JYRI VIITALA
OPEN BUSINESS – A METHOD FOR CROWDCREATING NEW
PRODUCTS AND SERVICES

Master of Science Thesis

ABSTRACT

This thesis presents a method to crowdcreate new products with an ideology called open business method that derives from open source movement. In addition to designing products, this thesis suggests that the model could be applied to services and physical manufacturing also. Key feature is the remuneration method, how a crowd can be compensated for their work in a flexible way. The way it works is that instead of getting salary, the net revenue is divided among the participants based on the worked hours. This requires splitting the work into tasks. And the amount of working hours that each task take, is estimated by the crowd. According to a concept called wisdom of crowds, this estimation would be very accurate.

Another key feature is how to capture value from crowdcreated products. Open source products are usually free. This thesis analyses how traditional business models, such as patenting can be used on an open, crowdcreation platform. Main issue with patenting is the disclosure of the innovation. In Europe this would prevent patenting and therefore an innovation should be revealed first only to a small review committee and later for the rest of the community. In some countries, such as USA, a grace period is applied and it allows the disclosure of an innovation before patenting, which is more friendly system for a crowdcreation platform. Not all products need patenting and for some it is not even possible. Advantages of a crowdcreated products against closed company counterparts is fast product development, efficient marketing through the crowd and the possibility to include the clients tightly to the product development.

In addition to presenting the open business method, this thesis explains the background theory of open movements, crowdsourcing and wisdom of crowd. Open source is a way to crowdcreate software, open hardware is the design of circuit boards according to open source ideology and open design is the same in other physical products. These concepts, as well as wisdom of crowds, are used as a basis for deriving the governance model, decision-making method and work breakdown structure presented in this open business model. The model requires work to be broken into smaller tasks. Mechanical Turk serves as a good example of how work breakdown can be taken into extreme. Other case studies include examples of crowdsourcing companies, such as Innocentive for technical challenges, Quirky for crowdsourcing new product ideas, LocalMotors for crowdcreated cars and Assembly, a software development platform that utilizes the compensation method of sharing revenue based on tasks.

TIIVISTELMÄ

Tässä työssä esitetään keino uusien tuotteiden suunnitteluun avoimesta lähdekoodista johdetulla menetelmällä. Menetelmää kutsutaan nimellä avoin liiketoimintamenetelmä. Tuotteiden suunnittelun lisäksi menetelmää voidaan käyttää palvelujen tuottamiseen ja tuotteiden valmistukseen. Pääosassa on palkitsemismenetelmä. Palkan sijasta yrityksen nettotuotto jaetaan osallistujien kesken tehtyjen työtuntien perusteella. Tämä vaatii työn jakamista pienempiin osiin. Jokaiseen tehtävään kulunut työaika määritetään joukkoarvioinnilla. Joukkoarviointi perustuu konseptiin nimeltä joukkojen viisaus ja on tämän mukaan menetelmänä tarkka.

Toinen pääseikka on liiketoimintamalli. Avoimen lähdekoodiin perustuvat tuotteet ovat yleensä ilmaisia. Tässä työssä tutkitaan kuinka perinteisiä liiketoimintamalleja, kuten patentointia, voidaan soveltaa joukkoluontiyrityksessä. Suurin ongelma patentoinnissa on idean julkistaminen. Euroopan patenttilakien mukaan julkistaminen estäisi patentoinnin. Sen vuoksi lupaava idea pitää joukkoluontisivustolla esitellä ensin pienelle arviointiryhmälle ja vasta jälkeenpäin muille. Joissain maissa, kuten Amerikassa, on käytössä armonaika, joka mahdollistaa idean julkistamisen ennen patentointia. Tämä malli soveltuu paremmin joukkoluontiyritykselle. Kaikkia tuotteita ei kannata, eikä edes voi patentoida. Joukkotyöskentelyn etuja tällaisissa tapauksissa suljettuihin yrityksiin verrattuna ovat nopeampi tuotekehitysaika ja tehokas markkinointi suuren joukon avulla sekä mahdollisuus saattaa asiakkaat tiivisti mukaan tuotekehitystyöhön.

Avoimen liiketoimintamenetelmän lisäksi tässä työssä esitellään taustaa avoimista liikkeistä, joukkohankinnasta sekä joukkojen viisaudesta. Avoin lähdekoodi on menetelmä luoda ohjelmistoja suuren joukon voimin. Avoin laitteisto-ideologialla tuotetaan piirilevyjä samalla periaatteella ja avoin suunnittelu-menetelmällä muita fyysisiä tuotteita. Nämä konseptit, sekä joukkojen viisaus-ideologia, toimivat pohjana tässä työssä käytetyille päätöksenteko-, hallinnointimenetelmille. Palkitsemismenetelmä vaatii työn jakamisen pienempiin osiin. Mechanical Turk on hyvä esimerkki kuinka työnjako voidaan viedä äärimmäisen pitkälle. Muita malliesimerkkejä tässä työssä ovat joukkohankintasivustot, kuten Innocentive, jossa joukkohankitaan ratkaisuja teknisiin ongelmiin, Quirky, jossa joukkohankitaan uusia tuoteideoita, LocalMotors, joka valmistaa autoja joukkovoimin ja Assembly, joka käyttää samantapaista tuotonjakoon perustuvaa palkitsemismallia kuin tässä työssä esitetään.

PREFACE

Open movements and crowdsourcing initiatives are special interests of mine, which is why I chose this topic is for my thesis. I also feel that it is the direction that work life will be heading in the future. I want to thank my former employers for giving me the opportunity to experience work life through multiple companies. I feel that it was these numerous experiences in different tasks that eventually lead me to come up with this topic. I also want to thank my examiner Jouni Mattila for the constructive feedback and good advices about the structure of the thesis. I also want to thank my friends and family for their support during the work.

Vantaa, June 5th, 2015

Jyri Viitala

TABLE OF CONTENTS

TERMS AND DEFINITIONS

CNC	Computerized numerical control, a computer machining technique
PMC	Project management committees in Apache community
GNU	Free computer operating system
GPL	General public license
CERN	European Organization for Nuclear Research
OHL	Open hardware license
TAPR	Tucson Amateur Packet Radio, amateur radio organization
FreeBSD	Computer operating system
MIT	Massachusetts Institute of Technology
Cc	Creative Commons
Nd	No derivatives, creative commons license extension
Sa	Share alike, creative commons license extension
Nc	No commercial, creative commons license extension

1. INTRODUCTION

Open source movement has been around for decades producing software with the help of hundreds of volunteers. It has been suggested that same method could be applied to physical products. Instead of companies creating products, people would, via internet. Such initiatives already exist, open hardware movement designs circuit boards in the spirit of open source. Open design movement designs physical products in the same way. This thesis introduces an open business method. A method how open source ideology could be applied to all business, designing products and services as well as manufacturing. And a method how participants could share revenue and get compensated for their work is discussed. As well as business models to capture revenue with the products. Drawbacks of traditional hierarchical management model is presented and the possibility for a crowd to self-organize itself is studied and a method how it could happen in practice is presented.

This thesis presents studies how a crowd can be smarter than individual experts and produce better decisions using a concept called wisdom of crowds. And why a company should be managed by a crowd instead of a CEO or a board of directors. And how a product design process could be broken into small pieces and performed in a fraction of the time it currently takes companies to finish a project.

Startups today face tremendous challenges in bringing new products to market. And the success rate for new products is very low. These efforts could be dramatically reduced by sharing the work load and the risks, leading to a higher success rate and better products. By crowdcreating products, instead of individual startups, entrepreneurs could exploit the help of industry professionals around the world. Instead of an entrepreneur handling everything by oneself, an expert of each specific field would execute each task.

Currently, companies' design data and design processes are closed from the public. This thesis points out the tremendous learning potential that is tight up to this closed information. By opening up data and development conversations to the public, an open business platform could provide learning material for students, and a method for new designers to gain the same industry experience that it now takes years to gather, in a fraction of time.

1.1 Methodology and scope

Literature review presents research results and industry facts of related fields, laying out a theoretical framework, which the ideas presented in this thesis rely on. This background information includes open movements, wisdom of crowds and problems with existing

working practices. This thesis introduces a theoretical model for crowdcreating products, called an open business method, and tools how this could be implemented. The feasibility of the model is analyzed by examining case studies of existing companies that operate partly in a similar manner. These include open source initiatives, open design companies and companies exploiting crowdsourcing.

1.2 Thesis structure

Chapter 2 – Theoretical background introduces the key terms and definitions for this work, as well as existing theory and research of related fields. Case studies introduces existing companies of related fields and analyzes the feasibility of the introduced model based on these companies.

Chapter 3 – Open business method introduces the new ideas and methods presented in this thesis for crowdcreating products. Finally, in chapter 4 and 5 the consequences as well as benefits and drawbacks of this model are analyzed.

2. THEORETICAL BACKGROUND

This chapter presents background studies about the topics related to the open business model. The open business model contains many hypothesizes and many of them can be verified by looking at previous studies and by studying example companies. These research results will then be used in chapter 3 to develop the model in more details.

This chapter also addresses the problems that current companies have, that open business model could solve. There is a clear need in the market for this kind of method to conduct business collaboratively and this need can be identified by looking at the problems in the way products are created currently.

2.1 Traditional start-ups and established companies

Startups and established businesses face partly same and partly different problems. Main difference is that startups are small, therefore lacking many resources, and their business idea is new and therefore they operate under high uncertainty. Established companies have a working business model, but they face problems due to the size of the organization and changes in the market. Sometimes established companies can be thought of as startups also. If they create a new product that is not in the market yet, this effort can be considered as a start-up (Ries, 2011).

2.1.1 Start-ups

Start-ups are typically founded by one or a few founders who do everything themselves. Startup founders have to know a little bit of everything, since the team is small. Startup founders have to also be committed and ready the make long days for their company. An angel investor Mark Cuban has said "Don't start a company unless it is an obsession and something you love" (Cuban, 2012). While that is a good advice for startup founders, it also tells a lot about the field. The need to be obsessed limits significantly who can become an entrepreneur. Many people would have good ideas, but not that many have the courage, stress handling ability, required skills, time and money to start their own business. Currently these people cannot start a business and good ideas and skills go wasted. Creating new business should not be that hard. It should, and it could, be made easier. It should be made so easy that even people who are not committed or obsessed, and only have little time per week to spend on the company, could contribute in creating new products.

According to a study 90% of start-ups fail (Griffith, 2014). This figure alone tells that something is wrong in the way that startups are built. Main reason for failure (42%) is reported to be the lack of market need for the product. This is easy to believe since it is

awfully easy to concentrate just to build a product without actually reaching out to the customers. Internet is full of stories of entrepreneurs who have built a product for a year, then come out with it, just to realize that no one needs it. One example of such is IMVU, 3D social network application. In his book Lean Startup, cofounder Eric Ries describes how they worked for six months to create the product, just to find out that no one wants to buy it. It turned out that they had built it on totally wrong assumptions about customer needs. IMVU did become successful eventually, but the point is that huge amount of entrepreneurs do the same mistake. (Ries, 2011).

Second biggest mistake (29%) is that startups run out of money. One thing that a startup needs money for is hiring personnel. They need talented people for their core competencies and talented people cost a lot. And hiring cheap workforce might lead to other problems. Third biggest reason (23%) is that the founding team is wrong. It might lack the talents needed or not have the experience, vision or enthusiasm needed. Founding team is also the top thing that investors look in a company. "Angels invest in people and ideas, not valuation models. Investors must perceive the management team as capable of delivering on the company's vision and goals" (Fallon, 2014).

Other reasons include such as poor product, get outcompeted, lack of business model, poor marketing strategy. All which root back to the founding team's skills and proficiency.

How could these problems be addressed? A better way to found a startup would be to found one together, collaboratively. The problem of creating unwanted products could be addressed by including the customers to the design process. The customers could act as normal workers of the company giving their input and getting the same compensation for their work as anyone else. Or a customer could act as an investor. If big part of the company's investors are customers, it must mean that the company is creating a product that customers believe in and need.

Second problem is running out of money. This can be because the founders are working in an inefficient manner, or the investors don't believe in the idea or the team. Investing in a startup is a big risk for an investor, since so many of them fail. But workers can be found without money too. This thesis presents a flexible compensation method for a crowd of volunteers. And if a group of hundreds of workers actively works on a project, including future customers, it should be a lot easier for an investor to believe in the idea and invest. The third problem is unskillful founding team. On a crowdcreation company, this would not be a problem, assuming that the crowd is big enough to possess all required skills and experience needed. And that a method for the crowd to work together can be established.

Another key issue is that currently the experience from existing startups is not finding its way to the ears of new startup founders, making them repeat the old mistakes again. On

an open platform, where all the information is visible to everyone, mistakes would be spotted quicker and best practices would spread faster than in traditional closed companies, since workers would work on multiple start-ups at the same time.

2.1.2 Established companies

Some of the main problems in established companies are related to their size, hierarchical organization, innovation ability and decision making processes. When people work is groups, problems arise that do not occur when working alone. Some of these drawbacks of traditional group meetings are listed below.

Conformity. "Many people go along with the group regardless of what they themselves might think as individuals. A famous experiment by psychologist Solomon Asch showed how powerful this effect is. Asked to choose which of three lines was the same length as a prototype line, nearly every subject chose correctly when acting alone. But then Asch put each subject into a group of several confederates, all of whom had been instructed to pick the wrong line on one of the "tests." Sure enough, almost 75% of the subjects agreed with the group at least once—even though many later confessed they knew the group's answer was wrong (see Figure 1)." (Brief, 2013)

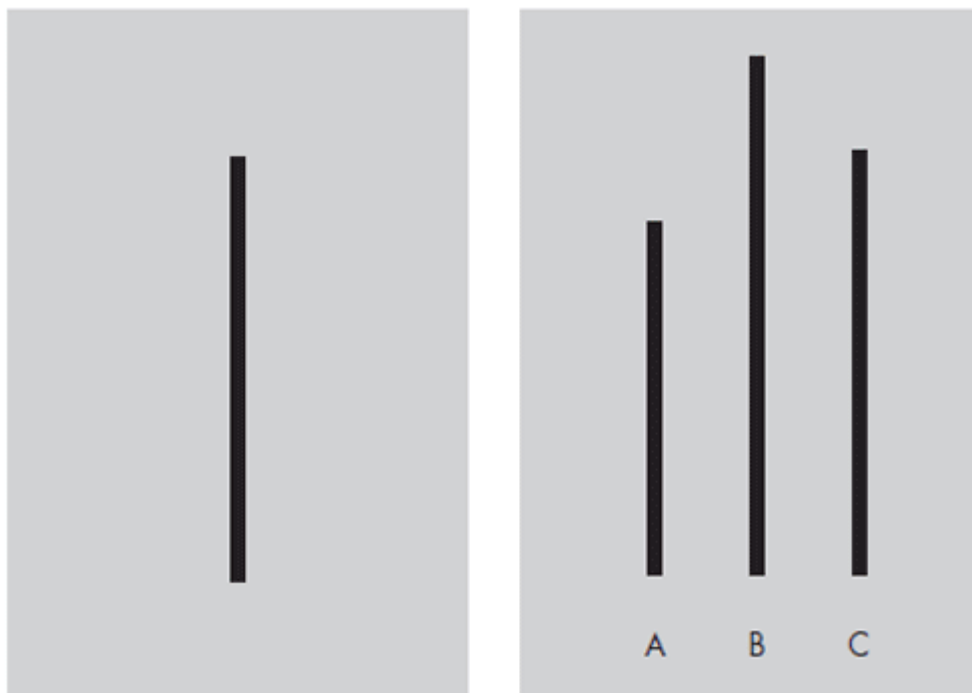*Figure 1 Which line matches the one on the left? (Brief, 2013)*

"In business, the tendency to conform often persuades dissenters to shut up rather than speak out. Warner Brothers, for example, invested $50 million in the film adaptation of Tom Wolfe's best seller The Bonfire of the Vanities. The result: a hugely expensive box-office bomb. "Many people involved … had doubts about the casting choices and changes

in the storyline, but they never voiced these doubts to the director," wrote Cabrillo College professor J. Dan Rothwell in a book on small-group communication. Meanwhile the director also had doubts "but because no dissent was voiced, he convinced himself that he had made the correct decisions."" (Brief, 2013)

Group polarization. One could think that a group would tend to moderate individual points of view. But in fact, the opposite often occurs: In a phenomenon known as group polarization, deliberation can intensify people's attitudes, leading to more extreme decisions. A study of US federal judges, for example, found that judges working alone took a relatively extreme course of action only 30% of the time. When they were working in groups of three, this figure more than doubled, to 65%.

In business this could have severe implications. For example a company's investment committee that is composed of people with a generally cautious outlook, could as a group make decisions that avoid risk altogether—or vice versa. Or in case of a go/no-go product-development decision. If group members making the decision are inclined toward innovation rather than conservatism, they may collectively decide to throw caution to the winds. (Brief, 2013)

Obedience to authority. Ordinary people are likely to follow orders, even if they have bad consequences, which Stanley Milgram's classic psychology experiment from year 1963 proves. In this test subjects obeyed instructions to administer electric shocks to other "subjects", (who were actually confederates pretending to be shocked), even when the harm to the other person seemed extreme (McLeod, 2007). Traditional businesses are built on employees carrying out their supervisors' instructions, and this can sometimes harm the business. As an example the worst commercial aviation incident in history: In 1977 a KLM plane attempted to take off from Tenerife airport while a Pan Am plane was on the runway. Official investigation concluded that the senior KLM pilot had taken off without clearance as a result of communication problems, including the reluctance of other crew to challenge his decision to go. The "captain was always right" effect was cited as a principal cause in the official report on the incident. (Brief, 2013)

This kind of obedience can have bad impact in technical development discussions, if the employee feels reluctant to disagree with the boss. Development discussions should be free for new ideas and thoughts as well as disagreements, but it can never be that if the participants are not equal. It leads to a situation where, not the best solution is chosen, but the one that the boss thinks is the best.

Groupthink. "One of the greatest inhibitors of effective group decision making is groupthink. Groupthink is a psychological phenomenon that occurs within a group of people in which the desire for harmony or conformity results in an irrational or dysfunctional decision-making outcome. By isolating themselves from outside influences and actively suppressing dissenting viewpoints in the interest of minimizing conflict, group members

reach a consensus decision without critical evaluation of alternative viewpoints." (Boundless, 2015)

Hierarchical organization. The main disadvantage of hierarchical organization is that it always comes up to one person to make the final go/no go decision of new ideas. This requires that person to be very knowledgeable of all the things that workers might suggest, which is not likely. If a worker happens to have a lot of experience of some specific field, and therefore comes up with a good improvement suggestion, the boss probably does not possess the same experience and therefore might not understand the value of the suggestion and would not approve it. This leads to the product being only as good as the boss is and a lot of the employees' expertise and knowledge is unexploited. Another problem with hierarchical management is that workers concentrate working on what they are told, rather than what interests them the most. A person can be assumed to perform a lot better when doing something that he is interested in and does out of own will rather than what is told to.

Internal versus external innovations. In his book Open Innovation: The New Imperative for Creating and Profiting from Technology, Henry Chesbrough points out the problems of traditional internal product development. Companies create innovations relying on their internal work force. This model use to work in the past but not in today's fast moving world anymore. The problem is that a company misses many innovations developed outside the company. And at the same time companies often come up with innovations that they cannot exploit themselves, but someone else could. A clear need to open up company barriers to outside world exists.

The examples above represent the problems faced in hierarchically managed traditional closed company. This thesis suggests that these problems can be addressed by three principles: openness, lack of leadership and freedom to execute ones ideas. Once everyone concentrates on what they think is wise, the decisions are decentralized and risks are smaller. Lack of leadership provides a way to exploit each individual's wisdom and expertise and openness offers a way for everyone to participate in decision making.

2.2 Terms and definitions

The following chapters explain the main terms used in this thesis. Most of them are related to open movements and crowd collaboration. Many of the terms are relatively new and do not have a clear established meaning, but are understood differently in different sources. A clear definition is provided under each term explaining how each term is used in this thesis.

2.2.1 Open business

The term Open Business Model was originally coined by Professor Henry Chesbrough, author of Open Business Models: How to Thrive in the New Innovation Landscape (Sundelin, 2009). Chesbrough's definition of open business model is that companies should include the acquisition of external innovation to their business model as well as the selling of unused internal innovations. Chesbrough mainly writes about innovations, and by open business models he means business models that support innovation exchange.

This thesis defines open business a bit differently. I this thesis open business means an endeavor to create products and services and business in collaboration with a big crowd. It contains three main features:

- Open participation. Anyone can participate. There can be restrictions who can do a certain task, but main principle is that projects are publicly open.
- Shared profits. Net revenue is divided among the participants based on the hours worked. Task difficulty, person's expertise, etc. can also affect the share.
- Self-managing crowd. The work is managed by the crowd, exploiting the concept of wisdom of the crowd. It can be a voting system or another type of mechanism. It can include only certain type of participants for a certain decision, but the main principle is that anyone can execute their ideas and there is no one single manager deciding the future of the business.

While the word open is often associated with freely accessible software in open source movement or free design files of physical products in open design movement, the model in this thesis does not specifically suggest that designs should be free to use by anyone or that they could not be patented. And while other open movements concentrate only on the design phase of product development, this thesis suggests that the model can be used in services and physical manufacturing as well.

2.2.2 Crowdsourcing

The term crowdsourcing was first coined by Jeff Howe, an editor of the Wired magazine in his article "The Rise of Crowdsourcing." Crowdsourcing includes two concepts—outsourcing and a crowd of online laborers— which forms the word crowdsourcing. (Brabham, 2013)

As an example, Innocentive is a company that posts clients technical problems and solicits a solution to those publicly from the crowd. And best answers win a reward. Another example is Threadless, which collects crowd's suggestions for new t-shirt pictures. Both of these are typical examples of crowdsourcing.

However, crowdsourcing is a new term and there is no clear agreement of what it is and what it isn't (Estelles-Arolas, 2012). Brabham defines crowdsourcing as "an online, distributed problem-solving and production model that leverages the collective intelligence of online communities to serve specific organizational goals." Brabham's definition is an example of a strict definition and he, in his book Crowdsourcing, declines many concepts of being crowdsourcing that other sources count as crowdsourcing. Typical examples of argument are historical examples before internet, such as the British government's "Longitude problem", which called for a solution to technical problem that killed sailors, promising £20,000 reward (Lynch, 2015). Wikipedia and Youtube are also sometimes referred as crowdsourcing, while declined by other sources.

Estelles-Arolas and Guevara analyze well the meaning of crowdsourcing in their article Towards an integrated crowdsourcing definition, by comparing definitions of different authors. One unifying concept that they find is that most sources define crowdsourcing to be something that happens via internet. This would rule out the historical examples. Some other requirements they use are the existence of a clear task and a goal, crowdsourcer is clearly identified, recompense received by the crowd is clearly defined as well as the value of the activity to the crowdsourcer.

Different types of crowdsourcing can be identified. Typical model is to ask a question from the crowd and choose the best answer. Innocentive serves as an example of such model. Another model is crowdfunding, where companies are founded by small sums from a big crowd of people. Kick-starter is one company operating in this field (Kickstarter, 2015). Another model is microtasking, where a bigger work is divided into small parts, such as translating a book would be divided into sentences, and then it is outsourced to the crowd. In this model each participant gets compensated. Example of such is Mechanical Turk and Duolingo. (Grier, 2015)

One model serves special attention. In this model you ask the crowd a question, but then, instead of choosing only one answer, you combine the answers together to form the final answer. An example of this model would be guessing how many balls is in a jar. If you ask from a big enough crowd and calculate the average of the answers you will get the final answer, which is very close to the truth. This is a concept called the wisdom of the crowd. A topic that is discussed more in a later chapter.

2.2.3 Crowdcreation

Since the term crowdsourcing requires a separate crowdsourcer to be clearly identified, another term is needed to describe processes where such a requester does not exist, but the process gets initiated by the crowd itself. For that, this thesis defines the term crowdcreation to mean all the processes where a crowd creates something on its own, with no external solicitor or manager. Many open source initiatives fall into this category. Although they often have a single manager, but they don't usually specifically solicit the

crowd for a solution for a certain problem, like happens in crowdsourcing, but the crowd creates new features on its own.

Some other sources, such as (Keifer, 2010) and (ISM, 2015) define the term crowdcreation as one type of crowdsourcing, where a crowd creates something, such as text, logo or a solution, in difference to other types of crowdsourcing, such as crowdvoting for example.

While in both definitions the crowd produces something creative, the difference is who manages the crowd's work and approves the results. In this thesis the term is used in the meaning that the crowd itself manages its work and decides on the working processes. Generally the crowd would also itself exploit the resulting product by selling it, but cases where the solution is ordered by another company are also discussed. The main point is that the crowd would decide on the outcome itself and the external company can only fund the process but not affect the process or reject the results.

2.2.4 Co-creation

Co-creation means creating products in interaction with the customers. The term was first introduced by C.K. Prahalad and Venkat Ramaswamy in their book The Future of Competition: Co-Creating Unique Value with Customers, 2004.

Traditionally companies have created products behind closed doors and the only interaction with customers happen via market researches and in market, when the product is sold. Co-creation brings customer more involved in the product development process and this happens via internet platforms. Companies can exploit the concept in different ways and in different levels. For example Nike collects usage data from the customers with NikePlus running experience platform, in which runners post their running data online and this is then used by the runners themselves, the company and by third parties, such as coaches. And a more extreme example is LocalMotors where customers can design their own product features and even assemble their product on their own. (Venkat Ramaswamy, 2014)

Sometimes co-creation is confused with crowdsourcing. Both include big masses contributing to product development. The difference is that co-creation is a wider concept that includes crowdsourcing. Crowdsourcing can happen without co-creation, but co-creation cannot happen without crowdsourcing. (Adams, 2013)

Co-creation is one of the key element in the model presented in this thesis. Main reason why start-ups to fail is, because they create products that no one needs. If a product would be created by the very same clients that use it, it would surely be a needed product. And open business method offers a deeper method for customers to be involved in product development than a traditional closed company can offer.

2.2.5 Open innovation

The term Open Innovation was introduced by Henry Chesbrough in his book: Open Innovation: The New Imperative for Creating and Profiting from Technology. Chesbrough defines the term as follows: "Open Innovation means that valuable ideas can come from inside or outside the company and can go to market from inside or outside the company as well" (Chesbrough, 2003).

Figure 2 illustrates his idea. The basic idea is that companies should exploit ideas from outside the company resources as well. And also to give out the ideas that don't fit to their own organization, but which some other company might be able to build into working products. This is oppose to the classic way where company generates ideas internally and then develop these ideas into products, also internally.
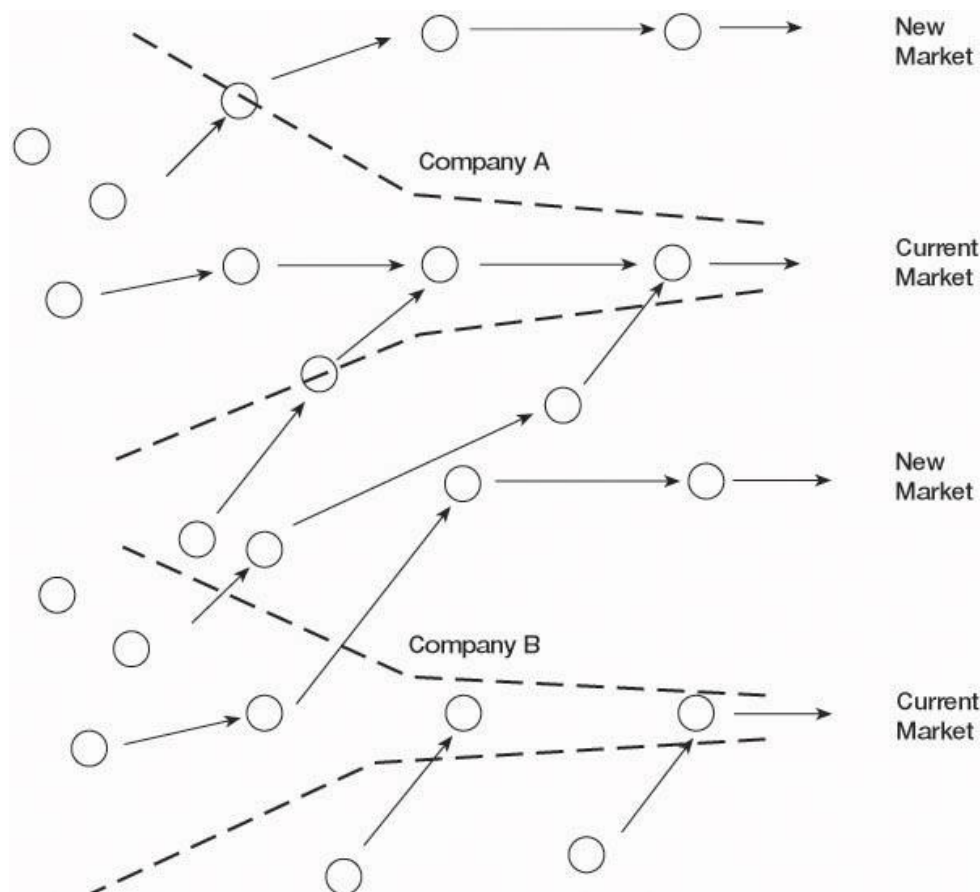

Figure 2 The Knowledge Landscape in the Open Innovation Paradigm (Chesbrough, 2003)

In his book, Chesbrough goes through example companies that has failed to bring to market their own innovations and presents reasons for companies to open up, such as shortened product life cycles and social and company benefits in trading innovations.

Open innovation movement can be seen as a sign of companies changing their working habits and opening company boundaries. Which is a promising sign what comes to the model presented in this thesis.

2.2.6 Open source

Open source is a term used in software development. It means that anyone has access to the source code of a software and anyone can copy or modify it. Since anyone has access to the source code, the software developed in this way are also given away for free and the contributors do not generally get compensated for their work. The movement was originally started by Richard Stallman back in 1985. He coined the term free software. This was later changed by a group of people to open source to better describe the concept. (Gabriel, 2005)

There are multiple open source projects, such as Linux operating system, Apache server software and Joomla content management systems for website creation. And Open Office, a competitor for Microsoft Office. Common to all of these is that they are free to use.

Open Source movement is in special interest of this thesis since it operates according to the model that a crowd creates the product, just like in open business model. Open business model in this thesis originates from the open source ideology, but is used in physical products and with the exception that it captures revenue and compensates workers.

A lot can be learned from open source projects. Especially what comes to management structure and worldwide collaboration. Two types of governance models can be identified in open source projects: centralized control by a single individual or organization, which is called benevolent dictatorship or distributed control awarded in recognition of contributions, which is called meritocracy. Two types of contribution models also exist: cathedral style, in which a small group of dedicated contributors create the product or bazaar-style, in which a larger crowd participates in developing the product. Often projects start as cathedral-style contribution and one person's dictatorship and change into bazaar-style contribution and distributed control, when more dedicated people join the project. Or they can be something from between. Figure 3 illustrates the possible governance and management models. (Ross Gardler, 2013)
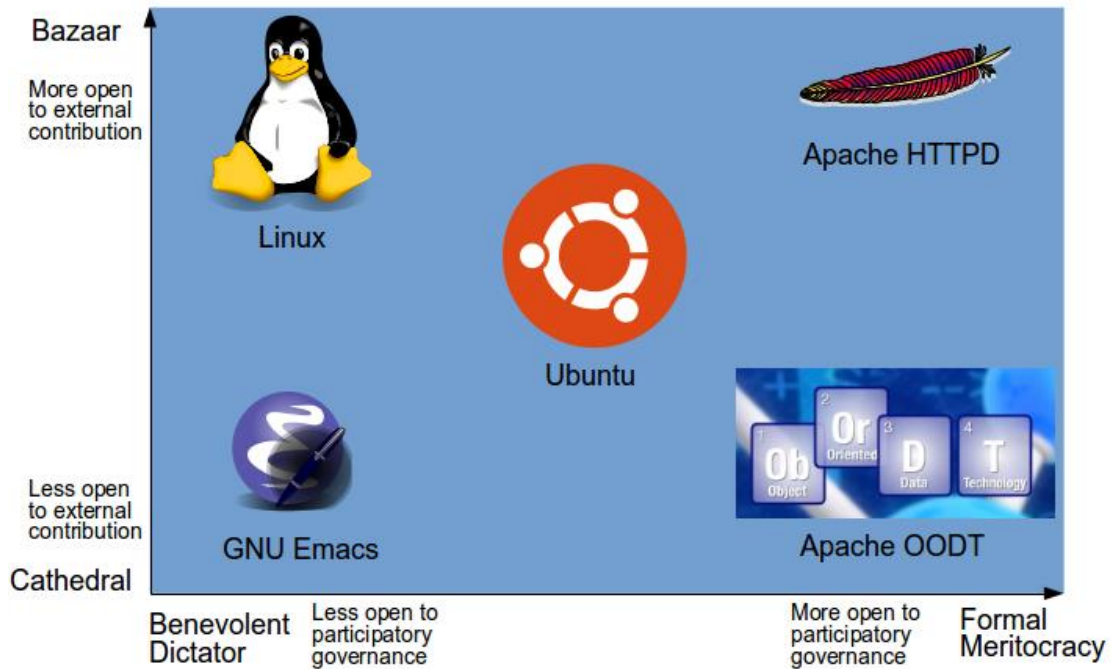
Figure 3 Governance and contribution models of open source projects (Ross Gardler, 2013)

From this figure we can see that Linux is a bazaar-style benevolent dictatorship. Linux uses broad range of contributors who suggest changes for the product, but the final acceptance of which changes are implemented is done by Linus Torvalds.

Apache HTTPD again is a bazaar-style meritocracy. Apache started when an original development team of a server abandoned their product and the existing users started to maintain it. When new contributors proved their commitment, they earned the merit to be part of the development community. Community grew in size and split into separate groups, each concentrating on different aspects of server development. These groups are called project management committees (PMC). PMC members control a project, accepting or rejecting changes from developers. Projects are normally auto governing and driven by the people who volunteer for the work. This is called "do-ocracy", power of those who do. And when coordination is needed, lazy consensus is used for decision making. This means that a few positive votes with no negative ones is enough to go. The rules require that a negative vote includes an alternative proposal or a detailed explanation of the reasons for the negative vote. The community then tries to gather consensus on an alternative proposal that resolves the issue. In the great majority of cases, the concerns leading to the negative vote can be addressed. (Apache, 2015)

Another interesting matter is which kind of tools open source communities use for collaboration, especially for communication. Main way of communication is mailing lists. This provides a way for discussion to happen asynchronously, which is important in a

project where participants come from different time zones. Emails are also archived appropriately, so they provide a valuable resource in form of recorded discussions and decision making. In addition to mailing lists, open source projects use code repositories and version control systems to manage the code. But for this thesis the most interesting point is that all discussion for complex projects can happen through mailing lists, with no face to face meetings like in so many traditional work places. (Gardler, 2011)

2.2.7   Open design

Open design is the designing of physical products according to the open source methodology. In a way open design is exactly what this thesis is about, with the exception that it only concentrates on products and the design phase, while open business model can be used to create services and in manufacturing as well. There are multiple platforms and projects that utilize the concept. Such as opendesignengine.net, ohwr.org, openhardwareclub.com, youmagine.com, open-electronics.org. Many of these are either electronic circuit board designs or mechanical designs that can be manufactured via 3D printing or CNC machining techniques. Therefore products that the users can manufacture themselves. The movement concentrates more on the design phase and sharing design data, and less on the manufacturing and selling. Like in open source, designing also happens for free and participants are not compensated for their work. Although, some business does evolve around open design products. The creator of RepRap, an open design 3D printing tool, also runs a company that sells those products.

Main difference between open source and open design is that software can be distributed with virtually no cost, while physical products require manufacturing. Therefore, when in open source a programmer can just copy the code and start doing changes, in open design a designer has to have the money and resources to manufacture the product in order to start to experiment with it. Another difference is that typical 3D CAD software used for physical designing is cost prohibitive while open source tools come for free.

Open hardware is a special type of open design movement, it concentrates on open design of electric circuit boards. Case studies-chapter contains one example of such, Arduino microcontroller. Other typical products of open design are OpenROV, an underwater remotely operated vehicle, 3D printable products at MakerBot Thingiverse, decoration products at Opendesignclub.com and furniture made of wooden plate by CNC machining at opendesk.cc.

2.3   Wisdom of crowds

A key element in open business method is that the crowd makes the strategic decisions itself and organizes the work on its own with no management. The question therefore is, how well can a crowd do this with no external management?

James Surowiecki in his book The Wisdom of Crowds suggests that a crowd, formed of diverse group of people can solve problems better than one single expert. An example of this is a test that a British scientist Francis Galton performed in the year 1906. The weight of an ox was to be guessed in a village fair. After the competition Galton bought the tickets to do some research of his own. He calculated the average of peoples' guesses and the result was that the crowd guessed the weight amazingly accurately. The ox weighed 1,198 pounds, while the crowd's average was 1,197 pounds.

The same was proven right in many other guessing experiments, such as guessing the amount of balls in a jar. Whenever a crowd has to guess the amount of something, it does it better than an expert. Noteworthy is that even some individual guesses are way too low, and some way too high, at the end they cancel each other out, and what is left is the good information.

The same works for other than guessing an amount also. Prediction markets are places where people can bet on who wins elections this year, or which sport team wins. Common to all of these is that a crowd outperforms single experts as a rule. (Surowiecki, 2005)

One specific example of wisdom of crowds in use is markets. The price of a product is a result of a crowd's common opinion on what a specific product should cost. There is no single leader telling how much everything should cost, but sellers and buyers together come up with a right price for each product. Sellers have their own opinion of how much it costs to produce the goods and how much they should get and buyers know how much they are willing to pay for each product. Together they form a system which defines the prices of everything for all of us. Therefore markets are actually an aggregation method for a big crowd of sellers and buyers to form a common opinion. And price is the product of this mechanism. And markets generally do a good in defining value. (Surowiecki, 2005)

One obvious way how wisdom of crowds can be exploited on open business platform is to let a crowd estimate how many hours it takes to finish a certain task. Instead of taking the workers word for it, a crowd can be let to estimate it and an average of the guesses is made.

Surowiecki suggests that four conditions must be met for a crowd to make good decisions:

- **Diversity**. Group consists of different types of participants, each having their own source of information.

- **Independence**. People's opinions are not determined by the opinions of those around them.

- **Decentralization**. People are able to specialize and draw on local knowledge.

- **Aggregation**. Some mechanism exists for turning private judgments into a collective decision.

Diversity means that the group has to be heterogeneous and each participant has to have some expertise and knowledge of his own to add value to the group. One experiment proved that a group that consisted experts and amateurs outperformed a group of only experts in problem solving. This indicates that diversity in itself is valuable, even if it means adding amateurs to the group. Problems arise if people are too similar and have too similar opinions. They enforce each other and extremes will not be cancelled out.

Independence means that people should form their opinions themselves without other people opinions influencing them. This is true for example in "guess how many balls in a jar" –competition, where people don't know each other's guesses beforehand. It is not true in traditional small group decision making, such as a committee. The course of the discussion, as well as the decision made by the group, is highly influenced by the people who speak first and by the people who have higher power.

Decentralization means that each participant is allowed to make a decision on their own without any higher management deciding for them. An example of such is Linux, open source operating system. Programmers can decide on their own which features to develop to the product, rather than being ordered. Another example is market. Each buyer decides on their own at which price to buy and at which not. An example of what is not decentralized is a company with a hierarchical management. Workers are told on which task to concentrate and decisions are made centrally, by the CEO.

Aggregation is a mechanism that gathers together each individual's knowledge and forms the final solution. In the case of the cow weight guessing game this was just the experimenter himself calculating the average of all the guesses. In the case of free market, price is the aggregator that combines the seller's and the buyer's opinions on the value of the merchandise. And in the example of open source software project, it is either the manager who chooses which features will be included to the product, or it is the crowd voting on it.

Surowiecki divides collective intelligence problems into three categories:

- **Cognitive problems**. This is a problem that has one right answer, or at least some answers are clearly better than others.

- **Coordination problems**. Coordination problems require members of a group to figure out how to coordinate their behavior with each other, knowing that everyone else is trying to do the same. How do buyers and sellers find each other and

trade at a fair price? How do companies organize their operations? How can you drive safely in heavy traffic?

- **Cooperation problems**. Cooperation problems involve the challenge of getting self-interested people to work together for a common good, such as paying taxes or dealing with pollution.

The examples mentioned before are all cognitive problems and a clear method exists for a crowd to solve them. Coordination and cooperation problems are more complex. On open business platform coordination problems would be such as coordinating the tasks. Who does what? How do we make sure that two people don't do the same task twice? And cooperation problems would be such as how to get the crowd perform unwanted, uncompensated tasks, such as cleaning data or fixing other people's mistakes, or to generally carry responsibility of the projects.

Some examples of well working coordination exists. For example pedestrians can remarkably well find their way on a crowded street. When two people have the same goal, to avoid collision, and both know what is the other one's goal, they can adjust well their own behavior to fit to the other. In one experiment a class full of students was asked where would they go if they had to meet a friend in the city and they had no way to communicate with this person. Most of the students chose the same location. When they were asked at what time they would go there, most of the students chose the same time. This shows that people are in some cases able to guess each other's actions, if they know that the other person is trying to achieve the same thing as they are. (Surowiecki, 2005)

Tax paying is a typical example of cooperation challenge. People are asked to contribute for a common good, even though it rationally it would make no sense for them to contribute. They would get the same benefits even if they didn't pay, and they would save money. Studies however show that people are willing to pay taxes. But certain conditions affect how gladly they pay them. People are happier to pay taxes if they know that everyone else is paying also, which means that the tax paying percent would be high. Or if they know that the ones who don't pay, get punished. A third condition is that they know that the money is well spent.

People do not always act only according to what is their own best interest, but other factors exist too. And these factors can be exploited on an open business platform. The fact that people are willing to contribute to a common good, such as tax paying, is one example. Another is peoples' motivation to do what they think is fair, even if it means that they have to pay for it.

This is proved in a game called Ultimatum. In this game two people are given ten euros to split between each other. But instead of just letting them decide together how to split it, one of them is randomly chosen to be the one who gets to decide the cut. He can offer anything he wants to the other, and the other can either accept it or reject it in which case

neither one gets anything. Rationally thinking, the other one should offer 9:1 split, and the other one should accept it, but this rarely happens. On the contrary, 5:5 split is most often offered and accepted. Therefore, the player chooses to be left with no money rather than accepting the small sum that he feels is unfair. It should be noted that the same players only play the game once, so the motivator is not teaching the other one a lesson. If the game is modified, by letting the participants think that the other one has earned their position as the one who chooses, by a fake exam for example, the dynamics of the game change. Now much lower offers are accepted, since the other participant thinks that the other one has earned their higher share. This kind of phenomenon can be used on a collaboration platform where a crowd has to decide the value of each other's work and keep up a high moral on the platform. (Surowiecki, 2005)

2.4 Work breakdown

At first it might sound like an inefficient method, to breakdown work into tiny pieces and then hand them out to many workers, while one hired guy could do the whole job with no extra management. But sometimes the benefits outweigh the extra management work. Such micro-task markets are already in use for some applications, such as translating books for example.

Mechanical Turk is a micro-task platform which crowdsources simple tasks, that require human intelligence, such as transcribing podcasts, classifying photos or judging the relevance of a search result. Each task takes a few seconds or couple minutes to complete and earn some cents. This way companies are able to finish a large work in fraction of time, and usually also cheaper than it would be when done in-house.

Dividing simple tasks like translating text to smaller pieces is easy, while more complex tasks such as mechanical designing on an open business platform would be more problematic. Some methods to divide complex work into subtasks exist. In their research paper CrowdForge: Crowdsourcing Complex Work, Kittur, Smus and Kraut propose a model where a complex task is first broken down to subtasks by the crowd and then aggregated back to a final work, see Figure 4. They use writing an article about New Your city as an example. First they ask the crowd to outline the article, coming up with different topics, such as History and Geology. Then the crowd is asked to find facts about each topic. And then to write a paragraph of those facts finally reducing to one final article. (Aniket Kittur, 2011)
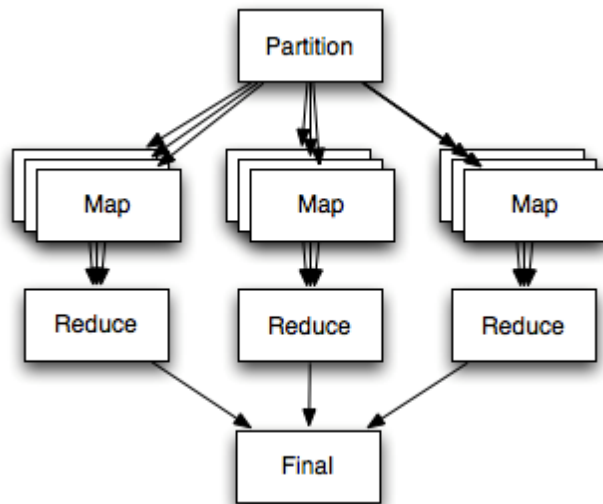
Figure 4 Work flow in CrowdForge method (Aniket Kittur, 2011)

It should be noted, that this model is developed for Mechanical Turk workers, who generally are amateurs and not motivated for long term tasks (Aniket Kittur, 2011). Therefore the model represents extremely divided task with low expertise needed. On an open business platform, more expertise can be expected from the workers and work would not have to be divided into so small pieces.

An interesting note on their paper is that work cannot always be divided into subtasks by one single person. They suggest that the work should be divided by a bottom-up approach, workers identifying the suitable subtasks. And dynamically, subtasks would be created as the work goes on, rather than all in the beginning.

On another research paper Kulkarni, Can and Hartmann suggest Turkomatic, a Price-Divide-Solve algorithm for dividing the work. This is a method to crowd-produce the workflow. First the crowd is given the whole work and asked can this be solved with a certain price, for example 20cents. If the user presses no, a task appears, asking the user to divide the work into smaller pieces. Same goes on until the user presses yes, it can be solved with 20 cents. Then he is requested to solve the task. In a case of writing an article, users would first divide the article into smaller sub-topics, based on their own judgment, and then these sub-topics would be written. (Anand Kulkarni, 2011)

By testing the concept, they found out that it works on some cases, but not always. Since answerers often are amateurs, they sometimes make mistakes, and the workflow doesn't allow others to fix the mistakes. When using the method with only experts, the system worked well. As a conclusion, it works, but sometimes needs supervision. (Anand Kulkarni, 2011)

What is specific about these micro-task markets, compared for example to open source initiatives, is that the users are not motivated towards a common goal, but asked only to perform a simple task, with no background information of the project itself, and a small

monetary reward as only motivator. In micro-task markets, the work can also be rejected by the job poster with no explanation, which does not encourage for big efforts. Therefore a micro-task market does not reflect well to an open business platform, where better motivators would exist. But what can be learned from these examples is, that work can be divided into even tiny bits, and it is cost efficient to do so, and it is happening. And also that it is possible for a crowd to divide work on their own and this way manage their own work.

2.5 Case studies

While such companies, which would operate exactly according to the open business method described in this thesis, do not exist, there are a number of companies and online platforms which operate partly in a similar way. Some parts of open business method can be verified by studying these examples.

2.5.1 Arduino

The Arduino project started in year 2005 when Massimo Banzi, a teacher at a design school Italy noticed that his students often complained they couldn't find an inexpensive, powerful microcontroller to drive their robotic projects. Banzi with two other partners developed an open source microcontroller and published the source code and the electrical design files. They manufactured a small patch and found out that there is bigger market for such a circuit board, and since then Arduino project has grown and the boards are known worldwide in open design communities.

The name Arduino was trademarked to protect the original quality of the products, which however has not stopped counterfeits. Lots of copycats also exists, good quality and bad quality ones. Arduino publicly accepts these copycats, as long as the name Arduino is not misused. Arduino does not make a lot of money by selling the boards. The way Arduino does make money, is by consulting companies who use Arduino boards. The interesting fact about a community like Arduino is, that the community evolves around the creators. The creators are the first to find out about new design improvements and the clients acknowledge this. Therefore Arduino creators are wanted as consultants on clients' projects. (Thompson, 2008)

Analysis: Arduino is a good example of an open hardware initiative. A physical circuit board is developed by a crowd. A disappointing fact is that the sales of the circuit boards do not bring in enough money to cover the development costs. Circuit board are easily manufacture by third parties. Therefore like open source projects, the Arduino community works for free. An interesting fact is that the creators are employed in clients' projects. If suitable tools were provided, the client could crowdsource these solutions from the community also. This could be a way to compensate for the community's work.

The difference between open hardware to other physical products should be noted. A circuit board comprises standard components, attainable by anyone and the manufacturing process is simple, it can be even made at home. This is not true for all physical products. It could be easier to gain revenue in products where brand, manufacturing methods and quality or distribution channels play bigger role than the design files.

2.5.2 Assembly.com

Assembly.com is an online platform where people create commercial software collaboratively. Each initial idea posted has a core team who initiate the project and manage it. The core team divides work into tasks, called bounties, and other people can execute these bounties. Each bounty earns App Coins, which equals certain percentage of ownership of the product. Users then earn royalties based on the amount of App Coins. The core team decides how much worth each task is, and the percentage of ownership depends on the total amount of App Coins distributed.

Here's an example of how awarding coins changes the owernship of a product over time:

1. @vanstee creates a product named Helpful and decides to give himself an initial share of 500,000 coins. @vanstee owns 500,000/500,000 coins, equaling 100% ownership in the product
2. @chrislloyd puts together a homepage for Helpful and is awarded a bounty valued at 50,000 coins.
   @vanstee owns 500,000/550,000 coins, equaling 90% ownership in the product
   @chrisloyd owns 50,000/550,000 coins, equaling 10% ownership in the product
   (Assembly, 2015)

Analysis: Assembly.com is the closest match to the open business model presented in this thesis. It serves as a good example of the compensation model in use. Noteworthy feature is that the revenue is shared among participants, based on the amount of work they do. And the openness that anyone can grab a bounty and work on it. The main differences are that in Assembly, a core team makes the decisions. So there still is higher management that decides on things. This means that the participants have to trust this leader to know what he is doing, otherwise the product will never become successful and their work is wasted. To gain this trust, managers should first work on other peoples products to gain reputation. This is mentioned as an advice on how to get participants to your project on Assembly's help section. Another interesting fact is that the manager also decides how much worth each task is. This means that you have to trust one person's estimation of the workload, instead of for example an average of a group. And of course, the biggest difference is that Assembly.com only works on software, not on physical products.

2.5.3 Quirky.com

Quirky.com is another interesting example. It builds the products internally in a traditional way, but it crowdsources a part of the design process. It works in the field of consumer products, such as house hold devices. The first thing they crowdsource is the idea itself. All their products come from the ideas people post. They select the best ones, make sure that it is not a copy of any existing one, and then develop it further. They also crowdsource the name for the product, and the price, sketches for the shape, product description text and some other things. Quirky compensates the community by giving 10% of gross sales to the participants. Each contribution earns a share of this 10%. Quirky calls this share an influence. As an example, submitting an idea earns you 40.5% influence.

Analysis: Quirky is an interesting example, because it crowdsources the idea generating phase of a new product design process. It acts as an idea bank and creates all their products from those ideas. That is something that open business model would also have, people posting product ideas. It also asks crowd's opinion on other stuff, such as the name. But Quirky only asks opinion, it doesn't let the crowd actually make the decision. What it also does not do is to include the crowd in the actual design process. Where Assembly.com hands out the design tasks to the crowd, Quirky uses in-house engineers to design the product and to manufacture it. Could Quirky also crowdsource the design tasks? Sure, why not? They could make tasks and let individual engineers from the crowd execute them just like Assembly. Quirky could also let the crowd actually decide on the matters, rather than just asking their opinion. If they would let the crowd decide and let the crowd do all the design work, they would be a pretty close to the model that of this thesis.

2.5.4 Innocentive

Innocentive.com is a crowdsourcing platform for technical problem solving. Companies post challenges, usually highly technical and demanding problems in different fields of science and set an award for the solution. Awards are also big, for example 50 000$. Then engineers and other professionals around the world solve these problems.

An example of a challenge: New Elastomer Material for a Spray Can Gasket. A gasket material that does not react with the stored substance. Award: $40 000. Solution must contain written part as well as experimental proof-of-concept data and a sample.

Another example: Atomization of Molten Steel on Industrial-Scale. The Seeker desires an improved method for the industrial scale atomization of molten steel. The solution contains a written proposal. Award: $20 000.

Analysis: Why Innocentive is interesting for this thesis is, because it involves finding a solution to a problem outside the company resources. It operates on the principle that a crowd is smarter and can provide better results than an individual company. This predicts that a crowd could be smarter in developing and designing any products than the small group of engineers in a company.

2.5.5 Mechanical Turk

Mechanical Turk, www.mturk.com, is a website which posts small tasks that require human intelligence and cannot be done with a computer. It is used for example to translate a big amount of text that with traditional methods would be too cost prohibit. How it works is that instead of hiring one person to do all the translation, you break the work into very small batches called HITs (Human Intelligence Task) and you let a big group of people (amateurs) work on those tasks simultaneously. Each task is usually worth of maybe $0.08. This way you get the work done quicker and cheaper than hiring a single professional. The quality can be assured by asking multiple persons to do the same task. And by creating another task to evaluate the quality of the previous one. Usual tasks include translation, classifying or naming photos, recognizing letters, picking more appealing product out of two options and transcribing media into text. All tasks on which humans are good, but a machines bad.

Analysis: Interesting point about Mechanical Turk it shows that it is possible to divide a big job into very small tasks cost effectively. One might think that it takes more time to create and manage the task than to actually do it. But Mechanical Turk proves otherwise. This makes one wonder what other kind of work could be divided into small tasks cost effectively? Could a typical design task of a mechanical engineer be split it into very small subtasks and then execute those simultaneously? Maybe a whole product design process can be split into tasks that are only the size of 1-2h. And then done those simultaneously. This way a product design process would be finished in weeks, instead of months.

2.5.6 Local Motors

LocalMotors.com creates commercially sold rally cars that are designed by the crowd. Like on Quirky, you submit your design, people vote on it and best ideas will be manufactured. Like on Assembly, projects call for help in needed subtasks. Contributors are compensated with a percentage of the revenue. Local Motors has also partnered with GE Appliances and created website called Firstbuild.com. It operates with LocalMotors.com principle but is about kitchen appliances. It reminds a lot the Quirky model of creating household products.

Analysis: Local Motors operates in an open business model in a way that part of the revenue is shared between participants and that it crowdsources part of the design process.

What is interesting is that in difference to Assembly it produces and sells physical products. Therefore it serves as a good example of open design model working in practice.

3. OPEN BUSINESS METHOD

The purpose is to develop a novel way of working collaboratively together, remotely without traditional company hierarchy. It would be implemented in practice by creating an internet platform where people first post new business ideas, goals. Then others create tasks that are required to achieve those goals. And then others execute the tasks. Everything would be self-managed by the crowd, meaning that the crowd creates the tasks with no single manager, and when there is a decision to be made, there would not be a single person making that decision, but instead the crowd would make it.

When one person posts an idea, others would start to post tasks needed to achieve that goal, such as "Do market research" and "Design the product" and "Sell the product". Each of these tasks would be further divided into subtasks. Then people would do those tasks and post the results on the platform. Each task would take a certain amount of working hours. The person who has done the task, would then earn that many hours. Finally when the goal has been achieved and the finished product/service is sold, the income would be shared among the participants. Each one would get a share comparable to the amount hours they have participated.

This is open business method. Anyone could join and all the income is shared among participants. And another key feature in this model is that there is no one single boss who decides on things. This means that any development idea can go forward and the model does not rely on just one person's professional expertise, but combines the whole group's knowhow. It relies on the wisdom of the crowd. And since no single person decides on matters, the direction where the company will be heading is defined by the crowd, instead of a CEO or a board of directors.

The model is comparable to open source projects, where a crowd together creates products and no one singlehandedly manages the endeavor. Or to ant colonies, where each ant carries one piece of building material eventually creating a fully formed nest.

It is noteworthy that not just anyone has to be accepted to perform a task. Selection criteria can exist. The purpose is, that an expert of the specific industry would perform each task. The revenue would also not be divided based only on the hours. Some tasks require more skills than others, and they would pay more. And a well done task would earn more than a poorly executed one.

What is the benefit of this working method is that one can start business without having money to hire workers. And the work can be done fast, since it is broken into small subtasks that can be done simultaneously by many people. And a new product idea would be

validated by a larger group, instead of just a few individuals, leading up to a higher success rate for new startups.

For a crowd to be able to work efficiently together, special tools are needed. Traditional hierarchical management structure works easily with no tools, except maybe a phone. But for a crowd that is spread over the world, to be able to communicate, more sophisticated communication methods are needed. Open source groups use mailing lists and voting systems, which work well already, at least in the open source software environment. These tools do not provide a method for sharing revenue though. Or a method to fully exploit the concept of wisdom of crowds. Therefore, for a crowd that consists of people from different fields, from amateurs to experts, creating physical products, more illustrative tools are needed. Next the methods and tools needed are discussed.

3.1 Work flow

First someone posts a new business idea. This is called a goal. That someone will describe the idea in as much detail as he can and explain why he thinks the idea could work.

Then other people start to work on the idea and build it forward. And this happens with the help of a task list. People will start to post tasks that are needed to be done to achieve the goal. For example first the idea needs to be verified. This can be done by for example via market survey. Therefore the first task might be: "Conduct a market survey to verify the idea". And this task could be further broken down to subtasks, such as:

1. "Conduct a market survey to verify the idea"
    1.1. "Find suitable interviewees"
    1.2. "Make market survey questions"
    1.3. "Ask the questions from interviewee 1."
    1.4. "Ask the questions from interviewee 2."
    1.5. "Ask the questions from interviewee 3."
    1.6. …
    1.7. "Ask the questions from interviewee 50."

It is worth to notice, that many of these subtasks can be performed simultaneously. If there is a group of 50 workers and each one finds one person to interview, they can finish subtask 1.1 in maybe half an hour. And while they are doing that, someone else can come up with a bunch of market survey questions, finishing subtask 1.2. And with a group of 50 workers, the subtasks asking the questions 1.3 – 1.7 can again be finished in less than an hour.

Another observation is that the questions are likely to be much better if they are done by 10 marketing experts, each making couple and then collectively choosing the best ones, rather than in the traditional way, in which one or two people create them all. It is also

easier to find people to interview, if a big group of people is asked each to suggest one, rather than one person trying to find all the suitable candidates oneself. A real life example of this kind of breaking the work into little pieces is Mechanical Turk.

After market research, it would be time to make new tasks. If the market research gives positive initial results the crowd would probably choose to continue with the idea. The next tasks could be such as:

1. "Design the product"
2. "Ask clients' opinion about the initial design"
3. "Decide how to continue based on client's opinions"
4. "Make design changes"
5. "Manufacture a prototype"
6. "Test the prototype"
7. "Do initial sales"

And this forms a roadmap that leads to the goal. The roadmap would be dynamic, meaning that new tasks would be created based on the results from the previous ones. For example when clients don't like the initial design, the next task is "Make design changes" instead of manufacturing the prototype.

Problems arise when two people create the same task twice. One of them is unnecessary. Or if someone creates a task that has no value for the project at all. They would earn hours by doing nonsense. For these situations a valuation factor is introduced. A crowd can vote how much they value a certain task or the goal that the task leads to. By default every task would get a factor 1. If the task is unnecessary, its value could be voted to 0 meaning that the worker does not earn any hours of that task.

Another situation, that one might consider a problem, is when two different approaches are created to achieve the same goal. For example two different marketing strategies or two different product designs. In a traditional company, one would be eliminated and the one that the manager prefers would be executed. But on an open business platform, resources are less limited, and two approaches can be developed simultaneously. And they should be, since often there is no way to tell which approach is better in advance. Again, value factor could be used to eliminate the obviously amateur approaches.

3.2 Goals, pre-goals and pivots

In order to make a big group of people to work to the same direction, a goal has to be defined that everyone will aim at. This chapter presents how the goal should be like and what happens if someone doesn't agree with the goal.

The creation of a goal starts by someone posting an idea on the platform. He will write what is the idea, benefits and why does he think that it is a good idea. An initial posted idea might look like this:

Title: A computer controller that replaces traditional mouse.

Description: While traditional mouse is used by one hand sliding it on the desk surface, this new controller would be used by standing and with both hands and maybe legs also. It is similar to controllers used in computer games, that one can rotate and move in free air and objects on the screen move correspondently.

Benefits: This would make working, especially 3D designing, much faster and computer working would be healthier also when you move while working.

Next the crowd would start to work towards this goal. The first problem that arises is that the goal is not clear enough. The same goal can be achieved by some kind of gloves that record ones movements or by external sensors that record hand movements.

Another problem is that this goal is not detailed enough. It doesn't define clearly the target market or the price, or whether it should work wirelessly or with a wire, or whether many versions of it is needed. All little details that would have to be defined in order for the engineers to design it.

To handle these issues alternative goals are introduced. One group might start to work on the glove solution, while another group likes the sensor approach better. And each goal should be defined in details, so that engineers who create the tasks, know where to aim at. There would not be a task, "model the wire" if a wireless approach was chosen. The purpose is that the discussion and dispute over different solutions would happen under the goal-section, and not anymore when tasks are being created.

In order to achieve the main goal, pre-goals are introduced. For example, if the final product is a full scale data management system, a pre-goal might be a lighter version of this, to test the market. It is crucial to define it as a pre-goal, so that there is no confusion among the programmers, which features should come to this version and which are planned for the final one. In the case of a physical product the pre-goal might be an early prototype to test the functionality, or it might be a pre-sales effort or a market research to find out the demand. Basically a pre-goal would serve as a milestone, when something sellable gets done, or some part of the idea gets verified and its purpose is to guide the crowd to work towards the same goal and avoid confusion.

The same goal can be achieved via different routes. Just like there is many possible routes to the city library, on a business roadmap there is also many routes to get to the same goal. It is not a bad thing to try to reach the same goal by two different methods. But confusion should be avoided. Therefore some kind of a description of the route might be

needed to explain people which kind of tasks to include in this route. A route might for example rely on Lean Startup methods, meaning that a minimum viable product is designed and shipped as early as possible. Therefore, that route should not contain tasks of excessive product design efforts, but tasks for early selling should be created. Another approach for the same product might be designing the first version for a certain customer and grow from there. Therefore only tasks that support that customer's needs should be created. Pre-goals for these two routes would also look different and would help to clarify the purpose.

A pivot is what a company does when they change their goal. It can be a change in product design, or a customer segment change or any other kind of change of plans. Start-ups do pivots often. It is not clear from the start how a product should look like or who are the most potential clients. A pivot for a computer controller could be for example that the clients don't like the controller with gloves. Then a pivot could be to make the version with the external sensors. Or if 3D CAD designers don't like it after all, but gamers do, then a customer segment change might be in order.

3.3 Tasks

Basic principle is that anyone can create a task at any time. And if the task does not fit the route or the goal, it can be deleted by anyone. There is two ways to ensure that participants do not misuse or sabotage the system. Either by programming each rule to the interface, or by letting the crowd vote down the misuser, limiting their profits or access. Open source projects, such as Apache, prove that a healthy community culture is achievable and people can adopt community rules without forcing. Therefore the method of letting the crowd guide participants towards good behavior is preferable.

A task should contain at least a description what needs to be done, such as "design covers for the computer controller". But it has to be defined also what kind of covers, plastic or metal, what is the shape? Now, there is two ways to do this, either it will be defined clearly on the task itself, or it is not and the worker is let to find the required information from the data management system himself. There would be previous tasks, such as "decide material for the computer controller covers" and "sketch the shape of the covers", so it would be possible for the person to do this.

What is good about writing all the information on the task itself, is that any random engineer can just grab the task and do it without knowing anything else about the project. So for an engineer, specialized in molded plastics, jumping from project to project doing only molded plastics design tasks, it would be easy to do the task when he finds all the required information right there. The good thing about not writing anything on the task is that it is remarkably easy to create the task when one only has to write the title and that's it. Easy and fast is also one of open business principles. So the best is to adopt a golden middle way. The task doesn't have to include any written instructions. But links from

previous tasks, that contain the needed information, could be added to the task. So the task creator has the possibility to give all required information within the task in an easy way. The task worker can also contact the task creator, to ask more instructions on the task. The assumption is that whoever created that task, knows enough about the project to know how the task should be done. And the worker doesn't have to learn anything about the project to be able to do a task.

It is worth of noticing that task do not always need to be created by hand. If an open business platform has an interface for customer support for example, a task could be automatically created once a worker answers customer's question.

3.4 Decision-making mechanism

Decision making can be thought of a two part process. First one comes up with different ideas how to solve the problem at hand and then chooses the best one. In hierarchical model the manager does both processes himself. Or he can listen to other people's suggestions and then choose the best suggestion. Manager makes the choice based on his own knowledge and experience.

Now, there is no reason why the same couldn't be done by a crowd. First people would post suggestions and then the best suggestion would be chosen by voting. The result would be the same, but the problem would be solved by a big group instead of one person.

Decisions are also often made in small groups, instead of just one manager. This model contains many problems also. One is that studies suggest that in small groups, the order of comments affects the course of the discussion. Earlier comments have more influence than the ones that come afterwards. This might be fine if the first person would also have the most information on the topic. But studies also suggest that the first ones to speak are not usually the ones with the most knowledge on the topic. And especially, if there is higher ranked people in the group, they tend to talk more even when they don't have knowledge on the topic. Because of these reasons the group as a whole can end up making bad decisions. (Surowiecki, 2005)

Another problem is, if they group is too homogenous. If all the members share the same ideology, for example taking a risk instead of choosing the safer option, as a group they end up taking higher risks than any of the members individually would want to take. This is called group polarization and it radicalizes members' opinions. (Surowiecki, 2005)

Open source platforms make decisions by discussion and lazy consensus. And this seems to work, at least for them. By voting, they take into account everyone's opinion. Although the discussion procedure in open source projects assumably has the same problems as the small group decision making in face to face situations.

The theory of wisdom of crowds requires that the crowd should think independently. If the participants are aware of others opinions, it can affect their opinion, causing group-think. This happens during discussion, people hear someone else's point of view first and it might affect their own opinion. They might discard their own idea, thinking it is stupid. But those ideas are valuable, because they represent diversity, another key factor in wisdom of crowds. Even if the comments were not correct, they make the others think twice their own opinion, making the group analyze the problem harder and resulting in better decisions. For good problem solving, four factors are needed:

- **Independency**. Participants should have a possibility to form independent decisions without being influenced by other people's opinions.
- **Diverse group**. A group should consist of experts and amateurs and people with different background. It is proved to make the decisions of the group smarter, and it cancels out polarization.
- **Decentralization**. Participants should be allowed to choose themselves on which problem to work on, rather than someone giving orders.
- **Aggregation**. A method to aggregate participants' contributions into a single solution.

Group size can be assured by making a rule that a decision requires a certain amount of participants to be valid. And diversity by asking participants which types of people they represent (programmers, electrical engineers, marketing people, young, old, men, women, manager, non-manager) and making a rule that certain type of diversity has to be present for a decision to be valid. Decentralization is assured in the open working mechanism where tasks and goals can be created by anyone. Everyone can choose their approach to the problem.

Independency is the most critical issue. On an open platform people can easily be affected by others. Innocentive is a website that crowdsources technical problems. It solves problems by choosing the best out of all the suggestions from the crowd. None of the suggestions are revealed to other answerers during the process. While this assures independency of thoughts, it does not let the crowd to build on each other's knowledge. Therefore another approach is introduced.

Independency can be assured by making people first post their suggestions and opinions independently, providing clear explanations on what they base their ideas. This way the most diverse range of points of view would be collected. Then these thoughts would be published. At this point, everything that the group knows about the topic is out in the open. If the problem is to come up with a marketing strategy for a new product, the data might look like this: someone might post which strategies at least don't work, based on his previous experience. Another participant might point out why the research data used for the strategy is not valid. Third participant might post examples of good marketing strategies for similar products that he has worked with. Fourth participant might post a

link to a research about the targeted customer segment. All valuable information, which would not all come up during a group discussion.

Now that all the data is out in the open, the next phase is to start to work on this data. This is where the crowd can build on each other's knowledge. Participants can comment on each other's posts, ask for prove, and use each other's knowledge, to modify their own idea. They learn about things that they didn't take into account earlier and can combine parts of others' posts to form their own final ideas. And since everything is out in the open already, no one can hide their thoughts anymore like in a normal group discussion.

The first posts should concentrate on facts and knowledge based on experience, rather than just opinions or suggestions. All the data should be somehow proved by the poster, either by presenting research results or explaining on which experience he bases his opinion.

Important thing is also that the first posts should be anonymous. This is to assure that everyone's thoughts are taken into account equally and that the reputation of the poster does not have an influence on how his opinions is received by the community.

Next, when all the knowledge of the crowd has been dug out and analyzed it is time to make the decision. The crowd would be asked to post another round, this time suggestions for a solution, and then these solutions are simply voted and best one is chosen. Suggestions from the first round would also be taken into account. This might sound like a time consuming process, but actually it would be only a couple of posts per person and take no more time than traditional face to face negotiations.

Making a decision means choosing one out of two or more options. And it only has to be done if only one option can be chosen. For example when arriving to a T-cross section, one has to go either right or left, but cannot continue straight. But if there is two people, then the other one can go left and the other one right and no common decision has to be made. Same goes for business, out of many ideas, everyone can choose their own way, as long as it doesn't harm the others. This is not true in a traditional company, where the company pays salary for the workers. If one worker works on something foolish, the company loses money. Therefore the boss decides which ideas will be worked on. In open business model the only thing lost is the worker's own time. Therefore anyone can be allowed to work on their own ideas and it is up to the others whether they want to help in that idea or not.

That is also how open source software projects work, people develop their own ideas and some of them are chosen to the final product and some not. But everyone is allowed to try, and that is the most important thing in harvesting innovative solutions.

Therefore, on open business platform, decisions are only made when they have to be made, when only one out of two options can come true. And the process used and time

spent, shall depend on the size of the problem. In small problems, the worker can choose himself, in bigger ones the procedure described earlier would take place.

3.5 Project tree

In order for people to collaborate with each other, they need to have a way to see what other people are doing and coordinate their own actions based on that. The user interface plays a bigger role on open business platform than in traditional companies or on open source platforms. In traditional companies people have years to learn the IT systems of the company. And on open source platforms, everyone are programmers, they only deal with one kind of task, coding, and they usually concentrate on one open source project at a time. On open business platform, people represent all professions from restaurant waitress to a researcher, therefore they possess different level of computer skills. People might also work on one project only to finish one single task of their field and move on to execute similar tasks for other products, therefore they need to quickly get the hang of the project. And because tasks vary from designing to physical tasks, the data management system has to accommodate many kinds of task results.

This chapter introduces one approach to solve the issue, a project tree. A tool to create tasks, see what others are working on and to quickly get the hang of the current state of the project. It is an illustrative way of expressing information and the purpose is that viewer can with one glance to get all the relevant information they need about the project.

First, things that a person needs to know about a project, in order to determine whether to participate or what skills are needed, are defined. Then a way to express those things in the quickest possible way is introduced.

Main things that a new participant would need to know about a project at any time are:

1. What is the goal of the project?
2. Which tasks needs to be done to achieve this goal?
3. Progress, which tasks are done, which still need to be done?
4. Pivots, what has been tried and what has failed?
5. Activeness, how many participants are working actively on the project?
6. Prospects, is there interested customers or early sales?
7. Duration, how long each task and the whole project take?
8. Which kind of skills are needed for the tasks? Are my skills needed?
9. Which things are most urgent? Which tasks should be done first?
10. Is there problems that prevents us from proceeding forward?

In order to express all of this information at one glance, all possible means to transfer information to another person has to be used. A graphical illustration is always faster to

understand than written or verbal. Gant's chart is a traditional way to express a projects state graphically. Gant's chart expresses:

- Duration of a project
- Each task and its duration
- Dependency between tasks

A project tree follows the principle of Gantt chart what comes to expressing duration and dependencies. Length of a task expresses its duration and simultaneous tasks are located next to each other. A project tree is kind of a three dimensional Gantt chart with extra features to express more information. The extra dimension is used to express the pivots within a project. Whenever a pivot happens a fork appears in the timeline and the new goal is added above the old one. This way the shape of the project tree tells the viewer already a lot about the history of the project. Alternative goals are expressed in the same way. Figure 5 illustrates how a project tree looks like.
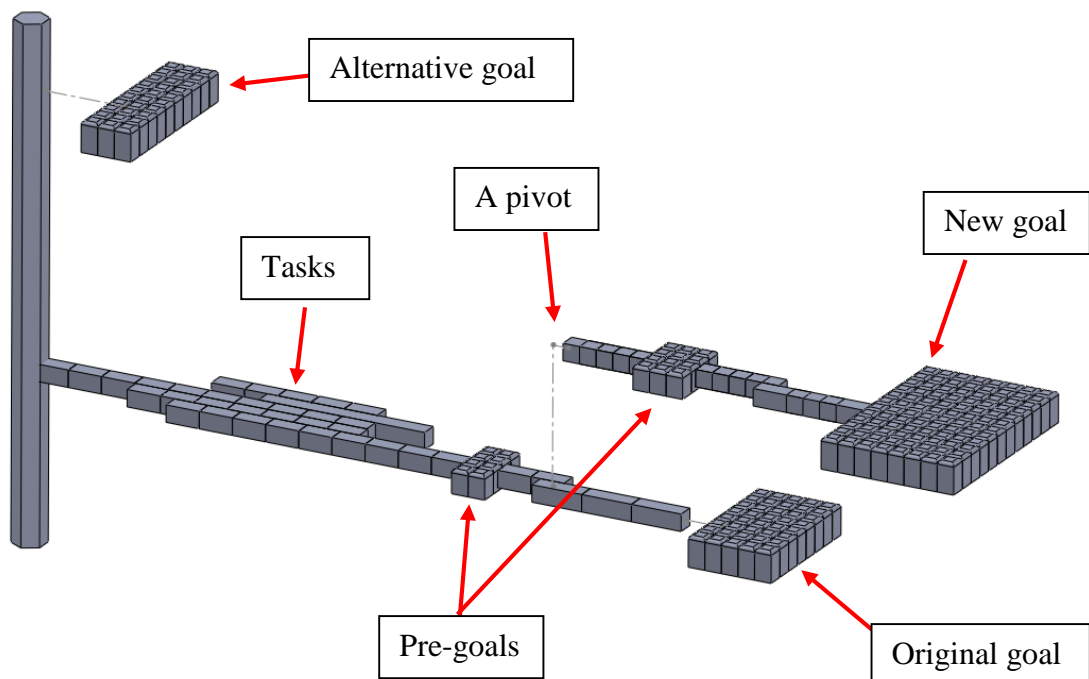
Figure 5 Project tree

From the top (Figure 6, illustrates the first level), the tree reminds a traditional Gantt chart, expressing the total time for the project to finish as well as the dependencies of the tasks. In addition the pre-goal and the final goal are presented. One difference exists, sequential tasks are not located below each other, but sequentially in one line to make it clear and concise. The line before the goal expresses that more tasks are needed in order to achieve the goal.

Sequential tasks are places on after the other in one row

Tasks are still missing

Figure 6 Project tree from the top, first level

Side view (Figure 7) shows alternative goals and pivots. It also tells when a pivot happened and this way one can trace back the reason for the pivot. The difference between a pivot and 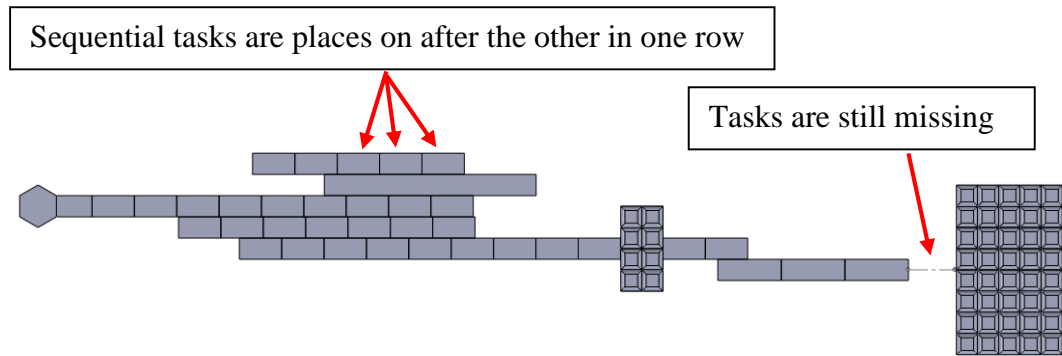an alternative goal is that a pivot happens when it has been proved that the original goal does not work, and an alternative goal is a product of disagreement on project goals. The shape of the tree can tell a lot about the project. Having many alternative goals means that there is not clear agreement how the product should look like and efforts are spread across many approaches. It may not be a bad thing. If new technology is being developed, it might not even be possible to identify one correct solution. Having many pivots might mean that the product is not needed and the crowd is having hard time trying to sell and therefore it can be a bad thing. Although depending on the business, sometimes many solutions has to be tried to find the one that works and pivots are just a natural way of developing new products. What the viewer really can see from this, is which approaches has been tried and which has not yet been thought of.
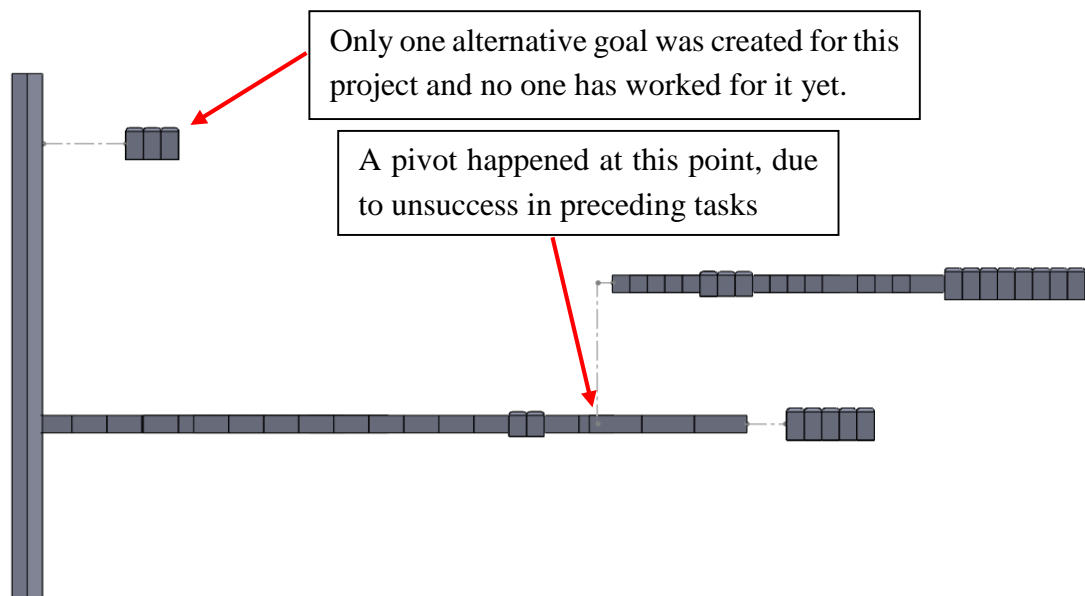
Only one alternative goal was created for this project and no one has worked for it yet.

A pivot happened at this point, due to unsuccess in preceding tasks

Figure 7 Project tree from the side

There is still multiple features that can be used as indicators. Such as:

- **Color**
- **Shade**
- **Shape**. For example, a task could be rectangular or round or rectangular with beveled edges, all indicating different things. Or there could be stars or exclamation marks on the project tree indicating something.
- **Size**. Length of a task already means the task duration. The height of a task could mean for example how many people are working on it. Size of a goal could indicate its popularity.
- **Movement**, such as fast or slow vibration. Movement can also be up and down or sideways, or rotating vibration. All different movements could be used to indicate different things.
- **Sound**. There could be a sound when pointing to a task. For example a voice saying the title of the task.
- **Changes.** Changes in color or size for example. A task turning from one shade of red to another, could indicate that it is overdue.

Some indicators are easier to spot than others. Colors for example are quick to spot. Therefore colors could mean something significant. Colors are also plenty. Therefore colors can be chosen to indicate different fields of profession. For example red can mean a marketing task, blue 3D design, yellow a strategic decision and so on.

Another feature that is quick to spot is movement. If there is one fast vibrating object on the screen it immediately catches attention. Therefore fast vibration can be chosen to indicate a problem with executing a task. The worker has encountered a problem and

couldn't finish the task. Then someone who knows the solution to the problem can go and fix the situation.

If the problem has nothing to do with any existing task, but rather some tasks are needed to be added, in order to solve the problem, that situation can be presented with an exclamation mark between the tasks.

Slowly vibrating task can mean that the task has been sitting a long time and no one has claimed it. This might be, because there is no one in the group who has the required skills. In such case the next step would be for someone to create a task: "Look for new participants with the required skills".

To show the progress, in other words which tasks have been done and which still need to be done, shade can be used. The tasks will turn into a darker shade after they are done. This way one can see with one glance which tasks are done. In addition to this, the platform can have a figure on the side of the tree, showing the progress in percentage or in hours.

With the extra features, the project tree is now able to indicate almost all the information that a participant needs to know. Figure 8 illustrates the tree.

Tasks are done this far. Finished tasks turn into gray. Green, yellow and red tasks are not done yet. The pre-goal was achieved and it has turn into gray. Green tasks mean mechanical designing, yellow means hydraulic design and red means sales.

At this point there was a pivot. The early product in the pre-goal didn't sell and a change of course was needed. The rest of the tasks on the first level were not finished.
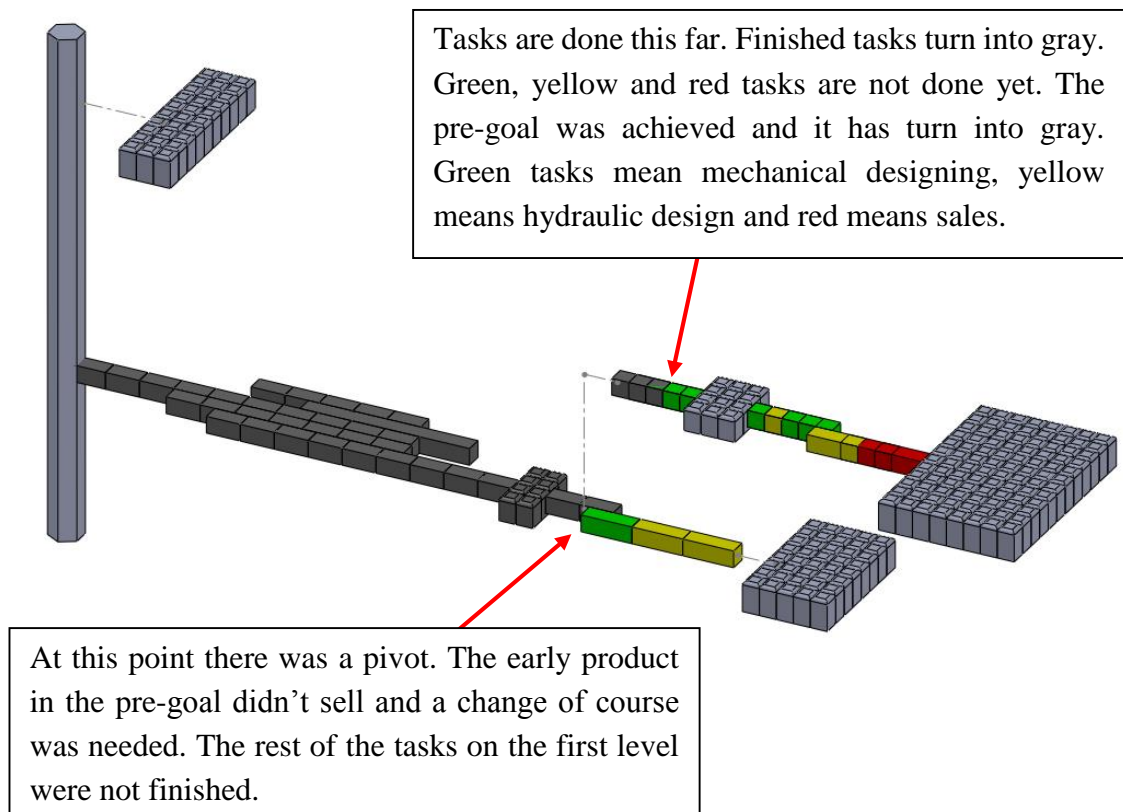
Figure 8 Finished project tree

The only things missing are features to indicate how active the project is and the prospects, whether there is early sales or customers interest. The early sales and customers interest is easiest to indicate with just a figure next to the project tree. The activeness can be shown as a figure too, for example how many posts per week or how many participants. Another way is to locate the tasks that are done into time slots when they were actually done. And long caps between tasks would indicate that people were not working full time on the project. Figure 9 illustrates the real timeline.
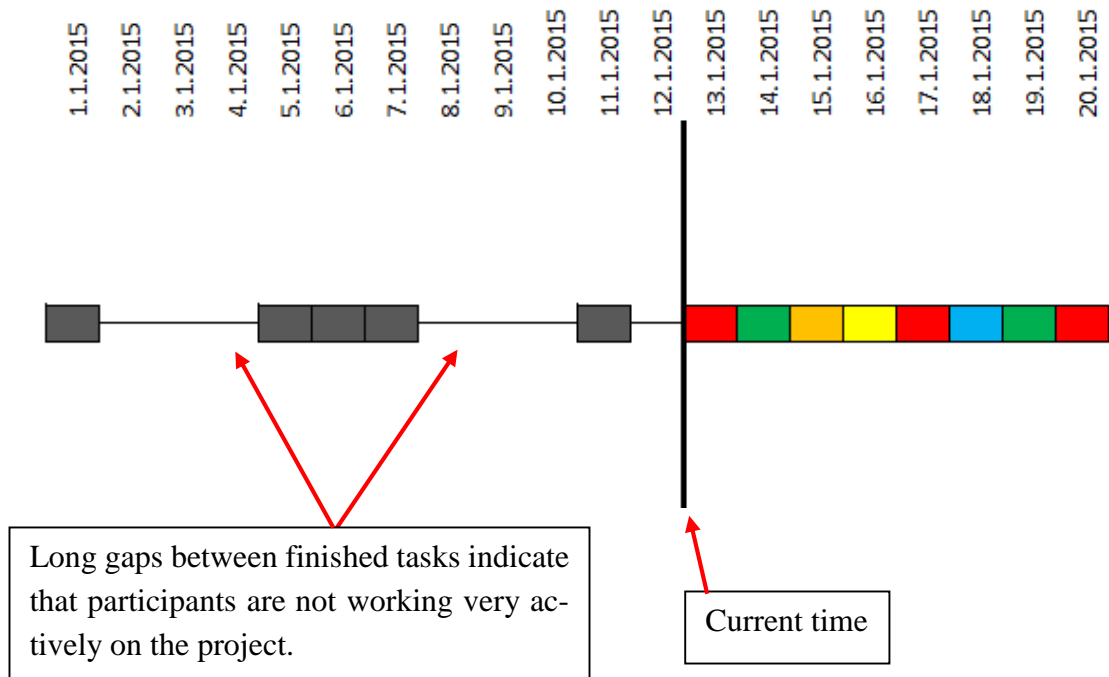

Figure 9 Real timeline

So far the project tree does not show what each task is. One way to show this would be double-clicking a task to open on a new page. Or hovering over the task with mouse and a title would appear. Both methods show only one task at a time. Another simple method would be to show the name when the viewer zooms close enough, see Figure 10.
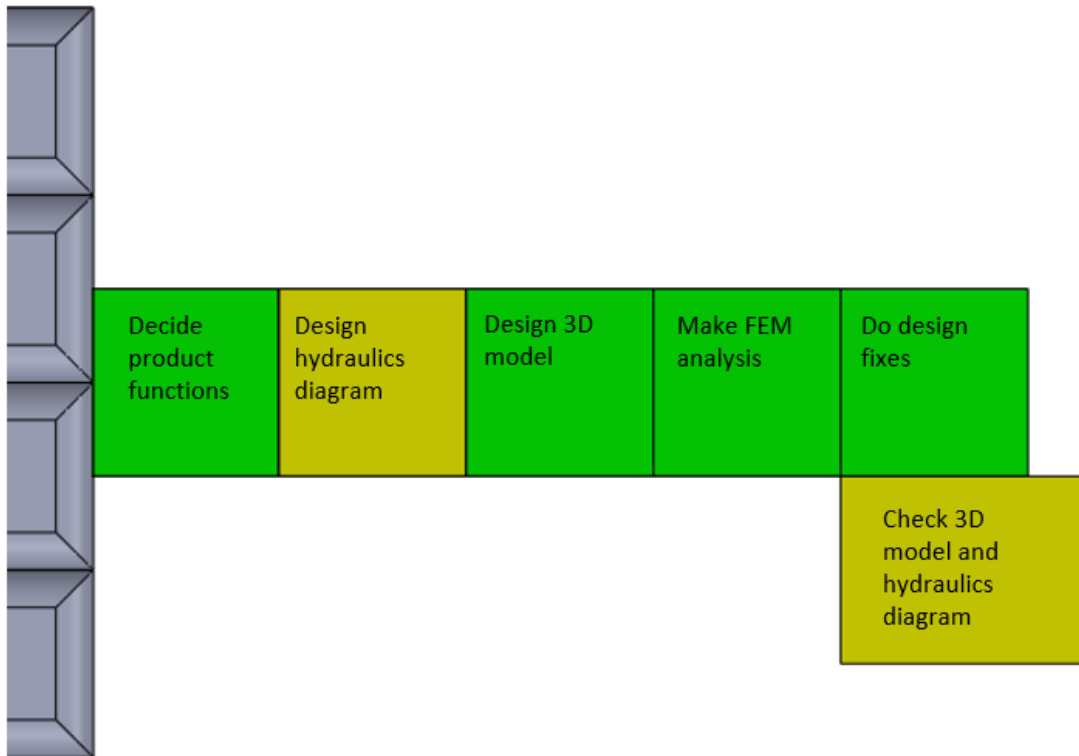
Figure 10 Task titles

And when the viewer zooms more, the task description appears, see Figure 11. Now one can pan to the left or right to see other tasks' descriptions.

Figure 11 Task description

Task based revenue sharing requires that everything that workers do, is written as a task on the project tree. But it does not have to be time consuming or bureaucratic. It can be fast and flexible. The tasks are used for two purposes, to define how much each participant has worked for the project and to show newcomers what has been done and what needs to be done. The tasks can be small such as one hour long or big such as weeks,

depending on the need. The tasks only need to be divided into smaller pieces if more people want work on a same task simultaneously to finish it faster. Tasks are also a way to tell others what everyone is working on. If someone wants to design a new function to the product, they can simply write a task: "Design new lifting mechanism", and then click that task and start working. The crowd can afterwards look at the result and evaluate, how long such task took and was it important for the product. The only thing a worker risks is their own time. Different kind of voting systems can be used to verify whether some tasks or goals are approved by the crowd and will be compensated for sure, in case a worker don't want to take a risk or they are executing a task created by someone else.

3.6 Compensation method

The main idea of the open business method is to divide the revenue between the participants based on how much each one has worked. If the company makes 120 000EUR sales this year, and material costs, software licenses, taxes, investors' share, etc. costs are 20 000EUR that year, it means that 100 000EUR would be left to be shared as a salary for the participants.

Executing a task would earn the worker the amount of hours that it has taken to finish the task. The only question is how to define this. It would be highly unreliable to trust the workers word for it. Nor is there any way to measure such. The solution for this problem to estimate the amount of hours by the crowd. As mentioned earlier, a crowd can very accurately estimate amounts, such as how many balls in a jar or how much does a cow weigh or who wins the elections. Therefore it can be expected, that a crowd's estimation on how many hours it takes to finish a certain task is as accurate as possible.

The time is not the only thing that matters when compensating work. Other things, that a person's salary normally depends on, is the level of difficulty of the task and how well the work has been done. The following formula can be used to calculate a person's income for a task.

$h \times d \times p \times r \times i = t$

Where:

h = Working hours estimated by the crowd
d = Task difficulty factor
p = Performance factor
r = Risk factor
t = Total amount of credits
i = Importance factor $(0 \ldots 1)$

Table 1 shows task difficulty factors. They depend on the education level required for a task as well as on the work experience. The task difficulty factor corresponds to salary

differences for different jobs in normal work life. Certain professions would be categorized to a certain class. For example mechanical designing could be in class 2.5 and sales tasks in group 3. This categorization could be changed according to public opinion. If the crowd votes sales tasks down, they could be moved to class 2.25 for example. Since work is done is small tasks, new professions could be formed also. For example mechanical designing could be divided into actual designing, meaning choosing shapes and materials and joining techniques, and to 3D modelling. 3D modeler would draw what the designer has decided and would require lower education than the designing tasks.

Table 1 Task difficulty factor

1	Simple mechanical tasks, no education needed
1.25	Low education needed
1.5	...
1.75	Some education and work experience needed
2	...
2.25	...
2.5	Expert level tasks, high education needed
3	Expert level tasks, high education and experience needed
4	...
5	CEO level tasks

Table 2 shows job performance factors. Crowd could vote task results up or down depending on the quality of the task results. Default value would be 1 and if the job is exceptionally well done, the worker could get 20% higher payment for that task. Zero would mean that no money is paid for that task and it has to be redone by someone more talented. This would filter out people trying to do tasks that they do not have expertise on.

Table 2 Job performance multiplication factors

0	Must be redone
0.7	Requires fixing
0.9	Low quality
1	Normal performance
1.1	Job well done
1.2	Excellent work

The risk factor in the formula means at which state of the project the work was done. Tasks in the beginning of a new product development earn more, since it is riskier to work for a startup, rather than for an established business. This way the first year workers would earn more than the fifth year workers with the same amount of hours.

Task and goal creation are something that an amount of hours spent cannot be defined. How much worth is an idea and how much worth it is to create a pre-goal or a task, are subjective matters. It might only take an hour to write an idea, but the value of the idea might be much more than one hour's work and it might have taken a lot of previous

experience to come up with such an idea. Since these are subjective matters, voting would be the best way to solve it. Just like in any market, the prices for ideas would eventually settle to a correct level, once the voters notice that not many ideas come in with too low prices and not many experienced professionals create tasks and roadmaps with too low pay.

Payment period. If the work done in the first year is compensated by the income from the first year, people might end up with nothing, if there is no income yet in the first year. This model of payment would work only if the income remain same every year, which is not true. The situation would be clear, if the total income of the business throughout its whole life time would be known. For example, if the business runs for ten years, after which it would stop existing due to any reason, it would be easy to share the profits. It would just be calculated according to the following formula:

$$\frac{Total\ revenue\ of\ the\ business\ from\ 10\ years\ time}{Total\ amount\ of\ earned\ credits} = Value\ of\ one\ credit$$

And a person's income would be the credits earned multiplied by the value of one credit. But in this way the revenue would be only shared after ten years, which is not very attractive model for the workers. A solution is to divide revenue each month, calculating the total revenue of the company till that point of time and to share it according to the earned credits. It has to be taken into account that some workers have got paid some already in the previous months. The formula for each month would be the following:

$$\frac{Total\ revenue\ of\ the\ business\ till\ this\ time\ point}{Amount\ of\ earned\ credits\ till\ this\ time\ point} = Value\ of\ one\ credit$$

This model assumes that a new worker has a right only for the revenue created after he joined the company, although he gets to enjoy the fruits of previous contributions in form of high sales. But if the sales goes suddenly down, and a new worker has put a lot of effort to the company in that month, he bears the risk of not getting paid, since the previous revenue has already been shared to the old workers. But since the payment period is short, and workers assumably work on multiple projects at the same time, the risk of wasting a whole month's salary is low.

Table 3 illustrates the share of revenue for one project for the first twelve months. Credits are working hours multiplied by the multiplication factors. More work was done in the first months of the project, which is typical for a product development process. Revenue has been rising throughout the year and so has the value of one credit. If a person worked 20h for the project only in the first month, with all multiplication factors being 1, he earned 1.25 EUR/hour in the fourth month. But he got paid more in each month without doing more work. In fifth month he got paid 2 EUR - 1.25 EUR = 0.75 EUR/hour. In 12th month his final hourly salary for those 20h worked in the first month turned out to be already 29.19 EUR/hour. And next year this figure would be rising, assuming that the

product is good and sales keep rising. The long delay in payment would be compensated by the risk factor, meaning that one worked hour in the first months would equal more than one credit.

Table 3 Revenue share for the first 12 months

Month	Credits	Total credits	Revenue	Total revenue	Value of one credit
1	2000	2000	0	0	0
2	2000	4000	0	0	0
3	2000	6000	0	0	0
4	2000	8000	10000	10000	1,25
5	2000	10000	10000	20000	2,00
6	1500	11500	10000	30000	2,61
7	1500	13000	20000	50000	3,85
8	1500	14500	20000	70000	4,83
9	1000	15500	20000	90000	5,81
10	1000	16500	100000	190000	11,52
11	1000	17500	150000	340000	19,43
12	1000	18500	200000	540000	29,19

It is worth of noticing that not every task would have to be evaluated by the crowd individually. Repetitive tasks, such as answering a customer's request, could have a predefined value, and only when the task significantly differs from the expectation, the worker could request that task to be separately evaluated. In general, tasks can be evaluated before execution or after, making it fast and flexible to execute them. And whenever the content of a pre-evaluated task has changed significantly during execution, the worker can request it to be re-evaluated.

3.7 Governance models

The main reason why tasks are created of every work that workers do, is so that the revenue can be shared between workers. And the main reason why hierarchical management structure is avoided, is so that innovation can bloom and ideas are not suppressed. This of course brings the problem how to coordinate the work. It was said in the tasks chapter that tasks can be created by anyone, meaning that anyone can do whatever they like. But it was also said that the crowd can vote some task's importance to zero, in which case the creator would not get compensated for that work. This function alone works as a one governance model. Anyone can try anything they like, but whether they get compensated for it or not, is up to the crowd to decide.

Basically it is easy to use a crowd as the decisive organ. First different options are presented and then the crowd votes on those. Main problem with such model is that it is slow. It would take days to make any decision. Therefore it should only be used for big questions and power should be left for the workers to decide on their own on smaller issues. The crowd can vote on bigger things, such as main goals and which route to go there, but

the one who executes a task should have the power to decide on smaller unclear matters, such as which shape the product will be. And if the crowd does not like the chosen shape, it is just reshaped later and both designers get compensated. Do first and evaluate later is good as a general rule, and whenever only one option can be chosen, a vote would take place. The lazy consensus and do-ocracy model of open source movements serves as a good starting point for an open business platform as well.

3.8 Funding

In addition to work, a business needs money in order to create its products and to sell those. Funding on an open business project would follow the same methods as in a normal company, with the exception that instead of one CEO being at the other side of the negotiation table, a crowd would be. The crowd would decide whether to take the offer or not via appropriate voting mechanism. Main difference is that since it is a big crowd working on the project, the investor might find it easier to trust the workers to deliver. On a normal start-up the investor pays attention on who the founders are. Even if the product idea is good, but the founders are inexperienced, the investor might choose not to invest. Open business platform removes the founder-factor of the equation, making each project steady quality what comes to the executors.

Other differences are that since it is a big crowd creating the product, the workers themselves might be able to fund the project. If the project has got initiated by customers who need that product, such a funding method would be more than appropriate. One especially interesting form of funding is crowdfunding, which is a popular way to fund risky start-ups nowadays and it would suit especially well for crowd managed projects. The problem with current crowdfunding sites, such as Kickstarter, is that it is only for donating money and not for investing in traditional sense in hope for returns (Kickstarter, 2015). Therefore an open business platform would have to have its own crowdfunding function on the platform itself to be able to share revenue with the investors.

Actually money is not the only intermediary in exchanging goods. A business does not necessary need money, but it needs raw materials and access to tools. Instead of people investing money in a crowdcreated project, which then would be used to outsource for example prototype manufacturing, a manufacturing company could invest in the company by manufacturing the prototype for them. This would be especially useful during quiet times, when the machine works shop would have little work, but would still be paying salaries. What makes this possible is the fact that the work on an open business platform is divided into small tasks, like the prototype manufacturing, and these tasks are open for anyone. Therefore an open business platform offers an easy way for a manufacturer to participate. The companies would be able to make use of the quiet times in production and contribute in building new products and earn piece of revenue.

3.9 Corporate laws and product liability

In order to sell products an open business platform has to establish a company. A crowd cannot start selling products and earning money without a business entity. And a business entity that could be purely crowd managed does not exist. Each type of business entity require individual people to form the decision-making body of the company. In a limited liability company this would be the board of directors. Board of directors can be formed only by one person, CEO is optional, therefore at least one person is needed to represent an open business platform what comes to legal aspects. This person would, in the eyes of the law, be the decision-maker and also carry the responsibility of the company's actions, such as what comes to dangerous and faulty products for example.

This does not mean that a crowd managed open business community cannot happen, but it means that the community has to choose one person to represent them as the director of the company. While the director officially leads the company, there is no reason why the decisions couldn't in reality be made by the crowd, if so has been agreed. And while this person is seemingly the director of the workers, he is still accountable for the shareholders, who again are the workers. The highest decision-making organ in a company is the shareholders' meeting, which is the workers (and investors), therefore the workers always have the highest power at the end.

Since the shareholders hold the highest power in a company, it also means that if a big portion of the company shares has been sold to an investor, this investor has the power to change the company to anything he likes, away from the original model of open business and to a closed company, if he wants. In this sense, law doesn't provide a proper way of getting investors and at the same time reserve the control of a company. The same would be true if instead of one investor, the platform would be crowdfunded. Only way to reserve the power to whom it belongs, the workers, is if the products are funded by the workers, or through presales. Or by investors, but instead of getting shares of the company, they would get only a piece of revenue for example.

An open business platform could open just one business entity, which would be then used for all the products, or it could open a new business entity for each product. Opening a new business entity for each product would be a safer option, since it would decentralize the power and therefore also the risk. Different products require also different tools and different data management methods and might be located in different geographical area, which gives even more reason to found a separate business entity for each product. The open business platform itself could even be just an open source project, maintained by all these sub-companies. The main purpose of the platform is to provide necessary tools for a crowd to manage its work. It could also provide other tools, such as CAD software licenses and data management systems, or those could be bought by each individual company separately.

In addition to acting as the formal representative of the company, the director would also carry the liability of the products. Product liability laws define that a director can be held personally accountable of any harm the products might cause to users. If the product is an electric device and it brakes and causes injuries to users, it is the director who is sued of the accident and who has to represent the company in the court. The same is true if the company breaks some laws during its operations. In this way the director is more than just a public figure and he should be given power to decide at least when a product is ready to be sold and whether it fills all the law requirements and regulations. And he should be compensated for these troubles correspondently.

As a conclusion, corporate laws are not especially built for crowdcreation initiatives, but they do not prevent such model being implemented either. At least one person has to be chosen to represent the company as the director, and some attention has to be paid how many shares are sold and to whom to not to dilute the control of the company from the crowd itself.

3.10 Licenses, patents

Patent laws differ from country to country and so does the patentability of a crowdcreateated invention. The main issue in patenting on an open business platform is the disclosure of the invention before filing a patent. Two types of countries exist: ones that operate an absolute novelty requirement and ones that operate a relative novelty requirement (Takagawa, 2015). Absolute novelty requirement means that the invention cannot be disclosed in any form before filing a patent. Most countries, especially European, follow this method. Relative novelty requirement is in use in USA for example. In this case the inventor can disclose the invention, after which a one year grace period applies, during which the inventor can file a patent. This grace period prevents anyone else from reading the idea and patenting it as their own and makes it possible to publish an idea on a crowdcreation platform. The inventor would post his idea on the platform and at the same time agree on a contract that hands over the IP rights to the crowdcreation platform. Crowd invention platform Quirky for example exploits this one year grace period in their innovations (Quirky, 2015).

In order to patent something on an open design platform in absolute novelty countries, the inventor has to patent the invention before posting it. Or to disclose it only to a few individuals, to a review committee of some kind on the open business platform. A board of experts would review an invention first and decide whether it is worth of patenting or not. If it is, it would be disclosed only to a limited amount of people in order to develop it to a state that it is ready to be patented. After which, it would be made public and open for improvements. The patenting the rights would be handed to the open business company. In this sense, the patent laws do not prohibit patenting on an open business platform in any country, it is just more troublesome in some countries. (EPO, European Patent Office, 2011).

While patent laws in most countries do not totally support crowd created inventions, there some been initiatives to harmonize the patent laws internationally. Currently each country has their own law, although EPC (European Patent Convention) applies to the countries that has signed the contract, but even that is only a set of national contracts rather than one unified contract (PRH, 2015). One of the topics included in harmonization is grace period. It was discussed in European Patent Office's workshop in 2014 (EPO, The economic effects of introducing a grace period in Europe, 2014). Currently some absolute novelty countries do have also a grace period, but it is limited to apply only to the cases where a third party reveals an innovation against the inventors will. The workshop discusses pros and cons of introducing a full grace period. While no direct conclusion was drawn and advantages and disadvantages was presented, it is promising that such a topic is under discussion and if it was to come true, it would be very beneficial for a crowdcreation platform.

Not every invention on a crowdcreation platform would have to be patented. Many products have too short life cycle to be patented. Many ideas do not have any patentable feature in them and cannot be patented, such as services for example. A good strategy that Quirky uses is to patent the long-lasting products, and use first to market tactic for short life time products.

Patents should also be filed only when they are needed. The purpose of a patent is to provide protection against competitors, and sometimes this protection is not needed. Electric car manufacturer Tesla recently opened up all its patents for anyone to use, stating that they had misestimated the need for them in the first place. They did not need protection against competitors, but what they did need is to have more companies developing electric vehicles and competing against the traditional carbon emitting cars, which is their biggest competitor. (Musk, 2014)

Copyrights protect a drawn, written or photographic descriptions of an idea. It includes blueprints and design files and computer code. The important fact of copyright is that it arises automatically without applying for it. And this means that no one else on a collaboration platform can use or modify original poster's design, which effectively prevents any kind of group work. To solve this problem, open source movements use licenses. Licenses are based on copyright law and they can permit others to use the design files while protecting them from unwanted use. An open business platform could also use similar licenses, but since all the results would be posted on the same platform another type of contract could be used also to hand over the copyrights for anyone who works on that specific open business platform. This would be useful so that no other open business platform could copy the files and continue the work without giving revenue for the original developers. However, even if licenses were not used on open business platform, it is useful to take a look at the licenses that open initiatives use.

In general, there are two broad classes of open-source licenses: copyleft and permissive. Copyleft licenses require that the derivative work must also be published under the same license. This means, if someone copies the files, and modifies them, the resulting files must also be open to everyone. Common copyleft licenses include the GNU General Public License (GPL), Creative Commons Attribution-ShareAlike license, CERN Open Hardware License (OHL) and the TAPR Open Hardware License. Permissive license allows a manufacturer to modify the design and use it in their product without publishing the design changes openly. They include the FreeBSD license, the MIT license, and the Creative Commons Attribution license. In addition, non-commercial clauses exist. With a non-commercial clause one can prohibit others from using the design for commercial purpose. Such licenses however are not actually compatible with open design, since they limit the use of the design rather than promotes it. (Mellis, 2015). Different extensions for Creative Commons license are illustrated in Figure 12.
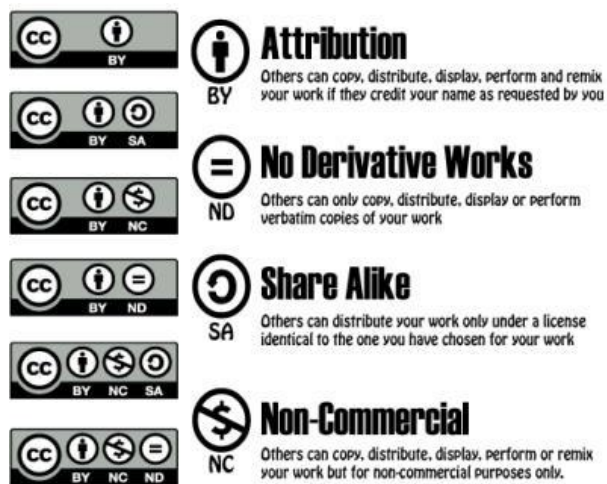

Figure 12 Creative Commons license extensions (Cicero, 2013)

What is specific to licenses used in open design projects, is that they only protect design files and not the physical device itself. Therefore, if someone manufactures the device, then reverse engineers it, he can use these new design files anyway he wants, no matter what the original license says. Therefore licenses are not flawless when used in physical products. For open source software they work fine, since software does not include any physical product.

As a conclusion, patent laws do not currently fully support crowdcreation in every country. However they can be used. Licenses are used to protect the design files. While they don't flawlessly apply for designs of physical products, they are generally used on open design platforms to protect the design and to make it possible for others to contribute. Open business platform could use licenses like open design movements do, or it could restrict the use of the files to happen only within the open business company via separate contract.

3.11 Crowdcreation in services and manufacturing

So far it has been discussed how to use crowdcreation in product designing. But the same model would work in other applications too. For example such as service industry and manufacturing. Next some of these applications are discussed.

Crowdcreation would be highly useful in prototype building where problems occur regularly and where the design faults first time come up. Normally a designer would be called to come to check on the problem and he would solve it. In open business model the problem could be recorded on video and published, and the crowd could post suggestions how to solve the problem. The same goes for manufacturing techniques. It would be highly useful to use a crowd's knowledge on deciding manufacturing techniques and manufacturing tools, instead of the factory personnel alone doing the task. After all, manufacturing possibilities depend highly on the design, therefore the designers should be, and in normal companies they are, closely connected to the manufacturing. An open business platform should make no exception in this.

Workers of the factory can also be paid according to the same principles as the designers, based on tasks with shared revenue. The assembly of a product would be divided into tasks, such as "attach bracket A to frame B". In case of a problem the worker would make a "create a problem report"- task. The task time could be estimated by the crowd afterwards, and the worker wouldn't have to wait for anything. One might think that there is many confusing situations in a factory where a quick solution is needed and there is no time to report anything. But by such a systematic way of working, panic situations would occur less often and problems would be detected in an early state, when there is still time for a proper solution. A task does not have to be created before executing it. A worker can first do the job and then create a task of what he did. And the crowd can evaluate this task afterwards.

Crowdcreation could be used in service industry as well. Such as a restaurant, delivery, hairdresser, translation. Instead of the restaurant owner himself designing the interior, the menu and the location of the restaurant, a crowd would do. A crowd of local people can be assumed to do a better job at choosing the best place for a new restaurant, than one individual. And the same group assumably knows better what they want to eat, rather than the restaurant owner.

Small business owners usually design their business based on their own skills and preferences, which might not be best for the customers. A restaurant owner might only know how to cook Chinese food, while the customers would prefer Greek dishes. A crowd, on the other hand, could exploit a wide range of skills and opinions, being able to concentrate on what is best for the customer.

How would the paid by task-model work in waitressing for example, where a waitress would have to be present a certain amount of hours anyway? Easily, the task would only say "waitress on this and this day, from 8am to 4pm". No subtasks, such as "deliver dish A to table C" would be needed. The tasks would only be divided into as small pieces as it makes sense. The good part about this model is that the waitress would have a clear incentive to do a good work, since her income would depend on it and customers would have a chance to rate her work. The best rated waitresses would also be chosen to the best revenue restaurants, giving a good motivation for a work well done.

Other applications:

Can a crowd for example play badminton? Instead of the player deciding to which corner to aim, a crowd would. Well, maybe. But the first problem that comes is the time limit. There has to be enough time for a crowd to tell the player to which corner they want him to hit. A second problem is that the crowd does not have all the information that the player has. In the case of driving a race car for example. A remotely controlled vehicle could be manufactured and a crowd could drive it, each having their own controllers, and software calculating an average of their steering and acceleration decisions. But what is missing here is the feeling to the road. That is extremely difficult to transmit to the users, and therefore they might not have enough information to decide the right speed for each curve. This is what decentralization is about. The one who has the most information of the problem, should make the decision. In the case of a race car, the driver has.

Interesting test would be to calculate the average of each drivers driving decisions and see if as a group they would drive fastest. Just like a crowd was able to guess the amount of balls in jar accurately by calculating the average. The answer is most likely not, since in a case of runners for example, the average time of the group can never be faster than the fastest runner. So in the case of the race car drivers, instead of an average, the best time of each section should be used and combine those driving decisions to get a best possible time. In ball guessing competition, there was a possibility to guess too high. But in a race, one can never go too fast, and that makes the difference.

In a case of a remotely controlled robot, such as in nuclear or space applications, the model might be more applicable. Since it is remotely controlled to begin with, the crowd would have the same information as a single user. If the task requires high accuracy and a steady hand, calculating an average of multiple users' movements could actually make sense.

As a conclusion the crowdcreation method can be used in applications where enough information of the situation can be transmitted to the crowd, and there is enough time to for a crowd to make the decisions. The model suits well for example for developing a new manufacturing method for a factory, since the crowd has enough time to solve the problem and the factory benefits of a large pool of talents in this case. Then again, a small

manufacturing problem on a busy factory line might be better solved by the worker himself, since it takes time to explain the situation to the crowd and it takes time for the crowd to make the decision. The root cause for the problem however would again be better solved by the crowd to avoid the same problem in the future.

3.12 Business models

Capturing value of a crowdcreated product is not as straightforward as in a closed company. Open source and open design movements are not good examples of this, and they are not even built for generating income for developers. This does not mean that it is not possible. A crowdcreation company can generate revenue just like a closed one. After all, a closed company with its employees can be thought of a small crowd. If that crowd is grown, and the working methods are modified to suit the crowdcreation ideology, as long as it remains closed for the competitors, it can practice the same business models as before. This would mean a modified model of an open-for-everyone -movement. A company that operates under the principles of crowdcreation, but has smaller crowd than the whole world. Such a modified version serves as a good starting point when thinking how to combine crowdcreation ideology and money making.

The problem with open initiatives is that once all the design data is out in the open, it can be copied by anyone and it is difficult to charge for development costs anymore. This is at least the case with open source software projects, since a software can be copied with virtually no cost or effort. It is partly true for open hardware projects also, since electronics can be self-manufactured at home, or easily bought from a third party manufacturer. In difference to open source, open hardware can create some value by manufacturing and selling the products, not enough to cover the development costs though. With other physical products than circuit boards, things might be different. Things like manufacturing techniques, quality and brand might become more important.

When it comes to services, such as running a crowdcreated café or a restaurant, or mechanical design service, things like patents and copyrights do not matter. What does matter is efficient working methods, wisdom in choosing the café location and menu, and smart and large marketing campaign, all things that a crowd can do well, but which cannot directly be copied by competitors.

These in mind, it is clear that openness does not prohibit business. And just because something can be copied, it does not mean that no one can make money with it. (At least what comes to physical products and services, with software easy copying is actually an obstacle). The world is full of products that do not have a patent and in those cases, the one who manufactures, markets and distributes them most efficiently is the winner. And a crowd has all the potential to do this best.

3.12.1 Business models in open source and open hardware

Even though open source software are free, some companies still generate money from them and some programmers do get paid for their work also. For example Red Hat sells open source software. The reason why they can ask money for software that can be downloaded for free, is because they offer technical support in addition to the software (Faletski, 2015).

Some companies donate money for open source projects, just to keep them running, since they need the open source product in their own business. Linux is an example of this. Companies such as Fujitsu, HP, IBM and Intel donate money for the development because their own products need Linux (Linux foundation, 2015). And this serves as an income for at least the creator himself, Linus Torvalds.

Open hardware project Arduino sells their product in addition to just give out free design data. Although the sales do not cover the development costs, it does bring revenue for the manufacturer. The creators of Arduino get money from consulting companies that use their products. From all of this it can be seen that even when the design is free, there is still ways to generate money with it.

3.12.2 Patents and first to market strategy

For products, for which patenting is essential, a crowdcreation company could use the same business models as traditional companies. By patenting innovations it can reserve the right to manufacture and sell its products. Patenting has limitations what comes to public disclosure on a crowdcreation platform, depending on a country. Therefore it might be needed that the invention is patented before publishing openly. Or that it would be disclosed only to a small number of individuals first. Or as mentioned earlier, the size of the crowd would be limited, resembling more of a traditional company in size.

Some limitations for business models do exist though. Such as overcharging spare parts, would probably not be possible, since the information about spare part manufacturer would be openly accessible by anyone. And pricing in general would be more difficult to keep as a secret than in a traditional closed company.

For products with short life time, patenting or other protection methods are too time consuming, such as phone accessories or decoration products. For these products first to market –technique could be used. As this thesis claims, a crowdcreated product has shorter development time than a closed company counterpart, offering a head start in early marketing and selling. This head start can be used to compensate development cost. This is a method that Quirky uses with its crowd invented products.

3.12.3 Branded products and services

For some products, the patented invention is too easy to bypass by another technique and a patent would not provide protection against competitors anyway. And in service industry patents and other protection methods are not even possible. In these cases other methods would have to be used to outrun the competitors.

A good reason for clients to buy from a certain company is that they trust the brand and the quality of the product. And that brand name can be used to charge more than the competitors. A company run by a big crowd has also better chances than a traditional company for offering around the clock support, engineer support in any world location, quick service, fast problem solving, short delivery time, and customized solutions for the client's needs. As seen from open source examples, clients value things like good technical support and are ready to pay extra for such. These can be used as an advantage for a crowd run company.

3.12.4 Importance of manufacturing

As discussed earlier, crowdcreation can be applied to physical manufacturing also. And if it is applied, the manufacturing process is likely to outperform closed model third party manufacturers, leading to better quality products and lower manufacturing costs.

With physical products, the manufacturing process is a big part of the design process also, therefore applying crowdcreation techniques into solving manufacturing issues at the factory holds great promises. Good manufacturing methods are also something that cannot be copied easily. One individual techniques maybe, but to change the manufacturing culture of a company is a different thing. To apply a new technique, such as lean manufacturing method for example, the directors of the company have to learn it, understand it, believe in it, and know how to apply it. A crowd would more likely possess such experts from the start. And by not needing any boss's permission, those experts could apply the techniques on crowd managed company right away. And that gives the competitive edge to an open business manufacturer.

What comes to being first in the market and short delivery times, manufacturing plays a big role. Physical products needs custom made components, which again require tooling design. Usually a supplier does not start tooling design until the design of the product itself is finished. By being part of the community, the manufacturer has a possibility to start tooling design earlier and follow the design changes on the platform. The information flow between designers and manufacturers would be smoother.

3.12.5 Presales and crowd marketing

One way to make sure that the development costs are compensated, is through presales. A project could be funded by the consumers that want to buy the product. In this case the developers would set the price of the product in the correct level in advance, to compensate the development hours as well as the manufacturing costs. Therefore it would require an estimation of the development and manufacturing costs. But such estimations are something that a crowd is good at, as examples from wisdom of crowds prove. It would also mean that the early customers might end up paying more than later customers. But this should be no problem, since it is the case with any new products right now, price is high when the product is first introduced and go down as the time goes by.

Such presales model can be seen in action in real life. It is actually something of a problem for a crowdfunding site Kickstarter. So much that in 2012 they made rules to limit such phenomenon. Users funded mostly products that they wanted to buy, with the expectation that the product would materialize. Which they often did not, due to the low success rate of startups. While Kickstarter did not like the phenomenon back then (now it has loosened the rules), it proved that there is a demand for such. People like to pre-buy products. And this would be a good way to fund crowdcreated products. Pre-buying itself would prove that there is a demand for such a product. And crowdcreation could assure that the product would really materialize after the customers have paid for it. Crowd creation could remove the need for trusting the abilities of the founders to bring the project through. Assuming that a crowd has a large pool of skills and steady performance compared to individual founders.

Another aspect in crowdcreation is its marketing possibilities. A big crowd can in itself already possess future customers and people who know possible customers. And it can spread the word faster than normal startups. This can be seen in action on crowdfunding sites. One reason for companies to use Kickstarter is that this way they generate early interest and publicity for their products. It is also a good source for market research (Leber, 2013). Projects on crowdfunding sites are also more likely to achieve their funding goal, if they receive enough public attention, in social media for example.

3.12.6 Crowdsourced product design

An open design platform could be used by normal companies as means to outsource their product design. For example in the case of a discovery of a new scientific innovation, the companies could, instead of all developing their own independent technology, exploit a crowdsourcing platform to do the work for them. By funding such a crowdcreation project, companies could save in R&D costs and in patent costs and disputes. Such a model could be useful in for example electric car industry. New innovations are needed to bring the battery technology to the level that it can compete with gasoline cars, as well as infrastructure for charging the cars. If the companies in electric car industry would finance a

crowdcreation platform to make the needed innovations and infrastructure, instead of their own R&D departments, they would all benefit of it. It would not matter that they don't own the innovations, if the real competitor is not the other electric car companies, but another technology (Musk, 2014).

Currently companies are already crowdsourcing technical innovations on crowdsourcing platforms like Innocentive. But instead of crowdsourcing only an invention to a specific problem, a whole design process to a finished product could be crowdsourced. Options for payment methods could be fixed reward, paid by hour or portion of the revenue. This would be especially beneficial in industries where a number of companies are trying to solve the same problem, allowing them to concentrate on other aspects of the business.

4. ANALYSIS

To analyze the benefits of the open business method, the model is compared to existing companies as well as open source initiatives. The model could have high impacts of economy as well as what comes to workers' equality. While the benefits might be big, a lot of challenges also exist what comes to feasibility of the model. Some of the challenges that might hinder the implementation of this method are analyzed in this chapter.

4.1 Economical effect

Evening out income differences. An open business company could provide work to people in third world countries, where jobs are generally hard to come by. Since everything is done online and participation is open, people in any country are free to join. Another big difference is that open business initiative would even out income differences between workers in different countries. A participant in a third world country would get the same share of revenue as a participant in a developed country (assuming that the platform does not use location of residence as a compensation criteria). Currently western companies already do provide work for less developed countries, but the salary they pay for the workers is not nearly comparable to western standards. Clothing industry is a good example of this. And the work that is usually outsourced is lower level manufacturing work, not design and development tasks. Open business could fix this flaw.

Finding job and talented workforce. One big benefit of online working is the easiness to find a job that corresponds to one's education. In many smaller cities suitable work places do not exist and people commute far or have to move to big cities. Sometimes even to other countries, if the country of residence do not have such industry, for example space industry. An online platform could solve this problem by providing talented workforce where it is needed and allowing people to specialize to the field that most interests them, rather than what is available in their home town. Crowdsourcing and freelancing platforms, such as Innocentive.com and Elance.com are already doing this, by for example connecting Indian programmers to western companies and crowdsourcing technical problems in high tech industries.

Unemployment. Unemployment happens when there is group of people who at the moment do not know what is needed to be done. There is always more work to be done in the world than there is people to do it. The only challenge is how to distribute this work. In traditional business the people who have new ideas are tight up to their day jobs and don't have an incentive to try out these ideas. And the ones who are unemployed might not have ideas or necessary skills to proceed alone. Open business platform could provide

a way to combine these two, enabling the ones who at the moment do not have anything to do, to use their time usefully trying out new business ideas.

Worker's rights. Another interesting effect that would come out of openness and lack of hierarchy, is the effect it could have on workers' rights. Clothing business is known of using cheap factories in third world countries, who do not pay minimum salary for the workers and where working conditions are dangerous, amount of working hours high and child labor common. Clothing companies do not have a way to track down these problems. Even if the factory they choose is a proper one, there is no way to ensure that this factory does not outsource the work to another factory who then abuses the workers. The problem is that there is always one single individual holding power over the others, such as the factory owner and the employer. Another problem is that information is not open and manager's actions cannot be tracked. If such a factory would be managed by a crowd of people, with no one possessing power over the others, and all the money flow and work tasks would be open automatically, there would be no way for someone to take advantage of the others. The only one who in this case could abuse the workers, is the foreign clothing company paying too low price for the clothes. Which would be solved, once the workers have means to demand higher price.

Pure food and traceability of goods. Some countries, such as China, are known of food scandals. Manufacturers put dangerous chemicals in vegetables to grow them faster, products like coffee and milk can also contain unhealthy chemicals. If a farm would be owned by a crowd, and it would use crowd labor, instead of long term workers, it would be a lot more difficult for anyone to use dangerous chemicals in the field. The benefit of doing so would be lower since one person does not own the whole field, and there would be many eyes reporting such, with no fear of losing their jobs. Transparency and dividing the work into tasks could also provide a way to track down a patch of vegetables from a supermarket shelf to the field it origins from. Since, if every work task needed to transport the vegetables from the field to the specific shelf would be written down, it provides a traceable chain of documentation that could be used to prove the quality. Same tracking method would work in clothing industry. In this way open business method would provide a new business model, quality inspected by the crowd.

4.2 Comparison to traditional companies and open initiatives

One of the main benefits of open business compared to closed companies is faster product development and the ability to include customers into the product development. Customers would act as any worker on the site, giving development ideas and participating in designing, and they would get their share of the revenue at the same time. This would provide a good motivator for customers to join the process. Product development would be faster already because it could happen around the clock, based on the time differences. And because a large pool of workers would be participating at the same time.

Other benefits are higher new product success rate. If development process is open and includes customers, it offers much better chances to separate good ideas from bad, than normal market research methods done by start-ups.

From a workers perspective, the flexibility is a key feature. One can work from anywhere and at any time. And a worker can work on the tasks that he finds interesting, which would mean that each task is done by someone who has enthusiasm for it, rather than by a person who works for it only for the money. A worker could also have a safer income, since the income comes from multiple projects instead of just one company. Most of the month could be used to work on risk free established projects and some part on riskier start-up projects, which would gain higher income if successful.

Even now, startups don't have to be alone with their problems. Some startups use advisory boards. This is a board of industry experts that the startup founder has himself selected and asked to join help in his startup. And as compensation, the advisor gets a small percentage of the shares. In a way, this addresses the same issue in entrepreneurship as open business model. It adds expertise to the start-up that compensates the lack of expertise in the company. The advisor tells how should be proceeded and the entrepreneur executes the tasks needed. One difference is that the entrepreneur still has the control whether to follow the advisors advice or not. They might still go against the advice and make mistakes. Another disadvantage is that the advisor can only give general advices and it is still up to the entrepreneur to know how to execute the advices. A third difference is that it is still just one or a few advisors giving advices, not a big crowd. Open business model relies on harvesting good ideas via crowdsourcing. And the crowd can contain future clients, who can tell what they want, and industry experts who might be more suitable for the give problem than the advisor that the start-up founder has chosen.

Open business differs from open source movements in a way that open source movement participants are usually programmers and designers and they only concentrate on designing the product and not so much on other aspects of the business, such as sales and customer support. The biggest benefit that open business can bring to open source ideology is the method to divide income between workers, which offers a way for the workers to get compensated. And once compensated for their work, participants could concentrate all sorts of tasks and not only on the ones that are most fun. So far open source initiatives have not needed such a revenue sharing method, since the product itself is not sellable, but physical products are, and that makes the biggest difference between open source and open business. Other difference between open business and open source is that open source movements release design files for anyone to use via open licenses, even for people who do not work for that open source movement. The purpose of open business on the contrary is not to produce products or design files that anyone can use for free. The purpose is to crowdcreate products, but at the same time to bring revenue for the designers. Therefore open business design files should not by default be open for anyone to use, but only to workers who have registered on the specific open business platform. This way

anyone can still participate, but no one has the right to copy the files and sell the product without original creators getting their share of the revenue. This would of course be difficult to control, especially if the files are open for everyone to view. But the idea stands. And in a modified version of open business, if the crowd would be limited to for example 100 workers, and new ones could only join by invitation, it would be possible to control and protect the design files, just like in a closed company, and still work according to the principles of crowdcreation and revenue sharing.

4.3 Open business as a learning platform

Employers value experience in employee selection more than anything else. Experience equals knowing how to do something and it can be obtained by working in many companies and projects, each project teaching something new. The bad part about learning by doing is that it is slow. It takes many years to gather the required experience. Therefore it is not efficient method to learn new things.

One could also learn from others, by following how others do their job. To become a project manager, one can follow how the current project manager does his job and learn from that. The only problem is that others' work is not public. Project meetings take place behind closed doors and information is exchanged by emails, not accessible to third parties. On an open business platform all information exchange would be visible to everyone. One could read through a discussion in which a material was chosen for a product. See how and why that material was chosen and learn from that. Or learn how to make strategic decisions, which things affect when pricing a product, how an experienced customer servant dealt with a difficult customer, how a good marketing letter was written by a marketing specialist. A project itself might take weeks and months to finish, but to read through development discussions only takes hours. Therefore learning by reading other, more experienced peoples' discussions or project diaries is fast and effective method to learn new.

A method to learn from others' experience already exists though. Startup founders write blogs what went wrong with their company, authors write books about famous companies' success stories, the internet is full of articles about marketing professional giving tips on marketing strategies. But there is a difference between reading such articles and following how a marketing professional actually does his job. Instead of just knowing which decision he made, one would learn which things he considered when making the decision. It would be interesting if Facebook's early development discussions were recorded. One would learn how exactly the founders decided to implement the very features that made the site a world success and more popular than other similar sites. Some of NASA's project meetings have been recorded in case of accidents. From these recordings one can easily see how decisions were made and where mistakes happened. These kind of discussions offer a whole new kind of study material. If constructed in a concise and clear form, it would be easy to read and it could serve as study material in schools also.

Another example of how sharing information can be beneficial is, if one company writes a document, for example a design handbook for the designers in the company, another similar company would greatly benefit of seeing this document. Instead of each having to make their own design handbook, they can copy it from the other. Then maybe even improve it and both would benefit. Good working practices would spread faster, if it would be possible to copy from others. If there is tens of companies in similar business, a big amount of work is saved.

There is another impact that an open business platform could have on schools. If an open business platform could provide people access to a job that fully corresponds to their education, no matter where they live, and if one could always find work even in the narrowest niche of profession, the consequence would be that students would not have to study a broad range of subjects anymore. Instead, they could concentrate on one very specific field. Currently, a mechanical engineer, as an example, has to know a bit of plastics, piping, welding, gluing, hydraulics and friction, etc. since one never knows which skills the next employer will require. In future, an engineer student could only study joining techniques, or friction, or some other very narrow field. And then this engineer would jump from a product to another on the open business platform, designing only the joining method used in each product. The research field of joining methods would advance greatly and all designed products would be on a whole new level of sophistication, since each feature would be designed by a dedicated expert of a specific field.

4.4 Challenges in open business method

Main challenge in open business method is how to coordinate work. How to split tasks and who does what. Although open source projects prove that a crowd can effectively create products, they still have some sort of management structure. The easiest way to fight this problem would be to start with a hierarchical management structure and gradually introduce crowd decision procedures. Another key issue is would the crowd take responsibility of the product, if no one is directly in charge. The motivators of the participants should be further studied and tested. In open projects there is usually always someone who is still the lead figure and takes the responsibility. On a crowd platform the main motivator is the piece of revenue if an unpleasant task gets done. Would tasks, such as solving one customer's cumbersome problem, get done? In current companies, such tasks get done because they are someone's responsibility. It is not sure would those tasks get done on an open business platform. One way to tackle this problem would be to introduce roles, areas of responsibility, and while the pay still is task based it would be lower or higher depending on how the tasks overall would have been performed.

One challenge in open business method might be the lack of physical presence. People might not be so motivated and committed, if working remotely. Y Combinator is a seed funding organization for start-ups, who in addition to funding, also couch the founders. They require the founders to move near them during the couching period, and the reason

for this is that the one time they tried, it just didn't work remotely (Y Combinator, 2015). This means that people need presence to get motivated and to perform well. (Although open source projects prove otherwise, they are remote and they seem to work).

Especially when working with physical products, it is difficult to design a product without ever seeing it. 3D CAD software do not display perspective correctly and depth visualization is very poor. 3D glasses do not help the problem either. Such tasks as choosing wiring routes on a big product, are ten times easier if one can see the product. Therefore it would be highly useful if the designers could also visit the manufacturer and see the product. If an open business product would become popular enough, some kind of showcase studios could be arranged to each country, as well as co-working spaces. This would help with the problem of physical presence.

5. CONCLUSIONS

The idea of a crowd creating products collaboratively has been established in the open source and open design movement already. The new things that this thesis introduced were the work compensation method and business models. Workers can be compensated by dividing the workload into tasks and evaluating each task's value with the help of the crowd and then sharing the revenue based on these evaluations. By having a compensation method, open movements do not have to work for free, and they can gain revenue based on the same business models, patents etc. that are used in traditional closed companies. All that is needed is a business identity for doing business and a platform and tools to create tasks and publish results. Such things as decision making mechanism for a crowd or work coordination already exist in open source initiatives. These methods could be used as it is, or in addition to those, some new aspects were introduced, which base on the concept of wisdom of crowds. Especially the crowd evaluating method for task compensation exploits this theory.

The implementation of the method could follow many routes. As seen in Case studies-chapter, many of the features are already in use, such as crowdsourcing solutions to technical challenges (Innocentive), dividing work into small tasks (Mechanical Turk), sharing the revenue among workers (Assembly), crowdsourcing new product ideas (Quirky) and crowdcreating new products (open source projects). Only thing missing is an initiative that would combine all of these. It is worth of noticing that the model comprises of three different parts, of which some can be implemented without another. These parts are:

- Compensation method
- Work breakdown into tasks
- Self-managing crowd

The model of sharing revenue based on amount of work done can be implemented without a self-managing crowd, with a normal hierarchical management structure. It does require work to be divided into tasks though. A crowd can also be used for decision-making without the other two parts. Since one part can be implemented without another, the model can be taken into use gradually, step by step. Just like open source projects sometimes start as dictatorships and then move towards mediocracy, an open business project could start as a normal closed company and gradually introduce the revenue sharing model and the crowd decision-making procedures. All that is needed is a record of worked hours in order to change the compensation model on the way. It is also clear that good method for self-managing crowd cannot be implemented at once. Therefore it is wiser to start as hierarchically managed project and introduce and test crowd decision-making tools on the way. A good way to experiment would be to, instead of one CEO, have twenty and then create smart tools for them to cooperate together.

Traditional closed companies could also exploit some of the methods discussed in this thesis. Internal prediction markets are used as a help for decision making by many companies. Solutions to technical problems could be crowdsourced either externally or just within the company. Companies could open up some of their information with other companies with similar but not competing products for a common benefit, such as design handbooks, industry reviews or project management methods. Or create collaborative data management projects. Development discussions within a company could be opened to the workers to promote a learning experience. Hierarchy in a company could be reduced, and compensation models could rely more on company success. A company could even let their designers design new products according to crowd creation principles, freely with no salary, but with the company tools and a share of that product's revenue.

6. REFERENCES

Adams, I. (2013, November 21). *OptimizationGroup*. Retrieved from http://blog.optimizationgroup.com/crowdsourcing-vs-co-creation-whats-the-difference

Anand Kulkarni, M. C. (2011). *Turkomatic: Automatic, Recursive Task and Workflow Design for Mechanical Turk*. Association for the Advancement of Artificial Intelligence.

Aniket Kittur, B. S. (2011). *CrowdForge: Crowdsourcing Complex Work*. Pittsburgh: Carnegie Mellon University.

Apache. (2015, April 15). Retrieved from https://www.apache.org/foundation/how-it-works.html#infrastructure

Assembly. (2015, April 15). Retrieved from https://assembly.com/guides

Boundless. (2015, May 27). *Boundless*. Retrieved from https://www.boundless.com/management/textbooks/boundless-management-textbook/decision-making-10/managing-group-decision-making-81/advantages-and-disadvantages-of-group-decision-making-388-5156/

Brabham, D. C. (2013). *Crowdsourcing*. MIT Press.

Brief, B. (2013, December 3). *Bain & Company*. Retrieved from http://www.bain.com/publications/articles/how-group-dynamics-affect-decisions.aspx

Chesbrough, H. (2003). *Open Innovation: The New Imperative for Creating and Profiting from Technology*. Harvard Business Press.

Cicero, S. (2013, October 9). *Open Electronics*. Retrieved from http://www.shareable.net/blog/how-to-choose-an-open-source-hardware-license

Cuban, M. (2012, January 10). *Entrepreneur*. Retrieved from http://www.entrepreneur.com/article/222524

EPO. (2011, September 5). Retrieved from European Patent Office: https://www.epo.org/learning-events/materials/inventors-handbook/protection/patents.html

EPO. (2014). *The economic effects of introducing a grace period in Europe.* Munich: European Patent Office.

Estelles-Arolas, E. F.-L.-d.-G. (2012). Towards an integrated crowdsourcing definition. *Journal of Information Science*, 1-14.

Faletski, I. (2015, June 4). Retrieved from Harvard Business Review: https://hbr.org/2013/01/yes-you-can-make-money-with-op

Fallon, N. (2014, January 2). *Businessnewsdaily*. Retrieved from http://www.businessnewsdaily.com/5711-angel-investor-tips.html

Gabriel, R. G. (2005). *Innovation happens elsewhere.* Elsevier Inc.

Gardler, R. (2011, February 22). *Oss Watch*. Retrieved from http://oss-watch.ac.uk/resources/communitytools

Grier, D. A. (2015, April 25). *Dummies*. Retrieved from http://www.dummies.com/how-to/content/understanding-the-five-types-of-crowdsourcing.html

Griffith, E. (2014, September 25). *Fortune*. Retrieved from http://fortune.com/2014/09/25/why-startups-fail-according-to-their-founders/

ISM. (2015, May 27). Retrieved from https://sites.google.com/site/crowdsourcewiki/home/forms-of-cs/crowd-creation

Keifer, S. (2010, May 8). Retrieved from https://outsideinmarketing.wordpress.com/2010/05/08/four-types-of-crowdsourcing/

Kickstarter. (2015, April 25). *Kickstarter*. Retrieved from https://www.kickstarter.com/?ref=nav

Leber, J. (2013, February 11). Retrieved from Technology Review: http://www.technologyreview.com/news/510616/backers-with-benefits-why-companies-are-outsourcing-to-kickstarter/

Linux foundation. (2015, June 4). Retrieved from Linux foundation: http://www.linuxfoundation.org/about/faq

Lynch, A. (2015, April 25). *DesignCrowd*. Retrieved from http://blog.designcrowd.com/article/202/crowdsourcing-is-not-new--the-history-of-crowdsourcing-1714-to-2010

McLeod, S. (2007). *SimplyPsychology*. Retrieved from http://www.simplypsychology.org/milgram.html

Mellis, D. (2015, May 8). Retrieved from OSHWA: http://www.oshwa.org/faq/

Musk, E. (2014, June 12). Retrieved from Teslamotors: http://www.teslamotors.com/blog/all-our-patent-are-belong-you

PRH. (2015, June 2). *PRH*. Retrieved from https://www.prh.fi/fi/patentit/patentointi_ulkomailla/eurooppapapatentti.html

Quirky. (2015, June 4). Retrieved from Quirky: https://www.quirky.com/about/ip

Ries, E. (2011). *Lean Startup.* New York: Crown Business.

Ross Gardler, G. H. (2013, November 13). *Oss Watch*. Retrieved from http://oss-watch.ac.uk/resources/governancemodels

Sundelin, A. (2009, 6 18). Retrieved from The Business Model Database: http://tbmdb.blogspot.com/2009/06/what-is-open-business-model.html

Surowiecki, J. (2005). *The Wisdom of Crowds.* New York: Anchor Books.

Takagawa, D. H. (2015, June 2). *Invention disclosure and patent grace periods: How disclosing your invention before filing a patent application can severely limit your ability to obtain a patent*. Retrieved from Oyen Wiggs Green & Mutala LLP: http://www.patentable.com/invention-disclosure-and-patent-grace-periods-how-disclosing-your-invention-before-filing-a-patent-application-can-severely-limit-your-ability-to-obtain-a-patent/

Thompson, C. (2008, 10 20). Retrieved from Wired: http://archive.wired.com/techbiz/startups/magazine/16-11/ff_openmanufacturing?currentPage=all

Venkat Ramaswamy, K. O. (2014). *The Co-creation paradigm.* Stanford: Stanford business books.

Y Combinator. (2015, June 4). Retrieved from Y Combinator: https://www.ycombinator.com/faq/