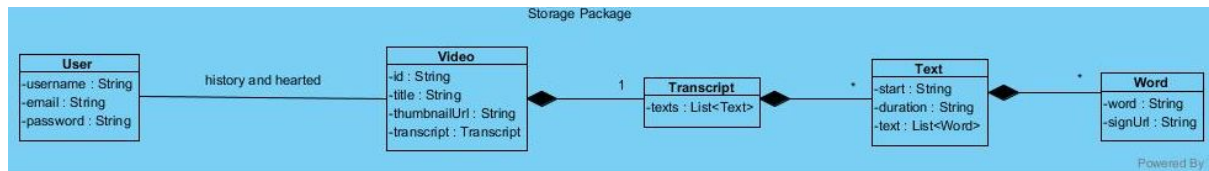
MainController is responsible for displaying the video with the corresponding sign language translation. It is the most important one because its job starts by getting the transcript of the video. Then, it prepares the words that are spoken in the video with the signs of them. Then, it administrates the GUI to make video and avatar play simultaneously.

HeartedController is responsible for keeping user's hearted videos at the database.

TrendingController decides the most watched videos at our website and keeps them at the database.

SearchController is responsible for bringing search results with regard to keyword.

2.3 Storage Package



At this package, first of all, all videos have a transcript besides their youtube id, title and thumbnail image url. Transcript consists of a list of Text objects. Each text object has a start time and duration from that time and the text that is spoken at this time interval. Then, the Text object will have a list of words which keeps words along with their sign video URLs. In addition, Users' relation with the videos will be kept at database; their histories and hearteds.

Example Transcript:

start: 13 **text:** imagine that you are a member of congress
start: 16.7 **text:** you have worked very hard
start: 19.1 **text:** you have knocked on thousands of doors
start: 22.2 **text:** sweating and shivering depending on the season
start: 25.9 **text:** you have made hundreds
start: 27.1 **text:** maybe thousands of phone calls to people you do not even know
start: 31.7 **text:** asking for their support
start: 33.9 **text:** begging for their money
start: 36.4 **text:** and now you have got one of these
start: 38.6 **text:** it is hanging on a door in washington dc
start: 41.4 **text:** it says you are a member of congress

3. Class Interfaces

3.1 Application Logic Package

class LoginController and LoginService
Responsible for authentication
Properties and/or Methods:
public boolean login(User)

class RegisterController and RegisterService

Responsible for authorization
Properties and/or Methods:
public String displayRegisterPage()
public boolean register(User)

class SearchController and SearchService
Responsible for bringing search results with regard to keyword
Properties and/or Methods:
public String displaySearchResultPage()
public List<String> getSearchResults(keyword)

class AboutUsController and AboutUsService
responsible for showing our project's information to the user when clicked to 'About Us'
Properties and/or Methods:
public String displayAboutUsPage()

class MainController and MainService
Responsible for displaying the video with the corresponding sign language translation. It is the most important one because its job starts by getting the transcript of the video. Then, it prepares the words that are spoken in the video with the signs of them. Then, it administrates the GUI to make video and avatar play simultaneously.
Properties and/or Methods:
public String displayWelcomePage()
public String displayPageWithVideo(videoID)
public Transcript getTranscript(videoID)

class HistoryController and HistoryService
Responsible for keeping user's watched videos at the database
Properties and/or Methods:
public String displayHistoryVideosPage()
public boolean add(User, Video)
public List<Video> getHistoryVideos(User)

class TrendingController and TrendingService
Decides the most watched videos at our website and keeps them at the database
Properties and/or Methods:
public String displayTrendingVideosPage()
public List<String> getTrendingVideos()

class HeartedController and HeartedService
Responsible for keeping user's hearted videos at the database
Properties and/or Methods:
public String displayHeartedVideosPage()
public boolean heart(User, Video)
public boolean unheart(User, Video)

public List<Video> getHeartedVideos(User)

3.2 Storage Package

class User

Holds user information

Properties and/or Methods:

public String username

public String email

public String password

class Video

Holds video information

Properties and/or Methods:

public String id

public String title

public String thumbnailUrl

public Transcript transcript

class Transcript

Holds all subtitles of the video

Properties and/or Methods:

public List<Text> texts

class Text
Keeps a list of words which keeps words along their start time and duration
Properties and/or Methods:
public String start
public String duration
public List<Word> text

class Word
Keeps a word and its sign video URL
Properties and/or Methods:
public String word
public String signUrl