Jussi Prokkola

LiveOps Product Management for Mobile Games

Business Information Technology (UAS)

Spring 2022

KAMK • University of Applied Sciences

KAMK • University of Applied Sciences

Abstract

Author: Prokkola Jussi

Title of the Publication: LiveOps Product Management for Mobile Games

Degree Title: Bachelor of Business Information Technology

Keywords: Product Management, Product Manager, Product Owner, LiveOps, Free-to-Play, Freemium, Mobile Game

In 2022 product management is still a new area of expertise in the gaming industry. It is part of a new movement in games that has revolutionized the way we make games. Product managers came to the gaming industry when free-to-play games became popular in the west. Product management is still dominating mostly in free-to-play mobile games and specifically in LiveOps, but it is making its way quickly into other games as well.

This thesis will go through what product management and live operations, more commonly known as LiveOps, are and whether they are important. LiveOps can include all the changes to a product after the initial release of the application. In this case games. LiveOps optimally produces and handles big data. Product manager's job is to use this data to the product's advantage, but often in games product management is still implemented into somebody else's job and is not done to its full potential.

The thesis will answer the question of how product management fits into mobile free-to-play LiveOps. The problem is that it is not a very studied area, because LiveOps and product management are still relatively both new to games. Although, product management has been historically done by other people for longer, product managers made it popular. This thesis compiles all the fundamental information into one place. It would be another as interesting subject to research product management before LiveOps, but this thesis does not really go there. This thesis answers the question of what product managers do and what they should know in the gaming industry. It can work as a guide to product managers who are interested in transferring from other industries to gaming.

Sources on product management in mobile games are still very limited. Therefore, a lot of the sources here on product management are from product managers themselves instead of literature. YouTube videos are used for example. LiveOps in mobile games on the other hand is slightly more studied area and literature will be used as sources. Some quantitative data is used but focus is heavily on qualitative data.

First this study will go through product management and product manager in short and then focus on LiveOps. LiveOps especially is a bigger concept than what will be gone through here, but the focus will be on mobile and product management specifically. As an example, this thesis will focus on a studio called Irrelevant Games, that was this thesis' author's company. The games were made to a platform called Roblox and they got hundreds of thousands of visitors for comparable data. Even though Roblox is not only a mobile platform, all the same product management and LiveOps practices were in place, as the games are free-to-play.

Tiivistelmä

Kirjoittaja: Prokkola Jussi

Julkaisun nimi: LiveOps Product Management for Mobile Games

Tutkintonimike: Tietojenkäsittely

Avainsanat: Product Management, Product Manager, Product Owner, LiveOps, Free-to-Play, Freemium, Mobile Game

Tuotehallinta on uusi asiantuntijataito pelialalla vielä tällä hetkellä vuonna 2022. Se on osa uutta liikettä, joka on mullistanut tavan tehdä pelejä. Tuotepäälliköt tulivat pelialalle, kun ilmaispeleistä tuli suosittuja lännessä. Tuotehallinta, eli product management, on hallitseva voima edelleen ennen kaikkea free-to-play-peleissä, ja etenkin LiveOpsissa, mutta se etenee nopeasti myös muihin malleihin.

Tässä opinnäytetyössä käydään läpi mitä tuotehallinta, eli product management ja ”live operations,” eli LiveOps ovat, ja kuinka tärkeitä ne ovat peleille. LiveOps voi sisältää kaikki tuotteeseen tehdyt muutokset sovelluksen ensimmäisen julkaisun jälkeen. LiveOps optimaalisesti tuottaa ja käsittelee dataa. Tuotepäällikön tehtävänä on käyttää datasta ilmenevää tietoa tuotteen eduksi, mutta usein pelejen tuotehallinta tapahtuu jonkun muun kuin tuotepäällikön toimesta, eikä siihen näin pystytä keskittymään täysin.

Opinnäytetyö vastaa sellaiseen kysymykseen, että kuinka tuotehallinta sopii free-to-play mobiilipeleihin. Suurin ongelma on se, että se ei ole kovin tutkittu alue, koska LiveOps ja tuotehallinta ovat molemmat melko uusia asioita peleissä. Tuotehallintaa on tavallaan tehty jo pitkään muiden ihmisten toimesta, mutta alalle tulleet tuotepäälliköt tekivät siitä suosittua. Opinnäytetyön suurin tehtävä on kerätä kaikki tieto yhteen paikkaan. Yhtä mielenkiintoinen aihe tutkia olisi tutkia kuinka tuotehallinta tapahtuu ennen LiveOpsia, mutta tämä opinnäytetyö ei kerro siitä. Sen sijaan tämä opinnäytetyö vastaa sellaiseen kysymykseen, että miten tuotepäällikön tulisi toimia ja mitä heidän tulisi tietää pelialalla. Tämä voi toimia oppaana tuotepäälliköille, jotka haluavat tulla muilta aloilta pelialalle.

Mobiilipelien tuotehallinnan lähteet ovat edelleen hyvin rajalliset. Siksi monet käytetyt lähteet ovat suoraan tuotepäälliköiltä itseltään kirjallisuuden sijaan. YouTube-videot ovat käytetty tietolähde. LiveOps-mobiilipelit ovat sen sijaan tutkitumpi alue ja sen lähteet ovat näin ollen myös kirjallisuutta. Jotain määrällistä tietoa käytetään, mutta laadullista tietoa on enemmän.

Ensin tämä tutkimus käy läpi tuotehallinnan ja tuotepäällikön käsitteistön läpi lyhyesti ja keskittyy sitten LiveOpsiin. LiveOps on suurempi konsepti kuin mitä käydään läpi, mutta painopiste LiveOpsin käsittelyssä on mobiilin ja tuotehallinnan kautta. Esimerkkinä opinnäytetyö käsittelee lopuksi Irrelevant Games -nimiseen studiota, joka oli tämän opinnäytetyön tekijän yritys. Pelit julkaistuun Roblox-nimiselle alustalle ja ne saivat satoja tuhansia käyntikertoja. Roblox ei ole varsinaisesti mobiilialusta, mutta kaikki samat tuotehallinta- ja LiveOps-käytännöt ovat käytössä. Julkaistut pelit olivat free-to-play-pelejä.c

Table of Contents

Terms and abbreviations

Product management = In this thesis product management means everything that is often done by product manager if the company has one, but can be done by others, such as a designer. Product management is usually still done by somebody other than product manager, especially in smaller companies.

Product manager = Refers to somebody who was specifically hired to implement product management's principles.

Conversion = Free-to-play games try to converse players to payers.

Onboarding = Giving the players a good first impression of a game.

CPI, Cost Per Install = How much one player costs for the game. Companies try to get players for as cheap as possible.

LTV, Lifetime Value = How much a player is worth on average.

Retention = A metric that tells if the players are returning players.

Monetization = Free-to-play monetization is based on services that the game provides.

Roadmap = A large picture of a game's potential future and where to steer it.

Gameplay loop = Every games point should be able to compressed in to few sentences, which is the loop.

1 Product Management

Every game developer knows that game companies have designers, programmers, and game artists. Most everyone knows about producers, quality assurance, marketing, community managers and many others. Product managers or product owners on the other hand are still quite rare in gaming industry. Product management has seen a major shift from pre-digitalization to digital era. And the same digital era in the form of data has made product management more and more important for every industry, including gaming industry. Where do product managers fit in the established group of game developers? Who are they and where did they come from?

1.1 Introduction to Product Management

"Without a product manager a company will continue to operate pretty well – to a point. Yet with a strong product manager a company can become great." (Anon J., Villaumbrosia C. 2017, 9-10.)

No one supports a company if they are not given a reason. This means that no one is going to buy a product that they do not want. Product manager represents the customer, so the company knows what to do with the product. (Anon J., Villaumbrosia C. 2017, 10.) In games it would be easier to think that designers think about the customers, and product manager focuses more on the business side of the game (Zaguirre I. 2017). It could also be thought that in games representing a customer means that a product manager's job is to find out what the players want and represent them so that the players want to play more and will come back for more. Product manager's goal eventually is to get the players to pay for the free-to-play games. Maximizing revenue requires the players to play as much as possible. Product management is not very common among premium games unless there is free-to-play monetization inside the premium game. In other words, retention and monetization are the critical pieces for the product management to get right if the company wants their free-to-play game to be a success (Herrick M. 2017).

In a way, modern product management started in 1931 when consumer goods company Procter & Gamble hired "Brand men" to focus on the brand through testing and feedback. This influenced the birth of information technology company Hewlett-Packard, more commonly known as HP, and soon after Toyota adopted product management as well. Both companies can give a big credit of their longevity to early adaptation of product management. HP also brought this new idea of customer centric, brand focused, and lean manufacturing to Silicon Valley, where the idea spread.

(Eriksson M. 2015; History of Product Management, n.d.) Lean manufacturing simply put means that the companies started building their products through feedback. Product managers were initially mostly about marketing while leaving the underlying product to others, but later product management was brought to the center of product development (Banfield, Eriksson, & Walkingshaw 2017, 10).

1.2 Transition to Video Games

The idea of Free-to-play business model, also known as freemium, is that a lot of players mean a lot of money, even if most of them are not spending any money on the game (Winchester H. 2014). In the past it was okay to fund free games with advertisements, even if the game did not have a lot of players. Massively-multiplayer Runescape's current CEO Phil Mansell can tell that advertisement revenues were very good, and that is how Runescape also started in the early 2000s. Then the advertisement revenue crashed and Runescape switched to subscription model and many other games followed, like World of Warcraft. (Mansell P. according to Winchester H. 2014.)

A free-to-play boom started in Asia because piracy was so rampant, and the studios wanted to counter that. The games were mostly in browser. These browser games were not that played in the west before Facebook made them popular with games like Farmville at around 2009-2010. People loved the games and the multiplayer possibility. This is when the first product managers came to gaming industry. (Courtemance O. 2018.) Before this other people were managing product managers' tasks, such as managing economies.

One reason why product managers came to the industry is that Facebook and Zynga were both from San Francisco area, which had a lot of product managers (Courtemance O. 2018). It sounds plausible, but the reliability of this statement is hard to check. More importantly for product managers to join was that it was the first time when they could start tracking everything the players are doing, meaning all their clicks and where they spend their time at inside the game. Product managers knew already how to handle and read data, so it was a natural transition. Now almost every studio is tracking their players, or at least they should. (Courtemance O. 2018.)

A little later Apple's App Store, and Google Play Store started growing exponentially. In 2010 mobile game revenues were around $6.7 billion (Rosemary N. 2020). In 2020 the mobile games generated around $116,4 billion (Wijman T. 2021). As the mobile game industry is growing so rapidly,

as well as the whole free-to-play industry, the need for product managers is increasing rapidly as well.

1.3 Product Managers

Product managers are not usually tied to one industry and can switch from one industry to another pretty easily, including gaming industry (Somayanda P. 2019). Some industries require some specific skills that are not used on others. For example, gaming industry increasingly expects basic programming skills from its product managers, because the teams are usually relatively small (Herrick M. 2017). This thesis will focus now on mobile game product managers. Most of the practises work on every free-to-play game out there, but everything will be told from mobile games' perspective.

The figure 1 shows where product managers fit in a team (Banfield etc. 2017, 4). It is from a product management book that does not specialize in games. Game studio Zynga's product manager Ponappa Somayanda, current director of product, agrees and used the same picture in his product management speech but replaced the "UX," which means user experience, with "Creative" (Somayanda P. 2019). So, everyone agrees that the product manager is in the middle of a lot of other things. It means for example that product manager needs to speak the same language as programmers. They need to understand logically what a programmer is telling them, even if they cannot do programming themselves, and they should be able to continue the conversation. Still, product management is a business function above else (Banfield R. etc. 2017, 4).

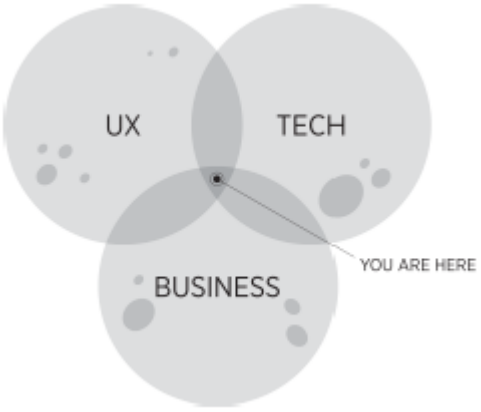

Figure 1. How Top Product Managers Launch Awesome Product and Build Successful Teams (Banfield etc. 2017, 4)

1.3.1 What Product Manager Does

EA's product manager Olivier Courtemance expects his new associate product managers to do three things in the beginning and shares a good list of their product managers' initial responsibilities. It is very important to note here that he is talking about associate product managers. Senior product managers can have different responsibilities.

Competitive analysis means that they should play a lot of games and see the competition. If EA was making a match-3-type of a game for example, the product managers should play every match-3 game they get their hands on and deconstruct why they work, or do not work. (Courtemance O. 2018.)

Content management means that product managers are the owners of everything that the players see every day. This includes all the things that keep the game fresh. The game is already designed at this point, meaning that the gameplay is mostly going to be what it already is. Product manager is the one updating the shop, designing with designers what kind of new content the game is getting, such as Christmas recipes for a cooking game during Christmas. Then product manager figures out the costs for those recipes. (Courtemance O. 2018.)

Investigation and Data Analysis then measures the reaction from this new content through data. The data will optimally tell a story and a company needs somebody to translate that story to spoken language. This is the defining feature of product manager in the gaming industry. (Courtemance O. 2018.) Even the top game right now needs to track everything inside their game, because they must be ready for what is coming in few years.

As an example to the data analysis, if the revenue dips during the weekend and the product manager need to find out why, it could be something simple like less money spent on user acquisition during the weekend, which means less players, which means less revenue. (Courtemance O. 2018.)

In the figure 2 Rovio's former product manager and head of studio Michail Katkoff shows his interpretation of what a product manager does. He clearly goes into much more detail, which probably means that the picture also includes things that product owner does, such as influencing roadmap. Depending on the company, product owner can either influence the roadmap, or set the roadmap. So, they can either ask and convince the team if their roadmap is what the team

wants, or product owner is the one deciding ultimately which roadmap is the best for the game. The roadmap can be set for even the next 12 months ahead (Somayanda P. 2019).

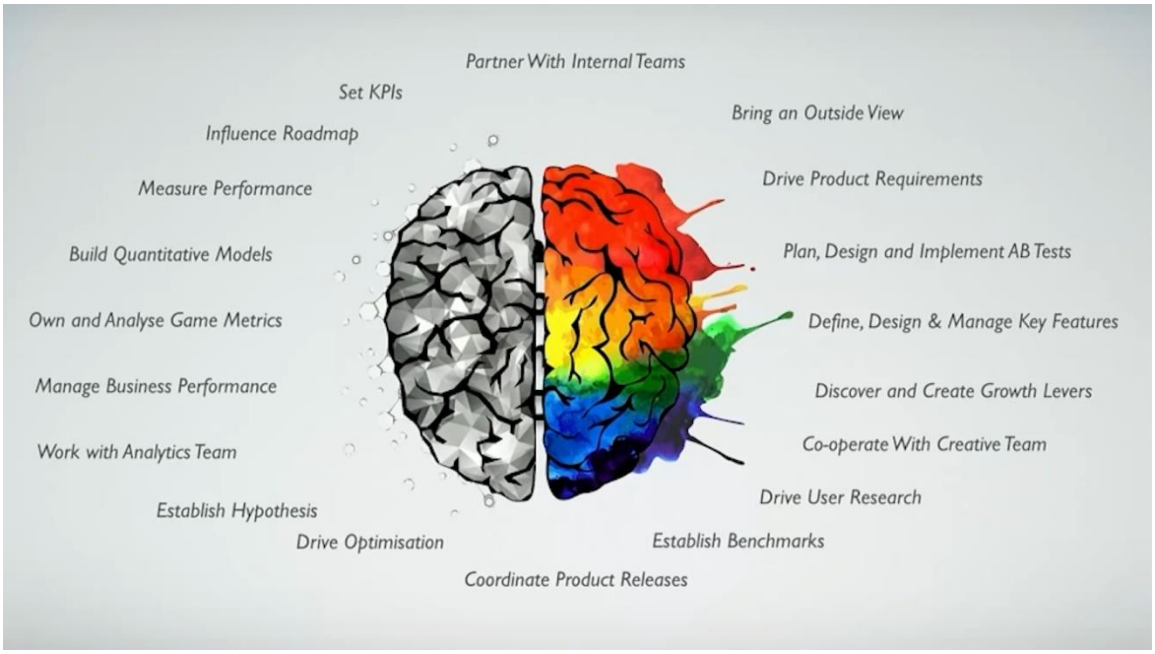

Figure 2. What Successful Product Managers Do Differently in Free-to-Play Games (Katkoff M. 2018)

In the figure 3 Mr. Somayanda shares his vision of what product manager does. Immediately it is clear that Mr. Somayanda and Mr. Katkoff share a lot of the same key points. Katkoff left “Know the market” out of his own list, unlike Somayanda and especially Courtemance, who goes into very much detail about the subject. It could be because Katkoff thinks it to be so obvious, or he just focused on other things. Interestingly Somayanda also says “Be a Player Champion”. It is a good reminder of the product managers origins as the representative of the customers.

Figure 3. Product Management In Games (Somayanda P. 2019)

Mr. Katkoff says that product manager has two key goals. The first goal is to own the business side of the game. Basically, the business can be divided into cost per install (CPI) and players life-time value (LTV), product manager represents the LTV. Which means that they focus on metrics such as growth, retention, and monetization. Product managers make most of their decisions thinking about these things first. The second goal for product managers is to help leadership make decisions, because they cannot always pull the trigger themselves. Product managers focus on data, so they are valuable advisers. (Katkoff M. 2018.)

As another concrete simple example of what product manager does, Katkoff talks about “low hanging fruits”, which means that there are some simple improvements and solutions to games,

that the product manager can spot (Katkoff M. 2018). These simple solutions do not usually need a lot of work after they have been spotted but can improve metrics by quite a lot.

Game studio Rovio was working on a game called Angry Birds Match, which is an Angry Birds™ match-3 game with a lot of events. In one of their events, they added a leaderboard. When the product manager was looking through the data, he noticed that the leaderboard increased player engagement and retention by a lot, because after finishing the event, their highly engaged players stayed in the event to get higher on the leaderboard. After the product manager noticed this, he made a hypothesis: What if, after a player finishes the event, he can finish it again? The player can get the price twice. It is a very small change to the game. Yet the result was that the revenue increased by 20%. (Katkoff M. 2018.) Higher engagement and retention increased their revenue because players spend money only when they are playing.

The product managers do not only need to use their own game's data. A website called App Annie is every product manager's best friend. It offers a lot of data with a subscription model. Katkoff told another simple example. Dawn of Titan's product manager saw from App Annie that their competitors did great with events, but Dawn of Titan's did not have any events on their roadmap. They then made a simple event to test events out, where the players had slightly better chance for better loot for three days. The studio hit their highest average revenue per daily active user, also known as ARPDAU, during that time. They started focusing events after that. (Katkoff M. 2018.) This thesis will go into more advanced examples later.

1.4 Professions

Product management in games was historically done by other people than product managers. Ideally for every product manager out there, every studio would have their own product manager. The reality is different. Some studios are smaller, and some do not prioritize product managers that high and split their tasks to other people. It is also possible to have product managers do other things inside the company and call them something different. Product manager is a new profession in gaming industry, and the variety reflects that. Here are examples of people who can be doing product managers job, or part of it, one way or another. After that, roles that support product management.

1.4.1 Working on Product Management

Product manager is often the title for somebody who does product management. They discover what players are looking for through data and looks to expand the product with their knowledge of the best monetization models (Yankulov K. 2020). As gaming companies are usually relatively small, product managers are often product owners (Herrick M. 2017).

Product owner is traditionally a product manager who is responsible of a specific product (Yankulov K. 2020). Product owners can lead a team of product managers and unlike on other industries, gaming product owners often really own the game and are accountable for the product (Courtemance O. 2018). Some companies only want their product owner to support the team and want them to convince the team to make the changes that the product owner wants. Data helps with the convincing, but it needs to be presented well.

Designer is usually the one who makes the most product management calls in the development phase of the game if there are no data personnel. This might also happen if there are data personnel, but they are not that experienced with designing aspects of the monetization, or if the company prioritizes designer's ability to design the game over product manager's ability.

Project manager, more commonly known in the gaming industry as producer, is often making a lot of business decisions. They are already out of the hands-on development, so they can easily transition to other things simultaneously. Producer is often confused with product owner. The key difference between producer and product owner is that traditionally the product owner is managing the big picture, which means the roadmap in large, whilst producer focuses on meeting deadlines of that roadmap.

CEOs are surprisingly often former product managers, so those CEOs can do product management as well, if they have the time.

1.4.2 Working with Product Management

Depending on the studio, different people are working with product management. Some are part of the management, marketing, or development teams, some are part of the potential data team. Here are few examples.

Data scientists could be the most important role for product manager. Product manager's decisions are based on analytics, with a little bit of intuition. Data scientist finds new ways to capture and analyse data.

Data analyst interprets the data and makes sense out of it. As game development teams are often so small, data analyst is often either the product manager, or data scientist. Social Point has one data scientist, one data analyst and one product owner per game (Zaguirre I. 2017).

Product marketer focuses on CPI, which is the opposite of what product manager does. It means user acquisition and marketing (Zaguirre I. 2017). Product managers are often called mini-CEOs, because they interact with, and coordinate so many people (Banfield R. etc. 2017, 11). Including product marketing, the opposite of what product managers usually does.

Customer support offers product manager some key information about players' happiness.

Director of product is managing multiple product teams in huge companies.

2 LiveOps

LiveOps is an idea where games are treated like a lasting service. In the old waterfall model game studios finish the game, launch, sell it, and then later possibly launch few DLC's. Usually in leaner model of LiveOps the game studios continue developing the game after launch. The difference between DLC's and LiveOps is in the focus. Premium games focus on the main game and then maybe add DLC's, while LiveOps games usually focus on LiveOps, and might not even give that much attention to the initial game at launch, depending on the studio.

Waterfall methodology is used in hardware and usually in premium games. It means that when product is built with waterfall approach, it moves through a very defined process until the end, like a waterfall, and it almost never changes (Anon J., Villaumbrosia C. 2017, 17). Lean approach is better for LiveOps, as it is more flexible. The idea in lean methodology is to release something relatively small, learn from it, and use the information to make it bigger and better (Anon J., Villaumbrosia C. 2017).

LiveOps is where product management and product managers shine. This thesis aims to give the reader a good idea of what LiveOps is and how product management fits in it.

As LiveOps keeps the game fresh, it is probably the biggest reason League of Legends and Fortnite keep staying on top of the charts, as the figure below shows. Interestingly Candy Crush Saga still holds 6th place and Pokémon GO 7th in 2021 December top grossing mobile games (Chapple. C, 2022). This many years on top of the charts is only possible with extremely good LiveOps.

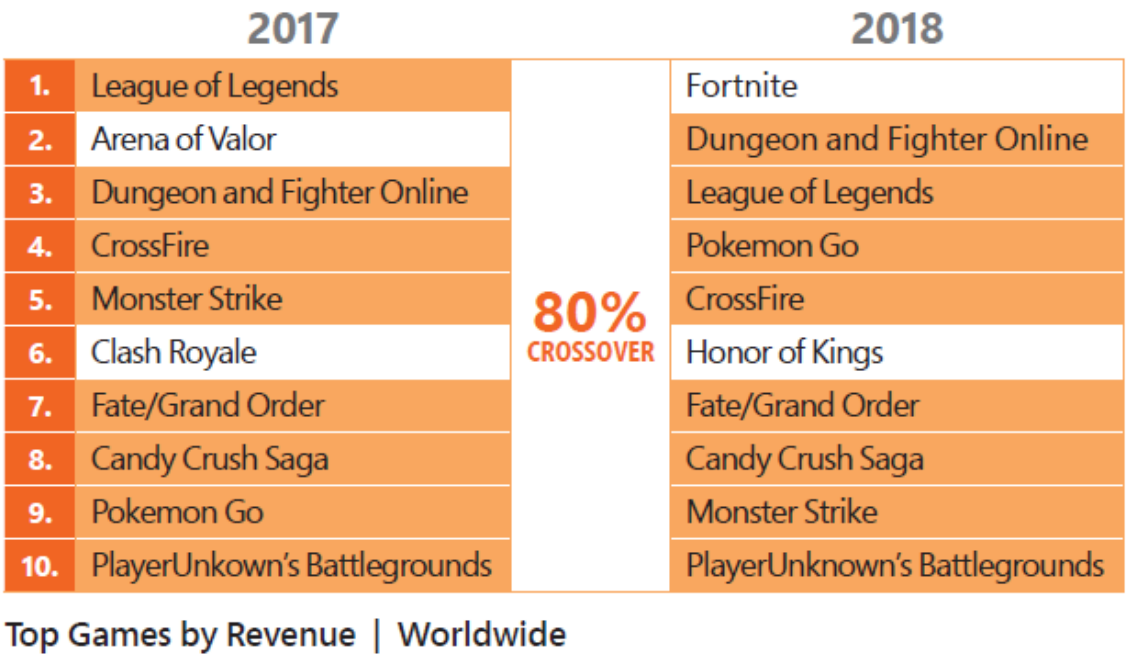

Figure 4. The Azure PlayFab Guide to LiveOps (Azure 2019, 7)

2.1 Introduction to LiveOps

New releases only represent a part of game studio Ubisoft's business now, according to their CEO Yves Guillemot. Ubisoft is more focused on long-term engagement with their players. The players play for longer at a time, but also for over a longer period. Even months and years. Ubisoft wants to offer them new experiences and content, which makes the lifetime of their games longer. (Said by Guillemot Y. according to The Azure PlayFab Guide to LiveOps 2019, 6.)

LiveOps focuses on players in its game design, as opposed to its older developer focused counterpart (Cox C. 2020). It happens through data, which the players provide by playing the game. Reacting to data means that the development stays novel. It is also a way to quickly see if the game has problems (The Azure PlayFab Guide to LiveOps 2019, 7). These problems can be then addressed. It is a lot safer way for the developers to publish a game, because the problems can be spotted quickly, because there are so many players compared to studio's internal testing. The developers can also see if the game is interesting enough and improve it if needed.

As the figure 5 shows, even a LiveOps game, that gets a bad start, can eventually turn out to be more than a decent premium game. In this case the orange line's premium game is going to die soon unless there will be a new DLC. There are no more purchases. While the LiveOps game is

only getting stronger and bigger, like the grey line indicates. Even after that game stops growing it can keep being profitable with good strategy and scalable backend services (The Azure PlayFab Guide to LiveOps 2019, 13).

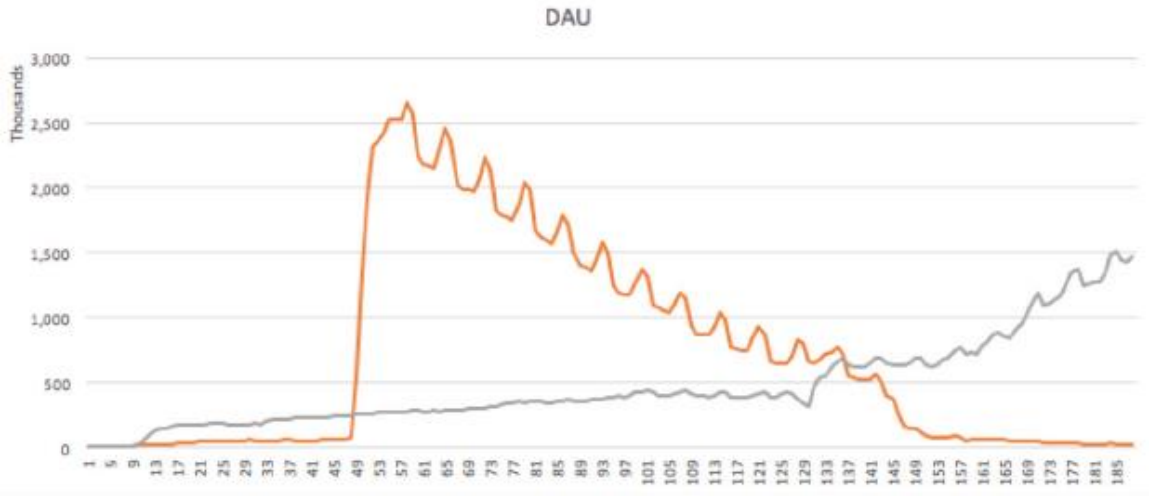

Figure 5. The Azure PlayFab Guide to LiveOps (Azure 2019, 8)

Instead of the old waterfall model, the developers optimally use a newer “lean” methodology (Banfield etc. 2017, 7). This allows the launch to happen a lot earlier with a less-polished and less-fully-featured game. It provides measurable data a lot faster. The next figure’s loop is called build-measure-learn feedback loop. (The Azure PlayFab Guide to LiveOps 2019, 8.)

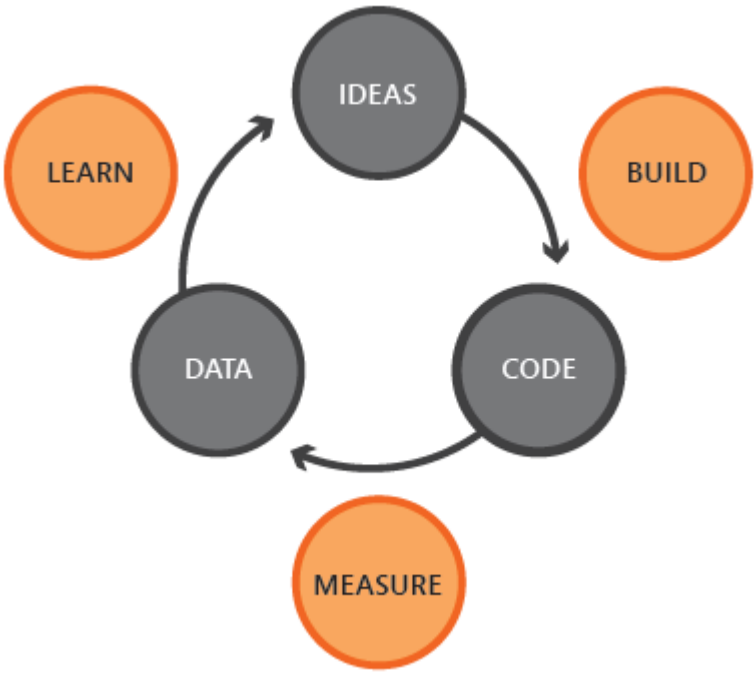

Figure 6. The Azure PlayFab Guide to LiveOps (Azure 2019, 8)

Figure 7 indicates where the build-measure-learn fits in the full lifecycle of a game. To put it simple, after launch the developers need to get new ideas, build, and code events among other things, get data and measure it, learn from it, and get new ideas again.

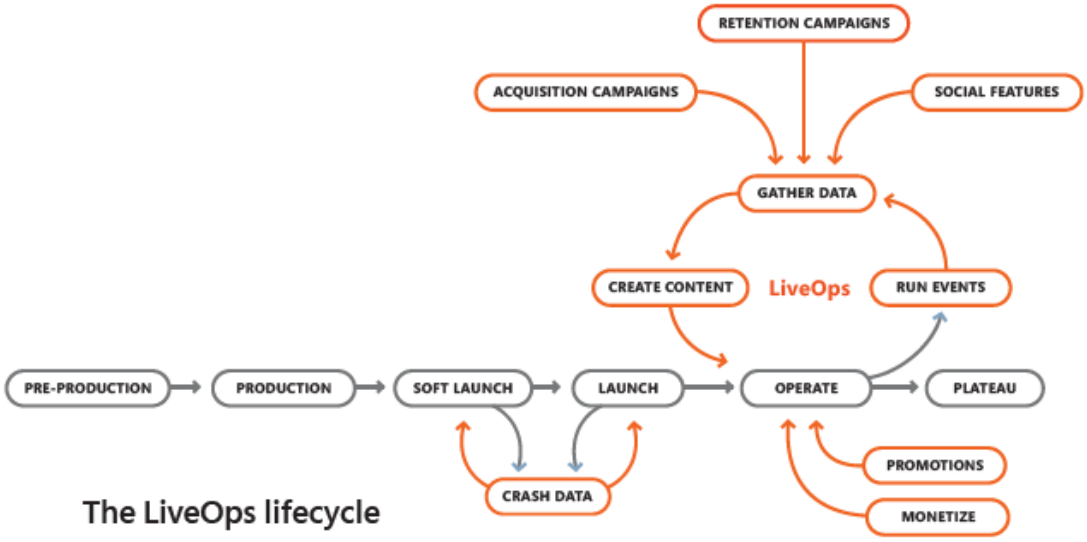

Figure 7. The Azure PlayFab Guide to LiveOps (Azure 2019, 8)

2.2 Players

LiveOps is something where the game needs to think about its players specifically. The idea is no longer to make a game and see if there are people who like the game. The idea with LiveOps is more about making a game and evolving to become always more likeable for the players. “When creating a game with the LiveOps mindset, you are not an auteur film director crafting your singular vision but a creative director of a television station, curating content and monitoring your audiences’ tastes” (The Azure PlayFab Guide to LiveOps 2019, 16).

2.2.1 Player Acquisition

User acquisition, also known as UA, refers in this case to gaining new users for an app. On mobile it is a strategy to achieve more installs. It is usually accomplished by advertisement and promotions (Adjust 2018).

There are multiple ways of acquiring players and product managers can have their hands on everything. Usually, they are divided into subgroups. A big portion of them is marketing, such as paid acquisition, cross marketing, and influencer marketing (The Azure PlayFab Guide to LiveOps 2019, 18). Or designing, such as the game's core loop, which can attract players.

Social installs are something that product management helps with directly. The data produced is something that is measured by analytics team and then improved. Social installs are player installs through receiving a direct or indirect invitation from a friend, who already plays the game (The Azure PlayFab Guide to LiveOps 2019, 16). The success depends on making it as easy as possible to invite friends and give out suitable rewards for both the inviter and the invited. Data and testing are the key.

2.2.2 Engagement

Engagement measures if players have something to do (Agarwal 2021). It communicates how strong the community is by seeing how much time they spend in the game. The community that is active plays and promotes the game, talks about the game, and offer feedback and suggestions. (The Azure PlayFab Guide to LiveOps 2019, 20).

Engagement measures, depending on the game and genre, either how long or how often the game is played or opened. Some games prefer to be played for a long period of time. Some, mostly mobile games, prefer to be played for short periods multiple times a day. It is good to remind the players to open the app accordingly.

Monetization can directly increase engagement. The best conversion item in player's first purchase is something that engages the player, which means that they would play more (Telfer A. 2017).

For example, Heroes Charge mobile game offers the players a package, where the player pays a small amount of money to get premium currency over a month. The catch is that the player needs to log in every day to collect it. It gets the player invested. First, they paid money, which already invests the player psychologically. Second, they don't get value for their money if they don't log in. (Telfer A. 2017.)

Engagement is essential for any good LiveOps. Some companies, such as Riot, only measure engagement and think it is the most important key performance indicator, KPI. (The Azure PlayFab Guide to LiveOps 2019, 20). This is an interesting statement. Although engagement can be a broad word depending on who measures it, it is safe to assume Riot collects other data and measures it as well. They might then see how the other data relates to engagement, and how the other data could improve their engagement specifically, but this statement, as it stands, sounds improbable in 2022.

2.2.3 Retention

Good engagement means that the players have something to do, but retention answers to the question of will they come back day after day to do that thing (Agarwal, 2021).

Retention is one of the most important metrics to track in mobile games, and to predict how successful a game becomes (Sobolev J. (N.d.). It tracks which percentage of players return into the game after the initial launch, and it is usually measured in three periods, Day 1, Day 7, and Day 30 (Herrick M. 2017). A game usually loses a certain number of players per day, while it tries to lose as few players as possible.

The formula to get the rate is the following:

Day N retention (%) = (Number of users who launched a game app on Day N) / (Number of users who downloaded that game N days ago) x 100 (*Reloadly* 2020).

LiveOps games want to keep the game engaging for a long time. An easy-way-out to improve retention is to make the games more entertaining, but it is not always so easy as it can sound. There are multiple strategies for better retention. The following examples are by Playfab.

Content is the most common strategy (The Azure PlayFab Guide to LiveOps 2019, 19). It means to add more content, such as guns, maps, story, or whatever the players want. It is probably the

most common strategy because everybody does it at some point. It does not necessarily mean it is the most effective strategy and developers should do this among other strategies.

Investment is another good strategy, which means to let players invest time and resources into progression. (The Azure PlayFab Guide to LiveOps 2019, 19.) As the player invests time and resources into progression, such as character or Supercell's game Hayday's farm progression, it becomes more valuable to the player. The psychology is the same as with TV series. If the customer watches a series for multiple seasons, it is more likely they will finish the whole thing. The game is doing something right if the progression is built around something monetizable. It increases the retention even more if the player has invested time as well as money into the same progression.

Social relationships offer some of the highest retention a game can achieve and is the third strategy to follow. (The Azure PlayFab Guide to LiveOps 2019, 19.) It is smart to build social networks and offer the players social clubs, such as guilds, for them to work towards similar goals. It can create some pressure for the players to come back, so they would not let anyone down, or they might be kicked out of their guild. Players might even form relationships (Courtemanche O. 2018.)

Core Loop is the last strategy Playfab talks about. (The Azure PlayFab Guide to LiveOps 2019, 19). But there are many more. This strategy basically means that no matter what strategies the game follows, the game should be good. If it is not good, the focus should be on making it good as soon as possible with all available resources. Monetization is often good to include as part of the core loop. One way to do that is to have an increasingly harder gameplay loop, which eventually needs a lot of grinding or paying money to get over.

Reminding the players to come back of course affects retention as well as engagement. Daily rewards are also a good reminder to get players to come back to. As an example, in Heroes Charge players get increasingly better daily rewards. If a player completes a sign-in award, they unlock a new set of rewards. Essentially, they are unlocking content to come back to. This might give the player the needed push to get over a gameplay loop. (Herrick 2017.)

With enough players and time, it is possible to answer if the game is good, and why at least some of the players are losing interest in the game. It is possible to find out the problems and then fix them. Day 1 retention tells if the players like the game initially. It is called onboarding. Day 7 tells if they still like it after getting a feel for the mechanics and characters. Day 30 retention tells how many loyal players are left after so many chances of stopping. (Sobolev. J. n.d.)

2.3 Data

Currently almost every website and application collect data of their users. We have all heard the horror stories of Facebook listening to us and offering us airlines and cruises after we talked about visiting a different country. Data is more valuable than ever, and increasingly so (Broda E. 2019). It is used to determine how people behave. And like in the Facebook-example, it is then used to offer specific services for very specific people, and a lot more. The same applies to games.

Data is to discover and understand the moving pieces withing information. IBM's big-data expert Jeff Jonas states that one needs to let the data "speak to you." Humans have tried to understand data for a long time in the past. Now we have quantified units that can be put into powerful algorithms. (Mayer-Schonberger V., Cukier K. 2013, 2.)

2.3.1 Analytics

Game analytics can be used to measure many different things and is the basis for product manager's job. First, to get the basic understanding of analytics here are some of the most common metrics collected in LiveOps. They are very well known by everyone working with data.

Average Revenue Per User, more commonly known as ARPU, describes the general health of the game. (The Azure PlayFab Guide to LiveOps 2019, 21). This metric is gotten by measuring all the individual players from a certain period and dividing them by the total revenue earned during this time (Sisense Team n.d.).

Average Revenue Per Paying User, ARPPU, tells more about how well the game's monetization is working (The Azure PlayFab Guide to LiveOps 2019, 21). This metric is gotten by measuring all the individual paying players from a certain period and diving them by the total revenue earned during this time (Sisense Team n.d.). Games must balance ARPPU and ARPU in a way that generates the most revenue. Higher ARPPU might mean that there are fewer paying users, so lower cost items or services might make the total revenue higher with more paying users (The Azure PlayFab Guide to LiveOps 2019, 21).

Conversion Rate is a metric that checks how many players become payers. There are multiple ways to define conversion rate. To calculate the percentage of users who have made an in-game

purchase is one of the ways. It can be calculated by viewing all the players who have made a purchase and divide the number by the number of total players. (Sisense Team n.d.)

Lifetime Value, LTV, is a metric that focuses more on telling how high game's revenue over invest is for each player. It is gotten by measuring all the individual players and diving them by the total revenue ever generated by the game. It is an important metric because the marketing team then knows how much they are willing to pay for one new player for example. A company does not want a new player to cost more than their LTV will be on average. (The Azure PlayFab Guide to LiveOps 2019, 22.)

These analytics offer a good basis for more informed decision making, but every game want to have their own custom game specific analytics. It is often product manager's job to know what data to collect and how that data could help in the larger scale of things. To have a better understanding of the analytics here are some examples.

In social mechanics, such as guilds, alliances and clans players share common goals and chat with each other. It can lead to better retention if the players log in more to chat with their friends. It can lead to more installs if players invite their friends. It can even generate more revenue because of the shared goals. Product Manager's job is to go through a lot of data and see how he can manipulate these unique things to move in the direction he wants them to move. When he can do this, he can control the game, and more importantly control the business. (Courtemanche O. 2018.)

Visualizing data is important, as seen in the figure 8. Seeing how the pictures live and breath, and comparing them to each other is the key to successful product management (The Azure PlayFab Guide to LiveOps, 2019). It is hard to tell if these graphs are from the same time period or even from the same game, but if they were, one could make a conclusion that there is a bug in the purchasing part of the game. When the purchases spike, also the total errors spike. So even though there are a lot of purchases, with 300 thousand errors it is safe to assume that there could have been even more purchases. Depending on the severity of the error. This is a very simple example of collecting and reading data, that does not take into account what happens before or after the

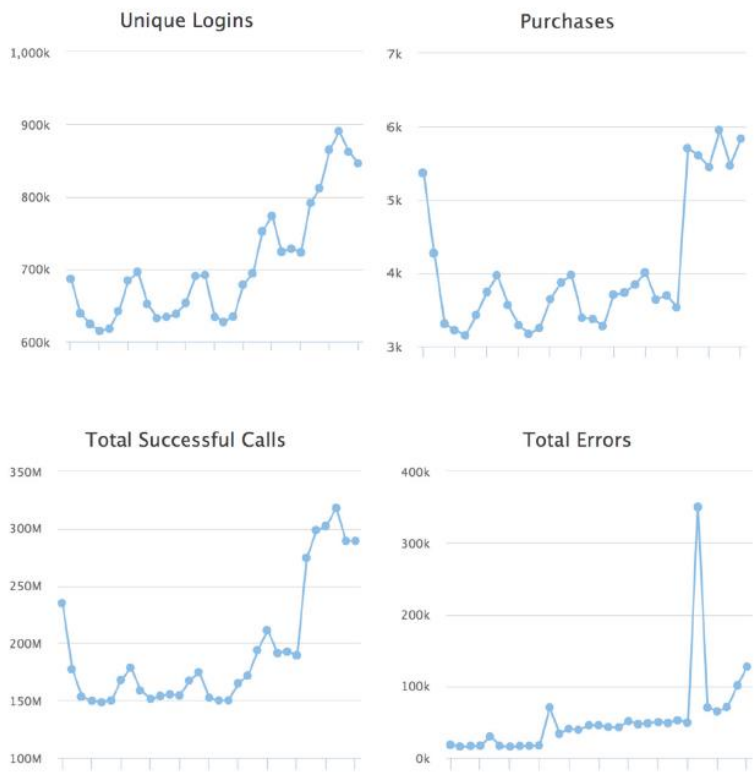

Figure 8. The Azure Playfab Guide to LiveOps (Azure 2019, 24)

Rovio's product manager Michail Katkoff talks about an example of a situation that he in his own words didn't do right in the past, and what his colleague from a game Dawn of Titans did right. As a lead product manager, he had a goal to increase the baseline revenue by 20%. He did not break down where the revenue comes from, nor add new features to affect the engagement or retention, which would have been good things to do. Instead, he added a lot of sales. As a result, the revenue increased, but in the end, it only changed the player behavior. The players started to wait for sales to buy things. (Katkoff M. 2018.)

He then proceeds to give an example of the right way. In Dawn of Titans their product manager got a goal to increase the revenue of events by 20%. Their approach was to see how the best players in the event were behaving. Data told them that the best players were the most engaged, had the most sessions, were the most active as well as very active in chats. They tweaked the game to push every player to act more like these highly engaged players. They hit the target. More importantly, it was a fix that kept hitting the target. (Katkoff M. 2018.)

To support Katkoff, Zynga's product lead Agarwal V. gives a list of what to do, in order, in a hypothetical situation where manager tells the product manager that revenue fell by 50% overnight

and it needs to be fixed. First, they need to see if the revenue is still low or falling, to see the urgency of the situation. Then they need to gather the team for help and see if the game is working properly. Third, they need to see if the number of players dropped for some reason, like less money spent on user acquisition. Fourth, to see if there are some unpopular sales currently. If not, maybe by coincidence the payers were not in the game, even if there were normal number of players. If that does not explain it, maybe it was about the sales' prices. Possibly yesterday the most bought product was $10, and today it is $5. Then, they should see if the new users acquired are from countries that spend less, and similar things to that. If nothing seems to work, then it is correct to see if the analytics tools are working properly. (Agarlwal V. 2021.)

After all the basic data that games are gathering, such as retention, engagement, monetization, and game performance, product managers also gather game specific in-game metrics. For example, how many players are doing something per day, such as battles, and how many times. This is then used to provoke a certain behavior, after something interesting is discovered through data. Also, how much resources the players are having and how they are spending them. Again, to provoke a certain behavior. Then it is possible to see if the players can be divided into categories, so that they could be offered more individual services. (Zaguirre I. 2017.)

2.3.2 Segmentation

Segmenting players means to divide them into different groups depending on various things, like in the previous example. The product manager of Dawn of Titans knew the best players through data. They use the term best players, because they made Dawn of Titans the most revenue, and wanted to guide other segments to play like them as well. That is the power of targeted data.

Custom segments can include high XP players, currently active players, lapsed players and players who have made purchases and lifetime value tiers. At minimum players should track the basics. Some simple segments are "new players", "non-spenders", "spenders", and "high spenders" (The Azure PlayFab Guide to LiveOps 2019, 26). Different segments prefer different content and monetization offers. (El-Nasr M, Drachen A, Canossa A, 2013, 87). With these basic segments games can track easily when players move up from new-players upwards (Azure PlayFab Guide to LiveOps 2019, 26).

With segmenting it is possible to create a personalized path for each player. The players can be divided into groups, then subgroups, then more subgroups. Eventually with algorithms the game

can offer very specific services for each individual player, considering what different players want and need. “Surface the right content to the right player at the right time” (Perez A, 2018). The areas where the game should offer different content include messaging, stores, game configuration, offers and promotions, limited content, and events (Azure PlayFab Guide to LiveOps 2019, 26).

It is very important to invest in the design of game’s backend tools, so it can treat players differently. It should be considered at the development phase. It is also worth considering to program machine learning for the game, so it can automatically segment players. This is the way to target the correct promotions and events to right players, and eventually do the “segments of ones” where everyone gets a unique experience tailored for them. (Azure PlayFab Guide to LiveOps 2019, 26.)

Segmenting is also the way to test the game after its initial release by having one group experience something and then having a control group experience something different. Then the data should tell which group liked the experience better.

2.3.3 Testing

It is impossible to develop games without testing them often in the development phase. LiveOps games should also test the game often in LiveOps phase. Testing should always be done with a hypothesis or a goal in mind, or else it might not give a clear vision of what the conclusion is when the results arrive (Azure PlayFab Guide to LiveOps 2019, 26).

Segmented testing can be done with traditional A/B testing, which means to have version A and version B of a feature, and then test it on two different groups (A/B Testing - Overview n.d.). GameAnalytics offer ready solutions to A/B testing for companies, but they do not currently offer multivariate analysis, which is a different type of segmented testing. It means that games can also measure more than two groups for more statistically significant results, which is often thought of as the “golden standard” (Azure PlayFab Guide to LiveOps 2019, 27). Game developers frequently only use the term A/B testing for any split testing, but it is safe to assume that often these spoken A/B tests target more than two groups, since it is more reliable.

According to Dynamic Yield chief marketing officer Yaniv Navot part of the “golden standard” is to do the A/B testing on specifically segmented players, and not just random groups. It is an easy

trap to fall into. Tests produce the best outcomes if they are also done on subgroups, instead of "average" players. (Navot Y. n.d.) This is not usually thought of or mentioned. Dynamic Yield is not a game company. What Navot says is still relevant, because Dynamic Yield focuses on helping companies to build and test customer services.

The point of A/B testing is to test which solution, for example a feature, monetization model or UI works the best and achieves the goal. It is important to note that a social game cannot give one group a cool tournament and other group something else, because people would get angry (Herrick M. 2017). With two groups in the game the other group works as a reference point. It is part of product management to know what kind of testing and data the game currently needs and figure out if it is A/B testable. Treating A/B testing like a feature, and implementing it in development phase is extremely important, because every LiveOps game needs to be testing all the time (Herrick M. 2017).

It is very hard to determine when data becomes reliable. There are games with hundreds of thousands of downloads, but they cannot get accurate data because they don't have enough daily active users. When a game is not massively popular, it reduces the number of tests it can run and it must measure only big effects, so the data is clearly reliable. (Azure PlayFab Guide to LiveOps 2019, 27).

2.4 Events

Events are usually short and new things for a game. The initial description is vague because events can be so many things. They keep rotating in and out of the games and giving players something new to do. Data can answer the questions of what to give to the players and when. Product manager knows to ask the right questions from data and then provide.

Successful events are arguably the most important part of LiveOps. They have been so successful in free-to-play games, that they have been implemented into many premium games as well. For example, Overwatch has had events every couple of months, where they add new skins into the game for a limited time, behind a loot box. They can be bought with money if the player is willing to gamble enough with the loot box mechanic. They can also be acquired by playing a lot. Both situations are good for the game. This is a very simple form of an event, and a good example of one. Matt Herrick calls these events **"limited-time content events"** (M. Herrick, 2017, 21:46). This section will use Herricks list as a basis for event types.

A spending event is more straight forward. The company gives the player prices for spending money. It is extremely important to have very cool things in the shop, or it does not work. (M. Herrick, 2017.) In free-to-play PC game League of Legends there are different skins in the shop for their countless champions. The game also releases a new set of skins every now and then, usually always slightly cooler than the last patch. Players can then buy those skins directly for a price. The event is on-going. A big reason it works so well for League of Legends is that the players get attached to their favorite champions, and they then want to look different from others who are playing the same champion.

A fusion event is a style that originated from Asian RPG games and are now popular all around the world (M. Herrick, 2017). Often in match-3-games especially there are multiple different heroes. A big part of these games is to fuse heroes together to make them stronger. They can have events around this gameplay mechanic. For example, giving double experience points for fusing two of the same class heroes together.

Double experience events do exactly what they promise. They give the player double experience points for a certain time. The point of these events is that the players play more, and while doing that, they are more likely to spend money. (M. Herrick, 2017.)

Tournaments are one of the more popular ones. Social events are very good and build long-term retention for the game. (M. Herrick, 2017.) An interesting thing to notice is that Herrick said this in 2017. In 2022 almost all the big casual mobile free-to-play titles run tournaments all the time. Supercell's games run multiple tournaments on top of each other.

The nature of live events can be many things, but PlayFab knows to tell a good event is always limited-time natured. Usually, a week or less. They have a great progression and reward loop, engaging theme and content, surprise, and predictability, which means that the game has surprising events in a consistent schedule. Good events also have good value limited-time offers, has community participation and a goal. It is important to notice players that an event is going on. (Azure PlayFab Guide to LiveOps 2019, 37-38.)

2.5 Monetization

Product management, as the name suggests, is very linked to monetization. Originally games like Space War from the year 1962 were free-to-play, but soon the games were put on floppy disks

and sold as premium games (Fields T. 2014, 21). In premium games the monetization is very simple and clear – Make a product and sell it. This was the main model for games for the longest time, so games did not really have to think about product management. Free-to-play games on the other hand rely so heavily on data in the modern era, that it was only natural that people came from other expertise areas to read the data and monetize the games better.

Often games copy what have been done in the past because it is a lot safer, but there are multiple ways to monetize free-to-play games. More ways are invented all the time because there are so many games, and few people always decide to take the risk of trying something new. Some of these new mechanics are adopted to most free-to-play games, and some inspire new genres inside of free-to-play. For example, casual games are usually built around similar monetization model. The whole core loop of the game is monetizable.

There are two major ways to make revenue with free-to-play mobile games. Microtransactions and advertisements. Some games, such as the whole hyper-casual genre are solely based on advertisements. This thesis will focus on microtransactions because they are more product orientated.

Satire show South Park elegantly tells how they think free-to-play monetization works in their episode “Freemium isn’t free”. Freemium is often considered a synonym to free-to-play, but it is debatable. It can also be considered to be just one free-to-play model. In the episode the Canadian devil says that a game needs to have an interesting core loop, or as they put it, simple core loop, so the players want to come back. Then the game makes the player feel good about themselves by complimenting them. After that the players need to be trained to use the game’s fake currency that they get for free. The game then stops giving it for free and offers the players a way to trade real currency into fake currency, so they will forget the value of money and forget that are spending real currency. Then because the game’s core loop is essentially about waiting for different things, the players need to be able to pay for not waiting. According to this South Park episode’s writer and director Trey Parker, this is the key to success. (Parker T. 2014.) Later he also goes into detail about how the game needs to find the addictive personalities through data and make them spend all their money. Three years later South Park released their own free-to-play mobile game.

2.5.1 Mictrotransactions

The economics of everybody being able to release their game into a flooded marketplace has put the prices down to free. It is impossible to make revenue with free games, so a model that was first discovered in Asia is rising to be the dominating force on mobile, microtransactions. (Fields T. 2014, 24.) Fields was still only contemplating in his book in 2014 that microtransactions will become the biggest monetization model for mobile and tablets in the world. By now it has clearly happened, and he was right.

The idea behind microtransactions is that a player is offered a free game, but inside the game there are different things to buy for a relatively low price. Hence the name microtransaction. Touro University compiled a list of four microtransaction types. In-Game currencies, random chance purchases, which means lootboxes, in-game items, and expiration, which means season pass (Duverge G. 2016). Playfab goes a bit deeper and has a list of seven microtransaction types, although Touro University's in-game currencies is a nice touch that Playfab does not talk about. In-game currencies are used to hide the true value of what players are buying (Duverge G. 2016). Playfab's list goes as follows.

Cosmetics are very popular form of microtransactions, more so on PC than mobile. People want to look different and are often willing to pay to customize a character in a game they like. It is more profitable in a multiplayer, so others can see the customization. It is very important to offer the players value for their money and be ready to create a lot of assets. (Azure PlayFab Guide to LiveOps 2019, 45.)

Account Upgrades are things like in-game storage or new character slots. Western players prefer permanent upgrades over cheaper, temporary ones. (Azure PlayFab Guide to LiveOps 2019, 46.)

Consumables are temporary or repeatable effects for the player, such as temporary experience or currency boosters (Azure PlayFab Guide to LiveOps 2019, 46). They can also be actual consumables, like health or mana potions, like in Empires & Puzzles mobile game.

VIP Programs, more commonly known as season pass, is an idea that free-to-play games should not only focus on the few people who are willing to pay endless amounts of money for different services. (Azure PlayFab Guide to LiveOps 2019, 46.) Season pass is a subscription model with a lot of names, and it is very tempting for every engaged player to acquire. The players get cool things for playing the game. In a lot of games, the season pass gives the players cool things for

playing even without paying, but they get more if they pay. If the player does not buy it, the game reminds the player of what they are missing out, which seems to be effective. It is extremely used form of monetization now because it lowers the pressure of success, because so many people subscribe to the season pass model.

Content Access is a common tactic for premium games in the form of DLCs, which means downloadable content (Azure PlayFab Guide to LiveOps 2019, 46). Nowadays it is common in free-to-play games on PC and console as well. An example of a mobile game that does this is Runescape, that sells "membership" with a lot of new content. Runescape was converted from PC to mobile.

Random Boxes, more commonly known as loot boxes, is based on giving the players an option to buy an item for themselves, that they do not know exactly what it is. Lootboxes can offer great value, but they are hard to design well. If a player does not like how the mechanic works in a game, they are unlikely to buy another one. It is important to know the value of the boxes and make it clear to the players as well. Some people do not like lootboxes, and it can be wise to give them other options. (Azure PlayFab Guide to LiveOps 2019, 46.) Fortnite can be thought of as a big lootbox simulator. Every house is a new place full of loot that the player can acquire. It could be one reason for its great appeal. People like to gamble.

Zynga's lead product manager agrees and says that the player needs to get a set amount of gratification for any purchase, because they can usually never convert the in-game value back to real money (Somayanda P. 2019).

2.5.2 Conversion

One cannot really talk about monetization without talking about conversion. Conversion in games mean to convert players to paying users. It does not only increase revenue, but it also increases retention, because players who pay money are more likely to invest time into the game as well (Azure PlayFab Guide to LiveOps 2019, 48).

Good conversion game design is based on offering players something that they want, and more importantly can buy. For example, different countries have different prices for items. A player in India buys items inside a mobile game usually for cheaper than their American counterpart (Judge A. 2015). Judge's article talks about how this trend or offering different countries different prices

is only beginning to happen in 2015, but in 2022 it is very common. Part of good design is also to make the purchase as easy as possible. (Azure PlayFab Guide to LiveOps 2019, 48).

Price increase can happen inside the same country. Then the task is to spot the ones who are willing to pay for more. This can be seen in many games after buying something. It is common to start getting pricier offers after the initial purchase, and even more so after the second. Even without talking about the first-time purchaser offer that many games offer for very cheap.

It was very common in the past to start monetizing the game immediately. The trend has shifted, because players do not usually spend money while they are learning the game, so the game might as well focus on retention at first (Telfer A. 2017). A game can take a while now to even start offering items for purchase. Also, a game can take a long time to fully show what it has to offer.

Barrier to entry means that there are things that stand in the way of the first purchase. First, the purchase should be as simple as possible. Every step, such as entering payment information or creating an account can funnel players even as much as 50% per step. Also, as the table below shows, the barrier to entry does not necessarily mean that the price is too high. In this example the game is an MMO, massively multiplayer online game. Offering a low-priced bundle does not mean that there are more buyers, because in this case only highly motivated players are buying anything. As the game leaves the $1 and $5 dollar bundles out of the game, the players start buying the more expensive ones. (Azure PlayFab Guide to LiveOps 2019, 48.)

Table 1. Azure PlayFab Guide to LiveOps (Azure 2019, 48)

Offers	First Time Purchasers Week 1	First Time Purchasers Week 2	First Time Purchasers Week 3
$1	2,435	Not Offered	Not Offered
$5	1,122	2,145	Not Offered
$10	1,034	1,207	2,032
$20	884	903	1,576
$35	704	704	906
$50	429	429	429
$100	166	166	166
Total Purchasers:	**6,774**	**5,554**	**5,109**
Total Revenue:	**$98,755**	**$103,545**	**$121,600**

A good trick to get the players to make the first purchase is to offer a permanent upgrade to the game, such as “double coin” boost in endless runner style games, where the players keep receiving double coins forever (Telfer A. 2017).

All these examples are excellent reminders of the product managers analysing, hypothesising, prioritising, and executing cycle. These results come from somebody hypothesising them at first and then executing a working plan.

3 Case Study of Irrelevant Games

A company Irrelevant Games Ltd was formed in January 2020 by three Kajaani University of Applied Sciences students. Writer of this thesis was the CEO of that company, Jussi Prokkola. There is good documentation of everything that was done. This chapter investigates what was done in Irrelevant Games' two games, and how they could have been improved by what is learned in this thesis. Both of Irrelevant Games' games are included, because both games did something better than the other, and therefore gives more comprehensive comparison.

3.1 Introduction

Irrelevant Games made Roblox games because it was a rapidly growing platform. As a team of students, it was thought that Roblox would be a great opportunity. Considering the skill level of students, it was more likely place to find success compared to other platforms, such as mobile, because the competition was not as developed yet. However, as the Roblox platform is mostly free-to-play, the same product management principles were applied as in mobile.

Irrelevant Games got a grant from the Finnish Government. The first game is called Jumpageddon. After the LiveOps of Jumpageddon was halted, full rights were acquired to a game called Ninjas with Guns, that the team had worked on previously with a bigger team. Two artist interns were hired to help with Ninjas with Guns.

In autumn of 2020, Roblox Corporation was looking for two teams to work directly for them. Irrelevant Games was in the top six applicants for the position but didn't quite get it. After that Irrelevant Games stopped making games in October 2020.

There is a lot of data from both games through computer tracked data, and a comprehensive market analysis and a journal made by hand. Every bit of data here is from the early 2020, or before. This thesis will not go into detail about what Roblox has become.

3.1.1 To Roblox

Roblox works on three different platforms. The player needs to have Roblox installed on either computer, mobile, or Xbox. It had 100 million active players at the time and was founded in 2005.

Roblox's monetization is based on players who buy premium currency called robux inside Roblox, and then individual game developers, such as Irrelevant Games, can make games inside the platform, where the players can spend the premium currency. It is up to the developers if they can monetize their games and they will get a share accordingly. Roblox provides the platform and takes a big cut of all revenue, which was counted to be approximately around 75-85%, but there was no official data about it. Usually, platforms famously take around 30% cut or less.

It is a saying in gaming industry that "Making a good game is like buying a lottery ticket to win a price", because the platforms are so full of games and success is not guaranteed even with a good game. Roblox still offered a good chance for good games to get visibility and success in 2020.

3.1.2 To Jumpageddon

Irrelevant Games' first game is Jumpageddon. The idea of Jumpageddon is to pass a random generated obstacle course during a time limit. If the player can do it, they will get in-game currency. When the time is up, the map starts over, and a new random generated obstacle course appears.

Jumpageddon was supposed to be released quickly and with lean project management start getting feedback from it as soon as possible. It took around 4 months to release Jumpageddon, which was longer than anticipated. After that it had its ups and downs, but eventually Jumpageddon had to be killed after only two months of LiveOps due to its low retention.

3.1.3 To Ninjas with Guns

Irrelevant Games' second game is Ninjas with Guns, in short NwG. It is a first-person arena shooter. The whole team of Irrelevant Games had worked before on Ninjas with Guns with a much bigger team. Full rights to the game were acquired from its previous student team and Irrelevant Games continued it with two new hired artists.

Because Jumpageddon was a game where movement was in a big role, Ninjas with Guns was potentially a good game to work on, because it also relied heavily on movement and the team already had experience of good movement. In Ninjas with Guns the characters could run on walls, like ninjas. They also had two jumps between landings. Basically, the characters moved very fast. It was a problem because players could not hit each other, and the feedback was that it was frustrating.

3.1.4 To the Team

There were initially three members In Irrelevant Games, so it was a very small company, and the roles were shared accordingly. Some roles were outsourced, such as UI, audio design and most of the Jumpageddon's graphics. Advertisements were designed by Irrelevant Games, but the making of them were mostly outsourced as well.

Jussi Prokkola ran the company, did user acquisition, project management, outsourcing, community management, quality assurance, built levels and most importantly for the topic in hand, took care of the game's economy and product management decisions.

The designer, Olli Uikkanen, designed most about the games, but also focused a lot of energy in programming, and little bit of graphics.

The programmer, Janne Sillanpää, did by far most of the programming. He was also responsible of being a data scientist.

3.2 Product Management

It is another thing to read about product management and good LiveOps principles, and another to put them into action. It is said that good LiveOps can save any game and make them profitable. Bad LiveOps on failing games does not save the games but it does give experience. The difference is hard to notice without previous experience. This section goes through how product management was implemented on Irrelevant Games' games, and how it could have been improved.

There was a very vague roadmap of when things needed to be ready, but with such a small team it was hard to implement. If one thing went wrong, at least 33% of the work force needed to work

on that problem. Mostly the team worked with short weeklong sprints. It is arguable what is the best solution for such a small team but having a clear and long roadmap would help any team, even if it is hard to follow.

Mr. Courtemance gave a list of three basic things for associate product manager to do in the What Product Manager Does chapter of this thesis. The following list goes through the list in the same order and compares the list to what was done. The team had an idea of what should be done, but as there was not enough experience, some of the steps were lacking.

Competitive analysis was Irrelevant Games' top priority. Since the team did not have any idea what the competition was, it was taken very seriously. All the top 100 revenue games were played on top of the most popular direct competition. It was learned that getting on a new platform without previous knowledge of it is very hard.

Content Management should be about giving the players new things to do. With such a small team and unfinished games, it was not possible. The games were updated regularly, but it was more about bug fixes and a lot of changes to the game. For example, because the movement was not as good as it could have been, it was updated and tested on all the players. Segmentation was not an option on the platform, which is not good. Segmenting players is very important. New gameplay features were implemented, but nothing new and exciting daily, which was a mistake. The team was not big enough to do everything.

Investigation and Data Analysis was implemented but could have been better. It is hard as it is to tell a story with data, but it is near impossible if there are not enough players. Both games peaked at around 100 simultaneous players, but they usually had less than ten players. On top of this, the team already had so much else to do, that there was not enough time to learn the principles of being a data scientist, or a good product manager. All the basic data, such as retention and engagement in large were tracked, but not that many custom data sets. Some of the players' clicks and preferences were tracked, such as favorite maps, guns, and store behavior. Yet with such a small number of players it was common to get different results one day and different results another day.

One of the biggest mistakes for both games was that there was no clear product strategy, as Mr Somayanda and Mr. Katkoff says games should have, as told in What Product Manager Does chapter of this thesis. There is an argument to focus on building a good game and monetizing after but building an excellent core loop for a game is very hard. As learned, building the game around monetization is in a way easier. In Roblox, as well as mobile, most of the top revenue games are

built around monetization. On mobile they do it very well and subtly. The games' core loops are very good as well. As South Park puts it comedically, if the game is about waiting, people might pay to not wait. So, if there is an inconvenience in the core of the game, a solution can be monetized, and that decision almost magically brings in better retention and engagement.

As both of Irrelevant Games' games were not built around monetization, it was extremely important that the core loops were good. Because the data showed that retention was low for both games, most resources were put on fixing the games and making them better. Monetization never really got to be as good as hoped, but there were some low hanging fruits that could have helped. One mentioned before is that the first player purchase could have been double currency for the player for a month, to convert the players, and improve engagement and retention. Instead Jumpageddon offered a good deal of its currency and Ninjas with Guns did not offer anything.

3.3 LiveOps

Both games were LiveOps games, that were made using lean methodology. After the initial launch the games were updated every week until they were killed. Because the player bases were small in both games, Discord was used a lot for customer service direct feedback.

On top of not having enough players for reliable feedback, the team was small. LiveOps enables an early release, and transition to do build-measure-learn feedback loop. Both games worked well in launch and there were no big bugs, but even the smallest things took a long time to fix, which changed the whole roadmap. Hypothesis here is that with a small team the game needs to be extra ready for LiveOps. Also, the best LiveOps methods should have been implemented before launch, such as what data to track. Segmenting players and machine-learning to segment them better should have been implemented in development phase, if Roblox enabled that.

As Roblox does not offer segmentation possibility, it was impossible to A/B test anything. All had to be implemented as they were. The changes into the game were made using a rough roadmap – an idea what the game should become eventually. Because any sort of testing was practically impossible, it was as impossible to know which changes caused what. Also, because of feedback the goal became to make the game more fun and more friendly to new players, some of the changes were not well received within the established community. D1 retention got eventually better through trial and error, but with the cost of already established player group. In a way, that was a success, but it would have been a lot better with segmentation and A/B testing. Though,

testing the new features could have been done with events. New gameplay features can be tested in limited time events before implementing them into the base game.

Irrelevant Games used GameAnalytics to track data. Including custom data. The problem with custom data is that it often needs to be programmed to be tracked. Because Irrelevant Games did not have a data scientist, it proved to be very hard and time consuming. The custom data that was tracked is not visible in GameAnalytics anymore, but the basic data is.

Because there were usually not over thousand players during a day in either of the games, it was hard to make informed hypothesis on different variables. With so few players it is important to try to track as high-level information as possible, which was done correctly with the games. The main priority was to see what increases retention. Engagement should have been equally important metric. Custom data only told what the players liked, such as guns and maps.

Roblox is a platform for PC, mobile and Xbox. Both games were only made for PC. Making them available for other platforms would have increased players, but would have opened a new set of problems, especially in balancing.

3.3.1 Engagement

The most engaging single thing for a game is its core loop. The problem for Jumpageddon was that it was slightly too unfair and hard. It was an obstacle course, but something about the gameplay was not liked, whilst its similar competitors did great. A hypothesis is that Roblox has very established movement built into the core of Roblox. The movement was remade completely for Jumpageddon. Every Roblox player who thought that they are very good in the platform's movement mechanics failed in Jumpageddon. Basically, too much was innovated. For game's safety roughly 70% should be borrowed from other already established games, 20% innovated and 10% being something completely new.

Ninjas with Guns had a community, but it did not have an active community. Neither of the games welcomed players back day after day for something engaging and preferably new. Ninjas with Guns had a lot of potential, but essentially its retention was too low to keep making content and events.

Roblox did not offer a good chance to remind the players to come back to play the game. There is a bell icon in the game, like in Youtube. The same problem persists inside Roblox that many

youtubers tell their viewers during their videos. People do not press the bell icon unless reminded. It is harder to do in a game than in a video. In Roblox every time a notification was sent to the players who had pressed the bell some of them unpressed the icon.

The best conversation item is a purchase that engages the player. For example, Heroes Charge mobile game offered premium currency over a month for a small amount of money. Jumpageddon offered a first player purchase package, but it was not engaging. It offered direct currency. Ninjas with Guns did not offer anything. This would have been a very good engagement tool to add into both games, which both had their own currency.

3.3.2 Retention

As retention is one of the most important metrics to predict how successful a game is and becomes, both of Irrelevant Games' games gave a reason to doubt their potential. It is impossible to tell what retention is good in Roblox, because there was no public data about that. Based on mobile standards, it is safe to assume these were not good numbers.

Playfab says in its guide to LiveOps that content is the most common strategy to improve retention. As the games were not quite ready for release compared to the team size, content was the most used strategy for Ninjas with Guns and Jumpageddon. More content was released every week for couple of months for each game. Even some gameplay elements for Ninjas with Guns, such as melee and grenades. Events should have been higher priority, but Ninjas with Guns only had basic double coin events. Jumpageddon had double coin events, but also one unique event. A very popular Brazilian youtuber played Jumpageddon, so a Brazilian event was made, which will be discussed in detail later.

A well-known golden standard for retention is 40-20-10, which means that D1 retention should be 40%, D7 20% and D30 10%. As the picture shows, Jumpageddon did not have those numbers. It is visible in the screenshot below by the huge drops after marketing or influencers. D1 retention was 13%, which is close to unredeemable on mobile. D7 was less than 3%. The first big spike is from a Brazilian youtuber Julia_minegirl, who played the game. There was no marketing at that time, but an event was made for Brazilians. It slowed down the fall, but it did not stop it. The next spike is from marketing, the next small bump is from a smaller youtuber and the last one is from marketing again. It was decided that marketing is not worth it, because retention could not be

fixed. As the game was inherently single player, it did not matter that much for players if there were other players inside the game or not.

Figure 9. Jumpageddon, Daily Active Users 2020. The data is available from the author himself.

Roblox does not reveal its golden standard, which could very well be different from mobile, but 13% was too few, because the players fell so quickly, and because conversion was not very high.

Investment is the second strategy Playfab discusses about, as told in Retention chapter of this thesis, under LiveOps. The players should have the possibility to invest time and resources into progression. As the games were built gameplay first, this was not considered in the development phase of the game. World's top revenue games were all built gameplay first because gameplay is by far the most important thing. If the game gets players and performs well, everything is possible. Everything else can be added later, such as more possibilities to invest time and resources into progression. Jumpageddon did not have any investment possibilities. Ninjas with Guns had levels, and the levels determined who the players played against. More experienced played play against other experienced players. Sadly, that aspect of the game was not very monetizable and did not bring the needed progression. A way to monetize it could be to sell experience boosts. Maybe leaderboards for levels could have improved the desire to buy those. Jumpageddon focused a lot on leaderboards, but it only seemed to entice hacking, which then needed to be prevented through time consuming anti-cheats.

Social possibilities offer some of the highest retention returns for a game. Ninjas with Guns is an all versus all first-person arena shooter. It was top priority to add teams into the game, so the players do not need to play alone, which happened eventually in one of the better updates the

game saw. It visibly grew retention. Other aspect of social relationships are the social clubs, such as guilds. Clan-system for Ninjas with Guns was a high priority for a long time but could never be implemented because of the long list of things that needed to be done before. Clans would have probably increased retention by few percentages as well.

Figure 10. Ninjas with Guns, Daily Active Users 2020. The data is available from the author himself.

Ninjas with Guns had even worse retention. Less than 10% at first, and over 12% later. Its D7 was the same as Jumpageddon's, which indicates that more NwG players that came back D1 also came back D7. Over hundred people joined Discord, and those people can be considered as community, which would come back for every update. Most players did not use Discord though, and there was not a real option to contact them if they did not join. Irrelevant Games group inside Roblox has over 600 people. Ninjas with Guns had a lot better conversion rate, and revenue. The biggest problem for NwG was that it was a very fast paced arena shooter, and that genre is historically very unpopular. Casual players could not get the hang of it quickly, and even slightly better players could dominate the arena.

Which brings the topic to its conclusion with Playfab's last point, core loop. Jumpageddon did not have good enough core loop to keep the players engaged, and Ninjas with Guns had a solid core loop, but not enough players. And as the game was inherently multiplayer, the servers always needed to have players, or it would die very quickly. This last point will be discussed more in user acquisition.

3.3.3 Monetization

Monetization is a hard thing to get right. A thing that affected both games was that there were never many players. Maximum number of players for both games was around 100. During those times the conversion rate got a lot higher, which indicates that players are more willing to pay inside a free-to-play game that is popular, than in a game that is not popular.

Jumpageddon's monetization model was not well thought through and it showed. It relied heavily on consumables, which are not popular in the west. It sold little bit of cosmetics, but mostly short boosts to the gameplay, such as higher jumps and jet pack. If the boosts had been longer, they could have increased engagement when bought. Same goes to the first player purchase item, like talked in the engagement section. Still Jumpageddon's first player purchase offer was its most bought item, which shows how good conversion method that is. Jumpageddon's fans were mostly from Brazil, which was in turmoil at the time because of Covid, which could have had an impact in the revenue.

Ninjas with Guns on the other hand loaned its model from other games, which is usually the right thing to do. Because Ninjas with Guns was built around gameplay and not monetization, the monetization was added on top of it as skin in the form of lootboxes. Even though the conversion rate was not comparable to industry standards on mobile, the game did alright with its monetization.

Robux is the premium currency in Roblox. Every games' monetization inside Roblox works inherently with robux. Both games had their own soft currency which could be earned inside the game. Jumpageddon only sold one thing with robux, and everything with its own currency. As robux was already a premium currency, Ninjas with Guns started treating robux as its own premium currency. Ninjas with Guns essentially had two different currencies, which is the best model and most used in mobile. Some of the things can be earned, some better things can only be bought with premium currency.

Ninjas with Guns had two artists working on it, unlike Jumpageddon. It was extremely hard in Jumpageddon to give the players enough things to buy. Every update in Ninjas with Guns had some new skins for the players and their guns. This is one of the key elements to LiveOps and could be done because of the artist interns.

Balancing the one soft currency the game had was a big task. Most people did not like how slowly they got it, but it is hard to determine how reliable that feedback is. People usually like to get things fast. Data would have helped, but there were not enough players.

3.3.4 The Event

Both games had small double coin events, which did not help much. These kinds of small events would only show in graphs if there were a lot of players. Mr. Herrick said that double experience event is one of the most common ones, so a double coin event is okay too. Ninjas with Guns had levels, so experience could have worked as a small event as well.

Herrick's mentioned spending events and limited-time content events should have gotten more attention. Whenever Ninjas with Guns released a new set of skins, they could have been more visible and buyable with premium currency. Or they could have been new sets in new kinds of lootboxes. Tournaments were in the Ninjas with Guns' roadmap, but never got live.

Jumpageddon had a massive boost to its players because of a Brazilian youtuber called Julia_minegirl played the game. She had close to 6 million subscribers at the time. It was very early days for Jumpageddon and it kind of blurred the big picture. It seemed like a jackpot, which lead to wrong decisions. It was decided that the game does not need marketing immediately, because the game was doing so good for couple of days. Marketing could have gotten Jumpageddon to Roblox's front page, which in turn could have helped the game a lot more. Instead, an event was made in honor of Julia_minegirl, and especially Brazilians in general.

Most of the players were from Brazil at the time because of the video, so Brazil event was made after three days of hard work. After which Irrelevant Games decided to never crunch again. The event had three quests that required the players to play the game to complete them. Every quest was harder than the last, and after every quest they players got a Brazilian themed reward.

It was close to Herrick's last event architype, a tournament event, but it was closer to Playfab's general "good event". It was limited-time natured, it had a good progression and reward loop, engaging theme and content, surprise, and predictability. It only lacked good value limited-time offers, which is the monetization part. People were socializing in the game and pushed towards a similar goal. It performed okay and can be seen in graphs as something that slowed the player churn. In a way, this was one of Irrelevant Games' biggest achievements, because events are in a

such a key role to good LiveOps. If the games had performed better, the events should have been the main priority.

The Brazil event led to another Brazilian youtuber playing the game with few hundred thousand subscribers. It did not lead to that many new players though. Couple of streamers played Ninjas with Guns a few times, but that did not have an impact either.

3.3.5 Player Acquisition

In Roblox player acquisition is different from anywhere else. The bulk of the work happens through in-app advertisement, that is visible for the next 24 hours for random people. The next picture shows what kind of data was gathered. Mostly advertisements' performance was measured. Next to the day of the week the graph tells the time when the advertisement was placed. So, the players are from that time specifically. And because the advertisements are sometimes placed over 24 hours from each other, it is already possible to see the decline in players. As seen from 19th to 21st.

Advertisement type means that money was spent on specifically a skyscraper advertisement type, because it was realized that they produced the best outcome. Impressions mean how many people clicked the advertisement. CTR means how many clicks the advertisement attracted compared to how many saw it. Beginning stats compares players, likes etc. to the next cell of stats, which was then measured at the end of the 24 hours when the advertisement ended. SUM tells the difference.

From the next table alone, it is possible to see a lot about how the game performed. With such a small game maybe the most important statistic is the "Discord" number, which is almost the same as dedicated players. The players really liked the game and helped to develop it further. CTR was dropping every day, because eventually only the best performing advertisement was used, which was not a clickbait advertisement. Clickbait advertisements attracted more players but dropped the game's like/dislike ratio a lot, because people do not like to be tricked into playing extremely hard games, which the games essentially are.

Table 2. Jussi Prokkola 2020. The market research is available from the author himself.

	Ninjas with Guns				
	Day	18.8.2020	19.8.2020	21.8.2020	22.8.2020
	Week day	Tue 22:22PM	Wed 21.51PM	Fri 02.00AM	Sat 1.59AM
	Bought robux				
Ad type	Sponsored (robux)				
	Banner (robux)				
	Scyscraper (robux)	2000	2000	2000	2000
	Sum	2000,00	2000,00	2000,00	2000,00
Ad perf.	Promised impressions		181673	160712	159213
	Impressions	197482	156122	120539	126975
	Compared to promised		-25551	-40173	-32238
	CTR %	1,13 %	1.11%	0.94%	1.07%
Beginning stats	Playing	16	16	9	5
	Likes	474	496	518	541
	Dislikes	191	194	204	213
	Favorites	1771	1921	2050	2148
	Visits	21700	23700	25500	27100
	Group	259	266	273	282
	Discord	40	42	59	62
Stats	Playing	12	9	8	0
	Likes	496	518	541	561
	Dislikes	194	204	213	224
	Favorites	1921	2050	2148	2230
	Visits	23700	25500	27100	28500
	Group	266	273	282	286
	Discord	42	45	62	63
SUM	Playing	-4	-7	-1	-5
	Likes	22	22	23	20
	Dislikes	3	10	9	11
	Like/Dislike ratio	88,00 %	68,75 %	71,88 %	64,52 %
	Favorites	150	129	98	82
	Visits	2000	1800	1600	1400
	Group	7	7	9	4
	Discord	2	3	3	1
	Event				
	Event description				
	Comments			Changed the	

Roblox offers a button for people to invite their friends into any game they are currently playing, but neither of the games offered a reward for that. A reward would have been a good for more players. On top of that, it is more likely that the players would have come back if their friends were playing as well.

4 Conclusion

Considering the number of sources available for product management, this thesis succeeded in answering comprehensively to the question of what product management is in LiveOps in gaming industry.

Product management has a long tradition in products. After introducing product management, the thesis goes to product management in gaming industry, which leads to product managers. As the thesis focuses on LiveOps, it is introduced right after. After the basics of LiveOps, monetization is talked right after data. With all this information, Irrelevant Games' games are compared to the established and effective principles that were studied earlier. A lot could have been done better, but some things were done correctly as well.

It is certain that product management in gaming is different from other forms of product management, yet product managers can transfer between industries. Some industries, like gaming industry, require unique skills compared to others. Product managers are responsible of making the product profitable. Sometimes, as profitable as possible. LiveOps is the best place for product managers, because that is the time when their job is the hardest to integrate to other professions.

The problem of product management in gaming industry is that the information is not in one place. There are no books or studies about the specific subject yet, if less specific technology and application books are not considered. This thesis can help in guiding to understand the gaming industry product management better. The gaming industry is often a lot different from other areas of expertise, but apart from the basic development process, a lot of gaming industry is covered here. A lot of information that can not be found elsewhere from one place.

Product managers giving different speeches in YouTube work as a big reference point for the information here. A good thing to remember about that is that people can misspeak even by accident. As the author of this thesis cannot always question his seniors, some information could not be correct. Assessment can be made that it does not affect the reliability of information, because the thesis often reminds that things are done differently depending on the company. If some information is wrong, the damage of misinformation is confined into the specific company that the person is talking about.

This thesis hopefully works as a guide for more in-depth guides to product management and LiveOps. A whole thesis could be written about the data, or even just product managers' role in data.

A lot was covered in this thesis, which should be considered. It gives a good overarching understanding of product management, but because of the amount of information in it, it barely even grazes the surface on some areas. Hopefully a product manager who wants to transfer from other industry to gaming industry can especially find the information useful. The thesis does not give the readiness to understand everything, because a lot of the things can only be learned through experience.

5 Sources

Azure. (2019). The Azure PlayFab guide to LiveOps. Washington, United States: Microsoft. Author of this thesis has a different version of this study on him, but a shorter yet very similar version is available 26th of May at https://playfabdocs.blob.core.windows.net/live-ops-guide/PlayFab_LiveOps_Handbook.pdf

Mayer-Schonberger V, Cukier K. (2013). Big Data: A Revolution That Will Transform How We Live, Work, and Think. New York, United States: Harper Business.

El-Nasr M, Drachen A, Canossa A. (2013). Game Analytics: Maximizing the Value of Player Data. New York, United States: Springer

Fields T. (2014) Mobile Social Game Design: Monetization Methods and Mechanics. Florida, United States: CRC Press

Anon J., Villaumbrosia C. (2017). The Product Book: How to Become a Great Product Manager, 2nd edition. San Fransisco, United States: Product School.

Banfield R., Eriksson M., Walkingshaw N. (2017). Product Leadership: How Top Product Managers Launch Awesome Product and Build Successful Teams, 1st edition. California, United States: O'Reilly Media.

Adjust. (N.d.) User Acquisition | Meaning. Available 5th of May 2022 at https://www.adjust.com/glossary/user-acquisition/

Calderon J. (2020) Acquisition isn't everything. Available 5th of May 2022 https://www.reloadly.com/blog/mobile-game-retention/

Herrick, M. (2017). How to Monetize Games by Gaming Product Manager. Upskill Product School. [Video. Uploaded 15th of September 2017.] Available 20th of April 2022 at https://www.youtube.com/watch?v=wRGXtgClqhs

Agarwal, V. (2021). PM and Analytics in the Gaming Industry. Upskill Product School. [Video. Uploaded 21st of October 2021] Available 20th of April 2022 at https://www.youtube.com/watch?v=hUIGRhewIg0

GameAnalytics. (N.d.). A/B Testing – Overview. Available 29th of April 2022 at https://gameanalytics.com/docs/s/article/A-B-Testing-Overview

Navot Y. (N.d.). Segmented A/B tests: Avoiding average experiences. Dynamic Yield. Available 26th of April 2022 at https://www.dynamicyield.com/lesson/ab-testing-without-segmentation/

Courtemanche O. (2018). What Is Gaming Product Management Like. Upskill Product School. [Video. Uploaded 9th of March 2018] Available 26th of April 2022 at https://www.youtube.com/watch?v=JvzfYyArnnQ&t=911s

Katkoff M. (2018). What Successful Product Managers Do Differently in Free-to-Play Games. GDC. [Video. Uploaded 4th of February 2021.] Available 26th of April 2022 at https://www.youtube.com/watch?v=yrG76QIH7Gg&t=320s

Perez A. (2018). User segmentation approaches for games. Medium. Available 17th of May 2022 at https://medium.com/googleplaydev/user-segmentation-approaches-for-games-4457aa57d56e

Duverge G. (2016). Insert More Coins: The Psychology Behind Microtransactions. Touro University. Available 26th of April 2022 at https://www.tuw.edu/psychology/psychology-behind-microtransactions/

Somayanda P. (2019). Product Management In Games by Zynga Lead Product Manager. Upskill Product School. [Video. Uploaded 4th of March 2019.] Available 17th of May 2022 at https://www.youtube.com/watch?v=FO6wkBiAL1c

Judge A. (2015). What's ideal price point for mobile games in India? Pocket Gamer. Available 26th of April 2022 at https://www.pocketgamer.biz/asia/mobile-mavens/61622/indian-mavens-whats-indias-ideal-price-point/

Telfer A. (2017). Free to Play Monetization: Making the First Purchase. Available 26th of April 2022 at https://mobilefreetoplay.com/free-to-play-monetization-making-the-first-purchase/

Broda E. (2019). Our Personal Data is a Lot More Valuable than We can Imagine. Available 26th of April 2022 at https://medium.com/swlh/our-personal-data-is-a-lot-more-valuable-than-we-can-imagine-60af3b920da0

Sobolev J. (N.d.). What Is Game Retention? How to Boost it For Your Game?. Available 26th of April 2022 at https://gamingshift.com/game-retention/

Sisense Team. (N.d.). Game Revenue Metrics — What Matters and When. Available 26th of April 2022 at https://libguides.kamk.fi/c.php?g=657740&p=4630231

Chapple C. (2022). Top Grossing Mobile Games Worldwide for December 2021. Available 26th of April 2022 at https://sensortower.com/blog/top-mobile-games-by-worldwide-revenue-december-2021

Cox C. (2020). Understanding LiveOps, Doing LiveOps. Available 26th of April 2022 at https://www.gamedeveloper.com/design/understanding-liveops-doing-liveops

Eriksson M. (2015). The History and Evolution of Product Management. Mind the Product. Available 26th of April 2022 at https://www.mindtheproduct.com/history-evolution-product-management/

Productfolio. (N.d.). History of Product Management. Productfolio. Available 26th of April 2022 at https://productfolio.com/history-of-product-management/

Yankulov K. (2020). Product owner vs. product manager: Who runs the show? Available 26th of April 2022 at https://www.productboard.com/blog/product-owner-vs-product-manager/

Zaguirre I. (2017). Product Management in the mobile gaming industry. Available 26th of April 2022 at https://www.youtube.com/watch?v=y13qDnINgDQ

Winchester H. (2014). From Floppy disks to freeware: the history of F2P gaming. Available 26th of April 2022 at https://www.techradar.com/news/gaming/from-floppy-disks-to-freeware-the-history-of-f2p-gaming-1248467

Rosemary N. (2020). Mobile Gaming Industry Statistics. Available 26th of April 2022 at https://askwonder.com/research/mobile-gaming-industry-statistics-bzwrnpkde

Wijman T. (2021). The Games Market and Beyond in 2021: The Year in Numbers. Available 26th of April 2022 at https://newzoo.com/insights/articles/the-games-market-in-2021-the-year-in-numbers-esports-cloud-gaming/

Parker T. (2014). Freemium Isn't Free. [Video. Uploaded November 2014] Available 26th of April 2022 at https://www.southparkstudios.nu/episodes/jy5lbq/south-park-freemium-isn-t-free-season-18-ep-6

”