Adapted implementation of WikiTrust into crowdsource sign language dictionary

Kenneth Andersen

Kongens Lyngby 2019

Technical University of Denmark
Department of Applied Mathematics and Computer Science
Richard Petersens Plads, building 324,
2800 Kongens Lyngby, Denmark
Phone +45 4525 3031
compute@compute.dtu.dk
www.compute.dtu.dk

Summary (English)

The goal of the thesis is to prove the possible to implement the design of WikiTrust into another crowdsource platform that has different content structure and still satisfy the minimal requirement of reputation system, where the users are enabled to evaluate the level of reliable in each contents.

The degree of satisfaction will be based on the parameter of transferability of reputation's measure mechanism, scalability that the output of reputations are sound independent of capacity of the users that are a part of customer segments and whether WikiTrust is adaptable in platform with low or no contents.

The thesis is based on an actual platform called Wign that is a crowdsource dictionary providing the suggestion of translations between written and sign language which at the time of writing is only in Danish. Wign has significant small customer segment that it easily compromised by the malicious actions when its existence is reached to the potential hacker, because its contents are public available and requires no user authentication. Then the thesis is about to implement the redesign work as an extension into Wign that is implemented in Laravel Framework. Wign's content is a collection of translations that consist of a composite of the following axioms, one string, one video and one text. Performing a partial alteration in video is not possible and generation of a video always is unique then the uniqueness has to be considered when measuring the positive reputation in this composition of axioms which is called a post.

Summary (Danish)

Målet for denne afhandling er at påvise at det kan sig gøre at indføre implementing af omstrukturering af WikiTrust ind i crowdsource platform med utraditionel indholdsstruktur, og stadigvæk tilfredsstilles den minimumkrav ved opstilling af omdømmesystemet, hvor mulighed for at kunne tage stilling til indholdets troværdighed stadig er tilgængeligt.

Vurderingsfaktor tages i udgangspunkt for de tre egenskaber, omsættelighed af barometer i omdømmet, skalerbarhed så påligehed i hvert indhold foreblive stabilt uden påvirkning af antal af tilsluttede brugere som udgør i målgruppen og egnet til omstilling så WikiTrust stadig kan være anvendelig i platform med tomt eller sparsomt mængde af indhold.

Specialet tager udgangspunkt i implementeret platform, Wign, hvilket er en crowdsoruce tegnsprogsordbog som tilbyder oversættelse mellem skriftlig- og tegnsprog, som i nuværende stund er kun tilgængeligt på dansk. Wign har forholdvis en indsnævret målgruppe, så platformen er yderst sårbar ovenfor udenforstående angreb. Risikoen bliver kritisk så snart nogen potentiel hacker får kendskab til platformens eksistens, idet platforms indhold er åbnet og kræver ingen autentificering af bruger.

Som i kølvandet for mangel af beskyttelsesmekanisme drejer specialet derfor sig om redesign og implementering løsning som en udvidelse til Wign. Wign er udviklet i Laravel Framework. Samling af indhold i platformen består af sammensatte aksiomer hhv. en ord i kort streng, en video og en lang tekstform. At foretage en rettelse i video kan netop ikke lade sig gøre, hvor generation af videoen opnås sin unik som skal tage i stilling, når der skal bestemmes grad

af positiv pålidelighed i kombination af disse aksiomerne, kaldt opslag (post på engelsk).

Preface

One day in 2009 I was frustrated that I cannot just focus on my study at technical college (HTX) where I studied biotechnology and mathematic, because there were many translation of technical terms were not reachable. At this time there were many other sign language user who had passed their youth education program then anyone should have the collection of translations in technical terms that I can use. Unfortunately the sign language interpreters have duty of secrecy then they could not give me the name to previous students. The interpreter providers had their own internal collection of translations that primarily were only expanding by their employees (the interpreters) and is inaccessible to outsiders. It lead to the translations become a commercial value that the providers use to let the potential customers to order their interpreters. I expressed the issue and my frustration in a blog and it became a project with few stakeholders which Wign was born as a platform of sign language encyclopedia. Today Wign is running as entrepreneur company running by me and my partner, Troels Madsen from IT-University in Denmark.

During my education in bachelor and master level I often compare the approaches I learned at each courses with Wign, how the platform could be improved, for allowing to understand how the approaches can be implemented in reality. Then when the planning of master thesis should start, I was very determined that my master thesis must based on the connection between theories with the practices with a cooperation to a company to obtain my understand how the best practice will look like in the work life.

The search of cooperation did not succeed, thankfully the tutor for this master thesis, Christian D. Jensen, was ready to setup my master thesis with cooperation in Mathematical and Computer Science at Technical University of Denmark

(DTU). He wrote a scientific journal with Kasper Lindberg about enhance the trustiness of contents in Wikipedia that is called wikiTrust that he wants somebody to do the implementation. Then the topic becomes a combination of implementation of wikiTrust and my wish to include a company, so the master thesis is about to redesign and implementing the wikiTrust with different structure of context where the video acts as primary source instead of text.

This thesis was prepared at DTU Compute in fulfillment of the requirements for acquiring an M.Sc. in Engineering within Computer Science with Christian D. Jensen as tutor and Wign IVS as improvement of their business case.

The thesis deals with redesign of a text-based reputation system into composite-based reputation system that is implemented as an extension in Wign's deployed platform that is written in Laravel Framework.

The thesis consists of working process of redesign which is documented in the paper and source code that acts as test environment then it is runnable on local machine.

The readers are expected to have the insight of basic approach in computer science and have some experience with programming work. The knowledge of topic within computer security may improve the understand.

Herlev, 1st January 2019

Kenneth Andersen

Acknowledgements

I would like to be grateful to my tutor Christian D. Jensen for supporting to find a collaboration with a company that I want to do my master thesis for getting more experience of best practice. Unfortunately it was not succeed, and thankfully the tutor was ready to offer me to be tutor for this project as collaboration with DTU and accepted my suggestion to include my company as real case for the programming work. The meetings with the tutor were very useful that they always made me more clear about the further work.

I want to give a huge thank to my friend, Brandon Marin from California, USA. In the beginning of my long summer trip in California, Brandon was always ready to ensure that I can connect to internet when necessary to complete my project description when we were far from any cities during our travel. He also provides himself to proof reading the project description before I had chances to ask him.

Troels Madsen was this one who developed the current Wign platform. We are owner in the company Wign IVS, but because of the allocation of tasks I never did the programming work, so I want to thank to him for being willful to introduce me the php and Laravel framework that I never worked with and the further support for many minor debugs that were appearing during the thesis.

The thankful also goes to Coilin B. P. Jeritslev for allow me to use him for sparing about the approach of mathematic modeling that I want to obtain the appropriate property by simple mathematical formula that decides the number of ballots that have to be generated based on number of total users in each voting (more about that later in the thesis).

I want to send my thankful to my friends, including Brandon, Troels and Coilin too that were interesting about what I worked with the thesis and listing interesting to my considerations and reflections. I benefited myself by getting new idea how to describe the technical approach(es), found a solution myself by spoke about that or become better to formulate the statements in the thesis. It also benefit my motivation, so I want to thank them for that.

I want to express my gratitude to my interpreters Stella Lund and Rikke Plett for their dedicated and high qualified work as interpreters when the topic in my master thesis is based on my entire master study which required them to get the idea what the subject is about for enabling offer the high quality interpretation during the meetings with my tutor and my defense.

A thank also goes to CFD Tolkebooking for providing the stable deliverance with a team of fixed interpreters during my 5 and half years of study, when the working conditions as sign language interpreter is concerned lately in Denmark. It entails to a high quality interpretation at the lectures with dedicated cooperation and preparation.

A last and big thanks goes to my family for their love and support.

Contents

Chapter 1

Introduction

Currently the information technology still is often used as tool reach a solution of the problems that exist in the Western World since the personal computer starting to be a part of the citizens' life and no longer only were used within the research community. Throughout the Information Technology Era more and more solutions of the issue are reachable by using the IT. IT often will be used in the community if the approach is well-proven or there is a well-defined model that was successfully implemented into a system.

Today people are trying to implement an IT-system with a part of human skill, ie. reasoning thinking that is based on its own opinion, social and behavior, and language/communication skills, and they all still can be improved. The technologies only can make the decision that is based on the exist knowledge which the machine can derive to a new conclusion by combining those knowledges to obtain a new knowledge. A IT-solution is almost always developed in a closed system assumption, therefore we see that contents in a broader collection of knowledge, ie. Wikipedia, still are produced by human. The approach in collection of sub-informations into one place by humans is known as crowdsourcing[EANGdG15].

Crowdsource enables the users to reach the synthesize effect by combining their knowledge independent of each other to obtain the information about one specific subject. Unfortunately there has been issuing to keep the level of reliability

high among the users in the case of Wikipedia, when any one can generate or alter any contents without the need to validate the background of this user actual match to the content the user wants to contribute (or destruct in the manner of malicious purpose). Those users who can evidence as an expert within a field, cannot be sure that they also act as an proper user. Somebody obtains the knowledge through formal education, but other ones may also reach the different kind of this knowledge through their practical or experimental experience which not always can be documented on a certificate. Instead of using the certificate for obtaining the authorization to perform the contributions, the solution will be based on the reputation system designed by the registered users in the system which the thesis will address about the approach how to satisfy the requirement about reliability.

The thesis is going to define how the reputation system developed by Lindberg and Jensen[LJ12] can be remodeling for allowing to adapt into a platform called Wign[1] that has different structure of the content. Wign is a crowdsource sign language dictionary that currently is available for the contribution of translations between Danish and Danish Sign Language. The thesis will address the discussion whether the remodeling of the reputation system can be satisfied with the following properties;

Transferability - The contents in Wikipedia are purely based on text, and Wign contains a lot of unique combination of independent medias (video, string literal and text) then it has to address if the reputation system also can be implemented in the system with different structure of the content.

Adaptability - All new systems always start with empty or lightweight content, then the approach will take a look if the reputation system is usable in this case or reputation system makes more sense to be implemented in system with well-developed contents.

Scalability - The idea of reputation is depending on the size of active users, and in this case that Wign has very small customer segment. Is it still possible to implement the reputation system and still satisfy the degree of reliable among the registered users in Wign that they can consider all the contents to be valid?

Deliverance of the thesis is this report and the source code for the entire implementation with the extension of reputation system. There also follows a short documentation how to run the implementation on local machine by running the makefile.
There will be several parts in design that were not implemented in the platform

[1] `http://www.wign.dk`

because of limited time during the master thesis work. The missing parts will be presented in detail in implementing section.

The paper follows the working process of best practice within the field of computer science. Later in this will goes in deep about the author's preparation and executing and own reasoning about the deliverance. Then it ends with a collection of term clarification for ensuring that the readers understand what the author is trying to communicate in the paper when mentioning those terms.
Second chapter will enlighten the research papers or the existing works that will based on the further discussion in the paper.
Third chapter is an analysis part that will consider and deliberate the different problems that will appear as follow of implementing the reputation system into Wign in the aspect of three properties mentioned upper in this section and the discussion of how to maintenance the secure policy for avoiding the vulnerability that may occur during the design or implementing work. Those critical parts will be discuss further in the next chapter. The chapter will concludes with a section of requirements specification.
Fourth chapter explains about the work of extension of the existing system that is running on Wign's server, so the reputation system can be added. Each parts of the extension will be goes in details how they will be structured and implemented so they do not violate any the requirements and still holds the properties.
Fifth chapter will describe how the author did solve the approach in the implementing level in specified parts in the source code to make the further maintenance work easier.
Sixth chapter focuses on the evaluation of outputs and behavior in the platform are as expected and satisfy the requirements.
Seventh chapter is a discussion section that will take those parts of implementation from previous chapters in consideration.
The report concludes with a summary about the thesis work whether the requirements are obtained by the implementing work and with some suggestion for further works to improve the obtainability. There are several appendices after the conclusion section that contains optional materials for reading i.e. "10-pages" about Wign and documentation how to run the platform on local machine.

Term Clarifications

Sign language users refers to the people who have one or more sign language(s) as a part of their mother language or is able to communicate in this sign language freely. The people includes both deaf and hearing people.

Post in the platform mean that one post is an unique set of a word, a video and a description that works as a suggestion of translation between written language and sign language.

Sign has the equal mean as semantic expression that a sign can interpret as a word in written language. In this case with Wign 'sign' is a combination of a video and its description that the description is used to clarify the definition or other that may necessary to inform the user if they want to use this. "Sign" will never be referred as a frame of text or pictogram used for the information purpose in public area in this paper.

Chapter 2

State of the Art

2.1 Wign

Wign is a crowd source platform that provide the sign language users the opportunity to exchange and search for the translation from a written language to sign language. Currently Wign only is available in Danish. There is roughly 20.000 people that speak danish sign language, including 4.000 deaf people and then their family members, professionals (teachers, interpreters and so on) and people who learned the sign language [Ass]. The launch of the platform took the place in 2012 as the following of a project, because at this time all companies which provider the interpreters, also developed their gloser collection between Danish and Danish sign language that only are accessible by their employees, in few cases also for their customers (deaf people). Then every time a deaf person starts at a new study, they often have to agree what to name every single technical terms in Danish sign language, because they either do not know who holds the gloser of technical terms or the interpreters have client confidentiality then they were unable to transfer the gloser for the "reuse". Then there was a normal that deaf students have to also invent new translation for those terms, no matter whether another person may have the gloser of translations. Unfortunate the platform had to close already a couple years after the launch because of poor code maintenance. The platform was reopen in 2016 with new developed implementation, then some of videos are the only sources that were transferred

from the old platform. Therefore there are not many data that can be tracked back to 2012 (Appendix C).

The platform has two main functions, **post** a suggestion of translation and **find** a translation. A post contains a video that shows expression of this term following with a word and a description as optional. The users can add more expressions to the same word as either additional way to express this word or as correction of expression that was expressed wrongly in the previous one video. Beside post and find-function, the users also are enable to like a post as indication that they also use this sign themselves or acknowledge the correctness of this video. If a word contains more than one video, the videos will be sorted in the order of likes that the video with most likes will be present on the top and least likes in the bottom.

Right now the users are unable to alter or remove any contents made by other or themselves, it is done to protect the content against both unawareness and on purpose malicious attack. The login-feature was not available right before the begin of this thesis work which expose to the malicious attacks when the intruder explored the vulnerability. However the users currently are allowing to report a post by flagging which the post will be removed from the word-page temporarily and will be check by a moderator for the further decision.

It concludes the following features that are accessible by everyone on the platform:

New sign Create a new post in new or exist word from the list of requested words with a video following with a description.

Search The user can use the search engine by type the word or tag keyword or looking at the list of 25 word that were posted recently or the list with all words that contains one or more post.

Request If the user searched a work and get no results, the user is offered to add the word to the list of request words that will be visible to anyone.

Report The user can report a post, if it is inappropriate or contains some significant mistakes and the post will be removed temporally. When reporting the user is asked to fill the reason from a menu, explain in text in detail and enter the e-mail address for the further contact.

Like Each post has the like-button that the users can click to tell that they are using this sign. The post with most likes, will present on the top of the list of posts for this word.

2.1.1 Existing Business Logics

The current deployed Wign was developed in 2016 and later was extended with hashtag feature in descriptions and enable to find them through the search engine on main page.
Wign contains the following business logics:

Word that contains a string literal as key word for the search engine. One word can works either as **requested** word that the user is looking for a sign for this word or a word that has one or more signs that are referring to this word as a **translation**.

Sign is an axiom expression form in a sign language which is presented by a **video** with the following of a **description** as an additional to clarify the definition of this sign if the word is not sufficient. Each sign is follow by a set of **likes** that is added by users according to their IP-address. The sign with most likes will be present top on the list of signs and the sign with least likes can be found on the bottom. Since there was no login then anyone can submit a new content to the platform, then there exposes the vulnerability to generate an improper content, either in text or video. Therefore the post can be **reported** by anyone, and it will be hidden temporarily until the admin decides the further action.

Tag is a collection of keywords that appear in description of the posts that the word begins with a hashtag for allowing to browse them in the search engine.

Blacklist is a list of IP-addresses that were added manually when they act against the policy or show clearly that they obvious are not a part of the customer segment but have made an worthless contribution (rambling with their hands in the video). They will be banned from the platform if the IP-address matched.

2.1.2 User Traffic Report 2018

The platfrom was relaunched in 2016 with empty content, then the first phase people did not use the platform as tool beside posting new signs. Then the user behaviour has been changed lately, so to make sure that the report makes sense, the time interval starts at 1st January 2018 and ends at 1st September where the report was exported from Google Analytic that Wign is using to track the sessions performed by the users.

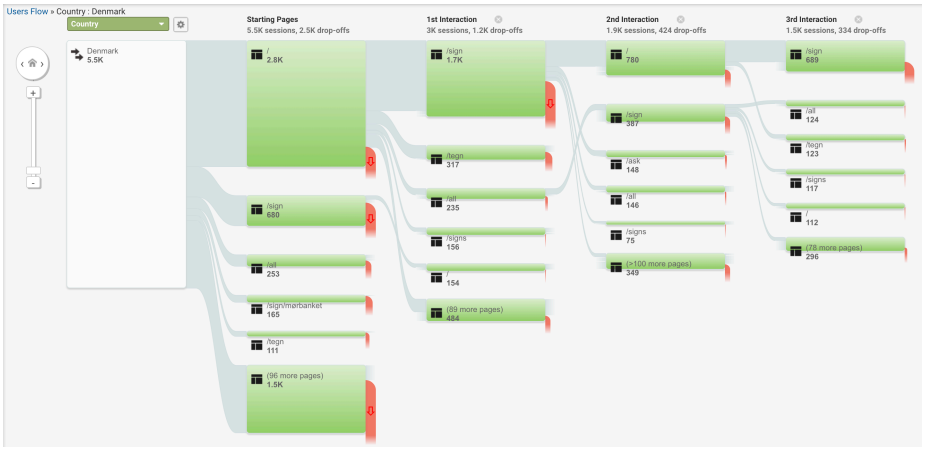


Figure 2.1: Overview of the userflow in the platform by the users with Danish IP-address

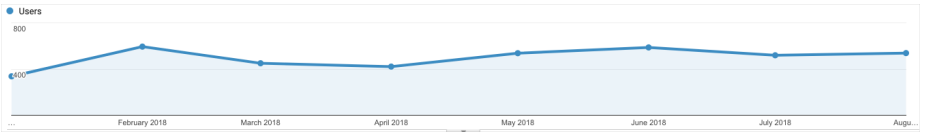


Figure 2.2: Number of visitors every month in 2018 (Jan - Sep)

This year there have been 3007 unique ip-addresses that had triggered some sessions in the platform and the users have 2,11 sessions in average per visit. The number of sessions can interpreter that mostly of them are entering the staring page and type a search word to find the sign and watch expression of the sign if the word is found in the database, otherwise mostly of the user will terminate the session after that (figure 2.1). Every day there is 22,5 visitors in average (or 17,5 visitors measured in median), so monthly visitors ranges between 400 - 600 in 2018 (figure 2.2).

2.1.3 Current Security Model

Wign works only on integrity in this current aspect of CIA-model[Cab15] then the users only have the access to create a post, then the established contents will not be comprised with the accessible features. When an user wants to correct an existing content, they can either write to us in e-mail or send a message through Wign page on Facebook. The admins perform the change directly on the mySQL database. The changes will not be stored in the database because

of the current structure does not support storing the session history. There happened few times that the user accidentally upload the post with misspelling or undone work, they then will report the sign by pressing the flag-icon which the post will be removed from the page temporarily. The user usual type in the comment when reporting that the post was not ready to make public and they create one new post themselves. Unfortunately in few cases the user misuse the report-feature to remove a post that this user may be disagree how expression of this sign should be, or because of unawareness about own sign language one may think that is only one way to express whenever there is a range of variation for this mean. Currently there are two owners that act as administrator with the right to perform CRUD-actions (create, read, update, delete) which will do the further action when people report a sign.

Blacklist is implemented in Wign to ban IP-adresses of the visitors that performed malicious or untoward actions. It is however unclear if the blacklist has improve the credibility, because Wign is using Amazon Web Service which has built-in security policy that prevent against DDOS-attack or other that can be detected through the user traffic to Wign. The blacklist will be updated manually by the admin because the the malicious actions happen on the video which will be in the level of semantic that only will be caught when watching this video or an user has reported this video.

The users are currently unable to alter any posts, because the platform does not support the user login-feature. The integrity will not break, because the users only can create a new post and anyone can report to make the post invisible, but the administrators will check and make it available again if the report of this sign is unreasonable. The integrity however only can be break by the administrator if user of a sign is not agree about the correction made by the administrator, but the user id unable to recreate to the previous content.

The platform only is tracking the users through the IP-adresses, but they never will be asked to inform their personal information beside when reporting a sign that they must enter their e-mail. Then the users are enable to be anonymous when watching the videos on the platform. The users are allowed to post anonymous sign, but the anonymity only apply to the information about name, e-mail of this user, but face will be public available.

Wign has to critical requirement to the availability, but with poor degree of availability will lead to less user traffic which is undesirable. In the case when an user want to show a page with a list of results (request words or words that contains one or more posts), the page will make the request to get all words that exist in the database and show them in one page with any containments. There is also possible to ask Wign to show all signs that has this tag in their description. A malicious user can perform DOS-attack by generate a lot of

videos with same tag and then repeating query to show them in one page then the server at Amazon will be burden by transmitting the huge load of data that contains data about all videos.

Since Wign does not have authentication process, it is not possible to make an access policy for the users. Then anyone can watch videos in Wign which may not be suitable with EU's GDPR, because without the authorization anyone can access to the video which contains some personal information. The malicious users can, with the right knowledge and tools, extract the video and publish in anywhere on the internet without the permission from author of the video which violate the protection of integrity that the video was used in wrong context.

As mentioned that Wign is developed in Laravel framework which is an extension of php-language. The protection mechanism against i.e. XSS attack or SQL injection is integrated in the language and Laravel still is releasing new version for improve the framework and currently the exploit of vulnerability is not an issue at these days. This time of writing Wign is using Laravel version 5.7, and version 5.8 is available and in the documentation of upgrade does not contain the serious patches[1].

2.2 Platforms of Sign Collection

There are many dictionary-like platform in each sign languages in their country. There is the follow kinds of dictionaries depending on their purpose and how they running and expanding their dictionary.

Research-based dictionary is running by researcher and therefore the dictionary only is expanding with new sign by the employees. The economic ressource usual is limited, then the expansion process is running slowly or frozen. However the contents are highly reliable.

Commercial dictionary is running by a business that have employees to fill the dictionary up with the translations themselves and the external contributions are excluded. The reliable level may vary much depending on their reputation whether their work are valid by the mother language users.

Crowdsourcing dictionary may be similar to commercial dictionary, but instead of excluding the external contributions the crowdsourcing is open for any contribution. The dictionary can be controlled either by registered users or the administrators.

[1] `https://laravel.com/docs/master/upgrade` (accessed: 27/11/18, 16.26)

Internal logging is primarily used by the sign language providers for exchange the information between sign language interpreters' colleagues when several interpreters are working for the same customer, then the interpreters want to exchange the information to the next interpreters with the new translations that were agreed between this interpreter and the customer which the interpreter will upload to their internal system . The internal system has its purpose to improve the quality of interpreter's work when they can be more prepared and be aware about the topic or situation at the place of assignment. The internal system often is secret and only is available to company's own interpreters.

It is difficult to know specific number of internal loggings there exist in each countries, because there is no qualitative or quantitative documents about the use internal logging at interpreter providers and none is telling openly about how they approach the exchange of information between their interpreters in public media. But since all providers with several interpreters should have their own internal logging system and only is available to their customers. Then it can conclude that there is many internal logging systems compared to the three other kind of dictionaries. To run own internal logging system often is affordable by the providers because they get paid by the customers, either private or governance customer, then they can setup the logging system and the interpreters will use the system regularly at work. Their system are only for their interpreters and their customers. If one customer decided to change the provider, the customer also will lost the access to the logging system, then every work of collection is unavailable, then the customer has to start all over by introducing the new interpreters the terms in this sign language.

Research-based dictionary has high cost revenue because of the requirement to hire the academical employees, collect or produce the sparse data from the medias (videos in sign language), analyse the data and publish the results on dictionary require more time because working with video takes longer time than when working with written language. Therefore the research-based dictionary is low performance and high cost.

Commercial dictionary har better competition condition compared to research-based dictionary because of no required to hire high-paid employees and less requirement to time consumption on the research work which commercial dictionary instead can focus on producing the contents themselves. One of the drawnback by letting the employees do the job to feed the platform with new contents that only based on their vocabulary that may not contain all the local dialects or technical terms. If the dictionary is open up for the external contribution to contact the employees with the terms, there will be risk of bottleneck, if the platform only can be updated by their employees.

	source	claim	target
singular	▶	—	▷
plural	▶▶		▷▷

Table 2.1: Definition of the reputation graph

Crowdsourcing dictionary was a solution to non-transparency when there were many internal logging systems with duplicates without somebody may know that, avoid the bottleneck when there were too few employees to handle the incoming contributions and make the cost low to collect the term.

2.3 Reputation Systems

The reputation system is better known as an extension of crowdsource platform that will enhance the reliable in each content that is generated by users that its background may not be authenticated. Instead of requiring the user to authenticate itself, other users can perform a mutation authentication on each other to obtain a certain degree of authorization or reliability.

The book about the web reputation systems[FG10] uses a custom graph that is not available to type in latex, then in the paper we will use the symbols of semantic:

$$\blacktriangleright\!\!\longrightarrow\!\triangleright \tag{2.1}$$

where there are three parts in the symbol (table 2.1).

Source always is a subject (user) that perform an action that interprets as a claim on a target that either can be an object (post) or a subject (reviewing other users). The plural can be used as a collection of individual reputation statements that can be a collection of likes on a post $users \overset{like}{\blacktriangleright\!\!\blacktriangleright\!\!\longrightarrow\!\triangleright} post$, or an user performs the same action on multiple targets which will be experienced with the voting system in the platform $user \overset{votes}{\blacktriangleright\!\!\longrightarrow\!\triangleright\!\triangleright} votings$. The reputation statement also can be combined in the way that other users can report an user based on its contributions $users \overset{report}{\blacktriangleright\!\!\blacktriangleright\!\!\longrightarrow\!\triangleright} (user \overset{contributes}{\blacktriangleright\!\!\longrightarrow\!\triangleright\!\triangleright} posts)$. The claim "contribute" can understand as a opinion statement "I contribute with a post because it is a correct post", but instead of long name of the claim, it can be simplified to "contribute". The contribution on post can be done in many ways, but this one that is visible to other users will be generation or alteration of a post, then

"contribute" can be divided into two conjunction of reputation statements that the report is based on $users \overset{report}{\blacktriangleright\!\!\blacktriangleright\!\!\rightarrow\!\!\triangleright} (user \overset{submits}{\blacktriangleright\!\!\rightarrow\!\!\triangleright\!\!\triangleright} posts \sqcup user \overset{edits}{\blacktriangleright\!\!\rightarrow\!\!\triangleright\!\!\triangleright} posts)$.

Users in the platform have their own purpose or ambition to benefit the outcome they receive by the platform. Only 1% of the entire user community will act as creators, 10% (include the creators) are synthesizers. All users in the community also can be called consumers[FG10, ch. 1].

2.3.1 Implemented Reputation Systems

The implementation of web reputation system appears in several global well-known webpages they have their own benefit of using the web reputation system in their platform.

Facebook

Facebook is one of world's most popular social media in 2018. The users can logged in the social media to read about what other people are doing or posting.

Facebook uses the reputation system implicitly to calculate the value in each post. If the post gets some kind of those feedback (figure 2.3), the algorithm will assume that this post has an exciting content that the users will get a nice experience when they are reading many great posts. So the algorithm will make the post appears repeatedly on users's newsfeed to keep them stay or revisit Facebook.

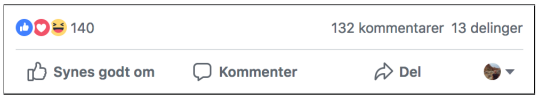


Figure 2.3: User's feedback on this post in Facebook, where this post got 140 different kinds of likes, 132 commented this post and 13 shared the post on their page

Wikipedia

Wikipedia is crowdsource platform that has a large collection of facts about anything. Wikipedia is having their issue that users can alter mostly of the

contents without any restrictions then some malicious users did damage some content by either deleting or writing fallacious facts.

Wikipedia has different level of protections (figure 2.4) that the lock is added or removed by the administrators[Wik]. The access control is managed by the roles that Wikipedia has the following roles that have different degrees of authorization: Unregistered/Newly registered, Auto-confirmed/Confirmed, Extended confirmed, Pending changes reviewer and Admin.

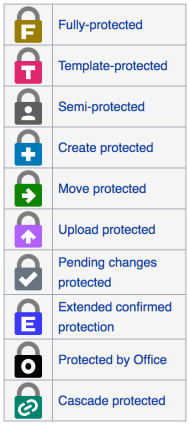


Figure 2.4: Different kind of locks according to level the author shall be for allowing to alter the content

Stackoverflow

Stackoverflow[2] works as a forum within computer science, mostly for the programming work. The users can create a post as a request for the help to solve bug(s) in their program, and other users can contribute with their suggestion to the solution and they also can review each other how well their solution works and the communication skill.

The information flow within reputation system are pretty explicit, since all users will enable to read the summary about this user's reliability, both its amount of contributions and review received by other users. Figure 2.5 shows that this answer has 334 up-votes that is the largest number among the other 20 answers for this request. Small profile-icon on right tells that the user has 11,771 reputation score, 2 gold, 24 silver and 26 bronze badgesThe comments below is a discussion of this answer.

[2] `https://stackoverflow.com/`

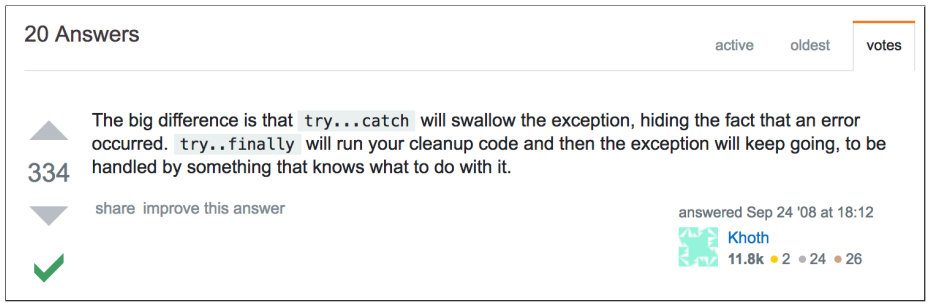


Figure 2.5: Screenshot of best up-voted answer in this request

.

2.4 WikiTrust

WikiTrust is a design of the reputation system that is extended from the existing implementation in Wikipedia so it introduced with a level system on registered users and documents as authorization mechanism. The voting also is introduced in the design, if level of an user is lower than level of the document then the user is unauthorized to alter content in the document, and the alteration will be sent to the voting to determine if the alteration is good enough to overwrite on the document. WikiTrust introduces the terms, Quality Confidence Value (QCV) as user's level and Integrity Level (IL) as post's level of reliable. The terms are a parameter to determine if reliable of the user is high enough to alter the post.

Post's IL can be changed by the author's QCV that IL cannot be set higher than this QCV. User's QCV only can be promoted or demoted at each single step, but the change of QCV will be decided by other users. The voting will be used when there has to vote about a pro- or demotion, additionally also alteration of a post, if author's QCV is lower than post's IL. All votings have the level of origin that is taken from the target post's IL or target user's QCV. The voting will invite users that are in the following levels when the origin is level L_i

$$L_i,\ L_{i+1},\ L_{i+2} \tag{2.2}$$

After the collection of votes has terminated, there will begin the count of votes to determine whether the voting is approved or declined. Since there are involved multiple level, therefore each voter gets the weight in their vote according to their level that weight from each levels must be $\mathcal{W}_i < \mathcal{W}_{i+1} < \mathcal{W}_{i+2}$. Authors of WikiTrust paper emphasize that highest level has more power of influence in the voting that they can decide the outcome of the voting without depending on other levels. It is not infelicitous distribution of power, then they also modify

the voting process that number of voters from each level Λ_{R_i} must be $\Lambda_{R_i} > \Lambda_{R_{i+1}} > \Lambda_{R_{i+2}}$.

To let the voting concludes with a approval, the result must decided by the weighted majority. The majority will be defined by a threshold that is a specific share of the weighted votes in total. The threshold musts be higher if the importance of the voting is higher, where higher level indicates as more important for ensuring that no malicious users will get passed easily.

2.5 Concept of E-voting

The voting is a part of democratic decision process that ensures that everybody are involved in the decision makings. The physical voting can happen as public or secret ballot. We only will discuss the property in a secret ballot. In the physical ballot the secret ballot can be satisfied with no much work, because there are enough witness that ensures that the box with ballots stay in its place and ensures that no ballots will hidden away from the counting. The voters also can ensure to keep their vote secrecy because all the ballot papers are identical and untraceable and the physical law has the theory of entropy that will create the unique randomness so the order of incoming ballots will not be in the same order when they are sent to count. The only challenge in the physical is to validate the result from the counting, because it is humanity to make mistake.

The properties will be a challenge to implement in the e-voting, because there is no entropy in the computer. If the traceability has to remove to compensate the anonymous, then there cannot to be sure that nobody does not cheat the voting by vote multiple times or alter the voting because there is no witnesses that follow the entire process of secret ballot, but only by software and hardware. The only benefit by the e-voting is validation that will be obtained fast and sounded if all the ballots are valid.

Implementing an e-voting will require the consideration which properties that are most essential and compensate other properties that may be difficult to obtain.

2.6 Laravel Framework

The framework is used to implement Wign, where there contain all the necessary components for allowing to include all the features into one project which goes

from execute the source code as test or deployed environment, interaction to the database, web protocols and implement of the core functions and views. The programming language is primarily in php with some custom extension, ie. translation to SQL query so programmer can write php-language as SQL-query, HTML and Java Script.

2.6.1 Model-View-Controller

According to Laravel Framework they uses MVC structure. However if we take a look in the default source code in a fresh project, there are many other folders than MVC, where models and controllers are stored in app-folder and views are stored in resource-folder. The controllers do not communicate with the views, but goes through route. Then the framework is a modification of MVC.

Chapter 3

Analysis

WikiTrust is a security model that based on the real behavior on the web that is open for any contributions from users on Wikipedia that has large number of contributions that outcomes in articles with text as primary source with figures as secondary source. The measure of reliable on each users can be performed by calculate the difference in the text when one document has been edited, when much of text was left unchanged, it can interpret as peer-validation that the author of this section is honest or is correct[LJ12]. Wign is, in opposite of Wikipedia, based on the post as a translation or a suggestion of the translation between a term in string of literal and a video as primary source. Each post also contains a description as a definition or explain when to use this expression (term or sign). The primary source is a set of term and video and will lead to an unique semantic. The video cannot be edited in same way as a string or text can which lead to the question if the wikiTrust can be implemented in other kind of abstraction content. Furthermore the user traffic in Wign is much smaller than Wikipedia, then number of contribution, either in production, edition or review will be fewer, then the question will be if the security goals still is accomplished with a small community.

3.1 Authorization in WikiTrust

The reputation system in WikiTrust is based on the level that is related to every single content and user. With the levels it follows with a voting protocol that decides if a new change should contained with a review and who should participate the voting for ensuring that the voting can be concluded in a decision.

The registered users will be allocated with a level, called **Q**uality **C**onfidence **V**alue (QCV). Post as a set, will be assigned with an **I**ntegrity **L**evel (IL), then if two that have same word but different video or description, do not have to belong to the same IL. Then if an user edits the word, video or description in a post, the platform firstly has to check whether the user's QCV is on higher or equals to post's IL for allowing to execute the alteration, otherwise trigger the review which will be discussed further in this chapter.

3.2 Composition of the Post

The words, videos and descriptions are the primary parts of the content that will grown the value for the use of platform. When combining one from word, video and description, it will produce one post with its own semantic that works as a translation or suggestion of translation from written language to sign language.

In wikiTrust model, the access control is based on three properties, a set of authors $\mathbb{A}$, a set of posts $\mathbb{P}$ and an finite set of integrity levels $\mathbb{I}$[LJ12]. Integrity level (IL) is assigned to each complete set of posts that was constructed by author a. The author a can set an IL value on this post, as long it satisfy the rule:

$$\texttt{setIL}(a : \mathbb{A}, p : \mathbb{P}, i : \mathbb{I}),\ \textit{where}\ i \in [1..qcv(a)] \tag{3.1}$$

where

$$\texttt{setQCV}(a : \mathbb{A}, i : \mathbb{I}),\ \textit{where}\ i \in \mathbb{I} \tag{3.2}$$

However it is desirable that the voting should not be trigger unnecessary often, but simultaneous the content should be accurate. IL can assign to the data set in the following ways;

When a post has to be generated, it needs three axiom components (word, video and description) which will make good sense that each axiom holds an IL, when each axioms can be generated by different authors that the third one decided to combine them into a post. It can protect the axiom components against

Word	Video	Description
$IL_{\mathbb{P}}(w, v, d)$		
$IL_{\mathbb{W}}(w)$	$IL_{\mathbb{S}}(v, d)$*	
$IL_{\mathbb{W}}(w)$	$IL_{\mathbb{V}}(v)$	$IL_{\mathbb{D}}(d)$

Table 3.1: Definition of the tuple types in content of the posts (* this tuple is used in deployed version of Wign)

malicious attack, but the lowest IL can also be the "best protection" in bad manner when looking at the post as composite of the axioms. When an user wants to do malicious alteration on a post, and one of the component has higher level than the user for allowing to make instance alteration, the user just can pick another with lower level and perform the malicious alteration then the post will lost its value and becomes useless. If the IL should rely on the semantic level then it should based on the post with one from each kind of axiom. When combining the axioms, there is two possible to define the IL. The first one that IL still belonging to the axioms and the post gets a reference IL that is based on

$$IL_{\mathbb{P}}(w, v, d) = \texttt{max}([IL_{\mathbb{W}}(w),\ IL_{\mathbb{V}}(v),\ IL_{\mathbb{D}}(d)]) \tag{3.3}$$

The reason that (3.3) has to find the max IL-value among the axioms, because if using average or minimum, it will violate the "no write-up"-property when lower one is enable to alter something on higher one just because the reference IL still allow the alteration. The property is not vulnerability-free, because when one axiom appears in multiple posts with different IL, it is possible to damage all the posts by just altering the axiom on the post with lowest IL and then the alteration also will happen on all posts. The solution will then be a modification of (3.3) that let all axioms of the post be a pseudonym and axioms do not contain IL. It means that the alteration only will happen on this post, but not affect on other posts that have the same axiom. This case with mutation dependance only is making sense with the set of words, because the word is this one axiom that will appears in multiple posts. When an user generate a new post, the user may mistakenly added the wrong word to this post. When the word will later be changed, it does not mean that all other posts with same word also are wrong. Then the set of posts will not based on reference to the three axiom, instead the IL will determined by its author QCV as shown in (3.4).

$$IL_{\mathbb{P}}(p) = [1..QCV(a)] \tag{3.4}$$

3.3 Community Members as Moderators

Contents of the platform will be implemented in the way that the users are taking the decisions to put the platform in better position without much need to get the administrators involved. There are three type of users that will be mentioned repeatedly in the paper: Guest, Entry and Active. Guest users are those who does not have logins, entry users will be addressed later in the chapter and active users are those who are authorized to utilize the full features that are available on the platform.

3.3.1 Entry users

The idea of how to prevent the small community from being controlled by majority malicious users can be solved in one way by implementing the challenge to new registered users to prove that the user really can be a contributor in the community. The entry users can be distinguished that they are always has zero on their QCV which is default value when registering in the platform.

There will be some restriction on the range of actions that the entry users can perform until their QCV is promoted from zero. The restriction will be the unauthorization access to specific actions or only allowing to perform some action certain number within or without a timespan. The restriction should be a balance then it works as gateway to the full feature, but it also has to be usable for them that they can works as consumers.

	guest	*entry*	*active*	*admin*
Search	0	1	1	1
Create	0	limit	1	1
Read	0	limit	1	1
Alterate	0	0	1	1
Request	0	0	1	1
Vote	0	0	1	1
Like	0	0	1	1
Report	0	0	1	1

Table 3.2: Privileges of each roles (0 = disallow, 1 = allow)

Table 3.2 is much straightforward because Wign only contain one line hierarchy of privilege (figure 3.1). It assumes that all users with the role as 'admin' (or administrators) is always honest and act fairly.

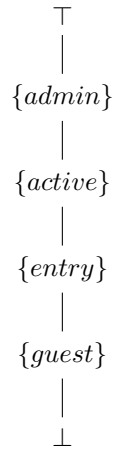


Figure 3.1: Security lattice of roles according to their privileges[SdV01]

The search feature is necessary for enabling to make the platform useable for the entry users, however when showing the results either in posts or request words in detail, some information will be hidden to protect the anonymous or has no relevance. The access to create also will be limited. All the posts created by the entry users, automatically will be sent to the voting. The entry users also should not enable to make infinite many posts that may either burden the voters or affect the result of the voting. All other features are disabled because of not being reliable enough to contribute some contents to improve the value or only is relevance to those are entitled to vote. In this case of Wign, Wign har more interesting to grown the number of content than improve the quality of existing contents, because the collection of contents is very far from complete. In the time of written, Wign has almost 3500 signs for roughly 3000 words, according to The Department of Danish Sign Language at The Danish Language Council there assumes to be least 9.000 signs[oDSL], where an average Danish adult have roughly 50.000 - 70.000 in their vocabulary out of roughly 1 millions words in total, which the current amount of content in Wign can interpret as lightweight content. Therefore the setting of privilege for the role 'entry' looks like in this way to indirectly encourage the users to perform certain actions, including generate new posts. If the platform does not have some preferences or is or almost is saturated. The active users then can judge if the entry user can provide something that the community can trust and is seeking for the qualified contributions.

Entry users can be promoted through the voting when the user succeed has contributed with positive values. The promotion will be trigger when the user succeed to complete the specific amount of contribution which can vary in each

platforms. In this approach with Wign, the entry users are only abled to create post with the word that appears on the list of request word or post with fresh word. It can prevent the entry user to replicate an existing video into the new post, if the view where the voters will se when giving the vote, has to be simple. If the voters is encourage to check all the videos from the same word to determine if the replication has not taken the place beside validate the new post create by the entry user, it may demotivate the voters to make sure that their vote is based on satisfied evaluation. In those lightweight platforms, there are gigant range of possible post that not yet is created. Then when entry users may misuse the lightweight to generate a lot of posts that its sign is in anyones vocabulary or is available on the research-based dictionary[1]. Unfortunately it is not easy to limit the possible to generate a post with fresh word in the case of that the request words only contains few words or only very special, ie. academically words that most people never used the word. The misuse of "easy points" will however only happens in the beginning, because later the contents will be filling up (if they are approved by the active users) with posts of basic word/sign, and the misuse will becomes harder to take advantage of. The request words always is more desirable to use for creating the post, then there is possible to make a time-based limitation on how many posts the entry users can generate that is not taken from the list of request words, and less or no limitation when creating directly from the list. So it will not be impossible to promote from the entry user, when there are other way to prove their reliability.

The active users also can be demoted to "entry user", then they will redo the work. But if an entry user is demoted, the user account should be excluded from the platform. The exclusion can be decided by the voters for the demotion or the administrators have to be involved in the final decision to ensure that the framing has not take the place.

3.3.2 Voting Security

The security in any votings is crucial if the users are willing to participate the further votings, It may affect them positively, if they experience that their opinion and decision is well-protected both during and after each votings. Fujioka, Okamoto and Ohta[FOO93] addressed the definition of secret and secure voting should contains the following properties:

Completeness - every ballots redeemed by its owner and voted correctly, should be counted.

[1] `http://www.tegnsprog.dk`

Soundness - no dishonest voters can make the voting becomes invalid.

Privacy - all the votes must be secret

Unreusability - None can make multiple votes

Eligibility - only entitled voters can vote

Fairness - none can get influence on result of the voting

Verifiability - the result cannot be falsified

Privacy property will be satisfied if it is impossible to figure out what the user has voted at a specific voting. However the property only holds for those all those users, but not for the people who have the access to raw data in the database.

For ensuring that the property of fairness holds, it is necessary to ensure that none users can sway other voters by generate fake likes on each post to obtain the misleading size of users that acknowledge the post. Therefore the entry users are unable to synthesize the existing contents, ie. by liking, otherwise one person can create many users for this purpose like we experience in commercial profile on social media that bought likes or followers[CPea15].

The platform provides two kinds of decision-makings, depending on the number of active users which is those users who are not banned from the platform, neither has QCV on 0. If there is not enough participants, the voting will be decided by one of the administrators.

3.3.3 Ballot Capacity

A voting needs voters for allowing to attain a decision. However the users are visiting the platform for collect the information they may need in their daily. According to the paper by Lindberg and Jensen[LJ12], the users should not be assigned to every single votings otherwise they may fell less likely to participate the votings. At the same time results from any votings should be representative by opinion of the community as good as possible, then theoretically after infinite number of votings, the number of allocated ballots to each the members must be equal. The idea is desirable, but the members cannot be expected to always react on the votings or visit the platform on regularly basis. One user may lost its access to its mail that recover the password will be impossible, then the user may need to create a new user and leave the old user. There may also be some users who use the platform barely or some users literally ignore the request to

them to participate the votings. "Ballot" acts like authorization control that ensures the users with the ballot is legitime to vote this voting, those without ballot should not be allowed to vote expect the administrators which will be addressed later in this section.

If we take a look how people act in the real world, when their democratic participation was requested by inviting them to a general assembly an association or similar. Somebody may decide to appear up at the general assembly, other may be absent which enhance the power to them who turned up. Some associations may have regulation that specify that the absent members had to be counted in the total number of voters. The approach also can lead to be stuck in a situation that the majority want to approve a decision, but there are not participants enough to exceeded the threshold of majority. Then this practice ought to be avoided as a solution for the implementing in the platform. The users on internet should instead not to encourage to vote at every single, otherwise the votings barely will passed because of missing on the response by the passive users, also those users who were allocated to many votings, may feel less likely to vote. Therefore the ballot distribution should be sound in both ways that it both include enough voters to obtain the reliable result and not include too many voters that may lead to many blank votes which make the decision less accurate and possible let the improper or less correct contents stay visible on the platform longer time. Equation 3.5 satisfy the properties that in average case no users will participate more than half of the votings and if there is too few active users ($x < \Lambda_{min}$) the distribution will not takes the place, and instead the administrator will do the review.

$$f(x) = \begin{cases} \lfloor log(x)^2 \rfloor & \text{if } x \geq \Lambda_{min} \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

Unfortunately equation (3.5) is optimal in case with small community where the function is following the number of users, but if it grows towards infinite size, the number of ballots will becomes constant. Then the function should grown faster than (3.5), for being usable in the case with huge community, but also keeping below the half of total active users. Equation 3.6 makes the number of voters more appropriate in both small and large community, however in the beginning of function the number of voters exceeded the half of total users. That is why the equation is added with the condition that number of voters must be less or equals to the half of total users, otherwise the review will be sent to the administrators. The natural numbers n can be adjust to determine the part of total users that should allocated the ballot or how much the administrators can follows the reviews generated by the community whose users with no authorization to alter the post. According to capacity of different kinds of contributor, it is only 10% of total users who are make synthesize to the existing contents which can be alteration, make some like on a post or vote

on a voting[FG10, p.15ff]. n in Equation 3.6 is a parameter that control the share of voters by the number of active users.

$$g(x) = \begin{cases} \lfloor \frac{x}{2+n} + log(x)^2 \rfloor, \ n \in \mathbb{N} & \text{if } \lfloor \frac{x}{2+n} + log(x)^2 \rfloor \leq \lfloor \frac{x}{2} \rfloor \\ 0 & \text{otherwise} \end{cases} \tag{3.6}$$

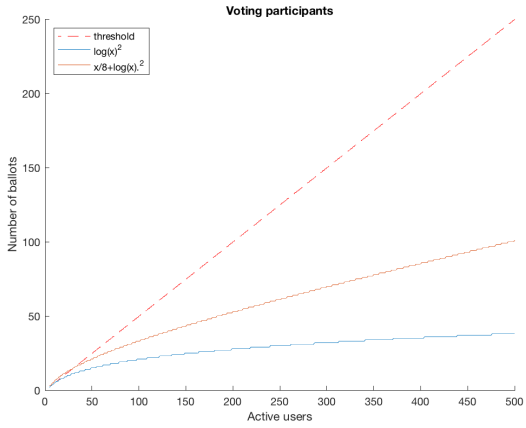


Figure 3.2: Determine the number of ballots when a voting is triggered ($n = 8$)

It is our interesting that the grown of participants will not getting closer to $log(x)^2$ when x goes against infinite. Then if the equation (3.6) has $n = 6$, then

$$\lim_{x \to \infty} \frac{\frac{x}{2}}{\frac{x}{8} + log(x)^2} = 4 \tag{3.7}$$

that proves that the equation (3.6) with $n = 8$ will increase with one participant when x has increased with 4 in infinite size. Then the equation can be used in any platform with different capacity of users, since n can be adjusted to determine the share of users that shall considered to be voter where the function will goes closer to $log(x)^2$ when n is higher than 6.

3.3.4 Ballot Distribution

The allocation of ballot can be done by entitled or behavior distribution that based on whether the users' history of activity has to be determine whether the user is entitled to make a vote. The intelligence distribution can improve the level of reliable in each posts and users when the decision from each voting are

based on the majority. The lack of majority decision, when the voting expired without majority decision, will lead to slow process to the next alteration of one existing post.

Entitled distribution is based on that the users are not excluded from the access to the platform, neither including those entry users (where their level is on 0). The voting also need to exclude specific users in the case of conflict of interesting (when a voter has to vote that is about itself or initialized the voting by the same user). With the legitimate voters there only extract that will be allocated with a ballot for the voting.

Behavioral distribution will based on the consideration if this user can be expect to make an authentic contribution beside being entitled to the voting or just ensure that the user eventually will vote. The exclusion mechanism must be reversible for avoiding this person to drop the user and create one new in the case the user is stuck in being permanently non-entitled to vote. The mechanism can either be a punishment that last for a period or temporarily exclusion until the user performs the reverse action. Punishment will be given when the user did something improper, either repeatedly or once that cannot be reversed, i.e. repeating vote true in any votings. Temporarily exclusion is used when the user has not logged in for a while or does not vote the votings that is allocated to this user that can be reserved when the user logins or votes again.

Behavior Detections

Behavior is an optional supplement to the ballot distribution to increase the reliable a bit more by ensuring that the contributed users are so honesty as possible. The main contribution can be a create, edit or vote which has significantly mean on the value of the platform. The mini contribution (like, request etc.) will not be mentioned further in the analysis of behavior detection. The behavior detection only is about to decide if this user is suitable to join a voting according to its recently pattern of behavior. The detection also can use to demote, restrict some actions for this user, but it will be an explicit information flow for this user, if some features suddenly becomes unavailable, then the user may motivated to create a new user to obtain the feature again. Then this approach of restriction will not be discussed further.

There may be some users who have certain pattern of the behavior that either framing somebody by personally or quality reason. It is hard to distinguish between whether it is a cyber bully or the victim never contributed with a content that the user may think that it is constructed or expressed wrongly. Minority may notice something that majority did not, then they should not

register as rebel just because of they often vote different than outcome of the decision. The behavior detection should make attention on those users who always react identical on the votings.

The detection can be based on the statistic calculation of the deviation. The distribution of final results will be used as reference to see if there is some incoherent in the behavior by the users. The user can get significant deviation if the user i.e. has high percentage of decline-votes or the user always vote something that is opposite to the final results.

If the detection has true positive, the platform can call the punishment or temporarily exclusion, depending on if the improper actions is irreversible. Temporarily exclusion is suitable if the actions can be withdrawn or perform proper actions afterwards is possible. Punishment will run within a timespan, but it may lead to undesirable states that the the users may see the platform as pretty useless or avoid the punishment by creating a new account, if the user has not much to lose in this account. Therefore punishment may not a best solution to prevent the malicious actions. The temporarily exclusion can be used when someones pattern of the behavior has clear deviation of the average. The exclusion can be done by reduce frequency of ballot distribution to this user, as long the deviation keeping too high.

As mentioned that the minority may get right that the majority may not noticed, then there have to be careful when implementing the behavior detection. Therefore the detection will not be included in the further working process, since Wign has no data about how the users will act in the voting system.

3.3.5 Decision-making

All outcomes from the votings must be majority to obtain the representability as additional property to the voting security. The degree of majority can vary depending on how critical the voting is about. As basic the vote can be **approve**, **decline** or **abstain**. Abstain can be caused that the user does not know what to vote. Those who did not participate the voting, will count as decline. The abstain votes can count either as decline or leave the decision to other voters which give other voters more influence on the decision. If the abstain-voters gives the more power to other voters, the representability does not hold, because final result can be flipped if it depends on what to interpret the abstain-voters. In the reality the abstain can be used when the voter experience the conflict of interesting. In the platform the voting only will get a partial of the total users and those users with conflict of interesting will not allow to vote. The conflict of interesting can be that the user did triggered the voting itself or the voting

is about this user.

Each user is assigned to a QCV that represent the reliable level this user is on. The platform has the interesting to contain highest reliable value in all contents which assume that the users with higher QCV will generate qualified contents than those who is on lower QCV. Then the QCV should have influence on the voting [LJ12]. When somebody has more influence in the voting, it has to ensure that when promoting somebody, the promotion is more demanding when promote to higher QCV which mean that it requires more supporters. So the threshold τ of decision-making has to be dynamic.

$$\tau \in [0.5...0.95] \tag{3.8}$$

τ will starts low when the voting is not critical, i.e. letting the entry users to promote, but the threshold will increase when promote to higher level to ensure that there are wide trust by other users. The reason that threshold stops at 95 % lies behind that in practice it is near impossible to obtain 100 % approver, when somebody may unable to vote or just one who may dislike this user. So 95 % can avoid the state of stall.

3.3.6 Duration of the Voting

In general a voting usual has a duration so none will run infinitely, if the threshold of majority never exceeded, then the votings also need the terminal property so all votings will contain the final decision in any cases, when there is too many inactive users who are allocated to this voting. As in general voting, a blank vote counts as against, but it also can interpret the blank vote as absent and will not be count in the final result which gives more power to those who voted, but the method only is used when voting for simple majority when voting to pick one of multiple motions or candidates. Votings in the platform will only be about to approve or reject the motion, then the blank votes will count as reject. Then when a voting terminates in any way, it always has a final decision.

The duration will depend on rate of activity performed by the users and average of time to response in votings. The reason that the rate of activity should taken into account because if the time to response is very long than the average time before one content is created or altered, the users may lost their patient and find a trick way to moving the platform in better place around the votings. The duration should not be too short that the users will miss their voting that

expired before their next regularly login, neither too long that may lead to undesirable by the users.

The duration can be set with a static number in days, but then it may match a specific period, i.e. workweek, but mismatch during the public vacation. Therefore a dynamic duration will ensure that it does not run too long during the workweek, but extend the duration when the vacation season has started. The parameter can be

$$duration = \begin{cases} average(days\_since\_login[user]) * 1.5 & duration > n \\ n & \text{otherwise} \end{cases} \quad n \in \mathbb{N} \tag{3.9}$$

Equation 3.9 will avoid the special case when the average become zero which all the votings will theoretically expired immediately when checking. n in (3.9) should be a fixed number that is based on empirical search of rate on the activity on the platform or time to response in the votings. The duration is unfortunate is not fair to those whose rate of login is lying above the average, then they will miss the votings. Therefore the algorithm has to balance between the representative and effective, so no votings were hurried or idled. The multiplication 1.5 is based on the assume that the users have time to participate the voting when they are logged in, then in average the users only can give their vote at first or probably second login when the voting is triggered.

3.3.7 Administrator as decision-making

In the case of platform with small community, it is unavoidable to not include the users too often in the votings. Simultaneously none users should participate more than half of the votings if it has to avoid that the user or a small group of users is controlling the platform. If using the formula (3.5) or (3.6) for ballot capacity, there still can happen that a group of users may control most of the voting, since the voting indirectly can be recall if the group see that their votes was not a part of the final decision, the group just can trigger the voting again and hope that the distribution of ballot will allocate to multiple of members in the group. Therefore the administrators need sometimes to overtake decision-making, when they suspect that a voting is not executed fairly, and if need, also allocate a temporarily exclusion or demote to the user(s). Hopefully it is not necessary, because the community better can detect the improper behavior and trigger the voting of demotion than the team of administrator can do, if the behavior detection did not returned with a positive response.

If the platform has high risk of corruption because of low number of users, the administrators will be an active part of the votings. When a voting is

triggered, it will automatically allocated to the administration that only need one to make the decision. The process can be disabled when the administrators cannot follow the incoming stream of votings or requests for the attention of the administrator or when the number of active users exceeded the threshold, the process can switch to the democratic voting policy.

The administrators have the primary response to react on those votings which are assigned to them.

3.3.8 Score System

The score system can augment the accuracy of the reputation system among the level-system (QCV and IL). The score system does not depend on the level-system, but the level-system can impact how easily the user can collect the score because of the authorization for allowing to do the actions around the voting.

The mean of being on a level of QCV may interpret different by the users. So there may appear unfair experience when somebody is on too low level or other user should be on lower level, when comparing with other users on same or higher QCV. Score system will makes it easier for other to see how much value this user has given to the community and how much can the community trust on this user.

The score is partial irreversible that the database will not be filled with many redundant data about when the scores were collected or deducted. Then each users will holds a score that is increasing or decreasing through the time the user was active at the platform. The score only can be given to other which mean that the user can give the score indirectly to other by acknowledge others contribution as input reputation statement, either by individual level (like) or collective level (voting). The reputation process will be the score from the votings can be counted either by number of approve voters with or without weight according to QCV of the voters or only based on the final result from the voting independently the voters[FG10, p.26ff].

WikiTrust uses the algorithm to determine how much one content was altered for allowing to calculate the score for the part that was keep unchanged as the editor is recognizing the author's work[LJ12]. This algorithm is not directly transferable to other platforms that do not have long text as primary source. In the case of Wign, it has three parts; word, video and description that will obtain the primary source when synthesize them into one post, where it will become usability as dictionary. All the alterations do not go to the voting, will miss the opportunity for the author to collect the score, then it will benefit

if the algorithm can be implemented, then no matter which QCV the user is on, the user can collect the score either through the voting or just by the alteration. Again the users should not enable to collect the scores themselves, then score obtained from the alteration will happen when another user makes a small correction, ie. expand the description with additional information which can be indirectly interpret as that the editor recognize the correctness in the author's contribution. The alteration can be append, overwrite or delete on word, video or description as shown in table 3.3, when looking on what to interpret when performing those alteration(s) on the post.

	Word	Video	Description
`Append`	+	$+^*$	+
`Overwrite`	+	−	(+)
`Delete`	−	−	−

Table 3.3: Measure of value by input reputation performed by editor on existing post

`Append` will just add more information to the existing post which can understand as recognize the author's work, however it is not possible to extend a video, then the star ('*' in table 3.3) means the video will be added to another post with same word and probably same description that can interpret as the existing post is recognized but there is another variation of the expression that the editor wants to add. `Overwrite` only mean that the text is erased with another text, not erased with nothing. Overwrite can be used to correct the spelling error or spell another word that was expressed in the video, then it is a recognize that the video is correct. However it is not same in video, because there is not possible to modify the video so the entire video has to be erased with new one recorded by the editor. The description may described definition or usable of the expression wrongly then the entire description may be rewritten, only if the video keeping stayed then there can be a partial recognition. If it just a part of the description that is overwritten, then it is a recognition on the unchanged part. `Delete` is an obvious signalment that the contribution has no values to keep on the platform when delete the entire axiom. It can be that the word or expression in the video do not exist. When deleting the word or video, the post also will be deleted. The description also may understand as no values and should not be stayed, but it is expected that overwrite will be used in this case than delete, then it should pay attention to those who delete the description may act on malicious purpose.

`Append`

Word : Extends with additional literal(s)

Video : Adding new video in another post with same word

Description : Extend with additional text

`Overwrite`

Word : Change the spell, remove some literals

Video : Erase the video with new one (if the expression is not precise)

Description Proof reading, rewrite or remove some clauses

`Delete`

Word : Remove entire word

Video : Remove the video

Description : Remove entire text

The implementing of score system with alteration of the video is not challenge, but it will be challenge to implement the system with word and description, because it has to work on the textual level for allowing to distinguish between append or overwrite, if they will lead to different degree of allocated score.

3.3.9 Transparency

Knowledge is power. When a user get more access to different kind of information, the users may expected to act differently than if the user is unawareness about the context. The platform has the interesting to get many traffics and contributions by the users as possible. To motivate the users to participate the platform, the user needs to feel success and see that they are influencing how thing should move towards. Then the platform has to consider how much information should be available to the users to attract them to use the platform and simultaneously confine the amount of information to avoid the conflict there may occur among the users.

All the users who participate a voting, will hope that their vote will form the final result. If they are informed explicitly about the outcome that the user is disagree, then the user may make the alteration to the state that was outvoted that either go to or around the voting. With too few information about the output of all votings can avoid the user to trigger the voting again just because this user is disagree with the result. The case still is unavoidable, because the is implicit information flow as soon the user joins the voting, the user is enable to figure out about the output of the final result no matter if the user never was informed. Therefore to keep the spiral of alterations low, the outcomes will not sent as notification to anyone, so the users may forget about the voting.

Only the main entrants will be informed when the voting is triggered and the outcome of the votings, including the user who has been demoted by the voting.

Then the explicit information flow has to be need-to-have, it they shall be available for the users so they can perform informed decision. Simultaneously the flow should not contain much information that may indirectly encourage the users to act improperly to reach the better position or make sure that their influence is conclusive. It can be that another user may stalking and forcing the user to vote in certain way which violate its integrity. Unfortunately those informations that should not be available explicitly, may be available in the implicit information flow as discussed about the voting to ensure that the voters can vote by form the basic of informed information, because in the case that the platform used composite data that cannot add a reference or prove its validation beside presenting its definition and when to use clearly.

3.4 Requirements Specification

The platform musts enable the users to contribute a content by generate a bound of data set that contains a word, a video and a description that they either are generated by the user or take from another existing content as correction or additional to the existing content. The users also can put their indication on each content whether the user is agree about that. The development of information shall based on majority decisions by the users.

3.4.1 Business Logics

User Account

Anyone should be allow to create an user on the platform and still feel that their integrity of contributions is protected. Then the user musts be authenticated before enabling to contribute or read the contents. There must be clear about who contributed in each post that all users can read them.

Guests who have not logged in, shall not allow to read any contents in the platform, neither contributes in any ways. They only can register themselves or login.

A fresh created user will starts with $QCV = 0$ which only is not allowed to generate a request for a translation of this word. The user has limit number of

post the user can contribute per day, but less limitation when contribute with new post from the list of request words. Each 10th post the user generated, the voting of promotion will be triggered automatically. If the promotion is declined, the user has to repeat the contributions. Voting for demotion of entry user can be triggered by another user. If the demotion is succeed, the admin will perform final check to see if it is reasonable to block the user.

The ordinary users with its QCV on least level 1 have access to generate post or request word. The users also are enable to edit any posts, but those posts that is on higher IL than user's QCV will not be overwritten immediately. The alteration will sent to the voting. The users also can visit other users' profile page and also trigger a voting of promotion or demotion. It also is possible to ask to promote themselves. The users also is entitled to participate any votings that they are not involved.

The administrators (admins) have same privilege as the ordinary users with additional features that the admins can make the final decision on any votings that the result from the voters will not count. The admins also have the responsibility to approve the block of an user as result by the voting.

All users must have their own profile page that they read about whether their contributions are visible or altered by other. In the page the users also can see if they are requested to participate one or more votings. If the user has trigger a voting, the page will show if the voting is pending or terminated with the final result. The page also allow the user to edit their personal information or decide to delete the account.

An user is allow to delete its account which all its personal information must be clear to satisfy the GDPR[Com] expect the contents. All the videos generated by the user must not be visible to other users, otherwise all the contents keeping visible but without showing explicitly name of the user as author. If a post has one or more previous video, the current video shall erased with the older one, otherwise the entire post has to be hidden.

Value-based Contributions

The post must contain one word, one video and one description for allowing to present on the platform as a suggestion of translation. The posts with same word will be present on the page that the user can read. The post must allow the users to report, if its content does not follow the terms of condition. Users can like the specific post then each post contain a collection of likes which will be presented as count visible to the users. The count will used to order the

posts in decrease order that the post with most likes will on the top. All post has its IL that is lower or equals to QCV of the author. The author can decide which IL the post shall get. When a post is altered, there always must save all the old posts. The users can look all the old posts that were altered towards this current post and enable to redo an older post.

Request words shall enable the users to make the request and each request has their collection of users that have requested this word.

Democratic Decision-makings

A voting must ensure that all entitled users will take into account when allocating the ballots. The decision from all votings must be representative by the majority either when the majority is reached or the voting is expired. The voting shall require a certain percentage of the approver for allowing to pass the voting, where the percentage musts be higher when voting about some high QCV or IL. One user only can make one vote per voting. Decision from the expired voting cannot be rejudged, when users voted the expired votings. The duration of the voting must be reasonable that users with average rate of login have opportunity to vote.

None users shall enable to exploit who did vote what in the votings for allowing to protect the confidentiality and integrity. Outcomes shall only announce to those who triggered the voting or the voting will affect this user's QCV.

If number of user with entitle to vote is lower than the lower threshold then the admins will decide the outcome. The lower threshold is adjustable so the admins can decide if they can follow the stream of votings that will not be voted by the users. If the users are gonna do the voting, one voting musts never allocate the ballots to more than half of the total users that are entitled to vote. The number of voters should not be linear, but reduce slowly the share of total users when the number become greatly. A voting shall, if possible, involves users from three different QCV or more if there were allocated to fewer voters than expected. The content the voting is about will use its IL or QCV as point of reference for whom QCV it has to get the ballot.

When an user is removing its account, all the awaiting votings that the user has not voted, will be removed, so it will not affect when counting the result of the votings.

3.4.2 Reputation System

The reputation system will help the user to perform their criticism thinking on the basic of informed data when reading the platform for their further use.

All the posts must be clear about who created or made the latest alteration beside the IL. The number of likes also shall be shown in every posts.

The users shall experience the transparency in their contributions or that affect them.

Chapter 4

Design

Wign is a platform that is deployed on the server therefore this chapter will expose the approach how to extend the existing system when remodeling the crowdsource to include the reputation system.

4.1 Redesign the Existing System

Firstly we has to edit the existing data that have mismatch with the new design. The structure of the content has to redefine to follow the post-structure. The approach when create a request word also need to modify, so it based on the authenticated users instead of IP-addresses.

Word	**Sign**			**Post**		
Word	Video	Desc.	$\longrightarrow$	Word	Video	Desc.

Table 4.1: Redesign of the content-based structure

The sign-model in table 4.1 contains the data of video and description and is pointing on the word-model that contain the string literal of word. The new design supports the editor feature, then all the axioms shall be stored. So the

sign-model shall be replaced that video and description get their own model that the abstract post-model is pointing on word, video and description. The change will not affect the search engine, because the word-model is unchanged.

In the previous structure the likes is named as 'votes', then it needs to rename to 'likes' for distinct difference between the votings and likes. In the old setup the like contains IP-address and sign id as relation to sign-model. Then they have to modified that like bound the relation between the video-model (notice not the post) and the user-model.

In the end the request word also has the similar structure as like that each row contains the ip-address who triggered the request to this word. The structure has to change to a relation between word-model and user-model.

4.2 Database Structure

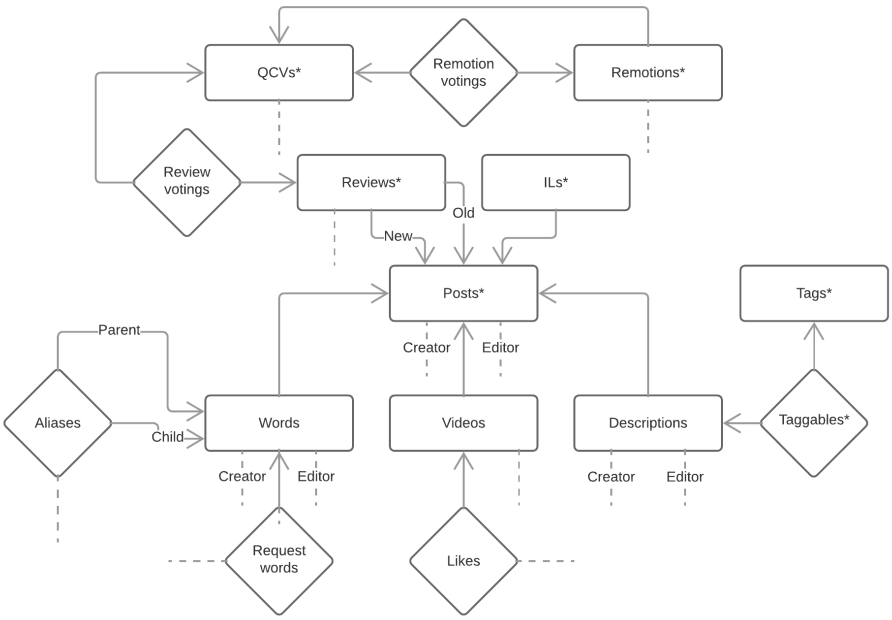


Figure 4.1: Database structure of the platform without user table

All the blocks above of 'Posts' block in figure 4.1 are introduced in the extension. All those dash lines are pointing to 'Users' block that is shown in figure 4.2. All

the rectangle squares are business models, and those rhombuses are pivot that works as a relation between two business models. Those where the name ends with the star (*) means that the block only can perform soft delete which they will get one more column with timestamp of this time the row was deleted, but not remove from the database, but just make the row invisible. Review/Remotion votings works as a ballot and a voting paper in one. The specification of each columns in the blocks can be found in the appendix D.

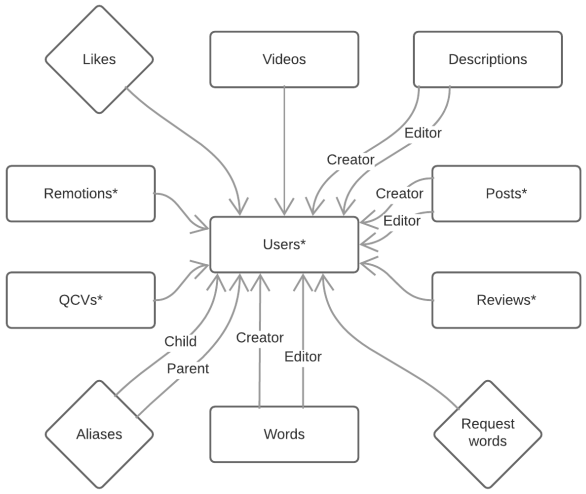


Figure 4.2: User relation to other tables in the database structure

Aliases and Tags/Taggables will not be addresses further in the paper, because they have not any impact on the reputation system.

4.3 Setup of Reputation Framework

The reputation system will apply on the users and posts. According to the discussion about levels of reliable the post will get IL and user will get QCV. IL and QCV is an integer that can be between 0 up to 5. The maximum level (5) has to be constant because there follows a couple of arrays whose size must match the number of available levels. The arrays will be used in the votings. The QCV levels shall also work as authorization mechanism for the users.

The level-system will supplemented with score system that is based on the feedback the user receive from other users. The score is an positive integer and is irreversible. Each user holds their own score.

4.4 User Features

The extension of Wign will introduce the user-model that has the main role to develop the platform with new contents. Any contributions must be trackable to the user who either created or edited the content.

The authentication and authorization mechanism must be implemented for ensuring that only the registered users can get the access to the contents in the platform. Then when an user that is not logged in, shall only allow to read the overall information about the platform that does not contain any personal information, including photo or video and no other as shown in table 3.2.

Entry users (where their QCV is 0) only is allowed to search the translation contents, but accessing to the information about other users and participant to the votings still must be unavailable. The entry users must generate 10 posts in total, if they want to trigger the promotion of themselves into the voting. The number of maximum contributions will be

- 1 post per day with free chosen word
- 3 posts per day if the word appears as request word.
- 10 posts between each votings

When the limitation is exceeded, the user will unable to upload a new post.

The users who is entitled to participate the voting, shall ensure that they will get a notification when one or more votings are pending on the user to vote.

Each user has their own user page that they can see which information the platform knows about them. The personal informations shall allowed to be changed expect the QCV. The users can trigger the promotion of its QCV. All the posts that the user needs to sent to the voting, will be shown in the page about how many votings are on pending state. The page also will show a statistic that summary their contributions:

- Number of contribution in word, video and description
- Number of posts created and edited by this user and how many of them are currently visible.
- Number of voting allocated to this user and how many of them did the user voted and not voted.

All users also have the guest version of themselves that will be shown to other users. The guest page only show name, QCV and which posts the user has contributed. This page allows another users to trigger Promotion or demotion of this user.

4.5 Administrator Privileges

Those who has the admin role, will get more access to additional features in user management and reputation system for allowing to keep the malicious actions low. There is only one kind of admin-role, then for avoiding that some admin may add or remove other admin freely the feature of adding or removing the role. Only those who has the access to the back-end of the platform, will be enable to add or remove the role at the users. When an user gets the admin-role, its QCV will remain unchanged but the user gets a pseudo QCV on 5. It will allow the admin to alter a post that may have improper content.

The administrators with its role will not get any ballots to the votings, but instead if there are some votings that are not allocated single ballot, then it is a indirect allocation to the administrators. There only requires one administrator to vote for allowing to make the final decision. It musts be clear about who made this decision.

If an entry user is worth to ban from the platform, it will be administrator's liability to block the user from logging in the platform to prevent more voting that obviously will be declined.

4.6 Post Feature

When an user wants to generate a new post, either by submit a new or by edit an exiting post. The new post must contain a word, video and description with a set of likes and IL as a part of reputation system that already exist in the platform or is generated by the user (4.1).

$$P(w, v, d, \{l\}, IL) \tag{4.1}$$

The word shall works as axiom that contains one or more words that is separated with whitespaces. Semantic in this string literal form cannot keep same when dividing into multiple subgroups. For example the string "Tree is green" is

composite of "tree", "is" and "green" that can be divided without lost some detail in the semantic. Other example is "come on" that reaches a different semantic when combing them. Therefore the word-model musts allows multiple words into one string literal, simultaneous the word shall be restricted to express as a whole sentence.

The video shall contain only one expression or one "sign-literal" that is a translation of this proper word. The duration of the video shall be limited for ensuring that the users only signing that is equals to the word and not longer, ie. including the description in the video.

Description has the text as type that contain a explanation or definition of the expression or translation. The text also can contain hashtags that will show url-highlights that the user can press to redirect to the list with videos whose description contain the same hashtag as pressed recently. When a new description is generated, the text will be checked if the symbol "#" occurs as prefix in any word that shall generate the relation to the tag and erase the normal literal to url-reference.

Likes is belonging to video that is counting the number of users that liked this post. The likes will remain when somebody alters the word or description, but if the video is altered the count of likes will be zero again.

Integrity Level (IL) musts be a round number between 0 and a fixed maximum number which in this case the number is set to 5 for matching with the voting system that will be addressed later in this chapter. The user can set the post with a number of IL that is between 1 and QCV of this user. None posts can have IL on 0, otherwise the entry users will enable to altered. When a proper user alters a post, the user can set IL on new level that still is lower or equals to this user's QCV. If IL is the only element in the post that is altered, it only will be accepted if the user's QCV is higher or equals to IL. The voting will not be triggered in this case.

The users with authorized to edit, will allow to edit any posts. To keep all the previous posts, there has to create a new instance of the axiom that is altered, if one of axioms (word, video or description) is altered.

$$P_i(w_i, v_i, d_i, \{l\}, IL_i) \Rightarrow \begin{cases} P_{i+1}(w_{i+1}, v_i, d_i, \{l\}, IL_i) & if\ w_i \neq w_{i+1} \\ P_{i+1}(w_i, v_{i+1}, d_i, \emptyset, IL_i) & if\ v_i \neq v_{i+1} \\ P_{i+1}(w_i, v_i, d_{i+1}, \{l\}, IL_i) & if\ d_i \neq d_{i+1} \\ P_i(w_i, v_i, d_i, \{l\}, IL_{i+1}) & if\ IL_i \neq IL_{i+1} \\ P_i(w_i, v_i, d_i, \{l\}, IL_i) & otherwise \end{cases} \tag{4.2}$$

if multiple cases in (4.2) are true then they have to merge into one post by

perform the operation:

$$(P_{i+1} \uplus P_{i+2}) \setminus P_i \tag{4.3}$$

if the definition of $\uplus$ is an extension of $\cup$ that includes all the duplicates:

$$\{A, B\} \uplus \{A, C\} = \{A, A, B, C\} \tag{4.4}$$

then all the newly altered will taken into the new instanced post with those that were not altered.

All the posts shall enable to be identified by all users who created the post in original format and who did this latest modification that the users can click to know more about this user in the case somebody may want to promote or demote this user.

4.7 Voting Mechanism

To initialize a voting, it musts contain a trigger mechanism, set of voters, content and aftermath(s). All votings must conclude with an approval or rejection. The votings shall terminate when the deadline is expired, not when the decision is made.

The review is doing the voting of posts and remotion is the voting about the user. They have almost identical data structure expect that the review ($R_{\mathbb{P}}$) holds the information about previous post that was altered and the upcoming post that has to be approve to remember which one that will be used after the voting. Remotion ($R_{\mathbb{U}}$) holds the target user's current QCV to remember which QCV the user was on when voting of promotion or demotion was triggered. The remotion also know whether the voting is about to promote or demote this user. Review and remotion shall act in same way during the voting process expect the aftermath.

$$R_{\mathbb{P}}(P_i, P_{i+1}, U, out) \tag{4.5}$$

is a state of the review. U is the user who triggered the voting because the user's QCV is lower than post's IL and *out* is a boolean value that tell if the result was executed to avoiding to repeat the aftermath then *out* is set to false in beginning of the voting. There are two cases in aftermath that only one will be executed depending on the final result. Process of each cases is shown in table 4.2.

	Approved	**Declined**
$P_i \rightarrow$	`delete`	`restore`
$P_{i+1} \rightarrow$	`restore`	`delete`
$out \rightarrow$	*true*	*true*

Table 4.2: Aftermath of review

$$R_{\mathbb{U}}(U_i, QCV, promote, U_j, out) \tag{4.6}$$

explains the state of remotion that QCV is the current QCV of the target user U_i, *promote* tells whether the voting is about to promote (*true*) or demote ($false$) the target user. The two last ones are same as review state, where U_j is the user who triggered the voting and may be the same user as U_i. The aftermath has three cases that beside the final result also depends on whether the target user has to become demote or promote. Process of each cases is shown in table 4.3.

	Approved		**Declined**
promote	true	false	
$U_i \rightarrow$	$QCV(j+1)$	$QCV(j-1)$	$QCV(j)$
$out \rightarrow$	*true*	*true*	*true*

Table 4.3: Aftermath of remotion

The voting has the majority threshold that vary depending on QCV/IL or if the voting is about review or remotion. Table 4.4 specifies which threshold the voting shall use.

IL	1	2	3	4	5
QCV ($\rightarrow$ / $\leftarrow$)	0-**1**	1-**2**	2-**3**	3-**4**	4-**5**
review	50%	60%	70%	80%	90%
promote($\rightarrow$)	50%	60%	70%	80%	90%
demote($\leftarrow$)	50%	50%	50%	50%	50%

Table 4.4: Value of thresholds in the votings

When a voting has exceed the majority threshold, it executes the corresponding aftermath, the voters still shall allow to vote if they did not vote yet before the date of expiration. If the voting passes the date of expiration, the aftermath will execute the 'declined'-case.

If an user is going to demote to 0 in its QCV that was decided by the voting, banned by the admins or deleted its account, all the votings that are allocated to this user and not yet redeemed the ballots, will be cancelled.

Since the votings will involve users from multiple, primary three, QCV groups that also has to ensure that none group can decide the voting themselves without other groups, so each vote has their weight depending on their QCV and higher QCV will get fewer participations than lower QCV. Each voter will get their impact based on their QCV (table 7.1). The IL or QCV (bold marks in table 4.4) is the origin to determine which QCV groups will be allocated the ballots (4.7).

$$\begin{cases} QCV(4),\ QCV(5) & if\ QCV \geq 4 \\ QCV(i),\ QCV(i+1),\ QCV(i+2) & otherwise \end{cases} \tag{4.7}$$

The ratio of number ballots to each groups is (50%/30%/20%) when three groups is allocated, (60%/40%) when only the two uppermost groups is allocated. The number of participants ought to be $QCV(i) > QCV(i+1) > QCV(i+2)$. If ballot distribution allocated to fewer voters than expected (equation to determine the capacity of ballots (3.6)), then the distribution will open up for all groups (expect QCV(0) and the leftover ballots will be distributed to them. If this case occurs, then preventing against only one group controls the voting will not hold if this group gets more ballots from the leftovers. The weight vote shall makes this procedure of ballot distribution rational that each QCV holds the weight as positive integer (table 7.1). If somebody voter has changed its QCV after allocated the ballot, the weight will based on the QCV that was allocated, not the most current QCV.

Voter's QCV (i)	1	2	3	4	5
Weight ($\mathcal{W}_i$)	1	2	3	5	8

Table 4.5: Vote weight of each voter depending on their Integrity Level (IL)

All users must enable to vote either approve or decline and the page of this voting must be consistency and clear about what the voting is about for allowing to make informed decision by either presenting the difference between the two posts or the information of this target user and its recently activities and whether the voting is about demotion or promotion.

The determination of voting's final result $\mathcal{D}(r_i)$ will be done by ([LJ12]):

$$\mathcal{D}(r_i) = \begin{cases} 1 & if \sum\limits_{j \in \Lambda_{R_i}} \delta_j(d) * \mathcal{W}_j \geq \tau * \sum\limits_{i \in L} |\Lambda_{R_i}| * \mathcal{W}_i \\ 0 & otherwise \end{cases} \tag{4.8}$$

where $\delta_j(d)$ is each voter's vote that is extended with weight based on voter's

QCV $\mathcal{W}_i$. The threshold τ determine the share of total voters in each levels that are allocated a ballot to this voting Λ_{R_i} within the range of levels L that musts exceed for reaching the threshold of approval decision.

4.8 Secure Policy

The secure policy should protect the registered users and the content in the database against the malicious actions. There will consider the three universal aspects of the security; confidentiality, integrity and availability and with some additional ones.

The users have the interesting that their contributions go to the purpose they are mean to be, then the contents must hold the integrity then nobody should be allowed to alter any content under another's name. If an user wants to alter, the user has to make a copy of the existing content and add the user's own identity as author (creator will never be altered) and make a soft delete on the old content if the change has passed so it is only the new one that is available. Beside the author protection, then contents also should be avoided to be misused both inside and outside the platform. The video is more vulnerable against the misuse because of the large of unique data and no cover of the identity/identities in the video, then the video can be altered into improper purpose or copied into another place the author didn't approve. The contents, primarily videos should therefore only be available for the registered users and is proof against generate an identical copy of the video (it is difficult to prevent the screen record or storing the video in local machine when loaded the web browser).

If somebody wants to promote or demote someone or themselves, it may be necessary to connect the contents produced by the user with the user for allowing the reviewer to get some basis for the decision when voting. Then the information flow will be explicit for the selected reviewers to read a part of the data by the user, such as quantitive data (number of contributions, number of rejected contributions and so on) and qualitative data (present of n most recently contributions). It has to make sure that the reviewers don't get access to the unnecessary information, ie. name, e-mail address currently QCV that is not critical for the decision. During the voting process the user should not enable to exploit who did vote in specific voting by monitoring the available data as implicit information flow[Gol11, p.219ff]. Then the information on each users, post and voting should be present in this way that they don't exploit more information than necessary. It also should prevent anyone to figure out who did trigger the voting for demote of this user.

Chapter 5

Implementation

This section will address how the design of the platform was implemented, however the platform is an extension of the deployed Wign with smaller amount of complexity. The paper only will highlight the parts that were developed during the master thesis work.

Several parts in the source code are deprecated but not clean up yet because of limit time available to perform the task. They only are marked as deprecated which will be crossed out when typed as indication that the method/class should not be used.

5.1 Laravel Folder Structure

Laravel Framework has a standard set of folder that will be created when installing the framework. The structure is based on the MVC as design pattern with some modifications[LR01]. Some folders are only updated through calling the '`php artisan <commands>`' and '`composer <commands>`' commands in terminals and other contains the medias or template to allowing to present the view that is adapted to video-player environment that Wign is using CameraTag as API solution. The following folders are most relevance when perform the further maintenance; app, config, database, resources, routes.

public>index.php initials the entire platform when executing, but Model in MVC executes in *app*, View in *resources>views* and Controller happen primarily in *app>http* with assistance from *routes*.

The documentation how to run the framework on local machine can read in appendix A.

5.1.1 >App

The folder holds 10 models that each files defines which data that are available, relations to other tables and scopes of query to make the query call more easier to write and maintenance.

The relationships have three different kinds, depending on which one is referring on what and whether it is a many-to-many relation that it requires to use a pivot table (rhombus in figure 4.1). The relation can be one-to-many (`hasMany()`), many-to-one (`belongsTo()`) or many-to-many (`belongsToMany()`). All tables that are defined with a timestamp, will automatically add `created_at` and `updated_at` column. If the table also extends with "Softdelete" that lets none data be removed, but instead it adds the column `deleted_at` that is null as default, but will set a timestamp when the object is deleted instead of delete the entire row in the database. All the tables that have *-mark in figure 4.1 and 4.2 are extended with Softdelete.

The many-to-many relations have their pivot table with its timestamp, but to ensure that the timestamp will be updated properly, it musts add the clause (`withTimestamps()`) in the definition of relation in the model:

```
1  return $this->belongsToMany('App\<target model>', '<pivot-table>',
       '<source model id>', '<target model id>')->withTimestamps()
```

and if the pivot table also has additional column(s) beside the IDs and the timestamps, it has to write the clause into the definition of the relationship to make them available to use in the program.

```
1  return $this->belongsToMany(...)->withPivot('<name>');
```

Console-folder enables to define the process that has to execute on the server, not at client's machine. In *Kernel.php* at the method `schedule()` has the statement that will execute once every day to check if some votings have exceeded the date

of expiration to perform soft delete to close for further votes.

```
protected function schedule(Schedule $schedule)
 {
      $schedule->call(function () {
          Review::where('created_at', '>=',
               Carbon::now()->subDays(config('global.vote_duration'))
               ->toDateTimeString())->delete();
      })->daily();
 }
```

where `config('global.vote_duration')` is a constant that is loaded from *global.php*.

The next folder, *http*, contains both the controller of each business models and middleware that handles the authorizations. The controller is used to receive and send the information to the view or route which will take the look in detail later. The middleware is added with 'admin' and 'entry';

```
public function handle($request, Closure $next)
  {
      if (! auth()->user()->isAdmin()) {
          return redirect('/')->with('message', __('text.no.access'));
      }
      return $next($request);
  }
```

where the user without admin-role will be rejected. The entry middleware acts in the opposite way,

```
public function handle($request, Closure $next)
  {
      if (auth()->user()->qcv->rank == 0) {
          return redirect()->back()->with('message',
               __('text.no.access'));
      }
      return $next($request);
  }
```

that the user with `QCV(0)` will be excluded from the access. After adding the two new middlewares they have to add in *Kernel.php* before enabling to call them in route.

```
protected $routeMiddleware = [
    <statement of middlewares>
      'admin' => \App\Http\Middleware\Admin::class,
      'entry' => App\Http\Middleware\Entry::class
 ];
```

The files in *service*-folder are deprecated, because mostly of them are moved in the controllers.

5.1.2 >Config

This folder is this most sensitive of all first-level folders that with a couple of mistype will lead to malfunction or unavailable service. The folder also musts be not public available, because it contains the key for accessing to external API and password to database storage.

Config-folder behaves like a master room that is controlling all the connections and set of different modes that has to run the platform, ie. select of an algorithm of encryption can be found in *hashing.php* and setup of database connection in *database.php*.

There are two files that were created dedicated for this platform. *wign.php* contains basic data about the platform and key(s) to the external API(s), the clause in regular expression that is used to detect the hashtags in description. Additionally there is a list of URL-paths that was implemented before using the shortcuts directly on the route-file which will be explained later, then the list should not be updated and neither removed because there still are few lines of codes that still are using this URL-path.

All the constants that are used in multiple files in the source code, will be initialized in the *global.php*, i.e. maximum level of IL/QCV, duration of the voting, but expect the arrays (threshold, share of distribution ballots) that are stored in the same file that is defining the mechanism of voting processes. Php-language does not interpret the array as array in the data structure, but as hash then it has different behavior which cannot adapt as global variable.

5.1.3 >Database

Database-folder contains the foundation of the database structure that the *app*-folder is using to generate the models. The foundation is lying in *migrations*-folder where each models and relations have their own table with columns of data. The feature to connecting between tables are defined in foreign table. It is important that the foreign table must always on the bottom in the folder for ensuring that it will be executed (migrated) last. The tables will be migrated by the command `$php artisan make:migration`, or if the tables have been updated then the command will save some time `$php artisan migrate:fresh`

Seeds and *factories*-folder only is relevance for the testing purpose. The file with list of tables that have to be generated for the testing can be found in *seeds* > *DatabaseSeeder.php*. *Factories*-folder contains the fake-generation of each models that fill the instances with fictive data that may similar to the reality whose number of instances is determined in seeder-files in *seeds*. Seeder is also using to create the relations between specifics instances according to the expectation how the relations will be created by the users. The seeders will be executed with the migration when call `$php artisan <migration-command> --seed` or `$php artisan db:seed` to just initial the seeder without migration.

5.1.4 >Resources

This folder has three subfolders that are used in the platform, but *asset*-folder only contains the templates of different visual elements that are used to create the visual identity on the webpage, ie. colors, sizes and custom buttons. They have no mean in the implementation of reputation system.

lang-folder contains the text files that provides the list of text lines that will be present in pages of the platform. Each language has own folder (*da* and *en* so far)

Everything that is shown on browser, are coded in the *views*-folder. All those php-files that are a full template page, are lying right below *views*-folder. Those folders contains the parts that are used to fill up the full templates or has special template, for example 'form' that contains the elements allowing the users to fill the data, then it gets the extension of built-in protection mechanism that is protecting against XSS or SQL-injection attack then they are collected into one folder for make the maintenance of security easier.

5.1.5 >Routes

web.php is only used in Wign to redirect the views to the desired view according to the user's action(s). Each clause is one redirection that using either '**GET** or **POST** in HTTP-protocol. If it has to send a view, **GET** shall be used, or if the user has filled some information that has to send to the server, then **POST** shall be used.

```
1  Route::get( 'edit' . '/{id}', ['uses' => 'PostController@getEdit',
       'as' => 'post.edit'])->middleware('auth');
```

This first argument in `get()` method is the URL-address that will present on the browser. The URL address also can extends with a variable that can be used to load the variable (`{id}`) into the controller which in the case the variable will redirect to show the editor-page of post that has this id-value. The second argument is the destination in the controller the route has to redirect, however instead of spelling the entire literal of method `PostController@getEdit` there is added a shortcut that `'uses'` defines the target of redirection and `'as'` is the name of shortcut that will used when a method or view has to redirect the user to a page by typing ie. `route('post.edit')`. The get-method ends with a `->middleware('auth')` that before showing the page, the server will check if the authorization has succeed. `'auth'` is a built-in control that check if the user is authenticated in this session. `'admin'` was extended in this platform for allowing to verify that the user has admin-privilege.

Those routes that are defined in a group that are using the same middleware, can be added by to avoid repeating type the middleware in each routes:

```
1  Route::group(['prefix' => 'review', 'middelware' => ['auth']],
       function() {
2    Route::get( 'new', ['uses' => 'VotingController@postNewReview', 'as'
         => 'review.new'] );
3    Route::post( 'update' . '/{id}', ['uses' =>
         'VotingController@postUpdateReview', 'as' => 'review.update'] );
4  });
```

The setting of middlewares will be done in `>app > Http` where `>middlewares` contains the files that defines the property of middleware and in `Kernel.php` activate the specification of middlewares that shall be used in the platform.

5.2 Internal Data Exchange

The data often will be sent between views and controllers where there are used several methods for the exchange depending on whether the data was filled by the user or a data that has to send from an instance another instance.

The first way to exchange the data, generated by the user, is form where there will added elements on the page that let the user fills with the desired data. When the page is loaded into a controller, the data can be read by using Request class. The incoming data can be validated according to the expected pattern in the inputs:

```
$this->validate($request, [
          'word'            => 'required|string',
          'description'     => 'nullable|string',
          'wign01_uuid'     => 'required',
          <literals of validation check>
      ] );
```

If the range of possible data is finite, then there will be used the implicit value in `<input>`-object

```
<input type="radio" name="approve" value="true">
```

where the variable is set in the last argument `value="<variable>"`, and then it can be read by the same class.

```
$this->validate($request, [
          'approve' => 'required|in:true,false'
      ]);
$bool = $request->input('approve') == 'true' ? 1 : 0;
```

in this case "true" is interpreted as string-literal, then it has to convert to boolean before storing the inputs in the database.

If the variable is taken from the database that shall be used in the URL-address for allowing to share the link, then the variable can be read in `web.php`

```
Route::get( 'new' . '/{word?}', ['uses' =>
     'PostController@getPostIndex', 'as' => 'post.new'] );
```

`word` acts as a variable that '?' indicate that the variable is allow to be empty, then in controller file the variable can be accessed as argument (`$word`)

```
public function getPostIndex( $word = null ) {
 <statements>
}
```

Sometimes the user preferably shall not read the data that is exchanging explicitly around the form, then Session class can be used to push the data in the stack which can be pop when reading the data.

```
session(['oldPost' => null, 'newPost' => $post->id ]);
```

```
public function postNewReview()
  {
      $newPost = \Session::pull('newPost');
      $oldPost = \Session::pull('oldPost');
```

5.3 Implementation of Reputation System

The views have own debug that can be toggled in *global.php* that will present more information on the views for the purpose of debug or tracking a specific data, ie. id of the post.

5.3.1 Integrity Levels and Quality Confidence Values

All posts have only one integrity level (IL), but the IL can set to a number when user decide to modify this when the authorization allows this number. Simultaneously the IL values have to be stored because they are used in the voting to know which post and which IL the voting has to based on.

```
$post->ils()->delete();
$rank = $request->input('il');
$post->ils()->save(new Il(['rank' => $rank]));
```

where `$request` is the function that loading all inputs received from the form that filled by the user. There is no default value of IL. IL can never set to zero

because the value is reserved to entry users. Before adding the new IL, the old one shall be deleted (softly), but still will keep in the database. In the source code the integer value of IL and QCV is called 'rank', then accessing to the integer can by typing `$il->rank` or `$qcv->rank`.

QCV does change the value in same manner as IL, since QCV only can be promoted or demote, then introducing to new user with the QCV will be

```
protected function create(array $data)
 {
    $user = User::create([
       'name' => $data['name'],
       'email' => $data['email'],
       'password' => Hash::make($data['password']),
          'type' => User::DEFAULT_TYPE,
    ]);
    $user->save();
      $user->qcvs()->save(new Qcv());

    return $user;
 }
```

where the piece of code happened in *registerController* because it is a part of built-in login and register scenario in Laravel. A new instance of QCV will set its rank to zero as default in the setup of database table:

```
$table->smallinteger( 'rank' )->unsigned()->default(0);
```

Therefore it is possible to leave the arguments to be empty (`...new Qcv()...`). When the user is demoted or promoted, it will happens in *UserController*

```
public function postPromoteUser( $id )
  {
      $user = User::find($id);
      $qcv = $user->qcv();
      if ($qcv->rank < config('global.rank_max')) {
       $qcv->delete();
          $user->qcvs()->save(new Qcv(['rank' => $qcv->rank + 1]));
      } else {
          // Do nothing
      }
  }

  public function postDemoteUser( $id )
```

```
14      {
15          $user = User::find($id);
16          $qcv = $user->qcv();
17          if ($qcv->rank > 0)  {
18           $qcv->delete();
19              $user->qcvs()->save(new Qcv(['rank' => $qcv->rank - 1]));
20          } else {
21              // Do nothing
22          }
23      }
```

The change of QCV needs to check if it still stay within the range $(0-5)$. The same also goes to IL.

5.3.2 Voting Processes

The voting can be trigger by any users, but users with lower QCV are expected to trigger more votings than those with higher QCV. Posts with higher IL will be sent to voting more often than those with lower IL. The implementation of voting generation is constructed with determination of number ballots, selection of users for the voting and ballot allocation to the users.

```
1   private function _voteDistribution($election, $rank, $remotionUser =
        null) {
2       $count = Qcv::has('user')->where('rank', '!=', 0)->count();
3         $log = (log($count));
4         $numUsers = (int)($count/8)+($log*$log); // number of
              participants
5         if ($numUsers < config('global.min_ballots')) { // Only admins
              can decide the voting
6             return redirect()->route('index')->with('info', 'Afstemningen
                  er igangsat');
7         }
8
9         $rankMax = config('global.rank_max');
10        $rank = $rank < 1 ? 1 : ($rank == $rankMax ? $rankMax - 1 :
              $rank);
11        $dist = $rank >= $rankMax - 1 ? self::BALLOTS_DIST_2 :
              self::BALLOTS_DIST_3;
12        $voters = null;
13        for ($i = 0; ($i + $rank <= $rankMax && $i < 3); $i++) {
14            // Get all users within this rank and excluded this user who
                  created the voting
```

```
15              $v = Qcv::where('rank', $i + $rank)->where('user_id', '!=',
                    $election->user_id);
16              if ($remotionUser !== null) { // If it is a voting of
                    remotion, then the target user also is excluded
17                  $v = $v->where('user_id', '!=', $remotionUser->id);
18              }
19              $v = $v->inRandomOrder()->take($dist[$i] * $numUsers)->get();
                    // Make a random pick of n users;
20
21              if ($v === null)  {
22                  continue;
23              } elseif ($i == 0 || $voters === null) {
24                  $voters = $v;
25              } else {
26                  $voters = $voters->merge($v);
27              }
28          }
29
30          if ( $voters->count() < $numUsers) {
31              $diff = $numUsers - $voters->count();
32              $v = Qcv::has('user')->where('rank', '!=',
                    0)->whereNotIn($voters)->inRandomOrder()->take($diff)->get();
33              $voters = $voters->merge($v);
34          }
35
36          foreach ($voters as $voter) {
37              $election->voters()->attach($voter);
38          }
39
40          return redirect()->route('index')->with('info', 'Afstemningen er
                igangsat');
41      }
```

It is crucial to pay attention that the ballot is distributed to QCV-model, NOT user-model as intuitive.

The number of ballot is only depend on the number of users that are entitled to vote. The number is calculated by the equation (3.5):

```
1   ($count/8)+($log*$log)
```

where `$t` is logarithm of number entitled users.

Then the next step is about to determine who shall participate the voting accord-

ing to their QCV and rank of the subject in the voting. `$voters` are collecting the participants from 2 or 3 ranks. If rank of the subject is 4 or 5, there only will be two ranks that are candidate as participants. During the collection of participants there also has to ensure that the author of this voting is not allocated in the voting, also the target user if the voting is about remotion this user.

```
$v = Qcv::where('rank', $i + $rank)->where('user_id', '!=',
    $election->user_id);
        if ($remotionUser !== null) { // If it is a voting of
            remotion, then the target user also is excluded
            $v = $v->where('user_id', '!=', $remotionUser->id);
        }
```

Thresholds that are used in the voting are not defined in *global.php*, because php acts differently with initialization of array, then the thresholds are defined in *VotingController*

```
const BALLOTS_DIST_2 = [.6, .4]; // Used when two high-most ranks
    participate the voting
const BALLOTS_DIST_3 = [.5, .3, .2]; // Standard

const VOTE_WEIGHT = [1, 2, 3, 5, 8]; // Fibonacci

const LINEAR_THRESHOLD = [.5, .6, .7, .8, .9]; // Standard
const UNIFORM_THRESHOLD = [.5, .5, .5, .5, .5]; // Used if demote an
    user
```

No promotions or demotions shall be triggered simultaneous that is implemented in this code below. It ensures that the threshold of majority and presentation of the voting to the voters match to the aftermath if the voting is approved, ie. user musts only promote to $QCV(i+1)$ when the voting is based on $QCV(i)$. All the previous QCVS have to be included, because one promotion and one demotion can be triggered at once to avoid that an user is repeating trigger the promotion to prevent other to demote this user.

```
if ( User::find($id)->qcvs()->withTrashed()->whereHas('remotions',
    function($q) use ($p){
        $q->where([['decided', 0], ['promotion', $p]]);
    })->exists() ) {
        return redirect()->route('index')->with('message',
            "Afstemningen er allerede igang");
    }
```

Not implemented

When there are not allocated enough ballots as expected, the statement of code that shall execute the second round of allocation to all entitled users that are not yet allocated. The feature does not work properly since it has been an issue to exclude those voters from the collection of candidates that shall be used to do the second allocation.

When an user triggers or performs the aftermath of a voting, the execution ought not to happen at the client. Unfortunately the approach of implementing an internal process at the server that shall run when the event is triggered, is different than when if it is based on time. Then right now the processes will be executed at the client which will slow down the load of the browser when a new voting or aftermath is called.

5.3.3 Entry-user Restrictions

The implementation of restrictions on entry users are coded in multiple files, then this section will provide a short overview of source codes.

As mentioned all new users start with QCV on 0, which is defined in QCV's table. The entry users cannot make infinite many contributions, likes posts, create new request word and visit other users's page. They can be defined in *web.php* in *route*-folder like here

```
1  Route::post( 'save' . '/{id}', ['uses' => 'PostController@postEdit',
       'as' => 'post.edit.save'])->middleware('auth', 'entry');
```

where posting an altered post required the user to be logged in (`'auth'`) and is not an entry user (`'entry'`). The middleware of entry has the implementation here:

```
1  class Entry
2  {
3      public function handle($request, Closure $next)
4      {
5          if (auth()->user()->qcv->rank == 0) {
6              return redirect()->back()->withErrors('info',
                   __('text.no.access'));
7          }
8
9          return $next($request);
```

```
10      }
```

if the condition is true the the user will stay on the same page (`redirect()->back()`) and get the message that tell the user that the feature is not available.

The entry-users also are not entitled to vote. So when a voting is created and is going to allocate the ballots, the small literal of code ensure that they will not be allocated to any votings:

```
1   $rank = $rank < 1 ? 1 : ($rank == $rankMax ? $rankMax - 1 : $rank);
2         ...
3         for ($i = 0; ($i + $rank <= $rankMax && $i < 3); $i++) {
4               $v = Qcv::where('rank', $i + $rank)->where('user_id',
                      '!=', $election->user_id);
5               ...
6               }
```

that `$i + $rank` never is 0 then QCV(0) never will become candidate as voter, since `$rank` only can be greater than 0.

When the entry-user is creating a new post, the post automatically will be hidden and sent to the voting from *PostController*:

```
1   if ($user->qcv->rank == 0) {
2             $post->delete(); //Keep invisible during the voting.
3             session(['oldPost' => null, 'newPost' => $post->id ]); //
                  Will be loaded in VotingController
4             return redirect()->action('VotingController@postNewReview');
5         }
```

The rest of restrictions are coded in views for the communication purpose so the user will not see the buttons that are available in back-end, but stay visible on front-end. It includes the following files that are available for the entry-users to read:

create.blade.php where the drop-down selection of IL is hidden and the default value will be 1.

post.blade.php hides the edit- and like-button and the authors's name are not click-able for redirecting to their page that is used to trigger the promotion and demotion.

nopost.blade.php in the part where the user can click to add the request word, is hidden.

profile.blade.php that the information about the user's participation to the votings are not making sense, therefore they also will be hidden.

which those elements are covered with the if-condition:

```
@if ($user->qcv->rank > 0)
 <visible/clickable>
@else
 <invisible/non-clickable>
@endif
```

Not implemented

The authorization-mechanism in middleware does not work as expected, then the middleware is erased with manual check of authorization in each functions that the entry-user is not allow to enter:

```
public function getEdit($id)
    {
    <statements>
  if (Auth::user()->isEntry() == 0) {
            return back()->with('message', 'Du har ikke rette-rettighed');
        }
        <statements>
        }
```

The toggle of visible and invisible objects are not implemented in all view-pages.

The restriction of contribution capacity is also not complete in back-end. The approach can be implemented by using `created_at` in the post-model to check the count of contributions generated by the entry-user. As addressed in design-section entry-user only can generate one post around the request word and three posts from request word. The approach can be implemented by check if the user still is allow to post if the count of contributions from this day for this posts around request word is 0 or if the word of the post is requested then the count shall be less than 3. The approach shall be implemented in `postNewPost` in *PostController*.

```
if (user.contributions(today()) < 1 || (post.word.isRequested() &&
    user.contributions(today()) < 3) {
 <accept>
} else {
 <reject
}
```

There is only restriction of 1 post per day that is implemented. The less restriction of using request work when create post is more complicated, because the list of requests only will be hidden when there is a post connected to this word. Then the restriction has to check that there is not another post with the same word that contain the hidden request.

5.3.4 Post Management

A new initialization of post will run on the method `postNewPost` (the first 'post' indicate that it uses the **POST**-protocol in http and the rest tells itself. First the method has to validate if all the required informations are filled and match the input pattern.

```
$this->validate($request, [
          'word'            => 'required|string',
          'description'     => 'nullable|string',
          'wign01_uuid'     => 'required',
          'wign01_vga_mp4'  => 'required',
          'wign01_vga_thumb' => 'required',
          'wign01_qvga_thumb' => 'required',
      ] );
```

then the work is extracted to find if it exists in the database of words, otherwise it will initialize a new word

```
$word = Word::firstOrCreate( [
          'word' => $request->input('word') ,
      ] );
```

the initialization of video and description is different, because they will not belong to multiple independent posts, then they will be initialized immediately.

```
$video = new Video([
          'user_id' => $user->id,
```

```
            'camera_uuid'       => config('wign.cameratag.id'),
            'recorded_from'     => $request->input('recorded_from'),
            'video_uuid'        => $request->input('wign01_uuid'),
            'video_url'         => $request->input('wign01_vga_mp4'),
            'thumbnail_url'     => $request->input('wign01_vga_thumb'),
            'small_thumbnail_url' => $request->input('wign01_qvga_thumb')
        ]);
        $video->save();

        $desc = new Description([
            'creator_id' => $user->id,
            'editor_id' => $user->id,
            'text' => $request->input('description') === null ? "" :
                $request->input('description')
        ]);
        $desc->save();
        self::_updateTags( $desc );
```

the video-model also has `'playings'` as count which is set to 0 as default in the database. Text in the description loaded from the form may be a null, if the user has not typed anything. However storing the text as null will lead to troublesome, so the null will be converted to empty string. Then the text will be read through for the hashtags and adding them in the tag.

Now all the axioms are ready to combine into one post by sending them in the method `_create` which is a private method in *PostController*.

```
  private function _create($user, $word, $video, $desp, $rank, $creator
      = null) {
        $newPost = new Post([
            'creator_id' => $creator === null ? $user->id : $creator->id,
            'editor_id' => $user->id,
            'word_id' => $word->id,
            'video_id' => $video->id,
            'description_id' => $desp->id,
        ]);
        $newPost->save();
        $newPost->ils()->delete();
        $newPost->ils()->save(new Il(['rank' => $rank]));

        return $newPost;
    }
```

The argument `$creator` is distinguished that if the variable is null then creator

of this post is this user who created the post, otherwise the post was altered then the creator shall remain unchanged by put the creator id from previous post (`$creator`).

The alteration of post has similar statement as create a new post, but instead of just initialize all new axioms, the method has to check if this axiom has been modified which has to be initialized a new instance, otherwise just reuse the axiom from previous post. :

```
$newWord = Word::firstOrCreate([
            'word' => $request->input('word')
        ],[
            'creator_id' => $post->word->creator->id,
            'editor_id' => $user->id
        ]);

        $newVideo = $request->input('wign01_uuid') === null ?
            $post->video :
            Video::firstOrCreate([
                'video_uuid' => $request->input('wign01_uuid')
            ],[
                'user_id' => $user->id,
                'post_id' => $post->id,
                'camera_uuid'       => config('wign.cameratag.id'),
                'recorded_from'     => $request->input('recorded_from'),
                'video_url'         => $request->input('wign01_vga_mp4'),
                'thumbnail_url'     =>
                    $request->input('wign01_vga_thumb'),
                'small_thumbnail_url' =>
                    $request->input('wign01_qvga_thumb')
            ]);

        $newDesp = $post->description->text ==
             $request->input('description') ?
            $post->description :
            Description::create([
                'text' => $request->input('description'),
                'creator_id' => $post->description->creator->id,
                'editor_id' => $user->id
            ]);

        $newIl = $request->input('il');
```

Then for executing the alteration it has to check if the alteration has taken the place in the post then again check if the altered post has to sent to the voting or

just execute the change immediately. If the alteration is not detected, it checks if IL was set to new value by user with higher or equals to QCV which shall be updated, otherwise return the user to previous page with error message that no changes were found.

```
if ($newWord != $post->word || $newVideo != $post->video || $newDesp !=
    $post->description) {
        $newPost = self::_create($user, $newWord, $newVideo,
            $newDesp, $newIl, $post->creator);

        if ($user->isAdmin() || $post->il()->rank <=
            $user->qcv()->rank) {
            $post->delete();  //Only show the new one.
            $flash['url'] = URL::to( config( 'wign.urlPath.create' )
                );
            return redirect( config( 'wign.urlPath.sign' ) . '/' .
                $post->word->word )->with( $flash );
        } else {
            $newPost->delete(); //Keep invisible during the voting.
            session(['oldPost' => $post->id, 'newPost' =>
                $newPost->id ]); // Will be loaded in VotingController
            return
                redirect()->action('VotingController@postNewReview');
        }
    } else if ($newIl != $request->input('il') && $post->il()->rank
        <= $user->qcv()->rank) {
        $post->ils()->delete();
        $post->ils()->save(new Il(['rank' => $newIl]));
    } else {
        return redirect()->back()->with('info',
            config('text.edit.no.change'));
    }
```

Each post has their own collection of likes. The like-feature is implemented in *LikeController*. The like is attach to video in the post.

```
  $like = $post->video->likes()->attach($user);
```

if the post that is liked by this user and pressed the like-button again, then it will remove the like

```
  $like = $post->video->likes()->detach($user);
```

The sort of video presentation in the page with common word will be ordered

by the count of likes from highest on top to lowest in bottom.

Not implemented

The implementation of feature for allowing to report a post was not upgraded to this database structure, then the it is not in functionality. The users shall be allowed to report in the case that the post does not belong to this platform and need to be removed. For avoiding the destruction of contents the users cannot remove the posts themselves, but they can alert the admin about potential improper content. Then the flag-feature shall connect the post with the user that reported this with necessary information filled by the user and the post shall be hidden temporarily. Then it shall be a pivot table between Post-model and User-model.

The like-button loses its functionality when the platform was upgraded.

5.3.5 Request Words

All the functions that are a part of the request word is to be found in *Request-Controller*.

The list of requested word is executed from this method

```
1  public function showList() {
2       $limit = config('global.list_limit');
3       $requests =
            Word::doesntHave('posts')->has('requests')->withCount('requests')
            ->orderBy('requests_count',
            'desc')->orderBy('word')->paginate($limit);
4
5      return view( 'requests' )->with( compact('requests') );
6   }
```

the word-model has the relation both as request word and as post with word, then the query has to distinguish them by only collecting those words that are requested by the one or more users, but does not have a post yet. The list is ordered by the number of users that requested for a translation of this word.

If an user trigger a new request, the method `store()` is triggered and checking if the word-literal already is in the database before adding the request.

Not implemented

The controller was based on the IP-address, therefore it is not a method for removing the request because of weak degree of authentication that cannot be trusted that it will come from the same user when wanting to remove its request. Now the user-management is available, then it will makes sense to implement this feature, but it has to ensure that the users only can remove those who still have not a post, otherwise it will affect the authorization mechanism at the entry-users when their ability to contribute depends on whether the posts are posted from the list of request words.

5.3.6 Admin Role

The admin-role is defined in model `User.php` as constant value

```
1  const ADMIN_TYPE = 'admin';
2    const DEFAULT_TYPE = 'default';
```

that is accessible by `Auth::user()->type == <default/admin>` in views. Furthermore the middleware will be used for the authorization of accessing the page from route-file *web.php* (`->middleware('admin')`).

In menubar there is added a button if the user is an admin that redirecting to admin panel.

The admins are enable to ban or delete any users. The ban-feature can be done by visiting the user's page `guest.blade.php` which the pro- and demotion buttons are erased with ban-button, if the user has the admin role.

```
1  @if(Auth::user()->isAdmin())
2        <button id="btnBan" class="btn" style="background-color:#f1a899"
             onclick="location.href='{{ route('admin.ban', $user->id)
             }}'">
3            {{__('text.guest.ban')}}
4        </button>
5    @else
6        <button id="btnPromote" class="btn"
             style="background-color:aquamarine"
             onclick="location.href='{{ route('promotion.new', $user->id)
             }}'">
7            {{__('text.guest.promote')}}
8        </button>
```

```
9
10          <button id="btnDemote" class="btn"
                style="background-color:#f1a899" onclick="location.href='{{
                route('demotion.new', $user->id) }}'">
11              {{__('text.guest.demote')}}
12          </button>
13      @endif
```

To ensure that the votings still are based on the active users, but not disturb the result when remove redeemed ballots.

```
1   private function _detachBallots( $user )
2     {
3         $qcvs = $user->qcvs()->withTrashed()->get();
4         foreach ($qcvs as $qcv) {
5             $qcv->reviewVotings()->detach();
6             $qcv->remotionVotings()->detach();
7         }
8     }
```

The admins will do several tasks that some of their action may overlap each other which shall be avoided in most of cases, mostly when they shall involve in a pending voting. Then instead of allocate the oldest and pending voting to the admin there will pick a random voting to reduce the occurrence of handing the same voting to multiple admins.

```
1   public function getVote()
2     {
3         $review_count = Review::where('decided',
              0)->doesntHave('voters')->count();
4         $remotion_count = Remotion::where('decided',
              0)->doesntHave('voters')->count();
5
6         if ($remotion_count + $review_count > 0) {
7             $random = (bool) random_int(0, 1); // random pick to show
                  either review or remotion
8
9             if ($remotion_count > 0 && ($review_count == 0 || $random)) {
10                $remotion = Remotion::where('decided',
                      0)->doesntHave('voters')->inRandomOrder()->first();
11
12                $user = $remotion->qcv->user;
13                $user->created_at =
                      $user->created_at->toFormattedDateString();
14                $user->post_count =
```

```
                $user->withTrashed()->postsEditor()->count();
15
16              return
                    view('partials.adminRemotion')->with(compact(['remotion',
                    'user']));
17
18          } else {
19              $review = Review::where('decided',
                    0)->doesntHave('voters')->inRandomOrder()->first();
20
21              $newPost = $review->newIl->post;
22              $oldPost = $review->oldIl === null ? null :
                    $review->oldIl->post;
23
24              return
                    view('partials.adminReview')->with(compact(['review',
                    'newPost', 'oldPost']));
25          }
26      }
27      return redirect()->route('admin.index');
28  }
```

Not implemented

The feature that allow an admin to cancel the voting is not implemented. The feature shall help to reduce the number of votings that will obvious be voted down, ie. sexual context in the video. When the admin shall be involved in a voting, it has to ensure that the voting is not terminated. If the voting has voters that some of them did voted, then it will be necessary to use soft delete for allowing to respect those users who voted since their participant becomes valueless when the voting is decided by the admin. The soft delete shall be added in the table of review voting and remotion voting and all users are deleted softly and add the admin's final decision and execute the aftermath.

5.4 Testing

The platform has many information flow that are explicit, where the original implementation of Wign is simple and only have explicit information flow, therefore the test driven-development is not implemented in the platform. Instead the platform is tested manually with the help of use cases. Some of cases can be

validated through the browser where the the page returns with code of 200 and has no discrepancy elements on the page, ie. showing wrong video or unreadable description text. Other tests can be validate by checking if the database is updated with those data as expected.

Debug mode can be activated in *global.php* and imported in the view-file `$DEBUG = config('global.debug')` which will print more information on the views for improve the possible to track the data through the pages by insert the code snippet `@if($debug)`, and will be hidden hen deactivated.

Figure 5.1: The upper figure is in debug mode where id of this post is visible in the edit-button ("Ret"), and lower one is not in debug mode

The interaction between database and views also are tested by using the seeder and factory where the database is started with empty. For testing purpose it is preferable to see a realistic or look-alike contents that will be looks like in Wign's deployed environment.

Chapter 6

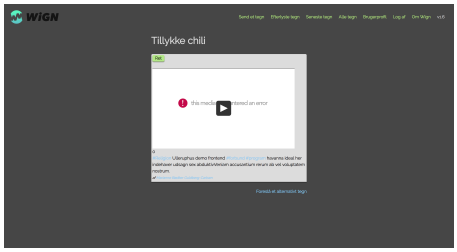
Evaluation

The implementing work is an extension of the existing platform as mentioned in the beginning in the paper. The extension is introduced to the new feature, the reputation system which contains several modules for obtaining the actual feature of being a reputation system that each content has their parameter of reliable that indirectly is controlled by the users through their individual contributions and community contribution by voting when the alteration of a post cannot be authorized by this user. Figure 6.1 shows the four essential page that is a part of the new implemented reputation system.

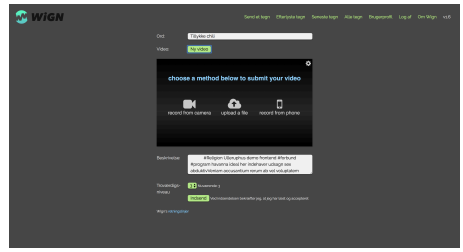
The page to create a new post is similar to figure 6.1b, but the edit-page will filled up with the data in those boxes when loading the page. The page also has the button which will be used when the user wants to record a new video, otherwise it will show the current video if the button was not pressed.

When an user triggers a voting that is based on the number of users that are entitled to vote that is 75 in this case, then the ballots will be generated to those users (figure 6.2).

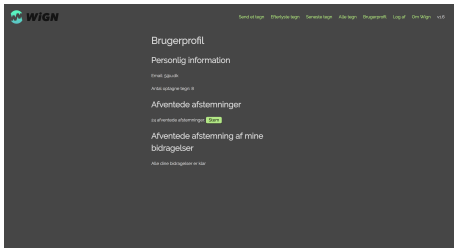
The restriction of entry user to contribute with a new content will meet when trying to enter the page to submit a post (figure 6.3a). When somebody searches a word that does not exist in the platform, normally the user will be encouraged to either create or make a new request, but entry users are not allowed to make



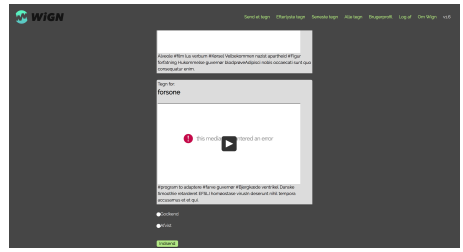
(a) Presentation of post(s) that the edit-button is added to each post



(b) Page to alter content of the post (word, video, description or IL)



(c) The profile page with the button that redirect the user to the allocated voting



(d) The page of information about the alteration of a post the user has to vote

Figure 6.1: Screenshots of the new extensions

a new request (figure 6.3b).

Administrators (admins) will get one more object on the menu-bar that will provides the admins the necessary overview of the current state in the platform and those that need the admins's attention. Currently there is only one button with count of votings that are pending on the action that has to be done by the admins (figure 6.4a). If an admin has witnessed an user that is acting maliciously, the admin is allow to ban the user from logging in the platform (figure 6.4b).

The scalability in wikiTrust is implemented by the entry-user approach which restrict the amount of contributions and those contributions are controlled by regular users. Simultaneous the slow process of expansion of the contents are prevented by using the request word that has less restriction of contribution by the entry-users. The If there is not users enough, then the admins will act the necessary work to grow the collection of posts and keeping the malicious users from getting too much power of influence (too high QCV).

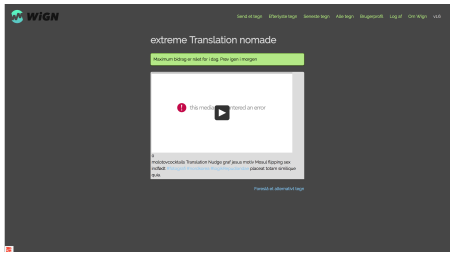
id	review_id	qcv_id	approve	created_at	updated_at	rank	user_id
1020	85	94	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	3	34
1019	85	308	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	3	105
1018	85	103	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	3	36
1015	85	1	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	3	1
1016	85	112	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	3	41
1021	85	19	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	3	8
1017	85	37	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	3	12
1023	85	76	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	3	25
1022	85	297	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	3	98
1026	85	309	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	4	106
1028	85	134	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	4	53
1025	85	161	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	4	60
1027	85	224	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	4	81
1024	85	47	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	4	15
1030	85	310	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	5	107
1029	85	23	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	5	9
1031	85	283	NULL	2018-12-26 23:33:22	2018-12-26 23:33:22	5	97

No errors; 17 rows affected, taking 1,5 ms

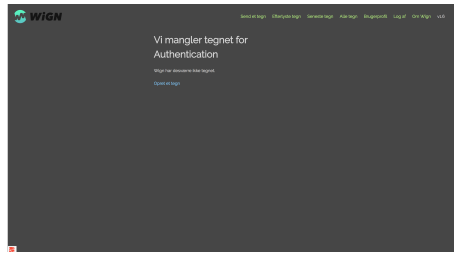
Figure 6.2: Each row in the table represent a ballot allocated to this user according to its QCV with the column of rank that will works as weight in this vote

The issue to implement the wikiTrust into a platform with no or low on number of content has similar solution as the approach of scalability. The main difference will be avoiding to let the users promote themselves cheaply by generate simple and well-known contents. The solution can be a preparation of who shall be invited into the platform with the task to fill up the platform. When an entry-user has contributed with 10 poor-valued posts and trigger the promotion, the voters still can reject the user if they consider that the user had used the cheap-trick to pass the entry-user. When the promotion is rejected, the user has to redo the activity to trigger the promotion once. The promotion and demotion always will require the voting, then we may expect that the invited users will act properly and check if the cheap-trick was not used to promote itself.

A content in the platform is constructed by posts that they each contain one word, one video and one description. Then the reputation calculation of each content is partly transferable, because it has to redefine the calculation so it is not depends solely on the text, but the meta context. Then in this case with crowdsource sign language dictionary the meta context is a post that is allocated an IL according to its author. Additionally the like-reputation is attached to the video, because the video is an unique and cannot be altered partly as discussed in section 4.6. Then the reputation calculation is erased with static level with like-feature as supplement for the mass-review by the users about this post.

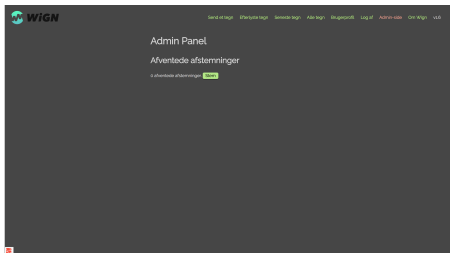


(a) The entry user has no edit-button and was rejected to contribute one alternative video with the same button

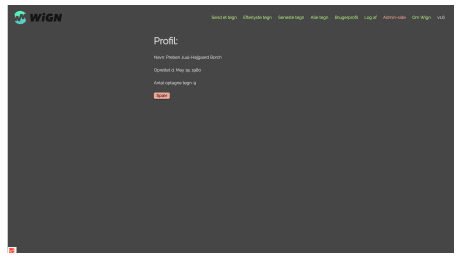


(b) The page does not provide the user to create a request for this word, because the user is an entry user

Figure 6.3: The restrictions of entry-user's features



(a) The button on this page will redirect the admin to voting that has no allocated ballots



(b) The page about other user will show the ban-button

Figure 6.4: The alteration in views when logged in as admin

Chapter 7

Discussion

7.1 Alternative setup of adaptability

When the minimum threshold of users are exceeded the voting will not be reserved to the admins. Throughout there will be not users on entire range of QCV which the allocation of ballots may too often need to allocate the leftovers to other QCV which will reduce the soundness against ie. framing by a groups of users or one QCV may dominate the voting. With the current setting if an user is promoting to the highest QCV as first person, then the voting will not enable to follow the normal allocation. So users on QCV_1 will be invited to vote the promotion of the user from QCV_4 to QCV_5 which is not desirable scenario.

The alternative approach can be a requirement that each QCV must be filled with least n users. Let us say that $n = 10$ then all votings of pro- and demotion will be decided by the admins until all QCV are filled up with users. The another alternative has more similar to dictatorship. When a voting will not obtain enough participants in first round, then the allocation will be cancelled and overtaken by the admins. Then through the time the users will be promoted and demoted several times, then there will sometimes occur a hole in the expected distribution of users in each QCV. By the dictatorship the soundness will not be weaken when in a sound number of users suddenly cannot obtain the enough participants in a voting, because the allocation will just be cancelled instead

of compensating by include users from other QCV or more from a QCV that already is invited in the voting.

7.2 Exclusion of lowest QCV in voting

In a specific case when the voting has $QCV \geq max - 1$ as origin then only two uppermost QCV will be invited in the voting ($QCV_{max} \cup QCV_{max-1}$). The decision to exclude QCV_{max-2} is based on that the user with lower QCV shall not allow to control the content with higher QCV, since the reputation system in the platform is setup that higher QCV is authorized to perform the action on lower QCV.

QCV_{max} will only requires to sent into a voting if the demotion of this user is triggered, otherwise user with QCV_{max} will never trigger a voting.

7.3 Set of weight in voting

In a normal case the voting has a collection of voters that has their own weight of vote that determine how much impact each shall get in this voting. All user can be demoted or promoted anytime which may happen that they are pro- or demoted right after they receive a ballot, but not yet have voted. The weight can either based on the time the ballot was allocated so it satisfy the distribution of power among the three QCVs that are participating in the voting, or based on the current QCV when the user voted then it matches the user's actual QCV.

The drawback to set the weight when the user votes lie in that the user may postpone to vote, because the user may hope to be promoted eventually to get more influence in the votings. It also will not satisfy the share of each three QCVs during allocation of the ballots, when the QCV can be changed until voted.

If the weight is based on the time the ballots have been allocated, then it may not be the actual state about each user, since their position may be changed, where somebody should get more influence, other less. The users who become demoted to QCV_0, will lost their ballot because their roles are changed to entry-user which has no right to vote. If the user voted before demoted to 0, the vote will not be removed.

The overview of pros and cons in the two kinds of vote weight is presented in table 7.1.

	Present QCV	**Allocated QCV**
Pros	Representative voting according to the current situation	It satisfies the distribution of the involved QCVs and the impact of each voters will not change through the time, then the users may vote immediately when they are logged in.
Cons	The user can wait to vote until the result from its promotion is passed. Avoiding one group of QCV is dominating the voting is not guaranteed	The weight of vote may not be representative. Some voters may lost its ballot when they is demoted to 0 which decrease the number of voters

Table 7.1: Comparison of different approaches in set of vote weight

7.4 Logarithm and Fibonacci

The WikiTrust paper discussed to use linear in value of the threshold and polynomial in weight of the votes.

The redesign the formula keeps the values of threshold, but changed the polynomial in weight of the votes with Fibonacci, then the voters with highest QCV still hold the most impact, but there will be easier for the lower ones to also keep the highest one in control. By using Fibonacci sequence there always will be $\mathcal{W}_i + \mathcal{W}_{i+1} = \mathcal{W}_{i+2}$. The capacity of impact in each cases of Fibonacci and polynomial are compared in table 7.2. It shows that using Fibonacci with amount of participants from each QCV as reverse Fibonacci sequence that the sum is 100% (50%, 30% and 20%), will be more fairly influence from each QCV.

7.5 Degrees of threshold

Each voting ought to require a certain threshold of majority that depends on the target post or target user the voting is about and which operation there will be initialized in the aftermath:

Fibonacci

$(n = 100)$	$i\ (0.5 * n)$	$i + 1\ (0.3 * n)$	$i + 2\ (0.2 * n)$	50%
$[1, 2, 3]$	50	60	60	85
$[2, 3, 5]$	100	90	100	145
$[3, 5, 8]$	150	150	160	230

Polynomial

$(n = 100)$	$i\ (0.5 * n)$	$i + 1\ (0.3 * n)$	$i + 2\ (0.2 * n)$	50%
$[1, 2, 4]$	50	60	80	145
$[2, 4, 8]$	100	120	160	190
$[4, 8, 16]$	200	240	320	380

Table 7.2: Amount of weighted impact from each QCV when all voters are voted the same within this QCV. The last column indicates the 50% of total weighted votes

$[50\%, 60\%, 70\%, 80\%, 90\%]$ or *linear threshold* is the current threshold that is used when voting the promotion of an user. The user will get enhanced in its authorization and more influence in the voting when promoted, therefore the threshold ensure that going to higher QCV require more acknowledges by the voters.

$[90\%, 80\%, 70\%, 60\%, 50\%]$ or *reverse linear threshold* is this one which is not used in the voting. This one can be used in voting about demotion that users on higher QCV will be easier to get demoted than the lower one to ensure that the those users on uppermost QCV always act properly. Those on lower QCV will not be demoted to entry-user easily. Unfortunately if a malicious user actual should demote to entry-user, it requires that there are active voters enough for allowing to exceed the threshold of majority. Then it may lead that mostly of users are on QCV_3 because those QCV lower than 3 are easy to promote but hard to demote and those higher than 3 is hard to promote and easy to demote. Therefore this threshold was not used in the voting because it will get too many users on QCV_3 and those malicious users will not demoted in some cases when they should, if there are not response enough in this voting of demotion.

50% or *uniform threshold* is used when voting the demotion of an user or make review for a post. It requires that half of weighted votes is half or more of the total weight votes. This threshold will both ensures the result is based on the majority and is not unobtainable when there are voters who have not voted after the date of expiration.

7.6 Extent of transparency

The users often want that everything is transparency in the platform, then they will fell powerful that they have access to much information about what happen on the platform. The developer and admins may want to curtail the amount of information that are not essential for the users nor the reputation system. Too much information may lead to undesirable actions, i.e. repeating the same voting when the result did not go as the user desire, framing an user when enable to stalk the actions performed by this user or most undesirable, using the information for the penetration attack.

7.7 Force a decision in the voting

The probability that the occurrence of improper contents will be bigger if they are from malicious users and their QCV are expected to be low. Then all their contributions are assumed to trigger the voting. If post in the voting is highly improperly, then it will not make sense to require the users to vote down. So in some cases the admins can improve the motivation of the users by letting them vote the proper contents by overtaking the voting with improper contents.

In the database it musts ensure to can distinguish the final result between approved and declined no matter whether it is decided by the voters or the admin(s). Additionally if the activity of the users are measured by their participations in the voting, then they will feel ignored when their votes are deleted from the voting.

Then if the admins shall allow to perform force decision into a voting, there will be necessary to include the soft delete on each ballot/vote. When the admin has perform the action, all the ballots which were not redeemed into a vote, will be clear from the database. The ballots that were redeemed will be soft deleted and adding one row with the admin's vote as indication of the final decision.

Perspective

7.8 Construction of the knowledge about this language by non-experts

The global community always requires that the qualified research work must be done by the people with right competences, because they have the awareness about the content of this field. Wikipedia encourages anyone to contribute and develop the knowledge about a specific field without any requirement to validate the contributor as an expert within this specific field. It is a traditional that the research work is done by scholars which may be change if the idea of wikiTrust is expanded in encyclopedia platforms.

This thesis has sign language as primary scope of the implementation. In the research community there are not many investors in research of sign languages. One of the explanation is that the sign language never is a majority language in any countries. The interesting of sign language exists at linguistic research departments. Beside there also is an indirect interesting in the sign language by the company that provides sign language interpreter to the people who want to establish the communication between spoken and sign language. The low on research to keep updating the knowledge about sign language lead to side-effect that the companies create their own internal system that their interpreters use to exchange the terms (beside the general information) used in different assignment they were sent to. Then the knowledge about newest terms in sign language often are to be found spread in sign language interpreter providers. It leads to commercialize the knowledge about sign language that the provider is used as value proposition for those who order the interpreter. Unfortunately in most

cases the collection of terms in sign language is maintaining by them that do not have the sign language as mother language. Because of duty of secrecy the collection cannot be valid by external person with sign language as mother tongue.

Then the redesign of WikiTrust in Wign will enhance the degree of validation at each term that appears in the platform, if the majority of the users have the mother language.

7.9 Explorer the name by familiar face

Community in this case is rather small, then people can easily identify name of the person only by recognize the face in the video. There it forms the basis of closed platform for the external to read the contents for allowing to protect some degree of confidentiality. Restriction of access from external users will reduce the risk that the videos will be copied and used that is not a part of the purpose that the registered users accepted when posting a new content.

Chapter 8

Conclusion

Outcome of this paper is an implementation of a web-based crowdsource platform as test-environment in Laravel Framework that is executable on local machine.

The platform has a collection of users and a collection of posts as primary objects. When a new user registered into the platform as an entry-user, its access to those available actions are restricted and the entry-user musts demonstrate to the reliable users that this entry-user is dignified user by performing the contribution in limit range until promoted. All the contributions either will be executed immediately or sent to the voting, if the author currently is not authorized to perform the action. The primary form of contribution can be create or alter a post.

The question about whether the WikiTrust is transferable into any crowdsource platform requires the redefinition of reputation calculation to handle on the meta-level instead of context-dependance. The core of reputation system is based on a subject as a reviewer that has a claim that can act as an action on an target which is an object in any characters that can be a content contributed by an user or the user as a contributor. Wign has word and description that is enable to perform partial alteration and video which is both unique and not enable to alter beside overwriting with one new video. They derive to a post that is assigned with an integer as Integrity Level (IL) determined by the author's

Quality Confidence Value (QCV) or lower but least 1. The IL controls the authorization to alteration that only users with QCV that is higher or equals to this IL, will be allowed to alter. Otherwise the suggestion of alteration will be sent to the voting. The posts also are supplied with a collection of likes that are generated freely by the users as a mass-reputation that shall work as an indication of the correctness in the entire composition in this post.

All platforms are started with nothing contents and there are two ways to build up the contents by either let a group whose users are trustworthy contributing with the basic contexts or implement a feature that is controlling the size of acknowledge giving to the user according to whether their contribution is high-valued or is requested by other reliable users. Wign already has the feature, request word that can be used in this purpose for leading the users to which content they are encouraged to generate as a contribution to the platform.

Some crowdsource platforms may have a small customer segment(s) which have the vulnerability to get controlled by majority malicious users when the register of an user is open to everyone. The solution of this problem can be obtained by make an invitation to dedicated users that are expected to be trusted as starting users or as admins. Those who registered themselves, will start as entry-user that has restriction, either not authorized to perform certain actions and accessing to specific informations or has limit number of actions within a stretch of time. The entry-users need to prove that they are worthy to be a regular user by making a serial of proper contributions and then trigger the voting of promotion for this entry-user which will be voted by the regular users or the admins, if there are not enough regular users in the platform.

Further works

At termination in this master thesis the implementation work can be improve with several features that will enhance the reputation system or make the reputation calculation more precise by adding the dedicated calculation that fits in this platform.

Clear purpose behind upload of new video

The current post structure only allow the video to be either added or overwritten. However it may be difficult to interpret the intention when adding a new video beside the exist post or overwrite the post with a new video, since an

expression in sign language also has its variation of expression. Then to obtain more precise measure of the intention the post can contain multiple videos that the first one acts as primary video that holds most likes and the other videos will act as variation of this primary expression. Then when an user overwrite a video, it means that the user find the expression in this video wrongly. If a video is added as variation, the user indirectly is acknowledging the video in this post. Last if the user create a new post with same word, it can understand as an expression that has different mean than the old post.

The work requires a small modification in the database which the time consumption is estimated to be roughly 5 hours work in redesign, implement the variation-feature and test this and other features that may be affected by the implementing.

Score system

The implementation of QCV does not tell much about the reliability of this user. A score system can be extended to make the possible to make a fine-measure of each user's level of reliability that is based on its activity in the platform by capacity of the contributions and participation in the votings, additionally the score system also will be reacted when the user's actions are cancelled by other users which can be remove of its contribution, voted down a voting triggered by this user. Then other users better can judge whether the content can be used if they also can read the score of this user.

The current implementation can be extended the feature by adding a new column in user table, if the score is irreversible. Then to collect all the way to allocate or deallocate the score into one place, it will be suitable to create a new controller in app-folder named *ScoreController* and create a config-file with the list of different size of score in each reachable actions for the easy overview. The duration of extension is calculated to be 3 hours, if there is a config-file ready to implement. Otherwise it has to goes through all the possible actions and evaluate which impact does each action do in the platform independently the background of the user, which will be 10 hours.

Critical sections in the platform

When a platform is accessible by multiple users simultaneously and they are allowed to alter the content, then the critical section has to take into account to prevent the collision when two or more users are altering the same post[And00].

Then if an user is altering a post, other users shall be blocked from entering the edit-mode in this post until the user complete or the server kick the user out of the edit-mode, if the user exceeds the maximum duration.

The voting also has its inopportune that other users will not enable to say if this post is in the voting for the upcoming review, because the post is not removed during the voting. Then another user may want to edit this post. At this current implementation there is not restricted to send different alterations of same post in voting, neither trigger a voting of pro- or/and demotion multiple times on the same user. The work to close the inconvenient error will require roughly 2 hours if the framework has built-in management of critical section, otherwise it can be programmed by using the timestamp in each user and post when deciding which one came first.

Chain of post history

Crowdsource platform with a good number of users are expected that the contents will be altered many times though the time. Some edit may be undesired by some users that they may want to reverse the alteration to the previous post. The platform now only has the track to previous post, if it was sent to the voting, otherwise it will not be possible to be certain which one post actual is the previous one.

The issue can be solve in 1-2 hours by adding a pivot table that tell which one post was altered to the new one. Then the tracking still is reachable when an old post is picked and erased the new one.

Exclusion of non-active voters

As discussed the drawback that there is not tracking the users who are not participating actively in the votings. It will lead to those voting that are declined whenever they are oughted to be approved. To improve the percentage of response among the users there can be added a temporarily exclusion or time-specified exclusion that a group of users will not considered as a part of the candidate when a voting is triggered during the exclusion.

The work will require some time for design to decide which exclusion is most proper for this platform. When the design is ready, there expected to use 3-5 hours for the implementing work and testing the feature and check if the ballot distribution still works properly.

Appendix A

Documentation

The source code of Wign is developed in Laravel Framework, then for the editor purpose PhpStorm will be a well chosen, because it provides the highlight of keywords and enable to jump to different files or functions.

The zip-file with source ought to be unzipped on the local machine. It is required to install Makefile and Docker for enabling to create a container to run the Wign environment on local machine. Unfortunately installation of Docker may complain if the machine has VirtualBox installed, because they have to overwrite the same place in bootkit. Make sure that VirtualBox is removed before installing Docker. The access to database also is a possible by using a mySQL-programme. 'Sequel Pro'[1] for mac and 'MySQL Workbench'[2] for Windows is suitable.

The installation file of Docker and tutorial is available in Windows[3] and mac[4].

1. Open command-lines program in your machine
2. Go to unzipped file with source code

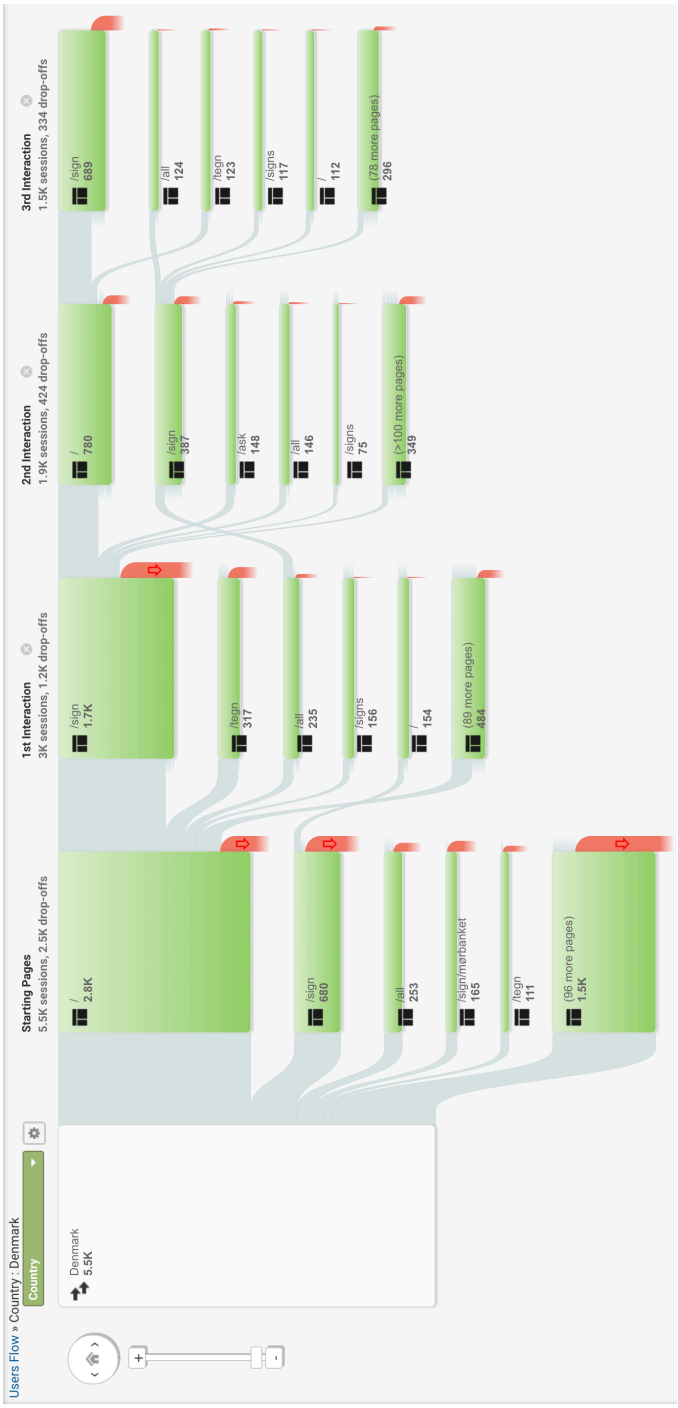
[1] `https://sequelpro.com/download`
[2] `https://dev.mysql.com/downloads/workbench/`
[3] `https://docs.docker.com/docker-for-windows/install/`
[4] `https://docs.docker.com/docker-for-mac/install/`

3. execute '`make up`' (it may run in few minutes)
    - (if issuing to start, try to execute '`rm .built`' and redo the make up.
    - if you prefer an empty database, then uncomment the entire statement below `run()` in *database > seeds > DatabaseSeeder.php*
4. open a browser and go to '`http://localhost:8080/`'
5. disable storing the cache in your browser, if you are going to modify the source code
    - to login with an user,
        - **username**: '`<x>@u.dk`' (where <x> is Integrity level that can be one number between 0-5)
        - **password**: '`user`'
    - to login as admin,
        - **username**: '`a@a.dk`'
        - **password**: '`admin`'
6. open a new tab in the command-lines
7. execute the file '`beamMeUpScotty.sh`' to enter the container
    - type '`php artisan`' to get list of commands
8. for accessing the database, open the database program (i.e. Sequel Pro)
9. create a new connection and type:
    - Host: db
    - Username: user
    - Password: pass
    - Port: 3306

Get fun

Appendix B

User Flow Chart


Users Flow » Country: Denmark
Country
Denmark
5.5K
Starting Pages
5.5K sessions, 2.5K drop-offs
/
2.8K
/sign
680
/all
253
/sign/morbanket
165
/tegn
111
(96 more pages)
1.5K
1st Interaction
3K sessions, 1.2K drop-offs
/sign
1.7K
/tegn
317
/all
235
/signs
156
/
154
(89 more pages)
484
2nd Interaction
1.9K sessions, 424 drop-offs
/
780
/sign
387
/ask
148
/all
146
/signs
75
(>100 more pages)
349
3rd Interaction
1.5K sessions, 334 drop-offs
/sign
689
/all
124
/tegn
123
/signs
117
/
112
(78 more pages)
296

Appendix C

About Wign

The file is developed internal in the team that will be available on the platform in English. The text was translated from Danish.

What is Wign?

Wign is a joint platform for users who speak Danish Sign Language as their mother tongue. Wign gives the users the opportunity to develop Danish Sign Language alongside with the emergence of new notions in society.

Wign is a social Sign Language encyclopaedia where the purpose is to give people the opportunity to contribute with and share signs from Danish Sign Language with each other in addition to contributing with new signs for notions. The contents of Wign have been made by and for Sign Language users in Denmark.

The purpose of Wign

It is to give the users a joint forum and an easy and efficient way to share signs with each other. It is our vision that a new sign should be easily proliferated. Wign is a supplement to Ordbog over Dansk Tegnsprog (The Danish Sign Language Dictionary) – see www.tegnsprog.dk. The difference is that Wign is expanded by the users whereas Ordbog over Dansk Tegnsprog is based on research in Danish Sign Language meaning that it takes time before a new sign is available.

The idea is to make Wign accessible for everybody and in more sign languages. We think it will lead to an increased focus on strengthening Danish Sign Language and the sign language users' consciousness about Danish Sign Language. In addition to the above, it will also minimise the number of intermediaries when a new sign needs to be proliferated. It also means that the number of variations of a sign will be reduced leading to a more standardised sign of a notion or term. It is a well-known phenomenon in Danish Sign Language that there are many different signs for certain words, especially technical terms. Not only is it the intention to standardise these signs, but also to enrich Danish Sign Language.

The users can use Wign by searching for a specific word, i.e. from Danish to Danish Sign Language. This optimises the time for e.g. deaf students to search for a certain sign – even if Wign has a broader target group.

Since Danish Sign Language is an independent language with its own structure and grammar which do not resemble Danish at all, it also means that it is not always feasible to translate directly from Danish to Danish Sign Language and the other way around. It is therefore an advantage to have in-depth knowledge of Danish Sign Language to be able to utilise Wign optimally. Sometimes, a sign is not equivalent to a word and the other way around. At times, it can be necessary to rephrase the sentence in the target language when translating it from Danish to Danish Sign Language.

Wign encourage the users to be critical of signs appearing in Wigns. Some signs can be considered a suggestion of a possible translation whereas other signs give a conclusive translation. It depends on the context.

The team

The team behind Wign consists of Troels Madsen and Kenneth Andersen. We are proficient in software and IT. It is our task to ensure that Wign is always accessible for all sign language users and to update the platform concurrently with the needs of the users and the development of technology.

We're also putting an effort in collecting all existing videos from external parts and companies. These videos show separate signs. This is to avoid them disappearing with time and to make them easier to find again. Until now, Streetsigners and 'Center for Døveblindhed og Høretab' (Centre of Deaf-Blindness and Hearing Loss) have donated their materials to Wign. We're very interested in getting signs from internal sign collections in workplaces who have sign banks.

The history of Wign and varying signs

Danish Sign Language has existed since 1807 when Peter Atke Castberg implemented teaching of the Deaf in Denmark. A language environment creates a common understanding for the meaning of each form of expression. Over the course of time, more and more deaf people have begun attending university leading to their spending less time in their own language environment, i.e. they read everything in Danish and English – not in Danish Sign Language – and other students are more likely to be hearing. This means that they study alone reducing the possibility for them to pick up new signs for technical terms from other deaf people. It's a prerequisite for a language to be stimulated by means of consistent interaction between language users.

The need of a platform for people to share signs was present for a long time, from that point of time when deaf people began bringing sign language interpreters along for their educations. 1966 marks the year when Asger Bergmann was the first deaf person which attended a youth education programme. Since then it's been common for deaf people to reinvent the wheel: they made a new sign for a notion without knowing that there was already an existing sign. It's assumed that the existing sign was already created by a former student, but since the sign language users in society consist of fragmented groups not in contact with each other, it's difficult to avoid this.

In the first years of this millennium, the number of sign language interpreting providers increased. All providers had their own internal sign bank which was only accessible for their own employees. It was normal for a sign bank to consist of either a transcription, cf. http://dansktegnsprog.dk/hjaelp/hvordan-transskriberer-man-dansk-tegnsprog/, or a written description of the sign, i.e. an explanation with Danish words as how to carry out the sign correctly. A big part of the documentation of how to do a sign for a specific meaning or notion in Danish Sign Language is explained in Danish because it hadn't been feasible to record oneself and upload the video internally. It's been a challenge to carry out a sign correctly based on a written description of its movements and forms. A big part of the technical terms hasn't got homogeneous signs. This makes it more difficult for people to explore something in depth within their own profession in Danish Sign Language. Sometimes, it has been necessary to make sure which notion was used to the various signs.

The idea of creating Wign.dk stems from a post by Kenneth Andersen on the blog of the Danish Deaf Youth Association (DDU) in 2009. A sign language interpreting provider (12K) read the blog and contacted DDU. 12K were interested in starting a project. The first version of Wign was launched in

January 2012 and got a lot of support from sign language users. Unfortunately, the success didn't last long because there was an abundance of errors making the experience of using the platform a bad one for the users. The number of visitors fell shortly thereafter. Few users continued uploading new signs to Wign. Moreover, Wign was closed after a couple of years due to collaboration difficulties with the website hosting service. The errors remained unsolved.

The re-opening of Wign

People, however, kept wanting the concept. Troels Madsen contacted DDU in 2016 to offer that Kenneth Andersen and he should undertake the project. Shortly after, in 2016, the website was re-opened, albeit in a leaner version where only the core functions have been kept. Wign was later subsidised by a grant from DUF (Danish Youth Council) and from Fonden for Entreprenørskab (The Danish Foundation for Entrepreneurship). Later the same year, Wign was founded as an entrepreneurial company.

Guidelines

The persons behind Wign do not discriminate in favour of or leave out a sign. Wign strive to secure equal views of each sign. Wign also take part in making sure that the guidelines and rules of how each sign should be expressed are followed for all contributions and in following up on all reports from a user.

Appendix D

Database Contents

Users

Name	Type	Specification
id	integer	primary key
name	string	
email	string	unique key
password	string	store in hash
type	string	'default' or 'admin'
last_login	timestamp	
ban_reason	text	

Posts

Name	Type	Specification
id	integer	primary key
user_id	integer	
word_id	integer	
video_id	integer	
description_id	integer	
creator_id	integer	
editor_id	integer	

Words

Name	Type	Specification
id	integer	primary key
word	string	
creator_id	integer	
editor_id	integer	

Videos

Name	Type	Specification
id	integer	primary key
playings	integer	
user_id	integer	
video_uuid	string	
camera_uuid	string	
video_url	string	
thumbnail_url	string	
small_thumbnail_url	string	

Descriptions

Name	Type	Specification
id	integer	primary key
text	text	
creator_id	integer	
editor_id	integer	

QCVs

Name	Type	Specification
id	integer	primary key
rank	small int	
user_id	integer	

ILs

Name	Type	Specification
id	integer	primary key
rank	small int	
post_id	integer	

Reviews

Name	Type	Specification
id	integer	primary key
decided	boolean	
new_post_il_id	integer	
old_post_il_id	integer	
user_id	integer	

Review votings

Name	Type	Specification
id	integer	primary key
approve	boolean	
review_id	integer	unique
qcv_id	integer	unique

Remotions

Name	Type	Specification
id	integer	primary key
promotion	boolean	(promote or demote user)
decided	boolean	
qcv_id	integer	(target user through its qcv)
user_id	integer	(created by this user)

Remotion votings

Name	Type	Specification
id	integer	primary key
approve	boolean	
remotion_id	integer	unique
qcv_id	integer	unique

Likes

Name	Type	Specification
id	integer	primary key
user_id	integer	unique
video_id	integer	unique

Tags

Name	Type	Specification
id	integer	primary key
tag	string	unique

Taggables

Name	Type	Specification
id	integer	primary key
description_id	integer	unique
tag_id	integer	unique

Aliases

Name	Type	Specification
id	integer	primary key
child_word_id	integer	unique
parent_word_id	integer	unique
user_id	integer	

Request words

Name	Type	Specification
id	integer	primary key
word_id	integer	unique
user_id	integer	

Bibliography

[And00] Gregory R. Andrews. *Foundations of multithreaded, parallel, and distributed programming - Chapter 3: Locks and Barriers.* Addison-Wesley, 2000.

[Ass] Danish Deaf Association. Spoergsmaal og svar - ÃŸh, hvordan taler jeg med en doev? (danish).

[Cab15] Marko Cabric. *Confidentiality, Integrity, and Availability - Chapter 11.* 2015.

[Com] EU Commission. EU's general data protection regulation.

[CPea15] Stefano Cresci, Roberto Di Pietro, and Marinella Petrocchi et. al. Fame for sale: Efficient detection of fake twitter followers. *Decision Support Systems*, 80:pp. 56–71, 2015.

[EANGdG15] Enrique Estellés-Arolas, Raúl Navarro-Giner, and Fernando González-Ladrón de Guevara. Crowdsourcing fundamentals: Definition and typology. *Advances in Crowdsourcing*, pages pp. 33–48, 2015.

[FG10] F. Randall Farmer and Bryce Glass. *Building Web Reputation Systems.* O'Reilly, March 2010.

[FOO93] Atsushi Fujioka, Tatsuaki Okamoto, and Kazuo Ohta. A practical secret voting scheme for large scale elections. *Advances in Cryptology - Auscrypt '92. Workshop on the Theory and Application of Cryptographic Techniques Proceedings*, 1993.

[Gol11] Dieter Gollmann. *Computer Security.* Wiley 2011, 3rd edition, 2011.

[LJ12] Kasper Lindberg and Christian Damsgaard Jensen. Collaborative trust evaluation for wiki security. *In Proceedings of the Tenth Annual Conference on Privacy, Security and Trust*, July 2012.

[LR01] Avraham Leff and James T. Rayfield. Web-application development using the model/view/controller design pattern. *Proceedings Fifth IEEE International Enterprise Distributed Object Computing Conference*, 2001.

[oDSL] The Department of Danish Sign Language. Brevkasse: Hvor mange tegn er der paa dansk tegnsprog? (danish).

[SdV01] Pierangela Samarati and Sabrina De Capitani di Vimercati. Access control: Policies, models, and mechanisms. *Lecture Notes in Computer Science (including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2171:pp. 137–196, 2001.

[Wik] Wikipedia. Wikipedia:protection policy.