



UBISOFT INDIA STUDIOS

INTERNSHIP REPORT



Right at the beginning of my internship, on the first day, I was asked to sign a Non Disclosure Agreement.

Since the game I have worked on is not announced yet, the IP and project data will not be suggested or exposed by any circumstances, meaning:

- There will be no documentation related to the game's development
- There will be no screenshot of the engines or any related internal softwares
- There will be no display of internal communication or management records
- There will be no mention of game features related to its IP
- The content exposed will solely be related to the development process, not the project itself

This internship report must not get into other hands than Rubika's educational team. I am sincerely sorry for not entering into further details concerning my work.

Thank you for your understanding.



CHAPTER 1 – Presenting the company

Page 5

CHAPTER 2 – Getting into the project

Page 8

CHAPTER 3 – Designing an AI

Page 14

CHAPTER 4 – (Re)designing controls

Page 21

CHAPTER 5 – Extra work

Page 26

CHAPTER 6 – Taking a step back

Page 28





During my 4th year at Supinfogame – Rubika as a Game Design & Management student, I found several opportunities for my summer internship.

My past experiences included a Game Design internship at Gameloft Vietnam, very formative, but not aligned with my **career goals of working on console AAA games**. And last summer, I was a Game Design intern at Ubisoft in Pune. It showed me a vision of the AAA development, which is the industry I want to be part of. Unfortunately, the duration was too short and prevented me from truly understanding how big projects are made.

I also knew that because I had already worked at Ubisoft India Studios last year, the production team would trust me more than any other company, meaning I could get higher responsibilities that if I just was a new intern. I knew I could also push to work on fields I'm really interested in, which are the **3C and AI** in my case.

Furthermore, seeing **how a project evolved in a year** is something I was very curious about:

- How did the different teams and their members change?
- What is the current state of the project?
- Did the creative direction change by any sort?

Finally, it would be a lie to not say that I enjoyed my time in India last summer, as I fell in love with the culture and made a lot of true friends.

The choice of going back to Pune was therefore a very wise option for my 4th year internship, both from a personal and a professional point of view. I directly contacted the HR Director, submitted my application once more, and got an offer as Game Design intern.



CHAPTER 1

PRESENTING THE COMPANY

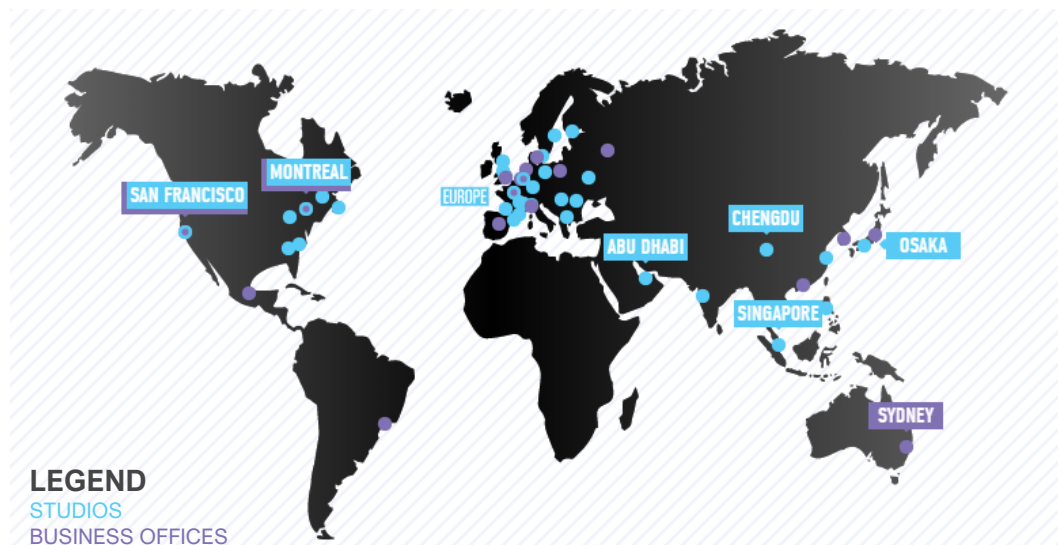


Ubisoft is a French game developer and publisher, founded in 1986 by the Guillemot family.

Over the last decades, it expanded its studios in Europe, North America, Asia as well as some business offices mainly for communication and marketing in South America and Oceania.

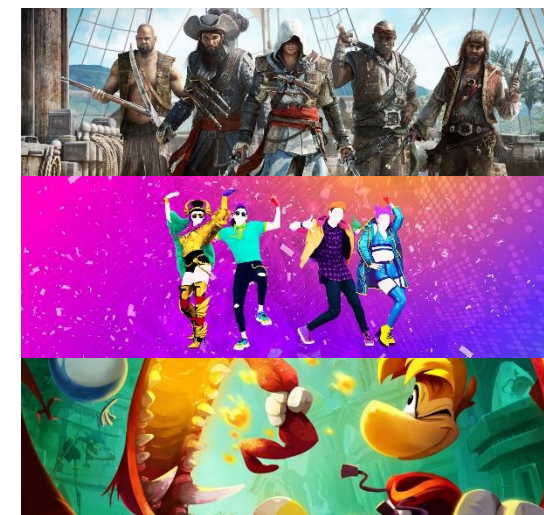
The company's portfolio is very diverse, ranging from AAAs to mobile games, to educational games, to even movies, making it the 4th largest video game company in the world. The company has now more than 14,000 employees, spread around over 40 studios.

Ubisoft around the World



Among Ubisoft's IPs:

- Assassin's Creed
- Just Dance
- Far Cry
- The Tom Clancy's games
- Rayman
- Watch Dogs
- The Crew
- Steep
- So many more...



But you must already be aware of this, the interesting part is on the next page



It all started back in 2008, when Ubisoft acquired a studio from Gameloft in Pune, the second biggest city in the state of Maharashtra. Pune's population is over 3 million, but it is also 3 times the size of Paris. Its economy revolves around manufacture (vehicles), education (more than 40% of the foreign students of the country), and IT, which is why acquiring a studio there seemed like a good strategy to make a first step in the sub-continent.

Ubisoft India started with 35 people. Jean-Phillipe Pieuchot became its managing director. Over the years, the company worked on titles such as Just Dance, Far Cry 5, and Steep.



Jean-Phillipe Pieuchot
Managing Director

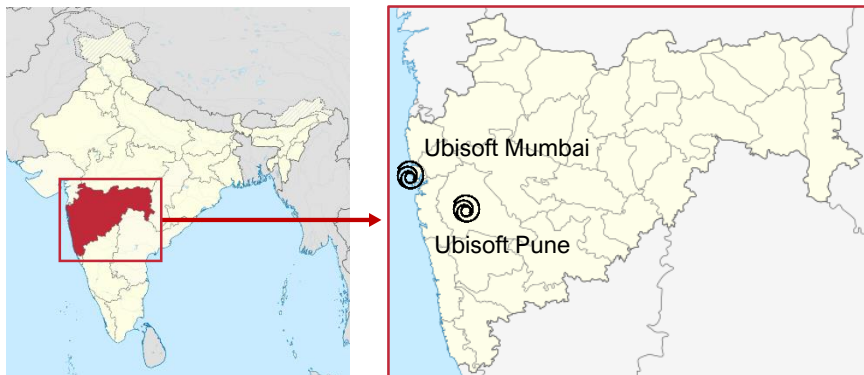
Ubisoft Pune (> 1000 people)



Main activities:

- Console QA testing
- Porting
- Co-development (with other studios around the globe)
- Mobile development and QA testing
- **Console development (including my project)**

Maharashtra in India



Mumbai – Pune takes from 3 to 5 hours by car, depending on the traffic!

Ubisoft Mumbai (> 120 people)



The office opened during the summer 2018. Its goal is to make the company attractive thanks to its location in Mumbai (formerly known as Bombay). It is currently growing up and its objective is to develop AAA games made in India only.

Main activities:

- Console development (also including my project)



CHAPTER 2

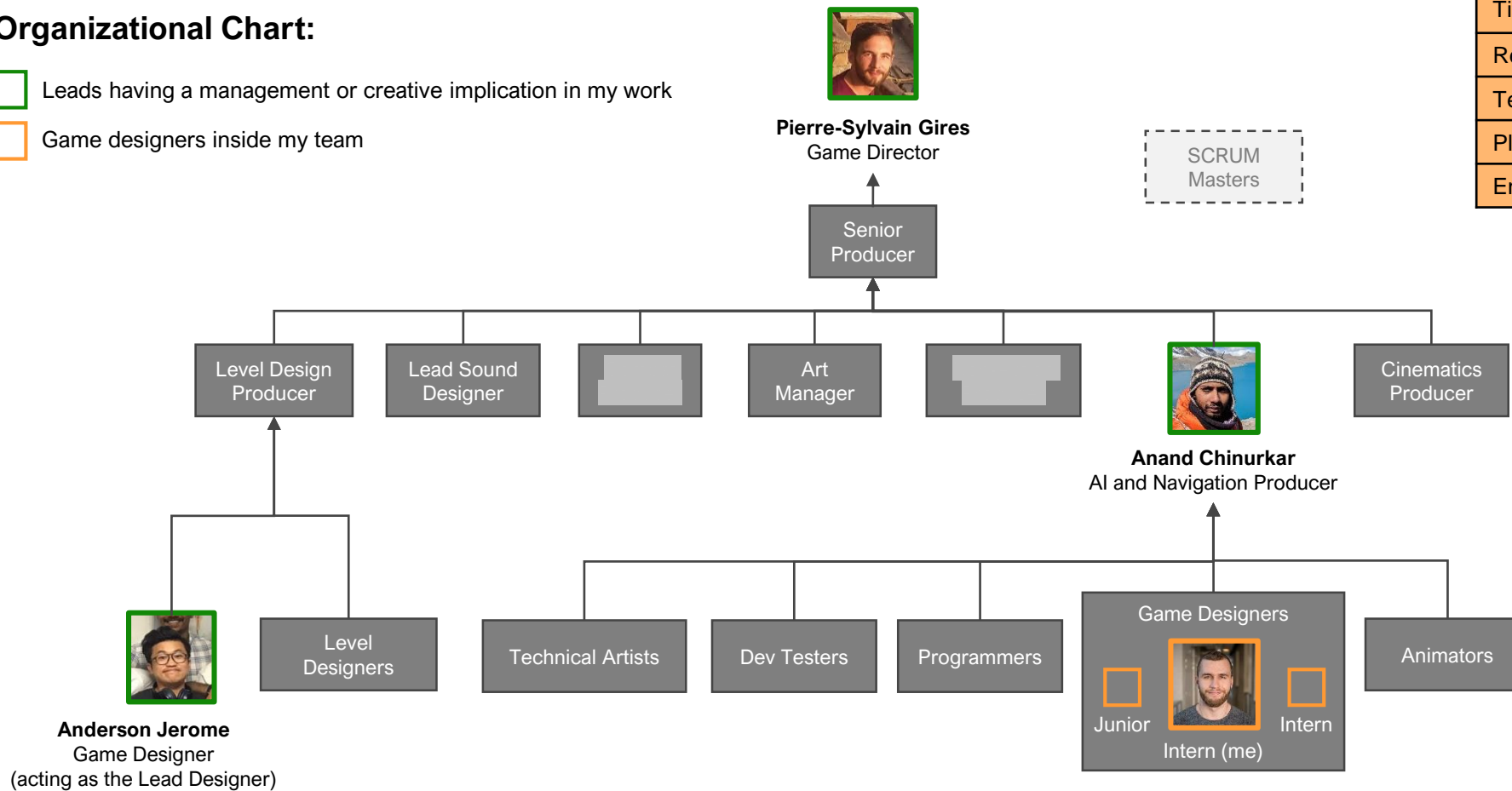
GETTING INTO THE PROJECT



The project started at the beginning of 2018, a few months before I joined Ubisoft Pune during my 3rd year internship.

Organizational Chart:

- Leads having a management or creative implication in my work
- Game designers inside my team



SCRUM Masters

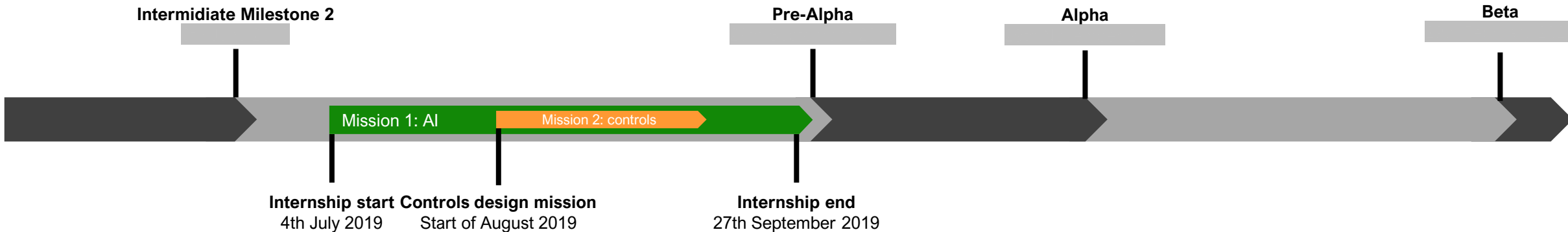
Title	
Release date	
Team	Over 100 people
Platform	
Engine	Anvil



Last year, Pierre was the Art Producer, and there was no vision owner behind the project. Since then, Pierre became the Game Director!



Timeline:



When I ended my Game Design internship in 2018, the project had one level graphically done as a target render, no character was final, and the design was at its early stage. When I arrived this year, all levels were in production, but at different states, and the design, although lacking some aspects such as the AI and minor features, had far more advanced. In other words **the FPP milestone had already passed** and the creative direction was set.

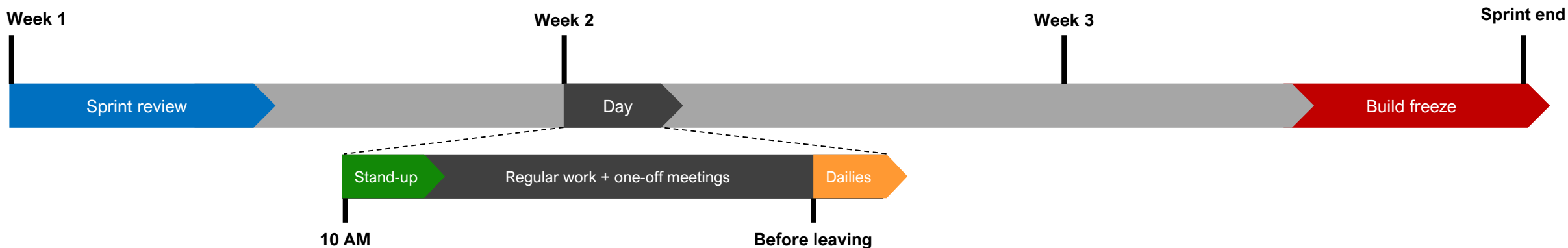
[Redacted text]

1. Because the team knew I had already worked on the project, I was quickly assigned to the task of **designing a whole ally AI which is present in a huge part of the game (mission 1)**
2. One month after I joined, I was also asked to **propose new control schemes for the game (mission 2)**

At first, my manager was the Level Design producer, but I quickly shifted to the AI & Navigation team, counting over 10 people, managed by Anand Chinurkar, my mentor.



Sprint:



Like most Ubisoft developers, our team used a process inspired by the **SCRUM methodology**, with the help of SCRUM Masters. Sprints last 3 weeks and during the first 3 or 4 days, there is the Sprint review: Pierre and other leads from Mumbai, such as Vincent, the Art Director, come to Pune to play the build and check if all stories are done.

A user story represents a requirement, assigned by a producer to someone from his team, with a specific priority and story points, which represent the overall effort needed to finish it. Once all user stories have been reviewed, there is a big call with all the teams of Pune and Mumbai with a report of how the story points went during last sprint. Once over, new stories (or previous stories not done) are assigned to every team member. I personally felt like Pierre had too much work on his shoulders, since he had to both carry the vision and validate every story. Taking him off the task of validating story would give him more time to check what the teams are doing from a creative perspective and give feedbacks.

L1	L2	L3	L4
Functional	Fully integrated	Polished	Highly polished

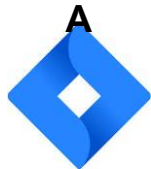
We also had a rating system to help us estimate the level of finish of required for specific user stories.

Starting Wednesday of last week, there is a build freeze, meaning the only modifications that will be pushed for the build are debugs. It is why, if I have a story related to the build, I had to push it before the Wednesday of the 3rd week.

Every day, the team gathered in a **stand-up meeting** where every member tells what he has done yesterday and what he will do today, in a very short amount of time. Before leaving (after at least 9 hours of work), we also had to write a mail listing what we had done during the day, called « dailies ». I felt like this system was extremely useful to detect if there are any stories blocking stories from other people, without taking too much time over the day.



JIRA



Linked to the engine and the documentation, JIRA was a very powerful to apply the SCRUM methodology.

My user stories were either in the backlog or in a Kanban Board (To do, Doing, Done) cleaned at the start of every sprint.

It was my job to prioritize my tasks, estimate the time it's going to take (not the amount of story points), and

Something very convenient about JIRA is that I could create my own sub-tasks for my user stories.

Micro-managing the user stories by myself (except creating them and estimating the story points) was quiet efficient to make me more responsible and precise in my own work.

Outlook



Mailing software used for **indirect communication**. Very classic and efficient, I mostly used it to ask a feature to be dev-tested before being implemented, write dailies, discuss about a design issue concerning too many team members, and get updates about the engines, such as crashing versions and build freezes.

Outlook was also used to make proper demands to the HR team or the admins.

I also used to receive a lot of mails from the internal communication of Ubisoft India Studios and Ubisoft World.

Furthermore, outlook was also used to book rooms to organize meetings if needed.

Microsoft Teams



Replacing Skype, most of the team **direct communication** was done on Microsoft Teams, through single conversations or groups, online calls, and screen shares if necessary.

I personally found it far more efficient than skype, because of all its options for conversations, event creation, tagging system, and file sharing system.

Skype

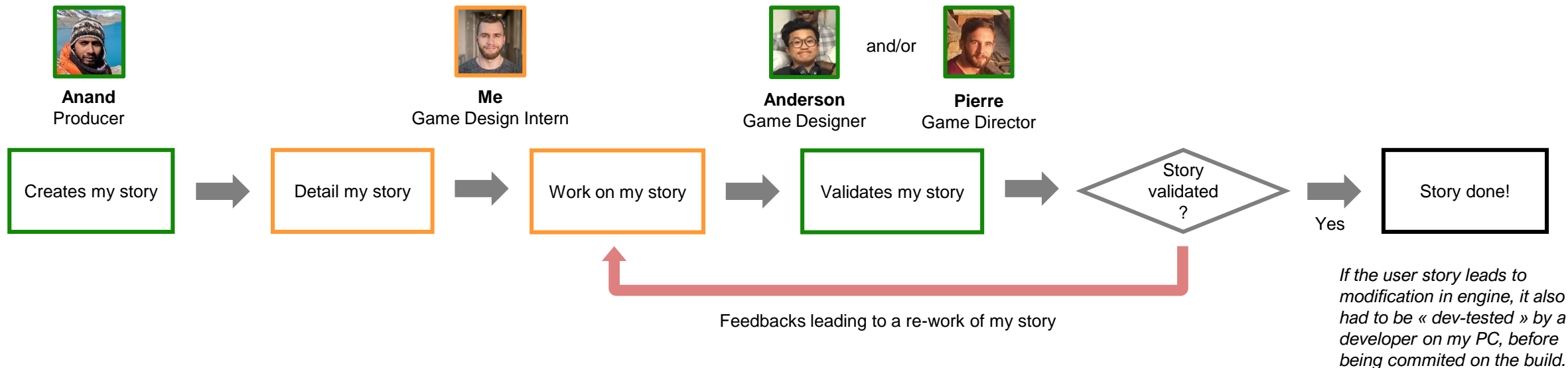


During my 2018, the whole company was using Skype for Business. Now, most of the employees use Teams but some are still transitioning toward it.

It is why Skype was still used for direct communication with the HR team, the admins, or the IT if they need more details concerning a request I sent through a ticket.



User story process:



When I arrived, Anderson looked at what design work had not been done regarding non-combat, **since I could not shift toward the combat team because the training in engine would have taken too much time** for a 3 month internship. Among the designs pending, the most obvious one was AI design, which I was very happy to work on since it's one of my main interests. Once assigned to the AI responsibility, I quickly moved to Anand's team, and all my stories were planned according my time remaining at the company.

Anand was very flexible about my stories, meaning he could create them during a sprint or post-pone them for the next one if necessary (and still fitting the planning). This was extremely helpful, since sometimes I encountered grey areas that weren't designed yet, but that I was dependent on.

On the other hand, it happened that **Anderson and Pierre hadn't the same vision over features I was responsible for**. At first, it was confusing, but in the long run very useful, since Anderson was focused on identifying design problems while Pierre had a more global vision and a clear idea of what had to be done for the following milestones. In all cases, they ended up taking a common and rational decision.



CHAPTER 3

DESIGNING AN AI





Mission:

As said earlier, my first and main task was to design an AI, but not any kind of AI: **my mission was to design an ally AI** (except its combat behavior). This AI helps the hero throughout the story and has a special narrative link with the hero that needs to be felt through the behavior as well.

Therefore, I quickly divided the AI into smaller aspects, to make user story creation and validation easier while following the same logic as programmers:

1. AI's actions metrics
2. All types of behaviors and their sub-behaviors
3. Idle system
4. Reaction system
5. Death conditions (including its death camera design)
6. Animation and mood list

People I worked with:

From my team

- Poonam and Jayanth (programmers), implementing the AI in engine
- Chandra (game designer) his work on the hero's metrics was my reference for the AI's metrics
- Naresh (technical artist) blending animations, moods and making them ready for Anvil integration

From other teams

- Hrishikesh (sound designer) helping me technically and creatively with dialogues
- Muadh (level designer) making sure the LD ingredients fit the AI's metrics
- Alan (level designer) integrating the AI in a level as a test
- Raaghav (camera designer) prototyping the AI's death camera
- Dipayan (combat designer) connecting the AI's combat behavior to its whole design

Softwares used:



Engine used for some of Ubisoft's open-world games with third person view (Assassin's Creed, Ghost Recon, Steep, etc...)



Diagramming software, very handy to create quick flowcharts such as decision trees for AIs



Wiki software, used to hold all of the project's documentation. The writing editor is quiet simple and has many layout options and plug-ins such as video integration.



Mainly used to explain AI's position behaviors in top-view, or whenever I felt using visio for the same type of diagram would have taken me more time.



Animation blending software

Internal software directly plugged in Anvil. Animations from this software can directly be triggered from Anvil.



Dialogue editing software

Internal software used to stock in-game dialogues, upload the recordings, and create codenames so that they can directly be triggered from Anvil.



Regarding my mission on the AI, **about 90% of my work was documentation** as Poonam and Jayanth were the two programmers in charge of her AI. My goal was therefore to make the documents as clear as possible since they were using them to prototype the AI. All variables, animations, sounds, fx, dialogues, etc... were always mentioned with their specific code names. I was the one going after the other team members such as sound designers and technical artists to make sure programmers had all necessary data just from my documents.

Making them clear enough for programmers also helped explain better the AI to Anderson and increase my user stories from being validated.

Action Metrics:

Designing the AI's metrics was not that easy, but very tedious, since a lot of documentation has to be made to explain a feature in its smallest details. But mentioning them was important as they were used in any kind of behavior.

Unlike the playable character's metrics, something very convenient about the AI is that **all movements are predicted**. Therefore, all values usually never vary and the amount of animation transitions are much lower.

For example, if the playable character is sprinting, but then the player decides to suddenly make it stop, there has to be all kinds of conditions and extra transitions for it. With AI, this never happens as everything is planned and predicted.

Those documents were mainly composed of step by step diagrams of the AI's actions, with a sheet listing all distances and time variables, a sheet listing all parameters that will be tweaked in engine, animations, sounds, and FX.

Behaviors:

Dividing the AI into different behaviors was the first step in the AI's design. Different behaviors allowed my AI to be far more flexible and make sure some of her specific actions were not triggered.

But making defining those different behaviors was easy, **the hardest part was defining the conditions to trigger them, and the priority among them**, as it had to be both logical from a design perspective and easy for the programmers to make.

All behaviors had different intentions and a variety of possible actions inside of them. Those intentions were set at the very beginning and their objective was to make the AI's behavior align with the intentions of the story.

I often felt it was hard to define whether what I designed was matching with the story, until I remembered an exam for my 4th year about ludonarrative dissonance (conflict between narrative told through the story and through gameplay), which helped me make sure any action done by the AI was coherent with its personality.



Idle system:

The Idle was my last task and probably one of the hardest one to design, since the objective set to design the AI's idle wasn't to just trigger a static animation when it's not in movement, but make the character feel more alive.

I managed to do so by creating several types of idles, that like behaviors were triggered under certain conditions and had a priority list.

Reaction idles

Contextual animations, triggered when certain conditions are reached, for example, if the character is tired.

Object idles

Contextual animations, making the character interact with an object from the environment.

Progressive idles

Non-contextual animations, that will vary throughout the story to show the relation between the hero and its ally.

Working on the idle was very interesting since it did not influence the player's progression at all, but rather just added more depth to the character and the universe.

Reaction system:

Reactions are mostly dialogues, but they can also be animations and movements as well. They can be triggered in any behavior except the combat behavior which has specific combat reactions.

They are triggered whenever the player character performs an action with specific conditions and have two objectives:

- Guide the player through dialogues
- Add in-depth to the AI

Once the system to trigger them was designed (which I'm not able to detail here), I had to create some of them with a template that was both easy to read by programmers, which I often consulted to know if any detail was missing, as well as Chandra, the game designer who was going to carry on my work once I leave the company.

Something that was very confusing about the reactions is that all dialogues had previously been recorded and I had to design reactions based on those dialogues, not the opposite!



Death camera:

Fortunately for me, most of my AI's behavior was scripted and it was not supposed to die in the story, nor in non-combat sequences.

But to my surprise, 1 month after I started, Raaghav, a level designer I knew from my 2018 internship, who was now working on all the game's gameplay cameras, sent me a message: what is supposed to happen with the camera when the AI dies? So I went straight toward the combat team to ask them who is responsible for this task, since the AI only dies during combat or if killed by the player (which is a reaction I had already planned on my side).

The problem is **they had not started designing the combat behavior of my AI** since they were focusing on enemy AIs which had a higher priority, but I could not let Raaghav stuck with his story, so I quickly talked to Anand and the story of designing a death camera for my AI was added.

I had to take a quick decision regarding it, propose a camera behavior and document it. My idea was basically to make the camera transition from the combat camera which is a camera above the playable character, to a shoulder camera showing the back of the playable character and my AI as the point of focus.

Combat camera

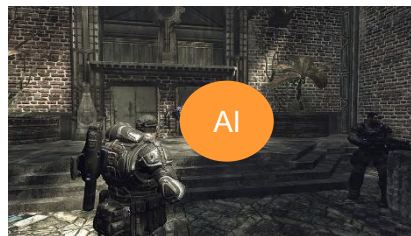


Source: Batman: Arkham City

Transitions to



AI ally death camera



Source: Gears of War 2

We both knew that such a movement could bring collision problems and obstacles on screen hiding the AI from afar. Unfortunately, I left the production before Raaghav had time to prototype it.

Animation and mood list:

Another responsibility I was surprised of, is the animation and mood list. Indeed, I had, as a game design intern, to define what were all of the character's animations (which is a job I thought someone of the art department would do). But it came up to make a lot of sense since I designed the AI's behavior by myself.

Animations from [redacted] were used as inspirations and placeholders, and the goal will be to use them in the final game as less as possible.

Moods [redacted] It's a technology creating facial expressions during specific animations or dialogues. One of my job was also to choose which mood (among a long mood list with screenshots of my previewed character's face) was going to be triggered at what moment.

Same as for the animations or the Idle design, I was intrigued by the fact they asked a game designer to do this instead of an artist, but I then understood how Ubisoft sees designers:

The game designer does not choose how an asset will be done, but the game designer chooses what asset has to be done, according to its design.

« I choose the function, but not the form »



Regarding my mission on the AI, **about 10% of my work was in engine** mainly for 2 tasks:

1. Following up the work of the programmers on my AI
2. Selecting reactions for my AI's reaction pack

Following up the work of programmers:

The AI was prototyped by Poonam and Jayanth, two programmers from my team. Their stories were created by Anand, my producer, and were based on how I had divided the AI's behavior through the different documents in confluence. It is why dividing the AI's features logically was important: a page was either a new story or a sub-task of a story.

While I was working either on controls or other aspects of the AI, programmers were working on validated documents I had previously made. **Whenever they encountered a design problem or a question, they would call me.** I quickly created a Microsoft Teams conversation with them, Naresh (technical artist), and Anand (producer, just observing) to make the communication easier.

I often came at their desks to directly test the AI in engine. It sometimes happened that features were lacking an aspect I had in mind, but I usually quickly realized my documents were not clear enough. It is why I often updated my game design documents with further technical information or edge case solutions they had pointed out.

They also used to call me if they had technical problems depending or any other member of the team. It was also my job to go to the sound designer if dialogue codes were missing or toward Naresh if an animation had to be explained.

Selecting reactions:

At the request of programmers, my second implication in the AI in engine was the selection of reactions for the AI's reaction pack.

Indeed, the project used Anvil's version [REDACTED], and kept all its database as well. Inside of this database, programmers found reaction packs, which are reactions made for a specific group of NPC.

Every reaction is composed of a danger event. A danger event is an event with a specific set of systemic conditions. If all those conditions are true, the danger event is called and triggers a reaction

I had to go through all the danger events (over 100) in engine, read their description, and judge if they could be useful for a reaction I had designed. Because this would make the programmers win some time, since they use the reaction system but don't have to create a danger event by themselves.

I also pre-selected danger events that I thought could be useful for the AI during combat, documented the list and showed it to Dipayan from the combat team.



My mission of designing an AI, almost from scratch, was my main mission on this project, and the one I spent the most time on. I learned several things to keep in mind when designing an AI, or designing in general:

- **Small intentions help**

Even though the game's intentions have been decided, they are usually not precise enough. It is why at the beginning of every document, writing small intentions, which are specific to the feature, but still match the general intentions, might be a great help in case a design choice has to be made.

- **What makes up a good Game Design Document**

A good Game Design Document is a document which is clear, being precise enough when presenting the system so that programmers have no questions to ask.

- **Working on the AI is a great role if you like communication**

So many people are involved: programmers, technical artist, animators, sound designers, other game designers, level designers, etc... This forces you to clearly communicate and solve a lot of dependency problems, which is something I love to do.

- **Calm and patience are a great tool when debating about a feature design**

It happened once that another game designer was really unhappy about one of my decisions. He had very strong arguments and a willingness to really prove he was right, but instead of flaring up, I remained calm. Staying rational, justifying your choices and not letting a bigger place for your ego is therefore extremely useful in all situations.

- **Game designers choose the function, artists choose the form**

It is one of the many visions of what should be a designer, and it is quite efficient in a AAA project, since it prevents the game designer to have extra dependencies (being dependent on other people's tasks) on his user stories.

- **The AI can be used to tell a story**

The story is not only told through dialogues or cinematics. Even as an AI designer, you can somehow contribute to the story by developing your AI and making sure its actions match with its character's personality, whether the action helps the player or just showcases the AI, to create a small hour of glory for it (which is an advice I received from Ryan Pergent, who made an intensive week about AI in 4th year).



CHAPTER 4

(RE)DESIGNING CONTROLS



Mission:

1 month after my internship started and I was fully working on the AI, I received a message from Anderson, who asked if I was interested by controls design. The 3C being my other main field of interested with AI, I was very excited and said a big « yes ».

He gave me the **mission to propose new control schemes for the project's controller**, as the former one received a negative feedbacks from the HQ:

1. 2 combat actions are mapped on the A input
2. More than 4 actions were are mapped on another input, but contextual conditions do not work since some of those actions are used in the same context (combat)
3. The current control scheme is not « modern » enough

People I worked with:

From my team

- Chandra (game designer) giving feedbacks for non-combat controls and ergonomics
- Puru (programmer) programming some actions for the playable character
- Jayanth (programmer) programming some actions for the playable character

From other teams

- Jayesh (3C programmer) helping me prototyping the controls in engine
- Shobith (combat designer) giving feedbacks for combat controls
- David (technical combat designer) helping me prototyping the combat controls

Softwares used:



Anvil

Engine used for some of Ubisoft's open-world games with third person view (Assassin's Creed, Ghost Recon, Steep, etc...)



Confluence

Wiki software, used to hold all of the project's documentation. The writing editor is quiet simple and has many layout options and plug-ins such as video integration.



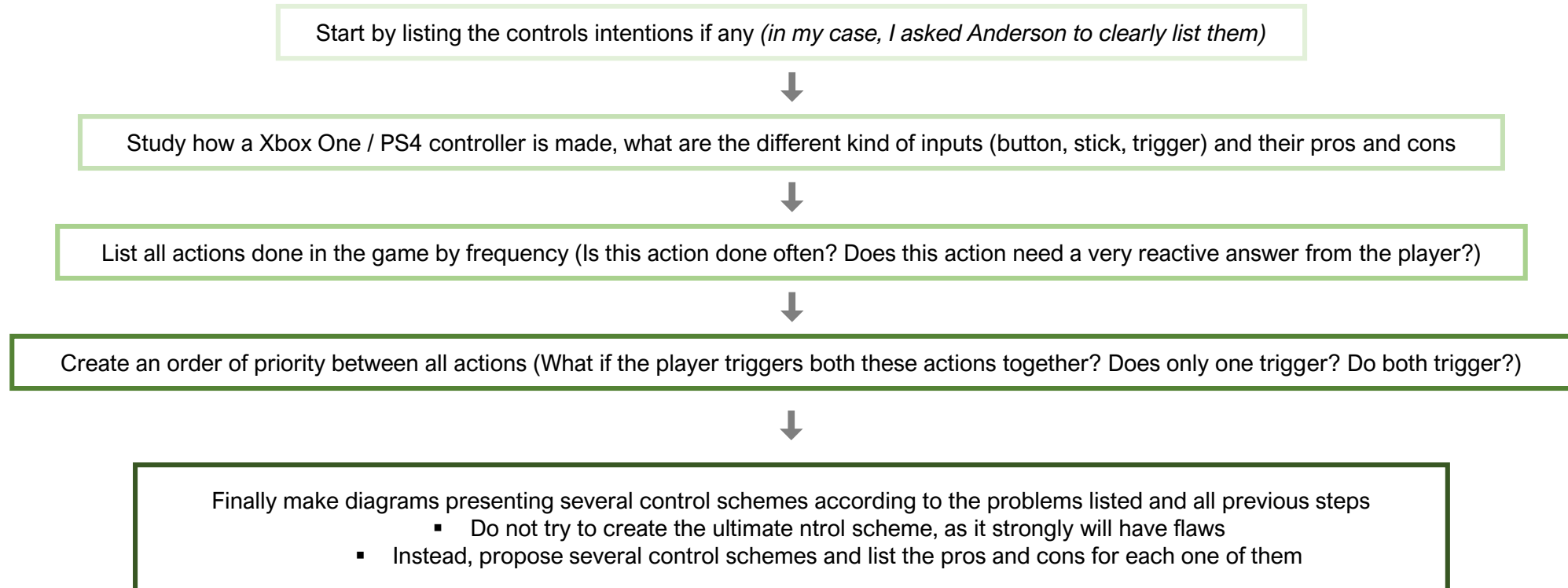
Powerpoint

Mainly used to create control schemes diagrams, used as references before prototyping



Regarding my mission on the controls, **about 50% of my work was documentation** as a lot of research had to be done on control design and Anderson wanted a variety of control schemes. Furthermore, I was not guided that much on this task and when I asked, the other designers hadn't a clear idea of what is the process, step by step, to design a controller.

During the first two weeks, I proposed all kinds of control schemes, brainstorming with Shobith, the control designer, and showing them to Anderson, without a lot of success. I also received feedbacks on the ergonomics by Chandra and some useful Gamasutra articles, but still not enough to know a design process. **I therefore ended up creating a design process for controls myself:**





Unlike the AI design, **the remaining 50% of my work was done in engine,**

This is mainly due to the fact that once I had proposed control schemes, Anderson wanted the modifications to be made in engine, but no programmer was available to remap the controls. But Shobith, with whom I had previously worked on several control schemes propositions, introduced me to Jayesh from the non-action team. Jayesh was responsible for the camera programming (the camera designer was Raaghav) and the control programming. He even created a debug button making testers able to switch between several control schemes.

The question now is, how do we implement control schemes in engine? The answer is that it's done through submenus and drop-down menus! Indeed, I had no authorization nor need to code in ZEN, which is an internal language, mixing both C# and C++ and making it able to make modifications in runtime. Instead, Jayesh showed me how to use the game controls manager, which is a submenu, to remap inputs, which was the first step of my work.

When fully understanding how to use the game controls manager, I wrote a tutorial, in case another game designer wants to remap inputs in the future, and I started implementing the control schemes validated by Anderson. It was easy and tedious, but **I also encountered many naming problems** because some actions were not named as actions, but as inputs themselves, some actions names were actually used for another action in the game, and some actions were inherited from [REDACTED] and had not been deleted yet.

Once the schemes were implemented, first Anderson, then Pierre, tried them and validated one, which was then dev-tested and directly pushed on the master build.

But remapping controls is just one aspect of controls design, I also had many propositions to upgrade many actions and trigger conditions. The problem is that those conditions were not all programmed by the same person. For example, Puru was a programmer responsible for some navigation actions, Jayanth was a programmer responsible for some other navigations actions, on the other hand combat designers (not programmers) were responsible for the combat actions. To solve this, I wrote a big mail and listed down very clearly what every member had to do on his side in order to enhance the controls.

Finally, some actions could not be remapped as they were hard-coded and not present in the game controls manager, which is also something I pointed out in the mail and that Jayesh solved on his side. I nevertheless felt very frustrated of this organization, since **it was not one person who was responsible for the controls in engine, but many, from different teams, with other priorities.**



My mission of redesigning controls, was my second main mission on this project. I approximately spent 5 weeks on it, and learned several things to keep in mind when designing future controls:

- **Everybody has an opinion over controls**

Designing controls seems like an easy task on the surface, but it was actually without a doubt the story I had the most trouble with. Mainly because I was easily carried away by people's feedbacks and opinions. Because unlike AI design, for example, controls are something very subjective, and everybody has its own vision of ergonomics. I spent a lot of time trying to make concessions to make everybody happy, but it's really not the way it should work. Once I started sticking to intentions, and only taking into account problems, button frequency, and action priorities, making decisions became much easier.

- **Do not try to find a unique and perfect solution**

Whatever you do, a control scheme will always have flaws and edge cases that make it anti-ergonomic. It is why I highly advice anyone designing controls from now on, to propose several control schemes with different justifications. The important part is that you have to list the pros and the cons of every control scheme to help the lead designer and the creative director judge whether the game has to be remapped this way or another.

- **Putting two actions on different inputs is not always the solution**

The first problem was that two very important combat actions were mapped on the same input. At first, both the HQ and Pierre wanted to

Combat inputs were directly assigned to every combat animation and for every transition, meaning David had to repeat the same modification in engine for more than 100 times (counting all combat animations)



CHAPTER 5

EXTRA WORK





Dialogues point of contact:

I was clearly not assigned to this responsibility, but because during my 3rd year internship, I had listed all dialogues and that I knew all files and the way the worked since I needed them for the reaction system of my AI, I served as a dialogues point of contact.

For example, some of the level designers were asking me where they could find specific scripted dialogues scripted to their level. Or my opinion was also asked regarding cutscenes dialogue, whenever my AI appears in a cutscene or a cinematic.

This is a responsibility which did not make me learn that many things, but I was happy I could contribute to the project in other ways than AI Design nor Controls design, and prevent people from wasting their time, since I could answer them very quickly.

Representing other interns:

When other interns from Rubika had an issue, they were asking me first how to solve it and who from the HR team or the administration team they had to contact, since I knew most of the people, due to my past experience in the company.

I also weekly held meetings with admins, who were helping us manage our house who had a lot of problems (running out of water, electricity, flood rats, etc...)

Finally, I invested a lot of time in helping other Rubika interns getting their Indian VISA ready, since around ten documents from different sources are needed, and the organization in Paris responsible for the VISA admission is very strict.

I was very happy to help other interns settle in India and the company in general, since when I arrived last year, I met a lot of problems and nobody else was an intern with the same situation as I was.

Game and Level Design workshops:

Something I had already done during my 2018 internship: workshops with designers from other teams in the Pune studio!

Organized by Prashant, a Level Designer from the Steep team, every workshop was covering a specific subject while exposing the rational game design and rational level design process used all over Ubisoft.

Those workshops were organized after the working days and could last up to 3 hours. I took notes of things I learned during those theoretical workshops, which I then used when working on my project, or that I think could be useful for my 5th year's project or my future career.

Unfortunately, those workshops were organized only 2 weeks before the end of my internship. I didn't cover as many subjects as I wanted.



CHAPTER 6

TAKING A STEP BACK



I cannot think of any real downside after completing this internship. Before starting it, I thought I had a sharp definition of what I wanted to do. Now that it is over, the definition of what I want to do is even sharper!

Working in a big company, on a AAA project:

With my past experiences, I already knew I didn't want to work on a mobile project. I also was really attracted by AAA development. **I am now sure I want to be part of this industry.** I loved the fact that I had to communicate with a lot of people, that the production and responsibilities were very organized, and that people were highly specialized. I also was afraid that because you're assigned to one task, you will only do this task, but at least at Ubisoft, I felt like if you have the opportunity and time to work on something you like, you can go for it and push for it! In my case, I'm very glad I didn't stop at control schemes propositions, but also prototyped them in engine and finally pushed the best version to be played by Serge, the Content Director.

My role in this industry:

I'm really interested in two aspects of Game Design: the AI (which I still love doing) and the 3C. But with this internship, I realized that **I am rather attracted to the controls design than the camera design.** I usually was the one designing the controls in my past student projects, and I also created a few alternative controllers. I therefore think pursuing my career in the field of control design could be a nice fit.

Furthermore, I spent more time this year looking at how the project evolved and what is the role of a lead designer and a producer. Even though my career goal would rather to become a lead designer, I understood many new aspects of the SCRUM method and what are leads real roles. Therefore, **becoming a lead designer after years of experiences is definitely an option I'm looking up to.**

I can do both documentation and prototyping:

Being a versatile is one of the most important qualities for a game designer. Even if at school, we became quiet technical last year (which is something I don't despise at all), I understood many things about how real game design documents should be made for production. At first, I was also quite afraid to touch to Anvil, but like any other engine, it's just a tool that needs some adaptation!

Strong career opportunity for after my 5th year:

I really enjoyed my role and felt like I could make a difference by working at Ubisoft India Studios. **Going back there again, but this time, as a regular employee, is therefore an option which I'm currently studying.**



Even though I felt very happy by working at Ubisoft India Studios for the biggest part of my internship, there are a couple of aspects in my work I would like to increase, after having gotten feedbacks from Anand and Anderson:

Prioritizing tasks:

At the end of my internship, I was satisfied with the amount of work I had done, but I felt I could have been more productive if I had better managed the priority among my responsibilities. For example, designing controls without really knowing the process made me lose a lot of time that I could have used to design the AI's idle system sooner and not work as much on the last week.

Becoming more rational:

One of the feedbacks I got was that at first, I took design decisions too much based on what I felt was right or wrong, without strong justifications. Fortunately, this is something I've been working on and as the time goes by, I felt becoming more rational helped me a lot through my choices.

Titled as an intern, working as a junior:

I loved getting a lot of tasks right from the beginning! That way, I felt highly motivated, but I also felt I sometimes had a lot of pressure, since a lot of the team members were depending on my stories state.

Understanding my qualities:

Doing this internship helped me realize what I was good at, and what people felt I was good at. First of all, my designs and documents were quite **clear** and made so that programmers could work without asking themselves too much questions. Furthermore, I increased my **communication** skills, helping me . My **calm and patient behavior** in a production team as a game designer is also something that will prevent me from creating useless personal conflict inside my future company. All those qualities are I think necessary if I later on want to become a lead designer.



Thus, I completed my second internship as a Game Designer at Ubisoft India Studios, making me able to see how the same AAA project evolves from pre-production to production. I was very lucky to work on both the artificial intelligence and the controls, which are my favorite fields in game design.

I learned a lot, technically and theoretically, but also humanly, meaning how to maintain good and honest relations with other team members, which is a quality I will keep forever, bringing more flexibility in the management of my 5th year's project. One of my objectives was to contribute to the project with my knowledge gained during my studies, my projects, and my experiences, and I truly felt like I brought something and left my mark in this game.

Of course it was sometimes difficult to adapt to the country, respect the tasks I had aimed at, and fix my working flaws, but the friendly and open-minded environment quickly made me feel at home and surrounded by friends. I met very inspiring people and hope to be once more part of such a nice and trustworthy team.



People from different teams inside the company, dressed up for India's Independence Day!



SPECIAL THANKS

JEAN-PHILLIPE PIEUCHOT
SAMIRA CHABANI
VANITA IYER

For making this internship possible

BHARTI TAUNK
AAKRITI SINGH
For being so helpful

SHWETA KP
RITESH BADDAP
NEHA ANGORE
For offering me a home

SWAPAN SHAH
For such a warm welcome

ANAND CHINURKAR
For being an awesome manager

PIERRE-SYLVAIN GIRES
For his involvement in my work

ANDERSON JEROME
For being stubborn because he knows he is not wrong

AND THE WHOLE TEAM!
I love you all, but there is not that much space here...

LEO BERRET
WILLIAM BELLIM
BENOIT CASSIN
DAVID BOUREL
Good luck for this journey

DIPAYAN PAUL
GEORGE ZACH
MUAD SOLKAR
SANKET NIDAVANI
AAYUDH PARAB
For being amazing friends

MARTIN JACOB
For being an entertaining guy next to my desk

THANK YOU FOR READING!



पढ़ने के लिए धन्यवाद