

# State Of The Art Crowdsourcing Applications <sup>\*</sup>

Advanced Internet Computing - Wintersemester 2013/14 <sup>†</sup>

Simon Zuend<sup>‡</sup>  
1025990  
e1025990@student  
.tuwien.ac.at

Martin Keiblinger<sup>§</sup>  
0825118  
e0825118@student  
.tuwien.ac.at

Klaus Nigsch<sup>¶</sup>  
1025991  
e1025991@student  
.tuwien.ac.at

## ABSTRACT

With Crowdsourcing transforming to the web, a new field rises. [1] Still, the prime examples of the Crowdsourcing (CS) paradigm are Wikipedia, Linux, Yahoo! Answers and Mechanical Turk-based systems.

Crowdsourcing Systems are on the rise, yet designing Crowdsourcing systems turns out to be surprisingly tricky [1].

This paper will describe what Crowdsourcing actually means and how the term is to be understood. But this is just one part. The paper will also give some insight on what the state of the art on Crowdsourcing Systems is and what we have to be prepared for, to deliver successful Crowdsourcing Platforms in the future.

## General Terms

### Terms

Crowdsourcing - Instead of using lots of computational power to solve complex tasks, let a multitude of humans solve these (for humans often easy) problems.

## 1. INTRODUCTION

Crowdsourcing is a modern approach of solving tasks of different complexity. This paradigm is successful because it is surprisingly computationally easy, but requires a crowd of human workers instead. While not being entirely trivial to design, they are (seemingly) easier to implement than complex algorithms and services.

The tricky part is to design a good crowdsourcing platform instead of just 'a' crowdsourcing platform. This paper gives some overview about the state of the art of crowdsourcing

systems and what challenges we are currently facing. What the reader will probably find interesting is the section about the future of crowdsourcing systems, since crowdsourcing is a pretty new paradigm and we do not know yet what we have to be prepared for.

As a note that we as team want to leave here is: We personally think, that the crowdsourcing paradigm has a bright future, if we react to the changes the future brings and adopt to them. What we must not do is take what we have currently as the ultimate crowdsourcing solution and take the paradigm as completely evolved. As this paper will describe: There are many key challenges that we should be prepared for and we have to be able to react dynamically and actually build our systems in a way to adopt to these changes. We believe that this insight is crucial.

## 2. STATE OF THE ART

In this section we describe the state of the art of crowdsourcing in general. We do not only want to show some modern example platforms, but also more general address the challenges modern crowdsourcing platforms are currently facing.

### 2.1 Challenges

According to [3] there are 4 main challenges in Crowdsourcing at Finpro, a foresight company. So, at first glance the reader might ask why we are talking about such a specific application of Crowdsourcing when discussing the general challenges of crowdsourcing. The reason is simple: We believe, these main challenges also apply to crowdsourcing in general.

In our eyes they are very well thought out and applied and reinterpreted to crowdsourcing in general, these 4 very basic challenges are present in Crowdsourcing platforms nowadays.

#### *Motivation.*

Motivation is the main challenge in Crowdsourcing nowadays. Crowdsourcing must be rewarding and appreciated. If platforms are not designed to motivate workers, resistance or rejection of the workers might happen. Crowdsourcing platforms should be built to be prepared for those reactions and should have countermeasures to react to such events.

#### *User Interface Simplicity.*

It might be obvious, but it is fundamental: If the platform is too complex, not easy or simply not attractive, no-

<sup>\*</sup>Tight Template by ACM

<sup>†</sup>Topic 2: Crowdsourcing

<sup>‡</sup>He is a mastermind.

<sup>§</sup>He did tons of work.

<sup>¶</sup>This author is the one who did all the really hard work.

body will use it. Much time should be investigated to design a simple and attractive user interface. The crowdsourcing paradigm really has to evolve in user interface simplicity. As you can imagine: There is a crowd of workers, and they will use the platform more than just once, so you basically want to make it as good as possible, not only to minimize your own costs in the end.

### *Training.*

In order to obtain good results, you want good workers. Training them might come in handy. Train them to understand the type of tasks they have to solve and encourage them to use the features your platform provides. This especially applies if your tasks are not trivial, but complex and time consuming.

This is also related to tasks that require special experts. You can read more about that in the section about the crowdsourcing future, in specific the challenges we will be facing in the future.

### *Feedback.*

The last important challenge is to provide feedback. Payment should not be the only reward for workers. Feedback is important. Let them know about the quality of their work or they probably will not be doing it for long.

One might argue this goes hand in hand with motivation, but we believe it is worth mentioning separate. Your platform has to be designed to provide feedback, and this is not a trivial task, especially when facing more and more complex tasks, because then measuring quality is also getting more difficult. This might lead to the need of supporting multiple feedback criteria in a crowdsourcing platform.

## **2.2 Example Platforms**

To analyse the state of the art of Crowdsourcing, a good start is looking at some existing Crowdsourcing systems. There is a wide variety of them, also aiming in totally different directions.

A very straight forward example is <http://crowdflower.com/>. They advertise with: 'There's No Limit to What CrowdFlower Can Do - Just Ask Our Customers'. CrowdFlower provide a platform to submit any work, which will be tackled by 5 million skilled contributors from all around the world. According to them, there are no limitations to the jobs, from simple to the most complex jobs.

Experts will help you in setting up your project, help you with the pricing and finding a suitable workflow for your project.

Another interesting example is <http://www.freelancer.com/contests/>. If you want something implemented, request your idea as a contest. You also set a reward / a price for freelancers (basically developers) to complete this task. And here exactly the idea of crowdsourcing applies: Human workers will solve the task for you, in different quality of course. You can then choose from the solutions and award the best entry, provide feedback to the developers and finally starting a handover process to legally own that entry.

This already shows the possibilities of Crowdsourcing. It is by no means limited to small tasks or a small mass of human workers. Everyone can register as freelancer and participate in the contest, and the task also does not have to be trivial for humans, but can also be complex.

To sum the examples up, some crowdsourcing approach

that uses experts only is <http://www.trada.com/>. This site offers a service to customers in the terms of: 'You pay us, and we will assign experts that you can lean on, while maintaining full control of your campaign, but gaining improved performance'. What this platform internally does: It has a crowd of experts and for each incoming customer campaign, they assign a set of those experts that is suited best for the task. So this system has a strong emphasize on choosing the best set of workers for a given problem, generally spoken. This example platform was chosen to point out that it is not that simple to deploy a crowdsourcing platform, by simple assigning a ton of workers to a given job and letting them solve it. If your crowd consists of experts, you especially want to make sure to assign the correct people for a given work to minimize your costs.

## **3. FUTURE**

What can we expect from future Crowdsourcing platforms?

[2] predicts an unsure future for this paradigm. We cannot foresee what will happen in these dynamics of the internet. However, they have compiled a list of key challenges to be addressed, given Crowdsourcing evolves. It is believed that it won't be simple, deskilled tasks to be solved any longer. The focus is to be prepared for complex crowd source / crowd work processes. The key challenges are:

### *Workflow.*

We can no longer assume that a simple, parallel approach will be enough for the crowdsourcing future. Complex tasks can be seen as small projects: They have requirements (that are dynamic), require experts from different fields and have all sorts of dependencies. This is why it will be extra important to be able to divide tasks into subtasks and reassemble results. We have to be prepared for this completely new design space of crowdsourcing workflows.

### *Task Assignment.*

We have to be very cautious about designing a system that assigns tasks in a certain way. Coordination is needed when sharing resources come into play. Tasks might have deadlines, and the pool of workers might be fixed. Different problems arise, for example how to design task queues. Traditionally, workers sort tasks by volume and recency. We should be open for new assignment algorithms. An upcoming technique is an algorithm that dynamically forms teams based on expertise.

Also keep in mind, that the requirements about the assignments will be tougher in future. Of course, Customers want their tasks to be solved very quickly. But also the workers want to be continuously fed with tasks that match their expertise/interests.

### *Hierarchy.*

Hierarchies are a system well known from traditional organizations and companies in general. Hierarchies enable coordination, quality control, decision making, and incorporate sanctions and incentives.

We haven't taken hierarchies in consideration for crowdsourcing so far, but having a hierarchy of workers could make it possible to enable a whole new level of tasks in terms of complexity. It might allow to design completely new workflows, and enable to form teams instead of individual work-

ers. Again, it cannot be foreseen how this will turn out.

### *Realtime Crowd Work.*

As already mentioned in Task Assignment, task requirements will get tighter. Work with very short completion time will definitely be a topic. An idea to tackle this problem could be 'flash crowds'. Individuals that arrive just moments after a request has arrived, working synchronously for this task. Currently, crowd tasks can take hours and days, simply because short completion time tasks are tackled too late and coordination is not well enough yet.

### *Synchronous Collaboration.*

So far, Crowdsourcing is still laid out to be seen as work that is tackled by a mass of individuals. We cannot expect that to be the solution forever. Cooperation is key for complex tasks, and is crucial with tasks that have shorter timescales. We also should not forget that there will be cultural and socioeconomic gaps that should be coordinated, and this is by no means a trivial task. Coordination cannot be simply integrated in today's crowdsourcing systems, because they are not designed for this yet.

### *Quality Control.*

Current Quality control systems in crowd sourcing are not well enough designed. They can be fooled, and workers see it as an issue rather than a feature. For them, it's like a penalty system. Since in future everything is foreseen to be fast paced and complex, quality control has to be taken care of, or the quality of the work might sink very fast. When workers are forced to deliver high throughput and solve complex tasks, they tend to drop quality. How a future Quality Control system may be designed is hard to predict, but we shouldn't take quality control as given and invest a lot of research work into this topic.

## **4. REFERENCES**

- [1] R. R. Anhai Doan and A. Y. Halevy. Crowdsourcing systems on the world-wide web.  
<http://delivery.acm.org/10.1145/1930000/1924442/p86-doan.pdf>.
- [2] M. S. B. E. M. G. A. S. J. Z. M. L. Aniket Kittur, Jeffrey V. Nickerson and J. J. Horton. The future of crowd work.  
<http://hci.stanford.edu/publications/2013/CrowdWork/futureofcrowdwork-csw2013.pdf>.
- [3] E. Hiltunen. Crowdsourcing the future: The foresight process at finpro. [www.jfs.tku.edu.tw/16-1/R02.pdf](http://www.jfs.tku.edu.tw/16-1/R02.pdf).