Utilising web analytics in the agile development of e-commerce sites – a software developer's perspective

Master's Thesis in Technology
University of Turku
Department of Computing
Software Engineering
2021
Joona Juusti

UNIVERSITY OF TURKU
Department of Computing, Faculty of Technology

JOONA JUUSTI: Utilising web analytics in the agile development of e-commerce sites – a software developer's perspective

Master's Thesis in Technology, 79 p., 2 app. p.
Software Engineering
May 2021

E-commerces have gained popularity exponentially since the dawn of the world wide web. To stay competitive, increase revenue and make their e-commerce site as good as possible, organisations have begun to utilise web analytics to make the development of the site data driven.
Agile software development has often been the desired way of building software in the last decades. Organisations are in increasing numbers trying to move to a more agile way of working in order to build better software.
In this thesis we examine how the web analytics of an e-commerce site can be utilised as well as possible in agile software development teams. We examine this web analytics process especially from the point of view of software developers in these teams. The research around this topic was conducted as qualitative research by interviewing four different software developers each having experience in developing e-commerces. Drawing from their experiences and opinions, we formulate some observations and guidelines for how organisations can potentially improve their efficiency in utilising web analytics as a part of their development process.

Keywords: e-commerce, web analytics, agile development

Contents

1 Introduction

During last decades, e-commerces have been growing rapidly. Globalization, development of information technology and its ubiquity at the developed and developing economies has caused changes in consumer behaviour. In recent years, e-commerces have been growing from 10 to 20 % annually. As a consequence of the covid19-pandemic, e-commerce sales increased approximately 32.4 % in 2020. E-commerce has revolutionized traditional markets and services because it makes it possible to shop almost anything anywhere at anytime. The popularity of e-commerces shows now regression but vice versa companies are willing to invest money in them and develop them even better. Obviously, the better the e-commerce shop is designed and implemented, the better it can be used to make more sales and control customers' purchasing behavior. Thus, the company generates more profit for its shareholders and fulfills its ultimate mission.

The aim of this research was to study agile software development and the utilisation of web analytics in e-commerces. The main interest was to examine this from a software developer's perspective and find interfaces between agile methodologies and web analytics as a driving factor for development. For instance, how can software developers harness the data of customer behaviour to develop the e-commerce even more profitable.

Three research questions (RQ) were formed in the beginning of the research. The purpose of those was to give proper frames and steer the study. However, the third RQ was almost ignored and the study focused mainly on RQ1 and RQ2.

RQ1: How can web analytics data of an e-commerce be utilized during its agile de-

velopment?

RQ2: What kind of roles can be discovered from utilizing web analytics during the development of an e-commerce?

RQ3: What type of skills should an organisation have in order to utilise web analytics in the development of an e-commerce?

The answers for these questions are presented mainly in Chapters 5, 6, and 7. Chapters 2, 3, and 4 are mostly background chapters which review the literature around the topics in question and give a proper premise for the actual research part.

In chapter two the agile software development methodologies and the various roles of agile teams are introduced. It is valuable to understand what structures and practises the work is based on currently. Chapter 3 introduces the present state of e-commerce and growth during the past years, different types of e-commerce and potential future trends. Chapter 4 deepens into the web analytics of an e-commerce in practise. The different methods to collect data, such as server log files and page tagging are presented there. The overall web analytics process and A/B testing are also reviewed. Finally, the hypotheses for the actual study part of the research are presented.

In chapter 5 the conducted study is covered. The study was implemented as a qualitative research by conducting semi-structured interviews with software developers experienced in developing e-commerces. The goal of the interviews was to validate the research hypotheses and also gain insight into how web analytics is used in practice. This chapter includes comprehensively all the information related to the execution the study and the achieved results.

Chapters 6 and 7 are analysing the conducted study even more deeply. They also discuss the shortcomings and potential improvements of it. The future prospects and needs regarding the following studies related to e-commerce are also reflected upon.

2 Agile Software Development

In this chapter we will explore agile software development by first going through its background and principles, after which we will examine more in detail some methodologies and roles related to those. By going through what agile development in general means, we can better understand the context in which we will examine web analytics usage. This will enable us to answer RQ1 comprehensively. In Section 2.3 we will go through some existing roles that can be found within agile development teams. The analytics usage will be reflected against these roles later on which will form the basis for answering RQ2.

2.1 Agile movement

Agile software development differs from traditional software development by focusing on people and social interactions. Instead of the traditional plan-driven development, software is created in an iterative and incremental manner focusing at a time on a small set of tasks prioritized by the customer.

Traditionally, work began with documenting a complete set of requirements. This initial step of requirements documentation started to spark frustration in software developers starting in the mid-1990s. This was due to the unstable nature of technology and the business environment. This shifting during the project led to projects getting out of date and customers being unable to provide definitive requirement documentation beforehand. [1]

As a result, practitioners developed methodologies and practices to embrace, rather than reject, higher rates of change. These methodologies were developed on three dif-

ferent continents: the Dynamic Systems Development Method in Europe; Feature-Driven Development in Australia; and Extreme Programming, Crystal, Adaptive Software Development, and Scrum in the US. All of these methodologies were developed independently but their practices and philosophies share fundamental similarities. [1]

The Agile movement emerged in February, 2001. Seventeen people heavily associated with the so-called ”Light methodologies”, met over at a ski resort to eat, talk, ski, relax and try to find a common goal. The concrete result of that two-day meeting was the ”Manifesto for Agile Software Development”. All of the attendants felt the need for an alternative to the prevailing methods of that time. Methods which were often heavyweight and documentation-driven software development processes.

The attendants all shared a set of compatible values. These values were based on trust and respect for human beings and promoting organizational models based on people, collaboration, and building the types of organizational communities in which they would want to work. One attendee jokingly said that the Manifesto was a ”mushy” statement. And in the end, it is not that far from reality since Agile Methodologies are first and foremost about values and culture.

The four underlying values that the Manifesto states are: [2]

- individuals and interactions over processes and tools
- working software over comprehensive documentation
- customer collaboration over contract negotiation
- responding to change over following a plan.

The Manifesto states that both the items on the left and the items on the right are valued, but it asserts the dominance of the ones on the left. From these four values stem the twelve principles of the Agile Manifesto. These principles are crucial for being able to be agile. All of the agile methodologies are built around and expand on top of these principles: [3]

P1 Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

P2 Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

P3 Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

P4 Business people and developers must work together daily throughout the project.

P5 Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

P6 The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

P7 Working software is the primary measure of progress.

P8 Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

P9 Continuous attention to technical excellence and good design enhances agility.

P10 Simplicity–the art of maximizing the amount of work not done–is essential.

P11 The best architectures, requirements, and designs emerge from self-organizing teams.

P12 At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Eliminating variations from processes was the goal that traditional process management aimed for. Some means for achieving this were continuous measurement, error identification and process refinements [4]. The assumption behind this approach was that errors are the cause of variations. This assumption still holds truth in some scenarios. In

addition to errors, the changes in the external environment cause variations too. These changes are basically impossible to eliminate with internal means which leads to embracing the change instead of avoiding it completely. According to Barry Boehm's life cycle cost differentials theory, the cost of change grows throughout the software's development cycle which is visualized in Figure 2.1. Figure 2.2 shows how embracing the change, in practice, means driving down the cost of responding to it.

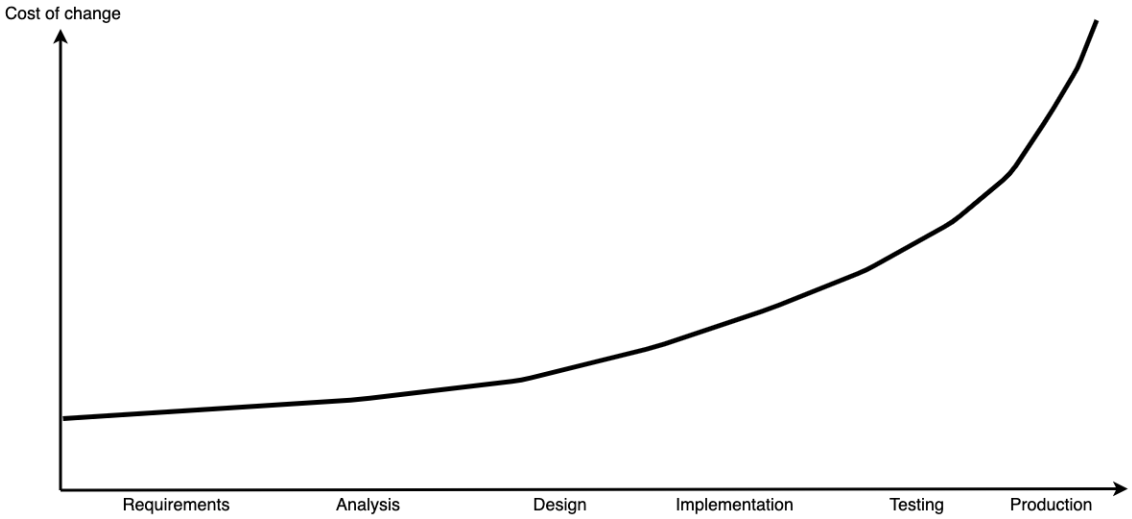

Figure 2.1: Cost of change throughout the different phases of a waterfall project

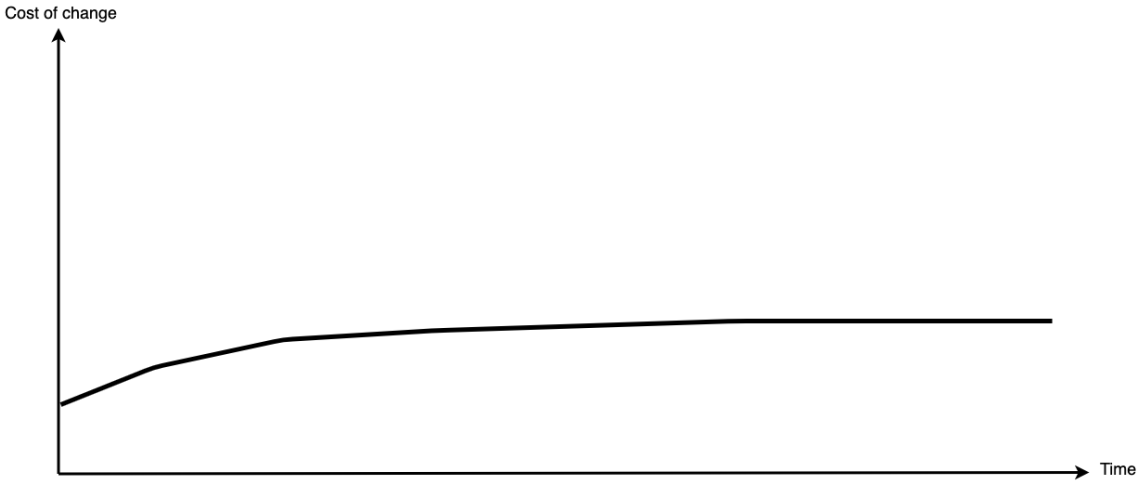

Figure 2.2: Cost of change throughout the different phases of an agile project

Agile development has meant that there has been a shift from document-driven de-

velopment to trying to satisfy the customer's needs at the time of delivery, not at project initiation. Changes often occur within a projects' life span, even within relatively short projects [1]. The changes that are outside of the development team's control, like changes in the requirements, scope and technology, are inevitable. Accepting this, the question becomes how to handle the unavoidable changes rather than avoid them completely. [4]

Even though agile methods center around people and communication, they also stress the unforgiving honesty of working code. It tells the developers and stakeholders what they really have in front of them, as opposed to promises as to what they will have in front of them. The working code can be shipped, modified or scrapped, but it is always real.

Agile methods try to utilize people as effectively as possible. Allowing people to communicate directly with each other face to face instead of through writing and reading documents, they can share ideas faster. Talking with customers, developers can resolve obstacles, adjust priorities and think of different paths forward in a completely new manner compared to them not working together.

2.2 Agile methodologies

The life cycle of a software development project has evolved through times. During the first wave of software development, the ad hoc wave, working software was the measure of progress. But due to the software development industry being very immature at that time, the cost of making major changes was impactful. This led to the rise of the second wave of software development in which the change was tried to eliminate altogether. If this had been successful, the cost of the change would have been reduced due to changes being close to nonexistent. The third wave, which has brought with it the agile software development, seeks to in fact reduce the cost of change itself and has seen the return of working software as the measure of success.

One big difference between agile methods and the old waterfall model is that agile

development is cyclic and iterative. In the waterfall method, the project was split into different phases, such as analysis, coding and testing. These phases had each their own roles assigned to them which produced a document of some sort at the of the phase and passed it along to the group in the next phase [5]. For the aforementioned phases, the documents could have been, for example, specification, design document or code. The different phases were finished consecutively one after the other. This resulted in the different groups ending up in silos which limit the communication and collaboration.

Iterations are an important practice in agile development and those can be seen essentially in one form or another in all agile methodologies. Each iteration usually includes most of the steps that happened in waterfall projects only within itself. Iterations force a shorter feedback loop than the traditional up front methodologies. Agile methodologies have a principle assumption that plans are not predictable [5]. By planning only the upcoming few weeks that make up the next iteration, a lot of possibly unnecessary work can be avoided. The goal of each iteration should be to have new code released.

Many of the agile principles are suited or even demand an iterative and incremental approach to software development. Being able to deliver software continuously (P1, P3) means that the software must be functional at the time of delivery. This naturally drives development to be feature driven which means that the software is built feature by feature and each of these features should work as an individual increment to the already existing software. Iterations are also suited for adjusting the priorities continuously. This is important because the value of different items can, and often will change, and new requirements and items come up during the project life cycle. What was valuable one day might lose its value the next day due to the changing environment and vice versa. Because the scope of the development only covers the current iteration, new items can be picked up very fast, at the latest in the beginning of the next iteration. By adjusting the priorities continuously, it can be ensured that the agile team is producing, not only valuable (P1), but in fact the most valuable software at the given time.

Many methodologies try to provide an inclusive ruleset which tries to cover all the things that could be done under any situation. Agile methods differ from this approach fundamentally. They provide generative rules which cover only the minimum set of things that must be done under all situations. The fundamental difference between these approaches is that inclusive rules rely on someone else to create the practices and conditions for all situations. Using a generative ruleset, however, means that the team members and their creativity to find solutions to problems is relied on in these methods. [4]

Extreme Programming (XP) was one of the earliest agile methodologies. Being an agile method, it is built on the values of communication, feedback, simplicity, courage, and respect. Unlike Scrum, it does not embrace having certain ceremonies regularly. Instead, at its core, it is a set of twelve different software development practices which, when used together, have proven to be very beneficial for software development teams. These practices had existed even before XP but this methodology brought all of them together for the first time. Instead of trying to find out the set of all practices that could ever be needed, XP provides the simplest set of practices that could possibly be needed and tries to eliminate the need for any more practices [6].

The twelve practices of XP are: [7]

- The Planning Game. The scope and timing of the next iteration is determined by customers based on technical estimations. No other functionality is implemented in the iteration than that which was agreed upon.
- Small releases. The system under development is put into production as soon as possible. New releases are then made on a very short cycle.
- Metaphor. Development is guided by metaphors which define the shape of the system.
- Simple design. The system should be as simple as possible at every moment. Any unnecessary complexity should be removed once discovered.

- Testing. Automated unit tests are written continuously by programmers. These tests should run flawlessly before development can be carried on. Functional tests are written by customers. All of these tests increase the confidence in the
- Refactoring. Programmers improve the quality of the system without changing its behaviour so that all the tests keep passing.
- Pair programming. Production code is written by two people at one machine.
- Collective ownership. Every programmer can improve any code anywhere in the system at any time.
- Continuous integration. New code is integrated into the system often and every time the system is built and all tests must still pass.
- 40-hour week. Overtime should never be worked for two consecutive weeks. Each week should include no more than 40 working hours.
- On-site customer. A real, live user sits with the team to be available to answer questions full-time.
- Coding standards. All code should conform to rules which highlight the importance of communication via code.

It is important to remember that these practices are just rules. Every team is unique and what works for one team might be different to what works for some other team. That is why the team can change these rules if they see the need for as long as the whole team agrees on the change [8].

Scrum is another agile software development methodology. The aforementioned XP focuses heavily on programming practices whereas Scrum is more concentrated on software project management. In Scrum, the iterations through which a project is carried out, is called a Sprint. The outcome of each Sprint should be a potentially shippable increment of the product.

The main components of the Scrum framework are roles, ceremonies and artifacts. The ceremonies are events which are held continuously throughout a Scrum project's lifecycle. Each of these ceremonies have a certain specific agenda and reason. One of the ceremonies, the Daily Scrum, happens daily but the rest of the ceremonies take place once per Sprint. The iterative nature of Scrum is depicted in Figure 2.3. It shows how each of the ceremonies is repeated on a Sprint-by-Sprint basis.

The artifacts of Scrum are the Product Backlog, Sprint Backlog and Burndown Chart. The Product Backlog is an overarching list of all of the different stories and tasks of the project. At the start of each Sprint, some of these stories and tasks are refined in the Sprint Planning meeting and picked into the Sprint Backlog. The Sprint Backlog will then hold only those tickets that will be implemented in the current Sprint. In Figure 2.3, it can be seen how Sprint Backlog items form the starting point of each Sprint. The last artifact, Burndown Chart, visualizes the correlation between the advancement of the team and the amount of work still left to do.

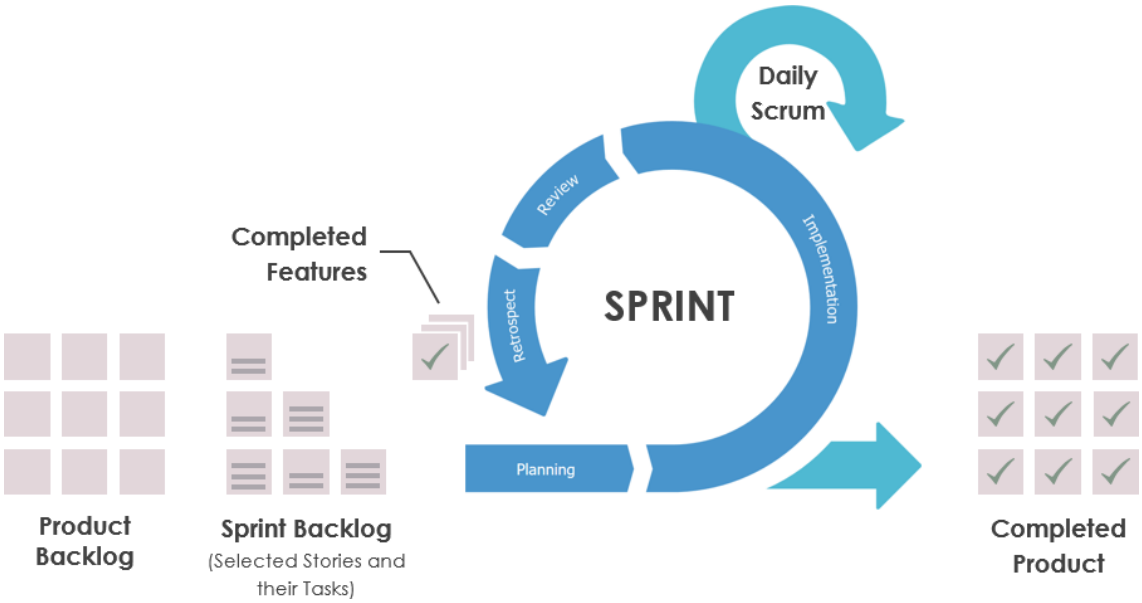

Figure 2.3: High-level overview of Scrum's Sprint structure [9]

2.3 Roles in agile teams

The roles in agile software development teams can be categorized in two ways. The first way being differentiating roles that are tied to specific individuals. For example, the product owner and scrum master in Scrum are such roles. The seconds categorization of roles is roles which are tied to the behaviour and actions of the team members. These roles in the latter categorization are not necessarily tied to specific individuals but they can be.

Self-organizing teams have become one of the hallmarks of agile software development. Even though they are strongly associated with software development nowadays, the concept had existed long before this. They can be reviewed from many different perspectives, e.g. organizational theory, knowledge management, socio-technical systems and complex-adaptive systems. [10]

In agile software development teams, the team members are considered as equals without a strict hierarchy between them. The team is empowered to make decisions by itself with democratic means. The democratic means are better suited to smaller teams which is one of the reasons why agile methods work better in smaller numbers. The traditional project manager does not exist in truly agile teams. However, this does not mean that management would not exist in agile teams. It manifests as more facilitative and coordinating than micro-managing.

Hoda et al. recognized six different self-organizing roles in agile software development teams [10]. These six roles are mentor, coordinator, translator, champion, promoter and terminator. Organizational roles, like developer, tester or business analysts, usually have boundaries which are adhered to in non-agile teams. In agile teams, the team members often operate also outside of their organizational role, especially when organizing themselves. Similarly, the six informal roles are such that one role can be fulfilled by many people and one person can also fulfil one or more roles.

A Scrum team consists of three roles: the Product Owner, the Scrum Master and the

Development Team. All responsibilities related to management are split among the three roles [11].

2.3.1 Scrum Master

The Scrum Master takes up the position typically occupied by the conventional project manager [12]. However in practice, the Scrum Master is not similar to a line manager but instead acts as a servant-leader and a coach [13]. The Scrum Master's responsibilities include taking care of the correct usage of Scrum's practices, values and rules in the project [14]. They should also ensure that the Scrum is implemented in a way which fits the organization's culture without compromising the expected benefits. If everyone within the project is not familiar with Scrum, the Scrum Master should be teaching it to them. [11] The Scrum Master should also clear all impediments that might block or prevent the Development Team from working as productively as they can while protecting the team from external noise [14].

2.3.2 Product Owner

The Product Owner role is often the most difficult one and it falls onto a single person in a Scrum team [15]. One of the responsibilities of the Product Owner is product backlog management. Even though the development team can participate in backlog management, the Product Owner still has the final say over items related to it [14]. Through keeping the backlog in a prioritized order, the Product Owner can ensure that the Development Team works on the most valuable items constantly. This optimizes the work done by the Development Team and produces the best return on investment. The Product Owner should represent the concerns of the different stakeholders of the project and its resulting product [11].

2.3.3 Development Team

The Development Team consists of individuals who form a team which is self-managing, self-organizing and cross-functional. The Team's responsibilities include figuring out how an item on the product backlog should be turned into an increment of functionality in an iteration. [11] The Team's responsibility is also to implement that functionality and deliver a potentially shippable increment of the product at the end of the iteration [16]. The Team members manage and plan their job themselves to maximize the Team's productivity. The Team is completely responsible for executing the work that belongs to it. [17] Individual Team members can have different and specialized skills but the members are collectively responsible for the success of each iteration. In fact, the Team should have all the necessary skills to be able to create product increments. However, there should not be subteams nor titles in the Team regardless of the domains addressed by the work performed by the individuals. [16]

2.3.4 Mentor

Adopting agile methodologies and moving towards self-organization can be difficult. A mentor is the member of the team whose significance is emphasized during the initial stages of embracing agile methodologies. At this stage, familiarizing the team with agile software development is an important task of a mentor. This role is often played by an agile coach like the Scrum Master in Scrum. This holds true in relatively new teams but in more mature teams, the role of a mentor can be played by any member of the team with enough experience in agile software development. And there is no need for a full-time mentor in an experienced team and the role comes up

Agile practices can be simple to grasp for many team members but carrying those out on a day-to-day basis is challenging. The mentor provides the team members assistance and support to make the team confident with implementing agile practices. Especially in the beginning, some team members may have misconceptions about agile software devel-

opment. The mentor is the one who irons out these misconceptions while simultaneously encouraging the team members to voice their concerns and opinions.

It is important to adhere to agile principles and values continuously. Team members with little to no previous experience of software development are able to adopt agile practices easier than other members of a team. Experience of working in a non-agile team, however, leads to a more probable shift back to the old ways of working in initial stages of adopting agile methodologies. To prevent this from happening, the mentor emphasizes the significance of continuous adherence to agile. With the help of the mentor, the team can become and prevail as a self-organizing team.

2.3.5 Coordinator

Agile methodologies involve the customer a lot more in the development process than traditional waterfall methods. Basic Agile practices such as writing user stories, refining product features, prioritizing the stories and giving rapid feedback to the development team can be difficult to coordinate. Some factors that affect the coordination are the physical distance between the development team and their customers, lack of time commitment on the part of the customers and ineffective customer representation. Coordination across geographic and time zone differences is a challenge in a remote setting. Having a designated team member acting as a proxy between the team and the remote customers was found helpful by many teams. The coordinator synchronizes the distant customer with the development team and facilitates collaboration with customer representatives who may be perceived as ineffective otherwise.

As agile development is first and foremost about responding to change, it is important that someone is coordinating change requests even in the most distributed settings. To begin development, the team needs a list of customer requirements in a prioritized order. Therefore these requirements need to be collected and clarified before development can ensue. This duty falls into the hands of the coordinator as well.

While an agile team is reaching maturity, it may not be performing and delivering to its full potential. Especially the first sprints of the team are very volatile and the velocity of the team might vary a lot. The coordinator is there to manage customer expectations during this phase.

2.3.6 Translator

Because agile teams are cross-functional, there are team members with many different backgrounds. The members of a development team come from a technical background and use a more technical language in their communication, whereas the customers use a more business language consisting of vocabulary from the customers' business domain. This language barrier needs to be overcome in order to make efficient and frequent collaboration possible.

In some agile methodologies, user stories are initially written as stories by customer representatives using business language. The development team, on the other hand, needs tasks written in a technical language in a specific enough manner before the development can be taken up. The translation from business language to technical language happens when stories are split into smaller tasks.

Initially it seemed that business analysts played the role of the translator. Their ability to understand both technical and business languages made them a good fit to act as a bridge between the development team and the customer representatives. However, it became apparent that the role could be played by anyone on the team with good communication skills and understanding of business concerns.

There are tools that assist the team members in playing the role of the translator. A project dictionary is one of the tools. It is a document which explains different business terms and when to use them. The dictionary can be even used in naming variables in the code by providing a direct mapping from the business domain to the technical domain. Another tool that helps the translator is iterative reasoning which allows the abstraction of

technical solutions to higher levels to align them clearly with their business drivers. Cross functionality also aids the team members in carrying out the task of translation.

Depending on the maturity of the team, the role of the translator could land spontaneously on different team members. In less mature agile teams, there are often one or two individuals who play the role of translator based on their personal abilities or professional skills. Recruiting smart and pragmatic communicators was an approach that more mature teams took to ensure having people with natural translator skills in the team. Thus, mature agile teams have bilingual members throughout. These members can switch from technical to business language depending on the context.

2.3.7 Champion

Agile teams cannot thrive by working in isolation. The support of senior management plays a big part in the early stages of adopting agile methods and also afterwards. Organizational culture, contract negotiation, resource management and financial support are all influenced directly by senior management. If senior management does not have a proper understanding of agile principles and practices, it can have a severe effect on an agile team's ability to function up to its full potential.

Understanding the business drivers is essential for a champion. The champion must keep these drivers in mind in order to be able to secure senior management support. For the senior management, agile methods are not the only viable option for projects but merely one of many. The methods which provide the best returns on investment are usually the most tempting for senior management.

The champion must realize that agile methods might not be the best fit for all types of project contexts and avoid promoting them while being negligent of the surrounding organizational context. Explaining the benefits of agile methods and helping the senior management see the suitability of those in the context they are operating in, is the task of the champion.

In organizations where agile methods have not yet gained much foothold, the champion should help educate the senior management about them. One efficient way to educate senior management, is to show the agile methods in practise. This can be done by setting up pilot teams which can help showcase agile practices on a small scale. This is especially useful in projects which have not succeeded with traditional methods before.

The champion should be driving towards having more agile teams across the organization after proving their applicability with the pilot teams.

2.3.8 Promoter

The importance of customer involvement in agile development cannot be overemphasized. However, it often does not come easily and many agile teams are working in an environment where that factor is lacking. This can lead to numerous issues within the agile team. Having proper customer involvement, ensures that the customer's vision is actually used in the development of a product.

Customers, who are just adopting agile or have had bad experiences with it before, can have fallacies and distrust in agile software development. The promoter tries to understand the customer, their background regarding agile software development, and most importantly, their willingness to work in an agile fashion. This way the promoter can help the customer to assuage their concerns.

Depending on their previous experience, the customers can be unaware of their role and responsibilities in an agile setup. The promoter is there to help the customers systematically apply agile practices. It is important that the promoter emphasizes the benefits of agile software development. Without a collaborating customer, an agile team is unable to achieve its full capabilities.

2.3.9 Terminator

The last role may be somewhat controversial and it most certainly is not an easy one. Basically the role of the terminator is identifying team members whose personalities and practises might be harmful for the team's productivity. In this sense, the individuals themselves are not considered bad but they might not be suited to agile software development as well as required. As a result of this, the terminator participates in selecting the individuals who will become part of the team and trying to eliminate the non-suitable persons up front. As an extreme measure, the terminator can also seek to remove an individual from the team if that team member is hampering the team's functionality.

3 E-commerce

In this chapter we will examine the field of e-commerce, its current situation and future trends related to it. This will give an insight into the environment and activities where the software developers are working and the data is generated.

Electronic commerce, or e-commerce, means conducting traditional commerce through digital technologies. Traditional commerce refers to selling goods and services. In e-commerce these transactions happen online, either individually (e.g. Amazon, Alibaba, Zalando) or over time as a part of a subscription (e.g. Spotify, Netflix, the New York Times).

Strauss and Frost explain e-commerce as an internet application which is a subset of e-business [18]. They use the definition of e-business by Gartner Group that explains it as repeated optimization of a firm's business activities by means of digital technology. According to them, every business uses it and, in addition to e-commerce, it consists of digital communication and online research. E-commerce is the part of e-business that focuses on transactions. Mouruya and Gupta have a simpler definition for e-commerce: "an emerging concept that describes the process of buying and selling or exchanging of products, services, and information via computer networks including the internet" [19].

Due to e-commerce happening via the internet, it has been available for a relatively short duration. It has been available since the opening of the internet to commercial use in the early 1990s [19]. For example Amazon, the largest e-commerce site in the world, was founded in 1994, the same year as the first commercial web browser (Netscape Navigator)

became available. E-commerce has experienced enormous growth worldwide since the early 2000s. The amount of e-commerce customers reached one billion people already in 2013. Not only has the amount of e-commerce customers grown significantly but the amount spent online per customer on average has increased considerably. In 2016, that number surpassed 1500 USD. Table 3.1 shows these numbers from 2011 to 2016.

Table 3.1: Global growth in e-commerce [20]

Year	Number of e-commerce customers worldwide (in millions)	E-commerce sales per customer worldwide (in US$)
2011	792.6	1162
2012	903.6	1243
2013	1015.8	1318
2014	1124.3	1399
2015	1228.5	1459
2016	1321.4	1513

Because consumers have shifted towards trading goods online, traditional commerce has seen a decrease in popularity. An ever growing portion of retail sales is conducted through computer networks. Table 3.2 depicts how the share of e-commerce has increased in the USA in the last five years. The exact numbers might not apply globally but the trend is similar everywhere around the world. This has forced many companies to gear their business towards e-commerce. However, this does not mean that the companies would abandon their physical business. Rather, they have combined the online world with the existing business processes. [19] The benefits that e-commerce brings include faster speed, lower expenses and a potentially massive customer base which is not dependent on distance or time. These benefits have attracted businesses from a myriad of different fields such as retailing, communication, banking, transportation, education and medical

services. [21]

Table 3.2: E-commerce in the US [20]

Year	US e-commerce sales (in US$ billions)	US e-commerce sales as a percentage of total retail sales (in US$)
2014	399	9.7
2015	343	10.7
2016	390	11.8
2017	453	13.2
2018	523	14.4
2019	601	16.0

3.1 Types of e-commerces

Business-to-consumers (B2C) is the most famous type of e-commerce. Its aim is to do business with individual consumers online and it covers retail goods, travel, financial, real estate, and other types of services and online content. This wide range of opportunities to make e-commerce enables a lot of different business types such as online retailers, service providers, transaction brokers, content providers, social networks, market creators and portals. B2C e-commerce has grown exponentially since 1995 and over the next five years it will be growing over 20% annually according to data [22]. Because a tiny part of the retail market is online, there is still plenty of room to grow in e-commerce. These revenues will continue to grow but not forever with the current speed. Obviously when the online revenue conquers the market the growth of it will decline. This will not be for a long time and there are parts, like music, video, games, and entertainment, which have even longer time to grow before diminishing.

Business-to-business (B2B) is the most popular form of e-commerce. Evidently, this means that business takes place between two separate businesses. In total the B2B market is huge and at the moment e-commerce covers a small portion of it. The growth potential of B2B e-commerce is tremendous. The most commonly used business models in B2B e-commerce are net-market-places and private industrial networks.

In *consumer-to-consumer* (C2C) e-commerce, selling happens with the help of an online market maker i.e. a platform provider. The seller and buyer are both consumers. The role of the consumer is to prepare the product for the market, place the product for sale (or auction) and trust on the platform's ability to enable trade. On the other hand, the platform provider supplies catalog, search engine, and transaction clearing capabilities. This way the products can be displayed, discovered, and paid for, as simply as possible. Nowadays this kind of platform provider faces remarkable competition. The original C2C e-commerce platform providers in the United States have been for example eBay and Etsy but today Amazon with its third-party sales and Facebook Marketplace conquer the market. There are also plenty of new companies stepping into this C2C market. Uber and Airbnb are also defined as C2C platform providers even though those are on-demand services.

3.2 Future trends

Cloud computing provides substantial benefits to implement e-commerce. For instance, it considerably reduces the cost of building and operating websites. This is due to licensing services from cloud service providers (CSPs) instead of purchasing the hardware infrastructure and software as a product. Then companies can embrace different strategies while building out their websites, such as "pay-as-you-go" and "pay-as-you-grow". There are not only benefits for companies but for individuals too. Generalization of cloud computing has decreased the need for efficient laptops or desktop computers when using

e-commerce or other activities. Low-cost tablet computers or smartphones are sufficiently powerful when utilized technology is cloud computing. Similar phenomena are noticeable with corporations which do not have to own these infrastructures anymore. Instead of that they can reduce the cost and services can be maintained online with cloud computing.

Gary Davis predicted in 2018 that there would be 50 billion devices connected to the internet by 2020 [23]. In addition to increasing the number of devices that can be used to conduct e-commerce, they also create new ways of conducting e-commerce. For example smart speakers, which have become popular in the recent years, are an example of a new platform that can be utilized in e-commerce. As well, social networks have now and in the future more and more influence on e-commerce businesses and how they are operating, communicating and serving their customers. Some day e-commerce might be implemented via smart connected ”things”, like houses and wearable technology. The Internet of Things and artificial intelligence can create businesses which cannot be imagined at this moment.

E-commerce has become significantly more accessible for very small entreprenuers as well during the last decade. This is partly due to the tremendous decrease (over 50%) in the cost of building and operating a website. Also a lot of tools and services have been introduced lately which enable so-called no-code and low-code websites to be built. This means that building a website with these tools does not require much technical or programming knowledge. And this vastly increases the pool of people capable of starting an e-commerce. On the other hand, this way there is more competition between e-commerce entrepeneurs and businessses. Competitiveness of an e-commerce company can be evaluated from a strategic point of view. For instance, a few key industry strategic factors are industry value chain and nature of infra-industry competition. If plenty of e-commerce businesses utilizes tools and services described above, it may be difficult to stand out from the e-commerce crowd in order to succeed from a strategic perspective.

The enourmous progression of technology over the past decade has enabled lots of

new possibilities, however, it has also brought new challenges with it. For example, the highspeed broadband environments of today have reduced internet users' patience. People are not willing to wait for websites to load unnecessarily long. In 2018, only 47% of mobile users will wait for a page to load for more than three seconds [22]. In turn, the expanding amount of internet users yields more big data which creates more analytics possibilities to understand the behaviour and needs of customers. This generates new business opportunities for companies who have the ability to collect and access online data and analytical competence to understand it. This and the enormously increased online customer volumes have led to e-commerce businesses being data-driven with regards to the development of the e-commerce sites. The development of e-commerce business based on big data and web analytics has a great potential to create a competitive advance in the market.

4 Web analytics in e-commerce

In this chapter we will first examine what is meant by web analytics in the context of e-commerce. Once we have established what web analytics are, we will then delve into how the analytics can be used in practice. We will then reflect this information back to the knowledge on agile development and formulate some hypotheses on how the analytics should be used especially in agile development teams and organisations.

To be as effective as possible in conducting business on the web, continuous refinement of the web site is required [24]. But in order to refine the site, it is necessary to figure out what parts of the site are performing inadequately. Web analytics is the solution to this problem. The tools used for collecting data on the web and benchmarking it are all part of web analytics. E-commerces, that conduct business on the web, can thus take advantage of web analytics.

Web analytics is a broad term that covers many tools, techniques and methodologies. One way to divide web analytics tools are offsite tools and onsite tools. Table 4.1 lists a few different metrics for both offsite and onsite tools. As can be seen, offsite tools concern things that happen in the bigger picture around the whole internet. Online tools, on the other hand, are, at their core, directly connected to the web site itself. These two types of tools differ fundamentally from each other which is why they generate vastly contrasting outcomes. This issue is irritating and comes up regularly for those who bring together the metrics. It needs to be accepted that the tools produce different results. Instead of trying to compete with the different metrics they should be used to reinforce one another. [24]

In this thesis, the focus will be on the onsite web analytics, and further on the term web analytics will be used to refer to onsite tools specifically.

Table 4.1: Metrics in different web analytics tools [24]

Offsite metrics	Onsite metrics
Potential audience	Visitor's onsite journey
Share of voice	Visitor's drivers
News around the internet	Web site's performance

Web users produce a massive amount of web usage data which can be collected, measured, monitored, analysed and reported and therefore utilized to understand and improve user experience. This way web sites can be optimized to make customers satisfied and loyal which helps to accomplish business goals. [25] The data sets are generated by combining web site activity or clickstream records with demographic and other behavioral data. The collected data is complex and sizable. This collecting results in enormous databases which require automated analysis techniques to be managed. Taking advantage of analytics requires human effort to create strategic insights about the activity on the e-commerce site and the behaviour of the users and potential customers. Because the amount of data is incomprehensibly large with millions of clickstreams created every day, automated techniques which find and point out the essential patterns in the data are needed. [26]

4.1 Methods to collect data

There are different methodologies for collecting web site data. The two most common of those are server log files and page tagging. Using server log files is an older method and one that web analytics got started with initially. However, page tagging has emerged to be at least as important a method as log files if not even more important. [25] And today

both of these are often used in conjunction with each other.

4.1.1 Server log files

Web server log files used to be the most popular choice for web analytics due to their easy availability [24].

A very common but simplified flow for using web could be described with the following steps:

1. User types a URL in a browser
2. Browser sends an HTTP request to a web server
3. Web server sends back a response with the requested page
4. User does some action(s) on the page
5. Browser sends an HTTP request to a web server
6. Web server sends back a response

Steps 5 and 6 can actually repeat arbitrarily many times during one session depending on the web site implementation. Originally, web sites often loaded the entire page again when the user made some actions on the page. This was of course very wasteful, especially on large and complex pages, where only a small part of the web site would have been in need of updating. To solve this issue, a technique known as AJAX (Asynchronous JavaScript and XML) was created. Basically, the idea of AJAX is to fetch specific data from a web server. This is a much more lightweight and efficient method for responding to user interactions instead of fetching an entire web page every time. First AJAX was used via the XMLHttpRequest API and XML was indeed the widely used data format. More recently, however, the Fetch API has replaced the XMLHttpRequest API and JSON has become the most common data format.

In the above flow, server log files would be created between steps 2 and 3, and steps 5 and 6. This traditional data collection method has been in use since the dawn of the world wide web and the release of the first universally used browser, Mosaic, in 1993. [27] When server log files are in use, the server creates a log file for every request that it receives from a client. In web context these requests most often come from web browsers. This method in itself happens completely server-side and is thus independent of any clients [24]. This gives the website owner complete control and ownership over the information and its privacy [28]. In Figure 4.1, server log file creation is depicted with the flow A in the process.

The actual data that is stored for each request varies between implementations but it mostly contains HTTP headers and server activities in textual format [27]. Examples of the attributes that log files can consist of, but are not limited to, are the IPs of the request initiator and the server, process ID, datetime, HTTP request command, response status and size, time taken to complete the transaction, bytes transferred and the referer. The format of the used log files can be predefined, like Extended Log File Format [1] and NCSA Common log format [2], or it can be customised as required. Even though there is lots of different information to log, there are still limitations to the types of collectable data. Data that log files are not able to capture include for example user interactions within the web page and different mouse events like hover and click for instance. To analyse and extract log files, log analysis techniques can be utilized. [27]

4.1.2 Page tagging

The other common web analytics method is page tagging. This method emerged after server log file analysis to combat some of its shortcomings and also provide more accurate

[1] https://www.w3.org/TR/WD-logfile.html

[2] http://publib.boulder.ibm.com/tividd/td/ITWSA/ITWSA_info45/en_US/HTML/guide/c-logs.html#common

data. As opposed to server log files, page tagging is used on the client side. As web pages and applications have grown larger over time, the technologies around them have also become more complex. Caching is one technology that has become important and widely used with today's rich internet applications. In essence, caching means storing and allowing more efficient reuse of previously used data. Obtaining the data might include some resource intense calculations or lots of network request overhead which makes it unfavourable to do all of the required steps each time the data is needed. With caching this data could be stored in other places than the server, like the user's browser or content delivery networks (CDN). With caching in place the request sent by a client would not necessarily reach the server since some entity could have served the data in its place. This would result in server log files not being generated for that request and thus the data regarding this request would be lost. This is one important reason why client-side analytics are needed as well.

As page tagging is a client-side method, it utilises client-side technologies. Page tagging is often implemented with embedded JavaScript scripts but it could also use browser plug-ins and add-ons. Getting started with page tagging analytics is relatively simple. If an external analytics service, like Google Analytics or Adobe Analytics, is used, then all that is usually required, is downloading the analytics script from the service provider. For example, adding the analytics capabilities with Google Analytics' global site tag requires adding only the following snippet to a web site [29]:

```
<script async src='https://www.googletagmanager.com/gtag/js
?id=GA_MEASUREMENT_ID'></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());
```

```
  gtag('config', 'GA_MEASUREMENT_ID');
</script>
```

What this snippet essentially does is define a function `gtag` to the browser's JavaScript's global scope. This function can then be used wherever needed. Page views and events can be reported by calling that function with certain parameters in the desired locations. For example, when user adds an item to the cart on an ecommerce site, that could be reported with the following code:

```
gtag('event', 'add-to-cart', {
    'items': [
    {
      'id': '123',
      'name': 'T-Shirt',
      'category': 'Apparel/T-Shirts',
      'variant': 'Black',
      'quantity': 1,
      'price': '10'
    }
  ]
});
```

Even though this is just one example implementation of page tagging, the idea in most of the implementations is similar. Some kind of analytics reporting capability should be made available for the web site first. And when what should be tracked on the site has been figured out, then the code that is run on these occasions should be appended with the call to the analytics service. In Figure 4.1, page tagging is depicted with the flow B in the process.

Not only does page tagging fix the issue of losing analytics data due to caching, it also has numerous other advantages. Web sites are nowadays often accessed by non-human

user-agents (crawlers, spiders and search engines). These visits are not desirable in the web analytics data because they do not represent customers' behaviour. Page tagging excludes these visits from measurements and reporting since those do not execute script tags in HTML pages. This separation can be done in server log file analysis as well, but it is time-consuming. [28] However, if there are bots that do perform actions or execute code on the website, they should be excluded from the analysis data by some means. For instance in 2020, a bot created by Google was adding products to shopping cart on different e-commerce sites to ensure that the prices displayed on the site matched those shown on Google's own platforms [30]. In the e-commerces' web analytics data, this showed up as abandoned carts which may have resulted in erroneous analysis.

With page tagging, the acquired analytics data can be much more detailed which might assist in generating new insights and better hypotheses. This is especially useful because of two reasons. First, the scope of different devices and ways of using web pages has increased enormously within the last decade. Second, web site's today have increasingly more client-side features and interactivity the usage of which is best captured with client-side analytics techniques. And since page tagging is often used through a Software-as-a-Service (SaaS) provider, the organization does not need to save, maintain, or process the data locally making the maintenance of the data easier. [27]

These benefits have led to page tagging becoming a major technique for web analytics data collection in many organizations. On the other hand, these advantages make page tagging more invasive than server logging and require the web site to be modified in order for the analytics data to become available. [27] It is also important to note that some users might be excluded from the analytics data due to browser extensions which, for example, block third-party tracking cookies or executing the analytics scripts on a site. Also, if a user closes the site before analytics have been loaded, then that session would not be recorded in the analytics data.

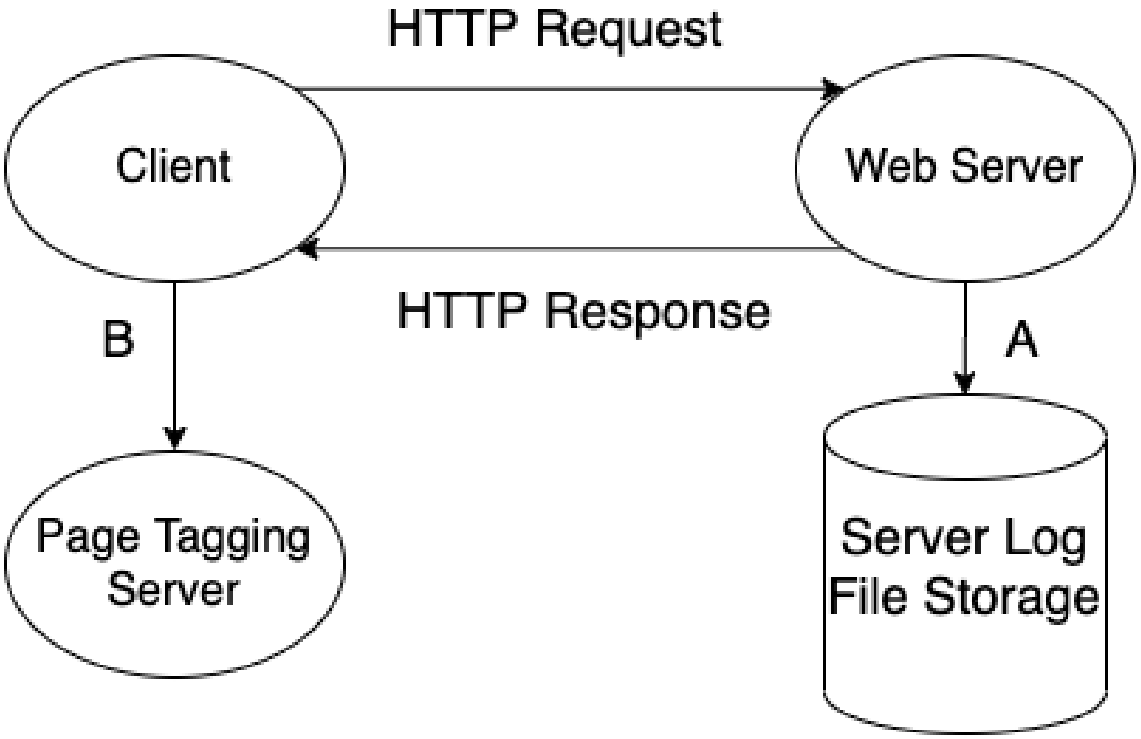

Figure 4.1: Web Analytics Techniques

4.2 Web Analytics in practice

For any website, not only e-commerce sites, visitor acquisition is extremely important and a very versatile activity. It can employ techniques such as mail, email, affiliate marketing and search. Due to the importance of this activity, many websites have a strategy and people working mostly around Search Engine Optimization (SEO). But in e-commerce sites, the main business is selling goods or services online, so acquiring visitors is not feasible enough on its own for any e-commerce business. Contrary to what some marketers believe, it is not the end of the process but only the start. [28]

4.2.1 Web analytics process

Figure 4.2 visualises this very well. A web site can be thought of as a funnel where some amount of visitors drop off at each different step of the customer journey until eventually some percentage of visitors actually make a desired action. The making of this action

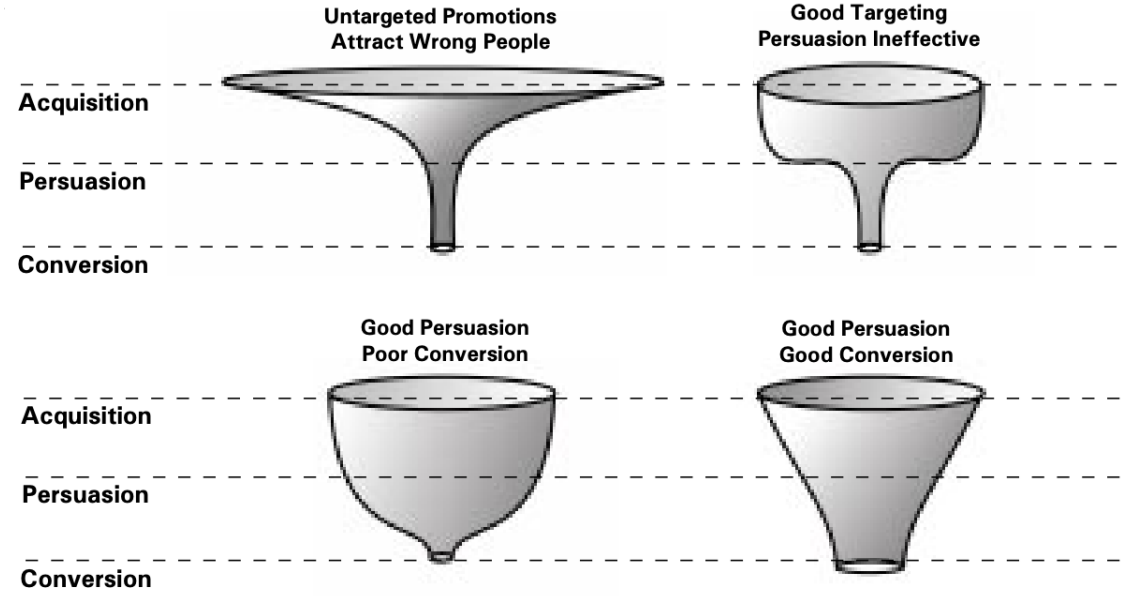

Figure 4.2: The customer lifecycle funnel [31]

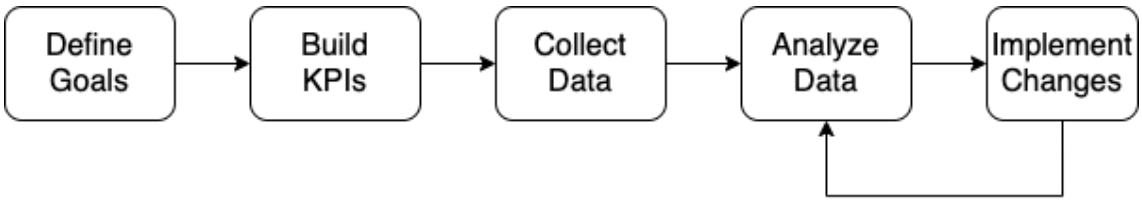

Figure 4.3: Web Analytics Process [28]

is often referred to as a conversion. What a conversion actually is, varies depending on the web site but in an e-commerce site it, in practise, always refers to a visitor making a purchase. The top-left funnel in the Figure 4.2 depicts the situation mentioned previously where a site gets a lot of visitors but only a small amount of those visits end up in a conversion. Getting from that situation to the situation represented on the bottom-right of the Figure 4.2, is where web analytics can help enormously.

Web analytics in practise is an iterative process. Waisberg and Kaushik recognised five steps in analysing the performance of a website which are visualised in Figure 4.3 [28]. The five steps are explored in detail next.

1. *Defining goals* to figure out why your website exists in the first place. Different

websites will have different answers to that question. These goals will serve as an important input for the next step.

2. *Defining metrics* to be able to measure if the desired objectives are achieved. This is usually done by constructing Key Performance Indicators (KPIs). The KPIs are the figures that are followed through the web analytics process. Each KPI should have an action linked to it or otherwise it would not be worth collecting. In order for KPIs to be good, they should be un-complex, relevant, timely and instantly useful. Table 4.2 shows an example comparison between two different possible KPIs. Visitor rate is a very self-explanatory metric which is not a good KPI for e-commerce sites. It is by all means an important metric, but on its own, it does not tell you anything about whether an e-commerce site is making any revenue or not. The conversion rate however, directly tells you if your site is making revenue. A conversion means a desired action on a website, for example, a purchase on an e-commerce site. Conversion rate is in fact calculated by dividing the amount of conversions by the visitor amount. So as an example, a site with 1000 visitors and 30 purchases, has a conversion rate of 3 percent. Other good KPIs for e-commerce sites could be for example, total revenue, average order value (AOV) or amount of products sold. However, it is important to note that this example only applies to e-commerce sites. On sites where each visitor brings revenue directly due to for example advertisements, visitor amount might be a good KPI.

3. *Collecting data* is the part where the actual web analytics are implemented into the system. This includes the techniques mentioned in Section 4.1. Also a structured approach to making experiments and implementing changes often means that the analytics implementation needs to be extended and modified regularly. And as the name of this step directly implies, the data also needs to be collected which might take time. If you are conducting experiments, depending on the desired confidence of the results, you might need to run the tests for some time. Nonetheless, this step

Table 4.2: Comparison of two example KPIs

KPI	Conversion rate	Visitor amount
Un-complex	Yes, very simple to understand.	Yes, very simple to understand.
Relevant	Yes	No
Timely	Yes, very easily available if implemented into the system.	Yes, often built-in to analytics tools.
Instantly useful	Yes	No, does not tell by itself how well an e-commerce website is achieving its goals.

should occur as often as the following two steps, so on each iteration. That is why Figure 4.4 actually depicts the analytics process better.

4. *Analysing data* includes, on one hand, doing statistical analysis for the data to make sure it is statistically correct and on the other hand thinking about the reasons why customers behave on the website the way the data reports. As a part of this step, different hypotheses on how certain changes, if introduced to the site, would affect the different KPIs. These possible changes are the actual outcome of this step which will be used directly in the next step.

5. *Implementing changes* means implementing a new version of the site which is, based on the analytics data, believed to be more profitable. If the changes are done for a new A/B test that will be conducted, the base version of the site will remain intact and it is only extended with a differing version. The change could also be just removing the A or B version of the site if the particular version has been identified as inferior.

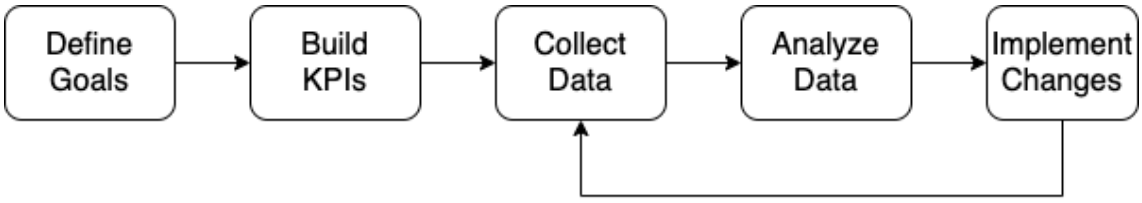

Figure 4.4: Updated Web Analytics Processes

4.2.2 A/B testing

According to Kohavi and Longbotham, A/B testing, also known as (online) controlled experimenting and split testing, is a technique that makes possible discovering causal relations with high probability. This is different from most data mining techniques which are used to discover correlational patterns. When conducting A/B tests, hypotheses like ”If a specific change is introduced, will it improve key metrics?”, are formed and then evaluated with real users. The procedure of running controlled experiments has existed for almost a century and with the growth of the web, it has become an essential tool for any website owner as a part of web analytics. Large websites, like Amazon, Google, LinkedIn and Facebook conduct thousands to tens of thousands of A/B tests every year which really demonstrates the importance of this technique. [32]

The idea of A/B testing a website is very simple and works the same way across different websites. The details may vary but the overall principle stays the same. A test setup consists of two versions of the site: a control (version A) and a treatment (version B). There might actually be even more versions of the site which are tested side-by-side but the simplest setup only consists of two versions. The two versions are then run for a certain period of time so that each version is shown to some percentage of the users. Figure 4.5 by Kohavi and Longbotham shows a high-level structure of an A/B test. After the data has been gathered, assuming that everything with the test went correctly, a decision regarding the experiment and the next steps can be made. A/B testing fits extremely well into the web analytics process depicted in the previous subsection. When

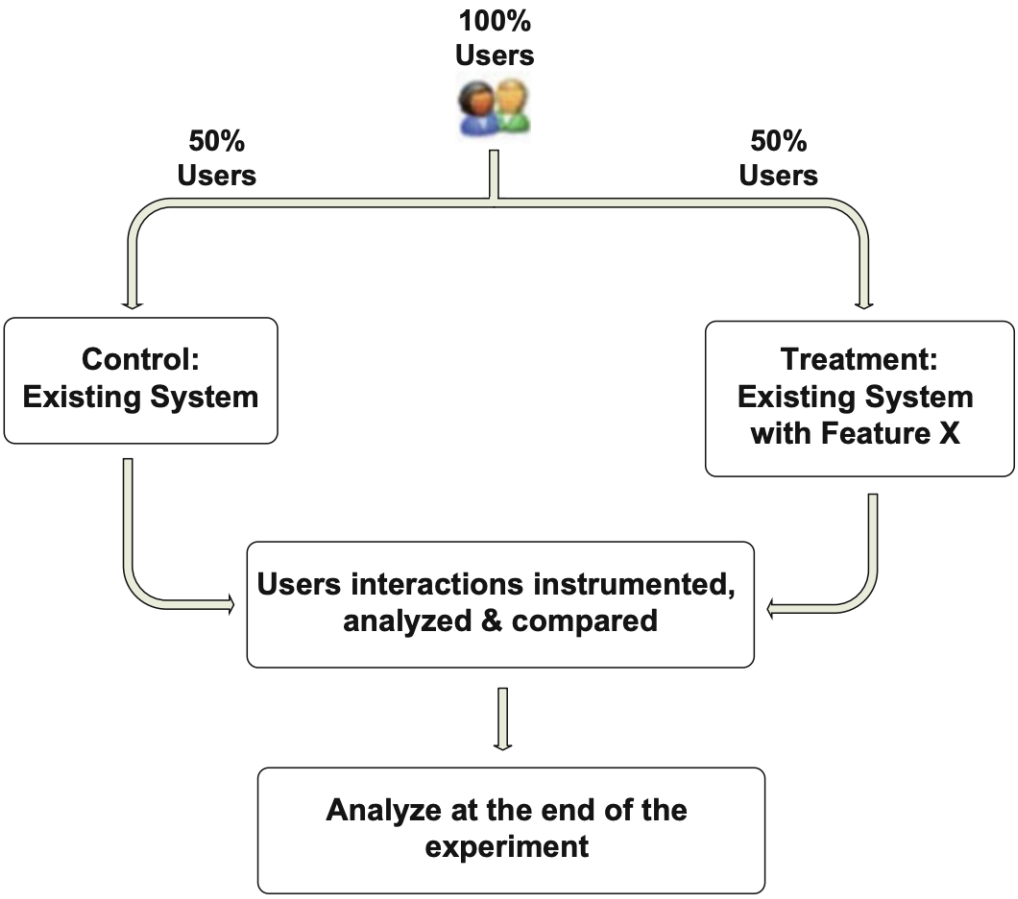

Figure 4.5: A/B test structure [32]

A/B tests are conducted over and over again, the last three steps of the process are iterated over just like the process represents.

The explanation above is very simplified and does not go through any details of A/B testing. There are many statistical intricacies and details related to it, for instance, how the randomization of users should be done or how long a test should be running to achieve a desired sample size. To ensure that the conducted A/B tests are statistically correct and the results are trustworthy, it is essential that there are people with expertise in statistics as a part of the team building the website.

4.2.3 Hypotheses

The roles which are discussed in the hypotheses are the Scrum roles which have been explored in Section 2.3.

Hypothesis 1 (H1): *A Development Team needs to have the statistical knowledge and skills within itself.*

Agile methodologies, Scrum in particular, emphasize that the development team is cross-functional and self-organizing. The team should have the knowledge and the power to implement the software as they best see fit. The team as a whole is responsible for what they build. That is why it is essential that the A/B tests are not analysed and planned in isolation from the team. The only part of the analysis that could be done outside of the team, is analysing the ”what”. In essence, this means doing the calculations on how the key metrics have changed, i.e. what has happened on the page. Figuring out why those things have happened and what should be done next, should involve multidisciplinary skills.

Even though the pure statistical analysis of any experiment could be done by someone outside of a Scrum Team, there are still other aspects to web analytics that support the idea of having the knowledge within a Development Team. When analysing why the users have behaved as the data suggests, new hypotheses and experiments will usually arise. When planning the next experiment, it is crucial that the person who knows about the details of conducting the experiments is present to give insight on planned experiments from the get-go instead of it being an afterthought. It is also not unlikely that the Scrum Team has questions about the past experiment. If the person who has done the analysis is part of the Scrum Team, they will most likely attend the different ceremonies together. This leads to closer collaboration and better communication instead of just passing reports and messages back and forth which is a desirable manner of communication like stated in the Agile Principle P6.

Analogous to the Scrum Master in a Scrum team, who observes the team and promotes the agile way of working within the team, the person with the analytics knowledge ensures that the team is using web analytics correctly. In addition to this, they can also share the statistical knowledge within the whole Scrum team and provide understanding on why things are done in a certain way.

And to add, everything mentioned previously was assuming that the A/B tests are technically setup and ran correctly. But as the different versions of a website and the testing setup itself are software in the end, they are prone to errors and malfunctions which means that the A/B testing cannot be separated completely from the software development. Thus, instead of building silos, collaboration between the different disciplines should be embraced.

Hypothesis 2 (H2): *A Product Owner needs to take part in defining KPIs in order to understand them properly and ensure that they are realistic.*

KPIs are often defined by business people who do not necessarily participate in day-to-day development much. It is important that the Product Owner understands what the KPIs measure and why they were chosen. This way they will better understand how they affect and relate to different features that the Scrum Team is asked to deliver. This shared understanding of the KPIs between the stakeholders and the Product Owner eases the communication between them and enables the Product Owner to reflect the stakeholders' wishes better on the Product Backlog. When the Product Owner prioritises the work, if a work is expected to affect some commonly known metric, it easier for them to understand the total return on investment. This is because the metrics should have been chosen so that it is easy to understand the monetary value linked to them.

It is also important that someone would take part in this activity to ensure that the KPIs are realistic and measurable right from the beginning. As the Product Owner is the person to communicate with business people and other stakeholders on a regular basis, it would make sense that they are the one to do this. However, this means that they need to

understand the technical limitations of the web analytics and the systems in use. With this understanding they can tell whether a KPI or metric is something that can be measured, or even offer better alternatives.

Hypothesis 3 (H3): *A Product Owner needs to utilise the information provided by the web analytics process when prioritising work.*

When a functionality or feature is requested from a Scrum Team, it can require A/B testing. Analytics data can help in making more realistic estimations on the impact of the changes that a Scrum Team will make. The Product Owner should embrace this and utilise it as much as possible when prioritising work. When new experiments are planned, calculating monetary estimates for these experiments should be feasible with the help of the analytics data. In order to maximise the return on investment of the work, it is important that they have realistic and reliable estimates on the monetary value of any planned work

Web analytics, and A/B testing especially, require that the data is collected before anything can be decisively concluded. This factors into prioritising work by imposing upon the overall work that is required to implement a piece of work. As opposed to something that would not be experimented with, a feature cannot be just built and released. This means that the overall time and effort to build an A/B tested functionality is different to doing the same functionality without the test. This is something that the Product Owner needs to be aware as well when prioritising work.

It is also important for the Product Owner to understand the surrounding operating environment and whether that could have an effect on the feasibility of running experiments. In an ecommerce site, there could be independent variables that the Product Owner should also factor in when deciding on what should the Scrum Team work on next. Since the point of web analytics is to provide trustworthy insights on the behaviour of the users on a website, it needs to be taken into account if something could compromise the reliability of data. For example, running an A/B test during a day when the volume of users

is known to be abnormally large, like Black Friday, could make the data of an experiment less trustworthy if the customers are known to behave in a different manner than normal during that time. That is why the Product Owner needs to be aware of things that could hinder the data analytics process.

Hypothesis 4 (H4): *A Product Owner needs to communicate how the Scrum Team's work is expected to be reflected in the web analytics data.*

In large organizations, there can be many different teams working around the same product. This means that there are different consumers from separate teams, who might not be well aware of what the other teams are working on. If one team implements something on the website that causes visible changes in the analytics data, it can interfere or confuse other teams. Another team could, for example, think that there is a bug in the website because of the sudden change in the data, even though the explanation behind the change is completely reasonable. If the team would start to investigate this alleged issue, all of the work would be in vain due to no issue actually existing.

And with teams working on the same product, they will be conducting A/B tests of their own. It is important to keep the other teams aware of the expected changes in the data so that they can take this into account when they are analysing the results of their own experiments. In the worst case, this lack of coordination could lead to the experiment being discarded and conducted again from the start.

These examples show that if the Scrum team fails to communicate how the analytics data is expected to change, it can lead to wasted effort on other teams' part which would be directly against the Agile Principle P10. And as the Product Owner is the one who communicates outside the team the most, it is natural that they take care of this.

Hypothesis 5 (H5): *A Scrum Master needs to understand what the different parts of the web analytics process are and why they are essential.*

To be able to ensure the correct usage of Scrum, a Scrum Master needs to understand the recurring phases of the web analytics process. If a Scrum Master does not understand

the role and responsibility of the different steps, it can make it harder for them to embrace and oversee the correct usage of Scrum within the team. With enough knowledge about the process, they know why some things are done in a certain way and why those things are also necessary. This way, they can see the context in which the team is operating in more clearly, and can recognise what actually are impediments and what are unavoidable parts of the web analytics process.

4.2.4 Software developers

Nowadays businesses share the mined data and the results of analytics to a wider audience which can utilize those at their everyday activities. The results can be reflected faster to business actions because of integrated knowledge discovery, analytic applications, business intelligence systems and transactions. This creates a closed loop between operations and analysis.

Analytics are consistently and widely used in different fields of business, such as merchandising, marketing, sales, supply chain optimization and fraud detection. Analytical tools are used to enhance business processes and the users of these tools are usually experts in their own field but not in data analytics or statistics. Earlier these experts relied on analysts whom they delivered domain knowledge to and then waited for the analyst to communicate the organized and analysed data back. Usually, results from analytics raise questions and some further analysis is executed before the acts based on analytics can be made. Nowadays experts are more dependent on analytic applications which are a fusion of data analytics and task-specific knowledge. Additionally, these kinds of applications can give recommendations about how to analyse the data and present extracted information advantageously to users.

Agile methodologies promote close collaboration between highly-skilled individuals. In particular in Scrum, it is essential that the team is cross-functional and self-organizing. This means that the knowledge and ability to utilize analytics should be contained within

the team. As discussed before, just enabling the usage of analytics already involves software development effort which in turn means that analytics cannot be utilized without software developers themselves. If we look at the values of the Agile Manifesto, we can see that responding to change is very important. Using analytics not only makes responding to change easier but it also highlights what needs to change. When something on an e-commerce site is not performing as well as it could and/or should, appropriately implemented analytics will expose it to the organization. And the reason why something is not performing, might actually be change itself, for example in evolved consumer preferences.

Inspecting how past experiments have gone, learning from those, and making future actions based on what is currently known is at the center of agile. This can be applied to numerous things with the traditional Agile Methodologies applying this first and foremost to development practices and technologies. Web analytics enables the inspection of how a web site is performing and learning how the customers use it. However, what web analytics does not tell you, is why the customers use your web site the way they do. This creates a need for skills to figure out this information.

Let us look at an imaginary scenario. Company A has gathered the following information from their web site using web analytics. They know that the average order value (AOV) of a logged-in customer is 40% higher than the AOV of a guest customer. They also know that only 20% of all customers login on the web site. This is all information that comes directly from the analytics. If the company was able to increase the logged-in user share to, for example 25%, it would mean an increase of 1.85% on the overall revenue (assuming that the AOVs stay the same).

On the other hand, the reason why the amount is only 20% or how to increase that amount is not something that the analytics can directly tell you. To increase the login rate it is necessary to carefully examine the login functionality on the web site with people who have expertise on user experience, visual design, growth hacking and even psychology.

This expertise can bring upon the best hypotheses on why the functionality is not used by more users and what would be the best actions to improve the usage rate.

But in an agile software development context, this planning and analysing should not be done in isolation from development knowledge. As we have seen, analytics by nature involve development work, which already by itself creates the need to have developers as part of this planning. But more importantly, it is fundamental, that developers take part in the discussions around the hypotheses and actions to take. If a decided outcome is to change the web page in any way, the required development work does not differ from any other development work in any meaningful way. So planning and refining these changes must not make an exception to usual planning and refinement where the presence of developers is crucial.

As web analytics is, at its core, a piece of software, it is also prone to errors as any other software. The consequences of the bugs may vary from very minor to extremely drastic. But to be able to make data-driven decisions and experiments reliably, it is immensely important that the analytics data is correct. For example, when conducting A/B tests, the runtimes of the tests can be rather long depending on the desired statistical accuracy. If the data is untrustworthy, it may be necessary to run a test all over again, which means that there is not only a lot of wasted time and effort but also lost revenue because of the delay. If decisions are made based on faulty data and the results are also reported incorrectly, the advantages of analytics might be completely lost, and they can actually become a hindrance. Very minor issues might not, due to sheer luck, disable making correct conclusions completely but it makes them much less probable. Also, if the bug in the analytics does not affect the metrics under inspection, analysis can still be conducted.

Bugs can appear in web analytics, as in other software as well, due to many reasons. For example, if someone makes some change to an e-commerce site, that change might introduce a bug in the analytics even if the change itself is not related to the analytics. This type of bug is known as a regression. The bug can make the revenue of the website

appear better or worse. The bug might not even be directly visible in the analytics, but nonetheless the bottom line is that at least some part of the data is incorrect and thus the analysis done on that data is not reliable.

The type of the bug might result in different actions being taken. A bug that results in a perceived increase of revenue might not lead to an investigation of the reason since the increase might not be thought to be caused by a bug. If this happens, it means that the bug will exist in the system until someone happens to find it and then fix it. The same is naturally true for bugs that are not visible in the analytics. If the bug manifests itself as a decrease in the revenue, this most likely leads to an investigation of the reason. A risk in that situation is that the reason for the decreased revenue might not be thought to lie in the analytics implementation itself and thus a lot of effort may be wasted in investigating other possibilities. And if the perceived decrease is large enough, the investigation might escalate quickly and a lot of people might get involved in it.

One way to ensure the correctness of the data, or at least mitigate the possibility of bugs in the analytics, is software testing. Testing is one way to ensure that a software is working as desired. Software testing and many practises related to it are not only very technical but also mandatory aspects to any production quality software. This once again underlines the importance of developers when designing, implementing and using web analytics.

The inner workings of any software are naturally most familiar to the software developers that have built the software. E-commerce sites are no different from this. Web analytics can be utilised in monitoring the performance of the site as well. Web sites can use dedicated tools to log, measure and monitor how the site is functioning from an operational standpoint. For example, HTTP requests can be logged without any web analytics related data to ensure that the different services are operating correctly. No matter how sophisticated these tools are, there can always be issues that slip through to the web site, even if these tools and other processes are in place to ensure the correctness of the web

site.

Some issues might be visible in the analytics data. When the page tagging data is used to track the user interactions and journey on a web site, the data can expose something that should not be happening or even be technically possible in the first place. People who know the site well, might recognise the former type of issues. But to recognize scenarios that should not be technically possible, requires knowledge of the web site's technical implementation. The developers, if assumed to view and analyse the data as well, have the best possibilities to find out if these kinds of issues exist and raise them. That is why developers should participate in the analysis of the data as well. And to be able to participate in that action, the developers should have access to the analytics systems, which often can be external third-party systems to which the developers might not necessarily have access by default.

4.2.5 Hypotheses

Hypothesis 6 (H6): *The software developers should have the same access to analytics systems as any other people.*

As analytics is a core part of any e-commerce site, the software developers are bound to come across it while working on the site. There are multiple occasions where the developers would benefit from having access to the analytics system and it would be almost mandatory to have it. First of all, like mentioned before, the analytics are just software. To build analytics capabilities and integrating analytics systems into a website, is not fundamentally different from any other software programming tasks. To verify that the implementation the developer has built is working correctly end-to-end, they need to be able to see that the data is showing up in the analytics system as well. Without access to the systems, they are not able to do this but are instead reliant on someone else reporting this information to them.

The same dependency on other people could also arise if a bug related to the analytics

is found by someone. It does not matter if the bug is in the analytics implementation itself or if it is actually in the system but is discovered through the analytics. If a bug like this is found by someone who does not have the capabilities or resources to investigate the issue further, it can be left needing more investigation and refinement from the software developers. When the software developer starts investigating an issue like this, they will probably want to check the analytics data for anything related to the issue to understand the root cause of it. And without the access to the analytics data, they are once again left with a dependency on someone else's reports.

The Agile Principle P5 speaks about giving the individuals the environment they need to get the job done. The lack of access clearly goes against this principle. P10 on the other hand promotes maximizing the amount of work not done. Having to ask lots of details from other people who are just looking these details up from some system, introduces extra work and communication overhead, which could be avoided if the developer could look this information up themself. We can positively say that a dependency like this should not exist in an agile team. The team should be empowered to solve the problems they solve on their own. In Scrum especially, the Development Team is completely responsible for executing the work that belongs to them. But with a dependency like this, they might not be able to accomplish this. Since a situation like this can clearly be seen as an impediment for the Team in a Scrum context, the Scrum Master should bring this up and try to gain the Team an access to the analytics data.

Hypothesis 7 (H7): *The software developers should inspect the analytics data regularly.*

As mentioned in Subsection 4.2.4, the analytics data might reveal issues for people who understand the system technically the best. If there occurs some customer flows or interactions on an e-commerce site which should technically be impossible, software developers have the best likelihood of discovering these as they are the experts with regards to technical implementation of a site. Even though issues like these could reveal themselves in other ways, inspecting the analytics data provides a systematic method into

discovering these instead of just relying on luck. This way a team is able to deliver working and more valuable software which is what the Agile Principles also emphasize. And since the practice of inspecting the data regularly most likely leads to earlier discovery of the issues, the potential income losses due to the issues are mitigated as much as possible. This creates a great incentive for the software developers to inspect the data and is also another argument for why developers should have the access to the analytics data in the first place.

4.3 Challenges in utilizing analytics

Web analytics is expected to provide advancements to business performance on multiple metrics. Despite the utilization of analytics in business becoming mainstream, there is a remarkable gap between relevant analytics and strategic business needs. This gap has been described by many challenges:

- *Long cycle time* which consists of the overall time for collecting, analysing and acting on the data. Business users should be empowered to make these improvements on their own mostly without much help from others. However, often businesses can face limits on reducing the overall time but it may be also due to contradictory desires of the business users.

- *The need for analytic time and expertise for the analysis* should be minimized. It increases the overall time of the analysis significantly.

- *Defining business metrics and goals* should be done clearly and realistically. Unreal expectations of data analytics often lead to business users making misguided efforts to achieve the results.

- *Identifying goals for data collection and transformations* should be done properly. This way businesses can exploit useful data for them. The decision about utilizing

analytics should be done from the start. If the analysis is an afterthought it can possibly limit the value of the results.

- *Distributing analysis results* can be challenging because most of the analysis tools are for quantitative analysis. These results are not in an understandable format for most. Therefore, the results should be translated into supportive visualizations and language that is generally understood. This way businesses can comprehensively put to use the results.

- *Integrating data* from numerous sources is often complex, expensive and challenging. This extract-transform-load process and its effects to data analysis are often underestimated.

These kinds of data analysis applications are designed to meet the growing needs of the business users. Good application should let users analyse data, give rapid insight, and reduce the need for the reliance on analysts. Hence, comprehensive models are more popular than impenetrable. Developing knowhow and user experience in analytics makes business users more demanding. This is due to understanding the expectations of return on investment. Business users, who are not professionals in data analytics, statistics or strategy, can even be impatient with thoughts on how to turn discovered information into profit or how to conclude effects of actions to company's business acts. Data analysis systems can meet these kinds of challenges too. Analytics integrated with businesses' other existing systems make it possible to combine measurements as well as actions. The designed and developed system is complex overall. The whole system combines data collection, storage, processing, and other issues specific to analytics and the business. Business users often need appropriate and concrete analytics. That has turned the interest to specific vertical applications which modify results and interfaces for these. To sum up, data analytics should only and exclusively provide actionable results and the effects of actions should be measurable. Only this way the main goal of utilizing web analytics in

business can be achieved and it is to maximize business value. [26]

5 Research

5.1 Research methodology

The research that was conducted in this thesis was qualitative research. The chosen methodology for collecting data was semi-structured interviews with software developers who have some experience in developing e-commerces. In general, the interviews aimed at gathering and describing the experiences and opinions from software developers' perspective on using analytics in e-commerces. Ultimately, the goal of this data collection was to validate the hypotheses which were originally presented in Section 4.2.

All of the interviews were conducted and recorded online with video-communication services during March 2021. Each interview was then transcribed to enable easier analysis. The amount of interviewees was only four, so the results presented in this thesis may not be generalisable. However, initial conclusions can be drawn from the acquired data and it also provides ideas for potential future research. These ideas include both examining the presented results from different angles and also exploring some extended analytics-related topics even further. The interviewees are listed in Table 5.1. The interviewees represented two different companies in total. The identifiers from the aforementioned table are used to denote the interviewees in the quotes as well. Accordingly, a letter R is used as a reference to the researcher.

Before the actual interview, I introduced the research goal of this thesis and also presented some of the theoretical background on the topics being researched, mainly the web

Table 5.1: Interviewees

	Role	Experience in software development (in years)	Experience in developing e-commerces (in years)
I1	Software Developer	16	4
I2	Software Developer	12	3
I3	Software Developer	2	0.5
I4	Software Developer	14	7

analytics process. Each interview consisted of the same open-ended questions in roughly the same order for each participant. In some scenarios there were also additional questions asked in order to clarify and specify the interviewees opinion on the asked matter. The interview questions are presented in Appendix A and they are examined more in detail in Section 5.2.

The language used in the interviews was Finnish so the interview questions presented here are translations which try to capture the essence of the original question. Furthermore, all of the quotes presented in this thesis are also translations which are kept as close as possible to the original wording. The length of transcript of all the interviews combined was 37 pages. So that is the overall amount of the research data that we have in this thesis.

5.2 Interview questions

As mentioned, the primary goal of the interviews was to gather data which could be used to verify the hypotheses which are presented again in Table 5.2. In the interviews themselves before going through the questions, the interviewees were familiarised with the topic of this thesis, what it tries to answer and find out. Also the theoretical web analytics process presented in Subsection 4.2.1 was introduced to them.

The approach that was taken was to formulate questions which would not directly

reveal the hypotheses. Instead the questions tried to gain a more universal view on how interviewees are using web analytics and how they think it should be used which in turn could be used to validate the hypotheses. Examining the reasoning that was given for the hypotheses in Subsection 4.2 was the way how most of the questions were formed.

For clarification, the questions were also grouped by rough themes but this grouping was never visible for the interviewees and did not really affect the interviews in any way. For each of the hypotheses, a single question can be identified that aims to answer the hypothesis in a more direct fashion. The other questions then try to build the context around those questions and gain a more general understanding of the topic. Next we will go through some of the themes and discover what hypotheses the questions in that theme try to answer and how.

Table 5.2: Summary of the hypotheses

	Hypothesis
H1	A Development Team needs to have the statistical knowledge and skills within itself.
H2	A Product Owner needs to take part in defining KPIs in order to understand them properly and ensure that they are realistic.
H3	A Product Owner needs to utilise the information provided by the web analytics process when prioritising work.
H4	A Product Owner needs to communicate how the Scrum Team's work is expected to be reflected in the web analytics data.
H5	A Scrum Master needs to understand what the different parts of the web analytics process are and why they are essential.
H6	The software developers should have the same access to analytics systems as any other people.
H7	The software developers should inspect the analytics data regularly.

5.2.1 Web analytics process

The first five questions try to gather a general overview of the interviewee's view on the web analytics process. The first two questions ask the interviewee directly about how they are using web analytics currently. After these, Q3 and Q4 dive deeper into how the analytics-related activities are organised. The answer to H1 should emerge with these four questions, especially with the last two of them.

The last question in this group, Q5, is not really related to any hypothesis particularly. It tries to find out how well the model for the web analytics process taken from literature actually matches the way the analytics is used in practise by the interviewees.

5.2.2 Scrum

Questions Q6 and Q7 are focused more on how Scrum and web analytics can be used in conjunction. Q6 addresses how the analytics data can be used when work is planned and prioritised as should be done in Scrum. This sheds light upon H3 but if it does not come up who should be responsible for this action, it can be asked about with a more focused follow-up question.

Even though the different roles and their responsibilities can already come up in the answers to the first five questions, Q7 summarises how well the different Scrum team members need to be acquainted with the web analytics process. Out of the hypotheses, this question should answer H5 especially. In addition to this, it will also bring forth any other roles on which have not been focused in the interview.

5.2.3 KPIs

Questions Q10 and Q11 try to explore the influence and role of KPIs on a Scrum Team's work. By asking these questions we try to find out how visible the KPIs are down at the level which actually implements and builds the site. Q11 addresses H2 directly. If the

Product Owner should participate in defining the KPIs, it should come out in the answers to the question.

5.2.4 Large e-commerce with multiple teams

Q12 and Q13 focus on the scenario where an e-commerce is developed in a large organisation with multiple teams. The former of the questions tries to elucidate what challenges a setup like this imposes on using web analytics as opposed to a smaller and a simpler setup. The latter of the questions, on the other hand, implicitly introduces one of the challenges of a complex setup and tries to find out how to mitigate the issue while simultaneously validating H4.

5.2.5 Software developers

It is clear that Q15 answers H6 directly. The wording of the question speaks about having access to the data instead of the analytics systems but in practice, the data is stored to some system so having access to it means having access to the said system. But in addition to speaking only about the developers having the access, the question also considers other people so it is possible to inspect an even larger group of people from the hypothesis' perspective than what was initially thought.

Q14 aims to find the answer to H7 by trying to find out if there are any activities where software developers can utilise their skills in conjunction with the data to begin with. By examining the nature of these activities, it can be seen whether they are something that are done on demand. If they are, it cannot be said that they should be done regularly since that would not necessarily bring any benefits.

5.2.6 Other questions

The other questions do not really fall into any of the previous groups. Q8 tries to figure out if there are any inherent drawbacks that using web analytics imposes on the development of an e-commerce. Q9 determines whether the trustworthiness, or lack thereof, of the analytics data is an issue in practice. And if it is, the follow-up clarifies if and how these situations are observed by the teams working on the implementation level.

The last question of the interview, Q16, asks whether utilising web analytics differs between e-commerces and other web applications in different fields. The focus of this thesis is on e-commerces. One of the reasons why the scope was narrowed so tightly, was that it was not clear whether the utilisation of web analytics differs between different fields. This question then tries to gain some initial insight into this matter and provide some ideas for discussion and possible future research.

5.3 Results

Here we will go through the results obtained with the interviews. We will examine the results by the same groups as the questions were examined in Section 5.2. Each of the hypotheses will be either validated or disproved in this section. If the hypothesis is disproved, it is not obligatory but can still be a valuable practise. This will be discussed later. A summary of these results is presented in Table 5.3. In addition to presenting the results for the hypotheses, we will also introduce other observations based on the interviews.

5.3.1 Web analytics process

The teams that the interviewees were working in during the time of the interviews included both teams with and without a separate analytics person. According to all of the interviewees, it is extremely beneficial that the team has the analytics and statistical knowledge within the team. It makes the Scrum Team more efficient, self-organising and

Table 5.3: Summary of the primary results

Hypothesis	Results
H1	Beneficial
H2	Beneficial
H3	Validated
H4	Disproved
H5	Disproved
H6	Validated
H7	Disproved

cross-functional thus embracing the team to take ownership of the end-to-end delivery of the implementation. Here is how I4 described the benefits of having an analytics person as a member of the team:

> I4: Yes, I would say that if they are a part of the team, it brings major benefits compared to them being a separate entity. That's because then, for example, the know-how about analytics is transferred much better for the rest of the team. And in theory, it makes the team more efficient. Also when a team is, for example, having refinements and wondering about implementing some feature then all of the information is available then and there. Or at least there's a person who can figure out these things or else someone else should take on the task to discuss these issues with the analytics people, which is at least slower. And I would imagine that when someone like this is present in the team, the correct outcome is reached faster, say we now have the feature, we have the correct analytics and the correct tagging in place. And these very trivial-sounding things, when you have the analytics person as a part of the team, then the testing of the analytics is easier because it could be that the developer doesn't necessarily have access or sufficient understanding of the

> analytics system or what the data should look like there. Then these all checks can be done on the fly. These kinds of benefits – or these are the benefits that I can think of off the top of my head.

In addition to making the team better and more efficient at their job, the added ownership can also increase the motivation of the individuals in the team:

> I1: It is really useful in my opinion. You learn – or there's a better visibility and this whole loop stays in the hands of the team. And not just like that someone outside the teams tells you implement this and that but the team can make a difference so it's really a lot more motivating than having someone come from outside of the team to say things.

Even though the existence of an analytics person in the team is clearly seen as beneficial, we can not say that it would be mandatory or a prerequisite for the team to be able to utilise analytics. It most likely enhances the way the analytics are utilised but nonetheless there were remarks that it is still possible to have the analytics as a separate entity or team even though it is not desirable. And as some interviewees were working or had worked in teams that did not have the analytics knowledge and skills within them and were still able to utilise analytics, we can say that H1 is not mandatory but definitely beneficial.

Observation 1: *To get the most out of utilising web analytics within the development of an e-commerce, this function should not be seen as a separate function from the development of the said site. The best results are achieved when people with different skills work together and communicate with each other which is central to agile development. Utilising web analytics is not an exception to this principle.*

5.3.2 Scrum

It is necessary to also utilise the data when the work is prioritised. And they all agreed that in the end, the whole team should try to contribute to this if possible as the team ought

to have the responsibility of the delivery as a whole. But it is true that the Product Owner has the final say on the priority of different pieces of work so they should utilise the data in this manner at least. So we can say that H3 is validated based on this data.

All of the interviewees actually did utilise the data in prioritisation and planning in their teams already, at least to some degree. Where the utilisation was still lacking in some cases, was prioritising smaller topics compared to larger entities. In some cases the impact of some issues, for example, could be very small yet a lot of effort can be spent on working on those which results in a relatively small return on investment. Here are I1's thoughts on this matter:

> R: If you think about your team's work which is done according to Scrum. How much do you utilise analytics and the analytics data in that planning and prioritisation?
>
> I1: Too little in my opinion. We could do it a lot more when we have bugs which are edge cases of edge cases. I often try to ask about how many really do this and end up in these situations and we do have the data but we don't really look at it. And I personally don't have access to it all and wouldn't necessarily even know how to look for the right data even if I had the access. So there's something we could improve.

Observation 2: *The analytics data could be tied even closer to all of the different pieces of work. This would make it easier to understand the impact and relevance of those topics and spread the overall knowledge on how the site is used and what sort of numbers are dealt with in each occasion.*

Regarding the role of Scrum Master in utilising web analytics, none of the interviewees thought that it would be mandatory for the Scrum Master to understand the web analytics process. Some interviewees agreed that it could be beneficial in some scenarios if the Scrum Master had a general view of the process. But it is not fundamental in order for the Scrum Master to succeed in their role.

Another point that was also brought up in the interviews, was that the role of Scrum Master, in practise, varies between different organisations and teams. Like I4 said:

> I4: (...) somewhere the emphasis of the Scrum Master's work could be more on the software and somewhere else it could be more on facilitating team work.

Because of this variation in the role in the real world, it is hard to draw conclusive hypotheses for Scrum Master. So even though H5 could apply in some scenarios and most likely does not hurt the Scrum Team, we cannot say that it is necessary for the Scrum Master to understand the analytics process universally across all different Scrum Teams.

5.3.3 KPIs

The common line of thought between the interviewees was that the people behind the KPIs can be some kind of business people outside of the Scrum Teams and none of them seemed to think that this is necessarily a bad thing. All of them mentioned that it is beneficial that the people who make decisions about KPIs have cross-functional knowledge. They need to know about the operational environment, the goals of the organisation and what is technically feasible in the first place. If those people do not have all the necessary knowledge, they should gather it by asking for it from the relevant parties, for example, from the Scrum Teams who are responsible for the collection of the data in the end. For example I1 describes their view like this:

> R: And when you think that someone has decided on and chosen these KPIs, then in an ideal situation, who do you think should be involved in making these decisions, like what we're going to measure, what we're going to choose as our metric?
>
> I1: That's an interesting question. I don't know if it would make sense to have everyone making decisions about these KPIs. But instead I think it

> should be those who steer the business to some direction. It's their job to define those metrics from which they can see what direction they are headed towards. Of course I'd hope that they would ask from as many as possible, for example the Scrum Team, whether this makes sense. But the decision is theirs because they need to be able to make decisions with the data on what to do to the business in the grand scheme of things.
>
> R: Yes, so you think it would be good if there was someone else's than those business people's opinion or view on whether something makes sense or is even possible to begin with?
>
> I1: Yes, exactly.

What all interviewees emphasised was that, no matter who decides the KPIs, the people in a Scrum Team should really understand those. This is because without proper understanding of the KPIs, it is hard to make informed decisions which would have the most desirable impact on them. Here is how I3 described this:

> I3: Yes, I think it is good to assimilate the information so that people know what are the KPIs and what is measured. Then everyone who does this job can think to themselves how what they're doing will be reflected on the KPIs. For example if you have to make a decision when you have different ways to do the same thing, then you can easily think whether one of the ways is better from the KPIs point of view.

They all said that the KPIs are almost always behind the work that they are doing at least on some level. However, the KPIs might not always be directly visible in the day-to-day topics that they work on but more often than not, there is some kind of a metric behind the scenes which the topic in question aims to improve. In these scenarios it could be useful to make implicit KPIs explicit. So with this data we can say that it is not necessary for the Product Owner to take part in defining the KPIs. But it most likely would make

it easier to spread the knowledge to the entire Scrum Team and thus H2 can be said to be beneficial.

Observation 3: *Regardless of the context an organisation operates in or who chooses the KPIs to be used with the web analytics, they need to be selected carefully. Even though they might not be particularly visible to the teams working on the implementation level, they likely direct much of the development work. Due to this, selecting them well needs to be paid attention to. If the KPIs do not measure desirable and relevant changes in the site, a lot of work will possibly be done in vain without any gained value.*

5.3.4 Large e-commerce with multiple teams

It was clear in all interviewees that, in a large organisation, communication between different teams is emphasized but becomes more difficult. This is why it needs to be paid attention to. There needs to be coordination between the teams so that the experiments which they are running do not interfere with each other. It is also important that the people who use and monitor the data on a regular basis get the information when something is about the change in the data so that they are prepared for the change. But how all of this is arranged in a team and the whole organisation can vary case-by-case.

A Scrum Team has a natural interface to the rest of the organisation it exists in through the Product Owner. But it is by no means only the Product Owner's responsibility to communicate this. It came through in the interviews that in the end, the whole team needs to ensure that the information is conveyed to appropriate parties. However, the Product Owner often has a major part in this but not necessarily always which is why we cannot validate H4 based on this data. If an analytics person is a member of a Scrum Team, then that person will most likely also be crucial in transmitting this information within the organisation. And even when the team takes care of forwarding the information, there can still be miscommunication issues due to the team just not being fully aware of all of the parties that are using the data in question, like I4 brought up:

R: And then, as a follow-up, if there is a situation where one team changes the data, or the way it is reported, and they know that that it is expected that the data will look different than before, then how should that be communicated to the rest of the organisation around the team?

I4: Yes, that is a challenging situation but the team is responsible for that primarily. And if there is an analytics person in the team, it would be really helpful. Because as they are a part of the team, the knowledge is already transferred automatically and they can then forward the information and think how it will affect other areas where the same data is used. And then in large organisations you can be completely oblivious of some other parties monitoring the data. These will then come up afterwards as some system is alarming somewhere and then it is business as usual that it's hard to reach them. But I do see that it is the team who is primarily responsible for informing about the changes and finding the people affected by the changes. In our case, the Scrum Master is perhaps also forwarding the information on their part.

5.3.5 Software developers

All interviewees agreed with H6 that the developers should have access to the analytics systems. And not only developers, but all people working on the same product should have access to the system. Or that should, at least, be the baseline. There were mentions of things that could change this, for example, security concerns and the expensiveness of having many users in the system mentioned by I1 and I2 respectively. But even in those scenarios where there are some obstacles in the way of giving everyone access to the systems, the interviewees thought that the organisation should somehow overcome or circumvent the obstacles so that people still have access to the data itself.

The developers can utilise the data together with their skill and knowledge in ways that other people cannot. One example of this is, in fact, using the analytics data to help

debug issues on a site. Like I1 describes, it is easier developers to spot certain issues by observing the data:

> I1: Well, a good question. Yes, there are, you can see, for example, error scenarios where the customers have ended up in. Or you can see what they have done before ending up in that scenario which is very useful.
> R: Yeah, and do you think that that is something which can be more easily observed from the data by a developer than an analyst or such?
> I1: Yes, you specifically need knowledge of the code and the site in general to know how it is possible to end up somewhere and what kind of things should happen at certain points. And someone who stares the numbers all day long might not have deep enough knowledge on when the code sends each event that it does.

Despite the fact that the data can be utilised in specific ways by developers, none of the interviewees really saw that it should be mandatory for developers to inspect the data regularly. And none of the interviewees did actually look into the data actively and regularly without a specific need for it. So developers are not the ones who should be regularly monitoring the data. Instead they should use it in situations where they find it could help them overcome issues or discover necessary information.

5.3.6 Other questions

There were a few different drawbacks and challenges that using analytics could impose according to the interviewees. Some of these issues were inherent to web analytics and thus basically unavoidable. However, the benefits that the analytics bring overweigh these inherent issues so analytics should not be discarded because of these. The other issues often arise as a result of poor utilisation or implementation of analytics.

One very clear inherent drawback of analytics is that the implementation will impact the performance and the speed of site. This was brought up by I4 but they also mentioned

that the effect is most likely very small if not completely indistinguishable. However, in modern e-commerces, where even seemingly small performance dips can have a large impact [33], this aspect is something that should not be forgotten completely. Especially if the performance of the implementation is poor, either knowingly or accidentally, this issue could prove to be detrimental.

Another inherent drawback of the web analytics process and A/B testing particularly is the fact that multiple versions of the site with different functionality needs to be built. Naturally this takes more time and costs more than just building one feature each time something new is implemented. Also just storing all of the data and processing probably costs something for the organisation. This is true especially when external third-party systems and tools are used. The benefit of being able to choose the better performing feature should make web analytics usage worthwhile however.

I2 and I4 both spoke about risks which could lead into micro-optimisations. I2 had experienced situations where someone in the organisation could have had some monetary compensation tied to, for example, specific KPIs. This would mean that those persons could then try to see those KPIs improved to the detriment of the overall performance of the site. I4 raised the concern of a scenario where the development of an e-commerce is very fragmented. If many different teams only focus on their own parts of the site and try to optimise those as well as possible, they could potentially lose the vision of the site as a whole. Then even though the individual parts work as efficiently as possible according to their own KPIs, the overall performance might not be as good as possible if the teams have failed to look at the bigger picture.

The last concerns that the interviewees had, were related to security and privacy issues. I2 and I3 talked about the fact that the analytics usage should not violate the users' privacy. They both thought that it would be good if the developers had some knowledge of these matters so that they would notice if there are clear violations but also thought that there would need to be experts who understand the legal side of these topics better. These

experts could then also verify that the analytics data is collected and used lawfully. I4 wondered whether the fact that an organisation uses analytics could potentially harm the image of that organisation and its e-commerce site. As these topics have started to emerge in ever increasing numbers in the last years and people's awareness of these has also grown, the organisation should pay attention to the proper usage of analytics.

All of the interviewees saw that there are definitely similarities in using web analytics between different fields. And it is not limited to only web analytics, the analytics can be used in a similar fashion in any consumer-facing software. In practice, the overall process is probably very much the same in different scenarios but the KPIs themselves can be different. But one thing that was very clear was that the interviewees definitely found the knowledge and skills in using analytics that they have gathered while working on e-commerces to be reusable elsewhere as well. So from a software developers viewpoint, there is not necessarily as big of a difference, which I1 communicated quite clearly:

> R: And now that we've been speaking about using analytics within e-commerces, do you see that there are any substantial differences between utilising web analytics in e-commerces or some other fields?
>
> I1: Not necessarily. The KPIs probably change but on a higher level, you still do the same things. And if you think that as opposed to selling things online you build a product which is used via a browser or locally on a desktop, then it is very beneficial to know, how the users are using the product so that you can notice error scenarios and debug those and, for example, optimise some flows and so on. Remove features that no one is using and these kinds of things.
>
> R: So on a high level the process is quite similar but the details may vary. So if you have been utilising analytics in an e-commerce then the knowledge that you have gained in it, is probably useful somewhere else as well?
>
> I1: Yes, without a doubt.

So the biggest differences are the actual KPIs and those need to be selected suitably. The nature of the KPIs can be simpler and more straightforward in e-commerces as they most likely revolve around money and change in those is directly reflected in an organisations' revenue. This viewpoint was brought up by I2:

> I2: From my own experience I'd say that e-commerces have a more "capitalistic" starting point. They revolve strongly around money and value, in a sense around those shopping cart things. So in general elsewhere you try to enable a smooth user experience for the customers instead of focusing on – well if you're not selling anything then you can't really focus on the monetary value itself. So there is a different angle in e-commerces where you try to get the customer to make a purchase. I think it is in a sense clear that of course it goes like this but e-commerces have a different incentive to use analytics because there is more to gain. The fact that you get someone to spend more money will, of course, be directly visible in the revenue versus you having a service which people are using but not spending any money on, so – well now of course that I mentioned this, it is possible that you're building for example a Software-as-a-Service which doesn't include an e-commerce but the ease of use and others will be visible in the revenue in some other way. So the incentive is there as well but in an e-commerce the link between those is perhaps very clear and easy to measure.

But once good KPIs are identified, understood and selected then the analytics process from that point onwards is quite similar in different fields.

6 Discussion

6.1 Practical implications

Parts of the results from the interviews can be applied to practise or given as valuable guidelines for teams working in e-commerce and wanting to improve their data-driven software development. This chapter gives answers to RQ1 and RQ2 and also gives some partial insight for RQ3. Here is a summary of some guidelines to help enhance data utilisation in teams which develop e-commerce sites.

All the interviewees found it useful to have analytics and statistical knowledge in the team. This is not considered mandatory but possibly extremely useful in practise. It can especially improve efficiency, self-organisation and cross-functionality of a Scrum Team.

Based on the experiences of the interviewees there was some utilization of analytics in planning and prioritisation of the work on a regular basis. However, it was considered a worthwhile practise which should be adopted more as a regular part of the work. As a remark, a Scrum Master does not necessarily need to understand the web analytics process. General understanding of the process is only needed sometimes and therefore should be evaluated case-by-case.

KPIs are often set outside of the Scrum Team and it was considered practical. Most important was that the Scrum Team understands the KPIs and what is measured. This helps them make the decisions based on the KPIs and reflect afterwards on whether their work has been profitable. Consequently, in practise the KPIs should be set in a transparent

and unambiguous fashion and the Scrum Team needs to understand those and strive to improve those via their work.

The communication related to data should be transmitted through the organization by the analytics person in a Scrum Team and/or by the team. Miscommunication issues are common even though communication between teams is emphasized. Communication needs coordination for example to avoid overlapping experiments or having enough time for getting prepared to change something in the data. The coordination can be arranged in many ways depending on the organization. The practises related to communication should be planned properly, clarified for the whole organization and encouraged regularly.

All the interviewees shared the understanding that all people working on around the same product should have access to the analytics system. This should be the baseline. In case of security or privacy concerns or expensiveness which prevents the access, the organization should overcome or circumvent the obstacles to enable the access for the people.

It is not necessary for developers to inspect the data regularly. Generally developers use the data when they have a specific need for it and it can help them overcome issues or discover necessary information on certain occasions. The data can be seen as one additional tool in the developers' toolkit which can prove to be helpful at times. But there is not really value in looking through the data without knowing what you are looking for.

6.2 Limitations

This goal of the research in this thesis was to examine the utilisation of web analytics from a software developer's perspective. While this was achieved, we can not be completely confident that the results hold true in a more general manner. In the end, the amount of interviewees was rather limited. In addition to their low amount, they all represented quite a similar profile as software developers. While we certainly were able to gather

some introductory insights, in order to get a more generalisable view of the utilisation of web analytics in e-commerces, it would be necessary to interview people with more diverse backgrounds.

Even though we presented three research questions for this thesis initially, eventually the focus shifted to the first two. Those two were also used as a basis when the interview questionnaire was formed. We have explored the first question thoroughly from Chapter 4 onwards. We did not uncover any new roles but examined how the different roles that can be identified from software development fit into the utilisation of web analytics. However, we have not studied whether there are any foundational skills an organisation should have in order to be able to start utilising web analytics in the first place.

6.3 Future work

As mentioned in the previous section, it would be beneficial to interview more software developers from different backgrounds to get a better and more solid understanding of this thesis's research topics. It would be interesting to see if the experiences and opinions of the developers vary between different types of e-commerces. One aspect which would also be interesting to find out is if and how these platforms or systems which offer different kinds of e-commerce functionalities out-of-the-box, like WooCommerce or Shopify, affect the web analytics process.

With respect to the first possible future research area, it would also be interesting to widen the role pool to include for example Product Owners, Scrum Masters and Analysts. The people in these roles could see the utilisation of web analytics in a different light which makes acquiring their perspective compelling as well. And as this thesis did focus only on software developers' point of view, we can not make comprehensive conclusions in this thesis. This is of course due to the fact that software developers are not building e-commerces alone but together with cross-functional teams so it is important to have

the opinions of all the different parties involved before making conclusions involving them all. These first two potential future research topics would give us an even more comprehensive answer to RQ2.

Lastly it would be interesting to try to find out if there really are some principles, practises, activities or guidelines which apply on a more universal level than just in e-commerces. The results and answers to the research questions of this thesis give a proper basis for finding these sorts of things in the future. As was brought up in Subsection 5.3.6, the interviewees did not seem to think that the analytics-related actions they were doing would have been unique to e-commerces specifically. So the results of this thesis suggest that there could indeed exist something of this manner which could be a potential future research topic.

7 Conclusions

This thesis introduces the important terms and phenomena for the topic before the experimental part. The thesis had a focus on onsite web analytics and how it can be utilized in software development work. The research that was conducted in this thesis was qualitative research. The chosen methodology for collecting data was semi-structured interviews with software developers who have some experience in developing e-commerces. In general, the interviews aimed at gathering and describing the experiences and opinions from software developers' perspective on using analytics in e-commerces.

Electronic commerce or e-commerce, means conducting traditional commerce through digital technologies. Traditional commerce refers to selling goods and services. In e-commerce these transactions happen online, either individually. One simple description for e-commerce is "an emerging concept that describes the process of buying and selling or exchanging products, services, and information via computer networks including the internet". Due to e-commerce happening via the internet, it has been available for a relatively short duration. It has been available since the opening of the internet to commercial use in the early 1990s. The enormous progression of technology over the past decade has enabled lots of new possibilities for e-commerce. Gary Davis predicted in 2018 that there would be 50 billion devices connected to the internet by 2020 and they have enormous potential to be deployed for increasing e-commerce business. Because consumers have shifted towards trading goods online, traditional commerce has seen a decrease in popularity. On the other hand, increasing demands for e-commerce has also brought new

challenges with it.

The expanding amount of internet users yields more big data which creates more analytics possibilities to understand the behaviour and needs of customers. This generates new business opportunities for companies who have the ability to collect and access online data and analytical competence to understand it. This and the enormously increased online customer volumes have led to e-commerce businesses being data-driven with regards to the development of the e-commerce sites. The development of e-commerce business based on big data and web analytics has a great potential to create a competitive advance in the market. Web analytics is a broad term that covers many tools, techniques and methodologies. Taking advantage of analytics requires human effort to create strategic insights about the activity on the e-commerce site and the behaviour of the users and potential customers.

Agile software development is presented in the thesis. Instead of the traditional plan-driven development, agile software development creates softwares in an iterative and incremental manner focusing at a time on a small set of tasks prioritized by the customer. The point of view of the software developers in the thesis are considered to work by the agile methodologies.

Seven hypotheses were formed based on agile methodologies, the role of software developers in Scrum Teams and utilising web analytics in e-commerce business. The aim of the research was to validate the hypotheses by interviewing software developers who have experience in developing e-commerces. In this study, two of the seven hypotheses were validated, three were disproved and the remaining two were found to be beneficial. Main results were as follows. The analytics person in a team is helpful but not mandatory for the team to utilise analytics (H1: beneficial). Based on the data from the interviews we can say that it is not necessary for the Product Owner to take part in defining the KPIs. But taking part in this makes it easier to spread the knowledge of the KPIs for the rest of the team (H2: beneficial). It is important for the Product Owner to utilise data when work

is prioritized and planned (H3: validated). Scrum Team often communicates with the rest of the organization via the Product Owner but the whole team is responsible for ensuring that information is conveyed to appropriate parties (H4: disproved). We cannot say that it is necessary for the Scrum Master to understand the analytics process universally across all different Scrum Teams (H5: disproved). All interviewees agreed that the developers should have access to the analytics systems. And not only developers, but all people working on the same product should have access to the system (H6: validated). Yet, the developers do not need to be constantly inspecting the data but instead this should happen more on an on demand basis (H7: disproved).

E-commerce is growing continuously and has global impact. The research related to the web analytics usage in e-commerces is novel and there are plenty of interesting research topics. Further research related to this thesis could be performed by interviewing more software developers from different backgrounds to get a better and more solid understanding of this thesis's research topics. Additionally, it would be interesting to see if the experiences and opinions of the developers vary between different types of e-commerces and it would also be interesting to widen the role pool to include for example Product Owners, Scrum Masters and Analysts. That would help to make more comprehensive conclusions around the topic.

In the future, some principles, practises, activities, or guidelines could be recognized which apply on a more universal level than just in e-commerces.

References

[1] Laurie Williams and Alistair Cockburn. Agile software development: it's about feedback and change. *IEEE Computer*, 36(6):39–43, 2003.

[2] Manifesto for agile software development. `https://agilemanifesto.org`. [Online; accessed 13-May-2021].

[3] Principles behind the agile manifesto. `https://agilemanifesto.org/principles.html`. [Online; accessed 13-May-2021].

[4] Jim Highsmith and Alistair Cockburn. Agile software development: The business of innovation. *IEEE Computer*, 34(9):120–127, 2001.

[5] James A Highsmith and Jim Highsmith. *Agile software development ecosystems*. Addison-Wesley Professional, 2002.

[6] Lowell Lindstrom and Ron Jeffries. Extreme programming and agile software development methodologies. *Information systems management*, 21(3):41–52, 2004.

[7] Kent Beck. *Extreme programming explained: embrace change*. Addison-Wesley Professional, 2000.

[8] Kent Beck. Embracing change with extreme programming. *IEEE Computer*, 32(10):70–77, 1999.

[9] Visual Paradigm. What is a sprint in scrum? https://www.visual-paradigm.com/scrum/what-is-sprint-in-scrum/. [Online; accessed 10-May-2021].

[10] Rashina Hoda, James Noble, and Stuart Marshall. Self-organizing roles on agile software development teams. *IEEE Transactions on Software Engineering*, 39(3):422–444, 2012.

[11] Ken Schwaber and Jeff Sutherland. Scrum. *URL: http://www. scrumalliance. org/system/resource/file/275/whatIsScrum. pdf,[Sta nd: 03.03. 2008]*, 2010.

[12] Viljan Mahnic and Slavko Drnovscek. Agile software project management with scrum. In *EUNIS 2005 Conference-Session papers and tutorial abstracts*, page 6, 2005.

[13] Julian M Bass. Scrum master activities: process tailoring in large enterprise projects. In *2014 IEEE 9th International Conference on Global Software Engineering*, pages 6–15. IEEE, 2014.

[14] Pekka Abrahamsson, Outi Salo, Jussi Ronkainen, and Juhani Warsta. Agile software development methods: Review and analysis. *arXiv preprint arXiv:1709.08439*, 2017.

[15] Hrafnhildur Sif Sverrisdottir, Helgi Thor Ingason, and Haukur Ingi Jonasson. The role of the product owner in scrum-comparison between theory and practices. *Procedia-Social and Behavioral Sciences*, 119:257–267, 2014.

[16] Ken Schwaber and Jeff Sutherland. The scrum guide. *Scrum Alliance*, 21:19, 2011.

[17] Ken Schwaber. *Agile project management with Scrum*. Microsoft press, 2004.

[18] Judy Strauss and Raymond Frost. *E-marketing*. Prentice hall press, 2008.

[19] S.K. Mourya and Shalu Gupta. *E-Commerce*. Alpha Science International, 2014.

[20] Shahriar Akter and Samuel Fosso Wamba. Big data analytics in e-commerce: a systematic review and agenda for future research. *Electronic Markets*, 26(2):173–194, 2016.

[21] G Sreedhar. *Improving E-commerce web applications through business intelligence Techniques*. IGI Global, 2018.

[22] Kenneth C Laudon, Carol Guercio Traver, et al. *E-commerce: business, technology, society*. 2019.

[23] Gary Davis. 2020: Life with 50 billion connected devices. In *2018 IEEE international conference on consumer electronics (ICCE)*, pages 1–1. IEEE, 2018.

[24] Brian Clifton. *Advanced web metrics with Google Analytics*. Sybex, 2010.

[25] Layla Hasan, Anne Morris, and Steve Probets. Using google analytics to evaluate the usability of e-commerce sites. In *International Conference on Human Centered Design*, pages 697–706. Springer, 2009.

[26] Ron Kohavi, Neal J Rothleder, and Evangelos Simoudis. Emerging trends in business analytics. *Communications of the ACM*, 45(8):45–48, 2002.

[27] Guangzhi Zheng and Svetlana Peltsverger. Web analytics overview. In *Encyclopedia of Information Science and Technology, Third Edition*, pages 7674–7683. IGI Global, 2015.

[28] Daniel Waisberg and Avinash Kaushik. Web analytics 2.0: empowering customer centricity. *The original Search Engine Marketing Journal*, 2(1):5–11, 2009.

[29] Google. Add gtag.js to your site — universal analytics for web (gtag.js). `https://developers.google.com/analytics/devguides/collection/gtagjs`, 2020. [Online; accessed 9-November-2020].

[30] Wall Street Journal. Who is the mystery shopper leaving behind thousands of online shopping carts? - wsj. https://www.wsj.com/articles/who-is-the-mystery-shopper-leaving-behind-all-those-online-shop 2020. [Online; accessed 3-January-2021].

[31] Matt Cutler and Jim Sterne. E-metrics. *Business Metrics for The New Economy, Chicago: NetGenesis*, 2000.

[32] Ron Kohavi and Roger Longbotham. Online controlled experiments and a/b testing. *Encyclopedia of machine learning and data mining*, 7(8):922–929, 2017.

[33] Akamai Technologies Inc. Akamai online retail performance report: Milliseconds are critical. https://www.akamai.com/uk/en/about/news/press/2017-press/akamai-releases-spring-2017-state-of-online-retail-performance-jsp, 2017. [Online; accessed 8-May-2021].

Appendix A Interview questions

The web analytics process in general

Q1: How do you utilise web analytics in software development?

Q2: How is web analytics reflected in you day-to-day work?

Q3: Does the decision-making regarding analytics happen within your team or somewhere outside of it? Pros or cons in either approach?

Q4: Who in the team needs the analytics or statistical knowledge? Who plans the A/B testing and analyses the results in your team?

Q5: Does the presented web analytics process match the way you're using analytics in practise? If not, are there any similarities?

Scrum and the web analytics process

Q6: Can the analytics be utilised in planning and prioritising work?

Q7: Which (Scrum) team members should understand the different parts of the web analytics process? Are there any downsides to someone not understanding these?

Challenges

Q8: Are there any fundamental drawbacks that using web analytics inflicts? What about other non-fundamental drawbacks?

Q9: Can the analytics data itself be distorted or less trustworthy in some situations? Are

these kinds of situations recognised?

KPIs

Q10: Do the existing KPIs affect your team's work and decision-making somehow?

Q11: Who should take part in defining the KPIs and who should understand those?

Large e-commerce with multiple teams

Q12: Do you see any challenges in a scenario where multiple teams work on the same product and the same data but utilise analytics independently?

Q13: How should the expected changes in analytics be communicated outside of the team?

Dev activities

Q14: Are there any other ways that software developers can utilise the data (than the primary process which was presented here)?

Q15: Should the analytics data be available to everyone?

Differences between e-commerces and other fields

Q16: Do you see any differences between utilising web analytics in e-commerces or some other fields?