

Laura Halén

**A CHANCE FOR THE COMMUNITY TO  
AFFECT GAME DEVELOPMENT**  
Incorporating player feedback to game development  
during Early Access

# ABSTRACT

Laura Halén: Incorporating player feedback to game development during Early Access  
Master's Thesis  
Tampere University  
Master's Degree Programme in Game Studies  
May 2023

---

The goal of this thesis is to research how Early Access affects game development, and how player feedback affects the development and design of the games in question. In this thesis I will explain why Early Access is used (or why it might not be the right development choice), how developers handle transparency during it, which methods are used in communicating with the audiences and what types of feedback is implemented in the development and design of the game. I will explain how the players could affect the game design in the case of my research participants, and how Early Access affected the development process of those participants.

Background research was done in an exploratory way, using both academic and non-academic sources, such as game developer interviews available online. After forming a base where I cover Early Access in general, and where I explain it as a part of a game development process, I conducted my own research using a survey and interviews. I then analyzed the data with a thematic analysis. Combining the survey answers and the interviews, a total of eight participants took part in this research. All of them are game developers, who have worked with an Early Access title. Thematic analysis was then used to raise themes from the data, and five themes were selected. These themes cover the reasons to use Early Access, strategies to give out information and gather information from the players, and incorporating gathered feedback to the development and design of the game.

Early Access was felt to be a good development choice when the developers were uncertain about some aspects of their game, or if they wanted to get feedback from the players to make the game better suited for players. Other reasons for using Early Access were mainly funding and marketing. All of the participants agreed that feedback was one of the main reasons to use Early Access, and thus they all had systems to communicate with their players. Different tools for transparency and informing players are listed in this thesis, as well as the ways the participants gathered feedback. Lastly, I discuss what type of feedback was incorporated to the game designs of the participants' games, and what limitations the developers had (i.e., what feedback was not taken into consideration).

According to this research, Early Access is a viable solution for some game developers and for certain types of games, but it requires a lot of work and transparency. With Early Access the game in development might be better suited for the players after the release, and some participants said that Early Access is the 'right way' to develop games. However, that requires commitment to communicate with the players. The developers need to adjust to the open development practices and set clear limits and restrictions for preventing things like tiredness, burnout, or work and personal life mixing. If executed correctly and managing the communications with the players in an orderly manner, Early Access is a suitable solution in creating games with the players, and the players have a chance to affect the game in a direction they would want it to go.

Keywords: Early Access, video games, game development, game design, open development, community, player feedback

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

## **Preface**

This master's thesis has been a journey, as my life has had its ups and downs during writing this. Sometimes this was the hardest thing to tackle, but then again, sometimes I welcomed the calmness of the familiarity of the work and its challenges. I have learned a ton researching and writing this, so huge thank you for all the research participants for sharing their experiences and workflows with me. I hope this thesis gives something back to you too.

I would also like to thank the people close to me, friends, and family. Thank you for all the support I have received through the hard times, and thank you for sticking with me through the three most difficult years of my life. I simply could not have done this without you.

Tampere, 6 April 2023

Laura Halén

## Table of Contents

1	INTRODUCTION .....	5
2	CONCEPTUAL STARTING POINTS .....	8
3	BACKGROUND THEORY .....	11
3.1	Background theory research methodology .....	11
3.2	Background theory content analysis.....	12
3.2.1	Always evolving game industry .....	13
3.2.2	History of Early Access .....	15
3.2.3	Game development process.....	17
3.2.4	Pros and cons of Early Access .....	22
4	RESEARCH METHODOLOGY .....	26
4.1	Collection of data .....	26
4.2	Research data and participants .....	29
4.3	Thematic analysis .....	30
5	RESULTS AND ANALYSIS .....	33
5.1	Reasons to use and not to use Early Access .....	33
5.1.1	Reasons to use Early Access .....	33
5.1.2	Reasons not to use Early Access.....	37
5.2	Methods of giving out information and gathering feedback .....	41
5.2.1	Strategies to give out information to players during Early Access....	41
5.2.2	Strategies to gather feedback from the players during Early Access..	45
5.3	Incorporating player feedback .....	51
5.3.1	Incorporating feedback into the game development and design.....	51
5.3.2	Limitations to incorporating player feedback .....	55
6	CONCLUSIONS AND FUTURE WORK.....	60
6.1	Key results .....	60
6.2	Brief summary and guidelines.....	63
6.3	Future work .....	65
	REFERENCES .....	67
	APPENDIX 1. INTERVIEW QUESTIONS .....	71

# 1 INTRODUCTION

In 2013 Steam launched its Early Access program, which enables game developers to publish and sell their games while they are still under development. As Steam is the largest game distributor in the world (Edwards, 2013), this enabled many game companies to profit from the program. Early Access has had its ups and downs, as some of the games developed in Early Access have had enormous success (for example, *Don't Starve*, *Subnautica*, *Darkest Dungeon*, *PlayerUnknown's Battlegrounds*), and some have had Early Access phases that disappointed the players and developers alike (for example, *DayZ*, *Last Oasis*). At its best, Early Access offers early funding, semi-open development, feedback from the actual target audience and a possibility to fine tune the game to suit the player base the best it can. The flip side of the coin is development teams being overwhelmed by constant feedback, using Early Access just as a funding option, or in worst case scenario; abandoning the whole project after getting a bit of money. The evolution of Early Access is still on-going, but nowadays the projects are more discussed and regulated, and players tend to know what to expect when they dive into a fresh Early Access title.

Early Access development is a topical subject, since game development has seen a shift to more participatory open development. Other technological advancements such as social media have enabled players to give feedback to the developers in real time, and in a sense, participate in the games design and development process. This has enabled game developer studios to create games more suited to the players of their games. Publishing platforms for games, such as Steam and Epic Games, have also enabled this shift to publish games in an early state, since they have their own Early Access programs that enable players to give feedback to the developer straight on the platform. To give some examples about the number of games that go through Early Access in Steam, in 2021 well over 1000 games launched with the tag "Early Access" (Daro, 2021). At the time of writing, 18.11.2022, there are 184 games in Early Access in Steam.

My initial interest in the subject comes from my background working with an Early Access title. That project taught me a lot, since at first, I felt like we were quite lost on what tools to implement to gather feedback, how to collect it and how to then apply it. The entire process felt a bit messy and overall was not well organized, and that piqued my interest in how other studios handle Early Access projects. There were things in the game that were changed, fixed, or balanced due to the feedback we got from the players,

but in my case the studio felt quite strongly about the core design, and thus the design was mostly left unaffected by the players' feedback. I find myself being curious whether this is also the case in other game projects as well, and how the developer control over the game is seen.

This thesis is written from the perspective of game developers, as the data gathered comes from developers themselves. I will put together a rough summary on why Early Access is relevant and matters, why it might be a good way to develop games and what kinds of tools game developers use to handle the process. I hope to shed light on what kind of feedback is gathered and implemented in the game development and design of Early Access games, and what are the limitations that Early Access brings with itself. My hope is to give back to game developers, and I wish that this thesis can be of assistance to future Early Access projects.

This leads to my research question: How does Early Access and player feedback affect game development and design? I wanted to learn how game developers handle the Early Access process; how (and what) information is displayed for players, how feedback is gathered, what parts of that feedback is brought to the game development and does it affect the design of the game. The main research question was divided into three sub-questions to help get a better grasp of the main question. These questions are:

1. Why do game developers use Early Access?
2. What methods do the developers use to give out information to the players, and what methods do they use to gather feedback in order to incorporate it to the game's design and development?
3. On what level is the player feedback taken into consideration and included in the design and development process, and what are the limitations?

The structure in this thesis is as follows: in the second chapter, "Conceptual starting points" I will go through how Early Access is commonly defined and not defined, how the term is understood and how it is used in the game development field.

In the third chapter, "Background theory", I have used exploratory research methodology for overviewing the previous works and background about Early Access, to better understand the Early Access development of games. I dive into Early Access projects and explain what makes them work, what are the parts needed and what are the pros and cons

of Early Access. I will explain what Early Access, open development and traditional game design are, and why Early Access is often seen as an effective way for the developers to gather feedback and adjust their game to be more suited for their future players. I will explain the history of Early Access and how it differs from traditional game design, and why open development and participatory design might be effective ways to create a game.

The fourth chapter is about the methodology that I use for my own research. I used a qualitative survey and semi-structured interviews to research game projects in Early Access and the developers' workflows during that time. I studied how player feedback affects game development and design, what are the limitations of said feedback, and what tools were used to gather and implement information and feedback. I will explain why I chose these methods and what I learned through the process. I will also explain how I used thematic analysis to code and analyse the data I gathered. After coding, the five themes that I compiled from the data were:

1. Reasons to use and not to use Early Access
2. Strategies to give out information to players during Early Access
3. Strategies to gather feedback from the players during Early Access
4. Incorporating feedback into the game development and design
5. Limitations to incorporating player feedback

In the fifth chapter I will go through the data and themes I gathered for my own research and discuss the results that I got from my study. The chapter is divided by the three research sub-questions I had, and in each of those sections I will go through the distinct themes I had. The first theme will answer the first sub-question, themes two and three will answer the second sub-question, and themes four and five will answer the last sub-question.

In the last chapter, the sixth one, I will conclude my research by summarizing the results and discussing the limitations that I had, as well as discussing the future work that can be done in this field.

## 2 CONCEPTUAL STARTING POINTS

Early Access is a phase in game development, where the game is released for the public to play, while still in an unfinished but playable state. This concept is also used in other fields such as software development, but they are not relevant here, as games differ from these other fields. The concept is useful for both game developers and players, as it is a phase in game development where the consumers can affect what the final product will include.

There are a couple of nominal definitions for Early Access. I have gathered a few of the most used ones here, starting by the most academic one. Sandqvist (2015, p. 16) defines Early Access as “Companies sell the game before the actual release and let gamers get access to the unfinished game”, and continues by saying that

By selling an unfinished game, the company can get revenues long before the game is finished. - - A positive benefit of this model is also that the company gets free game testers that can find and report problems and bugs in the game. If the users are invested in the game they might even support the company by giving suggestions about the game, such as potential desired changes and new features. The company gets valuable feedback that could otherwise be hard or expensive to acquire. (pp. 16-17)

However, as Early Access is quite a new practice, there are no clear structures, limitations, or guidelines to it. Early Access is widely described in a similar manner as what Wikipedia (“Early Access”, 2022) gives:

Early Access is a funding model in the video game industry by which consumers can purchase and play a game in the various pre-release development cycles, such as pre-alpha, alpha, and/or beta, while the developer is able to use those funds to continue further development on the game. Those that pay to participate typically help to debug the game, provide feedback and suggestions, and may have access to special materials in the game.

Steam, one of the biggest advocates for Early Access, gives a different point of view for players (Valve, 2022b): “Get immediate access to games that are being developed with the community's involvement. These are games that evolve as you play them, as you give feedback, and as the developers update and add content” Steamworks Documentation for developers is quite similar, but more precise (Valve, 2022a);

Steam Early Access enables you to sell your game on Steam while it is still being developed, and provide context to customers that a product should be considered "unfinished." Early Access is a place for games that are in a



playable alpha or beta state, are worth the current value of the playable build, and that you plan to continue to develop for release.

At the same time, Steamworks also gives a non-definition for Early Access, as it states that Early Access is not “a way to crowdfund development of your product,” which is clearly in conflict with what was stated in Wikipedia. Google gives a very broad definition for its Early Access products (Google Play Help, 2022): “Early access apps are apps that haven’t been released yet” but does not include the ‘access’ part of Early Access in the definition itself.

So, at least now, Early Access is still a concept that changes itself to suit the party that uses it. The use of Early Access started as a crowdfunding method, but after a while the developers realized that getting players to give actual qualitative feedback can make the game more suited for the players (Goslin, 2019). Early Access differs from the alpha or beta phases by the connection between the game developers and the players, as both parties communicate to make the game better, not just collecting data to fix the major bugs. Alpha and beta phases can be used overlapping the Early Access, since the game development usually follows those phases in the development of the game itself. Instead of giving my own definition of Early Access, I wanted to see how my own research participants viewed their own Early Access processes and experiences, and how they define the term Early Access. I will explore this in the future chapters.

As the term “Early Access” is still quite hazy, it might be difficult to measure it. When does the Early Access phase start, and when does it end? Is Early Access successful if a certain amount of money is made during it? Most of these measurements are made by the game development company itself or the publisher of the game. However, the development team usually makes these measurements public when entering the Early Access phase, as many developers decide to be transparent about their progress (Thominet, 2018). The development team might make certain goals for themselves that they want to meet during the phase, or a timeline for finishing certain features. Usually, Early Access phase is at the point where the game is playable, but there is still room for the changes that the player base suggests. The phase ends in the final release of the game, when the goals of the game development company are met, and the game is officially released. After the game is released, it is no longer in the Early Access phase. After that, the game usually no longer goes through big changes in the core features, but might get smaller updates and add-ons.

Even though the term is still a bit hazy, I think that it is important for the players and the developers to know when to use it, and what it means in the specific case when it is used. The players who buy the game in Early Access need to understand that the game is not finished, and that they are paying money to play an unfinished game. Therefore, I think it is good that platforms like Steam or Epic Games clearly state which games are in Early Access and provide a place for the players to give the needed feedback. Platforms like Steam state clear guidelines when the game is ready for Early Access, so the development team knows when they are ready to enter it, and the players know what they are getting.

## **3 BACKGROUND THEORY**

In this background theory section, I will discuss my explorative background research about Early Access. I based my own research on top of the results that I got from this background research, and in chapter 5 I will discuss the findings and how they compare to the background theory presented in this chapter. I will first discuss the methodology I used to gather the background theory data, and how I processed it. In the latter section I will then open the results and analyse them in a categorized manner.

### **3.1 Background theory research methodology**

My research materials consist of academic articles, books, Steam's own documentation and non-academic articles and interviews. As there are only few academic sources about Early Access and its effect on game development and design, I have had to rely on non-academic sources as well, such as interviews with industry professionals, and guides and tips that they have provided to the public. The lack of academic sources warrants an explorative study into the non-academic sources to map out the phenomena. Explorative research is often used when the subject of the study has not been studied in depth, and there are no conclusive academic sources, thus it does not strictly build on previous research. It allows the researcher to be creative and gives an opportunity to involve materials outside of the academic scope that is normally used, to maximize the discovery on the topic and to better understand it (Babbie, 2016; Stebbins, 2011).

This explorative study has been conducted in qualitative methods, such as watching and reading interviews, and then conducting content analysis for that material, in addition to literature review for the existing academic materials. Thus, my background research will rely both on academic and non-academic sources. This has then resulted in narrowing down my research question. A good thing to note is that the interviews that are a part of this study might have some biases, such as giving an interview for publicity reasons, hiding information that makes the company look bad, or giving out a positive image about Early Access because it suited that particular game project. However, the information gathered from interviews and articles is still valuable, especially when common themes arise from multiple sources.

To analyse the material gathered, I used content analysis to collect different themes to categorize the material to fit my study. According to Schreier (2012, p. 1), "Qualitative

Content Analysis is a method for describing the meaning of qualitative material in a systematic way.” As content analysis can be used to analyse all kinds of materials, such as written documents, photographs, video, and audiotapes (Berg, 2017), this was a useful method for me, since my materials consist of both written documents and video interviews. Content analysis is a way to determine the presence of certain words, themes, and concepts in the research material. The researcher looks for these in the texts and then codes them to categorize the whole and derive meaning from the materials (Berg, 2017). Berg (2017) categorizes the process as such in his book *Qualitative research methods for the social sciences*:

- a. Data is collected and made into text or otherwise organized to be “read” (e.g., field notes, transcripts, image sequences, news reports). The same applies to visual materials, such as photographs, drawings, cartoons, cartoon strips and graphic novels, film, or architecture.
- b. Codes are analytically developed and/or inductively identified in the data and affixed to sets of notes or transcript pages.
- c. Codes are transformed into categorical labels or themes.
- d. Materials are sorted by these categories, identifying similar phrases, patterns, relationships, and commonalties or disparities.
- e. Sorted materials are examined to isolate meaningful patterns and processes.
- f. Identified patterns are considered in light of previous research and theories, and a small set of generalizations is established. (p. 184)

As I gathered and coded themes from this first dataset that includes all the interviews, books, academic materials, and documentations, I ended up categorizing the material under these large themes: “History of Early Access,” “Game development process” and “Pros and cons of Early Access.” I will discuss all these themes and explain them more in depth in the “Background theory content analysis” chapter. This forms the basis for my own study.

### **3.2 Background theory content analysis**

In this analysis I will discuss the game industry as a whole, and its always evolving practises. This is to give a base for understanding Early Access and for understanding why it is used in the game industry. I will then briefly discuss the history of Early Access and how it came to be. Then I will touch the differences between traditional game development, fully open game development, and how Early Access fits between the two.

Lastly I will discuss the pros and cons of Early Access that were raised in the background research data.

### **3.2.1 Always evolving game industry**

Since the beginning of game industry, one of the traits of game production has been its rapid evolvement. Game production might have started as small studios and developers creating demos and coding in their parents' basements (Kuorikoski, 2014, p. 36), but with games becoming more and more popular, it expanded to larger, more successful game companies. In the 90s both games and hardware used in gaming evolved rapidly, and in the 2000s the social aspect made its way to gaming as the internet became more widely accessible (Engström, 2020, p. 29). The game development field evolved to bigger, more structured companies. These companies produced games that were financially successful, and thus game development companies became actual job opportunities, which then formed the gaming industry.

As game development studios did not want to share their processes or tools with other companies, or sometimes even with other departments of their own company, they created games behind closed doors in secrecy. This secrecy then became an obstacle keeping the developers back and making the development of the games harder, as information did not travel as it should have (O'Donnell, 2014, p. 42-45). However, as the field is developing rapidly, after some years the industry once again evolved and shifted. The flow of game development was moved back to a more open and inclusive form, as small indie developers started to share their knowledge and explore the options of outsourcing (Keogh, 2019).

Game development has the same roots as software development, and thus the game industry is also evolving as the economy and technology evolves around it. For example, things like the emergence of social media have affected the way games are made (Engström, 2020, p. 45; Keogh, 2019). From the days when games were distributed on floppy discs, we have moved to games as digital services distributed digitally all over the world. Games are accessible to anyone who owns a console, a PC or even just a mobile phone. Instead of buying a game, the consumer can subscribe to a service and play as many games as they want as long as the subscription is ongoing. Games are also somewhat expected to be updated after their release, so the cycle of game development does not end when the game is launched anymore (Weststar & Dubois, 2022) and the

amounts of updates and customization can be considerable (Kerr, 2017, p. 16). The rise of new types of games, monetization models, online platforms and distribution channels are something the industry has had to adapt to, and while some companies thrive, some are struggling to keep up. Even though there have been challenges, the game industry has managed to grow into an industry with significant revenues, worldwide production networks and regionalized markets (Kerr, 2017, pp. 2-4).

The evolution of games and the industry has given space for all kinds of games, development studios, as well as all types of processes to create games. In addition to this, as game development has many sides to it, such as software development, cultural and social aspects and the business side, games have generated workplaces outside of the core game development as well. Tools and workforce such as publishing, distributing, data analysing, community management, 3rd party platforms, 3rd party development tools, streaming, e-sports, academic research, and much more (Engström, 2020, p. 22; Keogh, 2019; Kerr, 2017, p. 3; Sotamaa & Svelch, 2021, pp. 16-17) are an essential part of the game industry ecosystem and the culture around it. Games have slowly but surely become a ubiquitous part of our society (Engström, 2020).

Once in a while, new game development practices rise and take place in the gaming world, which takes the industry forward and makes it evolve. For example, smaller indie studios sharing their practices and outsourcing some of their processes made the industry more open (Keogh, 2019), 3rd party game engines like Unity enabled game development to reach wider audiences (Engström, 2020, p. 36), crowdfunding rose to be a valuable funding option (Engström, 2020, p. 125), downloadable content and patches that created a games-as-a-service model (Weststar & Dubois, 2022), and open betas and Early Access have helped to bring the players closer to the development process (Goslin, 2019). New advancements in technology and the accessibility of the internet enabled distribution platforms like Valve's Steam to open shop, which then led to more self-published games (Engström, 2020, p. 130; Kerr, 2017, p. 14).

It seems that the development of the game industry and its practices, such as game development, are in constant movement and are always evolving. Early Access has been a fairly new development, where the developers of games are trying to involve players in an earlier phase to the development of the game, in order to create games more suitable for players (Goslin, 2019). Early Access has had its ups and downs, but it has been increasingly popular amongst developers, especially smaller indie developers (Lin et al.,

2018). It has enabled the players to bring forth feedback and participate in the development of the game, which may have led to the industry to be more open and receptive to feedback in the development phase, or even after the launch of the game. These things have also been brought up through other innovations as well, such as games moving toward games-as-a-service, where the end user is in the center and games are created this in mind (Weststar & Dubois, 2022). These evolvments however have brought on challenges of their own, such as developer companies needing to be experts in communications, react to new changes to the product in a short time frame and it makes developers directly accountable to players (Weststar & Dubois, 2022), which in turn might bring risks of crunch periods and burnouts (Kerr, 2017, p. 17). I will explain and open these propositions in the following chapters.

### **3.2.2 History of Early Access**

Early Access is a strategy in publishing games, where consumers are able to purchase and play the game in an unfinished state, and in turn, influence the game development process by giving developers feedback on the game. The developer company can decide for themselves when the game is ready for Early Access, but it must be at least playable (Valve, 2022a). Early Access model has become increasingly popular (Lin et al., 2018), as digital game distribution platforms, such as Steam, have enabled its growth.

When other software development started to use public and open beta-releases, Valve Corporation (owner of Steam) also realized the potential of those builds. Public beta-releases were used to meet user requirements better, which means that an early build of the software was released to a small audience of users to collect feedback and to make adjustments before the final release (Arafat et al., 2019). Steam launched its Early Access program in March 2013, and it allowed games to be released as public betas, when the development studio was still working on the game. However, the release through Steam Early Access program was not for small audiences, but it was public for anyone interested in trying out the game. This also allowed players, the core target audience, to give feedback on bugs and the game overall straight to the developers (Lin et al., 2018). This is of course very valuable, since players that invest their time and effort in giving feedback to the developers do so out of their personal entertainment, not because they are paid for doing it. In most cases the player actually needs to buy the game in question, though at a reasonable price. This means that the feedback given by the players is even more valuable, since it is sincerely trying to make the game better (Arafat et al., 2019).

Feedback has been titled many times as the core feature of Early Access (e.g. Arafat et al., 2019; Elahee, 2019; Lin et al., 2018; Sansone, 2014; Sherlock, 2014; Thominet, 2018). Growth of publishing platforms such as Steam and Epic Games, and the expansion of social media in general, has enabled game developers to gather feedback in multiple ways to make their games better suited for their players. There is an abundance of possibilities for game developers to communicate with the player base, and vice versa. During the Early Access phase, it is important that the developers are also transparent and give out information about their game, their goals, their update schedule, and so on, since those are the kinds of things that keep the audience in the development loop and makes it possible to gather feedback from the players in the first place.

Including players' voices in their projects has made Early Access games even more popular, since the audience feels like they are heard and that they can influence the design of the game, when at the same time they get to access and experience the game before its full release (Lin et al., 2018; Arafat et al., 2019). Seeing their feedback implemented in the game also gives players satisfaction, as they can see the progress made and witness the changes to the gameplay (Arafat et al., 2019). Player satisfaction is of course a significant thing in the market for games, as games with high player satisfaction tend to sell well, and the complimentary reviews travel between players in "word of mouth" style, marketing the game forward. Games in Steam Early Access program have on average higher player reviews during the Early Access stage than other games, as players are more likely to endure through bugs in the development phase and are happy that they have been able to contribute to the game and its full release (Lin et al., 2018).

One of the features making Early Access so successful and popular is the funding aspect. Even though the game is not finished yet, the developers can fund the rest of the development through the raised funds that the game brings while it is in the Early Access phase. It is a solution where everyone benefits as the developer gets to continue the development with the money Early Access brought and gets valuable feedback from the core audience, and the players get to access the game earlier and give feedback on it (Arafat et al., 2019). However, the Early Access model has some downsides to it as well, such as games never leaving the Early Access stage and the developers leaving the project before its final release, but I will discuss this in more depth in the "Pros and cons of Early Access" chapter.



### **3.2.3 Game development process**

Game development process can be roughly divided into two different ways of creating games, “traditional game development” and “open game development”. Traditional game development is a closed development type, where the developer creates a game within the company limits, without any help from players or other audiences. Open game development is the other end of the spectrum, where the whole development is open for audiences from the get-go. Those audiences can then influence the game development process from the beginning. Between these two types of developing games is Early Access, which starts as traditional game development and is later on opened up for players to influence the game development process.

#### **Traditional game development**

Traditional game design is a closed process, that can be divided into different parts. During these parts, the developer team goes through various stages of development, which can according to Aktaş and Orçun (2016) and Thominet (2018) be categorized into six distinct categories. These are: 1. idea/concept, 2. pre-production stage, 3. production/development stage, 4. post-production stage, 5. release and launch stage and 6. post-release/maintenance stage.

In the idea/concept stage the developers craft the game concept and key features of the game. These include the game's genre, intended target audience, platform, references, and the game's general idea. In this stage the developers write concept documents and the development plan. Necessary tools for creating the game are built or acquired. The pre-production stage includes things such as creating game and level designs, the story and scenarios, game mechanics, aesthetics, development principles and the design of the game world. This stage might also include a prototype of the game, that captures the key features intended to go to the final game. As Casey O'Donnell puts it in his book “Developer’s Dilemma” (2014, p. 27), these are the “underlying systems and structures” of the game. Production/development stage includes the development of all different elements that go into the game, such as coding, modelling, graphics, sounds, videos, text, and the actual integration to the game. The production stage can be further divided into two different milestones: alpha and beta. Alpha milestone usually consists of creating a functional and playable game. Beta milestone includes fleshing out the game and creating the rest of the content. The post-production stage is for testing and validation, which

includes quality assurance testing (QA-testing). QA-testing focuses on mechanics, playability, gameplay, user interface and the quality of audio-visual content. This is done to see whether the game meets market requirements. Post-production also includes debugging and fixing game balance and other issues that were found during QA-testing. The game can go through another beta stage after QA-testing, and in this stage some games go through “beta-testing,” which means releasing the game for a small audience to further notice and work out the kinks of the game. After polishing the game, it is time to move to the release and launch stage, which means that the game is released on chosen platforms through chosen distribution channels. Lastly, the post-release/maintenance stage includes things like developing patches, upgrades, and expansions.

However, as Sansone (2014, p. 109) puts it, “Development is changing rapidly, and designers are creating progressively more detailed and refined versions of the game that they have actual players try early and often to find out what does and does not work.” As the game development cycle has previously been more secretive and happened behind closed doors (O’Donnell, 2014, p. 41), the evolution of game development is overall shifting towards more open development practices. Many consider Early Access as it seems to be in the middle of traditional game development and fully open game development. This has opened the possibility for players to affect the development process and make the games more enjoyable for the player, and at the same time the model is more profitable for the developers (Sansone, 2014).

### **Open game development**

The other end from closed game development is open game development. Open development has many other names as well, such as participatory design, transparent development, performative development, community-informed development, and crowdsharing (Thominet, 2018). Open development is often described as a development that happens simultaneously as the product is already on sale or otherwise attainable, and the players of the game get to have some level of power on how the game is developed through giving feedback, and the developer is transparent about their processes (Elahee, 2019; Long, 2014; Thominet, 2018; Thominet, 2021). It is a user-centered design practice where the developer iterates on the project based on feedback from the players, and then communicates the changes and the roadmap back to the players (Thominet, 2018). So, instead of having fully closed process such as in traditional game development where only the development team can affect the development and design process, in open

development anyone who plays the game or gives feedback can possibly affect the direction of the development. Open development projects can be done fully open from the get-go, or the development team can open the project for audiences in later phases as well. In projects that are open straight from the beginning, the future players can start affecting the game design from the initial state, for example by giving suggestions on what type of game they would want the game to be, or what features it should have.

Thominet (2018) described that all the developers in his research about open development had these three methods in place: distributing access, developing transparency, and collecting feedback. These are the core features that open development needs to have in order to work. This means the developers must have different communication methods in place for achieving these three. For developing transparency, developers in Thominet's study posted summative descriptions of recent updates, day-to-day information about progress and plans for future development tasks. These were posted on blogs, forums and directly on the main menu of the game. They also used public schedules and roadmaps of upcoming milestones, also posted on social media. Sherlock (2014) also wrote that many developer companies use patch notes as a tool to being more transparent and communicate what changes have been made to the current patch of the game. Patch notes are official documentation that contain all the data about design changes and how things should be working mechanically. All this communication and transparency was felt to be important, as developers felt that the players had bought their games on the promise that the game would someday be completed, and they wanted to communicate their progress towards that goal to the players. Game developer Spock (2014) in his interview told a similar story; developers wanting to be transparent to the players, which meant describing their design goals, explaining their reasons for making such decisions and setting clear limits on the scope on the design of the game. These were posted and updated on public forums and blogs, which in turn kindled discussion with the players and enabled them to get feedback. He said one of the key reasons for him to go open in the development was because he got a group of people to contribute their good ideas to his projects.

Getting feedback and distributing access is a lot of work. This process requires developers to set up communication channels, maintain them, communicate with the players, soliciting and facilitating gathered feedback, analysing the feedback and lastly, implementing the feedback into the game (Thominet, 2018). There are multiple places to set up communication channels, such as forums, community messaging platforms (e.g.,

Discord, IRC), social media sites (e.g., Reddit, Facebook, Twitter), developers' or publishers' websites, Steam's own community hubs and directly through in-game feedback widgets. Communicating through these channels might happen in text forms, but other commonly used tactics are polls and surveys (Thominet, 2018; Thominet, 2021). Feedback can also be gathered through analytics that are recorded through in-game feedback widgets, or simply analysing player behaviour through players actions in the game. Later one of these is also called "passive feedback," since it does not require any active feedback from the player (Jacobs & Sihvonen, 2011). However, analytics do not always tell the whole story, since simply logging all actions do not tell the reason for those actions.

All of this means that developers have to become communicators, and that the process of open development is a mutual learning process for both users and developers. Players explain their experiences, and the developer answers with explaining how the game code and environment works and proposes a solution. Then the players test the offered solution and give additional feedback. (Brown, 2015; Lin et al., 2018; Thominet, 2021). This way updates are tried and tested by the players, and with the feedback from those players a new iteration begins and ends up published as a new patch. All of these communication tasks concerning the feedback cycle are sometimes split between everyone on the developer team to ensure that every bit of feedback is seen and acknowledged (Brown, 2015), or in some cases a new member or team is hired to act as a community manager or a community team to handle all the feedback and analyse it.

According to Thominet's study (2018), player feedback ranges from bug reports to larger suggestions to the game as whole. Most of the feedback is on the quality assurance level (bug and glitch reports), some on the recommendations for content or mechanics, and some on the larger whole, such as design suggestions. Jacobs and Sihvonen (2011) sum up that overall feedback is gathered in order to work together with the players on issues such as game balance and stability. Then again, sometimes developer control over the game is needed. This means that the developer has a clear understanding of where they want to take the game and stick to their original plans disregarding community feedback. This kind of control is necessary in some cases, since Jacobs and Sihvonen (2011) remind that sometimes the players want things to be added to improve their current situation, instead of seeing what makes the game engaging, challenging and sustainable. Players tend to want more of what is scarce in the game, which would make the game too easy or

unbalanced, when the developers have added that scarcity on purpose to create better gameplay. Thominet (2021) also defends developer control over the game by writing that game design is at its core an artistic endeavour, and thus the community feedback will affect only the portions of the game that the developers want it to. Game developer Ismail (2016) summed up developer control over the game in his interview as

We wanted to create this feedback loop where we would put out a version and see how people were responding to it and then update the game based on that. It was our vision. We very strictly said you do not get ownership of the game. You do not get to tell us what we do. We're going to choose what we do. You get to watch. You get to give us feedback. We'll take your feedback into account if we disagree.

Game developers Luck and Day (2015) also elaborated on the topic in their interview by saying “Interestingly, you don’t have to be 100% open or steaming every second of the day to the interwebs. It’s possible to be as open as you’re comfortable with, for as long as you’re comfortable with, on your current project.” As Thominet (2018) writes in his research paper, several developers stated that they also hoped to have constructive conversations with players about the actual design of the game. This way the control over the larger whole stays in the hands of the developers, but they are open and willing to take in suggestions from the players that would make the game better suited for its target audience.

Another way of facilitating interest and supporting the player base is allowing them to create their own ideas about the game. One of the most basic ones is allowing players to do modifications to the game (modding), include music or art the fans have created, host contests of character designs and implement the winning characters, allow players to help in creating story elements or help with translations, and so on. (Johnson et al., 2015; Thominet, 2018).

### **Early Access**

Early Access is in a way in-between traditional and open game development. It has the beginning of a traditional development cycle, but at some point, it shifts towards the open development cycle. This usually happens after the alpha stage has been completed in the form of traditional game development (Thominet, 2018). This way the beta stage is already fully open, and the developer starts to gather and implement the player feedback. This timing differs between developers, but at least one rule of thumb is that the game cannot move to Early Access before it has a playable build, and it should move to Early

Access in a stage where the players can still fundamentally affect game mechanics and content, at least on some level (Thominet, 2018; Valve, 2022a). As Early Access has become increasingly popular, Lin et al. (2018) described that Early Access model uses open development practices to elicit early feedback and in turn attract new players early on, while the company does not have to worry about length of the Early Access state and strict publishing dates.

The same three-point guideline applies to Early Access as what is the core in open development; giving players access to the game in early stages of development, being transparent in the development (giving out information on game state, design and goals, and the development process in general) and gathering feedback (creating spaces for getting feedback, giving opportunities to communities to affect the game development, gathering active and passive feedback). In Early Access projects the developer control over the game might be stronger than in fully open development projects, since the developer already knows the broad lines and designs they want to follow, as they already have a playable build of the game. However, it is important to leave space for change and adjustment in Early Access projects as well, even though they are only partially openly developed. After all, the reason for going Early Access is to develop the last parts of the game in open development practices (Elahee, 2019). Sansone (2014) states that with the model of Early Access, using iterative feedback cycle enables developers to see what the players actually encounter instead of what the developer has designed them to encounter and work out the kinks in places where the experiences do not line up. This way the game design gets refined as it is developed towards the finished product.

### **3.2.4 Pros and cons of Early Access**

Early Access, like other development methods, has its pros and cons. Many of the positive sides also depend on the developer studio, and how they approach the Early Access stage. Lin et al. (2018) described in their study, that as high as 81% of Early Access games in Steam have an equal or higher activity on their discussion forums than other games, and that developers of Early Access games receive crucial feedback from their target community during Early Access phase. They also observed that players who buy Early Access games are more tolerant towards bugs and glitches, since they know that the game is still in an unfinished stage. The players felt like they were part of the game production and felt satisfied with the fact that they were the first ones to play the game. This communication between the development team and the players also affected the

frequency of the updates the developers gave to the players positively, which is an effective way to keep the players engaged. Implementing the feedback from the players and communicating the changes made is also a way to keep the players engaged, as they feel like they have a voice and that they are part of the design of the game.

Sherlock (2014) wrote in his study about *Path of Exile* (Grinding Gear Games, 2013) that the design cycles were based on feedback given by the players in the open beta of the game (that served the same purpose as Early Access nowadays). The feedback from early players was the guide in the developers' design strategy, and that made the game successful and popular, as it directly served the player base. This was to be expected, since the players were the ones guiding the user experience design. Gandolfi (2018) also summarizes that the ideal game design using Early Access means informing and shaping experiences that players perceive as relevant. This was clearly done well in the case of *Path of Exile*. A similar example was found in Gandolfi's study (2018) on *Darkest Dungeon* (Red Hook Studios, 2015). *Darkest Dungeon* went through Steam Early Access, and the Early Access stage was crucial for the game. He describes that the developer team was self-funded and in need of testers, and Early Access was a solution for both problems. The team received hundreds of pages of constructive critique, feedback, and observations to help the team develop the game further. The feedback focused on balance, tuning and bugs. There was so much feedback that the team decided to hire a community manager to go through all the feedback that they got. The team tried to address as many issues as they could, but in the end, focused more on the issues that were brought up multiple times or to some features that had heated discussion around them. After the game was released, the developer team praised Early Access, but also stated that "Regular communication from a developer to their community is crucial in early access, and it does place additional responsibilities/demands on the team. However, we felt it was worthwhile."

In Lin et al. study (2018) they also found that the developers did not have the same pressure to release their game when going through Early Access compared to traditional game development, as they had more time to polish their game during the Early Access stage due to the early funding that the stage provides. This way the developers can choose the update schedule that best serves their development process, rather than rush features or release the game too early resulting in bad reviews. Hill-Whittall also mentions in his book (2015) that Early Access can be used as a funding option for games. However, this method for funding is not right for every type of game, and thus Early Access differs from

its sister phenomenon crowdfunding. When crowdfunding, the team does not have to provide a working build of the game, but just a concept of what the game is going to be like. In Early Access, the team has to give out a build that is playable, so for example one-time story telling games might not be right for Early Access. Hill-Whittall writes that “Sandbox games are a good fit for Alpha [Early Access] funding as players anticipate a fairly open specification in the early stages, and expect that their feedback will help shape the game during the course of development” (p. 223). He continues that Early Access is a great way to market the game, since it can draw in a large crowd that will eventually spread the word about the game.

This brings us to the main point that can make Early Access an unwise decision. As Lin et al. (2018, p. 793) puts it, “It is risky to use the early access model as the main funding source.” They elaborate by stating that if the game is released to Early Access too early, it may drive the players away, as it is not playable. If a game is abandoned by the developer team, the players might feel like they have been defrauded, as they paid for the game that is never going to get released. This can hurt the reputation of the developer team as a whole. This is also the reason Valve added guidelines in their Steam’s Early Access guide (Valve, 2022a) stating that “Early Access is a place for games that are in a playable alpha or beta state, are worth the current value of the playable build, and that you plan to continue to develop for release.” Valve also urges players to buy Early Access games that interest them in their current state (Valve, 2022b). Hill-Whittall (2015) also reminds that the pricing of Early Access games needs to be correctly evaluated, as the game is not finished and thus cannot be priced as a complete game.

Another issue is communication. If the developers do not communicate their end goals and the road to finish clearly enough, or they do not consider the feedback they get, the players might feel like they have been exploited or lose interest because they are not heard. Community groups may also have conflicts that might need solving and clear communication from the developer team (Lin et al., 2018; Hill-Whittall, 2015). Sometimes the feedback from the community is also just too much and too varied to be considered. Jacobs and Sihvonen described in their paper (2011) that sometimes Early Access projects might result in arduous and time-consuming iterative cycles, as it is too difficult to include all the feedback and player perspectives despite the developer’s best interests. This might also lead to tedious, long crunch times, where the development



studio is struggling to deliver a product in a short time frame, which in turn might lead in burnout of the employees (Weststar & Dubois, 2022).

Gandolfi (2018) summarizes some pros and cons of Early Access by stating that game developers can benefit from increased participation of target players in the early stages of development, and this approach span transparency, engagement, and commitment. However, poor communication, incoherence between testing and final product or disregarding player feedback might weaken the overall outcome of the game and result in bad reviews and no community engagement. In Hill-Whittall's book (2015) Markus Persson, the creator of *Minecraft*, summarizes his experience with Early Access as such:

Let their feedback influence your list of planned features and take on board popular ideas. But at the same time, keep to the principles of your core game idea. Immerse yourself into the community, even if there are bad spots. The good times are good! Support the YouTubers and streamers as they are your lifeblood. Modding is awesome. - - Be clear. Absolutely, unambiguously, massively clear on what it is your game is offering today, what you're planning to add in the near future and what your vision is for your game. (p. 223)

## **4 RESEARCH METHODOLOGY**

In this chapter I will explain the methodology and data of my research. I have used two different methods for gathering my data, a structured qualitative survey and semi-structured theme interviews. The structure of my interviews is based on the qualitative survey. I will also explain how I chose my target audience and why I decided to switch to interviews after doing a pre-study using the survey. In the end I will explain how I analysed my results using thematic analysis.

### **4.1 Collection of data**

I conducted my own research on Early Access and user feedback using a survey and doing interviews with industry professionals. First, I chose my method of research to be a qualitative survey, and I conducted a sort of “pre-study”, that resulted in three answers. Then I shifted to semi-structured theme interviews and got five answers with that method, which resulted in eight answers in total. My focus was on Finnish game studios that have a game that is currently in or has been through an Early access stage. I focused on games that are mainly for PC or consoles. All my interviewees will be pseudonymous to prevent any backlash, as I also asked questions about the negative sides of Early Access and problems in their development processes.

The idea with qualitative surveys is that it's a questionnaire where instead of a scale or multiple-choice answers almost every, if not every, question is an open field question where the respondent can describe their situation more broadly in their own words. This can also be helpful when researching issues that need anonymity, since as opposed to interviews, the answers from qualitative surveys can be gathered anonymously. When answering a survey, the participant can also take their time thinking about the questions and go back and forth between questions as needed. The participants can answer the questionnaire at any time that they seem fit. The questionnaire should be approximately five to six questions, since the time answering each question is much longer than in quantitative questionnaire, as the participants are encouraged to write about their views in a broader manner (Braun et al., 2021).

As qualitative surveys enable the researcher to reach a wider audience than traditional interviews, I wanted to use surveys to reach multiple gaming companies that were using

Early Access as a tool, instead of just the few in my area. This way the data would also be qualitative, and to get a deeper look at why the companies chose to go with Early Access and how they incorporated it into their design.

However, some issues surfaced as I continued to delve deeper into my research. The first issue with surveys is that they do not necessarily give a chance to ask follow-up questions, and another questionnaire should be made if follow-up questions were needed. If some new information comes up from the first survey, the lack of follow-up information can only be acknowledged in the paper, because it would require a new survey to study that information in depth. There is also no chance to ask questions regarding answers that are not clear. The second issue was the main issue; game developers tend to be busy doing their work, and it was very unlikely that I would get any answers to a survey beyond my own contacts. Thus, I used the survey as a pre-study, that got me three very enlightening and detailed answers, and took the questions to be the base of my interview questions.

The idea with interviews is to gather qualitative data straight from the industry professionals in spoken form, that is then transcribed. Interviews are great for studying issues that need in-depth answers and data (Cote & Raz, 2015). It is also a good approach when the researcher needs to ask additional questions and deepen the conversation on a specific topic if needed (Morgan, 1997). Interviews also provide deeper knowledge, as interviewees are free to answer to the questions with explanatory and broad answers. This is even better than in surveys, as the researcher can then ask additional questions or shift the conversation to get more detailed answers, as described above. In interviews the research questions are the backbone of the meeting, but they can be modified and shifted during the interview if needed, which results the questions being semi-structured.

Changing from surveys to interviews allowed me to ask additional questions and take control of the conversation and information, which the survey did not allow me to do. As the research questions are qualitative in nature, it was in my favour that interviews brought me more in-depth answers to my questions. This is because interviews excel at giving out deeper insights in qualitative form and achieving descriptions of detailed levels of individual experiences (Cote & Raz, 2015). This way I also did not need to rely on the fact that some people tend to write quite short answers in writing in comparison to speaking.

The survey answers were collected using Google Forms, and it took approximately 40 minutes for the participants to answer. All the answers were well thought, and the participants put time and effort into describing their points of view. The interviews were conducted through Discord and Zoom, and the interviewees were told to reserve about 30 minutes for the interview. However, there was no maximum time limit, and the interviewees were free to expand their answers the way they saw fit. On average it took 25-40 minutes to get through the whole interview. The interviews were recorded and later transcribed using the transcribe tool in Microsoft Word Online and fixing the incorrect and missing parts by hand. However, one interview was not recorded properly due to a technical error, so the contents of that interview are based on notes that were written directly after the interview summarizing the answers. The semi-structured interview body consisted of five different categories: background questions, purpose of using Early Access, strategies of using Early Access, implementing feedback, and adjusting to Early Access. All the participants were currently working in the game industry, and they all had been working, or were currently working, with an Early Access title.

The survey questionnaire that also worked as the body of the interviews, can be found in Appendix 1. Here is the explanation of why I chose to ask those exact questions. The first question is for identifying the interviewee in my thesis, as the answers are otherwise pseudonymous. The second question is for getting information about if Early Access was used solely for funding or marketing, or also for feedback and implementing that feedback to the game development. Strategy questions are for knowing how much the players got to know about the game when they were giving feedback. These questions also provide a feel about the developers' mindset about transparency of the game and the state it is in, as well as to know how the developers gathered feedback during their Early Access phase. The last strategy question is to know if there were other ways the developer tried to spark interest in their game, and if they gathered feedback in other creative ways than just in written format. The fourth and fifth questions are about how the developer implemented the feedback gathered from the players, and whether the phase where they moved to Early Access affected the collection and implementation of said feedback. The sixth question is to see if the feedback changed anything in the development or design process, or whether the developer control over the game denied feedback to change the game, and on what level. The seventh question is to know if all the feedback was considered, or if something was left out because of workload issues or communication issues. The last

content question is about the overall feeling the interviewee has from the Early Access project.

## **4.2 Research data and participants**

As mentioned, the research data was collected through qualitative surveys and semi-structured interviews. The survey data was gathered as a sort of ‘pre-study’ to see what kind of answers I would get and whether the survey would be the main method in this research. As the results were valid and the data is usable in this research, those answers have also been included, even though the data gathering method was changed after the ‘pre-study’. The main part of this study was conducted via interviews. With transcriptions it was then possible to analyse the answers with thematic analysis, which I will explain in the next section. All the interviews were conducted in Finnish, so all the direct quotes have been translated by the researcher.

All the participants were Finnish game developers who were working with a PC or console game that was currently in Early Access or had been in Early Access and already launched. Some of the answers were from developers who had just entered the Early Access phase, some had been in the phase for under a year but still had some experience with the practical side of it, some had been a longer time in Early Access, and some had already launched the game and they were able to talk about the phase as a whole. Research participants were found by handpicking Finnish game studios with a publicly available Early Access title. This was done by first searching Finnish game studios, and then going through their websites or Steam pages to see whether they had an Early Access title currently in development or amongst their previous games. Interviewees were then contacted through e-mail, and if that did not lead to a response, they were then contacted through Discord if that was possible. This excludes some participants that I already knew beforehand, and who’s contact info I already had. They were contacted in person.

As the participants and their answers to this research are covered, I chose to use pseudonyms to distinguish them in text. I thought about using their background information as an identification, but in the end, it was clearer for me to keep track of the answers using the names. This is because one of the background questions was the interviewee’s position in the company, and many of them had multiple positions, such as CEO, lead programmer and community manager. Some of the teams were quite small,

starting from only three members, which of course might lead to each member having multiple work tasks.

In the end, eight participants in total took part in this research. Two of the participants were women, and six were men. The participants are coded as follows:

Aimo: First person to answer to the survey

Bettiina: Second answer to the survey

Camilla: Third answer to the survey

Daniel: First interviewee

Esko: Second interviewee

Frans: Third interviewee

Gabriel: Fourth interviewee

Henri: Fifth interviewee

### **4.3 Thematic analysis**

After collecting the research material through surveys and interviews and transcribing it, the material was thematically analyzed. As my dataset was quite small and qualitative in nature, thematic analysis was a great way to raise themes and meanings from it. Qualitative data means that the focus of the data is on the text, images, sounds and meanings, and it is non-numeric and less structured compared to quantitative data. This is because the data collection process is less structured and more flexible and focuses more on content questions than just 'yes-no' questions or datasets with numerical values (Guest et al., 2012, pp. 3-4).

Thematic analysis is a method to analyze qualitative data. It provides accessible and systematic ways of working to generate codes and themes from qualitative data sets. Codes are small bits of interesting information gathered from the material, that are then used as building blocks for larger themes and patterns of meaning. Themes then provide a framework for the researcher to organize and report their findings (Clarke & Braun, 2017). Thematic analysis consists of six phases according to Braun and Clarke (2022, pp.

31-34): “(1) dataset familiarisation; (2) data coding; (3) initial theme generation; (4) theme development and review; (5) theme refining, defining and naming; and (6) writing up.” During this process the researcher creates meaning by comparing code frequencies, identifying code co-occurrence and/or graphically displaying relationships between codes within the dataset, and in the end creating themes (Guest et al., 2012, p. 10). So, thematic analysis is a method for developing, analyzing, and interpreting patterns across the gathered dataset in order to create meaning.

Using thematic analysis, the researcher is a subjective part of the coding and creating themes. The codes and themes do not just emerge from the material, but the researcher needs to construct them, thus being a part of the process. The subjectivity cannot be removed or managed, but instead it needs to be understood that the interpretations taken from the material are the researchers own (Braun & Clarke, 2022, pp. 35-40). Therefore, my own standpoint and background as a researcher might affect the results I have gotten from the dataset, and someone else going through the same dataset might bring up different results.

All the survey answers and transcribed interviews were coded and then those multiple codes were drawn under separate themes. During this thematic analysis, I raised these following themes from the materials:

1. Reasons to use and not to use Early Access
2. Strategies to give out information to players during Early Access
3. Strategies to gather feedback from the players during Early Access
4. Incorporating feedback into the game development and design
5. Limitations to incorporating player feedback

My themes are mostly centered around player feedback since that was the premise and foundation for my research. The themes raised from the dataset also helped me in forming my research sub-questions, which are:

1. Why do game developers use Early Access?

2. What methods do the developers use to give out information to the players, and what methods do they use to gather feedback in order to incorporate it to the game's design and development?
3. On what level is the player feedback taken into consideration and included in the design and development process, and what are the limitations?

In chapter 5 I will go through these five themes. The chapter is divided by the three sub-questions, each one discussing one or two of my themes.



## **5 RESULTS AND ANALYSIS**

In this chapter I will analyse the answers that I got from my research. This chapter is divided into three sections, based on the research sub-questions I have. The sections are “Reasons to use and not to use Early Access”, “Methods of giving out information and gathering feedback” and “Incorporating player feedback”. At the end of each section, I will answer the sub-questions, which in turn will help me answer my main question, “How does Early Access and player feedback affect game development and design?”

### **5.1 Reasons to use and not to use Early Access**

There are reasons why the developer might want to use Early Access, and reasons why Early Access might not be the best solution for a game. Both types of reasoning came up from the surveys and during the and interviews, and I will next discuss why some developers used Early Access in their projects, and why they though Early Access might not always be the solution to use.

#### **5.1.1 Reasons to use Early Access**

As indicated in my background theory section, the biggest reason to use Early Access is feedback from the players, followed by funding and marketing reasons. These are also the reasons my research participants pointed out. All of them stated that they wanted to gather player feedback on certain issues, such as overall ideas and concepts, technical standpoints, play experience, balance, features, game mechanics and for getting an idea when the game is ready to be published. They also pointed out that Early Access can be used as a form of quality assurance testing when trying to fix bugs, glitches, and crashes. Using Early Access also provided an instant feedback loop, where the developer got the feedback instantly after pushing a patch and could then push out instant fixes to issues raised by the players in a fast cycle.

At its best, Early Access became a sort of “symbiotic relationship” with the development studio and the players of their game. Both the developer and the players worked towards the best possible outcome. The developer got feedback and ideas while the player got to play the game early and had a chance to give that feedback to the developers. As my interviewee Daniel said about Early Access as an inclusive journey:

If Early Access is made to be an inclusive journey both for the developers and the players, then it works well. But if your communications do not work, you push out patches every now and then and nobody really knows what is happening, that's the other end of the spectrum.

As observed by Lin et al. (2018), in my study the participant developers were also comfortable with publishing a more broken build to Early Access, since players seem to be more forgiving about the issues in the game during Early Access. I think that this comes down to the “symbiotic relationship” as well, since the players acknowledge that the game is still in progress to be able to absorb the feedback from the players, and thus the players are more forgiving about bugs, glitches, and crashes. Henri's experience with this was clear; their first launch to Early Access was with a broken and undeveloped build that was full of bugs. Still, the players understood that the game's development was in progress, and the game got mostly positive feedback, such as ‘this game is broken but shows great promise’. Both Gabriel and Esko talked about their experience with players being patient: “One thing about Early Access that is great, is that if there are some random bugs or things like that, players are more forgiving during Early Access.” and “You get players that are more forgiving about the fact that the game is not finished yet, and you get feedback to fix issues. Then when you are ready to launch, those major issues have been fixed.” It was clear that players being more patient during the Early Access phase gave the developer more breathing room when it comes to developing the game, and it gave more time to the developers to handle any bugs before the actual launch of the game. However, both Gabriel and Esko also reminded that they had a playable build and a lot of content to go through at the start of their Early Access journey, which was in their opinion a good base to start, as they had more time to push out updates while the players were kept satisfied and busy playing the already existing content.

Adding to players being more patient about bugs, one of the interviewees, Frans, pointed out that developers might feel more free to experiment with different contents to the game, since the game is still undergoing development. This is something that came up in the background research as well (mostly by Jacobs & Sihvonen (2011) and Lin et al. (2018)), but it was never discussed in depth by any of the researchers. Frans said that “In Early Access the ‘rules’ are a bit looser, so that we can make bigger changes to the game if needed.” This was the same reason why Daniel and his team decided to go through Early Access with their project. As they had not developed a game like their Early Access game

before, they felt like they needed to test the reception of the game during a more relaxed publishing environment. Daniel elaborated on their situation:

Biggest reason [to do Early Access] was probably that this is an open world game, and it's a whole new thing for us technologically speaking, - - and we had to build our own engine and technology for this. We wanted to know how it works technically speaking, and how it works with all the setups that the players use. But then again this is also a new thing for us considering the game design as well, so even if you can look up other open world games to get some structure and ideas, we wanted to publish this kind of in-progress cross-section to see how it was received and then we reacted to that.

Based on my dataset, this kind of more relaxed environment considering the state of the game was clearly one of the main reasons to use Early Access. Testing the game out with actual players before the launch was seen as a good trial run to see whether the game would be successful, and what were the changes needed to make to shift the game to be more suitable for players.

Research participants also pointed out that when the players get the feeling of their ideas mattering, they also want to participate in giving feedback, and maybe even talk about the game to other players and encourage them to buy it as well. This also ties into what Lin et al. (2018) found in their study. The players might even want to support the developer just because the developer has been inclusive. Creating a game suitable for the players (as those players have had a chance to affect the development) helps with marketing and the launch. Esko talked about their experience with players: “[Players] might buy the game... or spread the word about the game, just to support us. - - And when we publish the game, they might help in the marketing side as well.” This brings us to using Early Access as funding and marketing.

Both funding and marketing are things mentioned in my background theory section. Lin et al. (2018), Gandolfi (2018) and Hill-Whittall (2015) all write about how Early Access can be a successful funding and marketing method, which gives the developers a possibility to work on their game longer and polish it to the state that they feel comfortable publishing. They also write about how Early Access can help gather an audience before launch, that then helps the developer with marketing during the publishing. These are the same things that the participants in my study stated. Esko talked about how funding during Early Access can help the team if they need more time to develop the game:

We kind of had to think about the financial aspect as well. We realized that it takes a lot of time to polish this game to be the gem that we want it to be, so

we had to get more funding in order to get years more time to develop the game.

Esko also talked about word-of-mouth marketing: “Players might buy the game just to support the developer or spread the word and so on.” Another thing that surfaced considering the funding, that my interviewee Frans brought up, was that if the game in Early Access is a live game, the development team can also test out monetization models. Frans said:

Yeah of course there’s the incentive to get funding, and when we are in Early Access with this kind of a live game, we are interested in the commercial performance as well, like what players are buying and what are they using money on, so that we can tweak that if necessary.

Frans also agreed that Early Access can be used as a marketing method, as the game gets two launches, where the first launch is paving the way for the actual launch. With an existing player base from the first launch, the second launch can be more successful. Frans summarized:

In my next project I would use Early Access as a clear marketing tool. - - So, you would have like two launches in a short time span, to keep up the tempo. - - Then the players wouldn’t forget the game.

Many of my research participants also felt that simply forming a community, doing community management, talking to the players, and communicating with them is a type of marketing itself. Baym (2015) wrote an article about musicians hosting communities on the side of their musical work, to be able to market themselves and make a living. As connecting with the community is rarely directly compensated, this is considered as a form of relational labour. Relational labour comes down to “regular, ongoing communication with audiences over time to build social relationships that foster paid work” (p. 16). This can also be applied in forming communities within the Early Access phase of game development, as forming a community, talking to the players and engaging them rarely brings in compensation by itself. However, as Aimo wrote in his response, “It can be a powerful marketing method to be active with your community, talk to them in Discord, forums, social media.” and Esko talked about their journey stating that

We noticed early on that when you involve the players in the development process, it helps in so many ways, especially because we are an indie developer. It helps the chances to make a successful game. And at the same

time, you can get ideas and feedback, and you form a community. And when you publish the game, the community helps with the marketing as well.

Even though relational labour might not directly bring in revenue, hosting the environment of a symbiotic relationship to get feedback might at the same time help with the marketing of the game.

Considering these statements, Early Access was seen as a good opportunity to get funding to finish the development of the game without sacrificing aspects of it due to money or time restrictions, and get marketing going before the actual launch. Participant developers felt that it might be easier to get the word out and market the game forward with an existing player base, which in turn would help with the actual launch.

Some of the participants also stated that in their opinion, Early Access is the ‘right way to develop games’. This comes down to the symbiotic relationship, as the developers felt like they were creating the game for the players, and getting the opinion and feedback from those players makes the game more suitable for them. Esko opened his opinion about this by saying:

I think that Early Access is super great, you get a better chance to create a successful game, so I fully believe in Early Access and the process. With the feedback and all, I think that it’s the right way to create games.

Henri shared the same feelings and stated that in his opinion Early Access is also the right way to create games. Frans talked about creating the game for the players by stating that “We take a lot of the player feedback into consideration, - - since we are making the game for the players, after all.”

Overall, all my participants had positive things to say about their Early Access experiences, and all of them mentioned player feedback as one of the reasons for going into Early Access. Most of them also used Early Access as a funding and marketing method, and felt that Early Access gave them more freedom, time, and certainty in developing the game.

### **5.1.2 Reasons not to use Early Access**

As mentioned previously, there are also reasons not to use Early Access. My research participants pointed out some reasons that should be at least taken into consideration when going to Early Access, or maybe even be considered as reasons not to use Early Access in a particular project.

The main reason many participants brought up was overwhelming feedback, leading to tiredness or burnout. Both Jacobs & Sihvonen (2011) and Westar & Dubois (2022) also discussed time-consuming iterative cycles and employees burning out. When doing Early Access, the developer company inevitably has to take in feedback from the players, as has been established in this thesis. However, the amount of feedback can be a lot to handle, in terms of just the sheer amount of it, but also when considering which feedback can be implemented, and the amount of time and work it takes to implement said changes. Processing the feedback is a task on its own and brings extra workload to the development project. On top of developing the game, communicating to the players, going through all the feedback, the developer team needs to push out regular updates to an incomplete game. When answering the research survey, Camilla wrote that “It was hard to do monthly, regular updates”, as players were expecting something new on top of the fixes to the old build. Most of the developers stated that due to workload constraints they could not implement all the changes the community suggested, since they wanted to keep everyone in good shape and did not want to risk anyone getting burned out. Daniel summarized this situation well:

When you consider the size of our team - - I’m doing three things at the same time, and everyone else does too, so we didn’t have time to implement [a suggested feature]. Or maybe we could have, but that might have resulted in tiredness, burnout, and stuff like that, so that wasn’t really an option.

Considering this, it is important to estimate the resources the team has before going to Early Access and prepare for a larger workload the Early Access phase brings. Even though Early Access can be a success with a smaller team (some participant developer teams were as small as only three people), it has to be clear where to draw the line considering the workload, the amount of communicating and how to handle the feedback. In a later section, ‘Limitations to incorporating player feedback’, I will explain this in more detail.

Another thing considering the communication and feedback that was brought up by a few participants, was mixing private life with work life, and being “responsible” to the players. I did not find any mention of this in my previous research, and I felt like this was an interesting point, as more than one of my study participants brought up this topic. As the community is quite closely involved in the development of the game during the Early Access phase, it might bring pressure to deliver working patches in time, and then to be available to discuss the contents of the patch. This might bring on the feeling of having

to be always online and reachable, and not just working hours, since the feedback comes around the clock, not only during nine to five. This is a good example of online technologies allowing work to invade free time of the developers (Sotamaa, 2021), as it might be difficult to draw the line between work time and ‘just quickly checking the messages on the phone.’ Daniel said that he ‘made an oopsie’ by using his personal Discord account before realizing the amount of work it brings, which is of course one step further in the problem of distinguishing personal time and work time:

I made the mistake of using my own personal account [in Discord], - - so now when I’m not working, I just have to think that ‘I’ll answer tomorrow, just ignore the messages even though there is the huge red bubble informing about messages I haven’t read, just don’t read them’ [laughter]. - - So, there is the danger of personal life and work life mixing.

Esko also said that even though he thinks Early Access is great, it would sometimes be nice to develop a game at his own pace in a bubble, without the pressure from the players. It can be tiring to have the players right there all the time during the development process, asking for changes and updates. This is something that the developer team has to be ready to adjust to if choosing to use the Early Access model. It is best to establish clear lines of private and work life and stick to those, even though the community would be active around the clock. Some players might think that developers should be able to answer any and all questions at all times and fix bugs instantly, so it is important for the developers to clock out after the workday and keep clear lines between work and personal life. This is especially important with an Early Access project, since the bleeding of work to free time is an established issue overall in the game industry across the world (Sotamaa, 2021), and Early Access loosens those boundaries even more. However, in Finland the separation of work and free time is almost a standard and much more accepted compared to the rest of the world (Sotamaa, 2021), so limiting work to working hours is advised, since that reduces the risk of tiredness and burnout.

One clear thing that might make Early Access a bad option is simply the type of game that is in development. Many of the research participants mentioned that the reason they decided to use Early Access was that their game was suitable for it, as the games had a lot of replayability, and they were not based on a single storyline. Hill-Whittall (2015) also mentioned this in his book. He stated that in the case of story games, it might be more appropriate to use other funding methods such as crowdfunding instead of Early Access. He wrote that sandbox games are a good example of games suitable for Early Access, since they have replayability and they can be shaped based on feedback. This applies to

the research participants' games as well, as many were sandbox type games, roguelike games, battle arena games or adventure/action games where the player could experiment, explore, and replay the game. Henri also mentioned in his interview that they had previously tried Early Access with a story game, and it did not have the reception that they had hoped for, as the game genre was not suitable for Early Access. Hence, when thinking about going to Early Access, it seems to be important to evaluate whether the game genre is suitable for players to be played again and again, and if the game can be shaped by the feedback the community gives. Early Access might not be the right option for a game where the players get an unfinished experience, play it once to see all the content, and do not come back since they already know the main contents and story of the game.

The last thing that came up from the research participants was that Early Access can also be the thing ruining the actual launch of the game. As mentioned before in the 'Pros and cons of Early Access' section, this can be a sum of many things, such as the team not communicating during Early Access or not engaging the community, or the team not being transparent, or the game not being suitable for Early Access. In the same way as a successful Early Access phase might help with the actual launch, an unsuccessful Early Access phase may affect the actual launch poorly. Gabriel summarized this by saying:

What we noticed with our game is that the reception of the game when going to Early Access determines the reception and success of the full launch. - - So, if the Early Access launch goes poorly, the 1.0. launch will go really poorly. - - Then you have to think about whether you want to try to launch with the quieter Early Access launch or with the actual 1.0. launch.

In my opinion, this is the summarization of things that developer teams should consider before going into Early Access. The team needs to think whether they are ready to take on the increased workload and the feedback from the community, in exchange for getting ideas and testing their game out, and eventually making the game more suitable for the players. If the developer just wants to fulfill their own ideas, does not communicate, or take feedback into consideration, Early Access is not the thing to do, as it may as well ruin the actual launch of the game. In that case it would be better to launch the game as a whole, and possibly reach a larger audience, as not every player wants to support and play Early Access games.



## **5.2 Methods of giving out information and gathering feedback**

As involving the players in the process of Early Access was thought to be an important part of the functionality of Early Access, it is important to understand how the developers communicate to the players. Here I will discuss the strategies game developers used to hand out information to their players, and how they felt about being transparent. After handing out the information the developers also need to have strategies in place to gather the feedback from the players, so I will also discuss which tools were used for that purpose, and how they were used. This is also important for understanding how different types of players and feedback were included, as not all players are willing to go to the forums or social media to give feedback, and they simply want to play the game. Using different tools to include those silent players is also important.

### **5.2.1 Strategies to give out information to players during Early Access**

Early Access has developed to be a viable tool in creating games while also including player feedback in the design. However, this means that the developer needs to communicate to the players and be transparent about their goals, update schedule, contents of their patches, and so on, since those are the kinds of things that keep the players in the loop and makes it possible for the players to give feedback in the first place. There is an abundance of possibilities for game developers to communicate with the player base, and vice versa. For my research it was important to know how much information was handed out to the players and how it was handled by the developers, to be able to see how the developers communicate with their players. This way I could see whether the players were kept in the development loop and whether they could give feedback to the developers. In this chapter I will discuss different methods the research participants used in giving out information to the players during Early Access.

Most of the research participants felt that being very transparent is important, and as Henri said, they try their best to inform their existing player base on what they are doing, in order to be open and inform players what the team is up to. However, on the contrary to what was established in the background research, a small number of my research participants also pointed out that being fully transparent might also be somewhat problematic. They stated that the players might not be interested in the things that are going behind the scenes, and rather would have a more polished experience with the game. Thus, in my dataset the results of being transparent were a bit different than in the

previous research, where all the sources emphasized the importance of transparency. My research participants were split on the issue of being transparent, and it can best be summarized by saying that the line of being transparent is something that the developer team must set by themselves considering how much they want to engage the community. Being transparent comes down to something Aimo wrote:

Problem with not giving any information about upcoming updates/features is that we can only react to player feedback after we have made the changes and players [won't be able] to give suggestions during design phase. Also, would be really important and easy to tell players that a bug they are reporting is fixed in the next update if we have a fix for it. - - In Early Access game[state] transparency is important to certain degree. To sell the game it's important that the players know we are working on the problems they are bringing to our attention. It can be difficult to find the good balance as if you give too little information, players can lose hope.

Being transparent might be a lot of work in itself, but also not being transparent enough might bring challenges of its own, like repetitive feedback on something that has already been fixed, or players feeling disconnected due to the silence from the developers' side. Still, the majority of participant developers stated that transparency is important and most of them aimed to be as transparent as possible.

In Table 1, I have gathered all the different tools the developers used in order to give out information to their players, as well as descriptions of those tools, and the number of developers using those tools (eight in total). There were 11 tools mentioned in total, considering that I did not mention all the different social media platforms as their own tools, but rather as a larger category 'Twitter/Facebook/Other social media'. This is because different social media platforms do not bring anything special or different to the table, as they all work in a similar manner. One thing to also consider is that the tools mentioned in Table 1 are gathered straight from the participant surveys and interviews, so if the participants forgot or otherwise did not mention a tool they used, it is not mentioned in this table or counted towards the usage.

Table 1. Tools to give out information.

<b>Tool</b>	<b>Description</b>	<b>Usage</b>
Steam community	Games in Steam have their own community pages, where the developer can post updates, patch notes and other content, and communicate with the players.	8/8
Roadmap	A document that is available for the players, which maps out what changes and developments can be expected in the future considering the development of the game.	7/8
Patch notes/Change log	Patch notes/change log or a list that consists of latest changes to the game, that is available for the players.	6/8
Discord channels	Discord is an instant messaging social platform where the developer can set up a server. The developer can then lead discussions, set up channels for feedback and give out information about the game and the development.	5/8
Forum posts	Forums dedicated for the game, where the developer can communicate to the players.	3/8
Events and talks	Events and talks such as developer conferences and publisher events, where the game can be presented and played.	3/8
Twitter/Facebook/Other social media	Social media where the developer can post updates on their development.	3/8
Live streams	Streaming the game through a streaming service for an audience, to show where the game currently is considering the development.	2/8
Web site	Company's own web site for promoting the game, and housing the roadmap, patch notes, etc.	2/8
Reddit	Reddit is a social platform for discussions. The developer can start a subreddit for their game, and host discussions about issues, updates, and other content.	2/8
Trailers	Trailers are short videos that show and market the game and showcase what is new in that current patch.	1/8

As can be seen in the Table 1, Steam community was the most used tool. All of the research participants used it at least on some level. This is due to Steam being “the most practical and easy way to handle announcements and other Early Access stuff” (Daniel). Many developers felt that because the game was published in Steam, it was a practical choice to also communicate through the same channel and to reach many of the game’s players.

Almost all participants also used some kind of a roadmap and/or patch notes to communicate the state of the development. In a roadmap they could communicate future features, and patch notes were used for communicating latest changes and fixes to the game. However, a few of those participants that used a roadmap, used it very vaguely. Instead of presenting certain future features, they presented something that was possibly happening, or features that they had been thinking but were not sure about implementing. This way the community was also able give feedback on which features they would like to see to be implemented.

Discord channels was a frequently used tool in communicating, as it was seen as very flexible and far-reaching solution in communicating to the players. In Discord the developers had set up different channels for different types of communications, such as channels for news and patches, feedback, discussion, bug reporting, suggestions, fan-art, and so on. The developer team then led conversations in proper channels and gathered feedback from the channels dedicated to that. Two of the developers said that Discord was a far-reaching tool to use, as they had so many players on their servers, that every time they ping on the server about news or patches, it reaches a lot of their players. Other two tools that served a similar purpose to Discord were forums and Reddit. Both these tools enabled the developers post about their progress and communicate with the players in a categorized manner. Even though forums and Reddit were not seen as symbiotic as Discord, the developers who used these tools felt like the developer team could reach a lot of players through them.

Social media and web sites were also seen as a way to communicate to the players, even though not as effectively as a platform that better enables two-way communications, such as Discord. Social media was used mostly for informing players that for example, a new patch has come out, and that the players should try it. However, having multiple platforms to communicate to players was important to the developers, so that they could reach as many players as possible. As Esko elaborated:

Yeah, I have sometimes compared it to a spiderweb, like when you are an indie developer and you want to market your game before the launch, you want to have followers on your social media, and it's important to use many different platforms to accomplish that. You know, many small streams form a river.

As not all players use the same social media platforms and tools to stay up to date with the game they are playing, it is important for the developer to use as many as possible to

reach as many of their players as possible, to inform them about new changes. This was the case in most of the participant developer teams.

Events, talks, and live streams were also used by some participants in giving out information. Two of the participants said that they had showcased their game in different events in order to market it, and at the same time gather feedback by watching people play it. One of the developers told that they show the game in the same event from time to time, which would then inform the players from the changes made to the game. Another way of showcasing the latest changes to the game was live streaming it, with dedicated streams for each patch. This way the players (and also non-players who are watching the stream) could see live footage from the current patch with new features in action. The last tool that was mentioned was trailers, which worked in a similar manner as the livestreams, though in a shorter and more compact mode. The purpose of each trailer was to showcase new features and fixes to the game, and at the same time market the game.

Overall, all the participants did engage the community in some ways, even though some developers more than others. All the developers had multiple different ways of communicating their progress to the players. The developers kept the players in the development loop using different tools and were at least mostly transparent about their progress. Some developers were more open to the feedback than others, but the consensus was that the game developer needs to be transparent about what their next steps are and what they have recently done to improve the game. These steps and changes were then communicated through the different channels mentioned. This lays a good foundation for the players to give back feedback on the game, as they can follow the progress of the game, and have dedicated channels for giving their suggestions.

### **5.2.2 Strategies to gather feedback from the players during Early Access**

Gathering feedback is a core feature of Early Access, as without proper ways of gathering it, the community has little impact on the game design. As discussed by Thominet (2018; 2021) and Johnson et al. (2015), there are multiple places and ways to gather feedback from the players, ranging from forums and social media to passive feedback. Firstly, communicating through these tools can happen in qualitative, text-based way, or in a more quantitative way by gathering data through polls or surveys. Second way to gather feedback is analytics, where the developer team would use passive data gathered from the players' game sessions or their save games, without any actual input from the players.

Qualitative passive data can also be drawn from watching players play the game, in a live setting or through streams and playthroughs. Third way to gather feedback consist of player participation, where the developer supports the community in allowing them to create their own ideas about the game and express them through e.g., modding, music and fan-art.

In my research the participant developers used a variety of different tools to gather feedback from the players. These can be seen in Table 2 in addition to a description of those tools and the number of developers using these tools. All the participants used qualitative text-based feedback, some of them used passive data analytics, some gathered quantitative data through statistics, achievements, and polls, and lastly a few gathered feedback in the form of modding or player participation. There was a total of 15 different tools used. Once again, the data is gathered straight from the participant surveys and interviews, so if the participants forgot or otherwise did not mention a tool they used, it is not mentioned in this table or counted towards the usage.

Table 2. Tools to gather feedback.

<b>Tool</b>	<b>Description</b>	<b>Usage</b>
Steam community	Games in Steam have their own community pages, which work as a forum where the players can give feedback in written form as well as give reviews of the game.	8/8
In-game bug reporting	Bug reporting tool inside the game, where the player can report bugs and other issues instantly when they encounter one.	6/8
Modding tools	Modding tools enable players to mod the game. Modding can help the developer to get ideas for their game, and/or find what is broken or unbalanced.	6/8
Analytics	Passive data gathered from the players while they play the game, or from their saved games, without active feedback input from the players.	5/8
Discord channels	Discord is an instant messaging social platform where the developer can set up a server. Server members can then talk and give written feedback on the dedicated channels that the developer has created.	5/8
Live streams and playthroughs	Watching live streams or playthroughs of the game to get instant feedback while the player is playing, to figure out what are the issues for the players and whether something is not working or is broken.	4/8

Table 2 (continued).

Forum posts	Forums dedicated for the game, where the players can communicate to the developers through proper channels.	3/8
Events and talks	Events and talks such as developer conferences and publisher events, where the game can be presented and played. The developer gets firsthand information on what works and what does not when watching people play their game.	2/8
Twitter/Facebook/Other social media	Reading social media posts to get a feel of what is working and what is not, and what is the overall feeling that the players have of the game.	2/8
Reddit	Reddit is a social platform for discussions. The developer can start a subreddit for their game, where the players can take part in discussions about the game, its contents and issues, and post anything relevant to that game.	2/8
Player participation	Asking for player suggestions to the content of the game, such as character names or short stories, or larger aspects such as player created music.	2/8
Steam developer statistics	Steam developer statistics is a statistics tracker, that offer insights about the players of the game (such as playtime, countries the players are in, etc.)	1/8
Achievements	Platform achievements, such as Steam achievements or PlayStation trophies. These give insights about the tasks that the developer has in the game, e.g. the developer can track how many players have gotten through the game if they have a 'finished the game' achievement.	1/8
E-mail	E-mail address that the developer has set up for the players to use for sending written feedback to.	1/8
Polls	Polls created by the developer to get votes from the players. These can be inside the game, or posted in other platforms, such as Discord, forums, etc.	1/8

Steam community was the most present tool concerning feedback gathering. The same way as Steam community pages were felt to be the most practical and easy way to handle Early Access announcements, it was also the place where the participants could gather written feedback from the players. Players can also leave a review of the game in Steam community pages, which is yet another way to collect feedback on what the players think of the game. Many of the developers explained that they communicated in Steam to be as transparent as possible on the platform where the game is available, which also resulted

in the community being active in the same space. Esko talked about their processes by saying that

We are mainly using Steam, Reddit, and Discord to host conversations, and we are constantly communicating with players. We have own channels in Steam and Discord where players can tell us their wishes, bug reports and have conversations about the game's features. And for example, in Discord, I'm usually there to discuss and have direct contact with the players.

As Esko explained, Discord was another tool that was regularly used side by side with Steam, precisely because it offers a flexible and far-reaching solution to qualitative communications and feedback channels. Discord was also the primary tool that many of the participants felt best served the purpose of gathering qualitative, written feedback, because it offers a clear way to categorize the discussion. Almost all developers stated that in their opinion the best feedback is the written one, since then the players could explain themselves and their feelings on what is working and what is not, and why they felt that way. Both separate forums and Reddit also credited towards the text-based qualitative feedback, as both of those platforms are suitable for hosting categorized, easily manageable conversation and feedback channels. Some developers also read Twitter, Facebook and/or other social media for player reviews, and one of the developers even took some feedback through e-mail. Even though those channels might not be the most organized ways to gather feedback, they count towards text-based feedback that is gathered straight from the players. Esko explained a part of their process of feedback gathering:

We are mostly trying to get the feedback in written form from conversations, like what is not working and what is done well. And even though we have a bug reporting tool, sometimes people write stuff that does not even concern that particular bug, like 'how can this be done so poorly' [laughter]. But the bug reporting tool is good in a sense that we have made it so that it sends a screenshot [of the game] and the player's save game with it, so we can usually see that okay, this is what happened.

Even though basically all developers valued qualitative feedback over other types of feedback, it was also clear from the developer answers that using some type of passive data or in-game feedback buttons was something that worked well side by side with the written messages.

In-game bug reports were used by many developers, as it offers many players who do not want to participate in discussion about the game a chance to inform the developer that something is not working, without having to go to the forums to report it. Usually, the



bug reporting tool would have set options to choose from, in addition to a free section to write about the issue in the players' own words. The tool would usually also add players' save when sent, or a screenshot of the game state. In some cases, the players could even draw on top of the image to better explain what went wrong. This was a great way to include more players in the feedback loop, and not listen to only those players that want to participate in conversations with the developers. This way the developer also can check the data from the save and catch more complex bugs or the reasons behind those bugs, that players might not be able to explain by words. One of the developers also used in-game polls to try and catch the opinions of players on certain issues without forcing them to go to discussion channels. This comes back to the fact that not all players are willing to go through the effort of being active on community activities but would like to present their opinion through the game.

Another way to involve players that are not the loudest on the discussion boards is passive feedback. Using analytics the developers could see the players' progression, what they had been doing, where they stopped playing, and so on. This way it was possible for the developers to adjust the game accordingly; for example, if most of the players get stuck on a certain boss or on a certain area of the game, it could mean that the boss/area is too difficult, or has something that the players do not understand, or has a game breaking bug. Daniel explained that their team had a system in place just for gathering passive data:

Players could opt-in to automatically send their save anonymously to our servers at set intervals. This way we had a tool to visualize their gameplay, to see where they went and what they did, where they died. We could add multiple different saves to collect data about multiple players, and form these kinds of visual maps about like, 'lots of players die here' so maybe the challenge there is too difficult. We also got hardware crash data from this.

Frans also said that in their case they value passive feedback a lot, and that they "track pretty specifically what happens in the games and what are the obstacles the players face. So, where the players stop progressing, - - what parts are too complicated or too difficult." Gathering passive data is a great tool when developing a game to get a feel of the game as a whole, but also from some more minor issues. Passive data can be a good tool in balancing the game.

Both Steam developer statistics and achievements also count towards passive feedback, as those are tools that the developer can use to map out the players' progress. Adding achievements for certain tasks in the game will give the developer information on how

many players have done those tasks, and if a low number of players have gotten them, it might be an indicator for the developer to double check whether the tasks are understandable or if they are too complex or difficult. Steam developer statistics is a great tool getting the overall information about the players of the game, such as the number of players who have finished the game. Gathering and going through different types of passive data can however be a significant process, and a few other participants stated that they simply did not have time to go through the amounts of data gathered using e.g., analytics, so they opted for gathering feedback mostly through discussion channels.

Half of the developers also mentioned that in addition to reading text-based feedback, they also watched players play their game. This happened either live during different events or watching streamers or playthroughs of the game. This way the developer could see firsthand how players react to things in the game, if they do not understand something, or if something is broken. This is also one of the passive ways to include players to the feedback loop that do not want to go to the forums or other discussion channels to separately discuss the issues they faced. Their feedback and experiences with the game are as important as someone's who takes their time writing about their observations, so watching those play sessions is a suitable way to include the more silent players.

Lastly, a some of the developers used player participation. This means that the developer supports the community in allowing them to create their own ideas about the game and express them through e.g., modding, music and other content. Modding was supported in many games, and some of the developers hoped to build a modding support in the near future. They felt that modding is a great tool in engaging the players and that it would give the modders the possibility to experiment on the game and express their own ideas. Some developers also had asked for other types of content from their players, such as names for characters, short stories that would appear in the game, and music that would be implemented to the game. This type of feedback might be categorized more towards player engagement than actual feedback, but it still counts as content that ends up in the game. These types of inquiries also keep the players active and playing the game, as they then see their own suggestions appear in the game.

All the developers participating in this research used a variety of different channels and tools to gather feedback from their players. Even though qualitative text-based feedback was the most used method, most of the developers also used passive feedback in some forms to go side by side with the text-based feedback. I think that it is safe to say that all

the participants gathered and processed a lot of feedback, that in turn affected their Early Access development. I will open this up more in the next chapter.

### **5.3 Incorporating player feedback**

After gathering feedback from the players of their games, the developers need to decide which changes to add to the game. This can be a challenging task due to the amount of feedback, limitations to budgets or time, and the balance of the design. As Jacobs and Sihvonen (2011) discussed, sometimes developer control is needed, since the players tend to want, for example, more of what is scarce in the game, which is affecting the game's balance. However, it is important to take player feedback into consideration. Implementing the feedback from the players (and communicating the changes made) is a way to keep the players engaged, as they feel like they have a voice and that they are part of the design of the game. This brings player satisfaction which in turn is a great selling point, as pointed out by Lin et al. (2018).

Thominet (2018) talked about player feedback ranging from bug reports to recommendations for content and mechanics to larger design suggestions for the game as a whole. In my research I also used this division as a base for trying to see how much of feedback was considered and was actually incorporated in the game during Early Access. Besides this division, there was not much information to be found in the previous research concerning implementing the gathered feedback. I hope my research sheds light on this front.

#### **5.3.1 Incorporating feedback into the game development and design**

My research participants had incorporated player feedback into their game development and design on many different fronts. The largest whole affected by feedback was changing the game's design into its final form. Developers had asked for player input on overall ideas and concepts to shape their game to be more suitable for players and for getting the features to match the community's expectations. A few of the participants stated that their game had the concept and the idea for their game, but they shifted and changed features, the length of the game, added core mechanics, and so on, due to the feedback they received during the Early Access phase. Daniel said that the development process was different from their previous projects, and that they had to improvise and use creative problem solving based on the feedback they got from the players and issues they

raised. Gabriel said that their development process had some uncertainty about some of the game mechanics, so they were able to test those out and polish them before the launch. One of the developer teams, Esko's team, was so open to feedback and ideas from players, that they did not even have a finished plan or script they could follow during development. They developed the game one patch at a time, and after each patch they went through the feedback and then discussed about what they are going to do next. He said:

When we have this feedback from players that we get all the time, it really helps in the decision-making process of where to focus next and what to do next. - - This way the players kind of become a voice in the design process. -  
- I strongly believe that the games made in Early Access are better for the player, because you get feedback [during the process].

Bettiina also stated that if an implemented feature was heavily criticized, it was changed during Early Access, or even removed and replaced with something better. All the participant developer teams were open to adjust their games to the feedback they received from the players and shift the development of the game to better match what the players suggested. This way even though the core idea would stay the same and the developers kept control of the main design plan, they were flexible to add changes to their plan.

Developer teams that had already finished their Early Access stated that the feedback helped them to feel when the game was at a good point to be launched. During the Early Access stage, they were able to scale the game length and content according to the feedback. For example, Gabriel explained that their design process was "quite organic and experimental, so it was a bit difficult to say what - - was the endpoint". This also applies to what Esko was saying about their process, that they adjust the game and its endpoint according to feedback they get. Daniel's team also scaled their game to be much longer than originally planned, to create a better play experience for the players. They noticed during the early stages of their Early Access process that they had players who had already been playing the game for hundreds of hours, so they tweaked the whole game design to be longer considering e.g., the power level of the player, and how long it takes to reach middle and the end game. This way the community feedback gave the developers more confidence in the game elements and the play experience, and they were able to shift the design to suit the players better.

Some of the participant developers had asked for feedback concerning game features. The players could suggest or vote for a feature they would want to see in the future (or to

change a current feature) using the previously mentioned tools, and the teams then decided whether to implement those suggestions or not. In Esko's case, he said that

Player feedback has become very important and a huge voice in guiding us in the right direction. - - We get suggestions that support the game as it is now, and add to that. Things that we too think are great additions to this game. So, in that way the player feedback supports the game development.

This way they took suggestions and developed the game with those suggestions in mind. Other teams, like Daniel's, also added features that were outside of their original design. These included things that were frequently asked by the players, such as a co-op split screen mode. Daniel said that they tried to implement feature changes and suggestions that they felt fit their game and did not affect their development schedule tremendously. Adding requested features is of course a very inclusive way to add community feedback into the game, as features are one of the things the core game is constructed of. Another thing that affects the core game is the content of the game, and that was also one of the things that some developers had asked for feedback on. Gabriel summarized their Early Access experience by saying that "The Early Access stage was mostly about fixing the game, or well, we added a lot of gameplay to it, fixed a lot of bugs, and refined the content." Both features and content affect the players' play experience, and thus asking and implementing feedback on those fronts likely better the experience the players have while playing the game.

One thing mentioned by the participants was balancing the game and its mechanics with the help of the community. This is an important point considering that the developers can become "blind" to their own product, and getting outside feedback on balance issues can be very valuable. Frans described their situation by saying:

We take a lot of feedback about balance and game mechanics since those are the most important things in our game. Player's games are so different from the games of our developers, so we get a lot of good feedback on the balance issues. Well, it's not always positive [laughter], but we get why it was given.

Balance is often considered one of the most important things in a game, as it affects the play experience radically and if the balance is off, it might even drive players away. Early Access was felt to be a good place for balancing the core gameplay, as the developers got a lot of feedback from multiple different testers. Of course, with some games the balancing can be more difficult than in others, but more of that in the "Limitations to incorporating player feedback" section.

Beyond the bigger changes to the games, all the developers also fixed bugs and glitches that were reported to them. Those were mostly fixed with a fast cycle, as those types of changes were faster to implement than bigger changes that impact the design of the game. Bettiina summarized that in their project “Clear bugs and usability issues were easier and quicker to implement.” Early Access was also felt to be the place to squish the biggest bugs before the actual launch, so that the game would be as playable and solid as possible.

I think that it is important to see that every participant developer gathered community feedback and implemented it to their game during Early Access. Using Thominet’s scale of ‘bug reports – content and mechanics – larger suggestions to the game’s design’, all the developers reached at least the middle part. Every developer fixed bugs, glitches and crashes that were reported to them, and implemented at least some content, mechanics, or balance suggestions they received. However, all the developers were hesitant to say that they changed the overall design, as they all had a clear vision of where they wanted to take the game. This truly was the case with most of the games, but there were a couple of games where I would argue that even the larger game design was affected. Even though the developer control over the core design was kept unaffected, some developers had taken suggestions concerning larger additions or changes to the design of the game and implemented them, e.g., Esko’s team discussing large future features and additions with their players before making the decision on which way to progress. I personally would count this towards the larger suggestions to the game’s design, even though the developer would have kept the core design in their hands.

A couple of the participant developers explained that they worked as sort of ‘filters’ when it came to implementing the feedback into the design of the game. This meant that they kept the project going in the right direction considering their core design idea, and they chose which suggestions to consider, so that the project would not derail. However, this of course also means that if the developer thought that an idea was good, it could be implemented even though it would affect the game’s larger design.

In conclusion, all my research participants took player feedback into consideration and developed their games accordingly. The changes ranged from small bug fixes to content, features and mechanics, and in some rare cases even larger changes to the game’s design. Overall, all the developers felt that implementing feedback and working with the community was important.

### 5.3.2 Limitations to incorporating player feedback

Getting and implementing feedback has its limitations. Not all changes are possible to be implemented, and in some cases the developers need to keep control of their design. This was also the case with my research participants, as many developers explained why they did not include all the feedback or kept their own choices over the community's choices.

The biggest limitation that all the participants brought up was not changing the core concept of the game, and not adding anything that was way out of line considering the original design of the game. Esko summarized their policy by saying:

We had a big picture, we started with inspiration from other games and TV shows and so on. And then of course we have our own wishes and ideas inside our team. - - We are working as a kind of filter. There are a lot of suggestions from players that go against the mechanics that we have in the game. So, for example, someone might wish for something where I can straight up see that if we implement it, the game will kind of shatter, and the mechanics that we have in the game now, conflict with the wish. In these cases, we work as a filter, like 'Ok we are not going to go this way'. - - So, in the end, we are the ones that make the decision of what goes into the game and what doesn't.

Daniel also said that they had the basic concept when going to Early Access, and those ideas and features were implemented in the final build. They also had clear boundaries they communicated with the players, including e.g., certain features they said from the get-go that they were not implementing, and some story points they were not going to change during the development. Bettiina also stated that they had a clear plan on how to move ahead before going to Early Access, and they stuck to the core of it when going through the Early Access phase. Aimo concluded that

I think it's important that developers have an idea of the game and how the finished product looks. Player suggestions can be valuable, - - but the end product needs to be coherent and that can't be accomplished by just slapping any and every idea that sounds good into the game.

The developer team keeping control over the game brings coherency and helps keep the game's big picture clear. Frans' opinion lined up with the other developers and he added that "Everyone has their own opinion. - - Players might see some individual things, but they might not see the big picture, so that's the problem." It is important to understand that the developers are the only ones that have the actual design in mind when developing the game, and they need to filter the feedback they get in order to create a coherent game. Even though some feedback is valuable, not all of it is. Some suggestions need to be put

aside for the game to work and be balanced. Henri also said that they sometimes ignore feedback on something that they know will be balanced in the future due to the changes of the game. This also ties to keeping the big picture together. It is important to understand, that in the end the game is the developers' product, and they are trying to create a game they wanted to bring to the world, while balancing between varied opinions and feedback from different players. Even though Early Access is an inclusive journey, the end product needs to be coherent and not every piece of suggestions can be considered. As Thominet (2021) and Ismail (2016) also stated, games are artistic endeavours, and it is important to let the developers keep their artistic choices in place and give them the power to decide which pieces will be added and which not.

Another thing limiting the inclusion of feedback was the developers view of how the game needs to be balanced. In a previous chapter I wrote that it was important to include feedback on balance, as it affects the game experience. However, in some cases this might be challenge, as some games are made to be difficult in essence. Henri said that it is important that the developers have the possibility to balance the game themselves, as the game might need a set tempo or a set difficulty. Gabriel also talked about their experience in balancing their game, and I think that it summarizes well how different games might need different types of control over the feedback when it comes to balancing the game:

We did take feedback about balance into consideration, but with our game it was kind of difficult, because the game itself is challenging and difficult. And when we went through the feedback, we thought about what the experience would be for players of different levels, like you have someone who plays for the first time, and then you have someone who's played a little but still is kind of new, and then is the average player, and then you have someone who's really good at the game, and then the last category we had was players who were like super good at our game and might even understand it better than us. So, when you get feedback from a mechanic that is supposedly unfair or too difficult, you have to think about who gave the feedback and how changing the mechanic would then affect all the other players of these different levels, and its surprisingly difficult to handle the feedback in these kinds of situations. However, I feel like we were able to balance the feedback fairly well.

Game development needs to have some control for the game to stay coherent and playable for a large audience, and with some games it means the developer needs to keep control of the balance, features, or content of the game. In Henri's and Gabriel's case the developer control meant keeping the balancing mostly in their hands, as they felt the balance of the game strongly affected the core gameplay and play experience. This does



not mean that they did not listen to any feedback, but they kept the balance of the game on a level they thought would best suit all the players of the game.

Besides these core design decisions, one clear thing restricting feedback getting implemented was time restrictions. Many developers said that they did not add anything to their development cycle that would significantly affect their development time. After all, the main idea of Early Access is to develop the game and launch it, not add each and every piece of feedback that players suggest to it. Bettina told that their development had strict development cycles and time constraints, so they had to leave most of the feedback out due to not having time to implement them. Daniel said the same thing on feedback concerning bigger features, such as online co-op. He said:

We didn't have enough time to make [online co-op] so that it would be stable and work properly, so we didn't do that. Especially when the project was already ongoing and we didn't plan [the game] to be online co-op, - - it was a huge 'no no' considering the workload.

Gabriel was thinking in a similar manner, as he said that they did not include big content or feature requests, as they were on limited budgets and timetables. He said that they considered implementing things like co-op or creative mode, but the workload was too much to actually finish those in time considering the time restraints. As discussed before, too tight schedules and too large workloads can lead to tiredness and even burnout. Even though the point of Early Access is mainly considered to be the feedback and implementing it, it is important to realize and remember the real-life limitations that game development has. Game development is dependent on funding which brings time restrictions with it, and the game needs to be launched in the end, so adding or changing big features needs to be carefully considered keeping those restrictions in mind.

Lastly, about half of the participant developers talked about the lack of resources limiting the community inclusion. As many developers had multiple jobs in the project (such as being a game designer, head of marketing, and a CEO at the same time), some of them felt that they simply did not have the time to go through all the feedback they got. There were also issues with controlling the feedback, such as how to systematically go through it all, how to categorize it, how to present it, or how to see which issues or features got most of the feedback and which were just single pieces of feedback. Some of the teams had hired a community manager or a PR agency to handle the communications between the team and the community. Aimo talked about his experience by saying that

We didn't have anyone doing community management full time, and I think we should have had. - - This is also something that people can easily be too short-sighted with as you can't build a community in a few days, it needs to be a constant ongoing process throughout the development.

Handling a community and the feedback they give can be a huge workload, so it can bring extra stress on top of the other development tasks, such as programming or designing. Gathering the feedback is one thing, but then processing it is yet another task that needs to be done to understand what the community wants, and to separate that from what just a few loud players want. Daniel talked about the issues they had concerning the amount of feedback:

We had so much feedback, suggestions and all that stuff, that just collecting them and changing one feedback into a few bullet points and when there's hundreds of them and from different channels... It turned out to be impossible to [go through them all] and still develop the game.

Even though all the developers in the end had managed to gather the feedback and at least on some level go through it to implement it, it was in some cases considered difficult due to the restrictions of not having anyone dedicated to handle the community. Some of the participant developers had decided to split the community management in parts so that everyone in the team went through their share of the feedback, and some had dedicated one or two persons in charge of the job on top of their other work tasks. This way the workload did not fall on one person's shoulders, but was distributed evenly to everyone, or to a few dedicated persons. It was very developer-specific how the team handled Early Access and the feedback, but it is safe to say that Early Access brought a new level of workload, and the teams had to adjust to that. This is something the developers needs to take into account when deciding to go into Early Access, but also it is important for the players to understand that the developer team might not have the time or resources to go through every single piece of feedback they get. The developer needs to be transparent and open about their process, so that the players understand the developer team's limitations.

These results show that Early Access and managing the community can be done with a smaller team and by sharing the workload, if the team is open and transparent about their situation and have clear boundaries between their work and free time. However, sometimes it might be a good idea to hire extra help with the community, as the community is after all a crucial part of the Early Access process. Limitations can be expected, as in the end the developer is in charge of the project and its design and balance,

and they have real-life constraints, such as funding and time. The most important thing is to stay open about these restrictions and explain why something is added and something is not, so that the players can get in the same mindset as the developer, and thus can also give more constructive feedback.

## 6 CONCLUSIONS AND FUTURE WORK

The goal of this thesis was to research how Early Access affects game development, and how player feedback affects the development and design of the games in question. In this thesis I have explained why Early Access is used (or why it might not be the right development choice), how developers handle transparency during it, which methods are used in communicating with the audiences and what types of feedback is implemented in the development and design of the game.

### 6.1 Key results

It seems that Early Access is becoming increasingly popular, especially amongst indie developers, and many game developers use it as a way to shift their development towards more open practices. As shown in this thesis, it is clear that Early Access can help developers design a game with the co-operation of its players. However, it is important to remember that the sample size of this research is only eight participants, and my focus was on Finnish game studios, with games for PC or consoles. If this study was done by interviewing developers from different countries or with different backgrounds, or with different game projects, the results may have varied.

The shift towards Early Access means increased focus on player feedback on multiple different channels and implementing the suggestions to the design of the game during development. Early Access affects the game development on many fronts, such as developers having to become communicators due to the open nature of the phase and communicating with the players about what they are currently doing, where they are taking the design and what their end goals are. The developers need to implement channels for being transparent and for gathering feedback. As players become part of the game development process, the developers need to be open to suggestions and willing to adjust their game design. At best the community becomes a voice in guiding the game development and design, and this may result in a game that is better suited for its players. Game developers need to work as ‘filters’ when executing their vision of the game and at the same time go through community feedback and decide which parts to add to the game. Depending on the level of transparency and receptivity from the developers, feedback can also affect the game development on many fronts, from bugs and glitches to game features, content, and balance, and even the larger design in some rare cases. Early Access

affects the game development so widely, that it is also important to remember the limits of developing a game with a community and taking care of good work-life balance and have clear restrictions to implementing feedback.

Early Access can be a successful way to push the game to its launch, and many games benefit from the early funding and marketing. The developer gets more time to develop the game as it is given to the players during the development stage and starts accumulating revenue early on. Players might help with the marketing, as they support the developer by spreading the word to other players. Early Access also gives the game two launches, which might play a part in getting audience's attention. However, amongst the participants in this research, the main reason to use Early Access was getting feedback from the players and developing the game in symbiosis with the audience. Feedback gathered from the players during the development helped the developers in creating a game that was more suitable for the players. Early Access was also seen as a great option when the developer was uncertain about some aspects of the game, or they were trying something they had not done before. This way the community feedback pointed them in a right direction during the development of the game. Other things brought up by the participants were, for example, that players tend to give more positive reviews to Early Access games and be more patient with bugs and issues that they encounter in the game during the development phase. Players also seem to be more interested in and dedicated towards a game that they give feedback on, as they feel like they are making a difference.

At the same time, Early Access may not be suitable for all types of games, and there are many things that can go wrong when giving access to the game in its early stages of development. If there is no communication between developers and players, or if the development team abandons the project during the Early Access phase, it can hurt the whole game company. Sometimes too much feedback can also turn into a mess that is too large to entangle and can cause the project to be delayed or stray too far away from the original idea, or even cause burnout amongst the developers. The development team needs to have goals that they are working towards, and they need to be transparent in restrictions about implementing feedback. Having coherent channels and tools in communicating with the players and gathering feedback ensures that the project stays on track. Being transparent and filtering the feedback will give the best possible base for the project. The development teams also need to make sure all team members have a good

balance between work and free time and that the communications with the community does not leak to their private lives causing issues with coping with work.

The developers participating in this research used a variety of different methods and tools when being transparent and giving out information about the development of the game to the players, and when gathering feedback. All the developers had channels for text-based qualitative feedback, such as Steam community pages, forums, Discord, and so on, where they actively engaged in discussions with the players, and encouraged them to give feedback. Inside these channels almost all developers used things like roadmaps and patch notes to communicate their situation to the players. Adding to the text-based communication, some developers used passive feedback such as gathering game analytics or player saves as well, which was felt to be a good way to gather information about the overall situation. This included the feedback from silent players, which was good, since not every player wants to take part in online conversations. On top of these, different developers used different tools when giving out information, such as livestreaming, social media and events and talks. Besides the tools mentioned for gathering feedback, some of the developers also used tools such as modding and other types of player participation for getting the players to come back to the game and for getting ideas for the game. They also watched live streams and playthroughs to gather information about what works in the game and what not, and social media and e-mail for gathering text-based feedback. A few of them also utilized statistics provided by Steam, game achievements, and in-game polls.

All the participants incorporated feedback into their game development and design. All of them fixed bugs, glitches and crashes reported by the community, and all of them included at least some changes to the game's features, contents and/or balance. Some participants did this more than others, and it varied by the game and the developer on which aspects of the game the feedback was considered. In some rare cases, even the larger design of the game was affected by the feedback, or the developer decided which features to implement with the community feedback in mind. However, in all the cases in this research the final design was affected by the community feedback in some ways. Some games became longer, some had new features implemented, some were balanced better, some had community-made content, some changed features that were weak or did not contribute to the play experience. The games became better suited for the players, because the players were a part of the development process.

There were limitations to incorporating feedback into the game's development and design, which consisted mostly of time and resource restrictions. Most of the developers participating in this research stated that they could not implement all features and changes the community wanted, because there simply was not enough time or resources to include those changes. Game development has time and monetary restrictions, and they need to be met, which means that not every piece of feedback can be implemented. Developers would not risk burning employees out or crunching through a huge workload because of a feature the players would have wanted, if the feature was too big to be implemented in the time frame they had. Another thing restricting the inclusion of feedback was not implementing anything that would clash with the original design of the game. This was the consensus between all the developers. They all had an idea of their game, and they implemented feedback that supported that idea or built on top of it. They worked as 'filters' when applying suggested changes and would not implement anything that would shatter the concept that they already had. Some developers also struggled with the workload Early Access brought with itself, so as they had issues with going through all the feedback and compiling it, they also could not include suggested changes that they missed.

## **6.2 Brief summary and guidelines**

According to the background research and the research conducted for this thesis, there are a few things to consider when thinking about going to Early Access. Here is a brief summary and guidelines for any developer that is considering using Early Access for their game project.

Firstly, think about whether the game in question is a suitable match for Early Access. If the game is based on story and narrative development, the game might not be a good fit for Early Access. If the story is the main point you want the players to experience, it might not be good to let them experience it while the game is not polished. As many people do not want to play the game again after getting the story experience, it might be better to polish the game and release it fully when it is ready. A game with replay value is a better fit for Early Access, as the players can come back to it after patches. For example, games like rogue-likes, sandbox games and base builders have replayability, and are thus better for open development.

Secondly, using Early Access means going open in the development. This means that you have to set up channels for communicating with the players, and for getting feedback on your game. It is good to prepare yourself for an increased workload, and act accordingly. This means either hiring outside help to handle the community relations and the feedback gathering, or making sure everyone contributes to the situation, or dedicate someone for the role of community management, but make sure they are not overwhelmed by all the other work tasks. If possible, use multiple different channels for communicating with the players. Be transparent and open to your audience about your project, your goals, and your progression. Tell your players where you are going with the project and where you are at the moment. Talk about the restrictions you have, whether it being a small development team or strict timetables. All these things help the players understand your situation, and to give better feedback.

Gather feedback using the multiple channels you have set up. Using text-based channels such as forums, Discord or Reddit, gives you deeper knowledge of what is working and what is not, as the players can freely describe the situation. On top of that, if you have the resources, use analytics. Use an in-game bug reporting tool, get player's save files, use Steam's developer statistics. This way you can get knowledge of the bigger picture, such as where players stop playing the game, where they struggle and if there are something they do not understand. With Early Access there are more people testing your game than what you would have with a quality assurance team, meaning that they will test out more scenarios and encounter more bugs and crashes, which you can then solve. Small streams form a river, so use as many channels for communication as you are comfortable with. Be it social media, watching live streams, using in-game polls, or hosting fan-art competitions on Discord servers.

Lastly, be open. Take in the suggestions. You have the possibility to fix more than just bugs in your game, as you can listen to your players and create a game that is truly made for them. You can keep your main idea and create the game you want, but keep in mind that adding player suggestions to your idea might make the game an even better one. You have to balance between your own ideas and the feedback you get, but when you are open to suggestions, your idea and the feedback combined will in the end create a better play experience. Do not include each and every idea, but filter the suggestions based on whether it will make the game better or worse and add those ideas that have value in your game's design.



This way you can create a game with your audience, where the players are a voice in the decision-making process. This can help your game become better, more suited for your players. This can also bring other positive things with it, such as funding and word-of-mouth marketing.

### **6.3 Future work**

Game development is a rapidly evolving field. Due to this evolving, standards and ways to work are changing and shifting. This also applies to Early Access. In the past game development has been done behind closed doors in secrecy (O'Donnell, 2014, p. 42-45), but it has already evolved to more open practices. Indie developers have started to share and distribute their knowledge, and this behavior has grown to be a part of the industry as a whole. As Keogh (2019) writes, "The culture of secrecy and subsequent endless cycle of re-inventing basic systems - - is challenged by this broader, more collaborative community of independent and informal developers less anxious about holding trade secrets close to their chest." Another thing that has opened the development scene and started to include communities in development practices is the shift to the games-as-a-service model. As Weststar & Dubois (2022) writes, "Some scholars suggest that digitization can help firms transition more fully to a servitization model since this allows for the development of 'functionally incomplete' products which can be 're-programmed' in dialogue with customers". I believe Early Access as a part of game development history has also contributed towards this emergence of open practices. Getting communities to work with the developer during the development of the game might help the success of the game, so now both indie developers and larger game development companies are doing it. This has led to game development shifting towards open development and to development companies publishing games that are still developed and/or updated after the launch. We are already seeing a shift towards the expectation of developers updating the game, even if it would have been published as a whole and not in an incomplete state. For example, some well-known games, such as *Fortnite*, have shifted and restructured majorly after their original release.

What this might mean in the future, is that maybe Early Access is just a "phase" in game development history, and maybe in the future game development sees a shift in open development practices overall, without the need to define a separate Early Access stage. One of the interviewees of this study, Gabriel, also said something similar:

There is this expectation nowadays that games are updated after their launch, so the line between Early Access and not Early Access is a bit fickle. And I think that it's more like that the developers themselves can decide if a game is in Early Access or if it's not in Early Access. There is no clear line. - - So, does it make sense to go into Early Access when players might not want to support an Early Access project, because some of them have been done poorly, or they don't notice the game since Steam doesn't give as much featuring to Early Access games, or whatever is the reason? So, then you have to think about if you want to go into Early Access or just do a full launch and then update it afterwards.

Early Access can be a risky move considering that it might not get as much sales as a fully launched game, players might not be interested in supporting an Early Access title, or because Early Access games do not get as much visibility in marketplaces. Gabriel also pointed out that Early Access is also very much tied to Valve and Steam, as Steam is the largest marketplace to host Early Access games in one place. If Valve decides to change the rules or shift Early Access practices, it will affect the model in a fundamental way.

In the future it would be interesting to study whether some companies are more likely to publish their game fully, and then keep updating it, rather than opt for going to Early Access, as players expect the games to be updated anyway after the release. It is interesting to see what happens to Early Access in the future, as it has been one of the models paving way for open development practices.

## REFERENCES

- Arafat, M., Qusef, A., & Al-Taher, S. (2019). Steam's Early Access Model: A Study on Consumers' Perspective. *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, 336–341. <https://doi.org/10.1109/JEEIT.2019.8717501>
- Aktaş, A., & Orçun, E. (2016). A survey of computer game development. *The Journal of Defense Modeling and Simulation*, *13*(2), 239–251. <https://doi.org/10.1177/1548512914554405>
- Babbie, E. R. (2016). *The practice of social research: Research Design, Three Purposes of Research* (14th ed., pp. 90-93). Cengage Learning.
- Baym, N. K. (2015). Connect With Your Audience! The Relational Labor of Connection. *The Communication Review (Yverdon, Switzerland)*, *18*(1), 14–22. <https://doi.org/10.1080/10714421.2015.996401>
- Berg, B. (2017). *Qualitative research methods for the social sciences: An Introduction to Content Analysis* (9th ed., pp. 181-200). Boston: Pearson Education.
- Braun, V., Clarke, V., Boulton, E., Davey, L., & McEvoy, C. (2021). The online survey as a qualitative research tool. *International Journal of Social Research Methodology*, *24*(6), 641–654. <https://doi.org/10.1080/13645579.2020.1805550>
- Braun, V., & Clarke, V. (2022). *Thematic analysis: a practical guide*. SAGE.
- Brown, J. (2015). *The Impact of Open Development on Unreal Tournament*. GDC Vault. Retrieved from <https://gdcvault.com/play/1022760/The-Impact-of-Open-Development>
- Clarke, V., & Braun, V. (2017). Thematic analysis. *The Journal of Positive Psychology*, *12*(3), 297–298. <https://doi.org/10.1080/17439760.2016.1262613>
- Cote, A., & Raz, J.G. (2015). In-depth interviews for games research. In P. Lankoski, & S. Björk (Eds.), *Game Research Methods: An Overview* (pp. 93-116). ETC Press.
- Daro, C. (2021, December 29). STEAM SHARED MOST PLAYED GAMES & TOP EARLY ACCESS GRADUATES OF 2021. Gamespace.com. Retrieved November 15, 2022, from <https://www.gamespace.com/all-articles/news/steam-shared-most-played-games-top-early-access-graduates-of-2021/>
- Early Access. (2022). In *Wikipedia*. Retrieved November 15, 2022, from [https://en.wikipedia.org/w/index.php?title=Early\\_access&oldid=1112380943](https://en.wikipedia.org/w/index.php?title=Early_access&oldid=1112380943)
- Edwards, C. (2013, November 4). *Valve Lines Up Console Partners in Challenge to Microsoft, Sony*. Bloomberg. Retrieved November 15, 2022, from <https://www.bloomberg.com/news/articles/2013-11-04/valve-lines-up-console-partners-in-challenge-to-microsoft-sony#xj4y7vzkg>

- Elahee, Y. (2019, June 13). *Tips & Tricks to manage Open Game Development*. Game Developer. Retrieved November 15, 2022, from <https://www.gamedeveloper.com/business/tips-tricks-to-manage-open-game-development>
- Engström H. (2020). *Game Development Research*. Skövde: University of Skövde.
- Gandolfi, E. (2018). Playing, debugging, learning: A proposal between game and instructional designs via extended prototyping. *E-Learning and Digital Media*, 15(2), 67–92. Scopus. <https://doi.org/10.1177/2042753018757079>
- Google Play Help. (n.d.). *Try new Android apps before their official release*. Retrieved November 30, 2022, from <https://support.google.com/googleplay/answer/7003180?hl=en>
- Goslin, A. (2019, November 13). Early access changed everything about games. *Polygon*. Retrieved November 15, 2022, from <https://www.polygon.com/2019/11/13/20961896/decade-of-early-access-minecraft-star-citizen-fortnite>
- Grinding Gear Games. (2013). *Path of Exile (PC Version)* [Video game].
- Guest, G., MacQueen, K. M., & Namey, E. E. (2012). *Introduction to applied thematic analysis*. SAGE Publications, Inc., <https://dx.doi.org/10.4135/9781483384436>
- Hill-Whittall, R. (2015). *The indie game developer handbook, Chapter 8: Funding* (pp. 213-239). Focal Press. <https://learning.oreilly.com/library/view/the-indie-game/9781138828421/xhtml/Ch08.xhtml>
- Ismail, R. (2016). “*Nuclear Throne*”: *Performative Game Development in Hindsight*. GDC Vault. Retrieved from <https://www.gdcvault.com/play/1023774/-Nuclear-Throne-Performative-Game>
- Jacobs, M., & Sihvonen, T. (2011). In Perpetual Beta? On the Participatory Design of Facebook Games. *Proceedings of DiGRA 2011 Conference: Think Design Play, 2011*, 8.
- Johnson, S., Avellone, C., Cheng, J., Goya, A., Campbell, C., & Sigman, T. (2015). *A thousand voices: open game development*. GDC Vault. Retrieved from <http://www.gdcvault.com/play/1021755/A-Thousand-Voices-Open-Game>
- Keogh, B. (2019). From aggressively formalised to intensely in/formalised: accounting for a wider range of videogame development practices. *Creative Industries Journal*, 12(1), 14–33. <https://doi.org/10.1080/17510694.2018.1532760>
- Kerr, A. (2017). *Global games: production circulation and policy in the networked era*. London: Routledge.
- Kuorikoski, J. (2014). *Sinivalkoinen pelikirja: Suomen pelialan kronikka 1984-2014*. Fobos.

- Lin, D., Bezemer, C.-P., & Hassan, A. E. (2018). An empirical study of early access games on the Steam platform. *Empirical Software Engineering*, 23(2), 771–799. <https://doi.org/10.1007/s10664-017-9531-3>
- Long, S. (2014). *Crowdsharing: Crowdfunding and Crowdsourcing*. GDC Vault. Retrieved from <https://www.gdcvault.com/play/1021428/Crowdsharing-Crowdfunding-and>
- Luck, M., & Day, D. (2015). *Open Development: Taking Desktop Dungeons from Prototype to Pre-Order to Steam*. GDC Vault. Retrieved from <https://www.gdcvault.com/play/1022177/>
- Morgan, D. (1997). *Focus groups as qualitative research: Focus Groups As A Qualitative Method* (2nd ed., pp. 8-18). SAGE Publications, Inc. <https://doi.org/10.4135/9781412984287.n2>
- O'Donnell, C. (2014). *Developer's dilemma: The secret world of videogame creators*. MIT Press.
- Red Hook Studios. 2015. *Darkest Dungeon* (PC Version) [Video game].
- Sandqvist, U. (2015). The Games They are a Changin': New Business Models and Transformation within the Video Game Industry. *Humanities and Social Sciences*, 23(2), 4–20.
- Sansone, A. (2014). Game Design Documents: Changing Production Models, Changing Demands. In J. deWinter, & R. Moeller (Eds.), *Computer Games and Technical Communication: Critical Methods and Applications at the Intersection* (pp. 109-123). Taylor & Francis Group, 2014. ProQuest Ebook Central, <http://ebookcentral.proquest.com/lib/tampere/detail.action?docID=1808804>.
- Schreier, M. (2012). *Qualitative content analysis in practice* (1st ed., p. 1). SAGE.
- Sherlock, L. (2014). Patching as Design Rhetoric: Tracing the Framing and Delivery of Iterative Content Documentation in Online Games. In J. deWinter, & R. Moeller (Eds.), *Computer Games and Technical Communication: Critical Methods and Applications at the Intersection* (pp. 157-169). Taylor & Francis Group, 2014. ProQuest Ebook Central, <http://ebookcentral.proquest.com/lib/tampere/detail.action?docID=1808804>.
- Sotamaa, O. (2021). *Studying Game Development Cultures*. *Games and Culture*, 16(7), 835–854.
- Sotamaa, O. & Svelch, J. (eds.) (2021). *Game Production Studies*. Amsterdam: Amsterdam University Press. <https://doi.org/10.1515/9789048551736>
- Spock, J. (2014). *Bringing the Community into the Dev Team—A Look into Open Development*. GDC Vault. Retrieved from <https://www.gdcvault.com/play/1021475/Bringing-the-Community-into-the>
- Stebbins, R. (2001). What Is Exploration? In *Exploratory Research in the Social Sciences*. SAGE Publications, Inc. <https://doi.org/10.4135/9781412984249.n1>

Thominet, L. (2018). How to be open: User experience and technical communication in an emerging game development methodology. *Communication Design Quarterly*, 6(2), 70–82. <https://doi.org/10.1145/3282665.3282672>

Thominet, L. (2021). Open Video Game Development and Participatory Design. *Technical Communication Quarterly*, 30(4), 359–374. <https://doi.org/10.1080/10572252.2020.1866679>

Valve. (n.d.). *Early Access (Steamworks Documentation)*. Retrieved November 2, 2022a, from <https://partner.steamgames.com/doc/store/earlyaccess>

Valve. (n.d.). *Introducing Early Access*. Retrieved November 2, 2022b, from [https://store.steampowered.com/earlyaccessfaq/?l=english&snr=1\\_200\\_200\\_Early+Access](https://store.steampowered.com/earlyaccessfaq/?l=english&snr=1_200_200_Early+Access)

Weststar, J., & Dubois, L.-É. (2022). From Crunch to Grind: Adopting Servitization in Project-Based Creative Work. *Work, Employment and Society*. <https://doi.org/10.1177/09500170211061228>

# APPENDIX 1. INTERVIEW QUESTIONS

## Interview questions

1. Background questions: age, what is your position in the company?
2. Why did the team decide to use Early Access?
3. Strategies:
  - a. What strategies are used to give out information to the players? How was information handed out to the players (Roadmaps? Forum posts? Streams? End goals?)
  - b. Do you think that transparency about the game state is important to give to the players?
  - c. How was information gathered from the players (Gathering information in written format? Forums? Discord? Social media? Gathering feedback from in-game activities through analytics? Watching streams of players playing the game?)
  - d. Were the players given tools to try and modify the game (Access to mod tools?) or otherwise use their own creativity considering the game (art competitions to include winners' art to the game?) and give feedback based on that?
4. Which parts of the feedback are included in the game design and what is left out?
  - a. Bug reports? Recommendations for content/mechanics/balance of the game? Larger suggestions for the game as whole? Why? How were analytics used? (Helping question to the main question to get more qualitative feedback)
5. At what point of game development did the team move to Early Access?
6. How do you feel about developer control over the game (meaning that the developer gets to develop the game however they see fit, no matter the feedback/response of the players)? Where is the line?

7. How did the team adjust to the workload that Early Access brought?
8. Would you do it again?
9. Anything else to add?