Reputation System for Collaborative Community

Computer Science And Engineering Department

Shubhendra Pal Singhal
Kaushik Sridhar
Punit Agrawal

Indian Institute of Technology , Bombay
April 3,2018

Department of Computer Science and Engineering

INDIAN INSTITUTE OF TECHNOLOGY, BOMBAY

Certificate

The project entitled Reputation system for Collaborative Commnunity submitted by Shubhendra Pal Singhal,Punit Agrawal,Kaushik Sridhar is ap- proved for Summer Internship 2018 programme from 16th May 2018 to 6th July 2018, at Department of Computer Science and Engineering, IIT Bombay.

Names of Students

_____

Shubhendra Pal Singhal
Kaushik Sridhar
Punit Agrawal

Mr. Nagesh Karmali
CSE Dept, IITB
Project In-charge
5cmProf. Deepak B. Phatak
Dept of CSE, IITB
Principal Investigator

Date:July 6,2018
Place:IIT Bombay,Powai

Contents

1   Acknowledgement

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend my sincere thanks to all of them.

We are highly indebted to Prof. D. B. Phatak , Nagesh Karamali and Abhijeet and Nitin for their guidance and constant supervision as well as for providing necessary information regarding the project also for their support in completing the project.

Our thanks and appreciations also go to all the team members in developing the project and people who have willingly helped us out with their abilities.

2 Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Shubhendra Pal Singhal
National Institute of Technology,Tiruchirappalli

Kaushik Sridhar
University College of Engineering,Osmania University

Punit Agrawal
National Institute of Technology,Patna

3   Abstract

This project aims to develop a reputation system for System of Collaborative Community . The organized workflow would be ensured but every activity of the person would be governed by only one person while in automating the reputation system, the peer to peer reputation model[10] has been implemented where one person who is the publisher can help a person who is a author by suggesting the changes which article requires.

4   Why Reputation System?

In collaborative Community , the user can generally have the roles of publisher,community-admin by the permission of super-admin,but this does not ensures fairness in the system so this is why instead of having control in one hand[10].

The super-admin is the person who controls the whole workflow of the system and concentration of power in one hand is not advisable at any time.

The implementation of reputation system consists of two major roles:
1. Implementation of Upvote and downvote button.
2. Implementation of change in user reputation based on the role and upvote and downvote which needs a research model where all the specifications required for updation of reputation is implemented.
3. The difference between system reputation and community reputation.

We have automated the system where the reputation points of system and community helps in deciding the roles which ensures that the system provides a fair chance to all the people in the system and the role can be given based on their time devotion to various activities like editing,drafting article etc. in the system. Automation reduces the workload of the super-admin and ensures that he just have the role to manage the system and not dominate it by concentration of power in his hands. Basically , power has been spread out to different people across communities to ensure fairness in the system and thus automation helps in distributing the power.

5   Entity Relationship Diagram of Collaborative Community

The relationship between the different groups and community is well explained by using a technique ER diagram.
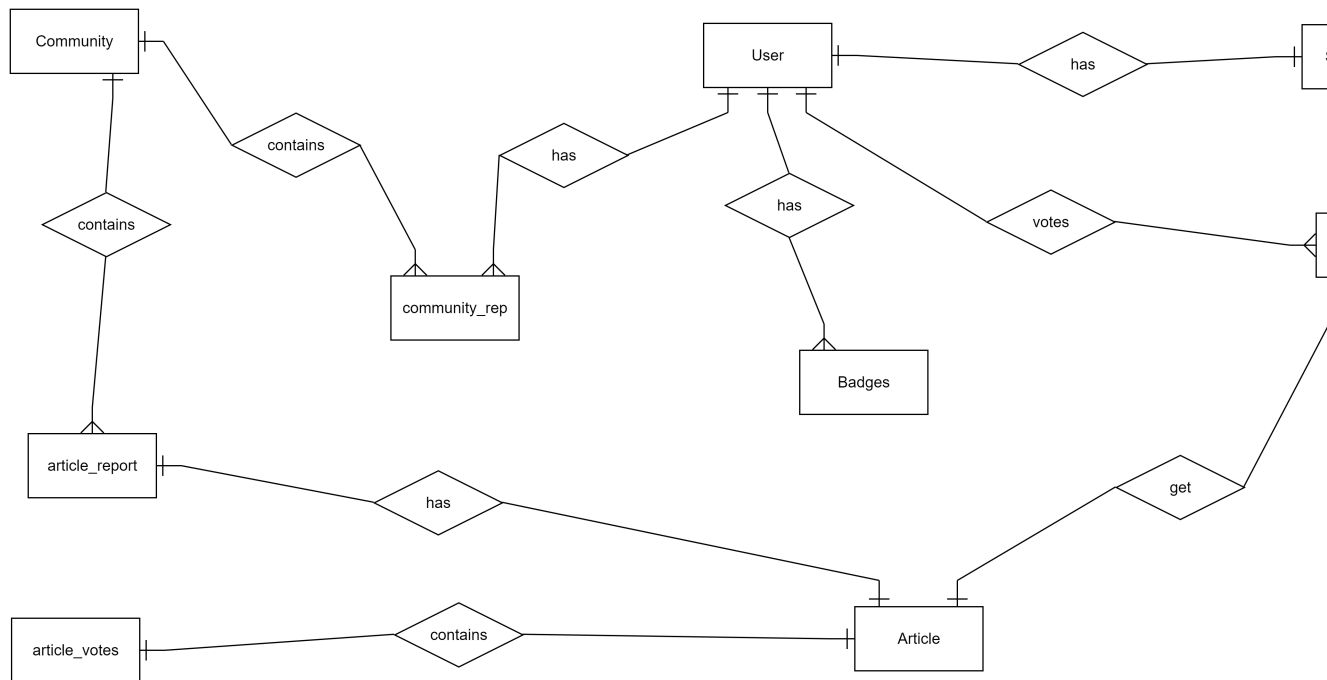
6 Introduction

Reputation Systems are the systems which allows to present the actual statistics about the active participation of users and their respective badges. The reputation system built so far have always been for the whole system where the reputation factor increases with respect to the domain of contribution. In Collaborative systems the major problem we faced was that reputation has to be implemented for both the community as well for the whole system.

The system includes various communities and a user can be a part of various communities ,thus having different roles in communities. So the overall participation measure could be done if the concept of system reputation could be implemented. The main problem faced again was that this reputation calculation is code intensive and thus an easy way out was brought that if we make a separate model for reputation and try to call it as soon as community reputation for that user changes,it will ensure a smoother updation which the user can see from his own dashboard. The next problem that faced was the implementation of badge system where the badges needs to decided for small steps taken by the user for encouraging more participation from the users and trying to make the model even bigger.
The reputation for community is the symbol of the role of who is going to be the next publisher and the total reputation of a user in the system introduces a concept of who is more learned and active in the system[1]. The respective problems of choosing parameters of selection of publisher and allocation of votes to different users in the system has been solved in this project and we will also be discussing about the implementation of reputation model as a whole which is compatible for any system.

6.1 System of Collaborative Community

Collaborating Communities project[2] is a web based application which allows users to create groups within the community, join group, add their content: mainly articles or collated articles. User has an option to join different communities. Person can be a member of multiple groups. Each group has different roles like author, publisher, etc. Author of a group can submit content related to that group. The content submitted will be visible in that community. There can be versions of the same content. The content is published in the community only after it is published by the publisher. The published content cannot be deleted. The content can be plain text or text with videos, images, and links.

The System for Collaborative Communities is a medium for users to create communities based on their ideas. The community can be open to all, or a closed community. The community members can contribute to the community by participating in various activities happening in that community such as adding various contents like articles, collating articles related to education.
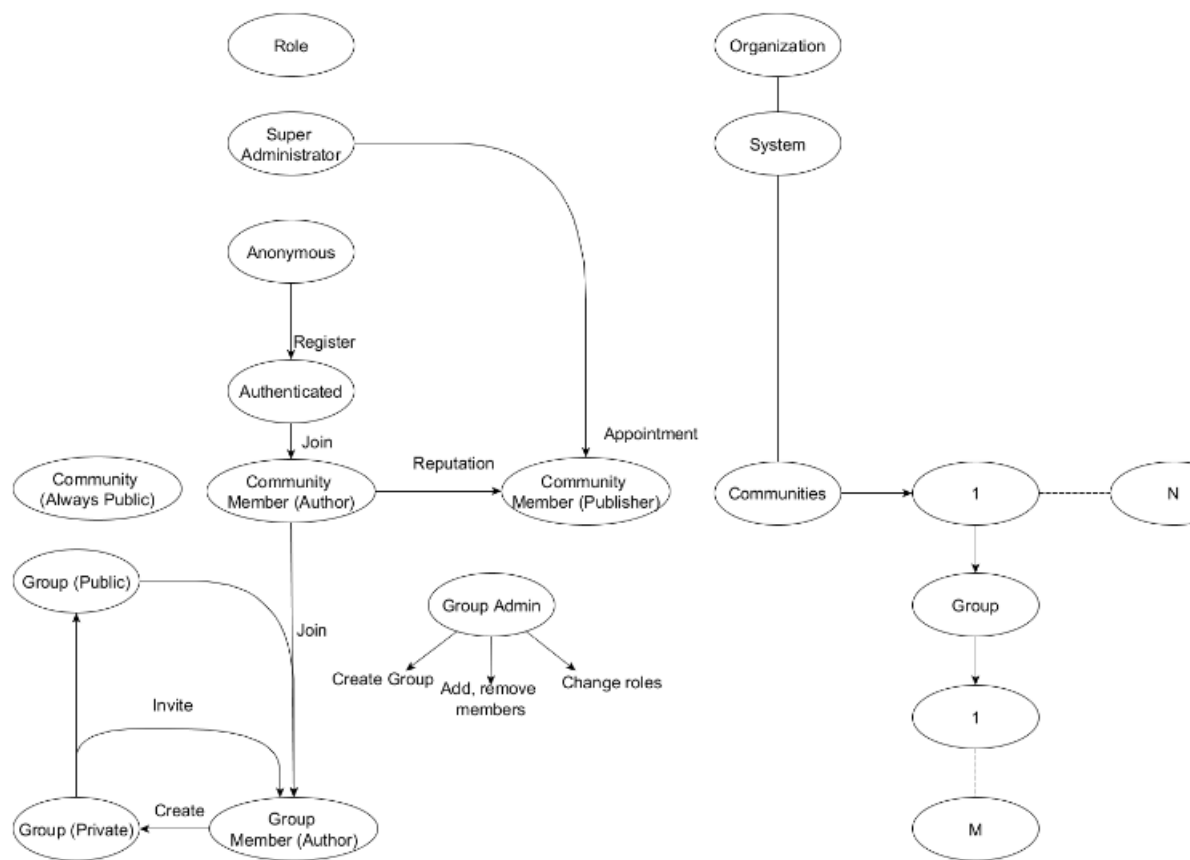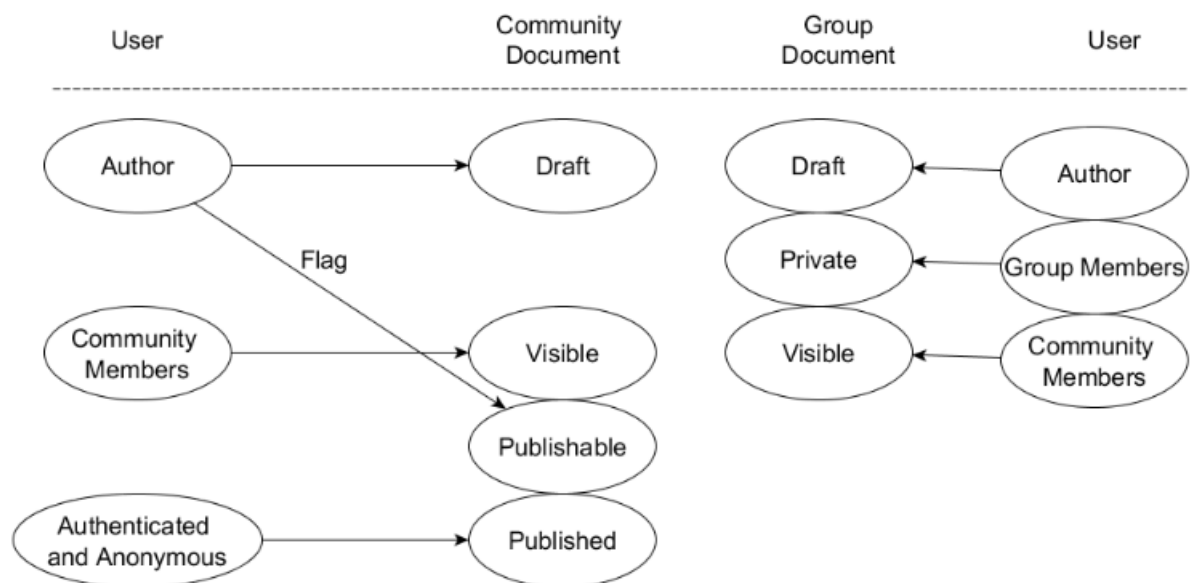
Fig 1: User Description

Fig 2: Article Types

7 Workflow of the Reputation System

7.1 Role Allocation and Dependancy

The super-admin is the creator for whole platform of collaborative system which has just one role with respect to to reputation system and will also be serving the role of publisher for every community for the first month of implementation. Role of the super-admin is to approve a request of creation of community by a user and he can approve this request if and only if there is no other community existing with that same concept. The community created will be tagged with all those names which can belong to the same community,thus avoiding the confusion of how community creation can be wisely judged by just a search.
. The authenticated users who will be interested in a community (by seeing the description) can join and draft/edit the content. One user can always be in multiple groups and then improve existing articles or content. The communities, thus formed, will have the roles of author ,publisher(only after 1 month) and community-admin. Those who draft the article or any content have the option of either privatizing their groups or they can make it public so that everyone can collaborate. If the group is made public then group-admin has the power to just remove them. But if it is private then the group-admin will be the first person in the group and has the control of granting permissions of entering and removing the group. The people who feel that they can contribute can join the group. Group in turn has just two roles open . First is the group-admin,then users who contribute to make the article reach ,publishable state and last is the author. As a user can be in more than one community, two types of reputations are maintained :
1.) Community Reputation : The reputation of user in a particular community which has been obtained based on the grading system given under section 1. 2 for community.
2.) System Reputation : Article in Stack overflow corresponds to one community in our system and user participation might differ for different communities,so the allocation of points should be different. Case studies on Stack overflow[4] has shown that they do not give importance to the above problem and thus addressing this problem we have developed a formula.
The reputation that is the cumulative of all the community reputation that user is a part of.

8 Research Model

8.1 Grading System

The upvote in an article is a measure of how good is the version of the article and downvote is a measure of how can a version improved[1]. The point allocation system has been developed considering various factors including the chances of malpractices and certainty of cracking the system.
The point system is as follows:
Creating community (approved by the superadmin):+25
Article published : +5 to the user of last edit and to others(explained in section 2.4), +5 to publisher.
Article Reportedwrong(Approved) : -5 to the user of that version and publisher gets -5.
Article Reportedwrong(Rejected) : -5 for the user who reported.
Article/comment upvote/downvote: +/-2

8.2 The Badge system

Community Author 0 to 2000
Community Publisher 2001 onwards[3]

The table below shows all the stages of badges allocation[3] :

Activity	Badges				
	1 to 5	6 to 15	16 to 30	31 to 50	Above 50
	Author / Publisher				
Articles Contributed	AC	AC	AC	AC	AC
My Articles Published	AP	AP	AP	AP	AP
	Publisher				
Articles Revised by me	RA	RA	RA	RA	RA
Articles Published by me	PA	PA	PA	PA	PA

9   Glimpse of Implementation of the System

The system has been implemented in a very straightforward manner. The implementation of upvote,downvote button and flag for reporting the article is merged with the existing template. Usage of these buttons will change the number of upvotes and downvotes in the version of the article and then according to the condition specified in grading system, user reputation will change accordingly as that moment only.

There are several case studies which will be based on different scenarios[8] :

1. **Person Joining the community** : The user gets +25 reputation on his dashboard and his system reputation also increases by +25 at this moment. If the user is new to the system , then he gets +25 as the starting reputation.

2. **Person editing the article** : The person reputation increases as soon as he hits "save" button which inturn increases his reputation by +2 for initial implementation.

3. **Person reporting the article** : The person who reported the article has to give the reason of reporting the article. This article goes in the tab of "Articles Reported" where users of that community can view that reported articles. The community admin will have an additional option of approving or rejecting the flag.

3. 1 If he approves the request, the reputation of publisher goes down by 5 and user of that last edited version by 5. The reputation of person who reported increases by 5.

3. 2 If he rejects the request, the reputation of person who reported will be decreased by 5.

4. **Person changing to publishable state** : The publisher gets the notification to review the article. If publisher accepts the article and publishes the article, then the publisher would get +5 points,author +5.

10 Detailed Implementation of Reputation system

10.1 Implementation of Upvote/Downvote
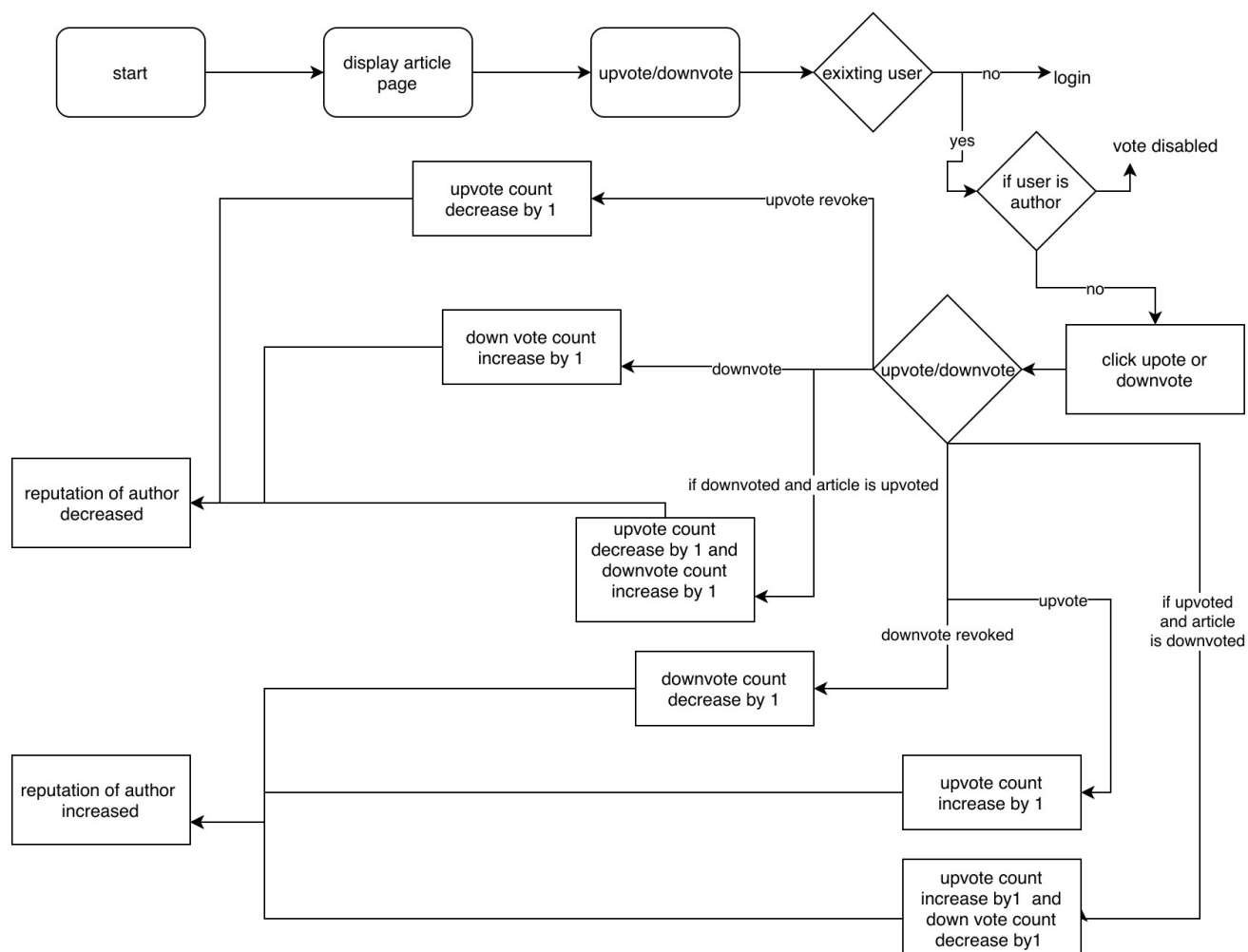
Opening of start page shows top articles and when a user click them , then he is directed to page of display article. Existing user can signup to edit the article or upvote or downvote. The user is not authenticated has just the right to see the article and no other right like upvoting etc. is given to him.

The author of the article should not be able to upvote his article. So a check is maintained whether the user is author. If he is a author then the upvote or downvote button both are disabled. If the user is not the author of that article, then the click too has 4 types which is explained in detail in following points below :

1. Click upvote/downvote button : The upvote/donvote button on clicking means that downvote/upvote button is disabled and the count of the upvote/downvote of article needs to be increased by 1.

2. Double upvote/downvote button : The upvote/downvote button is not clicked now and simultaneously the count is also decreased.

3. Click downvote while upvote button is enabled : The user wants to remove the upvote button and wants a downvote instead. So the upvote button is unclicked and downvote is clicked which leads to decrease in upvote by 1 and increase in downvote by 1.

4. Click upvote while downvote button is enabled : The user wants to remove the downvote button and wants an upvote instead. So the downvote button is unclicked and the upvote is clicked which leads to decrease in downvote by 1 and increase in upvote by 1.

The implementation is clearly described in the model Voting with views. py as business logic mentioned above. All the corner cases have been both handled internally in database and also has been implemented for the ease of the user to understand the concept of upvote/downvote. There are certain rules and regulations that the roleplayers have to follow to understand the system. The author not being able to upvote his article is implemented because of the very fact that every author instead will try to upvote his article which might lead to misleading result because one upvote can cause a lot of difference in the system.

The implementation of the functions described above in views. py are under one function updown. The function has parameters article-id and type which indicates the above types mentioned,not order wise. The implementation is using the concept of upflag and downflag which indicates if the upvote or downvote is clicked or not which ultimately helps the system to decide that how the user is reacting to the given particular situation.

The next big thing that was to be implemented was the simultaneous change in reputation of different roleplayers depending on the situation they fall in provided they are following the default grading system otherwise the form type will ensure the required changes in the default values. This section is described in the next segment below.

## 10.2   Implementation of Reporting of an Article
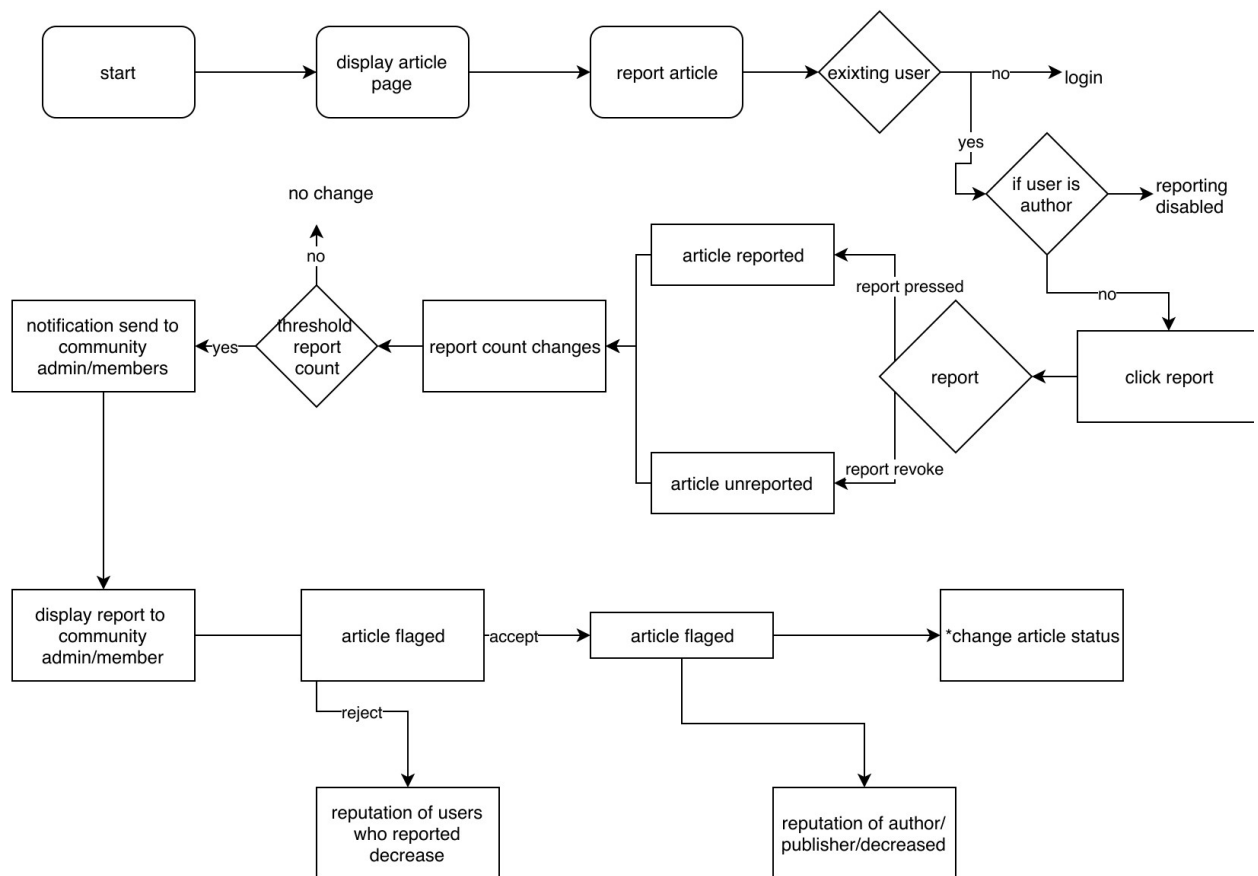
Opening of start page shows top articles and when a user click them , then he is directed to page of display article. Existing user can signup to edit the article or upvote or downvote. The user is not authenticated has just the right to see the article and no other right like upvoting etc. is given to him.

The user if author will obviously not report his own article and thus all the other authenticated users will click the report by flagging the flag. The flag has two types

1. Flag clicked : The flag is flagged and is brought to the section of the tab of "Articles Reported".

2. Flag doubleclick : The flag needs to be reported and the article is removed from the tab section if it does not have any report flag up.

The report is then checked by the community-admin and hence the reputation changes depending on the approval or rejection.

10.3   Implementation of Badge System
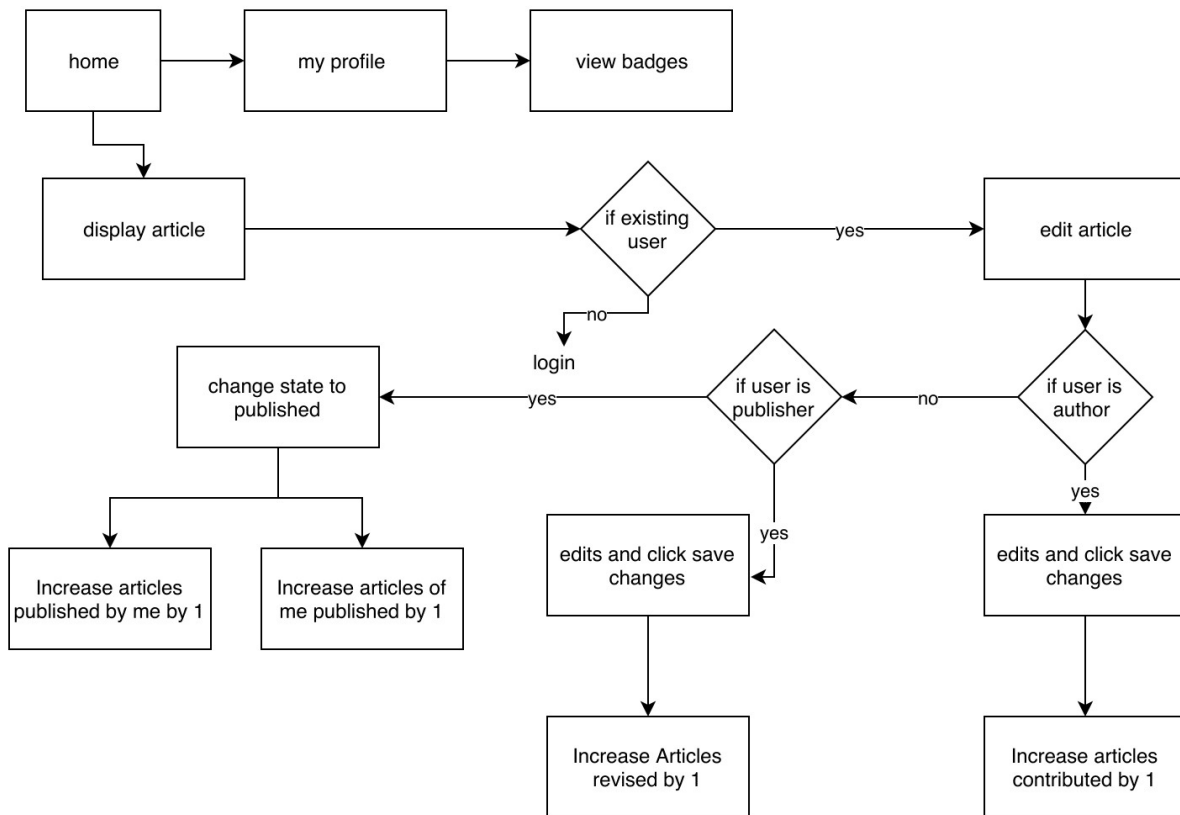
The badge system has been designed carefully where every activity like editing the article is recognized depending upon the count. The badge system has been implemented for 4 systems i. e. Article contribution , Article Published , Publisher publishing the article and Publisher editing the publishable article.

The further subcases is used to distinguish the levels of contributions, a user has, in each of the specified field and based on the numbers , the user can see the badges earned on his dashboard. The badge system is always updated as soon the count is increased. The badge system is implemented by using another model in Voting app where the Integer fields have been used to store the number of count of each field and then showing badge according to the if-else conditions which is in views. py.

10.4   Use Case Diagram
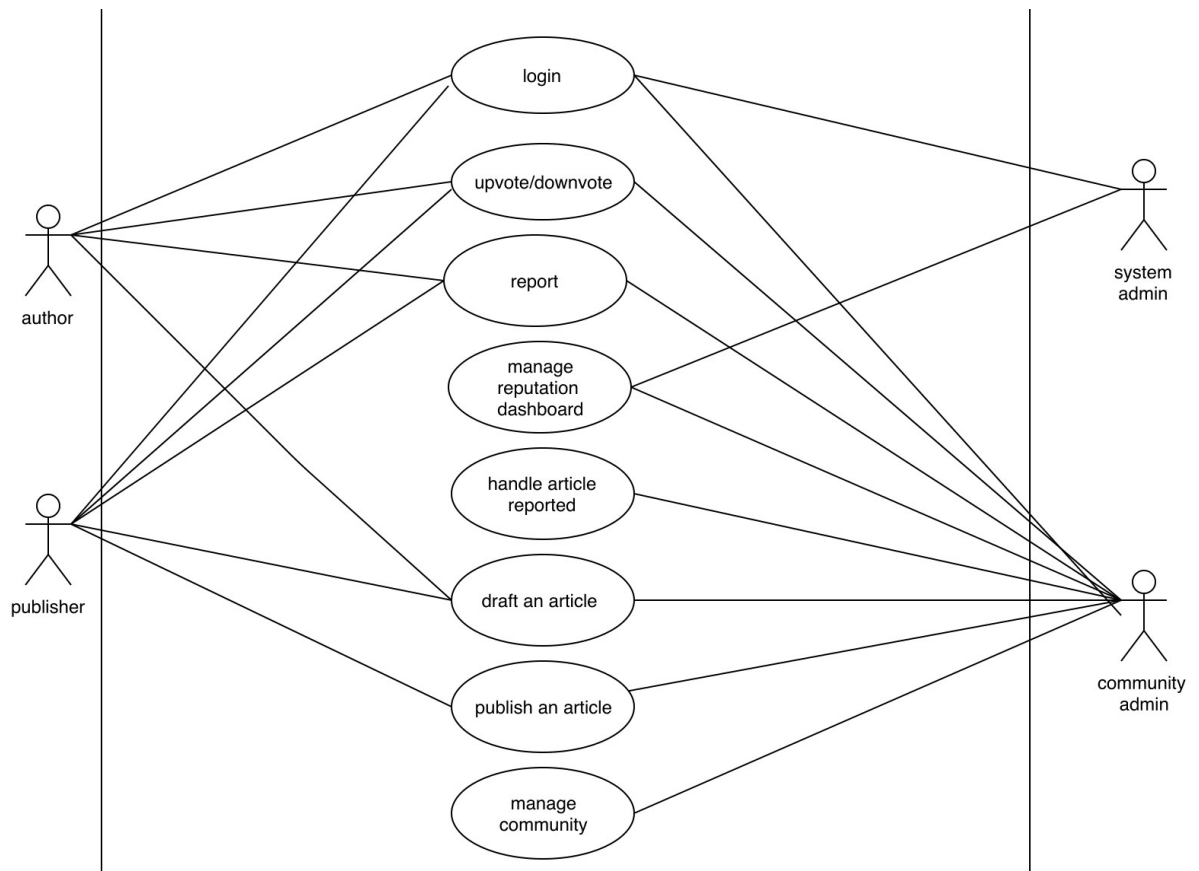
The roles of each and every user domain is specified so that if user knows the responsibilities for each and every role. It aims at providing 100% transparency.
The main reason of publicly showcasing the roles is to gain the trust of the users within the community. Every role would encourage the user to do more and trust Collaborative community as the main focus of Reputation system is both Transparency and no flaws in the system and we have tried to gain both using Grading system and Use case Diagram.

	Login	Upvote/Downvote	Report	Manage reputation Default Values	Handles Article reported flags	Draft/Edit Article	Publish Article	Manage Community
Author	✓	✓	✓			✓		
Publisher	✓	✓	✓			✓	✓	
System Admin	✓			✓				
Community Admin	✓	✓	✓	✓	✓	✓	✓	✓

11   Badge Designs

The badge system and designs follows the table in the given order.

Activity	Badges				
	1 to 5	**6 to 15**	**16 to 30**	**31 to 50**	**Above 50**
	Author / Publisher				
Articles Contributed	AC	AC	AC	AC	AC
My Articles Published	AP	AP	AP	AP	AP
	Publisher				
Articles Revised by me	RA	RA	RA	RA	RA
Articles Published by me	PA	PA	PA	PA	PA

Respective Designs are :
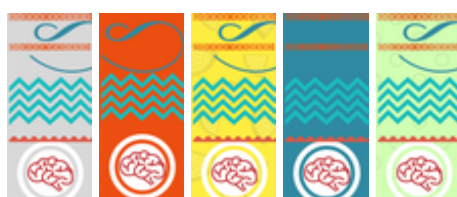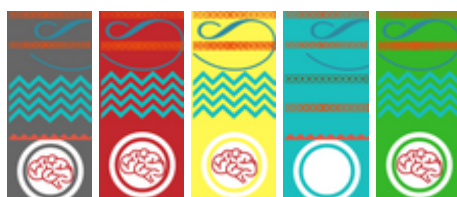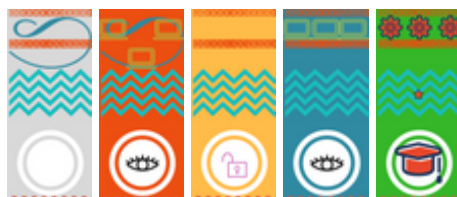
12   Testing

13 Result

14   What is different in our system?

Every reputation system aims at achieving 100% transparency with no break or loophole in the system. The stack overflow[1] has achieved the rule of no loophole as in any circumstance the system can detect the activity of the user and can rate the activity as negative by decreasing the reputation or by increasing it if any good activity takes place.

Our system implementation has too achieved the level of no loophole as in any circumstance , it can handle it by discarding or taking action against it , which ensures the system to be consistent and is not affected by the wrong activity of any one user or group of users. But our system has reached a decent transparency level too where the grading system and each and every possible activity is listed which cannot be broken by the user. The user has no choice left but to follow the rules because if he does not , then the system will itself take care of it by decreasing the reputation,thus affecting the user. This too ensures that any user who joins the community abide by the rules and regulations and can view any information about the system which he might require when generally his domain changes or something goes wrong with his reputation.

The main feature that we have presented is that transparency and no loophole both have achieved successfully with the help of the research model and the way of implementation.

The second most important point is that reputation has been integrated with all the other systems like recommendation and event logging which aims at a very important aspect of what to show the user. The stack overflow too has implemented it on a very large scale and with a huge user database but we too have achieved the same level implementation with as of now less users. The reputation system has weighted on peer to peer system and user interaction.

It is very important to understand that why any user would join our system. The system has different systems which attracts the user in different fields. Any student who has questions and has a low reputation hesitates in questioning in Stack Overflow or any such community but in our system , the user is encouraged to post any article and the reputation is independent of the number of users in the system which ultimately ensures that a user does not hesitate our system and is not worried about the number of people in the system,rather our system becomes more efficient as the number of users increases.

The problem of increasing number of users is that they might lead to increase in default values of reputation to become publisher so in that case , user has choices left i. e. start thinking of a community which is not there and create it and the users would gain roles later and user will do initially or the user has other option of contributing more to join communities and draft articles and make it reach published state.

Thus , the system aims to achieve GHP(Gross happiness rate) more as compared to the Gross ArticlePublish rate.

15   Improvements for future model

15.1   Why all contributors who helped in editing should get equal points?

The system consists of various versions of article. Our basic aim to select those versions which actually improvised the article in a better fashion. Let us consider $x_1, x_2, \ldots x_n$ as the number of versions after the version of first draft as x_0.
Consider u_k : upvote of the kth version And d_k : downvote of the kth version
Let the selected versions be denoted by s_i's. U() is upvote and D() is the downvote.
If the factor of $U(x_i - s_{i-1})/D(s_{i-1} - x_i)$ is positive then this means that there is a substantial increase in upvotes or decrease in downvotes. So we select only those versions which have the positive value.
ALGORITHM:
Let the set S denote the selected version of article for point allocation including x_0.
STEP 1 : k=0,j=0
STEP 2 : For i = 1 to n:
STEP 3 : if($U(x_i > S(s_k)$)) if($U(x_i - S(s_k))/D(S(s_k) - x_i) > 0$):
STEP 4 : S = S U x_i
STEP 5 : k=k+1
STEP 6 : else:
STEP 7 : i=i+2
STEP 8 : Endloop

15.1.1   Point Allocation for version of article

Now the ratio of (u_j+d_j):Number of views is compared and seen that if the ratio is close to one then it is a good article else the article is not worth because number of views are way higher than u_j+d_j.
Obviously u_j+d_j can never be < Number of views. If this is true then something is wrong in code. . . .
So according to the tested statistics the allocation is : We maintain a bank where the number of versions selected as s_i's will each have +5 point so bank has 5n points if n are selected.

Case 1: So those articles which have ratio close to one will have 70% of 5n.
Case 2: Publisher will have 20 % of 5n.
Case 3: And rest goes to remaining versions of article.

Now further if there are two or more articles in every case then equal distribution will be followed.

15.1.2   Proof of the above allocation system

The proof describes that the factor of U(xi - sk)/D(sk - xi) is the improvement factor which will always ensure the selection of those articles which have higher peaks leading it to grow to the publishable state.

If the x0 has U(x0), D(x0),then the next selection on the basis of the algorithm ensures that if the factor is positive then either upvotes have increased or downvotes have decreased. So this ensures the improvement in the article.

The proof will be done by using the principle of mathematical induction.

Base: If the draft is x0,the selection of next version is basically U(xi-x0)/D(x0-xi) >0 and U(xi-x0) >0 or D(x0-xi) >0.

This ensures that increase in upvote is there or decrease in downvote is there which indicates that we should select the version i. This selection proves the improvement factor selection algorithm.

Induction : If the previous selected version is sk then,the selection of next version is basically U(xi-sk)/D(sk-xi) >0 and U(xi-sk) >0 or D(sk-xi) >0.

This ensures that increase in upvote is there or decrease in downvote is there which indicates that we should select the version i if the above condition is satisfied thus making it close to the the publishable model. This selection proves the improvement factor selection algorithm.

16 Conclusion

The major problem of how to assign the roles without the help of super-admin has been resolved by the implementation of reputation model considering various thresholds and rules which govern the consistency and proper flow of system. The problems faced by the collaborative community of identify publishers and people with highest reputation ,the concept of system reputation is introduced. The allocation of points to the contributors should not be even which is presented by a possible algorithm with proper proof. The badge system for every unique activity has been resolved. Thus, the reputation model has been a base for not only these problems listed above but has also helped the notification systems and recommendations system where they just need to use the article upvotes and views algorithm(described above) to recommend the content. // The problem of concentration of power in super-admin hands is also solved. The super-admin is the person who controls the whole workflow of the system and concentration of power in one hand is not advisable at any time. If collaborative community has no reputation system , then the biggest problem occurred was that the organized workflow would be there but every activity of the person would be governed by only one person while in automating the reputation system, the peer to peer reputation model has been implemented where one person who is the publisher can help a person who is a author by suggesting the changes which article requires.

The complete and full implementation of reputation system is also handled :
1. Implementation of Upvote and downvote button.
2. Implementation of change in user reputation based on the role and upvote and downvote which needs a research model where all the specifications required for updation of reputation is implemented.
3. The difference between system reputation and community reputation.

The system is automated where the reputation points of system and community helps in deciding the roles which ensures that the system provides a fair chance to all the people in the system and the role can be given based on their time devotion to various activities like editing,drafting article etc. in the system.
Automation reduces the workload of the super-admin and ensures that he just have the role to manage the system and not dominate it by concentration of power in his hands. Basically , power has been spread out to different people across communities to ensure fairness in the system and thus automation helps in distributing the power.
Notification system uses upvote-downvote and reputation implementation to notify users,publishers about events like who became publisher,community-admin etc.

17 References

[1] Dana Movshovitz-Attias, Yair Movshovitz-Attias, Peter Steenkiste, Christos Faloutsos, "Analysis of the Reputation System and User Contributions on a Question Answering Website: StackOverflow", 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining

[2] Documentation of Collaboration Community,GNU Licensed,
https://docs. google. com/document/d/1ENp_UAspLg_c291cjvQxqlCIbnbyjxeIaoxCdl4xqes

[3] Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, Jure Leskovec,"Discovering Value from Community Activity on Focused Question Answering Sites: A Case Study of Stack Overflow", Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, 2012

[4] Andrew Marder, "Stack Overflow Badges and User Behavior: An Econometric Approach", Proceedings of the 12th Working Conference on Mining Software, 2015

[5] Bo Adler,Luca de Alfaro,Ashutosh Kulshreshtha,Ian Pye. Reputation Systems for Open Collaboration Communications of the ACM 2010,Copyright 200

[6] Page Ranking Algorithm,text is available under the Creative Commons Attribution-ShareAlike License,https://en. wikipedia. org/wiki/PageRank

[7] Josang, A. , Ismail, R. (2002, June). The beta reputation system. In Proceedings of the 15th bled electronic commerce conference (Vol. 5, pp. 2502-2511).

[8] Stack overflow reputation and moderation which describes about the various scenarios of voting,User contributions licensed under cc by-sa 3. 0 with attribution required. rev 2018. 6. 11. 30708,https://stackoverflow. com/help/whats-reputation

[9]A Josang, R Ismail - Proceedings of the 15th bled electronic "The beta reputation system", 2002 - domino. fov. uni-mb. si

[10]M Gupta, P Judge, M Ammar - . . . of the 13th international workshop,"A reputation system for peer-to-peer networks",2003- dl. acm. org

[11] C. Boyd A. Jøsang, R. Ismail. A survey of trust and reputation systems for online service provision. Decision Support Systems, 43:618–644, 2007 [12]B. T. Adler, K. Chatterjee, L. de Alfaro, M. Faella, I. Pye, and V. Raman. Assigning trust to Wikipedia content. InProc. of WikiSym 08: International Symposium on Wikis. ACM Press, 2008