



Bilgi University Department of Digital Game Design

ANALYSIS REPORT

GAME 401

Kaan Eskikalıcı

Derin Su Kutaydın

Kağan Samestı

Ethem Anıl Göl

Bora Kılıç

Project Advisor: Baykar Demir

General Summary

This report gives a comprehensive analysis that guided our decision-making process in developing our game concept. It includes the key facts and current trends of the game industry that shaped our creative direction, as well as our vision for positioning the project within the market. Through research, we identified our potential target audience and analyzed their gaming preferences to ensure our concept aligns with their gaming style.

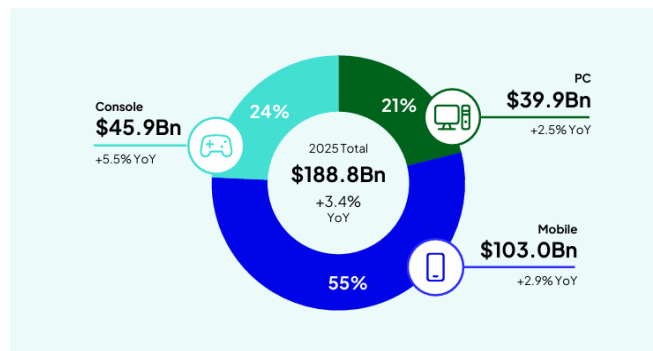
Additionally, the report outlines the technical framework of our development process. It explains the game engines and design programs we plan to use, along with our current level of proficiency and creative potential with these tools. Insights gained from interviews with experienced professionals in the gaming sector are also summarized, offering valuable advice on production challenges, teamwork, and market adaptation.

Finally, the report provides a step-by-step draft of our game-making journey — from ideation and prototyping to design, testing, and final implementation. By combining analytical research with creative exploration, this document serves as both a guide and a reflection of our ambition to develop an engaging and high-quality game that resonates with players and contributes meaningfully to the modern gaming landscape.

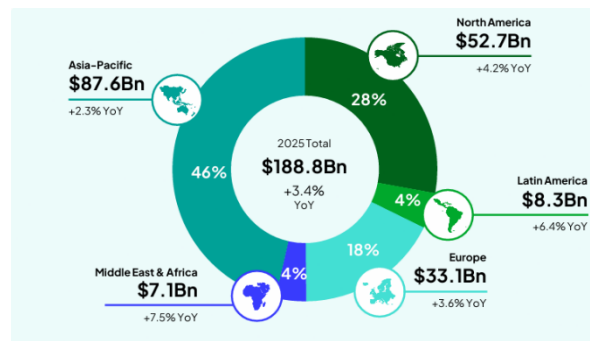
Game Theme and Topic Analysis

Our game is based on Turkish Mythology and Shamanism. The genre we want to develop is a Turn-Based RPG with Roguelite elements. We were drawn to this subject because, although it hasn't been seen much in games before, it actually has a very rich structure for games. We believe it is an original and interesting subject.

Today, the gaming industry is roughly split 50% mobile and 50% PC/console. Due to the nature of our genre, we are creating a game with deeper gameplay that is not superficial. For this reason, we aim to bring the game to market on PC and console platforms.



Geographically, the Asia-Pacific region has the largest market share worldwide. China, Japan, and Korea dominate this share. The fact that RPGs are the most preferred genre in these countries, and that RPGs rank high in Europe and North America, is one of the main reasons driving us towards this genre.



Turn-based games are an ideal choice for portable devices such as the Nintendo Switch. Especially with the increasing popularity of portable devices like the Steam Deck and ROG Ally, we definitely plan to develop our game to be compatible with these devices.

Analysis Report

For example, games like Darkest Dungeon, Slay the Spire, and Into the Breach are games with high profit margins on PC and consoles (especially on Nintendo Switch).

As a result, we can classify our game as Turn-Based RPG + Roguelite. Turkish mythology, with its rich characters and settings, presents an excellent opportunity for this genre. We believe that the setting will enrich the game's genre.

Game Market Analysis

Game Market Analysis Interview

Kadir Demirden

General Manager at Motion Blur

What do you think is the secret to the success of an indie game?

“These are such difficult questions... They can't be answered off the cuff, but I can say this: I'll give a general answer. No one knows. No one in the world knows. Why? I'll tell you why, and you'll understand. The reason is: Theories can be created, kids. There are tons of theories out there about what will happen in the game, what will happen, and so on. Think about it: The person who created these theories doesn't know how to do it either, right? So think about it this way: if I were Hideo Kojima, or Cory Barlog, or whoever, if I knew this, if I knew how to make a game successful, every game I made would be a hit and I'd be the best developer in the world, right? In fact, think about it this way: EA pays the world's biggest developers, and you hear that the games are flopping. So this is such a philosophical thing, such a complex thing that the answer to this question cannot be given in a single way; there are thousands of parameters here. Because what does it mean? A game is made up of thousands of parameters, we agree on that. We agree. There may be fewer parameters, but it captures something specific. Now, I have some theories about this in my head, just like everyone else, and there are areas where I think differently, just like everyone else, but these may not be consistent. For example, what I did in my project may be consistent in my project but may not fit your project. Why? Because projects are very diverse; there are parameters on top of parameters, and then another parameter that changes that. Now, if we need to answer this question on such a curve, we need to say this: Follow general conventions, okay?

Why follow general conventions? To find new mechanisms and test them with people to see if they like them. You need to be someone who can understand this. So, I'm testing a mechanism with you, and based on your feedback, I need to understand what you're stuck on there. Are you stuck on what I want, or are you stuck on something else? I need to understand that. The game being responsive, this responsiveness triggering a certain pleasure center in my brain and releasing happiness hormones from there—that's the scientific side of it. Think about it: pressing that button should give me pleasure. visual camera angle, I explained these a while back on YouTube, there are many parameters, you need to support all of them with your graphics, particles, whatever, and after you've done all that, you need to take it to a general test and see the happiness in the user's eyes during the general test. If you can do these things, the game will be successful. If these things don't happen, it usually means the game will be unsuccessful. Whether you're EA or an indie company, indie companies need to experiment to be successful. They'll experiment, analyze people: what do the players like? They'll ask, "Can we make them happy this way?" That's the general framework."

Which platforms are more advantageous for achieving success in the Turkish market?

"Let's rephrase this question: To be successful globally. Because success in Turkey means nothing; nothing happens in a sector where we play our own game. We will aim for global success. We will always think about how we can be successful globally, we will do business globally, and if we want to do business globally, we will make ourselves technically perfect. We will be perfect, okay? We will do the best job we can technically. Whether people like it or understand it doesn't matter to us anymore. We will know that we are doing the best we can, and that's why we need to think globally. There is no such thing as being successful in Turkey; we need to be successful in the world, okay? That's how we're changing it. We will be successful globally because if we want to make money and get somewhere, we have to be successful globally. We have no other choice."

What are your personal expectations from an indie game?

"Well, right now I can't play many games, and I can't follow the community very closely either because I'm under so much pressure, but what are my expectations from a game? My expectations from a game are this: When I sit down to play a game, I want it to flow. What do I mean by this fluidity, this responsiveness? For example, let me tell you about something that happened to me at a convention one day—I think that will explain what I mean much better. They described a mechanic to me and asked, "Would you play this game?" I said,

Analysis Report

“Sure, I'll play it,” and they ran it. I said, “I don't understand this, what do I need to do?” They said, “You need to do this, you need to do that.” I press the button, I press the button, something happens, a particle appears, sometimes the particle appears late, sometimes it appears on time, sometimes it works, sometimes it doesn't. I said, “Is this the mechanics of the game? I don't understand.” They said, “No, it works when you press it.” I said, “Why isn't it working, man?” They said, “Sometimes it glitches.” Well, that can't be right. So, multiply this example in your head wherever you want. When you multiply this example, understand this, bro: if the game promises me something when I press the button, it has to give it to me, bro. It has to work perfectly.”

What are the most critical areas of expertise for a small team's success? If you were in charge, which area would you choose to specialize in?

“What a small team needs to do is quickly resolve conflicts when they arise, focus on the task at hand, and support each other: psychological reasons. Technical reasons won't matter anyway. If you're making a small game, the technical difficulty is a ten out of ten. You're smart kids, you'll solve it. The problem you can't solve here is ego, arguments over knowledge, saying “it shouldn't be done this way, it should be done that way.” You can't solve those. So what will you do? There will be a consensus like this: “Look, even if we argue, this guy is acting ego-driven right now, okay, I'll handle it.” Right now, we're going to do the job, we're going to do it perfectly, and there will be a fight in everyone's mind. That's what will happen, bro. This guy is saying this right now, but he also wants this job to be good. Maybe he's favoring himself a little, but that's okay, it doesn't matter. He wants the job to be good. Let's motivate him with another motivation and get him into the job, because there's an incredible ego in Turkey, and people are crushed under that ego. I think the best thing is not to be egotistical, to concentrate on the job. Look, I always start with the philosophy of the job, because I think the most important thing is the philosophy, the philosophy of the job. What does that mean? You'll learn the technique anyway, give it five years and you'll learn whatever you want to do. But you can't learn this: how will I work with a team? how to make this job successful, how to solve this problem right now, this is the wrong method, how to convince these people of the right method, and how to make sure they don't back out once they're convinced, how to do this job... You'll do these things, you'll learn these things, like being part of a team, fighting together, finishing the job. For example, we got together and started a job, the job is half done, it doesn't matter, we did ninety percent of it, it doesn't matter, they shared us here, it doesn't matter, the job will be finished, that's the important thing.”

What are your thoughts on the future of the gaming industry? Which categories do you think will be more successful?

“I believe that the era of small teams, teams of 25-30 people, and teams of 40 people is coming in the gaming world. Why do I say this? The other day at D.I.C.E. in Vegas, a conversation was shared during the D.I.C.E. presentation, and I saw a few articles on this topic by some well-known analysts I admire. They are slowly starting to wake up to this. They predict that 20-30 person guerrilla teams will dominate the gaming world because think about it: if you want to make The Last of Us, you need \$250 million plus an advertising budget. If it fails, you've lost \$400 million, and Sony You've seen examples of this recently—how many games failed at the same time, despite having incredible advertising budgets of billions of dollars. Games are now very risky, so to reduce that risk, we're in a capitalist system where it's all about money. Companies will shift toward smaller teams with high effectiveness, solid technical knowledge, and good skills in the near future, I think. Of course, these small companies are small for 100 people, so think of them that way, because teams of 2000-3000 people are doing what teams of 1000 or 2000 people used to do, so there will be a contraction here, and you will see investment coming to small teams, that is, teams with the potential to get the job done.”

Ömer Faruk Güngör

Studio Director at Motion Blur

What is the secret to an indie game's success?

“For one thing, they can work guerrilla-style. For example, if you were working on motion capture at a Sony studio, like I mentioned earlier, it would go like this: the programmer would say, “Hey, we need this kind of enemy animation,” and then he'd go to the animator, who'd go to another animator, who'd check the animators' schedule, look at the mock-up schedule, and a week would go by. Here's how it works for us: You come to me, I put on my tights, I say, “Let's record,” I shoot it, and we give you the animation back the same evening. It's that fast. It's not just animation, this guerrilla system is incredibly fast, commercially speaking, of course... I don't mean cheap... but it gives you an advantage because think about it: You can't compete with me because I can do it at a much lower cost since I'm indie, and what's being discussed in the world right now, especially because of Wukong, is that for a game we made for over \$100 million, spending \$25 million on marketing isn't about making

Analysis Report

money back, it's about breaking even, and for that, we need to sell 3,200,000 copies at full price, which is very difficult. How many games can you sell? But I'm not like that. That's why even Kojima recently said something like this: When I founded Kojima Productions, I wanted to do big things with a smaller team of no more than 50 people. That's how I'd put it.”

Do you think a game's success is linked to its genre, or does the storytelling style make the game more engaging?

“Now, I'm a screenwriter, but the main thing is... the game... think about it this way: in cinema, there's a term called “show, don't tell,” and I think it's the same in games: show, don't tell. The more you integrate your story into the gameplay, the more effective it is, and the more enjoyable the game becomes. But the main thing is, is the gameplay enjoyable? Do I enjoy playing this game after watching the video? If yes, then is the story well told, does it affect me, is the music good? I can't say because we're in an interview, but we haven't announced our composer yet. I hope we're working with someone whose music you'll love, because the music is very important too. But first and foremost, is the game enjoyable? If not, then none of this matters, in my opinion, even though I'm a screenwriter.”

What is your biggest expectation from a game?

“I want it to tell a good story, and I want to be able to play for five hours at home without my wife bothering me. I haven't been able to do that since I got married because of work, but since I'm a screenwriter, the games I've enjoyed so far have always been story-driven ones: games like The Last of Us and Uncharted that tell a story and offer good gameplay. I also really like it when the development is refined. If there's something I mentioned that they could actually fix but left as default, I see it in my mind, and it bothers me a bit.”

What are the most critical areas of expertise for an indie team to succeed? If you were in their shoes, which area would you choose to specialize in?

“If I weren't a screenwriter, and if I didn't deal with business, marketing, and animation, I would want to be a material artist because material artists do incredible, artistic work. I love it, but I know nothing about it and have never done it. However, I think the most important area is not the basic criteria, but rather the crazy guy in charge of the project. Think about it: he wants this, and if he tells us, “Okay, Ömer, don't bother with this, it's enough for us,” then

Analysis Report

that's enough for us. He can even stop Ömer from wanting to work on it, but he always says, and unfortunately he does say, “Son, we can do better, we can be better, they've solved it, so we can solve it too.” The people he's talking about are Naughty Dog, but we can solve it, because as I just explained, when there's a desire to do it, it can be done, so...”

What is your preferred game engine for an indie game?

“As I always say, it depends on the project. If Unity serves the project better, then Unity, but if you're asking about our type of game, then Unreal, because it's used more in these types of games and I think it's more useful, and also because it's open source, we have this feature: we can open it up and read what they've written, intervene and change things. which is very important. However, the Witcher team moving over there will strengthen it significantly, especially if they're making an open world. Unreal isn't great there yet, but I think it will get even better.”

What are your thoughts on the future of the gaming industry?

“Hopefully, we'll sell over 10 million copies. That's my main focus. Beyond that, I hope that teams like Expedition, with 40-50 people, or Motionblur, with under 100 people, can find a more guerrilla-style pipeline, break free from Ubisoft and EA's budgets, and create more profitable, better games. I think it's shifting toward smaller studios, which is great for Turkey because we can't find 300 people who know how to do this, even if we had the money. We just can't. Of our 40 people, 39 of them, except for Kadir, have never made a game before, never finished a game before, except for him. When we told this to the IGM reporter from Japan, he was shocked and even wrote about it in his article. So we can't find that in Turkey, which is why I think a lot can change in the gaming industry with small, enthusiastic people who want to make games.”

Is there a category you expect to become popular for games?

“I was able to play RDR 2 last, then the project started, and I can't play games anymore.”

Serkan Özay

Co-Founder and Creative Director at Hero Concept

What was the most challenging part? Can you tell us about it?

“The pandemic. The game was made during the pandemic, so about 80% of it was done during the pandemic. The most challenging part for me was not having a fixed team and having to work remotely with everyone, freelancers and such. Also, back then, we weren't as well-known as we are now, so when I said, “We need this person on LinkedIn,” we didn't get 30 resumes flooding in. Of course, I'm talking about 2019-2020. How should I put it... There weren't many students available to work, and those departments weren't very popular back then, so we had a really hard time finding people to work with, especially for things like digital game design, which you're into. Back then, we had to create most of them from scratch. I can't quite find the right word, but let's say we trained them. Even now, I always do things like that. Instead of hiring someone who's already at an intermediate level, I usually take students and train them. That way, they pick up my super habits instead of someone else's. There's a bit of sarcasm here.”

After you developed this game, you experienced a sudden boom. Did you learn from the mistakes you made during the development process while working on other games, or what lessons did you learn?

“Well, success is relative, of course, but learning never stops, sir. Let me put it this way: you keep learning from every game you make. Plus, there's another aspect: you're not just making games here; ultimately, you also have an entrepreneurial identity. For example, recently, our team, which had 14 people last year, has shrunk considerably. So, right now, we're not just making games. If it were just about making games, that would be one thing, but we also have to survive as a business. You have a business life, you have an organization under your control, and keeping it alive is a separate thing, plus it's a huge human resources management task. So you have an animation team, a programming team, and so on. The daily performance of those individuals won't be consistent, their motivation won't be consistent, they might get another offer, they might want to work on their own projects, they might do something else. So there are many aspects to this. Learning all of this is impossible in Turkey anyway. I'm a creative director, so I'll only be working on creative projects. Nothing else really concerns me. For example, I also do art direction, game design, sometimes I deal

Analysis Report

with accounting, sometimes I chase investors, sometimes I deal with the business side of things. I also handle the PR side of things, as you can see. So, in situations like this, there's always something new to learn. Unfortunately, we can't list everything in the game, so you need to include this in your report. It's not like I just did the art, made the program, and that's it. Absolutely not. Don't try this at home.”

So how do you manage the division of labor within the team in this situation? How do you separate people like this? As you said, one person can work on more than one project.

“Yes, they can, that's right. First of all, you need to work with people who aren't afraid to get their hands dirty, that's one thing. Some people are like, “I don't want to know the technical side of it, I just want to do the artistic side, like just drawing,” for example. In the past, in games, before Hero Concept, in the game studios I worked at, I encountered that attitude a lot, like artists who only wanted to do art. Well, they might be right, but they're not. Because what you're doing has an integration aspect to it. You can't just put your graphics out there. Even if you're doing concept art, it's not like you just do the art and leave. You need to know the process it goes through when it's adapted to the engine and turned into an asset. “

You're an indie game developer, and we're indie too—we're a very small team, but could you recommend which engine we should use for the project to guide us?

“It doesn't matter how big or small it is; it starts with one person. You'll probably use Unity, or Godot could work too, but I'd go with Unity. It'll hold up for a while anyway, but the engine itself isn't that important. Also, what are you planning to make? If it's an FPS, use Unreal, man. I mean, if it's an FPS, because Unreal is already the default FPS engine. That's why even the least capable folks are making FPS games with it.”

Before you start developing the game, how do you decide on the game's genre and mechanics? What approach do you take?

“It depends on my current state of mind. If you ask me what kind of game I fantasize about, I've wanted to make a fighting game like Street Fighter since high school. I mean, I'm 47 years old, think about how many years I've wanted this, but I haven't taken on such a project yet. Because there are projects I can do, and projects I can't do. For example, if I tried to make a project like Street Fighter, you probably wouldn't be able to handle the server side of it because you need rollback servers and such. Suddenly, Street Fighter, Mortal Kombat, Dead

or Alive, and Injustice will pop up. But now, for example, we made a beat 'em up, and in beat 'em ups, we had Streets of Rage 4, which sold about a quarter of what Street Fighter 4 did, and it's the best in that genre, you know what I mean? Then there's Ninja Turtles, which sold around 3-5 million copies, while on the other side, some games sold 10-20-30 million copies. Getting involved with a few million copies means less competition. So, when choosing, you need to pick a place where the competition is a bit less but not completely barren.”

When designing the game, did gameplay or mechanics come to mind first?

“Here, we need to take a slightly traditional approach. What is at the core of this business? There are consoles, there are arcade halls, go back much further, okay? And what you're going to do is make a game for the console era, either your own or something else. Back in the day, when you walked into an arcade machine, you would see someone playing, look at it first from the outside, and then delve into its deeper mechanics. First, when you close your eyes—no, not close your eyes, let's say you stare at it—you have to choose what you'll see. It will look like a fighting game, but the next mechanic is what that fighting game will be like. Will you grapple the guy? Will you have special moves? I don't know, will you be able to hit the guy who falls to the ground? Will there be breakable objects? Look, these are all details. First, when you look from a distance, it's a general palette. Think of that arcade machine, or let's go back to the 1980s, say you go to your cousin's house, there's a console there, a Nintendo or something, and you see a quick image on the TV. That's what grabs you. That's the screenshot. When you look at the page on Steam, that screenshot is what grabs you first. Steam's video player has been redesigned, it's a bit faster now, but you know it's been problematic for a long time, you go full screen, it freezes, it doesn't work, but now it's starting to speed up a bit, so that image is important first, think of it like this, in the most basic aspect of a game, what a screenshot of a page like that looks like is very important. That's what grabs you.”

You said we should focus on more niche genres. Given the current trends in action, RPG, and shooter genres, do you think we should shift towards smaller genres?

“RPG seems like a big genre, but if you look at Baldur's Gate, it's an RPG, and Darkest Dungeon is an RPG too. Sure, Darkest Dungeon is selling millions now, but it wasn't a million-seller when it first came out. So, it was a game that a team like yours, or even a smaller team, could sit down and make. They did a great job with the stylization, creating that anime-like feel by slowly bringing the single-tower attack frame onto the screen and making it feel like you're hitting something. They worked smartly. So, instead of saying it's a small game, saying

it has a small scope might be more accurate in terms of genre. If you're making an RPG, like Baldur's Gate, you're bound to fail if you try to follow in its footsteps, but if you look at something similar in the same genre with scaled-down mechanics on a smaller, micro-scale, you won't be disappointed. You have to avoid going too 3D. When I say "too much 3D," I mean like overly detailed, hyper-realistic visuals—you need to avoid that. Because it's very difficult to capture that. The moment you do it, you have AAA companies as competitors, but they're more stylized, more artistic, appealing to a more unique community. Look, Darkest Dungeon is still a great game, Hades is a great example, but when stylization comes into play, it becomes niche, and then you're going for something that you can't appeal to AAA players anyway. The guy is already making a game for \$500 million, spending \$500 million on PR and marketing, so he doesn't even consider himself successful if he doesn't make a few billion in the first week. You're not there, and you never will be. Especially not while you're in Turkey. It's just too difficult. It's also about the money; you can't do something like that even if you have the budget in Turkey."

What do you think about the direction of the gaming industry and which genres will be more prominent in the coming years?

"These are two very separate questions. If you ask about the direction, it's currently terrible. Lots of teams are closing down, and lots of people are losing their jobs. For example, Obsidian closed down about 3-4 months ago, maybe 5 months ago, I don't remember exactly. They laid off 300 people there. Do the math, folks: if half of those 300 people decide to stay home and sell pickles, that leaves 150 people. If they form teams of five, that means 30 new indie teams will emerge. There aren't 30 good indie teams in Turkey right now, for example. I just mentioned Obsidian; how many more are there? People were laid off from Microsoft, from Xbox, from various PlayStation subsidiaries. What will they do? Plant onions? They'll make games. So, in the coming period, the indie era will explode again. 3D games with realistic graphics are already over, but they'll make those again. AAA will continue, but a new explosion in indie games is very likely. This also brings a danger: saturation. Suddenly, you'll have a lot of competitors. I always use this example, you're all in your 20s. Now, I'm from the class of '78. When I started working, there were probably less than 10 graphic designers in Turkey. Do you understand what I mean? So there was no competition. If you were a bit talented, enthusiastic, and willing to research—and let me add that getting information was difficult back then—you could stand out and get somewhere. Now, professor, you're probably competing with 100 people just at your school. Thousands in Istanbul, thousands across Turkey, and thousands worldwide. So, accessing information is easy—it's all on Google, YouTube, everywhere. So, what's left? It comes down to sitting down and working. Anyone who isn't lazy will get somewhere. Competition has skyrocketed.

When saturation gets really high, standing out in that game takes you to a completely different place. But I don't know about genres, sir. If you're talking about which genres, indie will rise, it will definitely rise. There's no way around it. But when you say indie, everything from A to Z comes into play.”

Despite the success of your first game, you've launched a Kickstarter for your second game. Do you have any advice for us regarding Kickstarter or funding?

“By the way, it's not the first game, it's the third game. As I said, it's a crisis period right now, everyone needs money. These are times when money is scarce. Kickstarter isn't a good platform, let me tell you that. I'd even say it's the worst possible platform for computer games. There are multiple technical reasons why you shouldn't get involved with it. We tried it back then; we had to do it, both financially and in terms of taking action, so we did it. If I had to do it again now, I wouldn't. Don't any of you have rich dads or something? Look, my advice is always this: while you're still students and don't have too many responsibilities, get together and make a game, but it doesn't have to be a super awesome game. Celeste isn't a very difficult game, guys. If you put your mind to it, you can make a game; you just need to lock yourself away for 5-6 months. It's no big deal; watch fewer TV shows, go out less, spend less time on Instagram, and the game will be done. That's what I'm trying to say. At this age, you either overclock or you don't, but it stays with you for life. So, I overclocked myself in my 20s, and I've been working ever since—I don't know how many years it's been. It hasn't gone away, and it's not going to fade away later either. That's why, at this age, once you get used to using your time well, you also need to learn to work as a team—meaning the hierarchy needs to function properly, everyone needs a sense of responsibility, everyone needs to be accountable to each other, and you need to figure out who can take on responsibility and who can't. Don't let your friends get in your way, guys. If they're slowing down the project, send them packing and bring in someone else. At least until you finish school, take on 1-2 projects and put them on Steam. Seriously, it's not that hard. There are tons of weird games on Steam; it doesn't matter if they sell or not. When the time comes tomorrow or the day after, you know you have a team effort, two games published, good or bad. When you're looking for a job somewhere tomorrow or the day after, this has value. It means you didn't just party during your student days. Okay, party, but don't just drift through life like a weed. What I'm saying is, valuing time is very important.”

We will talk about the funding methods I researched for our graduation project and the game we want to publish afterwards, the ones that are most suitable for us and what kind of funding method we will follow

First of all, let's start with the fund methods:

Self funding

It is the method by which our team members finance our game with their personal savings or income from their own business

Crowdfunding

Crowdfunding is the financing required for a project, initiative, or specific purpose by raising small amounts of money from a large number of individuals.

Grants & Competitions

Grants & Competitions are two main types of support for game developers: Gaming Funds (Grants), which provide direct financial support, usually in the form of non-refundable grants; Competitions are events where projects compete within certain rules to win cash prizes and visibility. Both help new game ideas come to life by reducing their financial risk.

Publisher (Publisher Agreement)

A game publisher invests in us and markets and distributes the game.

Early Access / Pre-Sales

Early Access is a model where a game is sold to players before development is completed and their feedback contributes to the development process. Pre-Sales, on the other hand, are when players purchase the game before the official release date to secure it, and sometimes this allows you to play (Advance Access) a few days before the full release of the game. Both offer developers early funding and market testing.

Investment / Angel Investors

Individual investors or small funds invest in exchange for becoming shareholders.

The roadmap for our group of five university students, who developed a medium-sized game using the Unity game engine for the PC platform, is as follows:

To ensure our financial sustainability, we will proceed with a four-stage approach that combines Self-Funding, Crowdfunding, Grants, and Early Access methods.

In the initial phase, we will rapidly develop the first playable prototype of the game (Minimum Viable Product - MVP) using our team's own resources (Self-Funding).

When this prototype is completed, we will take our project to international platforms and start a Crowdfunding campaign on Kickstarter or Indiegogo. This campaign is an important step in both gauging community support and evaluating development costs.

Simultaneously with this process, we will apply for national grants such as TÜBİTAK 1512 Techno-entrepreneurship Capital Support and sector-specific international funds (Grants) such as Epic MegaGrant in order to prove our innovative side. These grants will allow us to add more content to the game, providing additional financial security.

In the final stage, we will use the Steam Early Access model to bring our game to a large audience and start the revenue stream early. This method will allow us to prepare the game for the full version with active feedback from players, while at the same time generating sustainable income for our team.

This strategic funding roadmap is the strongest foundation for us to overcome the resource limitations of being a university student team and successfully publish our mid-sized game

Analysis Report

developed with Unity. Following the first step of our own resources, we will take the community power behind us with Crowdfunding, secure corporate financial support with grant applications, and achieve sustainable commercial success with Early Access and bring our project to the full version of our dreams. This integrated approach will not only complete a graduation project but also take the first solid step in our professional game development career.

Target Group Analysis

Analysis Report

Gender	Age	Weekly Hours	Genre	Roguelike	Turn-Based	2D/3D	Start Age	RPG	Platform	Control	Sound	Mode	Story	Focus	Stop Playing	Play Longer
Male	22	40+	FPS	Kinda fun	Yes	3D	6	Not really	PC	M&K	10	Multiplayer	4	Gameplay	Bugs, crashes	Competitive elements
Male	25	10	FPS	Nothing	Yes	3D	3	Yes	PC	M&K	9	Multiplayer	10	Gameplay	Work	Fun
Male	24	10	Sports / FPS	Nothing	Nothing	3D	15	No	PC	M&K	2	Multiplayer	1	Gameplay	Losing	Winning
Male	24	12	FPS	Good	No	3D	5	No	PC	M&K	1	Multiplayer	1	Gameplay	Work	Leisure
Female	21	1-2	Simulation	Likes them	Yes	3D	12	Yes	Console	Controller	10	Multiplayer	10	Gameplay	Loops	New details
Male	22	60	Indie	Likes them	Yes	3D	3	Yes	PC	M&K	8	Multiplayer	9	Gameplay	Boring loop	Great design
Male	22	8	Sports	Dislikes	Likes	3D	10	Yes	Console	Controller	8	Singleplayer	8	Gameplay	Nothing	Good story mode
Male	22	10	Party	Visually nice	Yes	3D	6	No	PC	M&K	6	Multiplayer	8	Gameplay	Bugs	Unique story, fun with friends
Male	24	3	Nostale	—	—	2D	16	No	PC	M&K	7	Multiplayer	7	Gameplay	—	—
Male	24	10	Football / FPS	Good	No	3D	12	Yes	PC	M&K	4	Multiplayer	8	Gameplay	Bugs	Good story, gameplay
Female	21	13	Gardenscapes	Boring	No	2D	8	Never played	PC	M&K	10	Singleplayer	9	Art	Eye strain	Colors, puzzles
Female	22	10-12	Adventure	Fun	Yes	3D	15	Yes	PC	M&K	6	Multiplayer	7	Gameplay	Repetitiveness	Competition
Male	22	28	Open World Action	Good for indies	No	3D	6	Addicted	PC	Controller	7	Singleplayer	9	Art	Boring loops	Doing something new
Male	22	16	Indie	Love them	Likes	2D	12	Likes	PC	M&K	10	Singleplayer	10	Art	FOMO	Game itself
Male	25	3	Story	Not bad	Yes	3D	3	Yes	Console	Controller	8	Singleplayer	8	Art	Responsibilities	—
Male	22	20	Strategy / Sports	Unsure	Yes	3D	8	Yes	Console	Controller	9	Singleplayer	10	Gameplay	Engagement	—
Male	29	3	Card Games	Kinda fun	Yes	2D	13	No	Console	Controller	9	Multiplayer	9	Art	Visually unappealing	Short match durations
Female	22	0-2	Story-based	Tension	No	3D	16	Yes	Mobile	M&K	8	Multiplayer	7	Art	Being alone	Friends
Female	21	12	RPG-like	Likes	Yes	2D	12	Yes	Console	Controller	10	Singleplayer	9	Art	Boring	Fun
Male	24	10	—	—	Not much	3D	10	Not much	Console	Controller	7	Singleplayer	8	Gameplay	—	—
Male	22	84	Sandbox	Enjoyable	Loves Persona	2D	10	Yes	PC	M&K	9	Multiplayer	7	Gameplay	Life	Life :(
Male	21	20-30	Action RPG	Fun	Yes	3D	2.5	Yes	PC	M&K	7	Singleplayer	8	Gameplay	Repetitive	Good story, curiosity
Male	24	2	FPS	Dislike	—	3D	7	Yes	PC	M&K	9	Multiplayer	2	Gameplay	Mediocrity	Realism, details
Male	20	8	Sports	Likes	Yes	3D	9	No	Console	M&K	8	Multiplayer	8	Art	Bad art, difficulty	Nice art, story
Male	22	35-40	Action	Likes	Yes	3D	4	Yes	PC	M&K	7	Multiplayer	8	Gameplay	Boring cutscenes, P2W	Story, design
Male	21	10+	Strategy	Don't know	Some	2D	7	Yes	Mobile	Controller	7	Multiplayer	7	Gameplay	Impossible progress	Challenges
Male	22	6	Sports	Enjoys	No	3D	7	No	PC	M&K	6	Multiplayer	7	Gameplay	Responsibilities	Progress
Male	22	6	RPG	Fun	—	2D	7	Yes	PC	M&K	3	Multiplayer	7	Gameplay	—	—
Female	22	5	Simulation	Likes	No	3D	7	Yes	PC	M&K	8	Singleplayer	10	Gameplay	Monotony	Strong story
Male	24	20	RPG	Inflated market	Yes	2D	9	Yes	PC	M&K	8	Singleplayer	6	Gameplay	Bugs, tedium	World, lore
Male	21	7	MMORPG	—	Yes	3D	12	Yes	Console	Controller	8	Singleplayer	8	Gameplay	—	Story, graphics
Female	21	2	RTS	Alright	Fun	3D	10	Yes	Mobile	M&K	5	Singleplayer	5	Gameplay	Being bad	Being good
Female	22	2	Board games	Boring	No	3D	12	Yes	Console	Controller	4	Multiplayer	8	Art	No time	Fun with friends
Male	28	20	Action / Survival	Likes some	Likes tabletop	3D	12-13	Yes	PC	M&K	10	Singleplayer	8	Gameplay	Repetition, bugs	Power fantasy
Male	23	16	FPS / Strategy	Dislike	Yes	3D	10	Yes	PC	M&K	7	Multiplayer	4	Gameplay	—	—

Table Summary

Total respondents: 35

19.10.2025

Analysis Report

Gender distribution: ~74% male, ~26% female

Average age: $\approx$ 22.8 years

Average weekly playtime: 15–20 hours (range 1–84)

Favorite genres:

FPS (8), Action (5), RPG (4), Sports (4), Indie (3), Simulation (2), Strategy (2)

Roguelike opinions: 65% positive / neutral, 35% dislike or indifferent

Turn-based games: ~55% like them, 45% not interested

2D vs 3D: 3D dominates (~70%)

RPG elements: 60% enjoy them

Platform preference: PC (70%), Console (25%), Mobile (5%)

Control type: Mouse & Keyboard ($\approx$ 65%), Controller (30%)

Music/Sound importance: Avg. 8/10

Game mode preference: Multiplayer (60%), Singleplayer (40%)

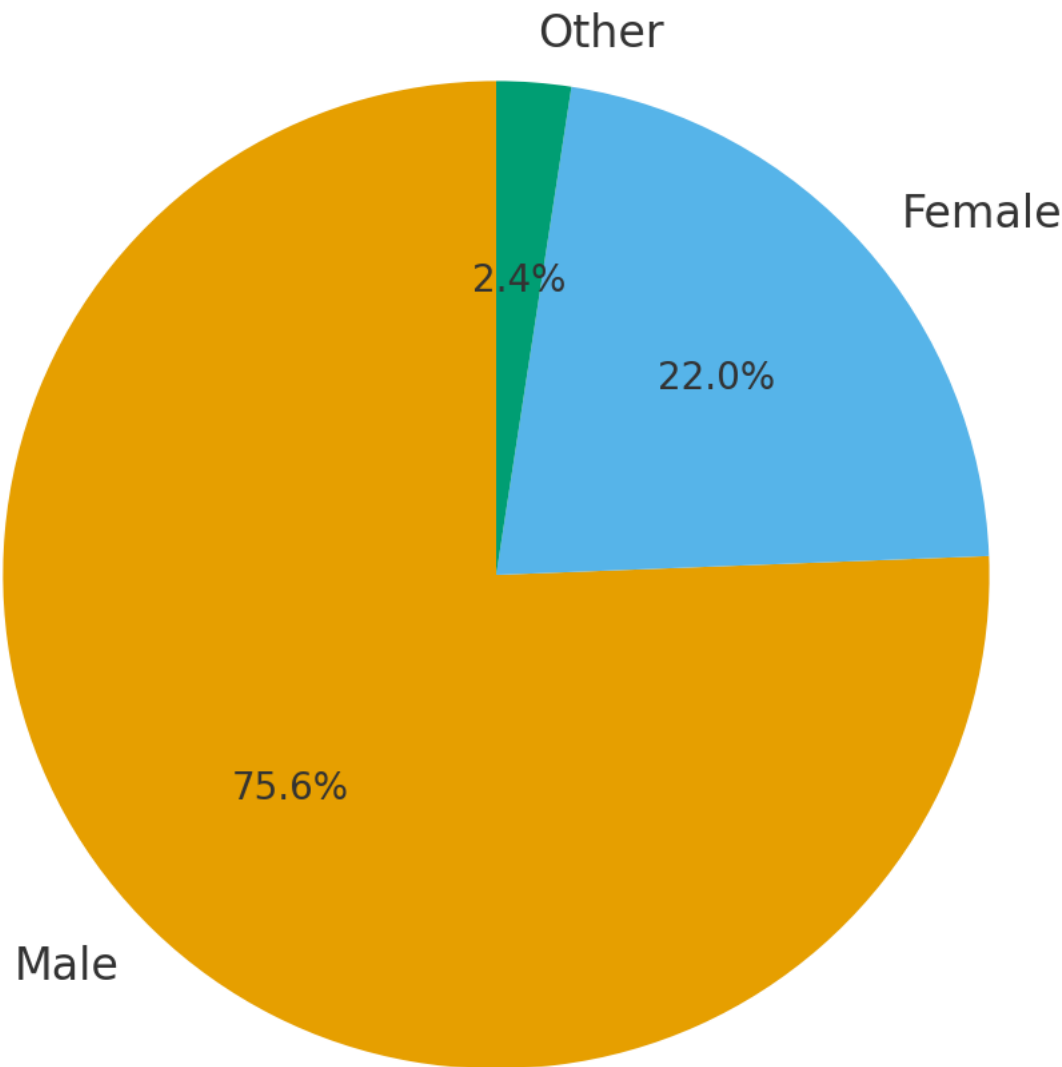
Story importance: Avg. 7.5/10

Gameplay vs Art: 70% prioritize gameplay

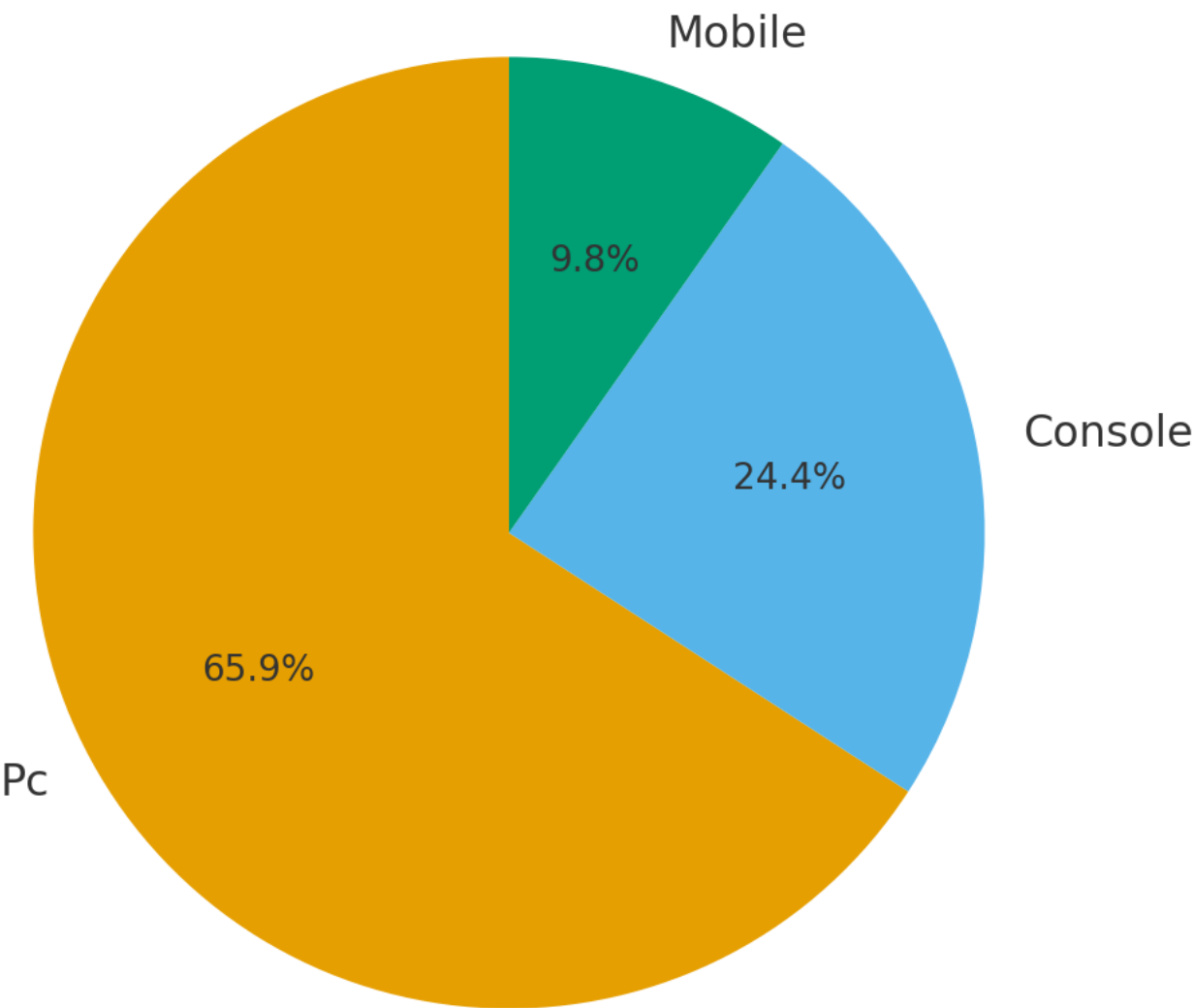
Most common reasons for quitting: Repetitiveness, bugs, responsibilities, boredom

Most common reasons for playing longer: Good design, story, competition, fun with friends, creativity

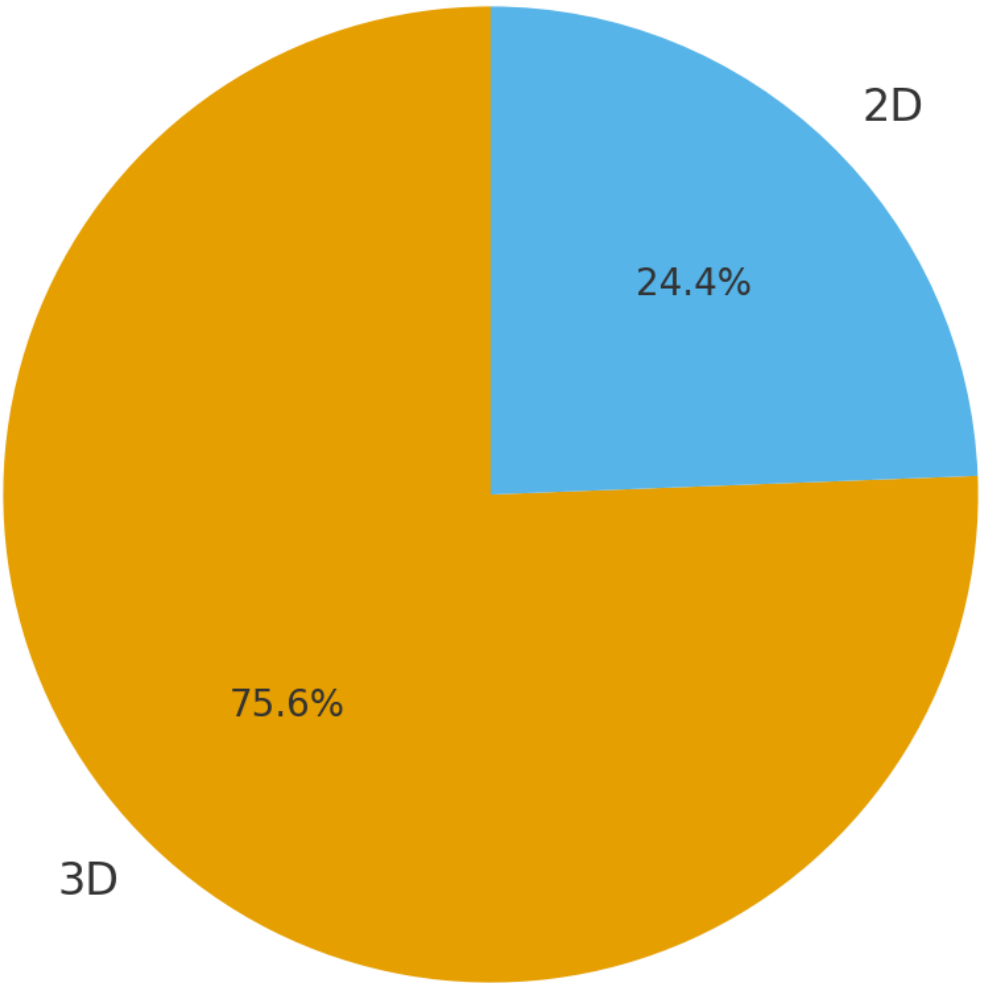
Gender Distribution of Target Audience



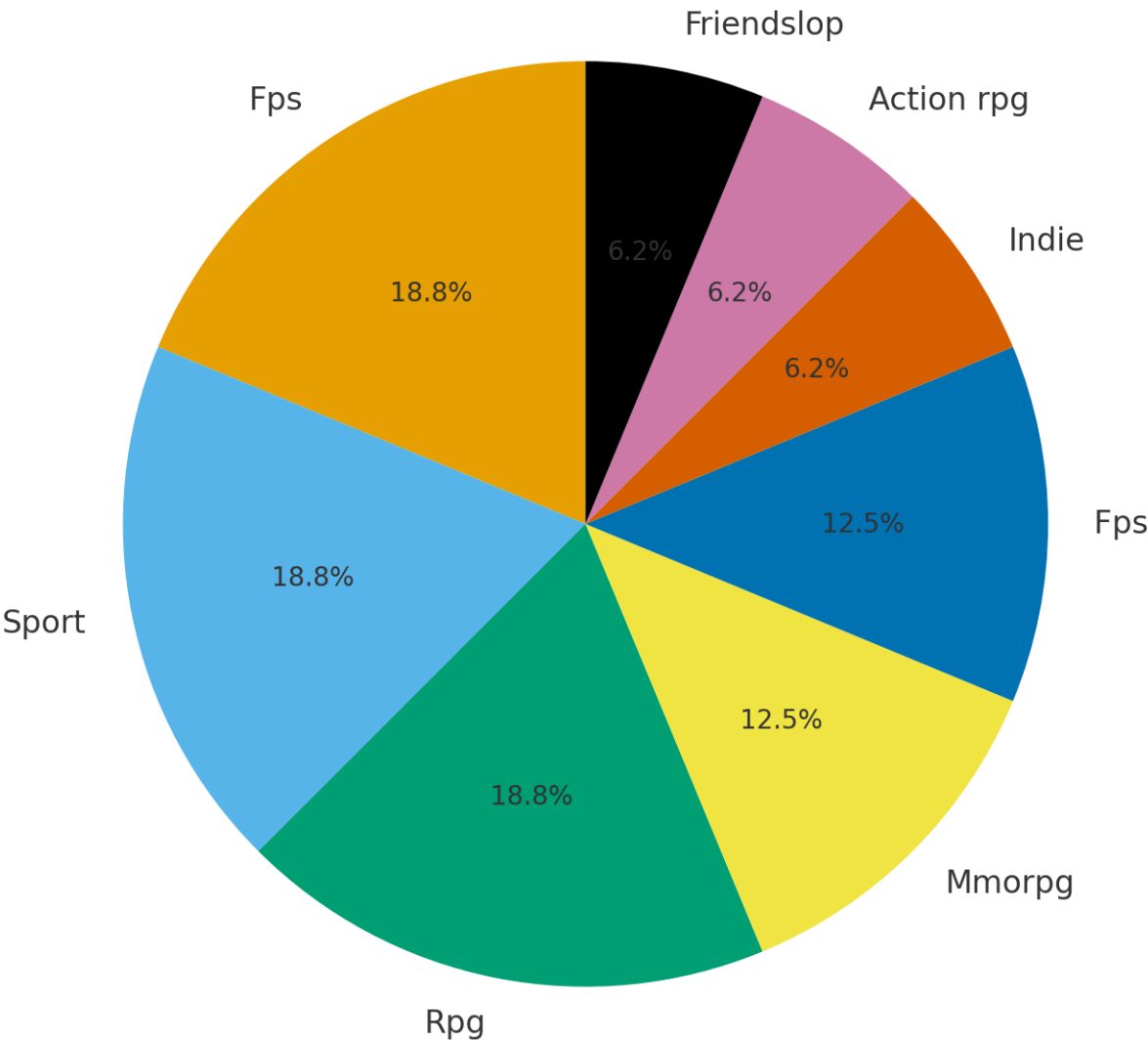
Preferred Gaming Platform



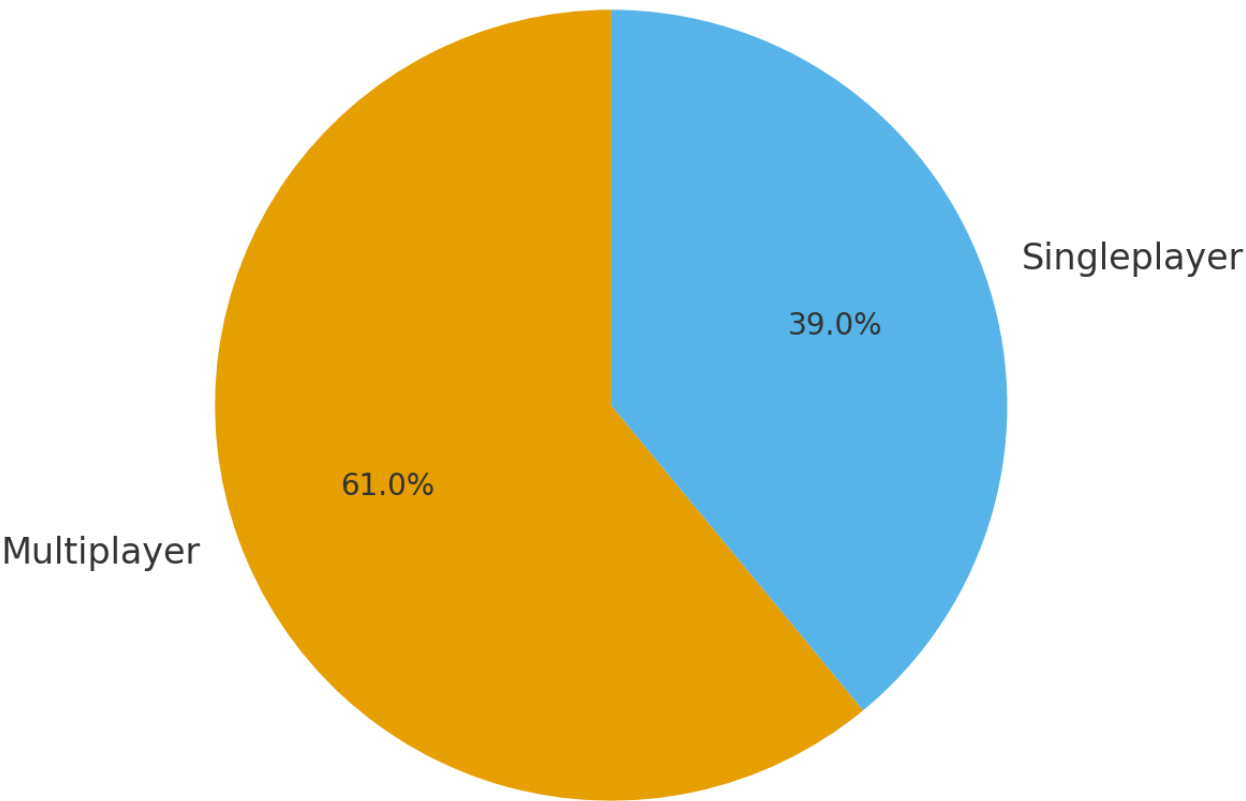
2D vs 3D Game Preference



Favorite Game Genres (Top 8)



Singleplayer vs Multiplayer Preference



Analysis Report

The survey of 35 participants and the accompanying interviews reveal that our main audience is made up of 21–24-year-old male gamers, gaming 10–20 hours weekly, primarily on PC with mouse and keyboard. As both Serkan Özey and Kadir Demirden emphasized during the interviews, this group is the most engaged and feedback-driven segment of the indie gaming community, to whom game clarity and control responsiveness are the indicators of quality.

Quantitative data show significant preference for 3D environments ($\approx 70\%$) and gameplay over art style ($\approx 70\%$), yet interviews refer to the importance of visual identity and sound design to achieve affective richness. Ömer Faruk Güngör emphasized that "music and atmosphere complete the player's emotional link," which confirms the high sound importance rating (average 8/10) of the survey.

Genre-wise, FPS, Action, and RPG dominate, while 60% enjoy RPG elements and 65% respond positively to Roguelike mechanics. This suggests that our audience values progression, challenge, and replayability, supporting Özey's argument that "niche, mechanic-driven designs offer more lasting engagement for smaller studios." Likewise, the 55% acceptance of turn-based systems shows openness to our Turn-Based Roguelite direction—if it balances depth with tempo.

Inspiring statistics concur that players exit due to repetition, glitches, or lack of meaning but stay longer due to good loop design, storytelling coherence, and competition. As Demirden claimed, "responsiveness and rhythm are what keep the player inside the loop," confirming the need for responsive, reactive design.

Overall, each data set heralds a single design imperative: a mechanically refined but stylistically distinctive Roguelite RPG with robust feedback mechanisms, expressive art direction, and cultural diversity. This coming together of quantitative player behavior and qualitative expert feedback validates our chosen creative direction in the Turkish indie scene.

Game Development Tool / Game Engine Analysis

UNITY 6 SYSTEM REQUIREMENTS

Windows

Operating System Version:

Windows 10 version 21H1 (build 19043) or newer (X64), Windows 11 21H2 (build 22000) or newer (Arm64)

CPU:

X64 architecture with SSE2 instruction set support, Arm64

Graphics API:

DX10, DX11, DX12 or Vulkan capable GPUs

RAM:

8 GB

Additional Requirements:

Hardware vendor officially supported drivers

Linux

Operating System Version:

Ubuntu 22.04, Ubuntu 24.04

CPU:

X64 architecture with SSE2 instruction set support

Graphics API:

OpenGL 3.2+ or Vulkan-capable, Nvidia and AMD GPUs

RAM:

Analysis Report

8 GB

Additional Requirements:

Gnome desktop environment running on top of X11 or Wayland windowing system, Nvidia official proprietary graphics driver, or AMD Mesa graphics driver. Other configuration and user environment as provided stock with the supported distribution (Kernel, Compositor, etc.) Notes: • Ubuntu 22.04: Wayland is supported with AMD graphics cards. • Ubuntu 24.04: Wayland is supported with AMD graphics cards and Nvidia graphics cards utilizing Nvidia proprietary graphics drivers 550 and above.

macOS

Operating System Version:

Big Sur 11 or newer

CPU:

X64 architecture with SSE2 instruction set support (Intel processors) Apple M1 or above (Apple silicon-based processors)

Graphics API:

Metal-capable Intel and AMD GPUs

RAM:

8 GB

Additional Requirements:

Apple officially supported drivers (Intel processor) Rosetta 2 is required for Apple silicon devices running on either Apple silicon or Intel versions of the Unity Editor

[Unity - Manual: System requirements for Unity 6.0](#)

BLENDER SYSTEM REQUIREMENTS

Windows

Minimum System Requirements

Operating System Version:

Windows 8.1 (64-bit)

CPU:

4 cores with SSE4.2 support

Graphics API:

2 GB VRAM with OpenGL 4.3 (see below)

NVIDIA: GeForce 400 and newer, Quadro Tesla GPU architecture and newer, including RTX-based cards, with NVIDIA drivers. AMD: GCN 1st gen and newer. Intel: Broadwell architecture and newer. Always make sure to install the latest drivers from the graphics card manufacturer website.

RAM:

8 GB

Recommended System Requirements

Operating System Version:

Windows 10 or Windows 11

CPU:

8 cores

Graphics API:

Analysis Report

8 GB VRAM

RAM:

32 GB

Linux

Minimum System Requirements

Operating System Version:

Distribution with glibc 2.28 or newer (64-bit)

CPU:

4 cores with SSE4.2 support

Graphics API:

2 GB VRAM with OpenGL 4.3 (see below)

NVIDIA: GeForce 400 and newer, Quadro Tesla GPU architecture and newer, including RTX-based cards, with NVIDIA drivers. AMD: GCN 1st gen and newer. Intel: Broadwell architecture and newer. Always make sure to install the latest drivers from the graphics card manufacturer website.

RAM:

8 GB

Recommended System Requirements

Operating System Version:

Distribution with glibc 2.28 or newer (64-bit)

CPU:

8 cores

Graphics API:

8 GB VRAM

RAM:

32 GB

macOS

Minimum System Requirements

Operating System Version:

macOS 11.2 (Big Sur)

CPU:

Apple Silicon or Intel

Graphics API:

GPU with Metal 2.2 (see below)

AMD, Apple Silicon, or Intel (Skylake and newer).

RAM:

8 GB

Recommended System Requirements

Operating System Version:

macOS 15 (Sequoia)

CPU:

Apple Silicon

Graphics API:

Apple Silicon

RAM:

32 GB

[Requirements — Blender](#)

ADOBE PHOTOSHOP SYSTEM REQUIREMENTS

Windows

Minimum System Requirements

Operating System Version:

Windows 10 64-bit (version 21H2) or later; LTSC versions are not supported

CPU:

Multicore Intel® or AMD processor (2 GHz or faster processor with SSE 4.2 or later) with 64-bit support

Graphics API:

GPU with DirectX 12 (feature level 12_0 or later) 1.5 GB of GPU memory GPUs less than 7 years old (update display drivers from the GPU manufacturer's website). Currently, we don't support testing on GPUs older than 7 years

RAM:

8 GB

Recommended System Requirements

Operating System Version:

Windows 10 64-bit (version 21H2) or later; LTSC versions are not supported

CPU:

Analysis Report

Multicore Intel® or AMD processor (2 GHz or faster processor with SSE 4.2 or later) with 64-bit support

Graphics API:

GPU with DirectX 12 support (feature level 12_0 or later) 4 GB of GPU memory for 4k displays and greater

RAM:

16 GB or more

macOS

Minimum System Requirements

Operating System Version:

macOS Big Sur (version 11.0) or later

CPU:

Multicore Intel® or Apple Silicon processor (2 GHz or faster processor with SSE 4.2 or later) with 64-bit support

Graphics API:

GPU with Metal support 1.5 GB of GPU memory

RAM:

8 GB

Recommended System Requirements

Operating System Version:

macOS Ventura (version 13.5.1) Installation is blocked on v10.15.x

CPU:

ARM-based Apple Silicon processor

Graphics API:

GPU with Metal support 4 GB of GPU memory for 4k displays and greater

RAM:

16 GB or more

[Photoshop 25.x system requirements](#)

ADOBE PREMIERE SYSTEM REQUIREMENTS

Windows

Minimum System Requirements

Operating System Version:

Windows 10 (64-bit) V22H2 or later

CPU:

Intel® 6th Generation or newer CPU – or AMD Ryzen™ 1000 Series or newer CPU Advanced Vector Extensions 2 (AVX2) Support required

Graphics API:

2 GB of GPU memory

RAM:

8 GB

Recommended System Requirements

Operating System Version:

Windows 10 (64-bit) version 22H2 (or later) or Windows 11.

CPU:

Analysis Report

Intel® 11th Generation or newer CPU with Quick Sync – or AMD Ryzen™ 3000 Series / Threadripper 2000 series or newer CPU

Graphics API:

4 GB of GPU memory for HD and some 4K media 6 GB or more for 4K and higher

RAM:

16 GB of RAM for HD media 32 GB or more for 4K and higher

macOS

Minimum System Requirements

Operating System Version:

macOS Monterey (version 12) or later

CPU:

Intel® 6th Generation or newer CPU Advanced Vector Extensions 2 (AVX2) Support required

Graphics API:

Apple Silicon: 8 GB of unified memory Intel: 2 GB of GPU memory

RAM:

8 GB

Recommended System Requirements

Operating System Version:

macOS Monterey (version 12) or later

CPU:

Apple silicon M1 or newer

Graphics API:

Apple silicon: 16 GB of unified memory

RAM:

Analysis Report

Apple silicon: 16 GB of unified memory

[Premiere Pro 24.x system requirements](#)

ADOBE AFTER EFFECTS SYSTEM REQUIREMENTS

Windows

Minimum System Requirements

Operating System Version:

Windows 10 (64-bit) version 22H2 or later

CPU:

Intel® 6th Generation or newer CPU, or AMD Ryzen™ 1000 Series or newer CPU Advanced Vector Extensions 2 (AVX2) support required

Graphics API:

Windows with NVIDIA GPU: GPU must be of the Maxwell generation or newer and have at least 4GB of VRAM Windows with Intel or AMD GPU: Discrete GPU with at least 4GB VRAM and support for OpenCL

RAM:

16 GB

Recommended System Requirements

Operating System Version:

Analysis Report

Windows 10 (64-bit) version 22H2 (or later) or Windows 11

CPU:

Intel® 11th Gen or newer CPU with Quick Sync – or AMD Ryzen™ 3000 Series / Threadripper 3000 series or newer CPU

Graphics API:

8 GB of GPU Memory

RAM:

16 GB of RAM for HD media 32 GB or more for 4K and higher

macOS

Minimum System Requirements

Operating System Version:

macOS Ventura (version 13)

CPU:

Intel® 6th Generation or newer CPU Advanced Vector Extensions 2 (AVX2) Support required

Graphics API:

Apple silicon: 16GB of RAM Intel Macs with a discrete AMD GPU: 4GB of VRAM

RAM:

16 GB

Recommended System Requirements

Operating System Version:

macOS Ventura (version 13) and later

CPU:

Apple silicon M1 Pro, M1 Max, M1 Ultra or newer

Graphics API:

Apple silicon: 16 GB of unified memory

Analysis Report

RAM:

Apple silicon: 16 GB of unified memory

[After Effects System Requirements](#)

ADOBE SUBSTANCE 3D PAINTER SYSTEM REQUIREMENTS

Windows

Minimum System Requirements

Operating System Version:

Windows 11 64-bit Version 23H2

CPU:

Intel Core i5 AMD Ryzen 5

Graphics API:

NVIDIA GeForce RTX 2060 Super NVIDIA Quadro RTX 4000 AMD Radeon RX 5700 XT AMD Radeon Pro W5700

8 GB

RAM:

16 GB

Recommended System Requirements

Analysis Report

Operating System Version:

Windows 11 64-bit Version 24H1

CPU:

Intel Core i7 AMD Ryzen 7

Graphics API:

NVIDIA GeForce RTX 3080 NVIDIA Quadro RTX A4000 AMD Radeon RX 6800 XT AMD Radeon Pro W7700

16 GB

RAM:

32 GB

Optimal System Requirements

Operating System Version:

Windows 11 64-bit Version 24H2

CPU:

Intel Core i9 AMD Ryzen 9

Graphics API:

NVIDIA GeForce RTX 4090 NVIDIA Quadro RTX 5000 Ada Generation AMD Radeon RX 7900 XTX AMD Radeon Pro W7800

24 GB

RAM:

64 GB

macOS

Minimum System Requirements

Operating System Version:

macOS 12 Monterey

CPU:

Analysis Report

Apple M1

Graphics API:

Apple M1

RAM:

16 GB

Recommended System Requirements

Operating System Version:

macOS 13 Ventura

CPU:

Apple M2 Pro

Graphics API:

Apple M2 Pro

RAM:

32 GB

Optimal System Requirements

Operating System Version:

macOS 14 Sonoma

CPU:

Apple M4 Pro

Graphics API:

Apple M4 Pro

RAM:

64 GB

[System requirements | Substance 3D Painter](#)

JETBRAINS RIDER SYSTEM REQUIREMENTS

Windows

Operating System Version:

10 version 1809 and later 11

CPU:

2 GHz

RAM:

4 GB

Linux

Operating System Version:

Two latest versions of Ubuntu LTS or Fedora Linux Gnome or KDE X Window System (X11)
GLIBC 2.29 or later

CPU:

Analysis Report

2 GHz

RAM:

4 GB

macOS

Operating System Version:

12.0

CPU:

2 GHz

RAM:

4 GB

[System Requirements - dotUltimate | ReSharper](#)

REAPER SYSTEM REQUIREMENTS

Minimum System Requirements

Operating System Version:

Windows: XP/Vista/7/8/10/11 (32 or 64-bit Intel/AMD, beta support for arm64ec) macOS: 10.5 - 15.x (Intel and ARM64) Linux: Requires GTK+3 GTK+2 supported with additional effort Supports x86_64, i686, armv7l, aarch64

CPU:

Intel Core i7-9700K | AMD Ryzen™ 5 5600 6-Core or above

Graphics API:

NVIDIA GTX 1050, RX 560

2 GB

RAM:

4 GB

[REAPER system requirements | Can I Run REAPER](#)

[REAPER | About](#)

GITHUB SYSTEM REQUIREMENTS

Operating System Version:

macOS 12.0 or later Windows 10 64-bit or later. You must have a 64-bit operating system to run GitHub Desktop.

GitHub doesn't publish a detailed hardware spec list (CPU speed, RAM, disk, GPU) but we asked AI and it says:

You'll want sufficient RAM (4 GB or more is typical for comfortable desktop app usage, especially when working with larger repositories).

Adequate disk space for the GitHub Desktop app itself plus local clones of repositories.

A stable Internet connection (to connect with GitHub, push/pull changes).

Permissions to install software on your system (especially on Windows).

On macOS, the app will use Keychain for credentials, so correct permissions for related directories may matter.

[Supported operating systems for GitHub Desktop - GitHub Docs](#)

[ChatGPT](#)

Technical capacities or deficiencies of the tools and why do we choose these tools for our project?

Unity

PROS	CONS
It is easier to make cross platform games.	Sometimes performance and optimization issues.
It is strong at 3D and 2D game development at the same time.	Engine is not open source.

Analysis Report

It has a huge asset store.	Not free after having more than \$200K USD of revenue or funds raised in the prior 12 months.
It is easier to find information about Unity on the web compared to other engines.	
It is easier to improve our skills on Unity Engine compared to Unreal Engine.	
In terms of not overworking the computer, it's better compared to Unreal.	
We learned a lot more information about Unity Engine in our first three years compared to other engines.	

Blender

PROS	CONS
All free usage and open source.	Less advanced rigging and animation tools compared to Maya.
Users can do modeling, sculpting, rigging, animation and rendering at the same application.	It is great for indie games but most of the AAA games uses Maya to make their creations.
There are a lot of great tutorials on the web.	

Analysis Report

Users can customize according to their expectations.	
Free add-ons.	
Pratik and easy to learn UI design and tools.	
We learned Blender as a modelling tool at school and that's why everybody has a clue about it at least.	

Adobe Creative Cloud (Photoshop / Premiere / After Effects)

PROS	CONS
Industry mostly prefers Creative Cloud apps.	Users should pay every month to use it.

Analysis Report

Great ecosystem between apps.	Contains many unnecessary applications and tools for game designers. It makes the applications resource-heavy.
Advanced feature sets and newly added AI tools.	There are open source and free alternatives such as GIMP.
User friendly and similar UI design.	
It is easy to find information about those apps on the web.	
Cloud support on the app.	
We learned Creative Cloud apps as lessons at school.	

Adobe Substance 3D Painter

PROS	CONS

Analysis Report

Industry standard for BPR material creation for games.	It is hard to learn for newcomers like us.
Efficient and good designed workflow.	There is a dependency on baking process. And it adds an extra step to asset pipeline.
Contains powerful procedural tools.	
Let the user free to stylize their project however they want.	
Free for us to use it because school gives us.	

Rider

PROS	CONS
Perfect performance and UI design compared to VS 2022.	It could be overwhelming for newcomers because of the crowded UI design and features.
Contains advanced code analysis.	
Cross platform support.	
Free plug-ins and AI support.	
We can use it for free because we are student.	
We used Rider as IDE before on our other projects.	

Reaper

Analysis Report

PROS	CONS
Free for 60 days and inexpensive price policy.	Less conventional interface compared to other DAWs.
Fast and light program that can work on every PC trouble free.	It is a DAW tool that you can make your own SFXs or music's not an audio middleware like FMOD or Wwise for implementation.
Highly open to customization and add-on usage.	It is not a fully professional tool like FL Studio.
Contains powerful batch rendering specification.	It is hard to find tutorials about Reaper.
We have a MIDI that we can connect to Reaper.	
We learned the basics of Reaper at the school.	

GitHub

PROS	CONS
Contains free usage for indie usage.	Hard to push large files because of storage limitations. Users should use LFS system for large files and it is harder to use it compared to desktop app.
Industry standard between developers.	It is less artist friendly compared to systems like Perforce.
Massive ecosystem. Integration with lots of IDEs and third-party tools.	Git LFS contains file locking as an add-on. But it is not native like systems like Perforce. It might cause conflict between binary files.
Contains strong tools for coders and team members.	
We used GitHub on our old projects. We are experienced about usage of GitHub Desktop app.	

Our team capacities toward the game design and development programs

Names/Programs	Unity	Blender	PS	Premiere	AE	Substance	Rider	Reaper	GitHub
Kaan	8	6	5	6	4	1	7	6	6
Derin	5	7	9	9	7	1	5	8	2
Anıl	7	4	5	4	2	1	2	2	4
Kağan	7	8	9	6	3	3	2	4	4
Bora	6	8	6	5	5	1	4	5	2

*We gave ourselves a score between 1 to 10 according to our skills. 1 is lowest 10 is maximum capability.

Timeline and Practical Planning for The Analysis

NAMES	Tasks	Time Spent
Kaan	Game tool analysis / interview	15h+
Derin	Interview analysis and transcription	15h+
Anil	Funding methods / interview	15h+
Kagan	Game market analysis / interview	15h+
Bora	Target audience analysis / interview	15h+

Final Introduction

In the final stage of our graduation project, our research evolved from an initial concept into a comprehensive case study that integrates academic questioning, professional insight, and practical design implementation. Centered on the development of a turn-based RPG with roguelite mechanics inspired by Turkish mythology and shamanic traditions, the project embodies the fusion of cultural representation and independent game development practice. Throughout the process, our approach was informed not only by technical and market analyses but also by a quality investigation based on interviews with three professionals in the Turkish gaming industry: Kadir Demirden and Ömer Faruk Güngör from Motion Blur Studio, and Serkan Özay from Hero Concept. Their perspectives shaped the theoretical, managerial, and creative direction of the project and helped us align our focus with the realities of modern indie game production.

Kadir Demirden's reflections showed us the complex, non-linear nature of success in the gaming ecosystem. As he emphasized, "no one truly knows the secret formula" for creating a successful game, because games are inherently multi-parametric systems of interaction and emotion. His insistence on player responsiveness and feedback-based repetition reinforced our own focus on designing mechanics that gives immediate engagement and satisfaction. Demirden also stressed the need for teams to operate with humility, open communication, and understanding that technical skill is only one part of a greater creative philosophy, a principle that guided our team dynamics and workflow structure.

Ömer Faruk Güngör expanded on this by advocating what he termed guerrilla-style production; a model of agile, low-cost, and high-flexibility development unique to indie studios. His explanation that "small teams can achieve efficiency by removing unnecessary bureaucracy" validated our modular and adaptive workflow. Güngör's philosophy that gameplay enjoyment must precede all narrative and visual ambitions encouraged us to refine our prototype through multiple playtesting cycles before implementing advanced narrative systems. Additionally, his view that storytelling should adhere to the cinematic principle of "show, don't tell" shaped our integration of Turkish mythological motifs directly into gameplay mechanics rather than confining them to dialogue or cutscenes.

Serkan Özey contributed a pragmatic, long-term perspective, connecting creative identity to market sustainability. He highlighted that the future of the gaming industry lies in small, stylistically distinctive teams capable of delivering artistic products at manageable scales. Özey's observation that "stylization and creative identity outweigh photorealism" resonated with our art direction decisions thus leading us to employ Blender and Substance 3D Painter to achieve a painterly, symbolic visual aesthetic rather than hyper-realistic rendering. His advice regarding team discipline and production hierarchy also influenced our internal task allocation system, ensuring accountability and collaborative efficiency across design, programming, and narrative departments.

Building on these professional insights, our funding and production roadmap integrates theoretical frameworks with industry-tested practicality. Following the multi-stage model of Self-Funding → Crowdfunding → Grants → Early Access, our strategy reflects Özey's and Güngör's emphasis on financial realism and entrepreneurial initiative. Özey's skepticism toward overreliance on Kickstarter underscored our preference for diversified funding mechanisms that balance exposure with sustainability.

Ultimately, this project stands as a combination of academic theory, cultural expression, and experimental industry learning. It demonstrates how localized creative visions—rooted in Turkish mythological heritage—can evolve into globally relevant design frameworks when combined with disciplined production, technical proficiency, and agile management. Combining expert insights with practical experimentation, we aimed not only to create a functional prototype but also to articulate a replicable model for young developers in emerging markets. This Final Introduction thus encapsulates the intellectual and creative maturation of our team, serving as both the conceptual cornerstone and reflective summary of a project that bridges culture, innovation, and independent development in the global gaming landscape.

REFERENCES

<https://unity.com/products>

<https://www.agicent.com/blog/comparison-between-game-development-tools/>

<https://www.anchorpoint.app/blog/git-vs-perforce-for-game-development>

<https://www.anchorpoint.app/blog/using-file-locking-in-git>

<https://www.artstash.io/resources/perforce-vs-git-for-game-development>

<https://www.daviddumaisaudio.com/why-reaper-for-game-audio/>

<https://www.diva-portal.org/smash/get/diva2:1767272/FULLTEXT01.pdf>

<https://www.femaleswitch.com/directories/tpost/1g36id0xx1-top-10-open-source-alternatives-to-adobe>

<https://www.jetbrains.com/lp/dotnet-unity/>

<https://www.jetbrains.com/rider/>

<https://www.jetbrains.com/rider/features/>

<https://www.juegostudio.com/blog/maya-vs-blender>

<https://www.mdpi.com/2079-3200/13/6/60>

<https://www.perforce.com/blog/vcs/git-vs-perforce-how-choose-and-when-use-both>

<https://www.quora.com/What-is-the-correct-pipeline-for-game-creation-for-an-indie-developer>

<https://www.reaper.fm/>

<https://www.redappletech.com/blog/blender-vs-maya/>

https://www.researchgate.net/publication/232876212_Using_the_blender_game_engine_for_real-time_emulation_of_production_devices

https://www.researchgate.net/publication/300481280_Using_Unity_to_Teach_Game_Development

Analysis Report

https://www.researchgate.net/publication/330292587_Engineering_Queerness_in_the_Game_Development_Pipeline

https://www.researchgate.net/publication/394124171_Blender_3D_as_a_Catalyst_for_Indie_Game_Development

<https://www.stephenschappler.com/2024/07/12/reaper-vs-pro-tools-for-sound-design/>

<https://www.stephenschappler.com/2024/09/12/my-reaper-environment-for-sound-design/>

<https://www.thegameaudioco.com/wwise-or-fmod-a-guide-to-choosing-the-right-audio-tool-for-every-game-developer>

<https://www.xda-developers.com/open-source-graphics-tools-that-outperform-adobe-creative-cloud/>

<https://www.youtube.com/watch?v=1twvtuKMSbY>

<https://www.youtube.com/watch?v=7wHP16uXjGg>

<https://www.youtube.com/watch?v=DtajSQt5GsE>

<https://www.youtube.com/watch?v=j1Ds7iuawll>

<https://www.youtube.com/watch?v=jTWuOnEOL74>

<https://www.youtube.com/watch?v=jUN7Y9rltAk>

<https://80.lv/articles/creating-painterly-style-tiles-material-with-substance-3d-designer>

<https://archive.headconf.org/head22/wp-content/uploads/pdfs/14642.pdf>

<https://business.adobe.com/blog/perspectives/how-and-why-adobe-is-making-open-source-a-strategic-priority>

https://cdn.prod.website-files.com/638106149a6441f0708f5c0a/6877b1a635a7eeb1b94994cd_lfs_performance.webp

<https://developer.adobe.com/creative-cloud/>

<https://epub.uni-bayreuth.de/4330/1/THE%20DESIGN%20OF%20INDIE%20GAMES%20A%20DIFFERENT%20PARADIGM.pdf>

Analysis Report

<https://fadingeek.medium.com/i-tried-using-proprietary-and-open-source-softwares-side-by-side-heres-my-thoughts-a1e6afae4acd>

<https://files.eric.ed.gov/fulltext/EJ1290793.pdf>

<https://flutumusic.com/2025/01/30/audio-middleware-game-development/>

<https://forums.unrealengine.com/t/substance-painter-designer-vs-quixel/20902>

<https://fosslinux.community/forum/software-applications/any-good-open-source-alternatives-to-adobe-creative-suite-for-linux/>

<https://garagefarm.net/blog/top-4-tips-for-switching-to-quixel-mixer-from-a-substance-user>

<https://get.assembla.com/blog/git-vs-perforce-game-development/>

<https://get.assembla.com/blog/perforce-vs-git/>

<https://github.com/agmmnn/awesome-blender>

<https://github.com/butStill/UnityResourceMegaList>

<https://github.com/cobeylab/version-control-training>

https://github.com/mikecroucher/Git_Academic_Benefits

<https://github.com/mikeroyal/Unity-Guide>

<https://guof.people.clemson.edu/papers/jcscw19.pdf>

<https://javierzumer.com/blog/2022/6/28/differences-between-fmod-amp-wwise-part-1>

<https://javierzumer.com/blog/2022/8/28/differences-between-fmod-amp-wwise-part-3>

<https://journals.umt.edu.pk/index.php/jdt/article/view/6771>

<https://matthaliski.com/blog/the-software-used-in-game-development>

<https://medium.com/@amit25173/godot-vs-unity-vs-unreal-775fa7d91b2c>

<https://medium.com/@jashan/going-from-visual-studio-to-rider-4b69320f91f2>

<https://medium.com/@void1gaming/visual-studio-vs-rider-c362cbb49f8>

<https://pinnguaq.com/learn/introduction-to-sound-design-with-reaper/>

<https://pmc.ncbi.nlm.nih.gov/articles/PMC4718703/>

Analysis Report

<https://practicalmedia.io/article/what-are-the-essential-game-development-tools-every-indie-developer-needs>

<https://qa.fmod.com/t/reaper-sound-design-workflow-one-project-per-game-or-one-project-per-sound/17138>

<https://rebusfarm.net/blog/blender-or-maya-which-3d-tool-should-you-choose>

<https://rocketbrush.com/blog/godot-vs-unity-vs-unreal-comparison>

<https://stackify.com/visual-studio-rider/>

<https://steamcommunity.com/app/365670/discussions/0/3192489172313267651/>

<https://www.adobe.com/creativecloud/business/enterprise/design-to-development-workflow.html>

<https://www.adobe.com/learn/substance-3d-designer/web/the-pbr-guide-part-1>

<https://www.adobe.com/products/substance3d/discover/3d-in-games.html>

<https://www.airwiggles.com/c/general/ardour-as-a-daw-for-sound-designers-compared-to-reaper>

https://www.reddit.com/r/Adobe/comments/si673z/decent_open_source_alternatives_to_adobe_creative/

https://www.reddit.com/r/dotnet/comments/1cwnvak/visual_studio_vs_rider/

https://www.reddit.com/r/GameAudio/comments/17plyct/wwise_vs_fmod/

https://www.reddit.com/r/GameAudio/comments/1cs7z2n/why_is_reaper_highly_recommended_for_game/

https://www.reddit.com/r/GameAudio/comments/m9xcx0/fmod_or_wwise/

https://www.reddit.com/r/GameAudio/comments/v15j7g/which_is_the_industry_standard_daw_for_game_audio/

https://www.reddit.com/r/gamedev/comments/gfpszk/substancePainter_vs_Quixel_Mixer_2020/

https://www.reddit.com/r/gamedev/comments/mrecf8/whats_the_difference_between_substancePainter/

https://www.reddit.com/r/gameenginedevs/comments/1k18r6t/my_powerful_blender_workflow_for_gamedev/

Analysis Report

https://www.reddit.com/r/opensource/comments/1hk01hf/why_is_adobe_still_making_profits_on_expensive/

https://www.reddit.com/r/Reaper/comments/q7j152/sound_designers_how_do_you_manage_your_reaper/

https://www.reddit.com/r/Unity3D/comments/10l04p3/why_would_you_use_rider_over_visual_studio_for/

https://www.reddit.com/r/unrealengine/comments/17c3saf/is_there_any_difference_in_using_github_vs/

https://www.reddit.com/r/ZBrush/comments/1dab0tt/game_asset_workflow_with_blender_zbrush_and/

<https://www.youtube.com/watch?v=oafQqMg5hco>

<https://www.youtube.com/watch?v=aMgB018o71U>

<https://www.youtube.com/watch?v=aOGUUkw2Lxc>

<https://www.youtube.com/watch?v=T7xxZElC4ww>

Cominted Labs. (n.d.). How to find funding for a video game – The ultimate guide - Cominted Labs. <https://www.comintedlabs.io/news/how-to-find-funding-for-a-video-game>

Screenwest (Australia) Ltd. (n.d.). Digital Games Production Fund Guidelines. <https://www.screenwest.com.au/wp-content/uploads/2022/12/221205-Digital-Games-Production-Fund-Guidelines.pdf>

Yakese. (2025, August 26). How to get mobile game industry government support? Ofisus Consulting. <https://ofisus.com/mobil-oyun-sektoru-devlet-destekleri-nasil-alinir/>

<https://www.intuit-research.com/wp-content/uploads/2020/05/APAC-Gamers-Study-for-Website.pdf>

Newzoo (Global Games Market Report)

Unity Gaming Report