Using AI-Supported Onboarding Systems in Video Games to Improve Player Experience

by

Lydia J. M. Choong

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2024

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

Lydia Choong was the sole author of all chapters of this thesis except Chapter 4. The contributions to this chapter are as follows:

- This research was conducted at the University of Waterloo by Lydia Choong under the co-supervision of Dr. Lennart Nacke and Dr. Jian Zhao.
- Lydia Choong was the sole contributor to the research questions, participant recruitment, study design, data collection, and quantitative analysis. For the qualitative analysis, Lydia Choong conducted all study sessions and transcribed all interview recordings.
- Lydia Choong was the primary coder and contributed to the coding of all interviews; Dr. Eugene Kukshinov and Dr. Sebastian Cmentowski each contributed to the coding of 55% of the interviews.
- Lydia Choong was the sole contributor to the qualitative analysis after coding was complete, and was the sole author of the chapter's manuscript.

As qualitative analysis is highly subjective, I chose to use an analysis method with multiple coders in an effort to reduce potential biases. I contributed the maximum amount of work possible for a single person within the design of the study.

Abstract

Video games face the challenge of providing onboarding that motivates new players to engage with a game beyond their initial experience. Interactive media inherently influences players' cognitive load during the learning process; video games must therefore determine a method of teaching new players game mechanics without exceeding their mental capacity for processing new information. Too much guidance can cause player frustration or boredom, while too little guidance can overwhelm. Instead of using restrictive onboarding methods, this thesis proposes that video games can use artificial intelligence systems that handle some in-game decisions to reduce new players' cognitive load. To demonstrate this concept I designed and evaluated *Joker*, a turn-based strategy game with an AI-supported onboarding system that suggests an action on the player's turn. I conducted a mixed-methods within-subjects study ($n = 20$) to examine the impact of AI-supported suggestions on new players' cognitive load and to better understand the relationship between AI-supported onboarding systems and player experience. Results indicate that AI-supported suggestions successfully reduce players' cognitive load, but that too low of a cognitive load negatively impacts players' ability to learn from the AI-supported suggestions. Players primarily learn through lived game experience, and they strongly value interaction, agency, and personalization during the onboarding process. Future implementations of AI in onboarding should therefore ensure that AI-supported onboarding methods maintain a player's ability to learn, and additionally use these dynamic systems to provide increased player control over the onboarding experience.

Acknowledgements

I would first like to thank my supervisors Dr. Lennart Nacke and Dr. Jian Zhao for their support over the course of my degree. I am honoured that you both made the decision to accept me as your student; I hope that my work proves to you that it was worth it.

I am very thankful to Dr. Will Zhao and Dr. Keiko Katsuragawa for agreeing to be my readers, and for providing me with their valuable feedback.

I would also like to thank Dr. Eugene Kukshinov and Dr. Sebastian Cmentowski for stepping up as coders and contributing to my qualitative analysis. I am eternally grateful for your help, and eternally apologetic for the long hours of discussion I put you through—the fact that we all survived is proof enough that anything is possible.

I am also grateful to the many wonderful people of the HCI Games Group: Joseph Tu, who stepped in to mentor me every term despite the mountains of responsibilities that never left his plate; Derrick Wang, who was always willing to provide me with wisdom or wheels or both, depending on the situation; Hilda Hadan, to whom I owe my eternal gratitude for trying to teach me—a known math un-understander—statistics; Dr. Reza Hadi Mogavi and his unending, infectious positivity (especially for food); and Sabrina Sgandurra, without whom I would have drowned in the impossible complexity of academia before this study could even set sail.

Additionally, I would like to thank the staff of the Games Institute for their support in running my study—specifically, for letting me monopolize the immersion lab for a month, for helping me troubleshoot all the technical difficulties that come with computers and fire alarms and power outages and printers, and for doing everything in their power to get my participants to show up.

I would also like to thank my family for their support over the past twenty months and more. I hope I made you proud.

Finally, I would like to thank the many friends who have been there for me on this journey. From my playtesters and emotional-supporters, to my behind-the-scenes spectators, rubber duck stand-ins, and more: I did it!

Dedication

For my big little brother.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

miniPXI Mini Player Experience Inventory 21, 22, 26, 33, 35, 60, 62, 65

NASA-TLX NASA Task Load Index 21, 65

PXI Player Experience Inventory 21

RTLX Raw Task Load Index 21, 22, 26, 33, 37, 58, 65

Chapter 1

Introduction

Video games are a multi-billion dollar pillar of the global entertainment industry. The appeal of interactive media draws millions of people to try new games each year. However, the interactive experience poses a unique challenge: players must learn how to interact with a game before they can play it. This initial learning period—also known as the *onboarding* process [4]—is a crucial part of new players' game experience. Tedious, uninspired, and frustrating onboarding methods are major reasons why players quit during their first experience with a game [12, 2]. Interactive media is also more likely to cause information overload, or excessive *cognitive load*, in learners than non-interactive media [31]. Onboarding must therefore strike a balance between benefits and drawbacks in the learning process: restrictive tutorials are thorough but frustrating, while "hands-off" guidance risks players becoming confused or overwhelmed.

Artificial intelligence (AI) has been used in video games for decades [9, 48]. Many games use AI to manage computer-controlled characters [50], create procedurally-generated worlds [47], or manage complex models of players' in-game behaviours [49]. However, research covering both AI and onboarding exhibits a significant knowledge gap regarding the impact of AI-supported onboarding methods on player experience; prior work has instead focused more on technical or novel implementations of AI-supported onboarding systems [22, 23, 51]. Existing literature also demonstrates a lack of qualitative studies on player preferences for AI during the onboarding experience. More research is therefore needed to properly investigate how game designers should use AI during the onboarding process, as well as to understand how AI-supported systems impact the onboarding experience.

In this thesis I present *Joker*, a turn-based strategy game with an AI-supported onboarding system that suggests an action to new players on their turn. Joker's core gameplay implements a high-complexity combination mechanic to overwhelm new players and increase their need for onboarding during their initial experience. I conducted a mixed-methods within-subjects study that used surveys and semi-structured interviews to compare player experiences between the base version (no AI) and the version with AI-supported onboarding.

1.1 Research Questions

My research addressed the following research questions:

1. **RQ1:** How does the presence of an AI-supported suggestion system affect a player's onboarding experience?
   (a) **RQ1.1:** What do players expect from an ideal onboarding experience in video games?
   (b) **RQ1.2:** What do players expect from AI-supported onboarding systems in video games?
2. **RQ2:** How do players perceive suggestions during gameplay?
3. **RQ3:** How do players learn during the onboarding process?

1.2 Results

The results of this study found that the presence of AI-supported suggestions during Joker's onboarding process successfully reduced players' cognitive load. However, it also demonstrated that a reduced level of cognitive load did not help players learn as effectively as an increased but manageable level. Too little cognitive load meant that players did not play an active role in the learning process. Players instead showed a clear preference for temporary structure at the beginning of a game that maintains player agency in regard to *how* they receive guidance and *when*. All study participants expressed a desire for engagement and interaction during the onboarding process, with emphasis on lived experience as a learning method. Furthermore, analytical results suggest that AI-supported onboarding methods have the potential to create more engaging and dynamic learning experiences that are personalized to players as individuals.

1.3 Contributions

This thesis makes the following contributions:

- a novel demonstration of an AI-supported onboarding system;
- increased knowledge about the impact of suggestions on cognitive load;
- an analysis of the relationship between onboarding and player experience; and
- design considerations for future implementations of AI in video game onboarding.

Chapter 2

Related Work

Two main areas of literature shaped this research: existing video game onboarding methods, and the current usage of AI in video games. This chapter describes the literature related to these two fields and identifies the research gap between their overlap.

2.1 Video Games and Onboarding

Every game proposes a new challenge to its designers: how will they teach new players how to play? *Onboarding*, a term traditionally used in the workplace to define the process of helping new employees succeed at a new job [4], also encompasses the methods game designers use to help new players succeed in a new game [35, 41]. Since onboarding is the first experience a player has with a game, it is critical that this initial interaction engages and retains new players [12]. If the initial game experience is too boring or too difficult, players may give up on the game altogether and quit [32]. This problem of player retention is especially relevant for live-service games, and even more so for free-to-play games, as they rely on active playerbases to support gameplay systems—such as matchmaking—and to spend money on in-game purchasable content [15, 33].

2.1.1 Onboarding Goals

Onboarding should make sure that the learning experience does not overwhelm new players. Cognitive load theory proposes that a higher level of interactivity contributes to increased *intrinsic cognitive load* when the amount of interaction required exceeds the capacity of a learner's working memory [31]. As video games are an inherently interactive medium, it is important to consider how a game's onboarding method affects a player's cognitive load. Mayer and Moreno propose various methods for reducing cognitive load in multimedia learning, such as by offloading some visual instructions to audio to balance the load between information processing channels [29]. In particular, the load-reducing methods of *weeding* and *signaling* act similarly to the onboarding methods of *training wheels* and *scaffolding* respectively: weeding removes some information to reduce load, while signaling adds additional information to guide learners through the provided information [29]. The related onboarding methods are well-discussed in existing research, and are described in more detail below.

Another area of research, *game approachability*, is defined as "the ease in which gamers are able to approach and avail themselves of games" [17]. This concept is highly related to game onboarding, since a large facet of game approachability is to help players find enjoyment in games as quickly as possible [17, 30]. However, while approachability principles can help designers identify flaws in their game's design [30], they do not provide designers concrete or actionable solutions. As onboarding in particular is specific to each game, prior research instead focuses on providing game designers with design principles to improve the initial player experience [28, 36, 21].

Finally, the ideal onboarding experience should be brief, yet still contain enough information for players to succeed [28, 32]. Shannon et al. emphasize the importance of introducing players to a game's mechanics quickly in order to prioritize learning through exploration [38]. Additionally, providing a memorable onboarding experience can help games retain players past their initial play session [12].

2.1.2 Onboarding Methods

Games implement a wide variety of onboarding methods. While each method is usually tailored to a specific game or genre, the broader teaching techniques used can be categorized into a smaller subset of groups defined in existing literature.

Tutorials

Tutorials are a common onboarding method in games. Matthew White further delineates tutorials as *didactic*—defined as upfront and intentionally intrusive—instructions, or *exploratory* prompts to encourage player experimentation [44]. Other literature also describes this difference as *explicit* versus *implicit* tutorials [10]. These "opposing" tutorial methods demonstrate the challenge of creating a balanced onboarding experience. Too little instruction means players are unable to figure out how to succeed on their own [18], while too much instruction can restrict agency and lead to player boredom or frustration [38, 32]. As well, the effectiveness of tutorials varies based on game complexity and an individual player's gaming expertise. Non-expert players and complex games generally benefit more from tutorials than expert players or simple games [2, 34, 45].

Additive Support

The concept of scaffolding has its roots in cognitive psychology. Just as with a physical scaffold, cognitive scaffolding provides support at the beginning of the learning experience, and is removed when a learner no longer needs the additional support [16]. In video games, this can be implemented by providing additional information on top of the standard game information to prevent new players from getting stuck [18]. Prior work by Faber et al. also investigated *adaptive scaffolding*—where scaffolding instructions adapt to the user—but the identified relationship between adaptive scaffolding and performance could not be generalized to areas outside of game-based learning [19].

Subtractive Support

In contrast to scaffolding, the *training wheels* onboarding method simplifies gameplay to focus players on learning basic gameplay first [28]. Then, as a player gains more experience, they are gradually introduced to more complex gameplay mechanics until they have access to the entire game. Allowing players to break high-complexity tasks down into simpler ones also can help prevent cognitive overload, though this can negatively impact the learning process when players need to use the "pieces" of a task in combination [43]. The training wheels method is also related to the concept of *sandboxes*: safe spaces for players to learn without risk, while still feeling a sense of accomplishment [21]. With less risk, players are then more willing to experiment and make mistakes while learning, since the punishment for doing so is either non-existent or extremely mitigated compared to "real" gameplay [21, 18].

Real-Time Support

The *personal advisor* onboarding method gives players advice based on their actions during gameplay [36]. This method is a reactive response rather than a proactive suggestion, as the advice appears after the player chooses an action in order to provide more detail on that action's consequences. It is similar to the *performance coaching* method [28], where the game acts as a teacher and uses suggestions to encourage optimal gameplay. Additionally, the *just-in-time* method—where information is provided to players exactly at the time they need it—can also be considered as a type of advisor or coaching strategy [18, 45, 16, 36]. Information with lower complexity is better suited for the just-in-time method, as it is less likely to cognitively overwhelm learners when presented during a task [43].

Unlike additive or subtractive support methods, real-time support does not modify the mechanics of a game. Additionally, it avoids the restrictive teaching methods of didactic tutorials while still allowing for player exploration. The just-in-time method in particular is one of the most common onboarding methods found in games [36]. Because of these benefits, as well as the prominence of just-in-time onboarding in existing games, I decided to implement a real-time onboarding method in the game I developed for this study.

2.2 Video Games and AI

Existing literature on AI for games extends back to traditional board games such as checkers, chess, and Go [48]. One of the most well-known implementations of AI in games is the *Deep Blue* chess system, which was able to defeat the world champion [9]. In fact, this concept of AI as a player is one of the three main areas of research on AI in games that Yannakakis and Togelius identify: AI for playing games, AI for generating content, and AI for modelling players [48]. However, game onboarding does not fall neatly into one of these categorizations; principles from all three categories are instead relevant to different parts of the onboarding process.

2.2.1 AI for Playing Games

Video games use AI players for many different reasons, such as to provide an opponent in a single-player game, or to expand the story of a game using *non-player characters* [50]. Non-player characters are well-suited for the personal advisor onboarding method, since they can appear diegetically and present advice within the context of the game's world [36]. As well, fostering long-term player interaction with an in-game character that mimics a companion can also increase engagement with a game [37]. On the other hand, AI as an opponent rather than an ally also has uses in game onboarding. Chen et al. found that players experienced less cognitive load when playing against AI opponents, and additionally emphasized the need for a game to match a player's cognitive load threshold to maintain an enjoyable experience [11]. Furthermore, Tan et al. propose the concept of adaptive AI opponents that can match a player's skill level in real time [39]. Dynamic opponents in tutorials should adjust the selection and timing of the skills they test to create a more engaging learning experience for the player [37].

2.2.2 AI for Generating Game Content

Many games use *procedural content generation* to dynamically increase the amount of content available in-game—such as by generating levels, characters, items, or more—without requiring additional work from artists or game designers [42]. When combined with player modelling systems, designers can use procedural content generation to make games more adaptive, personalized, and enjoyable for players to experience [47], which are all important factors to consider during the onboarding process. However, AI-generated tutorials are a non-trivial problem; games with complexity higher than simple arcade games are difficult for AI systems to properly understand, much less to explain in human-understandable ways [22]. While AI agents have been shown to be capable of generating game levels that teach a specific mechanic, Green et al. found that the AI's levels were often impossible for humans to play, or still required an advanced level of skill unsuitable for teaching novice players [23].

2.2.3 AI for Modelling Players

Player experience modelling uses AI systems to create a model of a player's unique experience during gameplay, which can then be used to personalize aspects of the game [46]. This concept of tailoring games to individuals is very applicable to onboarding, as individual players have different skill levels and therefore different needs during the onboarding process [14, 37]. For example, rather than use static tutorials, Benotti and Bertoa used natural language generation AI to display relevant text-based hints based on players' actions in a first-person shooter game [5]. This allowed for players to receive instructional support tailored to their current situation in the game. Another method of using player modelling systems in onboarding is *challenge tailoring*: the game adapts to in-game behaviour, and in response can change upcoming gameplay elements to better fit the player's modelled skill trajectory [51, 37]. Player models can also make predictions on a player's behaviour to determine when they might get stuck, become frustrated, or quit playing [49], all of which are extremely relevant aspects of player experience during the onboarding process.

2.3 Summary

In this section I examined existing literature on video game onboarding as well as the usage of AI in video games. Key findings in onboarding indicate the importance of managing a learner's cognitive load [31, 29], while also making sure that the onboarding process is neither too boring nor too difficult [32, 38, 18]. However, while both areas individually have depth and breadth of prior research, the space regarding the overlap of these two areas is not yet well-defined. Of the research that explicitly addresses both onboarding and AI, I found that current knowledge either focuses on the technical aspect of AI-supported onboarding implementations rather than on player experience [22, 23, 51], or does not come to broadly-applicable conclusions on how AI-supported onboarding impacts player experience [11, 39, 5]. There is also a lack of qualitative studies regarding AI and onboarding. Furthermore, while the just-in-time onboarding method is well-established as an effective onboarding strategy [36, 45], there is a knowledge gap regarding the potential implementation of AI in this type of real-time onboarding. With all of this in mind, the goal of my study was therefore:

- to demonstrate an implementation of the just-in-time onboarding method in an AI-supported suggestion system;
- to explore how AI-supported suggestions interact with new players' cognitive load;
- to contribute qualitative findings that reduce the existing research gap between AI and video game onboarding; and
- to better understand the relationship between AI-supported suggestion systems and player experience.

Chapter 3

Game Design

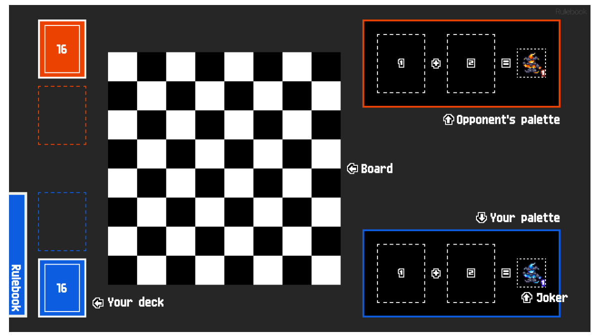
Similar to previous onboarding research in games [34], this study required that participants had never played the game selected for the study beforehand. This way, they could not affect their onboarding experience with prior knowledge of the game's mechanics. Additionally, since the focus of this research is AI-supported suggestion systems in onboarding—established as a knowledge gap in Chapter 2—it was necessary to create an AI-supported onboarding system regardless of whether the study used an existing game or created a new one. Using an existing game meant excluding potential participants who had played it before, and developing an onboarding system for an existing game also posed additional challenges (such as not having access to the game's code). I therefore determined that creating a new game specifically for the study was the most optimal choice.

3.1 Joker

Joker is a two-player turn-based strategy game designed to contrast familiarity and unfamiliarity. It uses the familiar game elements of playing cards and a chessboard to create an unfamiliar game. I made this design decision in order to focus the onboarding process on learning gameplay mechanics, rather than on learning how to interact with unfamiliar game elements (for example, having to learn movement on a hexagonally-tiled board). Figure 3.1 showcases screenshots from different sections of Joker, including: a) the title screen; b) the rules stage; c) gameplay from Condition A; d) gameplay from Condition B; and the end screens for both e) winning and f) losing.

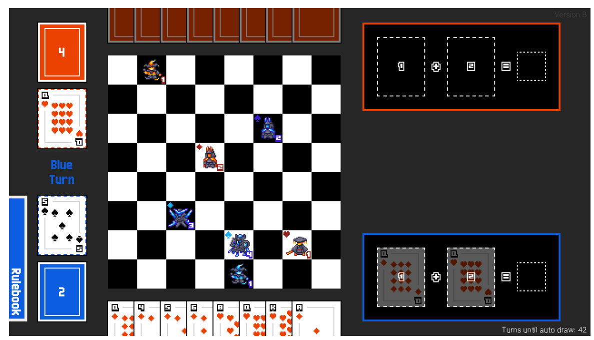
The core gameplay concept of Joker is card combinations: players combine pairs of playing cards to place new units onto their side of the chessboard. Both players have

(a) Title screen

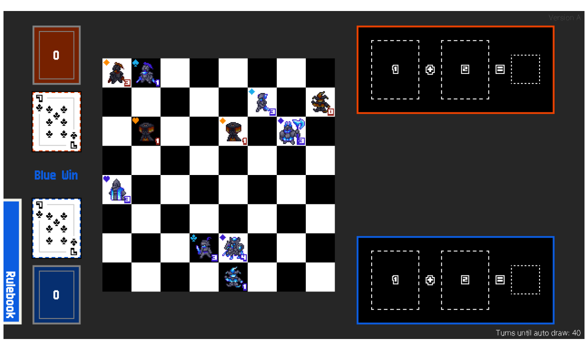


(b) Rules stage

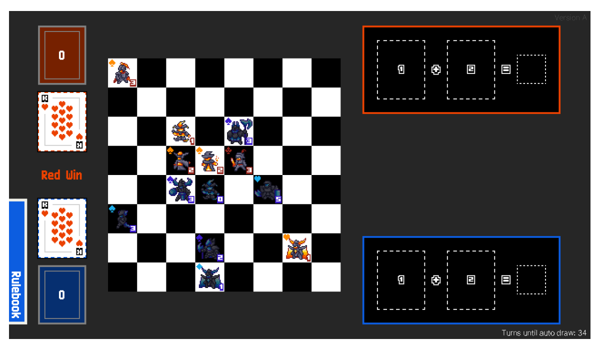
(c) Condition A gameplay



(d) Condition B gameplay



(e) Win screen



(f) Lose screen

Figure 3.1: Various screenshots from Joker.

a joker—similar to a king in chess—that they must protect. They take turns creating, moving, and attacking units in order to kill the opponent's joker and win the game. Additionally, Joker's card combination system is *intentionally complex* so as to overwhelm new players and introduce cognitive load. It also plays off of familiarity in a negative way; for example, higher cards are not strictly better, and the joker unit cannot defend itself the same way that a chess king can. As one of the goals of this study was to explore how AI-supported suggestions interact with new players' cognitive load, I needed to introduce a complex and overwhelming gameplay mechanic for the AI-supported suggestion system to manage. A low-complexity mechanic risked being easy for participants to understand, which defeats the purpose of examining the onboarding experience. With this in mind, I provide a summary of Joker's card combination mechanic in Tables 3.1 and 3.2 for the sake of completeness; however, it is not necessary to understand this mechanic to understand this paper.

	Advantage	**Weakness**	**Light Chroma**	**Dark Chroma**
Spade	Heart	Diamond	Knight	Bishop*
Heart	Club	Spade	Rook*	King
Club	Diamond	Heart	Knight	Bishop*
Diamond	Spade	Club	Rook*	King

Table 3.1: Table of suit aspects. The first card in a combination determines a unit's suit, while the second card in a combination determines a unit's chroma. *Advantage* determines what type of attack a unit deals. *Weakness* determines what type of attack a unit receives. *Chroma* determines a unit's movement pattern; movement patterns are similar to those of chess pieces. Patterns marked with an asterisk differ from chess patterns with movement limited to two spaces in a valid direction.

3.1.1 Technical Details

Joker was developed in Unity 2019.4.40f1. I created all of the visual elements (such as units and cards) myself using Clip Studio Paint and Figma. The fonts are free for personal use,[1]

[1] https://fonts.google.com/specimen/Inter/about
https://fonts.google.com/specimen/DotGothic16/about
https://www.dafont.com/mini-pixel-7.font

	Unit	Key	Combos	Extra
2	Bounty Hunter	B	1	Queen's move
3	Cavalier	C	1	
4	Duelist	D	2	
5	Evoker	E	2	
6	Fighter	F	3	
7	Gambler	G	3	
8	Herald	H	4	
9	Inquisitor	I	4	
10	Justicar	J	5	
11	Kingslayer	K	5	
12	Legionnaire	L	6	
13	Mercenary	M	6	
14	Knight	N	7	
15	Oracle	O	6	
16	Paladin	P	6	
17	Queenslayer	Q	5	
18	Ranger	R	5	
19	Sorcerer	S	4	
20	Tactician	T	4	
21	Usurper	U	3	
22	Vanguard	V	3	
23	Wanderer	W	2	Queen's move
24	Exalt	X	2	
25	Jaeger	Y	1	
26	Zealot	Z	1	Queen's move

Table 3.2: Table of units. The sum of two cards determines the type of unit created. Each sum has anywhere from 1–7 unordered ways to combine cards. Each unit corresponds to a letter of the alphabet; however, this is a "rule of thumb" and is not explained directly in-game. Each unit also has its own health point value and a unique ability (not listed for brevity). Additionally, three units have an extra characteristic: their movement pattern is always the same regardless of their suit.

the music is licensed under Creative Commons 3.0,[2] and the sound effects[3] are licensed under Creative Commons 1.0. The final version was built for WebGL and privately hosted on itch.io[4] for the duration of the study.

3.1.2 Game Elements

Joker has two main gameplay systems: unit creation and unit management. The deck, hand, and palette are part of the creation system, while the board is part of the management system. Units are part of both.

The **unit creation system** uses the different characteristics of playing cards to make new units. On their turn, a player can combine two cards from their hand into a new unit. The suits, numerical sum, and order of every pair of cards determine which unit is created. Since players can have up to eight cards in their hand at a time, the number of possible combinations for a full hand is $\frac{8!}{(8-2)!} = 56$. Furthermore, a unit may have different characteristics based on the cards used to create it. The rules for card combinations are given to the player in the rulebook pages shown in Appendix C. Again, this system was intentionally designed with high complexity in order to overwhelm new players and increase their cognitive load.

The **unit management system** contains all other aspects of gameplay: moving units, attacking enemies, and using abilities. This system has a different type of complexity than the unit creation system—rather than creating complexity through information-based decisions, it instead requires the player to make strategic decisions through unit positioning and planning. The rulebook gives the player information about the technical aspects of unit management, but it is up to them to discover gameplay strategies on their own.

3.1.3 Onboarding

Joker's onboarding consists of a rulebook and, in Condition B, an AI-supported suggestion system. The rulebook shown in Appendix C acts similarly to the rulebook of a physical board game, and the player is able to freely access it during gameplay.

The AI-supported suggestion system interacts with the unit creation system. It follows the just-in-time onboarding method to provide players with real-time support during the

[2] "Space Fighter Loop", "Video Dungeon Boss" by Kevin MacLeod (incompetech.com). Licensed under Creative Commons: By Attribution 3.0 http://creativecommons.org/licenses/by/3.0/

[3] https://www.kenney.nl/

[4] https://itch.io/

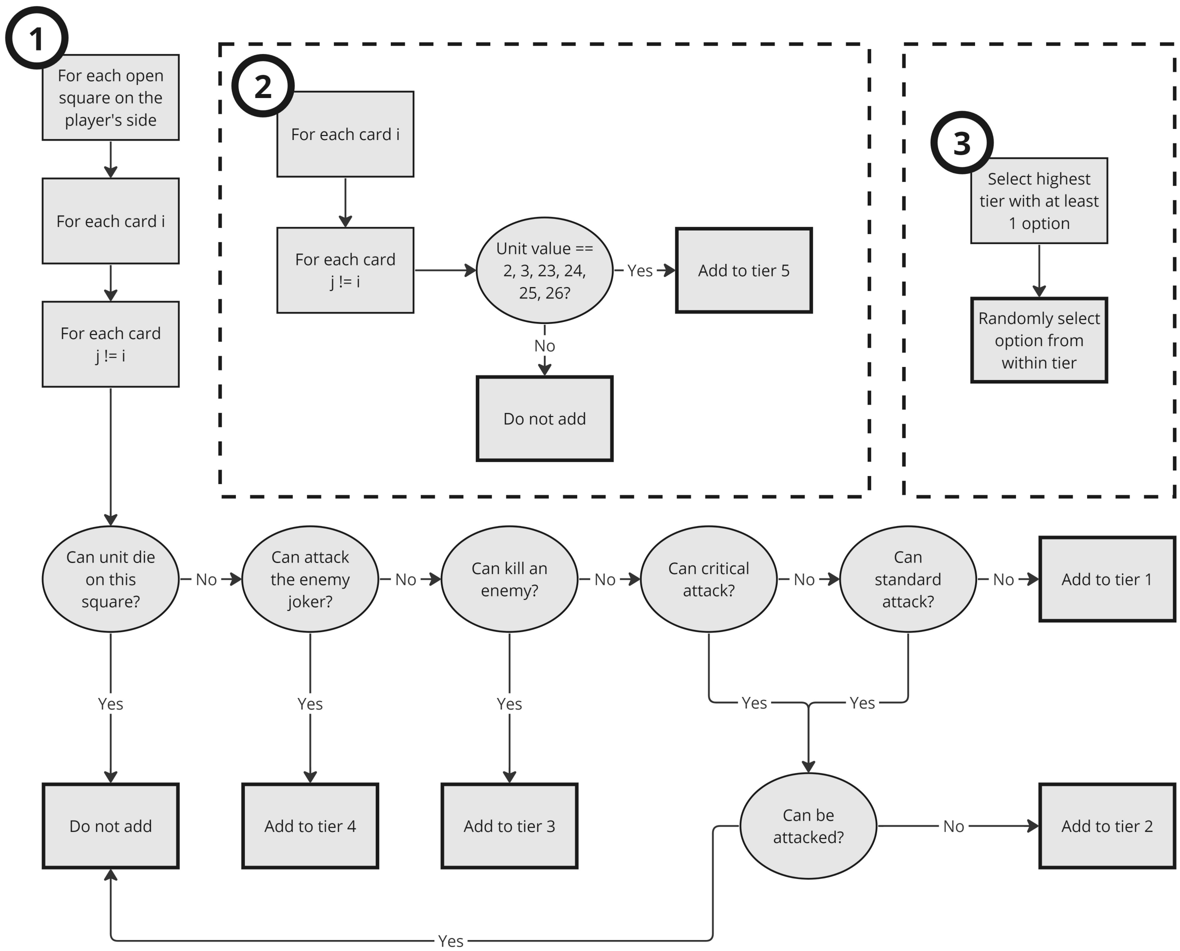


Figure 3.2: Visualization of the AI's suggestion-making process. The three steps always occur in the following order: 1) sort all possible unit combinations into tiers; 2) check if any priority units exist; and 3) randomly select an option from the highest available tier.

game. Each turn it suggests the player two cards from their hand to combine into a unit; however, it is not mandatory that the player follows the suggestion. This works well with the just-in-time method since a pair of cards has lower information complexity compared to Joker's other mechanics, and is therefore unlikely to overwhelm players when it is presented on their turn.

To make its suggestion, the system uses a decision tree-like function similar to the AI opponent's code that excludes all other actions except for combining cards. It evaluates every possible pair of cards on all possible squares of the board where the player can create a unit, then sorts the resulting units into different tiers based on different factors (e.g., if the unit can kill an enemy when it is created on a certain square). It also prioritizes creating certain units with powerful abilities by doing a second pass through the results and elevating the "best" units to the highest tier. Figure 3.2 describes this process in more detail.

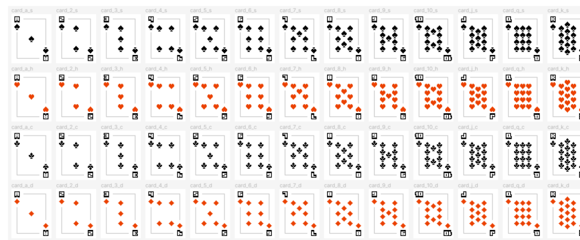
3.2 Development Process

I started development by creating a presentation about the concepts for Joker's game mechanics. Since the game was intentionally designed to be cognitively demanding (see Figure 3.3a), this presentation helped me to better explain my thesis topic during the initial approval stage. Once I received approval, I created a majority of the game assets in Clip Studio Paint, and additionally finished any assets that required precise layouts—such as the cards in Figure 3.3b—in Figma. I also used Figma to design the title screen and the game interface (Figure 3.3c), as well as to prototype the rulebook later on in development (Figure 3.3d).

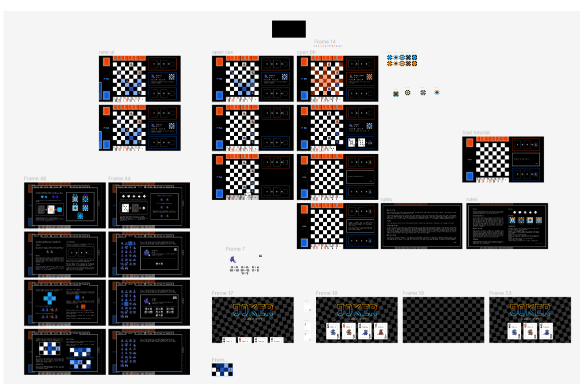
I then created a private GitHub repository[5] to more easily keep track of my code during development. After the initial commit, I created a separate branch named *wip* for active development; when I reached a development "checkpoint" I would merge *wip* to *main* and create a new release to track my progress. The initial version of Joker took about one month to create. However, as I made changes to the study's design over the course of development, I needed to constantly update the game to keep it aligned with the study's methodology (detailed in Chapter 4). This iterative development process is described below.

[5] Joker's code is not publicly available at the time of writing.

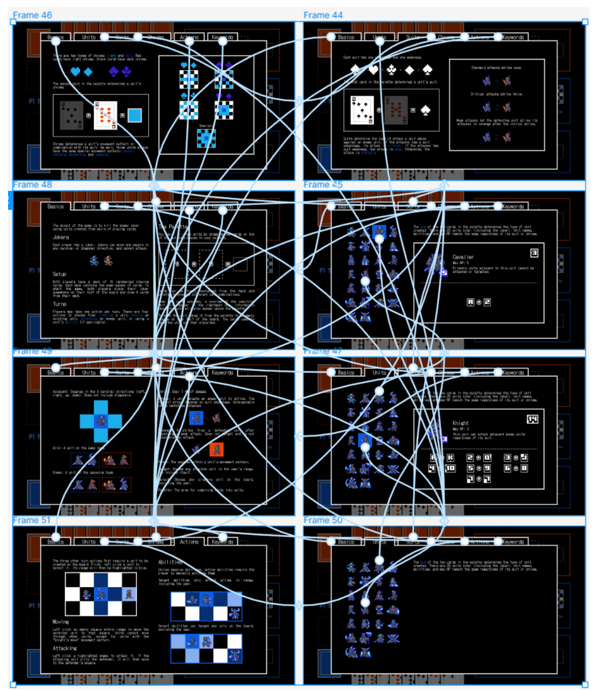
(a) Slide from initial presentation



(b) Figma assets



(c) Figma workspace



(d) Rulebook prototype

Figure 3.3: Images from the development of Joker, including: a) a slide from the initial presentation demonstrating the cognitive demand of multiple game mechanics; b) using Figma to create the precise suit layouts of the playing card assets; c) an overview of the many interface sketches I created in Figma; and d) the rulebook's functional prototype.

3.2.1 Version 1

Version 1 of Joker was very similar to the final version of the game. It consisted of two conditions: one with suggestions, and one without. In the condition with suggestions, the AI's suggestions encompassed card combinations, unit placement, unit movement, unit attacks, and unit abilities. As well, the rulebook element did not exist in either condition; participants simply played one condition first and the other condition afterward.

Feedback from playtesting was mostly negative. Players struggled to understand how to start the game, and most of them lost to the AI opponent in under five minutes. Players also felt that the AI's suggestions took away their agency and didn't help them learn the rules of the game.

3.2.2 Version 2

To address the feedback from Version 1, Version 2 incorporated a tutorial to teach players the different mechanics of the game. It consisted of four conditions: the first with no tutorial and no suggestions; the second with the tutorial and no suggestions; the third with suggestions and no tutorial; and the fourth with both the tutorial and suggestions. Participants were to play all four conditions during the study, with the order for each participant determined by a Latin square.

Feedback from playtesting was again mostly negative. Players felt that the tutorial took too long (5-10 minutes). Many players skipped the tutorial's text in order to get to the real game faster. As well, players who played a condition without the tutorial first struggled to figure out how to play. Most participants also lost to the AI opponent in under five minutes during their first condition, regardless if they had completed the tutorial or not. All participants found that repeating the tutorial during the four games was tedious and unnecessary. Finally, this version also placed too much emphasis on the tutorial, rather than on the AI's suggestions, and did not properly address this paper's research questions.

3.2.3 Version 3

To address the feedback from Version 2, Version 3 removed the tutorial elements and returned to a two-condition game similar to Version 1. The differences were that Version 3 added the rulebook and reduced the extent of the AI's suggestions in the second condition—it now focused solely on card combinations and did not make suggestions for unit actions or placement. Finally, this version also added a red highlight to both conditions for when the joker units were in danger.

Feedback from playtesting was overall positive. Players liked the freedom the rulebook provided; rules-focused players spent time reading the entire rulebook before playing, while game-focused players could skip reading and get right to the game, while still being able to refer to it when they needed help. The combination-focused AI increased player engagement and agency, since they could no longer blindly follow directions for the whole game. Players also lasted longer against the AI opponent, and some even managed to win. The red highlights helped players recognize when their own joker was in danger, which reduced the number of playtests where players instantly lost to the AI opponent.

Ver.	**Conditions**	**Features**	**Feedback**
1	A, B	**A:** X **B:** suggestions (all)	Not enough guidance Lack of agency
2	A, B, C, D	**A:** X **B:** tutorial **C:** suggestions (all) **D:** tutorial + suggestions (all)	Tutorial too long Low-quality guidance Conditions repetitive Not aligned with RQs
3	A, B	**A:** danger highlight **B:** danger highlight + suggestions (card combinations only)	Gets into game fast Increased agency Sufficient guidance

Table 3.3: Comparison of the three major versions of Joker.

3.3 Summary

I designed and implemented the video game Joker, as well as an AI-supported suggestion system that utilizes the just-in-time onboarding method. Joker has two built-in gameplay conditions: Condition A, the base game, and Condition B, which adds the AI-supported suggestion system to the base game.

Chapter 4

Methodology

The purpose of this study is to determine the impact of AI-supported suggestion systems on the onboarding experience of video games.

4.1 Research Questions

The following questions informed the design of this study:

1. **RQ1:** How does the presence of an AI-supported suggestion system affect a player's onboarding experience?
   (a) **RQ1.1:** What do players expect from an ideal onboarding experience in video games?
   (b) **RQ1.2:** What do players expect from AI-supported onboarding systems in video games?
2. **RQ2:** How do players perceive suggestions during gameplay?
3. **RQ3:** How do players learn during the onboarding process?

RQ1 targets the intersection of AI-supported suggestion systems and onboarding. Its purpose is to understand players' expectations for their preferred game onboarding experience, as well as their expectations for implementations of AI-supported onboarding systems in video games.

RQ2 targets players' perception of suggestions during gameplay (real-time support). It aims to better understand how players perceive and interact with suggestion-based support systems during gameplay.

RQ3 targets the learning experience during the onboarding process to determine what learning methods players rely on the most.

4.2 Procedure

This study uses a mixed-methods within-subjects approach with two conditions (A and B). Condition A was the base game. Condition B was the base game with an added AI-supported suggestion system. All participants were given both conditions. The order of the conditions was randomized across participants to reduce possible bias. After each condition participants filled out two surveys, then completed a semi-structured interview with the researcher.

miniPXI Survey

The Mini Player Experience Inventory (miniPXI) is a variant of the Player Experience Inventory (PXI) survey [1] with one scale per each of the eleven measures of player experience [25]. I used the miniPXI over the original PXI since this was a within-subjects study. The miniPXI has a recommended use case in studies that require greater efficiency and where the miniPXI is not the only measurement [25]. I therefore decided to use the miniPXI because this study collects other forms of quantitative and qualitative data, and because the within-subjects design requires that participants complete all tasks twice in an efficient manner.

Raw NASA-TLX Survey

The NASA Task Load Index (NASA-TLX)[1] uses six scales to measure a participant's workload while performing a task [26]. I used the Raw Task Load Index (RTLX) during analysis, a modified version of the NASA-TLX that removes the weighing process. There is mixed consensus surrounding whether the RTLX has higher or lower sensitivity than the original version [26]; as the weighing process is time consuming, and as this study collects

[1] https://humansystems.arc.nasa.gov/groups/tlx/downloads/TLX_pappen_manual.pdf

other forms of quantitative data as well as qualitative data, I decided to use the RTLX for simplicity.

4.2.1 Apparatus

Participants played the game on a desktop computer with a mouse, keyboard, and two monitors. The same computer was used for the entire study. The game ran in a fullscreen browser window on the primary monitor while the study survey (see Appendix B) ran on the left monitor. I placed a phone on the desk beside the participant to record each session's audio data, and used OBS[2] to record the gameplay data. Additionally, I informed participants that the screen recording software only recorded gameplay in the primary monitor, and did not record the survey.

4.2.2 Protocol

I met each participant at the lab entrance and took them to the study room. Once they indicated they were ready, they sat down at the computer and filled out the consent form and demographics sections of the study survey in the left monitor. After I confirmed consent, I started the audio and screen recordings. I then introduced the study and directed the participant to select the "Rules" card on the main menu of the game (see Figure 4.1). The participant was then instructed to take as much or as little time as they normally would when approaching a new game to go through the game's rulebook.

After the participant finished with the rulebook, I directed them to select the card on the main menu corresponding to their first condition (Figure 4.1). The participant then played one round of the game and filled out the miniPXI and RTLX surveys in the left monitor. Afterward, I conducted a brief semi-structured interview about their first experience. The participant was then offered a break.

Next, I directed the participant to select the card on the main menu corresponding to their second condition. The participant then played one round of the game and filled out the miniPXI and RTLX surveys in the left monitor. Afterward, I informed the participant I was stopping the screen recording, but continuing to record audio. I then conducted a semi-structured interview about their second experience and additional topics related to the research questions.

[2] https://obsproject.com/

At the end of the interview I informed the participant I was stopping the audio recording. I then thanked the participant for their time and awarded them $20 CAD in cash as remuneration for their participation.

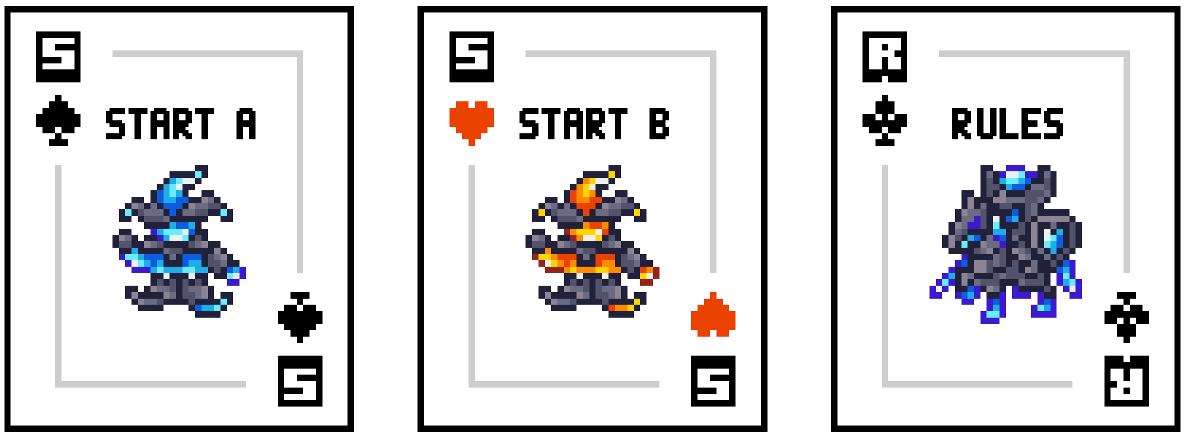


Figure 4.1: Menu graphics for the three game modes: "Start A", "Start B", and "Rules". "Start A" loads Condition A; "Start B" loads Condition B; and "Rules" loads the preliminary rulebook stage.

4.3 Participants

After receiving ethics approval (REB #45535), I recruited 21 participants in total. My recruitment threshold was 20 participants; however, one participant's session was interrupted by a fire alarm and therefore excluded from analysis. An additional participant was then recruited to maintain an equal number of participants per condition (10 each). The participant interviews reached saturation—the point where subsequent interviews produced little to no new information—within 20 sessions. This was expected, as prior research demonstrates that qualitative studies can reach saturation with as few as 9–17 interviews, especially when the participant group is more homogeneous [27, 3]. Furthermore, the most prominent thematic analysis researchers recommend 6–15 interviews for a masters-scale project [40].

The final dataset contains 20 participants with an average age of 25.5 years (min: 18, max: 43). 12 were male, 7 were female, and 1 was non-binary. All participants had completed some form of post-secondary education, and 19 participants self-identified as a student (see Table 4.1 for more detail). As well, to address my research questions, the target participant population needed to have experience with either video games or with AI. Since this was an in-person study, I decided to recruit local participants through both the Games Institute[3] and the University of Waterloo CSC graduate student mailing list.[4]

To participate in the study, participants were required to:

- be an adult (18 or older);
- be comfortable being audio and screen recorded;
- be comfortable using a computer (keyboard and mouse); and
- be comfortable sitting down for the duration of the study (up to 90 minutes).

These requirements were checked with a Qualtrics[5] screening survey (see Appendix A). Eligible participants were directed to schedule a session time with my Microsoft Teams Bookings calendar. Ineligible participants were informed they did not meet the study requirements. No identifying personal data was collected at any point during the screening survey.

4.3.1 Data Collection

I audio and screen recorded all participant interviews for later analysis. Every participant filled out the online Qualtrics consent form (included in Appendix B) at the beginning of the study that detailed what types of data I would collect during the session. Additionally, all questions in the demographic survey were optional. At the end of the session, participants received \$20 CAD in cash; I collected signed paper receipts of this transaction for administrative purposes. Participants were also given the option to provide their email address if they were interested in receiving the results of this study.

During the study each participant received an ID (e.g., "P1") corresponding to their session number. These IDs were preserved throughout the analysis process for convenience, but were randomized in the subsequent reporting process so as to provide greater participant confidentiality, as well as to account for the excluded 21st participant by realigning the IDs on a scale from 1–20.

[3]https://uwaterloo.ca/games-institute/
[4]https://cs.uwaterloo.ca/cscf/mailman/lists/
[5]https://www.qualtrics.com/

ID	Order	Age	Gender	Ethnicity	Education	Employment
P1	AB	23	Male	West Asian	Bachelor's degree	Student
P2	AB	27	Male	North African	Bachelor's degree	Student
P3	AB	22	Female	Chinese	Bachelor's degree	Student
P4	AB	22	Male	Chinese	Bachelor's degree	Student
P5	AB	22	Male	South Asian	Bachelor's degree	Student
P6	AB	30	Female	Chinese	Associate degree	Student
P7	AB	N/A	Female	West Asian	Postgraduate degree	Student
P8	AB	33	Male	West Asian	Postgraduate degree	Student
P9	AB	22	Male	South Asian	Bachelor's degree	Student
P10	AB	43	Male	West Asian	Postgraduate degree	Student
P11	BA	28	Non-binary	White	Postgraduate degree	Student
P12	BA	18	Male	White	Bachelor's degree	Student
P13	BA	23	Female	South Asian	Postgraduate degree	Student
P14	BA	23	Male	White	Postgraduate degree	Student
P15	BA	25	Male	Arab	Bachelor's degree	Student
P16	BA	27	Female	South Asian	Postgraduate degree	Student
P17	BA	23	Male	South Asian	Bachelor's degree	Student
P18	BA	21	Female	White	Bachelor's degree	Student
P19	BA	26	Female	West Asian	Postgraduate degree	Employed
P20	BA	26	Male	North African	Bachelor's degree	Student

Table 4.1: Table of participant demographics. The "ID" column denotes the realigned participant numbers, while the "Order" column indicates participants' condition order during their session.

4.4 Analysis

This study uses a mixed-methods design to collect both quantitative and qualitative data.

4.4.1 Quantitative Analysis Method

I collected three types of quantitative data during the study (session recordings and two quantitative surveys), all of which were statistically evaluated to identify areas of interest.

The session recordings were evaluated with a spreadsheet. I used the screen and audio recordings from each session to determine the length of each game, how long each participant spent reading the rulebook, and which games participants won and which games they lost. For the times, I used a spreadsheet to take the difference between the starting and ending timestamps, then used basic functions to calculate the average, minimum, maximum, and sample standard deviation for each subset of the data.

The miniPXI and RTLX surveys were evaluated in RStudio.[6] I first conducted Shapiro–Wilk tests and determined that a majority of the survey items were not normally distributed ($p < 0.05$) and therefore non-parametric (see Table 4.2). While some of the survey items did not reject the null hypothesis and could have been parametrically evaluated, I decided to use non-parametric tests across the entire dataset to maintain consistency. For the final analysis I used Wilcoxon signed-rank tests on all survey items (paired by Condition A and Condition B) to determine any significant effects.

4.4.2 Qualitative Analysis Method

I used thematic analysis to analyze the qualitative interview data following a six-step process originally proposed by Braun and Clarke [6]. This method emphasizes researchers taking an active role in qualitative data analysis, has flexibility well-suited to this study's data (both its homogeneity and the number of participants), and is the most-cited thematic analysis method to date [13].

Step 1: Data Familiarization

The first step of Braun and Clarke's thematic analysis method is data familiarization. This study's data was originally collected as audio recordings, so the recordings needed

[6] https://posit.co/download/rstudio-desktop/

miniPXI	Shapiro–Wilk	RTLX	Shapiro–Wilk
AA	$p < 0.001$	MD	
CH		PD	$p < 0.010$
EC		TD	$p < 0.010$
GR	$p < 0.010$	Pe	
PF		Ef	
AUT	$p < 0.050$	Fr	
CUR	$p < 0.010$		
IMM	$p < 0.001$		
MAS	$p < 0.050$		
MEA	$p < 0.050$		
ENJ	$p < 0.050$		

Table 4.2: Results of statistical tests to determine if the survey data was parametric or non-parametric. The null hypothesis that the data was normally distributed was rejected in 8 out of 11 miniPXI measures and in 2 out of 6 RTLX scales.

to be transcribed before analysis. I used the online platform Dovetail[7] to automatically transcribe the audio, and additionally cleaned all 20 transcripts afterward to correct initial errors.

Step 2: Generating Codes

Reflexive thematic analysis is an iterative and inductive process where codes evolve and change over the course of analysis [8]. While reflexive analysis distinguishes itself from the more rigid "codebook" and "coding reliability" methods through its flexible and organic process [7], I still incorporated aspects of both of these methods in my approach: I used iterative "codebooks" to distinguish each phase of the coding process, and I assigned multiple coders to each transcript to facilitate a more thorough understanding of the data through the coders' differing backgrounds. However, at its core this method of thematic analysis remains reflexive throughout.

Three coders in total worked on this project. I was first coder on all 20 transcripts, while two other coders each coded 11 transcripts. We coded line-by-line in a five-stage process as follows:

[7] https://dovetail.com/

ID	**Coder 1**	**Coder 2**	**Coder 3**
P1	X		
P2	X		
P3	X	X	X
P4	X	X	X
P5	X	X	
P6	X		X
P7	X	X	
P8	X		X
P9	X	X	
P10	X		X
P11	X	X	
P12	X		X
P13	X	X	
P14	X		X
P15	X	X	
P16	X		X
P17	X	X	
P18	X		X
P19	X	X	
P20	X		X
P1	X	X	
P2	X		X

Table 4.3: Breakdown of each coder's workload in the coding process.

1. **Codebook 1:** I used an inductive approach to code 10% of the data (2 transcripts) to create the initial codebook.

2. **Codebook 2:** I briefed the other two coders on my research and familiarized them with the data. Then, all three coders individually coded the same 10% of the data (2 transcripts) in order to create a diverse and rigorous set of codes. We iterated and expanded on **Codebook 1** with a hybrid inductive–deductive approach, then met together to discuss the codes and unify the transcripts. In this hybrid approach coders could both refer to existing codes from the previous codebook as well as create new codes to later discuss. We resolved any disagreements during the triple-coder process with thorough discussion and, if we could not reach a consensus, majority vote.

3. **Codebook 3:** In this phase, I reduced the number of coders per transcript from three to two. I individually coded another 20% of the data (4 transcripts), while the other two coders split the position of second coder (2 transcripts each). We iterated and expanded on **Codebook 2** with the same hybrid inductive–deductive approach, and again met together to discuss the codes and unify the transcripts. We resolved any disagreements during the two-coder process with thorough discussion, and used the third coder as a tiebreaker if we could not reach a consensus. At the end of this stage we found that the codebook had not changed as much from the second to the third iteration as it had from the first to the second, and did not anticipate any more drastic changes.

4. **Iteration 1:** I individually coded another 30% of the data (6 transcripts), while the other two coders split the position of second coder (3 transcripts each). We iterated on **Codebook 3**, and while it still changed from its initial version, it did not change enough to be considered a distinct codebook.

5. **Iteration 2:** I individually coded the remaining 30% of the data, and additionally re-coded the transcripts from the first stage with the updated codebook (8 transcripts total). This was to ensure that all transcripts followed a similar process of coding, discussion, and consensus. The other two coders split the position of second coder, with each one taking one of the initial two transcripts (4 transcripts each). We iterated on the codebook from **Iteration 1**, and again found that the changes we made were minimal.

The coding process is also visualized in Table 4.3. At the end of this process we discussed the final codebook—shown in Figure 4.2—to merge redundant codes and clarify each code's meaning.

Figure 4.2: The final codebook.

Step 3: Constructing Themes

Once the coding process was finished, I exported the final codebook from Dovetail and imported the codes into Miro.[8] Each code received a coloured "sticky note" labelled with the code and the number of times it was used. The final codebook in Dovetail used coloured groupings during the coding process to make it easier to find codes (for example, colouring onboarding-related codes red). However, these colours had no analytical influence during the sorting process and were retained purely to improve visual identification.

I then used an affinity clustering process to group related codes. I progressively sorted the notes in groups of two or more codes until there were no ungrouped codes, then repeated the sorting process on both individual codes as well as the groups of codes. Clustering is a flexible process and the groupings changed dramatically over time as I built my candidate themes. When I was satisfied with the theme clusters, I created working titles and mapped each theme to the relevant research question (shown in Figure 4.3).

Step 4: Reviewing Themes

Next, I evaluated the candidate themes with two methods: reviewing the extracts contained within each theme, and reviewing the relationship between the whole dataset and the themes [6].

[8] https://miro.com/index/

Figure 4.3: The initial theme clusters and working titles. In this stage the sticky note colours still align with the codebook groupings to make it easier to identify individual codes.

For the first method, I returned to Dovetail and analyzed the coded sections of the data included within each theme. When I found that a code's extracts did not fit its overarching theme, I made changes such as moving the code to a different theme or renaming the theme's working title to better fit its codes. I also split broader themes into more specific subthemes that better represented a group of coded extracts.

I then used the second evaluation method to make sure that the themes fit the dataset as a whole. I reread the participant transcripts to make sure that the themes did not misrepresent the encoded data, that nothing important within the data was missed, and that the themes had not strayed from the study's research questions. The resulting theme clusters from this evaluation are shown in Figure 4.4.

Step 5: Defining Themes

When I finished reviewing the themes, I began the analysis process by creating the theme definitions. It was important that I made sure each theme was equally robust, precise, and clear, since these factors contribute to the overall quality of the analysis [40]. Afterward, I finalized the theme names—both for the main themes and the subthemes—so that they clearly captured the content contained within their definition.

Figure 4.4: The theme clusters after the revision process alongside the working titles for themes and subthemes. The sticky note colours now indicate which codes make up which subtheme. Note that main themes still encompass all codes beneath them regardless of colour.

Step 6: Reporting

The final step of thematic analysis is to report on the themes—and, primarily, to use the data as evidence to support a theme's prevalence and legitimacy [6]. Braun and Clarke's method encourages a narrative approach to tell a "story" with the data that is more thoughtful than surface-level analysis, and is contextually relevant to the research questions [6]. I present the results of this step in Chapter 5.

4.5 Summary

This study collected both qualitative and quantitative data to address research questions surrounding the impact of AI-supported suggestion systems on the onboarding experience of video games. I described the study's methodology regarding its participants, apparatus, protocol, and data collection, and provided a detailed outline of the analysis methods I used to interpret the collected data.

Chapter 5

Results

This chapter presents the results of the study in two parts: the statistical analysis of the quantitative data, and the thematic analysis of the qualitative data.

5.1 Quantitative Analysis

This study collected three types of quantitative data: statistical information from the session recordings, the miniPXI survey data, and the RTLX survey data.

5.1.1 Session Recordings

The statistical information collected from the session recordings encompasses game times, the length of the rulebook task, and whether a participant won or lost each game.

Time Data

Since the rulebook is the first task across all participants regardless of their condition, the resulting data does not need to be separated into groups. The average amount of time participants spent reading the rulebook was 4m 40s (min: 1m 05s, max: 9m 25s). The sample standard deviation was 2m 05s.

Table 5.1 presents the average game times. The first section shows that losses on average were 3.5 minutes shorter than wins, with the sample standard deviations having similar values. The second section shows that Condition A on average took 2 minutes longer than Condition B. The difference in the sample standard deviations between Condition A and Condition B is also the largest in the table (44 seconds)—Condition A has the largest sample standard deviation and Condition B has the smallest. The third section shows that the second game on average took 1 minute less than the first game. Both the first and second games have similar sample standard deviations. Finally, the last row of the table displays the values for the total of all 40 games.

	Average	**Min**	**Max**	**Sample SD**
Wins	9m 45s	4m 08s	21m 31s	5m 07s
Losses	6m 14s	0m 34s	25m 34s	5m 11s
A	8m 16s	0m 34s	25m 34s	5m 41s
B	6m 19s	0m 39s	21m 31s	4m 57s
Game 1	7m 50s	2m 31s	25m 34s	5m 28s
Game 2	6m 45s	0m 34s	21m 31s	5m 19s
Total	7m 18s	0m 34s	25m 34s	5m 21s

Table 5.1: Table of game time data, sorted by: wins; condition; game index; and total.

Wins and Losses

I used the screen recordings to determine which games participants won and which games they lost. Table 5.2 shows the participant win percentages separated by condition and game, as well as the total win percentage per condition, per game, and in total. Since this is a within-subjects study design, the first game is expected to influence the second game through the additional experience. Indeed, the difference between the combined win percentages displays a slight increase from the first game to the second game; however, the the win percentage actually *decreases* amongst participants with condition ordering BA (30% to 20%).

Additionally, Table 5.3 further delineates participants wins and losses by dividing the results based on ordering: winning both games, winning only the second game, winning only the first game, or losing both games. While most participants lost at least one game, all participants who won their first game but lost their second game played Condition B first.

	Game 1	Game 2	All Games
A	20%	20%	20%
B	30%	50%	40%
Combined	25%	35%	30%

Table 5.2: Table of participant win percentages. There were 10 games per condition–game pair (40 total).

	Win/Win	Lose/Win	Win/Lose	Lose/Lose
AB	20%	30%	0%	50%
BA	10%	10%	20%	60%
Total	15%	20%	10%	55%

Table 5.3: Table of ordered participant wins. There were 10 game pairs per condition (20 total).

5.1.2 miniPXI Survey

Twenty participants completed two miniPXI surveys during their sessions: one for Condition A, and one for Condition B. The eleven measures are mapped to a 7-point Likert scale ranging from -3 (strongly disagree) to 3 (strongly agree). I transferred the data from Qualtrics to a spreadsheet, and began by comparing the two datasets with boxplots created in RStudio (Figure 5.1). Upon visual inspection I found differences between Condition A and Condition B. Since the datasets are independent and paired, I then used Wilcoxon signed-rank tests to validate these visual differences. This test hypothesizes that the median of the differences between a pair of measures is 0; if the test result shows statistical significance ($Z < -1.96$ or $Z > 1.96$; $p < 0.05$) then the null hypothesis can be rejected.

The results of the Wilcoxon signed-rank tests are shown in Table 5.4. Measure AA has a statistically significant difference, and therefore the null hypothesis that the audiovisual appeal of Condition A is the same as Condition B can be rejected. However, the results of all other measures are not statistically significant; in these cases the null hypothesis that there is no difference in player experience between Condition A and Condition B cannot be rejected.

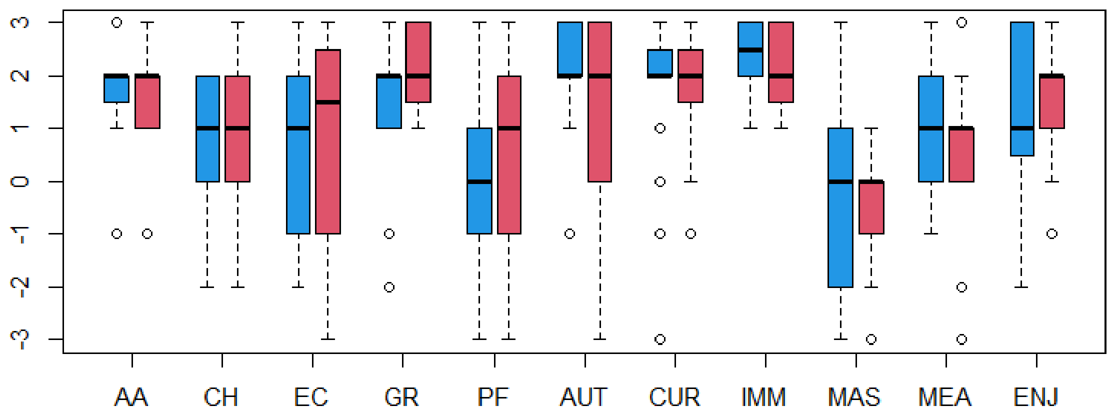


Figure 5.1: Boxplots of the miniPXI survey results. The first bar in each pair (blue) represents Condition A; the second bar (red) represents Condition B. Each pair of boxplots corresponds to one of eleven measures: Audiovisual Appeal (AA); Challenge (CH); Ease of Control (EC); Clarity of Goals (GR); Progress Feedback (PF); Autonomy (AUT); Curiosity (CUR); Immersion (IMM); Mastery (MAS); Meaning (MEA); and Enjoyment (ENJ).

	A		B		Variables	
	Median	SD	Median	SD	Z	p
AA	2.0	0.933	2.0	1.234	**2.442**	**0.031**
CH	1.0	1.333	1.0	1.348	-0.506	0.658
EC	1.0	1.785	1.5	1.877	-0.518	0.625
GR	2.0	1.501	2.0	0.759	-1.293	0.198
PF	0.0	1.806	1.0	1.732	-1.773	0.097
AUT	2.0	1.021	2.0	1.860	1.905	0.055
CUR	2.0	1.490	2.0	1.261	-0.227	0.918
IMM	2.5	0.801	2.0	0.813	0.861	0.531
MAS	0.0	1.838	0.0	1.099	0.321	0.780
MEA	1.0	1.165	1.0	1.342	0.254	0.825
ENJ	1.0	1.609	2.0	1.031	-1.594	0.138

Table 5.4: Table of miniPXI statistical results by condition (median and standard deviation), as well as the Wilcoxon signed-rank test results. Statistically significant results are bolded.

5.1.3 Raw NASA-TLX Survey

Twenty participants completed two RTLX surveys during their sessions: one for Condition A, and one for Condition B. The six scales are mapped to a 21-point scale ranging from 1 to 21. I transferred the data from Qualtrics to a spreadsheet and remapped the data from 0 to 100 with the formula $y = (x-1)\times 5$. This is because the RTLX does not assign a weight to each scale, and therefore the data can be linearly remapped. I then compared the two datasets with boxplots created in RStudio (Figure 5.2). Upon visual inspection I found some differences between Condition A and Condition B. The datasets are independent and paired, so I again used Wilcoxon signed-rank tests to validate these visual differences.

The results of the Wilcoxon signed-rank tests are shown in Table 5.5. None of the scales show statistically significant results, so the null hypothesis that there is no difference in task load between Condition A and Condition B cannot be rejected.

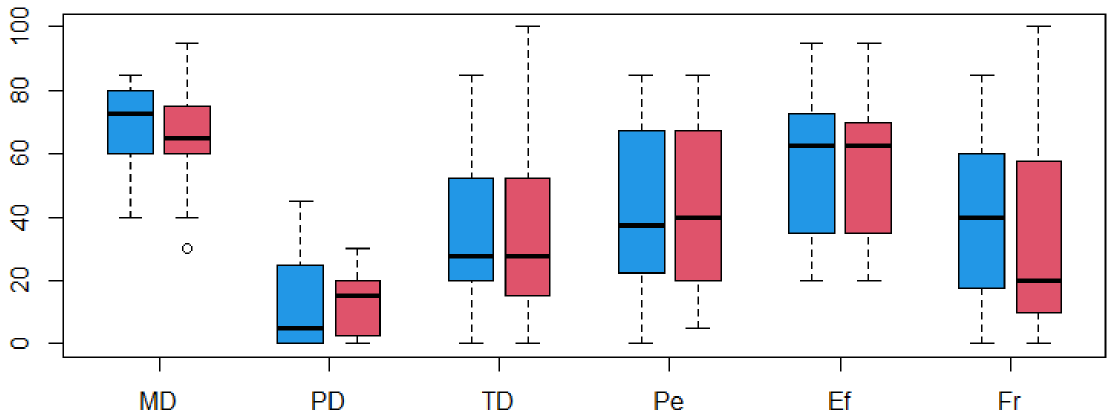


Figure 5.2: Boxplots of the RTLX survey results. The first bar in each pair (blue) represents Condition A; the second bar (red) represents Condition B. Each pair of boxplots corresponds to one of six scales: Mental Demand (MD); Physical Demand (PD); Temporal Demand (TD); Performance (Pe); Effort (Ef); and Frustration (Fr).

	A		**B**		**Variables**	
	Median	SD	Median	SD	Z	p
MD	72.5	13.851	65.0	17.128	1.526	0.131
PD	5.0	14.945	15.0	10.324	0.948	0.418
TD	27.5	24.000	27.5	27.447	-0.178	0.883
Pe	37.5	26.403	40.0	25.236	0.843	0.413
Ef	62.5	21.831	62.5	20.601	0.431	0.681
Fr	40.0	25.969	20.0	27.970	1.033	0.315

Table 5.5: Table of RTLX statistical results by condition (median and standard deviation), as well as the Wilcoxon signed-rank test results.

5.2 Thematic Analysis

I identified nine main themes within the participant data (summarized in Table 5.6). Themes 1–6 address **RQ1**, themes 7–8 address **RQ2**, and theme 9 addresses **RQ3**.

Themes	**Subthemes**	
(1) AI-supported suggestions in video games are usable	*Accurate AI-supported suggestions are trustable*	
(2) Suggestions remove players' incentive to actively learn	*"Give a fish" vs "teach to fish"*	*Reduced cognitive load reduces learning*
(3) Upfront guidance in onboarding is needed for learning	*Structure improves onboarding experiences*	
(4) Onboarding shouldn't overstay its welcome	*Onboarding should still be fun*	*Bad onboarding is a reason to quit*
(5) Players want to understand why an AI makes a suggestion	*Transparency increases trust in AI*	
(6) Onboarding is not one-size-fits-all-players	*AI can provided tailored onboarding experiences*	
(7) Suggestions are temporary supports	*Suggestions support discovery*	*Players want to rely on own abilities*
(8) Voluntary suggestions enable freedom	*Restrictive onboarding frustrates players*	
(9) Players prefer to learn through lived game experience	*Losing is learning*	*Using prior experience with similar games*

Table 5.6: Table of the nine main themes extracted from the data alongside their related subthemes, separated by research question.

5.2.1 Theme 1: AI-Supported Suggestions in Video Games are Usable

Participants believed AI-supported suggestions in the context of video games were inherently usable—i.e., that a player could follow a suggestion to achieve a desired result in-game. This concept of usability ranged from expectations of general information quality (P18) to assumptions around how an AI system interacts with the game itself (P16). Participants generally conceptualized AI systems as taking "inputs", such as game rules or contextual information, and providing "outputs" for players to use.

> *"Well, if you like, give them enough information, they will produce like the **high quality answers**, right?"* (P18)

> *"I would assume the suggestion it was giving me is a **good suggestion** for the specific state of my game."* (P16)

The concept of accuracy was also prominent. Participants felt that since an AI system is capable of manipulating large quantities of data extremely quickly, its results would be more accurate. P11 mentioned that they expected an AI to be "more correct" than a human since it would have data to support its conclusions, which associates the ability to process information with accuracy. And, because AI systems are inherently designed to process information, this in turn further associates expectations of accuracy with AI systems.

> *"I would feel like because we have such an expectation of AI to **always be more correct** or always showing the most statistically proven method, I would expect it to be pretty bang on."* (P11)

As well, the context of video games influenced participants' perception of AI systems. Participants felt that if a system was designed for a game and limited to that game's context, there was little risk of it affecting other aspects of their lives (e.g., healthcare or academics). No participants were concerned about using Joker's AI-supported suggestion system; the context of video games did not increase any participant's concerns with AI, and in some cases, such as P13, even decreased them.

> *"If I look at these games, I think I trust the system here because it's not something that is related to me academically or it is **not something which is serious** to me. This is for **fun**, right?"* (P13)

Subtheme: Accurate AI-Supported Suggestions are Trustable

Participants also felt that the competency of AI-supported suggestions could influence their trust in the overall system. Some participants thought that users would trust an AI system over their own intuition if they knew the system was a modern—or "advanced"—AI system. There was a prevalent perception among participants that recent advancements in AI technology, such as the oft-mentioned ChatGPT,[1] demonstrate how the abilities of a computer can easily surpass a human's.

> *"I definitely feel like if you tell users like, 'this is an **advanced** AI giving suggestion' that they're probably **much more willing to trust that** versus their own intuition."* (P14)

Since this idea was tied to the assumption that an AI system inherently knows more than a new player, participants felt that low-quality suggestions from an AI would negatively impact their trust in the system. While most participants were cognisant of the fact that absolute perfection isn't a realistic expectation for AI-supported suggestions, they still expected accurate suggestions anywhere from 60%–90% of the time. Many participants felt that an inaccurate suggestion would cause them to distrust future suggestions, and some participants indicated they would stop using the system altogether if the AI directly caused them to lose.

> *"[Imperfection] definitely **lowers my trust in the systems**, and that might be a reason that next time I would double check the answer I get from the AI."* (P7)

This demonstrates a relationship between the accuracy of an AI-supported suggestion and a player's level of trust in the system, which in turn relates to perceived usability: if an AI-supported suggestion is *accurate*, then a player is more willing to *trust* that suggestion and *use* it during a game.

5.2.2 Theme 2: Suggestions Remove Players' Incentive to Actively Learn

Participants felt that receiving a suggestion negatively impacted their incentive to learn during the game. Some participants did not realize that the suggestions were optional and

[1] https://chat.openai.com/

followed the AI's suggestions throughout the entire game. This blind adherence to the suggestions meant that the participants did not actively interact with the game and its rules to discover the mechanics for themselves.

> *"But this time around, since the AI was showing me what kind of troops I should play or I can play,* ***it took away my curiosity*** *and I just wanted to try out whatever the AI was telling me to do."* (P9)

A few participants even previewed the AI's suggestion, commented that they did not understand why it had suggested that, and then created the unit despite their confusion. For example, P5 read the description of the unit the AI suggested and did not understand its ability. Despite this, they still created the unit, even though they had no idea how to use it.

> *"This unit loops at the left and right side of the board...* ***Not exactly sure*** *what that means. Try it I guess?"* (P5)

As a consequence of this lack of active interaction, players did not increase their understanding of the game's rules by using the suggestions. P10 felt there was no benefit to having suggestions available because they did not think about the game when they followed a suggestion. Other participants echoed this concept of feeling "mechanical", with some commenting that it may as well have been the AI playing the game, not them.

> *"So, if from the beginning I see this hint, I should think, OK, I have to grab those cards and drop it here. So that's not,* ***does not benefit me at all****. So because, not me, someone told me, "OK, pick this one, pick this one drop there", that's like a mechanical.* ***There isn't any thought process in it****."* (P10)

Subtheme: "Give a Fish" vs "Teach to Fish"

Just as stated in this subtheme's titular proverb, participants felt that receiving suggestions did not set them up for future success. When they blindly followed a suggestion without thinking on their own, they did not actually learn why two cards created the resulting unit, or why the AI system had suggested that combination.

> *"While I was using the suggestion card, but* ***I didn't pay attention*** *to how they are created. So my learning experience in the creation of units doesn't improve."* (P8)

P13 compared having suggestions available to not using all of their senses; they felt that when the game did not provide them with support, they were forced to think about the game and form a proper strategy.

> *"If you have suggestions,* ***you don't use all your senses*** *and you're like, ok, I have suggestions, I'll go through, right. But if you are like, really put on your, you, you're just on your own, like, you know, there's nothing, right. So then you actually start thinking that, ok, what should I do if I have to go in?"* (P13)

Subtheme: Reduced Cognitive Load Reduces Learning

Regardless of any potential negative impact on the learning process, participants agreed that the presence of AI-supported suggestions reduced their cognitive load. They explained that they experienced less mental effort when they were provided a suggestion because they didn't need to think about the rules beforehand.

> *"The effort I took for the first game, like without AI, it's* ***more*** *than like, with AI."* (P3)

> *"I'd say [the effort] was less because I had something to start with and then I, I can tweak it however I want. Yeah,* ***the effort would have been less****."* (P5)

Having a suggestion as a starting point also helped some participants with the decision-making process—they would try out the suggestion, then play around with the suggested cards before deciding what to do. Many participants felt the game was *easier* when provided with additional support, despite the fact that the difficulty had not changed.

> *"I would say it's way* ***easier*** *and with* ***lower cognitive demand****, if you decided to just follow the prompts."* (P4)

However, while reducing cognitive load was one of the intended goals of Joker's AI-supported suggestion system, this subtheme also highlights a potential risk of *too little* cognitive load: if participants do not need to think about their actions in-game at all, then they are not actually learning.

5.2.3 Theme 3: Upfront Guidance in Onboarding is Needed for Learning

A majority of participants expressed a desire for upfront guidance during the onboarding process. Even when participants wanted the freedom to explore a game, they also pointed out that it is impossible to properly explore without direction. In the context of Joker with its rulebook-only guidance, some participants immediately opened the rulebook when the game started even though they had just finished reading it moments prior. This lack of guidance in the initial moments of gameplay left participants—such as P16—wondering what they were supposed to do first.

> *"I mean, I'd say I had freedom but it was kind of useless because I didn't know how to strategize or **what to do**. So I feel like I could do anything but it wasn't really going anywhere."* (P16)

Furthermore, participants used the AI-supported suggestions not only as instructions to follow, but also as encouragement to interact with the game. Displaying the suggested cards visually "pulsing" helped some participants understand how to use their cards to create a new unit. Some participants, such as P7, mentioned that demonstrating the suggested action helped them understand the textual descriptions in the rulebook.

> *"And after starting the game, I was instructed to, to use some cards. So, this were helpful to, to, **to get started with the game** as this is my first time."* (P15)

> *"I think I, I read them in the, instruction part for the first time, but I didn't understand them well. But when in the second round **it showed me** with the cards and in the game, I **learned** better what the rules meant in the instruction part."* (P7)

Subtheme: Structure Improves Onboarding Experiences

Some participants expressed a need for not just upfront guidance, but also a greater amount of structure during the onboarding process. Most participants expected Joker to have a tutorial, and some participants mentioned subtractive onboarding methods described in Chapter 2, such as reducing the amount of units available to them, or giving the enemy a specific set of units for them to learn how to play against first. The lack of structure in Joker left many participants overwhelmed with information and provided them without any tools to make their cognitive load more manageable.

> *"Just because I feel like a* ***more guided tutorial would have made a lot more sense****, versus just bombarding you with all the information and expecting you to remember it as you're playing the game."* (P14)

> *"So yes, it's maybe it's a little bit* ***too much****. So what I would do maybe as a game developer, so you can sort of let your user* ***gradually open new units****. You don't let them use all of the units in their first game. So I start with like five basic, basic units or 10 maybe. And then let them use more."* (P12)

Note that *structure* is used here as a different concept than *freedom*; participants mainly felt a lack of freedom in games when they could not do what they wanted in a particular moment—i.e., they could not skip a tutorial, or they had to perform a specific action to proceed rather than have the freedom to find a way forward on their own. The participants' idea of structure instead refers to the method of translating information from the game to the player, rather than the concept of restricting a player's freedom.

5.2.4 Theme 4: Onboarding Shouldn't Overstay Its Welcome

Participants showed a strong preference for shorter onboarding, and expressed a desire to get into "real gameplay" as soon as possible. Some participants wanted onboarding to last as short as 1–2 minutes, and other participants mentioned they would skip tutorials if the option was provided to them.

> *"I don't like to have it too much or boring because it makes me boring. So it be— eh,* ***one or two minutes*** *and have the, as much as information I can get."* (P8)

However, participants also recognized that different genres of games have different onboarding needs. P17 and P20 both contrasted the length of onboarding with the estimated length of gameplay, and did not mind a longer onboarding process if they felt it was necessary to understand a game's mechanics.

*"I guess it **depends on the genre of game** for me, because if I'm playing like a sports game or like a car simulator, I, I wanna get into a car **as quick as possible**. But if I'm playing like, an RPG that's gonna take 100 hours. I don't mind having to sit through like 10, 15 hours of onboarding if it explains all the mechanics in depth."* (P17)

*"I think **it depends** on how long the game is. Like, if you have a sense of familiarity how long the game will be, then it makes sense it takes **relatively** a bit of time."* (P20)

This highlights the idea that onboarding is proportional to the game itself—participants did not expect every game to dedicate the same amount of time to onboarding, but rather to have an onboarding experience relative to the game's complexity and length.

Subtheme: Onboarding Should Still be Fun

Boredom was frequently mentioned in relation to the length of onboarding. Many participants who wanted shorter onboarding experiences attributed this desire to becoming disengaged when trying to finish a game's tutorial.

*"Actually, it was very good, but it was **a bit too long** for me. So I, looked at the first steps. But after like the fifth or sixth step I was **bored**, so I just skipped the others."* (P7)

Participants wanted onboarding to still feel like a game, not a classroom, and wanted the learning process to feel as fun as possible within the constraints of learning the game's mechanics. Some participants pointed out that they played games to *play a game*, not to read. They preferred interactive teaching methods over text-based learning methods—such as Joker's rulebook—to increase their sense of engagement during the learning process.

*"Obviously, learning should be more important during the onboarding process, but there should, **it should be a bit fun** as well."* (P16)

*"I think on, onboarding process, **first it has to be fun**. Like, reading the rule book has no joy in it."* (P4)

Subtheme: Bad Onboarding is a Reason to Quit

Finally, participants felt that if a game's onboarding was too boring or too long, they might quit playing the game altogether. Some participants, such as P4, treated onboarding as a preview of the actual game—a boring tutorial implied a boring game, and therefore they could not trust that their experience would improve after they completed the onboarding process.

> *"If it's, if it's more than five minutes, **I don't think I would play it**."* (P2)

> *"I would say like, it has to be fun, or **people may get worried** like, people may just choose not to believe in the game is going to be good and just **abandon** that."* (P4)

This demonstrates the impact of onboarding on a new player's initial impression of game, as well as the importance of creating engaging experiences that attract and retain new players from the start. Games must make sure that the onboarding process provides players with an experience that makes them want to play *more* of the game, not less.

5.2.5 Theme 5: Players Want to Understand Why an AI Makes a Suggestion

The most frequent comment from participants regarding Joker's AI-supported suggestion system was that it did not tell them *why* it made a suggestion. Participants felt that receiving a suggestion with no explanation did not help them learn the game—even if they wanted to understand it, the game itself did not provide them with an avenue to learn.

> *"I guess if the AI does not give **the reason why** he, he's making this suggestion, it will not help the player to, to know how to play it in a good way."* (P15)

> *"I usually would like the, the AI to suggest them, but as well to give those recommendations with **why** or which or on what basis that I would choose those."* (P2)

P11 emphasized that understanding an AI would allow them to evaluate their own understanding of a game alongside the AI's suggestion. They wanted to learn actively, but without access to enough information regarding their decision, they were unable to make comparisons between their strategies and the AI's. This again plays into the idea of *giving* versus *teaching*: suggestions without explanations not only reduced players' incentive to think for themselves, but additionally did not support the players who wanted to learn from the suggestions in the first place.

> *"I always like to know like **why** systems are trying to suggest for me to do what they want me to do. Just because I like to see like what kind of information they're seeing and why would that one— why are they suggesting it versus what I'm gonna do? So, **is their suggestion actually better** than what I want to do? Or can I just still screw around with what I'm doing? And can we still come to the same conclusion?"* (P11)

Subtheme: Transparency Increases Trust in AI

Many participants also thought that an increase in transparency would increase their trust in an AI system. They felt that if they could understand the "thought process" behind the system's suggestion, then they could compare it with their existing knowledge to determine whether the AI's suggestion was trustable.

> *"If I know like **why they suggested these cards** then maybe if they are like trying to help me, then I, I can like **trust** them."* (P3)

> *"I'd say my trust in the system would be built on if it can explain to me **why you gave that suggestion**, which it didn't. So, yeah, like you would build trust only once you **understand** what the system is trying to do for you, right?"* (P5)

Even if participants thought that an AI knew more than them as a human, they felt they could only increase their trust beyond their initial threshold if an AI system provided reasoning for its decisions. This shows that players' desire for transparency in an AI-supported system is not just to increase their knowledge of a game, but also so they can build a greater level of trust in that system overall.

5.2.6 Theme 6: Onboarding is Not One-Size-Fits-All-Players

Participants recognized that their individual onboarding preferences shouldn't be generalized across different types of players. Some participants pointed out that other players may prefer to read rulebooks, even if they personally didn't, and that it was important to support as many types of players as possible. Other participants brought up accessibility—such as the impact of dyslexia on text-based onboarding—and felt that giving players access to multiple onboarding methods allowed them to choose which method worked best for them.

> *"**People are different**. Some, some people want to, I don't know, get the result faster and they don't mind having help, but some people want to achieve their results themselves like without help."* (P19)

> *"I feel like I'm a, like, **visualized learner**, better than like reading. And reading takes quite a long time for me. So, yeah, video may help."* (P3)

Subtheme: AI Can Provide Tailored Onboarding Experiences

Furthermore, participants also felt that AI had the potential to make onboarding more dynamic. They pointed out that older methods of onboarding were static, and therefore could not cover all potential needs new players might have. On the other hand, using AI to personalize the onboarding experience of each player would reduce the friction from games not meeting players' onboarding needs and improve their overall game experience.

> *"In like older sort of games, you would have like a very like constrained tutorial where it's like, this is a hardcoded scenario and you just go through it. But like with AI I feel it can create a much more **dynamic setting**."* (P5)

Some participants also liked the idea of an AI guidance system that was tailored to them as an individual. They described concepts similar to *advisors*—the real-time onboarding method—where an AI system would learn alongside them as they played in order to make suggestions that matched their playing style. These concepts would be an extension of the AI-supported suggestion system implemented in Joker, where the system could adapt to situations involving both the game and an individual player.

> *"AI should note about the **different strategy** and guess what is your strategy, and suggest based on that."* (P8)

> *"But if you use AI it possibly can like just **personalize instructions to the player**. Because if you see this player like losing every time, maybe you can give them like other instructions, because usually in games, they are just like scripted, like constant instructions. And yeah, so I believe it can help, because **every player is different** and they need different instructions."* (P18)

Not only do players want onboarding to be tailored at a broad level—such as through different learning methods or accessibility controls—but they additionally want onboarding to be tailored at the narrow level of individual player differences. This desire for dynamic onboarding demonstrates a player need for finer degrees of personalization that only AI-supported systems can provide.

5.2.7 Theme 7: Suggestions are Temporary Supports

Participants felt that suggestions should be implemented in games as an additive onboarding method—i.e., a temporary "scaffolding" that should be removed once a player surpasses the need for additional support. They pointed out that existing games often use suggestions only during onboarding, and expressed that they wouldn't feel like they were playing the "real" game until the suggestions were removed. Some participants even preferred it if suggestions could be removed after playing for a few minutes, while others wanted to disable suggestions as soon as they possibly could.

> *"I just want the guidance the **first time** that I play the game, I don't want it even the second time that I play."* (P19)

> *"I think like, because like in general, most games that we play, [suggestions] **don't really occur in the gameplay** as much. But if so, they usually occur, like, right in the beginning of the game just for like a few minutes and then they're gone."* (P6)

Subtheme: Suggestions Support Discovery

As well, a majority of participants felt that suggestions were most useful at the beginning of a game because they could guide players if they didn't know what to do next. Even if a participant didn't understand why the system made them a suggestion, they primarily wanted access to a form of support if they got stuck. Providing guidance through decision paralysis subsequently helps players discover more of the game, rather than grow frustrated with their perceived lack of progress.

> *"I guess there, I mean, having them is useful because **when you have no clue what to do**, you may think that OK, maybe there is a reason that the this computer is suggesting such a move. So I'll proceed with it."* (P1)

Some participants felt that suggestions could promote curiosity alongside discovery if they were presented in an indirect manner. P19 suggested that rather than giving players an answer right away, suggestion systems could instead nudge players in the right direction and "make [them] think". In this way, suggestions would lead players to discover more of the game instead of just presenting new discoveries in their entirety.

> *"In the direct suggestions, you know, it helps you, so you just click on the cards, but if it's indirect, **it would make you think** and when you think, you remember those and also maybe understand exactly what happens and would, you would learn much more, I guess."* (P19)

Subtheme: Players Want to Rely on Own Abilities

Participants' main motivation for wanting temporary suggestions was the desire to rely on their own abilities to succeed in a game. Many of the participants who enjoyed Condition A more than Condition B attributed the difference to their perceived agency—their actions were the sole reason why they won or lost a game; an AI-supported system had no influence over their results. Some participants felt that even when they won a game, they didn't really deserve it if they had used the AI's suggestions.

> *"I think it gave me better feelings to win **without going through the suggestions** that, that it gives me."* (P2)

> *"I should feel that **I am in charge** of making decision as [a] player."* (P10)

Furthermore, participants felt that the presence of suggestions meant they weren't actually the one playing the game. Even if an AI system had the ability to provide them with the "right" move to make in a scenario, they did not just want to be told what to do for the entire game.

> *"I think that would be the, the thing for me is just being like, well, **how much am I playing the game** versus being told how to play it?"* (P11)

> *"I want it to give, give me a good move, but I also **don't want it**."* (P16)

Participants preferred to use their own abilities to interact with a game, even if they did not achieve optimal results, since the challenge of testing themselves and improving their skills was a more important motivator than just winning the game. This lack of agency wasn't a lack of freedom for players to do what they wanted, but rather the removal of what incentivizes players to play video games in the first place.

5.2.8 Theme 8: Voluntary Suggestions Enable Freedom

Many participants felt that optional suggestions could provide them with more freedom during the onboarding process than traditional tutorials. Some of them considered this freedom as "freedom of choice", because it was up to them whether they wanted to follow a suggestion or not. Others considered this freedom as "freedom of information", and felt that context-based suggestions provided them with increased access to information about the game.

> *"It's just a suggestion, you can, you can use it, like you can follow it and you can not follow it. So, **it's your choice**."* (P12)

> *"But in this case, they show you like possible variants and I believe it's like the best variants which you can have but you still can like use another card or like change, I don't know, like sets of this equation. So I believe **it gives you freedom**, but it also like shows you the best variant."* (P18)

In both of these cases, participants noted that giving them the ability to decide how to use suggestions helped to distinguish suggestion systems from tutorials—they perceived prompts in tutorials as mandatory, while suggestions were perceived as optional.

Subtheme: Restrictive Onboarding Frustrates Players

Participants also pointed out that onboarding—especially traditional tutorials—could become frustrating when it took away too much of their freedom. This frustration often occurred in situations where players did not have the freedom to make their own decisions, or when players were forced to complete onboarding even if they already felt ready to play.

> *"If it is so limited, so constrained, so then it becomes so, I mean, I mean* ***bothering the players*** *so that like, doesn't work or, and then may be* ***frustrating****."* (P10)

> *"I do expect there to be an* ***option****, a question asking how familiar are you already with these kind of games. And if there's not, and they treat everyone as completely new to the game, I kind of get* ***frustrated****."* (P4)

Some participants also felt that while suggestions were more freeing than mandatory choices, the fact that the prompt automatically appeared at the beginning of their turn provided them with a lesser amount of freedom. They felt that giving them the suggestion right away did not give them the opportunity to think for themselves.

> *"So I was a little bit* ***frustrated*** *at the beginning with the [...] suggestions. Yeah. I tried to not do it like, on purpose even if it's like, it's a, it's a really good suggestion, but like,* ***I want to feel free to play*** *it, not to be like demanded to do this move."* (P2)

In these cases, the most common suggestions for an alternate implementation method were for the AI system to only provide a suggestion if the player actively requested it, or for the suggestion to only appear after a certain amount of time had passed.

> *"Maybe there should be you know like the* ***option button****. Where you show like a turn hint, click on hint and show you something, [...] or put something like a hint here or maybe after a* ***couple of seconds*** *if something is not happening and then hints appear."* (P10)

These types of suggestions still retain players' freedom to access additional information, but also increase their freedom in choosing whether they want to use a suggestion in the first place. Maintaining the feeling that suggestions are *voluntary* across all aspects of their implementation reduces their perceived level of restriction, and in turn reduces reasons for a player to experience frustration.

5.2.9 Theme 9: Players Prefer to Learn Through Lived Game Experience

All participants preferred to learn through hands-on experience with a game. Some even preferred to jump into gameplay *before* experiencing any onboarding—in these cases, participants wanted the ability to choose when they would receive guidance. It was important for them and their learning experience that they could try to learn on their own first and foremost, but still have onboarding methods such as tutorials or rulebooks that they could return to when they needed help.

> *"But usually when I would play a game, [...] I would just* ***try it out for a couple of rounds*** *and then if I keep losing, then maybe I go back to the rule book and see what's going on."* (P17)

> *"Actually, to be honest, I don't read rules, I just like* ***start playing*** *and then like just figure out how to play."* (P18)

Additionally, participants frequently mentioned interactive learning methods such as trial-and-error, improving through practice, and training modes that targeted specific gameplay mechanics as some of their preferred methods of learning a new game.

Subtheme: Losing is Learning

Most participants did not perceive failure as negative during the learning process. While they were still primarily motivated by the desire to succeed, they pointed out that losing provided them with an opportunity to assess their knowledge of the game and improve—for example, P13 retained a sense of accomplishment in their loss because they learned more about the game in the process, while P19 understood why they had lost and knew they could apply that knowledge to future games.

> *"This time I felt like, OK, even if I lost, I had some sense of accomplishment that OK,* ***I understood the game better*** *maybe, yeah."* (P13)

> *"Even if I lose like without suggestions, I feel like, OK, I was, it was, I was doing it.* ***I totally understood*** *what I did."* (P19)

Some participants also thought of failure as a natural part of the learning process, and felt it was important for new players to accept that they might fail if they wanted to use their failure as a learning experience.

> *"So it's good for you to have the set up of someone suggesting something as a beginner's approach. But when you're just understood what you can do about it, try it yourself, even if you fail, that's ok because* ***you will learn when you fail*** *the game."* (P20)

Subtheme: Using Prior Experience With Similar Games

Finally, many participants mentioned using their knowledge of existing games when approaching new games for the first time. P17 mentioned that some games expect new players to already have some familiarity with basic game controls, especially if the relevant knowledge is broadly applicable across game genres, such as with camera controls.

> *"Newer games expect you to have* ***some proficiency*** *in terms of like moving around and controlling the camera and so on."* (P17)

With Joker, participants drew upon their knowledge of existing games that were connected to Joker's familiar game elements. The most frequent comparisons were to playing cards and to chess, and some participants commented on how they applied their existing knowledge of these elements to learn an unfamiliar game.

> *"It was mostly based on the suits of playing cards, the normal playing cards. So it was* ***easy to grasp****."* (P9)

Additionally, some participants used their prior experience to analyze their performance in Joker. P1 pointed out that if they were playing chess, they wouldn't have made the move that caused them to lose the game.

> *"I play chess and the, you know, moving patterns were so much* ***similar*** *to, to chess and in a, in a chess game, I wouldn't have done this."* (P1)

This shows not only recognition of a familiar mechanic in an unfamiliar game, but also successful application of knowledge to learn what to do—or in this case, what *not* to do—in a future game. Lived game experience is therefore not limited to individual games, and instead is a collective experience that players apply to all games they play.

5.3 Summary

This section presents the analytical results of both the quantitative and qualitative data collected in this study. While statistical analysis of the survey data did not identify many significant results, thematic analysis of participant interviews identified nine major themes present within the dataset. I used participant quotes from the encompassed codebook to support my analysis of these themes, and presented my findings in the context of an overarching narrative approach.

Chapter 6

Discussion

The results of Chapter 5 demonstrate essential findings regarding player experience and AI-supported onboarding systems. In this chapter I further discuss these findings in the context of my research questions, existing literature, and future implications.

6.1 AI-Supported Suggestion Systems and Player Experience

People who play video games are no strangers to AI. As discussed in Chapter 2, video games have implemented various types of AI-supported systems for decades [48]. This familiarity lends support to the findings of **Theme 1**, where personal experience with AI in existing games influenced participant expectations regarding Joker's suggestion system. Video games are "for fun", so any implementations of AI-supported suggestion systems within the bounds of a video game are perceived as novel—a system designed to fulfill a limited purpose, with no potential to impact a player's life outside of its context.

6.1.1 The Trust Experience

Since players inherently expect an AI system in a game to function "properly"—be it an in-game opponent, a procedurally-generated level, or a suggestion system—they enter games with an existing level of trust dependent on the quality of an AI-supported system. **Theme 1** describes the impact of low-quality results on trust in AI-supported suggestion systems, and demonstrates that while players may start with an existing quality-trust relationship, even a single low-quality output has the potential to destroy it. However, the opposite is also true: starting with an existing amount of trust in an AI's quality means that video games do not need to take the time to build trust in the same way as other AI implementations. The rise of generative AI in recent years has sparked much debate regarding the legality, ethics, and privacy of AI-supported systems [20], but at the moment the three main types of AI in video games [50, 47, 49] generally do not seem to fall under the same scrutiny as language models or image generators.[1]

Theme 1 also demonstrates an amount of trust on the human-to-human level between players and game designers. Players expect designers to create good games; designers expect players to interact with their games. It would be actively detrimental for designers to create an AI-supported suggestion system that causes the player to lose the game—the players waste time following the suggestions, the designers waste time making the system, and the critically-important first-time experience [12] is wholly negative. This sort of symbiotic relationship between players and designers again reinforces trust, but also provides a possible explanation as to why video games are held to different standards than other AI-supported systems. If players perceive that video games have less of an incentive to violate their ethical standards than other implementations of AI, then there is also less of a reason to distrust implementations of AI in video games.

6.1.2 The Cognitive Experience

The AI-supported suggestion system implemented in Joker is rooted in cognitive load theory [31], and specifically in the idea that just-in-time suggestions can reduce the amount of information new players need to manage during the game. **Theme 2** identifies a clear relationship between reduced cognitive load and the presence of the AI-supported suggestions, where some participants even perceived a difference in difficulty between conditions that did not exist. While the quantitative results were not statistically significant, the RTLX survey still shows some relevant differences between the two conditions: the median of the *Mental Demand* and *Frustration* scales were lower in Condition B than Condition A (Table 5.5). This aligns with the qualitative findings in **Theme 2**—managing large amounts of information is the core concept of cognitive load [29], which contributes to mental demand and can cause frustration if a player cannot handle the load.

However, the win percentage data shown in Table 5.2 and Table 5.3 also supports the second idea of **Theme 2**: that suggestions did not actually help participants learn.

[1]Now, the crossover of generative AI and video games, on the other hand...

The win percentage *decreased* amongst all participants who played Condition B first (30% to 20%), and all participants who won their first game but lost their second game also played Condition B first. By contrast, the win percentage of participants who played Condition A first *increased* (20% to 50%). This suggests that participants who relied on suggestions in their first game did not actually learn from them, and could not translate their experience into the second game when they were on their own. Again, while the suggestions did demonstrate the ability to reduce cognitive load, they also demonstrated that *some* cognitive load is necessary for players to learn. It is still important to make sure new players can manage the amount of information they need to process, but AI-supported suggestion systems must instead find a careful balance between *help* and *hindrance* during the learning process.

6.2 Player Expectations for Video Game Onboarding

Theme 3 and **Theme 4** describe what players expect from an ideal onboarding experience. **Theme 3** identifies that guidance is a necessary part of onboarding; indiscriminately sacrificing guidance in the name of player freedom is not beneficial to the learning experience. Players will always need some amount of upfront instruction—though not necessarily everything—so that they can familiarize themselves with the basics of the game. Despite their more restrictive nature, tutorials [44] can meet this need for guidance when they are well-designed, and additionally are what players expect to receive at the beginning of a game. However, **Theme 4** also cautions that overly-restrictive or boring onboarding can cause players to quit. Even though players recognize the importance of upfront guidance, they also place importance on games delivering guidance in an engaging and interactive way. A game's onboarding method acts like a trailer for the "real" game, so if a player does not enjoy their onboarding experience, they have no reason to believe that the real game will be any different.

6.2.1 Structure is Support

Theme 3 also demonstrates the importance of structure during the onboarding experience. Players seek structure from all types of onboarding: the additional instructive support of additive onboarding [16]; the simplified learning environments of subtractive onboarding [28]; and the contextually-relevant advice of real-time onboarding [36] all provide forms of structure to improve a player's learning experience. This structure subsequently familiarizes players with the rules of a game and builds their confidence with the application of mechanics. Additionally, structured information can help video games provide players feedback on their progress. Without some type of support method, it is difficult for players to track if they know "enough" about the game to be successful. The miniPXI survey results (Table 5.4) show that the median response of the *Progress Feedback* measure is higher in Condition B than in Condition A, which supports the idea that structured guidance provides an avenue for player feedback.

6.2.2 The Power of Short and Sweet

Players also value onboarding that is *fast* and *fun*. **Theme 4** identifies boredom as the primary reason that players skip onboarding methods such as tutorials—they either take too long or aren't as engaging as the actual game. When learning feels equally as fun as gameplay, players are much more willing to engage with onboarding methods. Furthermore, players prefer brief onboarding methods when they are able to learn the rest of a game by just playing it. If the complexity or length of a game demands a more in-depth onboarding experience, players are willing to sit through a longer onboarding session to make sure they are prepared to play. While there is no exact method to calculate what length of onboarding is most appropriate, it is important to make sure that a game's onboarding experience is kept proportional to both its complexity and its overall length.

6.3 Player Expectations for AI-Supported Onboarding Systems

Players have two key expectations for AI-supported onboarding systems: *transparency*, discussed in **Theme 5**, and *personalization*, discussed in **Theme 6**. These expectations demonstrate an optimistic outlook toward the implementation of AI in video game onboarding; players strongly believe that AI systems are capable of providing them with support that is tailored to their own needs as an individual. However, players are also hesitant to blindly trust an AI, even if they feel that the system is capable. Providing a reason for the output of an AI—such as *why* an AI system made a suggestion—not only builds a player's trust, but additionally creates an avenue for them to learn.

6.3.1 Learning From "Why"

Methods of real-time support—such as advisors [36] and coaches [28]—provide players with additional information about their actions during a game. **Theme 5** demonstrates that players strongly prefer that AI-supported suggestion systems provide a reason alongside each suggestion. If they are told to make an action, they want to know *why* the action is good and *evaluate* if it connects to their existing knowledge in other ways. Without the *why* component, it's impossible for players to perform a complete evaluation—are they moving a piece to set up for checkmate many turns in the future, or only to get it out of danger? Providing a clear reason also helps players better understand if the AI is making suggestions that align with their own plans. However, since providing a reason for a suggestion also increases the amount of information available to a player, it is important that explanations have an appropriate level of complexity [43] so as to not contribute to cognitive overload. The optimal level of transparency in an AI-supported onboarding system should improve a player's learning experience, not impair it.

6.3.2 Dynamic Onboarding

Findings from **Theme 6** also show that players want an increased level of personalization in their onboarding experiences they currently do not receive from static methods. Players believe AI-supported player profiles [46, 49] have the most potential to support dynamic onboarding because AI-supported onboarding systems could adapt to how they play and provide them with guidance methods that match how they best learn. Tailoring onboarding to players as individuals would also alleviate frustration caused by static onboarding methods—an AI system can consider player differences such as experience and aptitude that static onboarding systems cannot.

Players additionally believe adaptive opponents [39, 37] can provide a more dynamic onboarding experience. One restriction of static onboarding involves opponents that are programmed to respond a specific way; players are therefore forced to make certain actions to continue through a linear chain of events. Adaptive opponents instead have the potential to respond to any of the player's choices, while still finding opportunities to teach the required actions. This not only reduces the initial level of restriction present in a game, but also increases a player's sense of agency during the onboarding process.

6.4 Player Perception of Suggestions

The findings of **Theme 7** show that players perceive suggestions as temporary supports. Even though suggestions themselves are a real-time support method, the temporary implementation of a suggestion system is also similar to additive support methods such as scaffolding. The end goal of scaffolding is to eventually remove the extraneous information once a player is ready to play on their own [16, 18]; temporary assistance from a suggestion system in the early stages of a game fulfills the same purpose. Another interpretation of suggestion systems is more similar to the advisor method: players feel suggestion systems act similarly to when friends try to help them learn a new game. However, this also shows the desire for temporary support, since players do not expect someone—or *something*—to hold their hand the entire time they play a game.

Players also feel that suggestions support discovery during their initial game experience. Onboarding cannot be too hands off without risking that players become confused about what to do next [18], so suggestions alleviate this risk by providing a possible direction for players to investigate. Players appreciate assistance the most in situations when they actually need it, but additionally prefer implicit directions when they are still discovering a game. Indirect suggestions are very important when maintaining players' sense of involvement in the discovery process, as players feel outright directions that tell them "this is what you should do next" prevent them from engaging with discoveries in the first place.

6.4.1 Agency is the End Goal

Theme 7 also identifies the clear prioritization of player agency within video games—players want to feel that they are the one in control of their in-game actions, not the AI. Boxplots of the miniPXI data (Figure 5.1) also show that the *Autonomy* measure supports this theme: Condition B's data covers a wider range of values, while Condition A is largely concentrated within the "agree" to "strongly agree" range. This suggests that players experienced a reduced sense of autonomy when they received suggestions for in-game decisions, even if they had the option not to follow it.

Agency is part of what motivates players to engage with games at all. The appeal of video games is that they are interactive media—unlike static media such as movies or books, the player is directly involved with what happens in a game. When the game itself tells players exactly what to do and when to do it, the appeal of interaction is removed entirely. Even when players are "stuck", or on a losing streak, or confused about what to do next, they want to use their own abilities to overcome these challenges. Success is meaningless if they need to rely on external support to achieve it. Player agency is therefore critical to consider when implementing support systems such as suggestions within games.

6.4.2 Preserving Freedom

Player freedom is a similar but separate idea to player agency. Chapter 2 examined how restrictive onboarding methods such as tutorials can negatively impact the onboarding experience [38, 32, 2] through reduced player freedom, especially when tutorials have mandatory interactions that prevent players from choosing what they actually want to do. **Theme 8** expands on the concept of player freedom in two parts: freedom of choice, and freedom of information. Since suggestions are voluntary by definition, players feel suggestions provide a greater amount of freedom than tutorialized support methods because they can choose what to do with the suggestion after they receive it. However, Joker's method of providing optional suggestions at the start of every turn did not increase freedom of choice; players not only need the freedom to use suggestions *how* they want, but also to receive suggestions *when* they want. Providing an optional suggestion when the player does not want it cannot increase their freedom of choice, because they did not have the choice of whether to receive the suggestion in the first place.

Players additionally feel that suggestions can provide increased freedom of information when given at appropriate times. For example, when a player needs to make a complex decision, suggestions can add valuable information for them to consider alongside their existing knowledge of the game. Preserving the freedom of information during the onboarding experience helps mitigate player frustrations with decision-making, while simultaneously enabling a greater level of access to information when players want additional support.

6.5 How Players Learn During Onboarding

Theme 9 suggests that lived game experience is the most preferred learning method for new players. Existing literature supports this concept, as one of the greatest onboarding challenges is engaging players during their initial gameplay experience [12, 32]. While designers might feel the urge to provide players with as detailed instructions as possible to fully prepare them before they start the "real" game, players show a clear preference for *interaction* during their learning experience—a type of lived experience. This makes intuitive sense: if players perform an action in-game, they are involved with the input and the output simultaneously. There is a clear difference between telling a player to press a button to attack, versus the player pressing a button and attacking. Directly involving the player in the learning experience makes it easier for them to understand how their actions influence what happens in the game.

6.5.1 Let Them Play, Let Them Lose, Let Them Cook

While it is still important to provide players with some amount of structure during the learning experience [18], **Theme 9** also identifies how players use all types of lived experiences to improve their understanding of games. Many games share common knowledge that is not always necessary to teach explicitly—e.g., the left control stick moves a character while the right moves the camera; health-restoring items are consumed when they are used; critical hits happen randomly; etc. Not every game mechanic deserves the same level of detail, and designers must ensure that they prioritize unfamiliar mechanics during the onboarding process. For example, Joker does not need to spend time explaining playing cards, or that each square on a chessboard only holds one unit, since these are common enough mechanics to assume familiarity. Focusing on a game's unique mechanics can also contribute to a memorable and engaging onboarding experience, which in turn will help avoid player perception of onboarding being boring or unnecessary.

As well, even though it may be tempting to want players to experience as much success in-game as possible, players are perfectly willing to use losing as a learning method. **Theme 9** shows a perspective of gradual learning, even in losses, because each lived experience contributes to a player's overall knowledge regardless if they win or lose. Again, there is a difference between telling a player "you lose if the enemy checkmates your king", and having a player experience a loss by checkmate. As long as players understand the reason that they lost—and feel that it was *fair*, which is sometimes a larger mental hurdle to conquer than that of understanding—then the loss itself becomes a stepping stone on their journey to in-game success. In the same way that experiencing a thrilling victory motivates players to play another round, making a game-ending mistake is the strongest motivator for players to never make that mistake again.

6.6 Summary

In this section I used my research questions to contextualize and discuss the results presented in Chapter 5 within the overarching topics of AI-supported suggestion systems, onboarding expectations, perceptions of suggestions, and learning preferences.

Chapter 7

Limitations and Future Work

This study was not without its limitations. While having a participant sample size of 20 was sufficient for qualitative analysis, this smaller size posed a few problems during quantitative analysis. Since the data did not demonstrate normality (which was expected, as it is Likert data), I was therefore unable to use paired sample t-tests in analysis—if I had used a sample size of 30, then normality would not have been required. Furthermore, my participant demographics were very homogenous; while this was an intentional choice considering the scope of the study, I also acknowledge that there is no guarantee that the presented results are applicable to a general population.

As well, statistical analysis of the quantitative data showed some unexpected results. Some of these can be explained by participant error: for instance, participants not remembering their first responses during the second round contributed to the difference between the physical demand of the conditions, even though the demand did not change. The miniPXI *Audiovisual Appeal* measure was also the only quantitative result with statistical significance, even though the audio and visuals of the game also did not change. I believe this is due to how Wilcoxon signed-rank tests handle ties, as well as again due to participants not remembering their answers between conditions. I also noticed during the study that many participants completed the RTLX Performance scale backwards—all other scales have "low" on the left and "high" on the right, but this scale has "high" on the left and "low" on the right. This is unfortunately how the NASA-TLX survey was designed, and so I did not make any changes to the presentation in Qualtrics to preserve the scale's integrity.

This study also uses a within-subjects design, and while I randomized the order for conditions across participants, I acknowledge that some amount of bias is inevitably present.

Since this study focuses on player experience, it was important for my analysis that participants could compare their experience with the two versions during the interview; future evaluations to target the impact of an AI-supported suggestion system with a more quantitative focus could consider between-subjects study designs instead.

Additionally, while I successfully designed Joker to fulfil this study's requirements, my own ability to design an AI system limited the system implemented in the final game. A true AI-supported suggestion system would ideally train a machine learning algorithm to generate suggestions; Joker's algorithm is static, and cannot adapt to the player or the game. The decision tree-like implementation is still an AI system—albeit a primitive one—and was still presented to participants in the same way that a machine learning algorithm would have been, but future studies on the concept of AI-supported suggestion systems and onboarding should implement modern AI systems to more accurately assess their feasibility, performance, and use cases.

Chapter 8

Conclusion

This study demonstrated the implementation of an AI-supported suggestion system as a just-in-time onboarding method to examine the impact of AI-supported suggestion systems on player experience. Analysis of the results found that while the presence of suggestions successfully reduced players' cognitive load, they also reduced players' incentive to learn through active engagement. Additionally, players' main expectation for AI-supported suggestions is that the system provides a reason for its recommendation; without an explanation, players cannot effectively learn from the suggestion. Since players primarily learn through lived experience with games—regardless if they win or lose—AI-supported onboarding systems must therefore carefully maintain a player's agency during the learning process, while also creating enough structure to prevent players from becoming confused, lost, or overwhelmed. No single approach to onboarding can satisfy the diverse range of player preferences, so to prevent the frustration that occurs when onboarding is too restrictive or too long, it is important that onboarding provides players the freedom to choose how and when they learn. AI-supported onboarding has the potential to support this freedom of choice by creating dynamic onboarding experiences tailored to each player's learning needs.

References

[1] Vero Vanden Abeele, Katta Spiel, Lennart Nacke, Daniel Johnson, and Kathrin Gerling. Development and validation of the player experience inventory: A scale to measure player experiences at the level of functional and psychosocial consequences. *International Journal of Human-Computer Studies*, 135:102370, March 2020.

[2] Erik Andersen, Eleanor O'Rourke, Yun-En Liu, Rich Snider, Jeff Lowdermilk, David Truong, Seth Cooper, and Zoran Popovic. The impact of tutorials on games of varying complexity. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, page 59–68, New York, NY, USA, May 2012. Association for Computing Machinery.

[3] Hikari Ando, Rosanna Cousins, and Carolyn Young. Achieving saturation in thematic analysis: Development and refinement of a codebook. *Comprehensive Psychology*, 3:03.CP.3.4, January 2014.

[4] Talya N. Bauer and Berrin Erdogan. Organizational socialization: The effective onboarding of new employees. In *APA handbook of industrial and organizational psychology, Vol 3: Maintaining, expanding, and contracting the organization.*, page 51–64. American Psychological Association, Washington, 2011.

[5] Luciana Benotti and Nicolás Bertoa. Content determination through planning for flexible game tutorials. In Ildar Batyrshin and Grigori Sidorov, editors, *Advances in Artificial Intelligence*, volume 7094 of *Lecture Notes in Computer Science*, page 345–356. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[6] Virginia Braun and Victoria Clarke. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2):77–101, January 2006.

[7] Virginia Braun and Victoria Clarke. One size fits all? what counts as quality practice in (reflexive) thematic analysis? *Qualitative Research in Psychology*, 18(3):328–352, July 2021.

[8] Virginia Braun and Victoria Clarke. To saturate or not to saturate? questioning data saturation as a useful concept for thematic analysis and sample-size rationales. *Qualitative Research in Sport, Exercise and Health*, 13(2):201–216, March 2021.

[9] Murray Campbell, A. Joseph Hoane, and Feng-hsiung Hsu. Deep blue. *Artificial Intelligence*, 134(1):57–83, January 2002.

[10] Shuangyuan Cao and Fang Liu. Learning to play: Understanding in-game tutorials with a pilot study on implicit tutorials. *Heliyon*, 8(11):e11482, November 2022.

[11] Yang Chen, Jian Ou, and David M. Whittinghill. Cognitive load in real-time strategy gaming: Human opponent versus ai opponent. *The Computer Games Journal*, 4(1):19–30, June 2015.

[12] Gifford K. Cheung, Thomas Zimmermann, and Nachiappan Nagappan. The first hour experience: How the initial play can engage (or lose) new players. In *Proceedings of the First ACM SIGCHI Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '14, page 57–66, New York, NY, USA, 2014. Association for Computing Machinery.

[13] Victoria Clarke and Virginia Braun. Thematic analysis. *The Journal of Positive Psychology*, 12(3):297–298, May 2017.

[14] Sara Czerwonka and Victoria McArthur. The impact of tutorial design on the novice gaming experience. In Constantine Stephanidis, Margherita Antona, and Stavroula Ntoa, editors, *HCI International 2022 Posters*, Communications in Computer and Information Science, page 33–41, Cham, 2022. Springer International Publishing.

[15] Simon Demediuk, Alexandra Murrin, David Bulger, Michael Hitchens, Anders Drachen, William L. Raffe, and Marco Tamassia. Player retention in League of Legends: A study using survival analysis. In *Proceedings of the Australasian Computer Science Week Multiconference*, page 1–9, Brisband Queensland Australia, January 2018. ACM.

[16] Vanessa Paz Dennen. Cognitive apprenticeship in educational practice: Research on scaffolding, modeling, mentoring, and coaching as instructional strategies. In *Handbook of research on educational communications and technology*, pages 804–819. Routledge, 2013.

[17] Heather Desurvire and Charlotte Wiberg. Master of the game: Assessing approachability in future game design. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems*, page 3177–3182, Florence Italy, April 2008. ACM.

[18] Heather Desurvire and Charlotte Wiberg. User experience design for inexperienced gamers: GAP—game approachability principles. In Regina Bernhaupt, editor, *Game User Experience Evaluation*, Human–Computer Interaction Series, page 169–186. Springer International Publishing, Cham, 2015.

[19] Tjitske J. E. Faber, Mary E. W. Dankbaar, Rob Kickert, Walter W. van den Broek, and Jeroen J. G. van Merriënboer. Identifying indicators to guide adaptive scaffolding in games. *Learning and Instruction*, 83:101666, February 2023.

[20] Fiona Fui-Hoon Nah, Ruilin Zheng, Jingyuan Cai, Keng Siau, and Langtao Chen. Generative AI and ChatGPT: Applications, challenges, and AI-human collaboration. *Journal of Information Technology Case and Application Research*, 25(3):277–304, July 2023.

[21] James Paul Gee. Learning by design: Good video games as learning machines. *E-Learning and Digital Media*, 2(1):5–16, March 2005.

[22] Michael Cerny Green, Ahmed Khalifa, Gabriella A. B. Barros, Tiago Machado, Andy Nealen, and Julian Togelius. AtDELFI: automatically designing legible, full instructions for games. In *Proceedings of the 13th International Conference on the Foundations of Digital Games*, page 1–10, Malmö Sweden, August 2018. ACM.

[23] Michael Cerny Green, Ahmed Khalifa, Gabriella A. B. Barros, Andy Nealen, and Julian Togelius. Generating levels that teach mechanics. In *Proceedings of the 13th International Conference on the Foundations of Digital Games*, page 1–8, Malmö Sweden, August 2018. ACM.

[24] Aqeel Haider, Kathrin Gerling, and Vero Vanden Abeele. The player experience inventory bench: Providing games user researchers actionable insight into player experiences. In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play*, page 248–252, Virtual Event Canada, November 2020. ACM.

[25] Aqeel Haider, Casper Harteveld, Daniel Johnson, Max V. Birk, Regan L. Mandryk, Magy Seif El-Nasr, Lennart E. Nacke, Kathrin Gerling, and Vero Vanden Abeele. miniPXI: Development and validation of an eleven-item measure of the player experience inventory. *Proceedings of the ACM on Human-Computer Interaction*, 6(CHI PLAY):1–26, October 2022.

[26] Sandra G. Hart. Nasa-task load index (NASA-TLX); 20 years later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50(9):904–908, 2006.

[27] Monique Hennink and Bonnie N. Kaiser. Sample sizes for saturation in qualitative research: A systematic review of empirical tests. *Social Science & Medicine*, 292:114523, January 2022.

[28] Rob Houser and Scott DeLoach. Learning from games: Seven principles of effective design. *Technical Communication*, 45, August 1998.

[29] Richard E. Mayer and Roxana Moreno. Nine ways to reduce cognitive load in multimedia learning. *Educational Psychologist*, 38(1):43–52, January 2003.

[30] Jonas Molin and Charlotte Wiberg. Game approachability: Remodelling testing of computer games with the GAID model. *International Research Journal of Computer Science*, 3, 2016.

[31] Fred Paas, Alexander Renkl, and John Sweller. Cognitive load theory and instructional design: Recent developments. *Educational Psychologist*, 38(1):1–4, January 2003.

[32] Randy J Pagulayan, Kevin Keeker, Dennis Wixon, Ramon L Romero, and Thomas Fuller. User-centered design in games. In *The Human-Computer Interaction Handbook*, page 883–906. CRC Press, 2002.

[33] Kunwoo Park, Meeyoung Cha, Haewoon Kwak, and Kuan-Ta Chen. Achievement and friends: Key factors of player retention vary across player levels in online multiplayer games. In *Proceedings of the 26th International Conference on World Wide Web Companion - WWW '17 Companion*, page 445–453, Perth, Australia, 2017. ACM Press.

[34] Mario Passalacqua, Raphaël Morin, Sylvain Sénécal, Lennart E. Nacke, and Pierre-Majorique Léger. Demystifying the first-time experience of mobile games: The presence of a tutorial has a positive impact on non-expert players' flow and continuous-use intentions. *Multimodal Technologies and Interaction*, 4(3):41, July 2020.

[35] Falko Weigert Petersen, Line Ebdrup Thomsen, Pejman Mirza-Babaei, and Anders Drachen. Evaluating the onboarding phase of free-to-play mobile games: A mixed-method approach. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, page 377–388, Amsterdam The Netherlands, October 2017. ACM.

[36] Lev Poretski and Anthony Tang. Press A to jump: Design strategies for video game learnability. In *CHI Conference on Human Factors in Computing Systems*, page 1–26, New Orleans LA USA, April 2022. ACM.

[37] Mark Owen Riedl and Alexander Zook. AI for game production. In *2013 IEEE Conference on Computational Inteligence in Games (CIG)*, page 1–8, Niagara Falls, ON, Canada, August 2013. IEEE.

[38] Amy Shannon, P O Box, Acey Boyce, Chitra Gadwal, and Dr Tiffany Barnes. Effective practices in game tutorial systems. *Foundations of Digital Games*, 2013.

[39] Chin Hiong Tan, Kay Chen Tan, and Arthur Tay. Dynamic game difficulty scaling using adaptive behavior-based ai. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(4):289–301, December 2011.

[40] Gareth Terry, Nikki Hayfield, Victoria Clarke, and Virginia Braun. Thematic analysis. In *The SAGE Handbook of Qualitative Research in Psychology*, volume 0, page 17–36. SAGE Publications Ltd, 1 Oliver's Yard, 55 City Road London EC1Y 1SP, 2017.

[41] Line E. Thomsen, Falko Weigert Petersen, Anders Drachen, and Pejman Mirza-Babaei. Identifying onboarding heuristics for free-to-play mobile games: A mixed methods approach. In *Entertainment Computing - ICEC 2016*, volume 9926 of *Lecture Notes in Computer Science*, page 241–246. Springer International Publishing, Cham, 2016.

[42] Julian Togelius, Noor Shaker, and Mark J. Nelson. Introduction. In Noor Shaker, Julian Togelius, and Mark J. Nelson, editors, *Procedural Content Generation in Games*, Computational Synthesis and Creative Systems, page 1–15. Springer International Publishing, Cham, 2016.

[43] Jeroen J. G. Van Merriënboer, Paul A. Kirschner, and Liesbeth Kester. Taking the load off a learner's mind: Instructional design for complex learning. *Educational Psychologist*, 38(1):5–13, January 2003.

[44] Matthew M. White. The tutorial. In *Learn to Play*, chapter 2, page 17–36. A K Peters/CRC Press, 2014.

[45] Matthew Martin White. L2P NOOB: Examining tutorials in digital games. *Loading...*, 6(1010), December 2012.

[46] Geogios N. Yannakakis. Game AI revisited. In *Proceedings of the 9th conference on Computing Frontiers*, page 285–292, Cagliari Italy, May 2012. ACM.

[47] Georgios N. Yannakakis and Julian Togelius. Generating content. In Georgios N. Yannakakis and Julian Togelius, editors, *Artificial Intelligence and Games*, page 151–202. Springer International Publishing, Cham, 2018.

[48] Georgios N. Yannakakis and Julian Togelius. Introduction. In *Artificial Intelligence and Games*, page 3–28. Springer International Publishing, Cham, 2018.

[49] Georgios N. Yannakakis and Julian Togelius. Modeling players. In Georgios N. Yannakakis and Julian Togelius, editors, *Artificial Intelligence and Games*, page 203–255. Springer International Publishing, Cham, 2018.

[50] Georgios N. Yannakakis and Julian Togelius. Playing games. In Georgios N. Yannakakis and Julian Togelius, editors, *Artificial Intelligence and Games*, page 91–150. Springer International Publishing, Cham, 2018.

[51] Alexander Zook and Mark O. Riedl. Temporal game challenge tailoring. *IEEE Transactions on Computational Intelligence and AI in Games*, 7(4):336–346, December 2015.

APPENDICES

Appendix A

Screening Survey

UNIVERSITY OF WATERLOO

Screening Questions

Are you currently an adult (18 or older)?

- Yes
- No

Are you comfortable with being audio and screen recorded?

- Yes
- No

Are you comfortable using a computer (keyboard and mouse)?

- Yes
- No

Are you comfortable sitting down for the duration of the study (up to 90 minutes)?

- Yes
- No

Powered by Qualtrics

Appendix B

Study Survey

UNIVERSITY OF WATERLOO

Consent

Project Title: Game Design: AI-Supported Onboarding and User Experience
Principal Investigators:
Lennart Nacke, Associate Prof., Stratford School of Interaction Design & Business, len@uwaterloo.ca
Student Investigators:
Lydia Choong, Student Researcher, Computer Science, ljmchoon@uwaterloo.ca

Video game tutorials often struggle to find a balance between mandatory introductions that delay actual gameplay, and the more overwhelming “hands-off” approaches that force players to learn through trial and error. Artificial intelligence (AI) onboarding systems aim to find a balance between these two extremes—get the player into real gameplay as soon as possible, but still provide them with relevant support over the course of gameplay to prevent them from becoming cognitively overwhelmed. The purpose of this study is therefore to understand the impact of AI-supported onboarding systems on cognitive load, player performance, and user experience during the onboarding process.

You must be 18 years of age or older to be eligible for this study, which is taking place in the Games Institute in EC1 at the University of Waterloo campus. The survey provided at the beginning of the study will ask you to provide your demographic information (age, gender/sex, ethnicity, occupation, education); you may choose to not answer any of these questions. You also must be comfortable using a computer (mouse and keyboard). After completing the demographic survey, you will play a turn-based strategy game on a computer and fill out a survey on your gameplay experience. You will play the game two times. The gameplay will be followed by a semi-structured interview about your experience.

With your consent, we will record the audio of the gameplay and interview sections of the study for later analysis. We will also screen record the gameplay sections of the study. We will inform you when we begin and end the recordings. You may request to stop recording at any time. After all participants have completed the study, all recordings will be transcribed and then deleted according to our data retention practices. No personal information, including the audio and screen recordings, will ever be shared. You will not be personally identified in any study report (neither by name nor voice). Any data gathered from this study will be stored on a secure platform accessible only to the researchers.

Time Commitment: Your participation in this approximately 90-minute study is entirely voluntary. You may refuse to participate or withdraw from the study at any time up. In appreciation of your time commitment you will receive $20 **CAD** in cash. The amount received is taxable. It is your responsibility to report this amount for income tax purposes.

Risks & Benefits: This study has no known or anticipated risks to participants. There are no direct benefits to participants, but the results will benefit the scientific community —specifically, it will contribute to a greater understanding of the relationship between AI systems and user experience, and in particular the area of player onboarding in video games.

Confidentiality: You will not be personally identified in any study report (neither by name nor voice). Any data gathered from this study will be stored on a secure platform accessible only to the researchers, and all sensitive information will be anonymized. However, when information is transmitted over the internet and/or stored on a remote platform, privacy cannot be guaranteed. There is always a risk your responses may be intercepted by a third party (e.g., government agencies, hackers).

Withdrawal: You can contact us to withdraw your consent to participate and have your data destroyed. We will keep our study records for a minimum of 8 years.

The researchers will answer any questions you may have over the course of the study. Your participation in this study is entirely voluntary and you may refuse to participate or withdraw at any time. The researchers will ask for your consent to participate in this study and for you to acknowledge that you have received a copy of this consent document. By consenting, you are not waiving your legal rights or releasing the investigator(s) or involved institution(s) from their legal and professional responsibilities.

Consent options:
I have read the information presented in the information letter about a study conducted by Lydia Choong, under the supervision of Dr Lennart Nacke, Associate Prof., Stratford School of Interaction Design & Business. I have had the opportunity to ask questions related to the study and have received satisfactory answers to my questions and any additional details. I was informed that participation in the study is voluntary and that I can inform the researchers if I withdraw this consent at any time.

This study has been reviewed and received ethics clearance through a University of Waterloo Research Ethics Board (REB #45535). If you have questions for the Board, contact the Office of Research Ethics toll-free at 1-833-643-2379 (Canada and USA), 1-519-888-4440, or reb@uwaterloo.ca. If you have any questions about the study, please contact the researchers (see contact information at the top of this document).

Consent

- ☐ I am currently 18 years of age or older and I agree to participate in this study (required).
- ☐ I agree to being audio recorded during the study to ensure accurate transcription and analysis (required).
- ☐ I agree to my gameplay being screen recorded during the study (required).
- ☐ I agree to the use of anonymous quotations in any thesis or publication that comes from this research (optional).
- ☐ I do not agree to take part in this study (please notify the researcher; you may then leave the study).

Participant initials here

× SIGN HERE

clear

Participant ID

Please ask the researcher for your participant ID (e.g. "P1")

Demographics

How old are you?

What gender do you most identify with?

- Male
- Female
- Non-binary / third gender
- Prefer to self-identify
- Prefer not to say

What is your ethnicity? Select all that apply.

- Arab
- Black (including African, African-Canadian, African-American, Caribbean
- Chinese (including Mainland China, Hong Kong, Macau, and Taiwan)
- Filipino/a
- Indo-Caribbean, Indo-African, Indo-Fijian, or West-Indian
- Indigenous (First Nations, Métis, Inuit)
- Japanese
- Korean
- Latin, Central, or South American (e.g., Brazilian, Chilean, Columbian, Mexican)

- ☐ North African (Egyptian, Libyan)
- ☐ Pacific Islanders or Polynesian/Melanesian/Micronesian (e.g., Cook Island Māori, Hawaiian Mā'oli, Fijians, Marquesan, Marshallese, Niuean, Samoans, Tahitian Mā'ohi, Tongan, New Zealand Māori)
- ☐ South Asian (e.g., Bangladeshi, Pakistani, Indian, Sri Lankan, Punjabi)
- ☐ South East Asian (e.g., Cambodian, Malaysian, Thai, Vietnamese)
- ☐ West Asian (e.g., Afghani, Armenian, Iranian, Iraqi, Israeli, Jordanian, Lebanese, Palestinian, Syrian, Yemeni)
- ☐ White (including European, White-Canadian/American/Australian/South African)
- ☐ Prefer to self-identify ____________
- ☐ Prefer not to disclose

What is your highest education level?

- ○ No schooling completed
- ○ Some high school, no diploma
- ○ High school graduate, diploma or the equivalent
- ○ Some college credit, no degree
- ○ Associate degree
- ○ Bachelor's degree
- ○ Postgraduate degree
- ○ Prefer not to disclose

What is your employment status?

- ○ Employed
- ○ Self-employed
- ○ Student
- ○ Retired
- ○ Unemployed
- ○ Other ____________
- ○ Prefer not to disclose

Break

Please **DO NOT** close this tab.
Please **notify the researcher** you are finished the previous section.

Condition 1

Please ask the researcher to enter your condition ID.

Please indicate how much you agree or disagree with the following statements:

	Strongly disagree	Disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Agree	Strongly agree
I liked the look and feel of the game.	○	○	○	○	○	○	○
The game was not too easy and not too hard to play.	○	○	○	○	○	○	○
It was easy to know how to perform actions in the game.	○	○	○	○	○	○	○
The goals of the game were clear to me.	○	○	○	○	○	○	○
The game gave clear feedback on my progress towards the goals.	○	○	○	○	○	○	○
I felt free to play the game in my own way.	○	○	○	○	○	○	○
I wanted to explore how the game evolved.	○	○	○	○	○	○	○

	Strongly disagree	Disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Agree	Strongly agree
I was fully focused on the game.	○	○	○	○	○	○	○
I felt I was good at playing this game.	○	○	○	○	○	○	○
Playing the game was meaningful to me.	○	○	○	○	○	○	○
I had a good time playing this game.	○	○	○	○	○	○	○

Mental Demand

Very Low Very High

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

How mentally demanding was the task? 11

Physical Demand

Very Low Very High

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

How physically demanding was the task? 11

Temporal Demand

	Very Low																				Very High
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
How hurried or rushed was the pace of the task?											11										

Performance

	Perfect																				Failure
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
How successful were you in accomplishing what you were asked to do?											11										

Effort

	Very Low																				Very High
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
How hard did you have to work to accomplish your level of performance?											11										

Frustration

	Very Low																				Very High
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
How insecure, discouraged, irritated, stressed, and annoyed were you?											11										

Break

Please **DO NOT** close this tab.
Please **notify the researcher** you are finished the previous section.

Condition 2

Please ask the researcher to enter your condition ID.

Please indicate how much you agree or disagree with the following statements:

	Strongly disagree	Disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Agree	Strongly agree
I liked the look and feel of the game.	○	○	○	○	○	○	○
The game was not too easy and not too hard to play.	○	○	○	○	○	○	○
It was easy to know how to perform actions in the game.	○	○	○	○	○	○	○
The goals of the game were clear to me.	○	○	○	○	○	○	○
The game gave clear feedback on my progress towards the goals.	○	○	○	○	○	○	○
I felt free to play the game in my own way.	○	○	○	○	○	○	○
I wanted to explore how the game evolved.	○	○	○	○	○	○	○
I was fully focused on the game.	○	○	○	○	○	○	○
I felt I was good at playing this game.	○	○	○	○	○	○	○

	Strongly disagree	Disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Agree	Strongly agree
Playing the game was meaningful to me.	○	○	○	○	○	○	○
I had a good time playing this game.	○	○	○	○	○	○	○

Mental Demand

Very Low Very High

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

How mentally demanding was the task? 11

Physical Demand

Very Low Very High

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

How physically demanding was the task? 11

Temporal Demand

Very Low Very High

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

How hurried or rushed was the pace of the task? 11

Performance

	Perfect																				Failure
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
How successful were you in accomplishing what you were asked to do?											11										

Effort

	Very Low																				Very High
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
How hard did you have to work to accomplish your level of performance?											11										

Frustration

	Very Low																				Very High
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
How insecure, discouraged, irritated, stressed, and annoyed were you?											11										

Break

Please **DO NOT** close this tab.

Please **notify the researcher** you are finished the previous section.

Thank You

Project Title: Game Design: AI-Supported Onboarding and User Experience
Principal Investigators:
Lennart Nacke, Associate Prof., Stratford School of Interaction Design & Business, len@uwaterloo.ca
Student Investigators: Lydia Choong, Student Researcher, Computer Science, ljmchoon@uwaterloo.ca

We appreciate your participation in this study and thank you for taking the time to help us with our research! A summary of the study is provided below. If you are interested in receiving more information regarding the results of this study or would like a summary of the results, please contact the research team through their provided emails.

Study Overview
The purpose of this study is to understand the impact of AI-supported tutorials on cognitive load, player performance, and user experience during the onboarding process. Video game tutorials often struggle to find a balance between mandatory introductions that delay actual gameplay, and the more overwhelming "hands-off" approaches that force players to learn through trial and error. Artificial intelligence (AI) onboarding systems aim to find a balance between these two extremes—get the player into real gameplay as soon as possible, but still provide them with relevant support over the course of gameplay to prevent them from becoming cognitively overwhelmed.

If you have any questions, feedback, or concern about the study or related research, please contact the research team through the emails provided above.

Confidentiality and Data Security
Your identity will be confidential. Your name will not be included in any thesis or report resulting from this study. Electronic data, audio recordings, and screen recordings collected during this study will be retained on a password-protected and encrypted server at the University of Waterloo for a minimum of 8 years, to which only researchers associated with this study have access. All identifying information will be removed from the records prior to storage.

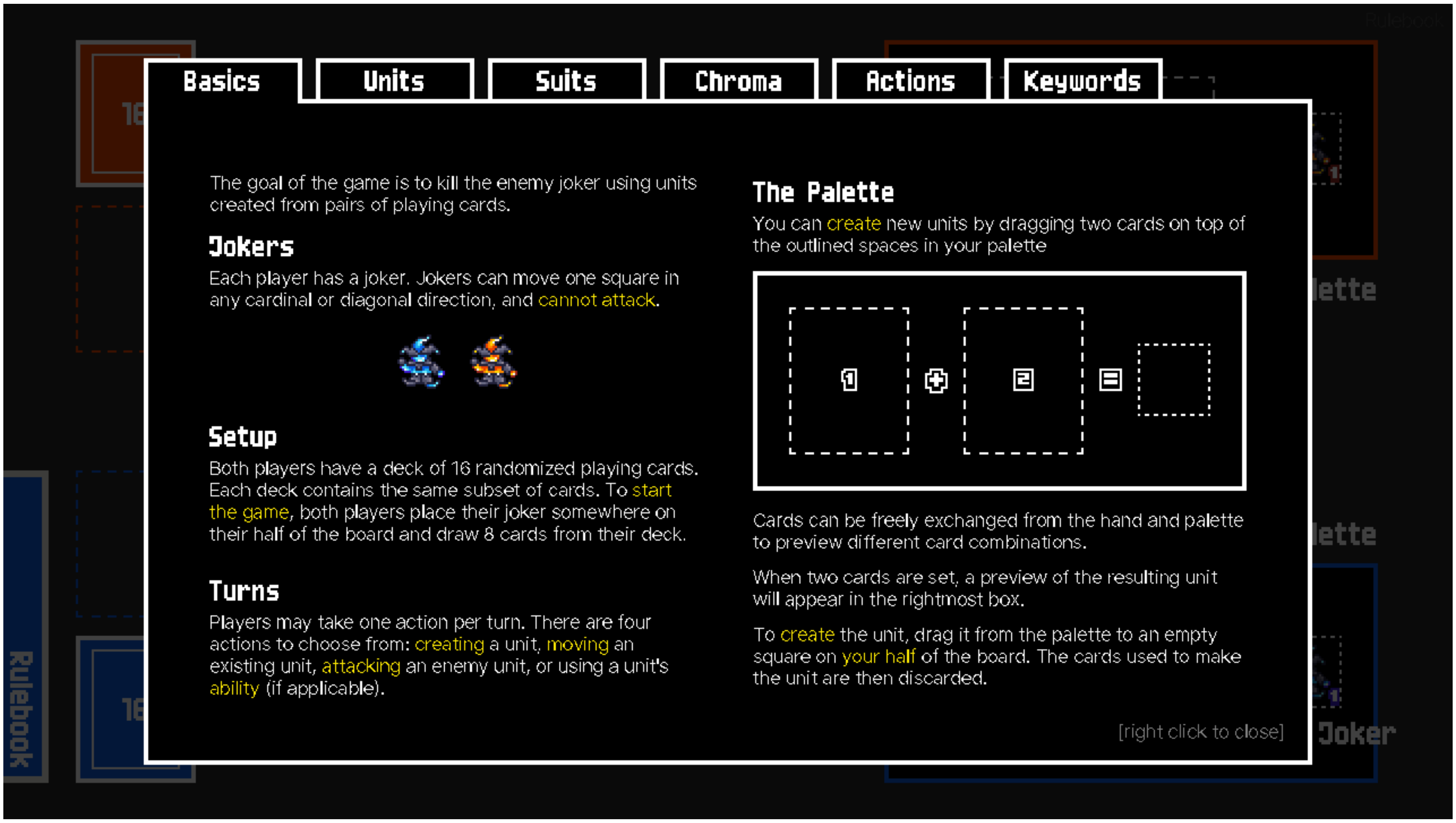
This study has been reviewed and received ethics clearance through a University of Waterloo Research Ethics Board (REB #45535). If you have questions for the Board, contact the Office of Research Ethics toll-free at 1-833-643-2379 (Canada and USA), 1-519-888-4440, or reb@uwaterloo.ca. If you have any questions about the study, please contact the researchers (see contact information at the top of this document).

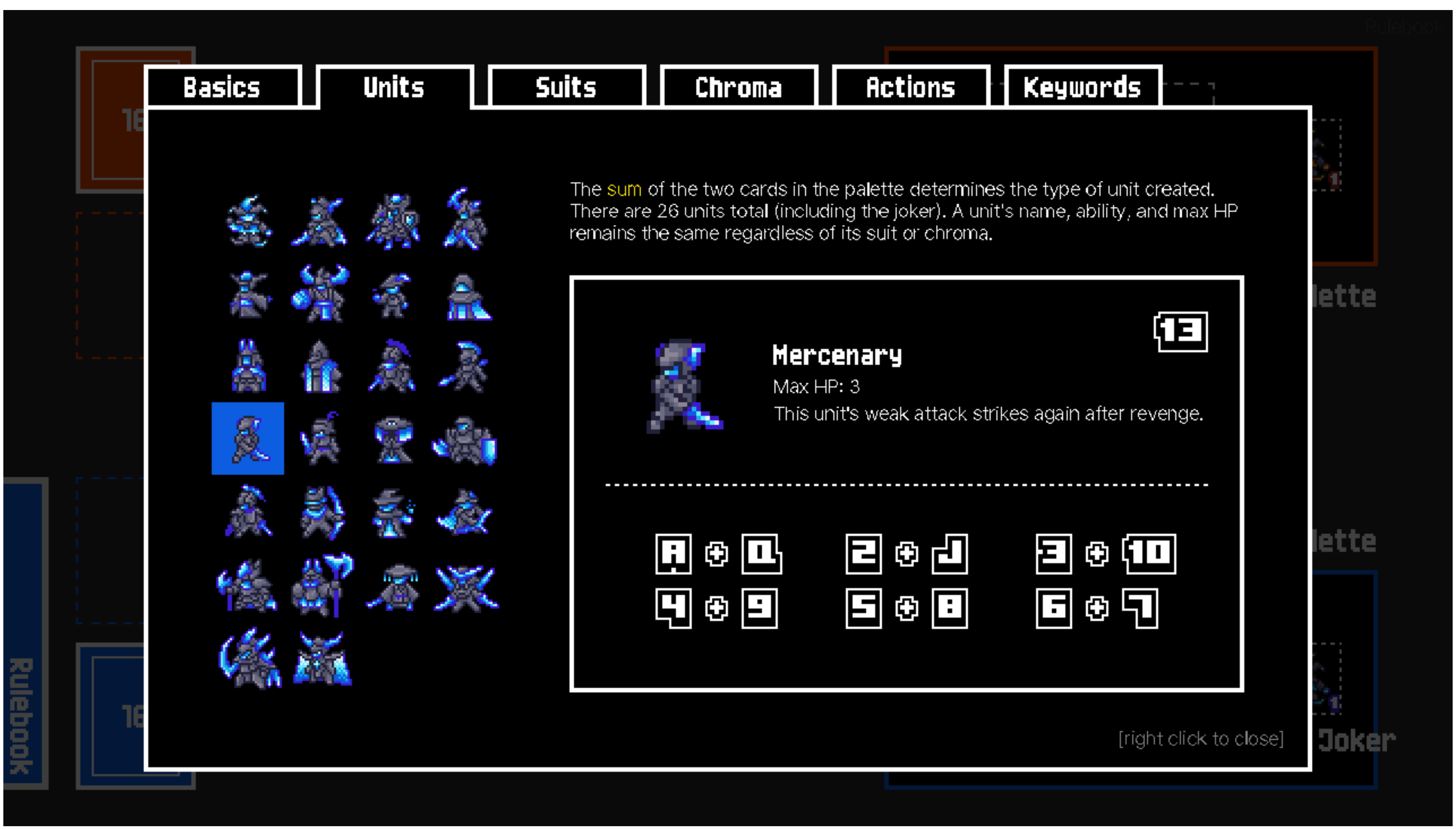
Please enter your email if you are interested in receiving more information regarding the results of this study (optional).

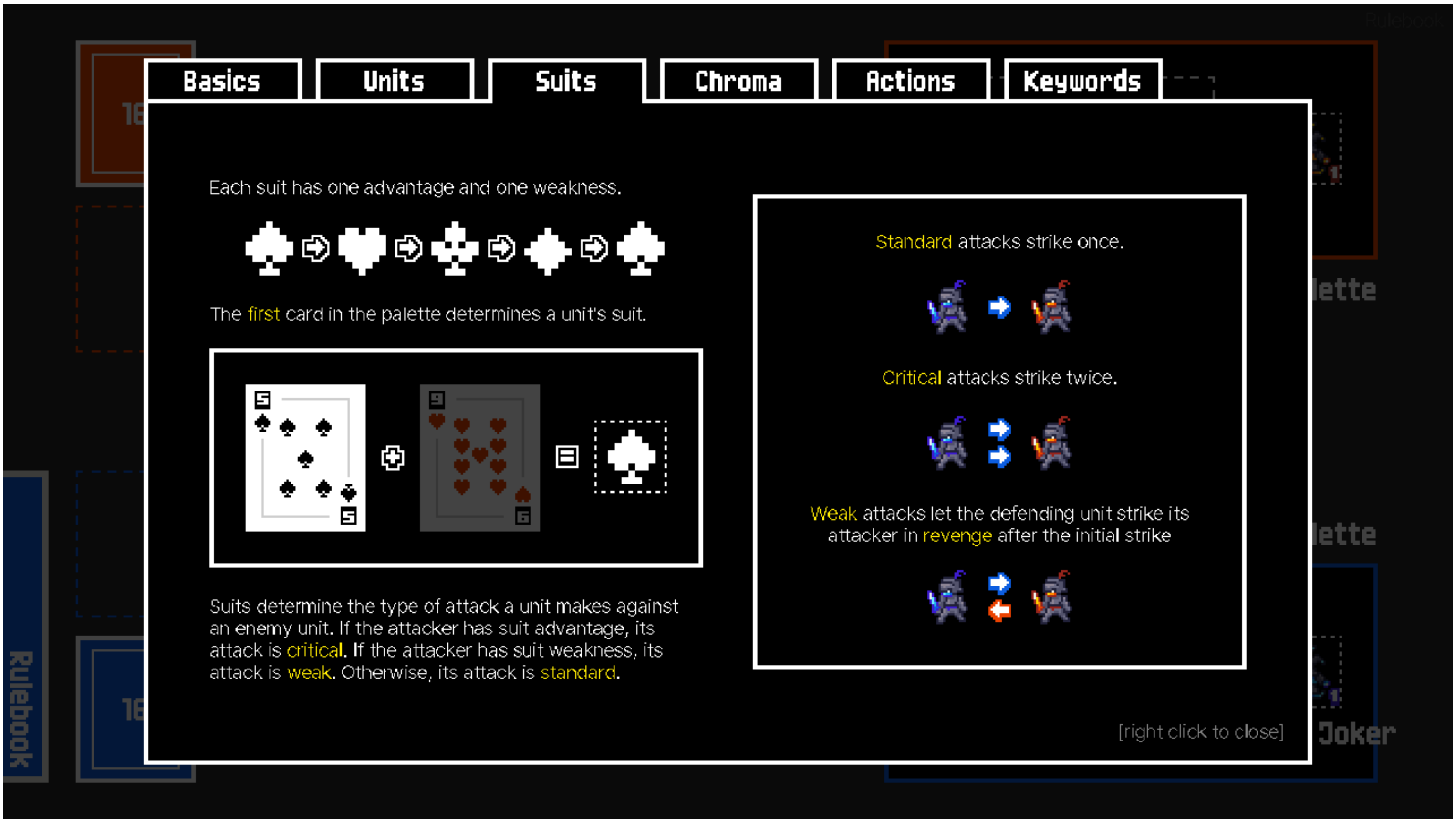
Powered by Qualtrics

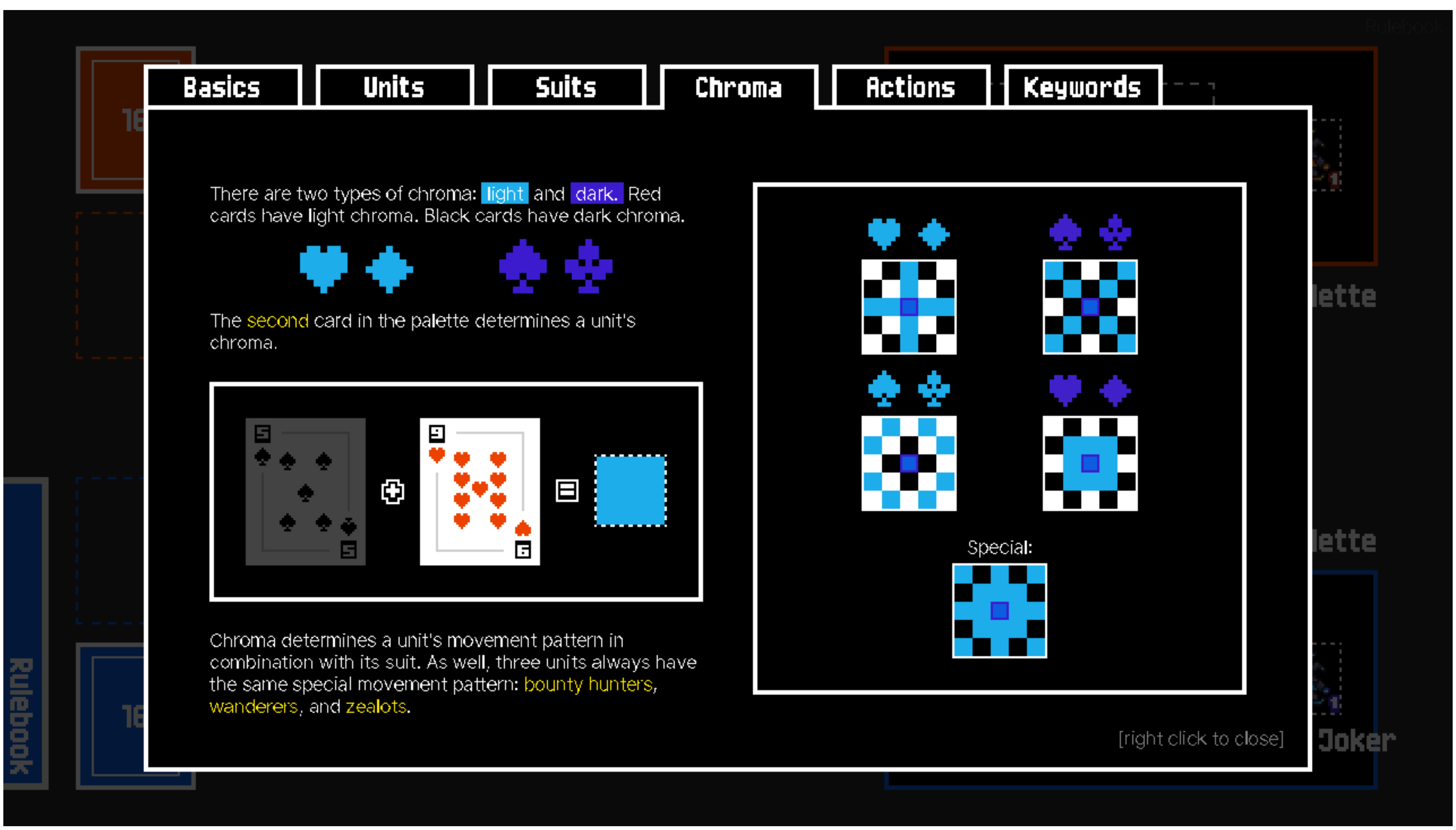
Appendix C

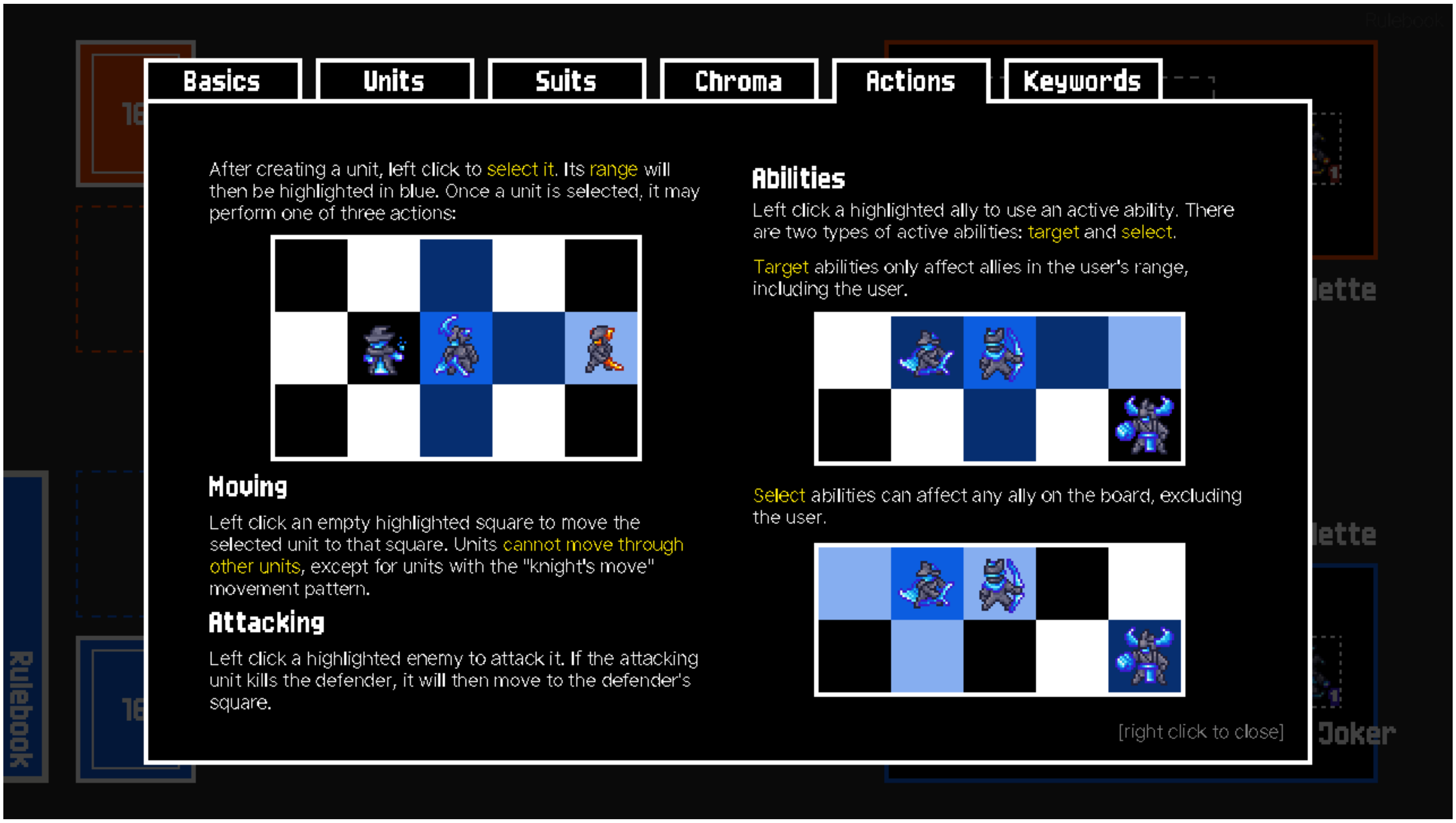
Rulebook

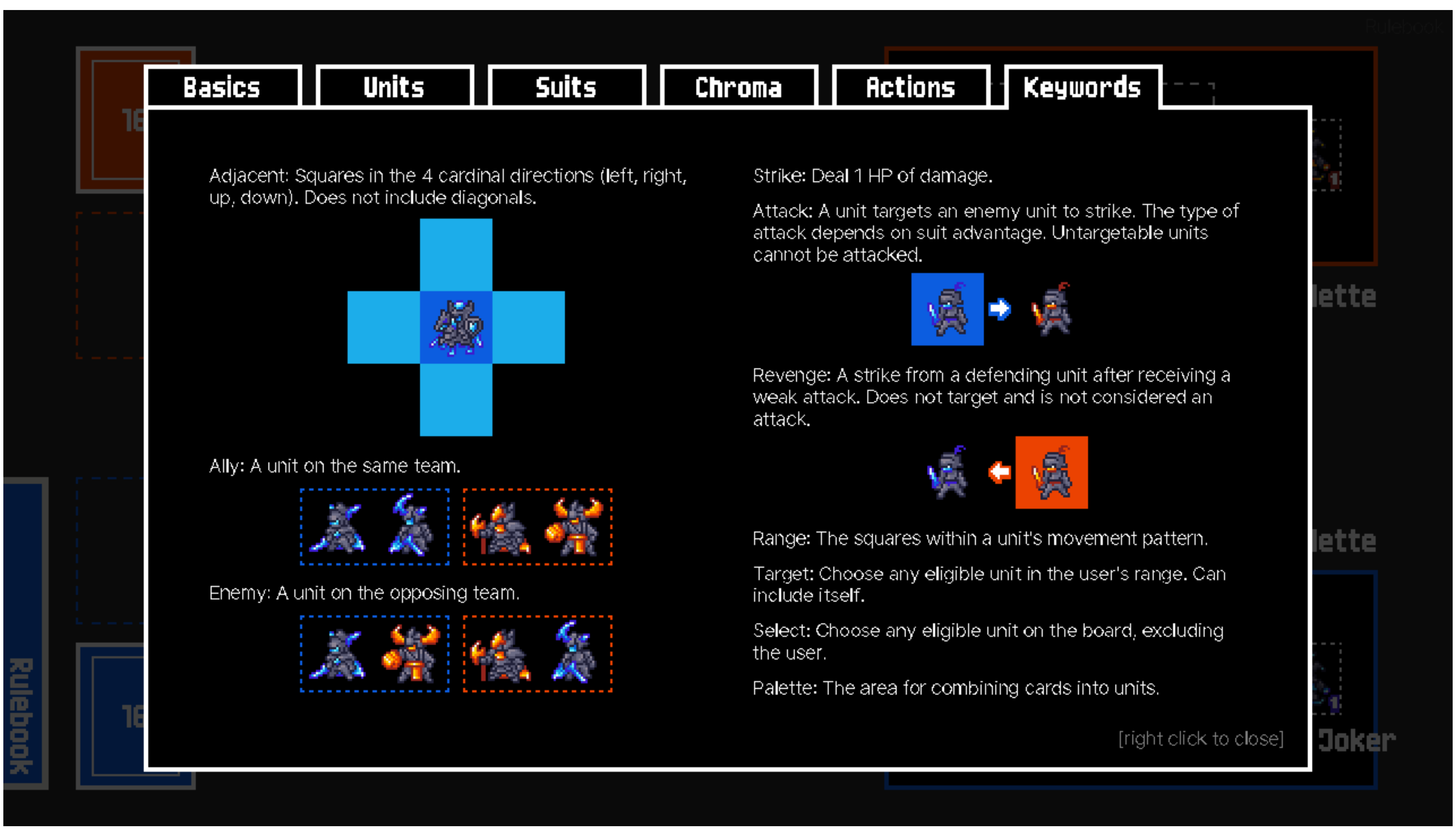

Rulebook
Basics
Units
Suits
Chroma
Actions
Keywords
The goal of the game is to kill the enemy joker using units created from pairs of playing cards.
Jokers
Each player has a joker. Jokers can move one square in any cardinal or diagonal direction, and cannot attack.
Setup
Both players have a deck of 16 randomized playing cards. Each deck contains the same subset of cards. To start the game, both players place their joker somewhere on their half of the board and draw 8 cards from their deck.
Turns
Players may take one action per turn. There are four actions to choose from: creating a unit, moving an existing unit, attacking an enemy unit, or using a unit's ability (if applicable).
The Palette
You can create new units by dragging two cards on top of the outlined spaces in your palette
1
+
2
=
Cards can be freely exchanged from the hand and palette to preview different card combinations.
When two cards are set, a preview of the resulting unit will appear in the rightmost box.
To create the unit, drag it from the palette to an empty square on your half of the board. The cards used to make the unit are then discarded.
[right click to close]
Joker


Rulebook
Basics
Units
Suits
Chroma
Actions
Keywords
The sum of the two cards in the palette determines the type of unit created. There are 26 units total (including the joker). A unit's name, ability, and max HP remains the same regardless of its suit or chroma.
13
Mercenary
Max HP: 3
This unit's weak attack strikes again after revenge.
A + Q
2 + J
3 + 10
4 + 9
5 + 8
6 + 7
[right click to close]
Joker


Basics
Units
Suits
Chroma
Actions
Keywords
Each suit has one advantage and one weakness.
The first card in the palette determines a unit's suit.
Suits determine the type of attack a unit makes against an enemy unit. If the attacker has suit advantage, its attack is critical. If the attacker has suit weakness, its attack is weak. Otherwise, its attack is standard.
Standard attacks strike once.
Critical attacks strike twice.
Weak attacks let the defending unit strike its attacker in revenge after the initial strike
[right click to close]
Rulebook
Joker


Basics
Units
Suits
Chroma
Actions
Keywords
There are two types of chroma: light and dark. Red cards have light chroma. Black cards have dark chroma.
The second card in the palette determines a unit's chroma.
Chroma determines a unit's movement pattern in combination with its suit. As well, three units always have the same special movement pattern: bounty hunters, wanderers, and zealots.
Special:
[right click to close]
Rulebook
Joker


Basics
Units
Suits
Chroma
Actions
Keywords
After creating a unit, left click to select it. Its range will then be highlighted in blue. Once a unit is selected, it may perform one of three actions:
Moving
Left click an empty highlighted square to move the selected unit to that square. Units cannot move through other units, except for units with the "knight's move" movement pattern.
Attacking
Left click a highlighted enemy to attack it. If the attacking unit kills the defender, it will then move to the defender's square.
Abilities
Left click a highlighted ally to use an active ability. There are two types of active abilities: target and select.
Target abilities only affect allies in the user's range, including the user.
Select abilities can affect any ally on the board, excluding the user.
[right click to close]
Rulebook
Joker


Basics
Units
Suits
Chroma
Actions
Keywords
Adjacent: Squares in the 4 cardinal directions (left, right, up, down). Does not include diagonals.
Ally: A unit on the same team.
Enemy: A unit on the opposing team.
Strike: Deal 1 HP of damage.
Attack: A unit targets an enemy unit to strike. The type of attack depends on suit advantage. Untargetable units cannot be attacked.
Revenge: A strike from a defending unit after receiving a weak attack. Does not target and is not considered an attack.
Range: The squares within a unit's movement pattern.
Target: Choose any eligible unit in the user's range. Can include itself.
Select: Choose any eligible unit on the board, excluding the user.
Palette: The area for combining cards into units.
[right click to close]
Rulebook
Joker