Vixen: A Games User Research Tool For Collection And Interactive Visualization Of Usertesting Data

by

Brandon Drenikow

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Science

in

Computer Science

The Faculty of Science

University of Ontario Institute of Technology

Supervisor:

Dr. Pejman Mirza-Babaei

August 2017

Copyright © Brandon Drenikow, 2017
Abstract

Visualization techniques can facilitate the understanding and exploration of relationships in usertesting data. For example, data from players’ in-game movement can be combined with interview data or questionnaire results. However, the process of amalgamation is not straightforward, because the underlying data often exists in different formats. Another challenge is making these visualizations simple enough to provide a quick overview for producers, but also detailed enough to be usable and practical for gameplay programmers. Hence, there is a need for an interactive visualization tool that can adjust data representation based on the nature and detail level of data required from different members of a development team. This thesis reports development efforts on a tool that assists data collection and provides a dynamic and interactive representation of usertesting data. The thesis reports two studies to evaluate the effectiveness of the tool with game developers.

Keywords: Games User Research, Visualization, Usertest, Mixed-methods, Analytics
Acknowledgment

I am very thankful for the multitude of mentors and friends that I have worked with to make this thesis possible. First, I would like to express my gratitude to my supervisor for his time, ideas and the opportunities. His support allowed me to focus on relevant topics in industry helping to prepare me for future endeavours. To Dr. Pejman Mirza-Babaei for his endless support and guidance throughout the master’s, without whom I would have not been able to complete this thesis. I would also like to thank Dr. Andrew Hogue and Dr. Bill Kapralos from the UOIT Games and Entertainment Media Research (GAMER) Lab and the Faculty of Business and IT for their support in conducting my research. Additionally I would like to thank Dr. Loutfouz Zaman, Dr. Andrew Hogue, and Dr. Karthik Sankaranarayanan for their comments that were instrumental in improving my thesis. I enjoyed my time at UOIT working in the constructive and entertaining GAMERLab environment surrounded by several talented individuals. Firstly, I would like to thank to Thomas Galati for his friendship, feedback and assistance throughout my master’s. Everyone at the GAMER Lab for their support over the past few years: James Robb, Wallace Balianic, Nathan Gale, Mike Gharbharan, Naeem Moosajee, Rob Shewaga, Alvaro Uribe, and last but not least, Cameron Choden. I would like to thank my participants for both of my studies to help improve Vixen to what it is today. I would also like to thank everyone who assisted with the development of Vixen: James Simpson, David F. Arppe, Thomas Galati, Chris Carmichael, Liam Syrvk, Montgomery Alban, and Derek Chong. I would also like to thank Ludia and the Core-Tech team for their support throughout my final semester. I would finally like to thank my family for their continuous support throughout my masters.
List of Publications

2017:

2016:

2014:
Statement of Collaboration

I hereby declare that this thesis has not been and will not be, submitted in whole or in part to another University for the award of any other degree.
# Contents

1 Introduction ............................................................... 2  
  1.1 Overview ............................................................. 2  
  1.2 Motivation ........................................................... 2  
  1.3 Working Definitions ................................................ 5  
      1.3.1 Human Computer Interaction ................................... 5  
      1.3.2 Games User Research (GUR) .................................... 5  
  1.4 Problem Statement ................................................. 6  
  1.5 Thesis Statement ................................................... 7  
  1.6 Contributions ...................................................... 7  
  1.7 Organization ........................................................ 7  

2 Related Work ........................................................................ 9  
  2.1 Introduction ............................................................. 9  
  2.2 Games from Academic Institutions ................................... 9  
  2.3 User Research Methods ................................................ 11  
      2.3.1 Observation ....................................................... 12  
      2.3.2 Think-aloud ....................................................... 12  
      2.3.3 Heuristic Evaluation .......................................... 13  
      2.3.4 Questionnaires ................................................ 13  
      2.3.5 Interviews and Focus Groups ................................. 14  
      2.3.6 Rapid Iterative Testing and Evaluation .................... 15  
      2.3.7 Mixed Methods ................................................ 15  
      2.3.8 Challenges of Mixed Methods ................................. 17  
  2.4 Data Collection and Visualization Tools ............................ 18  
      2.4.1 Quantitative Data Visualization Tools ..................... 19  
      2.4.2 Qualitative Data Visualization Tools ..................... 22  
  2.5 Summary ................................................................. 23  

3 Tool Development ............................................................... 24  
  3.1 Introduction ............................................................. 24  
  3.2 Scenario ........................................................................ 25  
  3.3 Initial Prototypes ....................................................... 25  
      3.3.1 Version 1.0 ....................................................... 26  
      3.3.2 Version 1.1 ....................................................... 27  
  3.4 Version 2.0 ............................................................... 33
6.3.3 A Combination Of Both Quantitative / Qualitative Data Over-
laying The Game-World, With Real-Time Filtering . . . . . . . . 86
6.4 Limitations and Future Work . . . . . . . . . . . . . . . . . . . . . 86
6.4.1 Observation Data . . . . . . . . . . . . . . . . . . . . . . . . . . 86
6.4.2 Biometric Data . . . . . . . . . . . . . . . . . . . . . . . . . . . 87
6.4.3 Further Study into Contributions of Vixen in GUR . . . . . . 89
6.4.4 Further Study into Commercialization of Vixen . . . . . . . . 89
6.5 Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 90

Bibliography 91
List of Figures

3.1 Vixen’s First Build (July 2014) .............................................. 27
3.2 Vixen’s Improved Line Renderings (August 2014) ....................... 27
3.3 Vixen’s With Oculus Rift Virtual Reality Headset (August 2014) ... 28
3.4 Vixen In The CAVE (August 2014) ......................................... 28
3.5 Vixen’s Improved UI Element (September 2014) ....................... 31
3.6 Vixen Bird’s Eye View of Unity Demo (September 2014) ............. 31
3.7 Vixen’s Webcam Visualization (January 2015) .......................... 32
3.8 Vixen Setup Interface for Developers (April 2015) ..................... 34
3.9 Top Down Space Shooter in Unity (April 2016) ......................... 36
3.10 Space Shooter filtered to 3 unique participants (April 2016) ......... 36
3.11 Third Person Survival Shooter in Unity (August 2016) ............... 41
3.12 Top-Down Twin Stick Shooter in Unity (August 2016) ............... 42
3.13 Side Scrolling Shooter in Unity (August 2016) ....................... 42

4.1 Flow Chart of Study Phases ...................................................... 45
4.2 Transcribed Data Results ...................................................... 51

5.1 P1’s 2D Game Similar To Flappy Bird ..................................... 69
5.2 P1’s Game With Vixen Implemented ........................................ 70
5.3 P2’s Game With Vixen’s Interface .......................................... 70
5.4 P2’s 3D Adventure Game With Vixen Implemented ..................... 71
5.5 P2’s Game With Another Participant ....................................... 71
5.6 P2’s Game With A Different Set Of Playtesting Data .................. 72
5.7 P2’s Game With More Participant Data Visualized ..................... 72
List of Tables

4.1 Participant Demographics for Visualization Evaluation . . . . . . . . 47
4.2 Participant User Experience Demographics for Visualization Evaluation 48

5.1 Participant Demographics for Visualization Evaluation . . . . . . . . 73
5.2 Participant User Experience Demographics for Visualization Evaluation 73
5.3 Participant Answers . . . . . . . . . . . . . . . . . . . . . . . . . . . 74
5.4 Participant Answers Continued . . . . . . . . . . . . . . . . . . . . . 75
## List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GUR</td>
<td>Games User Research</td>
</tr>
<tr>
<td>HCI</td>
<td>Human-Computer Interactions</td>
</tr>
<tr>
<td>UT</td>
<td>User Testing</td>
</tr>
<tr>
<td>M</td>
<td>Million</td>
</tr>
<tr>
<td>QA</td>
<td>Quality Assurance</td>
</tr>
<tr>
<td>UX</td>
<td>User Experience</td>
</tr>
<tr>
<td>RITE</td>
<td>Rapid Iterative Testing and Evaluation</td>
</tr>
<tr>
<td>FTUE</td>
<td>First Time User Experience</td>
</tr>
<tr>
<td>UCD</td>
<td>User Centered Design</td>
</tr>
<tr>
<td>BioSt</td>
<td>Biometric Storyboard</td>
</tr>
<tr>
<td>SCL</td>
<td>Skin Conductivity Levels</td>
</tr>
<tr>
<td>EMG</td>
<td>Electromyography</td>
</tr>
<tr>
<td>GSR</td>
<td>Galvanic Skin Response</td>
</tr>
<tr>
<td>TRUE</td>
<td>Tracking Real-Time User Experience</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>GEM</td>
<td>Games, Entertainment, and Media</td>
</tr>
<tr>
<td>CAVE</td>
<td>Cave Automatic Virtual Environment</td>
</tr>
<tr>
<td>VR</td>
<td>Virtual Reality</td>
</tr>
<tr>
<td>USB</td>
<td>Universal Serial Bus</td>
</tr>
<tr>
<td>UI</td>
<td>User Interface</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Overview

This thesis presents an investigation into the contribution of combining quantitative and qualitative measurements in Games User Research (GUR) within the larger context of Human-Computer Interactions (HCI). GUR professionals study tools that bring together different methods for gathering and visually representing playtesting data in order to provide unbiased feedback about the player's experience. A concrete understanding of players and their interactions can enable user researchers to collect more credible data to provide useful and motivating feedback when presenting the results of playtests.

In this thesis, approaches are introduced which use a variety of techniques to combine qualitative and quantitative data collection methods. The approaches are not designed to replace existing user research methods, but to extend and establish a combination of existing methods to provide versatile tools for games user researchers.

1.2 Motivation

Video games are becoming more popular every day and reach out to many industries. Since 2013, there have been 143 new game studios in Canada [38]. Video games can be seen as a form of entertainment with 155 million people playing video games in the United States of America [13]. It is critical that games are engaging to their audiences. There have been over 1200 new games developed in Canada in 2014 [38].

Therefore, it is important to ensure that video games have an optimized player experience. In order to make games more enjoyable for their users, there is GUR, which assists with this task. GUR is a field of work where one of their concerns are

\textsuperscript{1}often referred to users in HCI
with developing a set of techniques and tools to measure and study players’ behaviors to provide information for game developers in order to optimize the player experience.

There are two types of data that are gathered from the player and require evaluation. The first type of data is quantitative data and will often describe how many, how much, or how often an event occurs in a players’ User Testing (UT) session. This data is a set of numeric variables. The second type of data is qualitative data and is typically descriptive data and as such presents more challenge to analyze than quantitative data. The qualitative data will explain why quantitative has the results that occurred and how they came to be. Merging the two types of data together will speed up the analysis process.

My research and development falls under this topic as I want to make the GUR process more streamlined so other researches can improve the quality of their games efficiently. My work involves recording multiple types of data and creating a combined visualization in a clear and concise way for user researchers.

Effective representation of playtesting data is one of the current challenges for GUR. GUR is concerned with developing a set of techniques and tools to measure and study players’ behavior to provide information for game developers in order to optimize the player experience of their games. Thus, GUR tools are often used for the purpose of evaluation and gaining insights from players to enhance game designs decisions. One major challenge is the sheer quantity of data that needs to be analyzed and understood. This can be often daunting and confusing to decipher, requiring game developers to painstakingly review often large-scale playtest data to make improvements to their games. Hence, one important contribution to the GUR field would be making the interpretation of playtesting data effective and meaningful in terms of facilitating design decisions for game developers. To address this, we have been developing a tool to assist the collection and representation of playtesting data such as players’ in-game behavior (e.g., tracking their movements) and physiological measures (e.g., tracking their arousal state).

This thesis can be seen as a continuation of previous work on conventional user testing and biometrics [32] and gameplay visualization of telemetry data [33].

Games like Grand Theft Auto V ² and Destiny ³ have development costs in the $137 million (M) and $140M ranges respectively [2, 6] rivaling the budgets of major motion pictures which is showing large growth in the games industry. In the interest of minimizing expenses, these teams budget for playtesting and quality assurance (QA).

---

²Developed by Rockstar  
³Developed by Bungie
Based on SteamSpy's reports, there are about 1.4 billion games purchased to-date on the Steam platform, where approximately 25 percent of those are independent games. As of July 2015, over 3 million applications exist on the Apple App Store and Google Play [48]. Independent games have similar risks with larger game productions while having a much smaller budget. Smaller game studios often do not have the budget for arranging expensive playtesting, so there is a need to incorporate playtesting techniques for smaller teams and accommodate their budgets.

Video games are an industry that has exponentially grown in recent years with more players and developers for these games than ever before. New subfields are emerging because of this recent growth to the games industry. With these new fields within the games industry, comes a need for academic research which can assist with improving the processes involved with these new fields. One of these fields stemming from game development would be GUR.

Video games are becoming more popular every day and reach out to many industries. The successful video games in the game development industry have multiple influences for what created them to be a hit game. There can be influences from business such as marketing or influences from technical innovations but the influence that will be focused on is how GUR and playtesting positively impacts a video game. Understanding how a game's success can be quantified is important so that these positive impacts can be replicated for future developments.

One of the methods commonly used to score video game's success is by review scores curated from a group of game critics and reviewers. This can include the public audience as well because games are developed for commercial use so the consumers need to express their experiences. One of the common platforms for this is Metacritic which provides review scores based on critics and players. There are other notable review platforms such as Game Rankings and Review Trax which each provide unique review information for games in the market.

Other methods that can be used for measuring the success of a game. These can include but are not limited to the unit sales or revenue of a game, the hours played by the users, or the retention of the players. These can all be valid options for having a quantifiable measurement for one game’s success.

It is important to take into account that the games industry is changing from having an importance of units sold to a free to play model. This is important because it changes not only the way players play the game, it changes how games are developed as well. There is a heavier emphasis on micro transactions since the initial cost of the game is free.
This change in the industry will create more importance with having a good user experience. Without a good user experience, the game will have a poor monetization because players who do not enjoy a game will not spend money in the game. Video games must also be well designed such that the usability of the game does not hinder the user’s experience.

In summary, the industry’s independent game developers can be seen as a reflection of how students in academic institutions develop video games as they share similar amounts of resources and skills. Both types of developers also share similar challenges such as deadlines and evaluations from either publishers or professional evaluators. This thesis focuses on students in academic institutions and their lack of resources and skills directly related to GUR.

1.3 Working Definitions

Video games are an interdisciplinary research area, providing new research challenges to many fields. For example, research can range from a humanistic, artistic, and technical perspective of games. There is much literature in these areas, which cannot be covered in this thesis. While the focus is on GUR methods, the scope of this thesis is to study the interaction between a game and players, with the aim to provide informative feedback for game developers to assist with optimizing the player experience of their game in development.

1.3.1 Human Computer Interaction

Playtesting is becoming a common method to improve games and is building upon evaluation methods from GUR which is building upon psychology and HCI [34, 36]. Playtesting aims to improve a game’s design and player experience by performing a series of usability and user experience (UX) evaluations [31].

Video games have key differences from other software applications such as their purpose for use. With a software application, there is a real world task that needs to be completed and the user is assisted to complete the task with the software application. With video games, the objectives are fictional and only provide the user with an intended experience.

1.3.2 Games User Research (GUR)

In order to make games more enjoyable for their users, there is GUR, which assists with this task by helping reach their design goals by understanding players [59, 36]. Video games originally were developed for a Pay Once model and this caused chal-
lenges for the developers to sell their games. Having an initial paywall forced the consumer to commit to the game before having the chance to play it. This game is still used today in some games but there is a new model which is growing in popularity [9], the Free To Play model. With this model, developers release their game for free and make money through in-game purchases and ad revenue. This causes its own challenges for game developers. The new challenge is that the game developers must provide a high quality experience for the user in order for their game to be monetized [40]. This is where GUR could assist game developers with creating that quality experience the player has when playing a developer’s game.

There are limitations to this method of improving video game’s designs. For example, there are issues with gathering reliable data from users as described by Adams and Cox. The authors mention that having reliable questions within a questionnaire is useless without validity. Reliability refers to the consistency of a measure and validity refers to its ability to measure what is supposed to be measured [1].

1.4 Problem Statement

This thesis provides contributions to the field of Computer Science and specifically with areas of HCI and GUR. More specifically, this thesis will contribute to improving the process of common methodologies used in GUR. The focus of this thesis is to improve two specific portions of the process researchers would typically perform.

One of the important steps is the data collection process of a playtest session. For a playtest which calls for observation methods, the researcher will have to set up cameras to video record the user playing/interacting with the game. Depending on which stage of development the game is at, the playtest could also call for game metrics to be recorded as well. This will need to be programmed into the game code or could be manually tallied by an observer. This thesis provides the contributions that aid with these processes.

The second process is the analysis of data collected from the playtest session. One major issue that arises is the quantity of data that needs to be looked over. It can be often daunting and confusing to decipher, which may cause game developers to experience a slow, tedious task of going through the data to attempt to make improvements to their games [51]. Hence, one of the challenges is making the interpretation of playtesting data effective and meaningful in terms of facilitating design decisions for game developers.

Common self-reporting methods such as observation can be very time consuming and in turn, be very expensive to perform. This can be time consuming because
a researcher would need to monitor the initial playtest while taking notes on any interesting events happening during the playtest. An approach like this can cause researchers to possibly miss certain events that may have happened during the playtest. The current solution to this challenge is to have the playtest video recorded and have the researcher watch the recordings to find any events that may have been missed in the initial observation.

As an example, there is a 2 hour playtest for 10 players, a researcher would have 20 hours of initial observation and an additional 20 hours of rewatching that set of playtests totaling 40 hours which is approximately a full work week. The initial observation is unavoidable since the playtest needs to be 2 hours. However, the portion of the process where the researcher observes playtest sessions a second time can be greatly optimized using techniques that highlight key events for the researcher to watch.

1.5 Thesis Statement

Vixen is the combination of qualitative and quantitative games user research methods that provide a usable and interactive environment to evaluate playtest results for students developing games in academic institutions.

This thesis documents the design, development, and evaluations of the tool, Vixen, that I have created within multi-disciplinary teams of software developers and GUR experts. The aim for this thesis is two-fold: i) to demonstrate that a GUR tool for students developing games in an academic institution without proper GUR training or resources can engage in playtests for their projects, and ii) to evaluate the effectiveness of Vixen in use by teams developing games within an academic institution.

1.6 Contributions

My research and development falls under this topic as I have decided to develop my attempt at improving the GUR process more streamlined so other researches can efficiently improve the quality of their games. My work involves recording multiple types of data and creating a combined visualization in a clear and concise way for user researchers.

1.7 Organization

In this chapter, I have discussed what the game development industry is and why it is important to research game development related topics, as well as outlined at a high level what HCI and GUR are. I have also presented the main focus of the work
within this thesis, and described the contributions that my research has made to the field of GUR. The remainder of the thesis will cover:

Chapter 2: A brief overview of the current methods within GUR, as well as the tools that have been developed which contribute to the process of GUR. The motivations for why the tool is a need for game developers and researchers focusing on the topic GUR.

Chapter 3: An overview of the rapid prototypes and iterative design process that went into the design of the current iteration of the tool from my undergraduate studies. An overview of the success from the graduate GUR course prototype. Finally, additional comments on the final iteration of the tool where I explore the technical details that are working in the background.

Chapter 4: An initial study exploring the focus of my research, Vixen, which demonstrates on a high level how game data is visualized. This chapter will also explore Vixen’s versatility among various games. It explores the usability of the tool from the perspective of another game developer. Experienced Unity game developers are interviewed about what their opinion is towards the tool. The aim of this study is to identify the key usability and user experience successes as well as potential flaws in the design of the tool. This study also determines that the tool provides useful information for developers.

Chapter 5: An in depth study that includes three developers making different Unity games that use Vixen with the development of their game. The aim of this study is that Vixen saved time during data analysis process compared to other common GUR methods. There will be two protocols with this study, the game developers using the tool and the participants for the developers user testing sessions. The results of the studies are discussed, along with a conclusion to the thesis.

Chapter 6: Discusses how Vixen is beneficial to independent developers, how it provides meaningful information for developers, the contribution of GUR in regards to the tool, and how this thesis helps with time consuming analysis processes (contributions). Additionally, the limitations (challenges in GUR) of the work are presented with future areas of research are suggested.
Chapter 2

Related Work

2.1 Introduction

This chapter explores the background literature to the main contributions of thesis-identifying, measuring and communicating qualitative and quantitative data using Vixen. This literature review covers various insights into the field of HCI and the subfield of GUR. As computer games become more complex to develop, their global popularity has also increased [13]. A need for understanding players’ behaviors better and visualizing their in-game telemetry data becomes more valuable. HCI and GUR are fields that have a player or user centered approach to designing and evaluating games and software. Practitioners of these fields are becoming an essential part of game development in order to establish high standards of quality and usability.

The goals of this chapter are to provide a deeper understanding into the field of HCI and GUR to frame research effectively. The first section (Chapter 2.2) covers a review into games that have been developed within an academic institution. The section that follows (Chapter 2.3) covers a detailed review of different user research methods and techniques for performing playtest sessions. The second half of this literature review covers various data collection and visualization tools used in the industry (Chapter 2.4). The final section (Chapter 2.5) involves a summary of the most relevant topics related to the work presented in this thesis.

2.2 Games from Academic Institutions

Universities use video games as a medium for solving research questions or educational purposes [17, 45, 47, 52, 3, 46]. This unique subset of game developers share similarities to the independent game developers within the industry since the two types of developers share similar resource and time constraints. This section is a
key topic throughout this thesis as game developers who are developing for academic institutions are the main focus since they relate to other game developers in the industry.

The work presented by Bellotti et al. [3] describes an overview of serious games developed for academic institutions which is often the type of game built for academic institutions. The research explains the effectiveness of serious games in terms of learning outcomes as well as a systematic approach to enhance the design of serious games. An observation made by this research was that many studies conducted on games developed for an academic institution lack a rigorous assessment. The researcher specifically mentions the importance of evaluating the user performance.

Kapralos et al. [17] presented research focused on the impact of realism/fidelity and multi-modal interactions on learning. To approach this problem, the team began development for a serious game surgical cognitive education and training framework. The serious game was used as a research tool where various simulation parameters can be easily adjusted to suit the needs of the client. This project finished with a functioning prototype, however there was no evaluations performed on the project. This work did mention the reason for not evaluating the serious game which were working with limited resources. This team of developers could have benefited from a tool that assists with evaluating games.

Tawadrous et al. [52] performed an experiment to evaluate the effects of stereoscopic 3D viewing on knowledge retention. This experiment required the researchers to develop a game in order to find the results they were searching for. More specifically, the game was a serious game to train firefighters for fire safety. Once the prototype was complete, the researchers conducted the study with participants regardless if they were not apart of the target demographic (75% were game developers). This will not provide the researchers with accurate results for their study. The researchers needed to allocate more time to recruit the correct participants for their study. A tool that automates data logging and visualizations would have provided the researchers with more time to be allocated towards accurate demographics recruitment for the experiments.

Some games developed for academic institutions start as simple as a visual test environment, similar to the work presented by Stanfield et al. [47]. The research focuses on the impact of floating windows on the accuracy of depth perception in games using a small technical demonstration created with the Unity3D game engine. For the study, participants were mostly (95%) experienced with stereoscopic 3D so it showed the participants were carefully chosen. This study was preliminary as it
will require further evaluation as mentioned in the future work. Future studies can include a tool to assist with logging player experience to save on time resources.

More video games that were developed for academic research were the simulation-based serious games for anesthesia and epidural training [45]. These serious games were used to evaluate the main topic which was a comparison between seated and room-scale virtual reality. The work shows thorough evaluation of the two games, both having studies where participants simulate the experience of an anesthesiologist. The studies had performed observation on the participant and presented the player with a system usability scale and engagement questionnaire. An evaluation of player metrics were not logged for either of the experiments and can be seen as a missed opportunity.

Rapid Recovery is an excergame developed by Shroeder et al. [46] to provide a prototype game for shoulder rehabilitation and physical fitness. The work demonstrated a technical demonstration of what can be achieved with the Microsoft Kinect in collaboration with the fitness and rehabilitation device, Spincore Helium 6 Baton. The game described in this work was not evaluated but a usability test and effectiveness test was mentioned in the future work section to be planned in the next steps. These next steps can include a tool to aid with the implementation of testing methods for their future work.

Development cycles for these games often do not have dedicated time to user research or methods of evaluation. The reason there is no user research is because of the limited resources and skills among the team. There is a need for some tool to assist with providing user research to their games. This will provide the client of the project with an improved and more polished deliverable at the end of the day.

### 2.3 User Research Methods

HCI methods have made progress in understanding the usability of productivity applications and websites. However the specific characteristics of video games such as the addition of ‘intentional challenge and emotion’, mean that many established methods of user research cannot be applied in the same way [30]. Current methods of evaluating video games UX commonly include subjective self-reports through questionnaires, interviews, and focus groups as well as objective reports through observational video analysis [20]. Many of these methods have limitations when applied to video games.
2.3.1 Observation

Behavioral observation logs are the industry’s strongest analysis tool in video games user research. They can provide a basis for a detailed analysis of usability [39], as well as fun and game experience [41]. Observational studies (referred to as Behavioral Observation or Ethnographic Observation) is a technique where a participant is required to perform an activity (playing a game) while being recorded for future observation, or is directly observed by another individual. This technique is extremely effective since it is relatively cost-effective requiring very few resources and an observer can detect issues with player progression, player challenges, and if a player gets stuck or becomes unmotivated. An observer can also examine body posture and facial expression to estimate emotions [21].

While behavioral observation is great for gathering objective data and seeing exactly what players do, it is time consuming to record and analyze the data. Another drawback to observational studies is that the observer should be experienced enough to take objective notes which is not very intuitive for novice observers. Participants may also perform differently when being observed due to external pressure, meaning the behavior observed might not be completely natural [21]. Studying observational data as an indication of human experience is a lengthy and difficult process which must be undertaken with great care to avoid biasing the result [24].

An example approach for observation data may include the playtest area having a one way mirror or it is inside a modular room where recording devices capture the visual observation data from the playtest. One flexible camera is positioned so it can capture the user interacting with the platform they are playing with. A game capture device is connected to the PC or mobile device streaming the footage to the observers in another room. This allows for the possibility of developers entering the observation room to view the playtest in real time without disrupting the player.

2.3.2 Think-aloud

Think-aloud protocols builds off the observational studies method where the participant being observed is required to talk about their actions as they are performing them. The aim is to get inside the players’ thinking processes 'in the moment’, potentially revealing unobservable details and providing researchers with immediate feedback. Like observation, think-aloud can be recorded and viewed later. This technique is effective because it provides insight on the internal state of the participant providing great mental insight to gather behavioral data [22].
A downside to think-aloud is that it is unnatural for most people to talk objectively while playing without any prior training. Users may state several subjective claims about the play session, possibly skewing results. It is argued that ‘think aloud’ techniques cannot effectively be used within game testing sessions because of the disturbance to the player and ultimately the impact they have on game play [37]. Similar to observation, think-aloud is also very time consuming [22].

An example of this approach would be to prompt players to feel free to talk and express their thoughts during the playtest. A microphone is placed near the player to record any comments during the session. Users are not forced to express their thoughts unnaturally and for every action they perform.

2.3.3 Heuristic Evaluation

Heuristic evaluation provides a formal and accessible usability evaluation method, which can be used even before any code has been written. There are a number of different heuristic sets created for video game usability evaluation, including PLAY by Desurvire et al. (2009) [10], and Nielsen (1994) [37]. Although heuristic evaluation promises to be a low-cost usability evaluation method, it suffers with problems of subjective interpretation [57]. The heuristics chosen are variable and dependent on what needs to be evaluated. Some examples of heuristics are:

- Are clear goals provided?
- Are the player rewards meaningful?
- Does the player feel in control?
- Is the game balanced?
- Is the first playthrough and first impression good?
- Is there a good story?
- Does the game continue to progress well?
- Is the game consistent and responsive?
- Is it clear why a player failed?
- Are there variable difficulty levels?
- Are the game and the outcome fair?
- Is the game replayable?
- Is the AI visible, consistent, yet somewhat unpredictable?
- Is the game too frustrating?
- Is the learning curve too steep or too long?
- Emotional impact?
- Not too much boring repetition?
- Can players recognize important elements on screen?

Heuristic evaluations are effective since they require fewer people to conduct a test, however finding the correct expert to test with is not always easy [21]. The above guidelines suggest design changes grounded in a structured vocabulary agreed upon with the developers.

2.3.4 Questionnaires

Questionnaires are paper based or delivered online where they can be sent to a large number of participants with little effort. If the goal is to understand what people do, a questionnaire will be a sufficient method to employ. If the researcher needs to
identify why something occurred, an in-depth interview or focus group will provide more valid responses than a questionnaire [1].

When designing questionnaires it is important to consider 'reliability' and 'validity'. Reliability is the consistency of a measure and validity is the ability to measure what it is supposed to be measured. Questionnaires should not be too long, since longer questionnaires are completed less accurately. Additionally, obvious questions, repetition and readability should be focused on to ensure participants do not misinterpret complex questions and the frame of reference is clear. Questions should be grouped together under common themed headings to contextualize subsequent questions. Questions should be kept as short and simple as possible without complicated double negatives or loaded words while also avoiding leading questions. Questions should not require any interpretations regarding the meaning of a term [1].

Although questionnaires are very useful, but depending on the number participants and the primary goal of the tests one may not utilize this approach. Instead one would conduct interviews as explained below.

2.3.5 Interviews and Focus Groups

Interviews are conducted on a one-to-one basis and require a large amount of the investigator’s time for transcribing and coding the data. Focus groups are generally comprised of one investigator and a number of participants engaging in discussions [1].

Interviews and focus groups are a time consuming process that requires careful planning to obtain more detailed and thorough information on a topic than a questionnaire. During the interview, an investigator often follows a schedule of pre-prepared questions but can deviate when necessary in order to maximize information obtained. The more structured an interview is, the less likely it is for a participant to feel at ease and reveal important and relevant issues. The interview should be flexible enough to allow jumping between questions since sticking rigidly to a structure can annoy respondents [1].

Interviews and focus groups should follow fours main stages: background, letting off steam, addressing issues and tying up/debriefing. In the background stage it is important to establish a trust relationship with the participants to make it more comfortable for users to give more detailed information. The letting off steam stage is to allow participants to unburden themselves of issues they have so they don’t resurface later on in the session. The addressing issues stage is where the interviewer
must be careful not to bias the respondent. Lastly, the tying up stage is where the interviewer summarizes all of the issues re-affirming the information is correct [1].

When selecting playtesting methods one would chose to combine different approaches to mitigate the weakness of an individual approach. An example of this would be to combine methods from Rapid Iterative Testing and Evaluation [27] with First Time User Experience [9] and Use Centered Design [28].

2.3.6 Rapid Iterative Testing and Evaluation

A number of industry-standard usability approaches use traditional methods mentioned previously in combination; for example, Rapid Iterative Testing and Evaluation (RITE) [27] which employs observation and think-aloud techniques with the addition of an attending software engineer to rapidly alter the design based on the findings of usability testing. Changes can be made after observing as few as one participant, with altered designs subsequently tested on the remaining participants. Other variations include open-ended usability tasks, paper prototypes, and empirical guideline documents [28].

The RITE method is a discount usability test conducted in a fast and highly collaborative manner [27]. We also decided to focus on the first time user experience (FTUE) for each playtest. The projects at XL are still in early development and we decided that focusing on the first hour experience would be most beneficial for the teams. Based on the important factors in game testing and evaluation by Pagulayan et al. (2002), the following were areas we focused on using the RITE method: Game designer intent, ease of use, basic mechanics, starting a game, tutorials and instructional gameplay, camera, in-game interfaces, mapping input devices to functions, challenge and pace.

An approach following a user centered design (UCD) process can be based on methods developed by Pagulayan et al. (2002). A key part of UCD is understanding design intent, where the general design goal of a game is to create enjoyable experiences for its players [28].

2.3.7 Mixed Methods

This section looks at details of methodological studies where their main goal was to introduce a new combination of mixed methods for a better evaluation of player experience [44].

Microsoft’s Tracking Real-Time User Experience (TRUE) is a system built to gain deep insights into user behavior and improvement of design for other complex
systems such as games [18]. TRUE has 4 main traits: events in context, attitudinal data, visualization, and video. The events in context look at streams of data rather than aggregated frequency counts. The attitudinal data displays a brief survey at the end of each race or how it’s done in the Halo 2 study (every 3 minutes, ask a question relating to this part of the game they are on). The video captures users interacting with the products, which is synced with the timestamp for each event. The video is now automatically indexed which allows the ability to jump to positions in the video that we are interested in. This is an effective method for evaluating quantitative data with qualitative data. The case studies covered Halo 2 and Shadowrun being tested with TRUE. The system proved to improve the quality of design in the game. This work could have improved showing the visualizations trait in the case studies. By having the integration of linked video to back up the attitudinal data would have been beneficial.

Canossa et al. [8] explore patterns of behavior that could point to potential frustration before players resolve to quit a game. Their method is applicable independently from personalities of different players. The authors showcase potential behavioral data to visualize players’ interaction and navigation through a game environment. The authors decided to visualize the qualitative and quantitative data in their methods. They use spatial visualization and it allows experimenter to see through the eyes of the player in a manner similar to a video recording of a game session. This comes with the added benefit of having game metrics mapped on the game environment and result with quantitative results. This work’s limitation is that they have manual qualitative data logging. The authors have a follow up interview to confirm segments of play which resulted in undesirable frustration. This thesis will address the need for more automated qualitative and quantitative data visualization.

Mirza-Babaei et al. [32] created a new method called biometric Storyboards (BioSt) and compare it to classic user testing methods. The author presents the outcome of designing a game completely without the use of either classic user testing or BioSt user testing. The authors findings argue that BioSt can help video game designers create a better gameplay experience and deliver significantly better visuals and more fun to the games being developed. The tool visualizes data gathered from skin conductivity levels (SCL) and electromyography (EMG) measurements and user test sessions. Both of these were recorded because of their ease of use for tracking the data and visualizing the data. From this data, they were able to visualize smiling and frowning. They used these key points to indicate where along the timestamps did the player have greater than usual user experiences. The authors found using
either classic user testing or BioSt user testing will improve the gameplay of video
games. The work presented a line graph which is referred to as an aggregated player
experience graph, representing what designers think the exciting gameplay moments
are. The graph also indicates areas of difficulty and average time spent in each game
beat. However, using BioSt user testing will provide design feedback and more subtle
changes to game mechanics. The results did not show significant difference between
classic user studies and their style of studies and can be argued as a minor limitation
to the method. This thesis will explore a novel method that show different results
from classic user test methods.

Following more recent work by Mirza-Babaei et al. [33] the tool which visualizes
SCL with color coating the quantitative data lines in a Super Mario Brothers game.
Verbal comments are visualized using comment bubbles seen in the game world. This
work presented a work-in-progress of the tool which shows data of users who traverse
through levels designed by Eddie Shearer. The tool manages to show three different
sets of data. These sets of data are player position, user biometrics such as SCLs, and
finally verbal comments. The position data is visualized with lines in the game world,
the biometric data is represented with color spread across the lines, and the verbal
comments are represented by text bubbles in the game world. Both of these are clear
and concise visualizations of think-aloud data mixed with game metrics. This work’s
shortcoming is not evaluating their proposed method with their target users such as
game developers. The proposed method also does not accommodate for users who
may be colorblind and they will struggle to understand the biometric data. Overall,
this unified data visualization does propose a simple to understand method which
most developers of various backgrounds will have no problem adopting it into their
games. This work relates very closely to this thesis as it presents a tool to visualize
qualitative and quantitative data sets in games from various mixed methods.

2.3.8 Challenges of Mixed Methods

One major issue that arises is the quantity of data that needs to be looked over. It
can be often daunting and confusing to decipher, which may cause game developers
to experience a slow, tedious task of going through the data to attempt to make
improvements to their games. Thus, one of the challenges is making the interpretation
of play testing data effective and meaningful in terms of facilitating design decisions
for game developers.

There are several ways to interpret data from a playtest session. Previous work
has worked towards creating tools and systems that facilitate both qualitative and
quantitative data recording and visualization. Others have created tools that accommodate for specific measurements of data like game metrics. Game metrics are various forms of quantitative data that are affordable to implement into an independent development studio’s game. These metrics can involve a wide range of topics such as in-game purchases or in-game events such as the amount of times a player collects an in-game item.

Once the data has been recorded, the data now needs to be visualized in a clear and concise way. However, the data needs to also be brief for presenting to the game development team to help save on time it takes to develop their game to the next iteration.

2.4 Data Collection and Visualization Tools

There are several ways to interpret data from a UT session. Previous work has attempted to create tools and systems that facilitate both qualitative and quantitative data recording and visualization. Others have created tools that accommodate for either qualitative or quantitative data. The related works will go through work that researches only qualitative data or quantitative data. These works will still be relevant because one can theorize how to add the missing data type into the methods presented in the works. Before looking into the visualization of data, it is important to understand what data is of interest to record. With so much data to record, it presents a challenge to come up with which data is useful for improving video games.

In this section, one can find the common data that will be logged by tools created in the GUR industry. Each tool tracks different data for specific reasons.

One form of data that game developers track to help improve their games are game metrics [26, 18, 19, 15]. Game metrics are events that occur in the video game such as a character dying or reaching a checkpoint. These are important because they can give an overview of how players perform. An example of a tool which collects game metrics is Data Cracker. These collected game metrics are saved for later visualizations to find concerns in the design of the game’s levels. It is important to store these game metrics so the data can be visualized at a later time when the developers want to see how to improve their player experience.

Previous research has shown that tracking more than game metrics have helped improve the quality of video games. Mirza-Babaei et al. [32] present Biometric Storyboards, which record user SCL and EMG measurements within UT sessions. These types of data are known as qualitative data. This type of data is also seen being logged in other work since it is important to have both types of data so one
can understand why the quantitative data appears to be the way it is. Galvanic Skin Response (GSR) is another term for SCL and it can be found in other previous works as being a source of meaningful data to developers [32, 33, 53].

Other previous research has proposed that it will be beneficial for the developer to see both game metrics and biometric data from the user. Recent work records the users’ in-game character position and SCL levels as well as verbal think aloud comments [33]. Tracking these unique data sets provides clarity towards what exactly the user was doing in the game and how they were feeling. Canossa’s et al. [8] work records game metrics, video recording of user, screen capture, and in-game character position. This work was able to track both qualitative and quantitative types of data.

Another previous work introduces a tool called Lithium, which tracks combat and influence of users in a particular area of the game [15]. This includes bullets fired by characters, user movement and many other in-game metrics. Lithium is able to later visualize this data for spectators who are watching the users play the game. For the developer, this information can be very useful as it will be able to depict motivations the user has towards their actions in the game which can allow for the the developers to improve the level design and player experience.

2.4.1 Quantitative Data Visualization Tools

Once all of the important data that assists with improving a video game has been recorded, the data now needs to be visualized in a clear and concise way. This data needs to also be brief for presenting to the game development team. Quantitative data refers to where an event occurs, when does an event occur, and how often an event occurs. There are different ways to interpret and visualize quantitative data.

Earlier advancements in the GUR field focused on adapting classical HCI user evaluation methods to player experience evaluation [4]. This was a necessary step when establishing the foundations of a new practice. However, player experience is too complex to solely measure through one technique [44]. To progress the field, it is necessary consolidate the understanding and practice of integrating multiple techniques within a meaningful research study, and to address reporting of results as a key component of the research process [51].

However, gathering information can sometimes be a complex process, as this gathered data generally requires context. Otherwise, it may be interpreted incorrectly, and be used to draw misleading conclusions about gameplay. Microsoft’s TRUE [18] system provides survey data gathered during playtime to show a user’s emotional state as they play a game, giving emotional context to a play session. Similar efforts
have resulted in a push towards visual game analytics [26], which focuses on data visualization in order to allow for greater accessibility to all members of a game team. Having data represented in this "casual" [26] manner allows for greater amounts of feedback and reflection from designers which is ultimately beneficial for a game.

Visualizations aimed to assist game developers to better explore and analyze large gameplay data sets in order to better optimize their intended player experience. Although these graphical representations often make complex gameplay data more understandable [55], most of the current visualizations focus on displaying player behavior but do not address qualitative information on player experience, attitude, emotional experience, or motivation with some exceptions being, for example, the work of Kim et al. [18] and Mirza-Babaei et al. [32, 33]. The latter, for instance, shows relationships between a player’s physiological changes (e.g., change of feelings) and game events.

With respect to the analysis of movement data in particular, different path visualizations have been proposed so far. Commonly, the traces of different players are plotted individually and visual properties are used to depict a certain attribute. For example, Hoobler et al. [15] varied the thickness of the paths to depict the elapsed time. Similarly, Dixit and Youngblood [11] used color cycling to represent the flow of time. Wallner and Kriglstein [55] also used color-coding to reflect changes of various tracked attributes and coupled the continuous path representation with a nodelink visualization to represent discrete events. A more in depth review of path visualization can be found in Wallner and Kriglstein’s literature review [55]. However, to our best knowledge, none of the previous approaches have provided a 3D interactive environment for game developers to analyze and explore playtesting data.

Therefore, there are key research questions around reporting, visualization and analytics to be explored further [43]. Effectively reporting findings is not simple in GUR because games researchers and practitioners come from a variety of backgrounds (psychology, HCI, QA, production, market research, software engineering). There is not always consensus on how to effectively and validly report findings. Additionally, the complexity of findings (for example physiological measurements or game analytics) can mean that very detailed results must be communicated in a way that can be understood by people from a variety of backgrounds.

Analyzing the quantitative data with graphs has been the traditional method [26, 35, 55, 50]. The time is displayed along the Y and the game metric being evaluated is on the X. This has been an effective method for visualizing quantitative data such as game metrics. The limitations with this method however are that the user researcher
will not be able to see information such as where these events took place in the game. This information would be useful to the researcher since it will easily point out the areas that need work.

Ravaja et al. [42] showed in their research that quantitative data can be used to assist with the interpretation of qualitative data problems. Their work examined the emotional valence and arousal-related phasic psycho-physiological responses to different violent events in games. They collect this data through SCL and EMG activity. What they found is that the players have negative emotional responses to killing opponents and a relieving emotional response from when their own character died. They chose to visualize the results in graphs.

The solution to Data Cracker’s limitation is the work presented in Mirza-Babaei’s work [33], which visualizes the player position with a line in the game world. This accurately, quickly, and simply visualizes the users’ location and certain game metrics such as where the user died. Using line rendering is a common method for representing game metrics such as player position [11, 15, 33]. However, there are limitations with this work as well, such as, keeping track of and visualizing when certain events that occur.

There is other work that has been developed that visualizes in-game events and when they occur such as, e.g., Canossa’s et al. work [8]. Their work visualizes the game metrics with colored dots over the game world. They use the color of the dots to visualize time along the player’s movement. The other shapes represent various game metrics. One of the limitations of this work is that it can only visualize one UT session at a time. If more sessions were to be mapped over one another, the visualization would be too difficult to interpret.

Another common method for visualizing quantitative data is through the use of heat maps. Lithium, a tool to assist spectators commentating competitive multiplayer games, visualizes the game metrics with heatmaps and lines in the game world [15]. The lines in the game show what the location of player’s characters are relative to the game world. They have an interesting way of dealing with time for the lines. Thicker lines represent more recent time for that location. In Kriglstein and Wallner previous work, they present their recorded data with heatmaps and data clustering [19, 54]. The data clustering is interesting because of the choice to use enclosure and percentage based representations.

InfoVis is another tool which is not used for games but for visualizing the connections between papers [14]. Faisal et al. [14] do this by connecting lines with arrows pointing towards the paper being visualized. The obvious limitation to this work is
that it is not visualizing game metrics. The approach to how the data was visualized is the important part of this work. In an attempt to visualize clustering of data, it falls short in simple visual interpretation of the data when compared to work done by Kriglstein et al. [19].

### 2.4.2 Qualitative Data Visualization Tools

Accompanying quantitative data is data that answers the questions why and how. Qualitative data can often justify why quantitative data is appearing to be the way it is. Qualitative data can also show how the user was feeling during their UT session. In this section, present examples of qualitative data tools for GUR.

Sometimes qualitative data is not visualized within a tool but is instead used to answer why data looks the way it does. In this case, a video recording of users would verify quantitative data [8]. This can cause limitations as it can be very time consuming to interpret this data since video recordings of a UT session can be hours long. This can be particularly bad if multiple video recordings need to be analyzed.

There has not been much research done in this area where one tool would visualize for the researcher both qualitative and quantitative data. Here is where a clear gap in research is found. This gap needs more research and development because bringing the two data types together and visualizing them simultaneously saves time for the researcher. The position of a games user researcher can be very time consuming so any way to save time is invaluable. That being said, there is some research revolving around the unified visualization of both qualitative and quantitative data.

Blending the two types of data could be achieved in this example where Mirzababaei et al. [33] visualizes the SCL readings on the position line rendering. The color is pulled from qualitative data such as SCL and the lines are pulled from quantitative data such as the in game character’s position. The limitation that can be seen from looking at the qualitative data is if the user was colorblind, that data would be unclear. An alternative idea that could solve this problem would be to use line thickness to visualize the qualitative data. Another option would be to gray scale the data seeing as this qualitative data is simply a scaled value between yellow and red.

It is worth noting that in the work presented by Canossa’s et al. [8], they blended the two types of data together but without visualizing the qualitative data. They decided not to visualize the qualitative data because they collected that data through follow-up interviews [8]. Visualizing the interview process alongside the quantitative game metrics data creates unique challenges. Representing that data clearly and
correlated to the quantitative data explains why that data is the way it appears to be. This challenge can be seen as potential future work.

Woods’ et al. [58] work takes on the challenge and visualizes both qualitative and quantitative data in the software Transana. Transana captures in game screen recordings and video recordings of the user in one section, which is displayed next to think-aloud text. This is beneficial for a GUR team as it reduces the time it takes to get all this data together, synced up, and visible for analysis.

2.5 Summary

This chapter explored multiple research methods that are often applied when studying interactions and player behavior across the gameplay experience. This chapter also explored multiple tools that are designed to gather and represent data relating to video games. A key area to understand is the effective methods of measuring the player experience.

This chapter shows an idea of the types of data that can be visualized to assist with interpretation of video game data and enhance the player experience. An effective tool for visualizing both qualitative and quantitative data would take the best aspects of work presented in this thesis and apply it to one tool. Some of the best aspects would include using line renderings to visualize player location in the actual game world. Another visualization for quantitative data would be for larger data sets and that is to heat map the data. When looking at in game metrics, line graphs will easily visualize that information for the researcher.

The work presented in this thesis has a couple limitations that are quick to spot. First, the tools only work for one game and they are not dynamic for all or even most games. Another limitation is that the tools do not do a good job visualizing the user’s facial expression.

The current state of this GUR tool is a framework to design and develop new and innovative features to help improve the current processes for completing user testing. There are already improvements being made with games through optimizing game developer’s level designs. These improved level designs will be able to aid game developers in creating a more enjoyable user experience for their players. This tool has saved developers time in their production cycle through more intuitive data representation methods and simpler data collection styles.
Chapter 3

Tool Development

3.1 Introduction

This chapter explores the initial designs of Vixen and the iterations that grew into the current development build. Through exploring related GUR and analytics tools, I aim to gain a better understanding of how to effectively visualize data within the tool, Vixen. Additionally, by exploring the game development process in an academic institution, I aim to gain a better understanding of developing a positive user experience for software design tools.

One of the current challenges in GUR is the time it takes to perform playtesting as well as deciding which methods to perform to better understand player’s experiences with your game. Vixen aims to save time by automatically collecting important data regarding your user’s experience through various qualitative and quantitative data using mixed methods. A summary of the features in this tool include: recording player position, orientation, heart rate, distance from screen, video of participant, and audio. All of these features are visualized with the goal of utilizing simple to understand methods. This recorded information is transferable from participant to developer over a server. The importance of each feature is explained in detail throughout this chapter.

This chapter begins with an example situation where a game development team is motivated to include a tool similar to Vixen in their work cycle. Following this, I explore the initial prototypes that were designed before beginning my masters. In the next section, I walk through the major revisions and new features that were developed before the studies were conducted. Finally, the chapter concludes with a summary of the experience that explored the idea of commercialization of the tool, Vixen, through the Brilliant Entrepreneurship Incubator program at UOIT.
3.2 Scenario

An independent game developer or a student developing a game has recently started a project with a small team. The team works through developing the game with the procedures they are experienced with. The team is nearing the end of development for their game, however they do not have the budget or time to complete one-to-one testing with many users in a meaningful way. Conducting a test session and going through gameplay and webcam footage would take a significant amount of time out of development for them. They want to understand each player’s experience with the game, and pinpoint issues with specific gameplay sections. They decide to turn to the Unity Asset Store to find any easy to use GUR tools. They discover our tool in the store, and try out the “lite” version to see if it is worthwhile. They prepare our tool’s prefab in their game and attach all the line rendering and webcam scripts to the player ‘game object’. They perform an informal test session with the employees at the studio, and are able to visualize the game data in their game in a meaningful way. The data from each test session is automatically uploaded to their server, so they can easily access the data. They see the route that each employee took through a specific gameplay session, and they are able to playback webcam footage to see each employee’s reaction during their play session. The studio decides to buy the full version of the tool, and creates an .exe build of their game with the tool installed. Since they do not have the budget for a formal lab test, they decide to distribute their .exe to willing participants to play on their own time. They wait to hear back from their participants. After a few days they start to receive folders of data exported to their server. They can then use this data on their internal build to visualize how each of their participants played through a gameplay session.

3.3 Initial Prototypes

This research went through multiple early stage iterations in order to fail quickly and have a strong foundation for Vixen. In the earliest research and development stage, Vixen established the proof of concept with assistance from a summer research grant. The project continued development in the Incubator program at UOIT and explored the opportunity of taking off into the market. Vixen received a major iteration during the GUR course where I teamed up with a small group of undergraduate students to implement new features and improve existing ones. The next iteration of Vixen came from UOIT’s Brilliant Entrepreneurship Incubator program where the same team from the previous iteration assisted with adding finishing touches to the tool.
and preparing for an opportunity to submit this work to the Unity Asset Store. The final iterations of this work were the studies presented in the following chapters of this thesis.

3.3.1 Version 1.0

The goal of the initial prototype from the summer research grant was to recreate a tool based on the core mechanic of the work presented in Mirza-Babaei’s [33]. The first build of Vixen can be seen in Figure 3.1. This initial build presented a rapid prototype to prove the concept that a replication of the previous work was possible. The paper [33] presented a series of line renderings that had biometric readings associated with them. The first stage was to recreate the line renderings, then the biometrics since the biometric readings would be an overlay for the line renderings.¹

There were also text bubbles with player’s thoughts in the game world to assist with understanding what the participant was thinking at that particular point in time. This idea of presenting player’s thoughts with text bubbles is now commonly seen in Super Mario Maker. The text bubbles was a feature that was originally explored but ultimately removed due to it not appropriately being rendered in the 3D worlds that Vixen was later evaluated in.

Once the line renderings were developed, I used the Iom Lightstone Device to handle the recording for the GSR readings. This was chosen as it was the device I had available and could perform the task of collecting GSR data from a participant. There were some limitations to this device however, such as the limited amount of movement the participant could do before the data was inaccurate. Another limitation to the device was the intrusively large finger clamps, making it very difficult for the device to be used while holding a controller or moving a mouse around. The intent of the IOM Lightstone Device was for meditation purposes, but for this prototype it was acceptable to prove the concept that biometrics could be recorded and displayed over a game as an extra meta-layer.

This version of Vixen had a simple and effective UI element to allow the user to save and load the playtests participants completed. This UI allowed the user to name their playtest session and choose where to save the playtest data. The goal of this UI element was to provide a simple interaction that the user can complete at the end of each session to better organize each playtest session.

After completing the biometrics feature of Vixen’s initial prototype, I began showcasing the work at the Institute of Electrical and Electronics Engineers (IEEE) Games,

¹This iteration of Vixen was worked on with a fellow developer David F. Arppe.
Entertainment, and Media (GEM) Conference as a demo abstract [12]. At this event, I gained valuable insights from others in the field. Vixen received overall positive reviews from the people who explored the tool, pointing out how the tool is effective at interpreting player movement in a game world. The only concerns people had were with the way the tool tracked biometrics. This concern was mentioned because the device used was low fidelity and may not add significant value exploring further since it can be highly inaccurate. I also received feedback on the UI element requesting for a series of check boxes to allow for an option to filter the data. You will see that this feedback was integrating in the next iteration of Vixen.

3.3.2 Version 1.1

Following from previous research, I collect movement data, which is obtained at regular time intervals using telemetry methods and physiological data, which is captured
Figure 3.3: Vixen’s With Oculus Rift Virtual Reality Headset (August 2014)

Figure 3.4: Vixen In The CAVE (August 2014)
in the form of GSR to measure the player’s arousal state [33]. The movement data is represented by rendered lines that show the path of the players and has a specific color based on the recorded GSR value [23]. These colors range between green (lower level of arousal) and red (higher level of arousal) to provide a clear visual interpretation of the player’s emotional state throughout their playthrough.

Therefore, there are main key questions around reporting, visualization and analytics [43]. Effectively reporting findings is not as simple in GUR, because game researchers and practitioners come from a variety of backgrounds (psychology, HCI, QA, production, market research, software engineering). There is not always consensus on how to effectively and validly report findings. Additionally, the complexity of findings (for example biometrics or game analytics) can mean that very detailed results must be communicated in a way that can be understood by people from a variety of backgrounds.

The first goal for this version of Vixen was to build off of the previous version of Vixen with the goal of implementing improved systems for biometrics and line renderings. This was our primary goal because the previous version of Vixen was a prototype to showcase the proof of concept and receive feedback from the industry. Reviewing the source code with the new team created an agreement that the current systems can be improved. The next goal was to add additional features to provide the user with more playtesting data from new recordable components. This was chosen to be the next goal because with the current line renderings and biometrics, there was not enough information to recreate the playtest for the user of this tool to draw meaningful conclusions. The third goal of this iteration was to explore the idea of bringing this work to market using the Unity Asset Store. Being apart of the Incubator program at UOIT required the team to create a business plan for this product and prepare a business.

Being able to interact with the data in 3D environment is a key feature of the tool, where users (game developers) can explore (walk in the game) and interact (zoom in/out, show/hide data) with the visualization either using their computer screen for 2D viewing such as Figure 3.2 or a Cave Automatic Virtual Environment (CAVE) for 3D viewing (Figure 3.4) and an Oculus Rift Virtual Reality (VR) headset (Figure 3.3).

Figure 3.2, shows the approach for the first goal, to improve the line renderings and biometrics. One of the major concerns regarding the line renderings was the thickness

\footnote{Within the incubator program at UOIT, I was able to begin working with a new team of developers, Thomas Galati and James Simpson.}
and the color output. The line renderings were reduced in thickness to accommodate for instances of where many lines would be rendered in similar locations within the game world. The colors were adjusted to better fit according to the work of Mirza-Babaei et al. [33]. With the two changes to the visual output of the line renderings, the data could be interpreted more clearly. The choice to make the line renderings thinner was acceptable because the tool provides the user with the ability to move the camera around in the world allowing the user to move the camera closer and visualize the fine details of the data. The color change from light blue and white to green and red provided an improved interpretation of when biometric data changes occurred because these colors did not blend easily with one another providing a better gradient of color outputs which the previous colors could not achieve. The front end changes were not the only changes made to the graphical outputs as there were optimization opportunities available in the back-end of these systems.

For the process of the line renderings, the system had a series of unoptimized Unity line objects being rendered from each game object position transform logged to file. These objects were good for demonstrating a proof of the concept but were not practical in large scale as they took too long to load to be able to provide the user with a positive user experience when working with Vixen. To improve this system, OpenGL functions were used which had an instantaneous load time compared to the previous version. Low level GL functions provided the system with additional features for better customization of the color output for the line renderings. Before running the game to see the visual outputs, users would be able to select what color outputs they desire for low and high level of arousal. This addressed a concern provided at the IEEE GEM conference where an attendee mentioned the case scenario where a user may have a form of color blindness that causes specific colors to not be recognized.

The UI for this iteration was improved to include a check box list for each playtest session as shown in Figure 3.5 and Figure 3.6. This provided an additional layer to the interactions available to the user.

Once the first goal of this iteration was complete, the next task was to develop new features that provide the user with more information from the playtest sessions of their participants. The two new features that were developed for this iteration were video and voice recordings of the individual’s playtest session. Both of these new features were rapidly developed for this iteration to prove that these types of recordables can be collected and then later outputted to the user. The webcam output feature can be seen in Figure 3.7. There were no visuals associated with the audio output. For the webcam prototype feature, it was developed as an external C++ executable that
Figure 3.5: Vixen’s Improved UI Element (September 2014)

Figure 3.6: Vixen Bird’s Eye View of Unity Demo (September 2014)
ran off of OpenCV which is an open source computer vision library. Unity would be responsible for launching the executable and then the user would be responsible for closing the executable once the playtest session was completed and the audio recording and output was all automated. The webcam executable was responsible for logging each frame the webcam captured and then at the end of the session, convert the file to an appropriate filetype that Unity is compatible with. The audio recorder was developed separately from the webcam system as it allows for more freedom and control over what is being recorded for each playtest session.

With the additional features came more information being recorded and this created a new challenge of how to organize the data. Originally, all the data for one participant could be logged to a single text file. However, with video, audio, and text files for a single playtest session, I had to reevaluate structure for data management. In this iteration, our approach to this problem was integrating all of the data into Unity’s asset folder directly. Later in the thesis, I will discuss how this was not be best approach. In the asset folder held a folder containing all participant data, following that was a folder for each individual participant, then each participant’s data was separated into a .mp4, .mp3, and two separate text files. The agent or main script would handle retrieving the data from these folders and then pass it off to the appropriate rendering scripts.

The final task for this iteration was to build a business revolving around this work. This assisted the development of Vixen with evaluating who exactly will benefit from this tool most. The evaluations were completed informally as the team asked a variety of game developers if this tool would fit into their workflow and would be interested
in using this as opposed to their previous methods for evaluating games. From these evaluations it was discovered that the target audience would be independent game developers with limited resources and experience. The triple AAA companies already had tools in their workflow and would not be interested in switching to Vixen. With this information, some features were no longer pursued such as the integration with the CAVE and the Oculus Rift as most independent developers do not have access to these hardwares.

3.4 Version 2.0

With a new iteration to Vixen there was a new team and a new set of goals to accomplish by the end of the development cycle\(^3\). This version of Vixen was planned to have many new features and optimizations which became a manageable task with the assistance of the new team members.

Each of the following features presented have been integrated from the feedback I received between iterations. The features that have been integrated for this version of Vixen are:

- Server accessible data
- Orientation data
- Physical distance away from the screen
- Improved hardware implementation for biometrics
- Evaluations with multiple games
- Additional cleaning and optimizations

Each of these features are explained in more detail throughout the following sections of this chapter.

3.4.1 Line Rendering

This feature provides the developers with information based on the position and orientation of the game object chosen by the developer, typically the player’s character. The scripts involved tracking the location of game objects and their direction they

\(^3\)This version of Vixen was developed with students apart of the GUR course offered by Dr. Mirza-Balaei. The new members to the team consisted of Chris Carmichael, Liam Svirk, Derek Chong, and Montgomery Alban.
Figure 3.8: Vixen Setup Interface for Developers (April 2015)
are facing. This information is saved to file and then visualized as line renderings with arrows inside the lines to represent the position and orientation of the player.

This information is able to help understand the flaws in the level design as it provides a visual representation of the player’s navigation through the game’s levels right in the game itself. This can provide insight on areas of the level that may not be traversed or show signs of a person struggling to traverse. These visualizations can quickly show the developers the right areas of their level to fix in order to improve the user’s overall experience.

Following from previous research [33], I currently collect movement data, which is obtained at regular time intervals using Unity game object’s 3D position vector. This set of 3D vectors are stored to file with an accompanying timestamp to ensure that each point in space is accurately connected. The movement data is represented by rendered lines that show the path of the player [56].

To visually mark each timestamp, a 2D arrow sprite is rendered at each location. These arrows are oriented in such a way to represent the direction the in-game character was facing at that timestamp. This provides designers more insight on how users specifically traverse the game world. The visual orientation can aid designers with understanding segments of levels that may be a point of interest to the player. This implementation provides a lightweight system that does not have any hardware prerequisites.

Tracking an object in the scene requires adding the tracker script component to the game object you wish to track and it will record the object’s position and/or orientation.

3.4.2 Heart Rate

Originally, the physiological data was captured in the form of GSR to measure the player’s arousal state similar to Mandryk and Atkin’s work [23]. The movement data is represented by rendered lines that show the path of the player, each line has a specific color based on the recorded biometric value [33]. Following previous research, these colors range between green (lower level of arousal) and red (higher level of arousal) to provide a clear visual interpretation of the player’s emotional state throughout their playthrough. Raw data captured from GSR varies from player to player so it is important to normalize the data if a comparison of player data is to yield meaningful results. Following a GSR normalization formula used in Mandryk and Atkin’s work [23], where GSR min +/-3.5 and GSR max +/-3.5 are the minimum and maximum window centered on the data point at time t. This 7 second window is based on
Figure 3.9: Top Down Space Shooter in Unity (April 2016)

Figure 3.10: Space Shooter filtered to 3 unique participants (April 2016)
normalizing each point by the previous 3.5 seconds worth of points and the future 3.5 seconds worth of points to get a seamless transition between colors.

Changes in GSR values may provide clues as to changes in players' emotional state at any given time during gameplay. For example, a high GSR could indicate that the player is experiencing high arousal level. Given that a low or high GSR can only indicate a possible change in emotions, it is important to correlate this data with information presented by other data to more accurately estimate the emotional state of the user. This is done by logging a timestamp for both the position and GSR readings, followed by syncing the timestamps together so the two sets of data line up.

This feature later changed to provide heart rate information that is gathered via a Polar H7 heart rate sensor strapped to the user. The H7 is bluetooth-enabled, and therefore no wiring is required. An external process is used to poll the H7 periodically. The data that is gathered by that process is then saved to a file, which is then read by the heart rate module in Unity. This data is visualized as color on the orientation arrows, which can be customized as needed.

The user's heart rate provides important clues as to their emotional state at any given time during gameplay. A high heart rate could indicate that the player is feeling excitement, anger, or frustration. A low heart rate could indicate that the player is feeling relaxed or bored. Given that a low or high heart rate can indicate a range of emotions, it is important to correlate heart rate information with information presented by other modules in order to correctly deduce the emotional state of the user.

### 3.4.3 Video Player

One of the largest improvements from a preexisting feature was the video playback. This original functioning prototype was not fully automated and did not provide the best user experience because of it. There were also limited interactions with the video data which did not provide the user with optimal reductions for viewing the video playback data. These were some of the key points that were looked at when improving the video player for this iteration.

A classical evaluation method for player experience is observation [4]. This method can provide large amounts of qualitative data that can show designers context to the quantitative data represented from the other data recorded. This method can be very time consuming, yet displays a clear issue that needs to be addressed.

Inferring player experiences using facial expressions is a work presented by Tan et al. [49] which extends prior work on validating the feasibility of using automated
facial expressions analysis as a natural quantifiable method for evaluating player experiences. Their work effectively quantifies qualitative data such as joy, anger, surprise and neutral so it can be quickly and efficiently analyzed. The results show an average of all participant’s above listed emotions for two different games. The two games were Draw My Thing and Portal 2.

Understanding how to integrate the results from recording user research data and visualizing it in the game world has been done in different methods such as this example of Mirza-Babaei et al. [34] work, where the team unifies both quantitative and qualitative data together in one representation within the game world. The work shows results of the data being a series of colored line renderings of player positions in the game world with the color representing the player’s arousal state at any given position along the line.

To overcome this problem, our tool enables the user’s attached camera at the beginning of the testing session and record all input from the camera device connected to the computer. The video recording is completed when the testing session is finished and is then saved as a Unity compatible file type. This file is saved alongside the other quantitative data recordings within an appropriately indexed folder to organize each user’s data.

To go through the video recording, the Observe mode will need to be on so that the no-clip camera is enabled. Clicking on any of the nodes on the path allows you to go to that point in the play-through for the video and audio of that path. Since the video and audio media are separate, whichever of the media was recorded will be played even if the other is missing. The media are separate because it allows for better control of what information is being recorded. This can be appreciated for two reasons, first is the opportunity to reduce storage usage if one or the other is not necessary, two is to comply with any data privacy that may be in place.

Once the data is saved and organized on a per user basis, the videos are ready to be visualized in-game. This is done by having a small viewport in the corner of the game screen where the game developer can see the user’s experience recorded from the computer’s camera. To effectively skim through multiple user’s video recordings, clicking on the line renderings will bring the developer to the exact point in time that correlates to that video recording. This is effective because the quantitative data may show unusual player behaviors and this could be a point of interest to review in the video recordings.

The approach to recording the user’s face provides a number of benefits. Firstly, the facial recording does not interrupt the player performing the test [49]. The player
does not need to be asked for their opinion during the test since the observation can provide information regarding the player’s opinion. This method also has the advantage of interpreting basic expressions which are automatic because they occur whether the player wants them to or not [11].

This feature provides information based on the user’s experience during the time playing the game. This information is gathered using a webcam and microphone connected to the computer the user is playing on. Typically, this can be found through the integrated webcam and microphone built into most laptops or as an external Universal Serial Bus (USB) webcam and microphone. The scripts involved includes a call to an external process that records footage from the webcam and microphone and saves it to an ogg file inside the user’s data folder. This information is then seen through a video window which can be manipulated based on the line renderings in the Unity editor.

Understanding the user’s experience is the core to this tool and this feature provides the qualitative data to determine how the user experienced your game by observing their facial expressions body language. An affordable method for GUR is the observation method which is often accompanied by recording the user’s session for additional attempts to observe the session in case initial observation missed important events that may have needed to be noted. This can also offer as the initial observation if there is no way to reach the user playing the game (i.e., sending the player a build of the game and not scheduling a playtest with them when they play it).

3.4.4 Physical Distance From Screen

Developers prefer to see qualitative data that would reflect the player’s level of engagement playing a game by measuring their physical distance away from the screen with a depth camera such as the one built into the Microsoft Kinect 2.0 [5].

This feature provides information based on the user’s physical distance away from the screen. This information is gathered using the Kinect V2 being placed at the base of the screen the user is playing the game on. The script involved includes a call to an external process that gathers the user’s lean value and saves it to a txt file inside the user’s data folder. This information is seen on the orientation arrows as the width property, wider being closer to the screen and narrower being farther from the screen.

How far away the user is from the screen provides qualitative data measures which can be quantified into a series of numbers to be visualized in the game world.
It is important to know this distance as it explains the user’s approximate level of engagement towards certain portions of the game. If a user is closer to the screen (wider orientation arrow), then the user is more engaged with the game at this point in their playthrough and if the user is farther away from the screen (narrower orientation arrow), then the user is less engaged with the game at this point in their playthrough.

3.5 Version 3.0

The final iteration of Vixen before the studies that are presented in the following chapters of this thesis was completed with the same development team from the previous version of Vixen. This version of Vixen was funded for development by the Brilliant Entrepreneurship Incubator Program at UOIT with the Firely grant. The grant encouraged us to pursue the business perspective of this work with the goal to take Vixen to market. Parallel to this goal, the team continued to create new features built off of feedback from the previous iteration. Version control was integrated into Vixen to provide a more accessible tool for the end user. Documentation was also written at this stage to provide the user with a potentially improved experience when integrating this work into their own games. Lastly, this version looked at some final cleaning and optimizations that could be made to Vixen with the aim to further improve run time and user experience.

3.5.1 Custom Event Tracking

Following previous work completed by Medler et al. [26] and Kim et al. [18], tracking custom in-game events can provide useful information to designers for how to improve their game. However, both of these works used more traditional methods of representing the events with graphs.

The proposed method for this in this section is to have these in-game events be represented in the game world as 2D billboarded sprites with custom icons and colors chosen by the developers. This can give the designers more geographical awareness of where these events took place. This approach to visualizing in-game events has been done before in multiple different ways such as [19, 25, 42, 15]. This is my take on the topic and to the best of my knowledge, has not been done similar to this method. Examples of how these in-game events look can be seen in the Figures 3.9, 3.11, 3.12, and 3.13.

Events will display in Observe mode and currently can be shown as icons. To show the given icon the event name needs to be the same name as the recorded name as seen in Figure 3.8. For example, to record an event such as the player in the space
shooter destroying an enemy, one would call the event occurred function. The single line of code is as followed:

```java
VIX_EventManager.eventOccurred("Scored", playerPosition.transform.position, gameController.name);
```

This is recording each time a player kills an enemy and is displayed as green explanation marks in Figure 3.9. The in-game event visuals can be customized as needed with custom colors and icons chosen by the user.
Figure 3.12: Top-Down Twin Stick Shooter in Unity (August 2016)

Figure 3.13: Side Scrolling Shooter in Unity (August 2016)
3.6 Summary

This chapter explored the development of three major iterations of Vixen aimed at improving the communication of information to the user and creating an intuitive design for the tool’s systems. A key goal for Vixen is for a designer, programmer, and user researcher to use the same tool to identify any potential improvements that could be made to the game. Unity packages and extensions have been a popular method for rapidly developing an iteration of a game.

The development of Vixen began with an initial prototype from a summer research grant focusing on the proof of concept. This developed proof of concept was transitioned into the Incubator program to receive new prototype features and a concept of bringing Vixen to market. The next stage of development went to the GUR course where the goal of this stage was to round out the features of Vixen and begin evaluating the work. The last stage of Vixen’s development was the Brilliant Entrepreneurship Incubator program where the current features were polished and the concept of commercialization for Vixen was explored once again.

So far I have introduced multiple iterations of Vixen and discussed the internal development for each resulting prototype. The next chapter focuses on evaluating the recent prototypes to gather advantages and disadvantages for each. The ultimate goal is to highlight important criteria for communicating GUR results based on different discipline requirements. These criteria will be valuable for developing future prototypes of Vixen and other GUR tools in development.
Chapter 4

Study 1 - Evaluation Of Gameplay
Data Visualization

4.1 Introduction

In the previous chapter we discussed the development of three major iterations of a tool called Vixen. This chapter aims to evaluate the recent iterations through conducting ten interviews with professionals in the games industry who have seen Vixen. The interviews helped toward the iterative design and improvement of Vixen. The previous chapter discussed how Vixen was iteratively developed and prototyped through various teams and influences. This chapter describes the recent versions as a basis for a discussion during the interviews. The goal of the interviews is to better understand the communication and reporting needs for game developers and how well Vixen addresses some of those concerns.

This chapter discusses the method of conducting the interviews depicting a two phase process of conducting the interviews and then a workshop following to allow for users to integrate Vixen into their current projects. The details of the second phase will be found in Chapter 5. Following the method of this study, the results of the data was broken into key points that were divided into categories for better understanding the best approach to improve Vixen. This chapter continues with a description regarding how the results are relevant to the various roles within the game development industry. ¹

The evaluations were split into two phases, the first phase consists of the visualization evaluation and the second phase is the implementation evaluation. The first phase describes the semi-structured interview design and implementation. The semi-

¹This chapter is published at FDG2017.
structured interviews were conducted to evaluate the visuals presented by Vixen. The second phase is the implementation of Vixen into two games workflows. This process is explained in more detail in Chapter 5.

The goal of this research is to assess the needs game developers have for Games User Research in an academic setting. The tool has been developed to aim at providing developers with cost-effective and easy-to-understand ways of collecting and analyzing data on people playing video games. This research advances previous work in two key areas. First it explores current approaches and needs of game designers to develop mixed methods that improve effectiveness and efficiency of qualitative and quantitative data collection and analysis. Second, it advances meaningful visualizations of player experience analysis, bringing together qualitative and quantitative data.

4.2 Phase 1: Goals and Approach

Ten participants were recruited to participate in the interview evaluations. These participants were a variety of experienced game developers. The participants were required to have an undergraduate degree in a relevant field of game development. All of the participants in this study have graduated from UOIT’s Game Development and Entrepreneurship undergraduate program (reference Table 4.1). In person sessions occurred between 9AM - 5PM, Monday - Friday in the Game Science Lab and online
sessions were conducted on Skype in the Game Science Lab. The Game Science Lab allows for only one participant to be interviewed at a time. The overview of the procedure for each participant with this evaluation is:

- Welcoming the participant
- Session brief and consent form
- Demographics questions
- Introduction interview questions
- Demonstration of tool
- Semi-structured interview questions
- Debrief

Before starting the session, participants received the study brief and the informed consent form to read and sign. Participants were free to participate in the study or leave anytime during the study. If agreed to continue, participants were asked preliminary questions to gain an understanding of their background in game development. The demographics questions used to depict participant’s experience in the game development industry include:

- What genre of games have you developed?
- How long have you been developing games?
- What is your title as a game developer?
- What game engines have you developed games with?
- What is the largest team size you’ve developed a game with?
- What findings do you care about?

The responses to these questions can be found in Table 4.1. From what can be seen in this Table 4.1, there is a wide variety of roles that can be found in the game development industry that are relevant to Vixen. Each of these participants has a minimum of six years of experience to ensure that participants have adequate experience for quality feedback. Each participant has a wide range of game genres
Table 4.1: Participant Demographics for Visualization Evaluation

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Title</strong></td>
<td>Game Engines Programmer</td>
<td>Game Designer</td>
<td>Games User Researcher</td>
<td>Game Designer</td>
<td>Tools Programmer</td>
<td>Producer</td>
<td>Game Programmer</td>
<td>Game Programmer</td>
<td>Game Engines Programmer</td>
<td>Tools Programmer</td>
</tr>
<tr>
<td><strong>Years of Experience</strong></td>
<td>7</td>
<td>6</td>
<td>6</td>
<td>11</td>
<td>7</td>
<td>6</td>
<td>8</td>
<td>6</td>
<td>6</td>
<td>10</td>
</tr>
<tr>
<td><strong>Genre of Games</strong></td>
<td>Action</td>
<td>MOBA, Side Scroller</td>
<td>Platformers, Brawlers, Action</td>
<td>Serious Games</td>
<td>Shooter, Puzzle, Adventure</td>
<td>Puzzle, Shooter, Serious Games</td>
<td>Action, Platformers, Serious</td>
<td>Virtual Reality</td>
<td>Shooters</td>
<td>Racing, Platformers, Shooters</td>
</tr>
<tr>
<td><strong>Game Engines</strong></td>
<td>Unity, Game Maker</td>
<td>Unity</td>
<td>Unity, Unreal</td>
<td>Unity</td>
<td>Unity</td>
<td>Unity, Flash</td>
<td>Unity, Flash</td>
<td>Unity</td>
<td>Unity</td>
<td>Unity, Game Maker</td>
</tr>
<tr>
<td><strong>Team Size</strong></td>
<td>6</td>
<td>9</td>
<td>26</td>
<td>7</td>
<td>10</td>
<td>5</td>
<td>7</td>
<td>6</td>
<td>10</td>
<td>7</td>
</tr>
</tbody>
</table>
Table 4.2: Participant User Experience Demographics for Visualization Evaluation

<table>
<thead>
<tr>
<th>Type of Testing Performed</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modular</td>
<td>Interface Usability, Subjective Satisfication</td>
<td>Player Engagement, Immersion, Enjoyment</td>
<td>Evaluate Fun, Find Bugs</td>
<td>Usability</td>
<td>FTUE</td>
<td>Find the Fun</td>
<td>Find Bugs</td>
<td>Difficulty Scaling</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Observation</td>
<td>Questionnaires</td>
<td>Questionnaires, Semi-Structured Interview, Telemetry</td>
<td>Surveys (Custom, Standardized), Interview, Biometrics (GSR,EEG), Telemetry</td>
<td>Observation</td>
<td>Observation, Questionnaire</td>
<td>Observation, Questionnaire, Interview, Screen Capture</td>
<td>Observation</td>
<td>Observation, Questionnaire, Analytics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time Spent Testing (hours)</td>
<td>2</td>
<td>24</td>
<td>24</td>
<td>&gt;100</td>
<td>&lt;1</td>
<td>8</td>
<td>&lt;1</td>
<td>16</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>Result Presentation Method</td>
<td>Statistical Driven Discussion</td>
<td>Statistical Presentations</td>
<td>Detailed Report, Presentation</td>
<td>Discussion, Graphs and Figures</td>
<td>Casual Conversation</td>
<td>Notes Driven Discussion</td>
<td>Not Presented</td>
<td>Written List of Changes</td>
<td>Casual Conversation</td>
<td>Observed Summarized Numbers</td>
</tr>
<tr>
<td>Received positively</td>
<td>Well, Data Was Summarized</td>
<td>Ineffective for student game, effective for industry game</td>
<td>Difficulty to agree on next steps</td>
<td>Devs effectively found bugs</td>
<td>Well received by devs, not business recipients</td>
<td>N/A</td>
<td>Really effective, mechanics changed for the better</td>
<td>Yes, because it was mostly numbers presented</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
they have developed for, which provides a well rounded perspective when reviewing Vixen. All participants have strong experience with Unity and have stated that it is their game engine of choice. The participants have varying experience with working in different sized teams ranging from 5 to 26 different team members.

Participants were then asked a series of preliminary questions to gain an understanding of their experience with games user research and performing tests on their games. Participants will be asked specific questions such as:

- Have you ever performed any testing on a game you’ve developed?
- If yes, what was the goal of the testing?
- What methods did you perform while testing?
- How many hours did you spend designing and performing the tests for your game?
- How were the results of the testing presented?
- How effective were the results articulated to the team?

The responses to these questions can be found in Table 4.2. From Table 4.2, it can be seen that the participants have experienced a breadth of types of testing they have performed. However, the goals of these tests prove to be similar as the participants often search for solutions to improve the overall user experience. Observation is the most popular method used by the participants with 7 participants stating that is their primary method for collecting data from their evaluations. The participants however are often unable to allocate much time to the testing as 7 participants did not spend 24 hours or more on the overall testing design and implementation. When presenting the results of their evaluation, participants had varying degrees of experience as some participants such as P7 did not present the findings and others such as P2 and P3 have created presentations to showcase their findings. The team size results state participants mentioning that the team size assisting with the process since most teams were small. It was not necessary to create large presentations compared to having a casual conversation with the team.

After those questions, participants reviewed videos and pictures of Vixen’s features for both data collection and visualization. Figure 3.7 is a screenshot from the demonstration video that included voice over descriptions for implementing Vixen into a project and viewing the data. Other screenshots participants reviewed appear
in Figure 3.13, 3.11, and 3.12. The screenshots for Figure 3.13 and 3.12 also had short demonstration videos with no voice over descriptions. Following this, we entered a discussion with the participant regarding their opinion on the tool. If the participant struggled with engaging in the discussion, we used these questions to promote discussion:

- What do you think Vixen is showing you?
- How does Vixen differ from your previous methods of performing testing?
- Are there any concerns with the way the data is represented?
- Can you see any changes that need to be made in the level design?
- If this was in your game, how could this help your current or previous projects?
- What would they change or add to better suit your game?
- What limitations can you see with this tool?

The axial coding of these questions into significant categories can be seen in Figure 4.2. The participants were debriefed once all of the above questions have been asked. The debrief portion of the interview allowed participants to ask any questions or provide final thoughts on Vixen.

4.3 Results

The ten rounds of interviews resulted in 126 individual comments which were condensed into 6 categories. The interview results suggest Vixen has advantages and possible improvements to be implemented in future work. Suggested advantages and possible areas for improvement for Vixen are as follows:

- Summary of results at a glance
- Similarity to heatmaps
- Locations of level design issues
- Facilitates the discussion
- Fast and simple to setup into any project
- Modularity
All interviewees had developed games for at least 6 years and all largest team sizes mentioned in the interviews describe the interviewees to be working within independent studios averaging a team size of 6 (exception to P3). Every interviewee has worked with Unity in the past to assist with developing or prototyping games. The findings that participants cared about varied based on their discipline and previous experiences. To summarize findings that participants cared about were user experience, sales, and game metrics.

When discussing interviewees experience with GUR and testing games, the results show that each participant has performed some form of testing on at least one of their games they have developed. The range of quality between all the participants stretched from internal QA to full studies with consistent, prepared procedures that required proper analysis of data and presentation of that data to upper management. Only three participants (P2, P3, P4) mentioned performing high quality, well thought out with consistent execution of testing on their games.

4.3.1 Summary of Results at a Glance
After reviewing the videos and screenshots that showed the visual results from the playtest sessions, each participant agreed that the data showcases a summary of
results at a glance for each level. The following comments from interviewees display their understanding of the visuals that are present in the media they observed during the interview. P5 stated "Vixen shows where the player has moved their character, where they're facing, their arousal level, face and audio". Participants felt that it was clear and easy to identify the types of data from a glance and felt comfortable with explaining they see a summary of the data. This has been further verified as P2 mentions "Vixen shows webcam, orientation, and position data". P3 also stated that Vixen shows "Quantitative data of player playthrough, paths, position, orientation". Lastly, P10 mentions that player position, rotations, and events are shown.

Participants had mentioned general comments about the overview of data being shown by Vixen. P9 describes Vixen as a visual summary of what players have done in your game. P6 described Vixen in a similar manner as they stated that Vixen is a "visual representation of analytics". P4 added "A lot of position and event tracking over a game session" and then followed up with "Visuals are aggregating data". The participants were receptive to the idea that Vixen provides a summary of playtesting data from a quick glance. P3 agreed with this idea as they stated that what they see is a "Pure quantitative summary of data". As P2 observes the data Vixen displays in-game, they stated that there are "patterns and similarities in data". This is complemented with P8's comment as they stated that Vixen "Shows where player's go in each play session". P7 approves of the overall data presentation as they stated that Vixen "Encompasses every aspect of a playtesting session". Lastly, it was stated by P7 that Vixen "Shows actions users are taking over time in each play session accumulated". The interviewees share a similar opinion that Vixen can effectively create a summary of results at a glance.

Interviewees provided additional comments towards their opinions on the visual aspect of Vixen. For example, P1 mentioned "Colour usage is clever". Similar statements were mentioned by other participants such as P5 when they stated "Biometrics in 3D is impressive". However, the coloring was not clear to all interviewees immediately as P2 mentioned that they misunderstood what the colors represent. The interviewee did come to an understanding of the meaning behind the coloring without assistance shortly after further visuals were presented to the participant. As for the arrows representing the orientation of a game object in the game world, P3 mentioned that the "Orientation is nice for seeing which direction they're facing". Lastly, a general comment towards the benefits of including arousal levels via heart rate made by P5 mentions that this could be great for many games, even including
puzzle games. Overall, the participants of the interviews were very receptive to the overall visual summary that Vixen provides.

4.3.2 Similarity to Heatmaps

Specific participants (P1, P3, P4, P5, P6, P9, P10), mention that the method of displaying the data has similarity to heatmaps which provides a form of comfort when analyzing the data since they are familiar with interpreting heatmap results. Each participant had their own interpretation of how similar Vixen is to heatmaps as P1 stated "Vixen is showing me flow." and P4 agrees with P1 as they stated "Interesting seeing flow in level design". However, P3 stated "This looks like a quick way to have a heatmap". P4 specifically stated "What I'm looking at is a heatmap". Certain participants really preferred this approach to visualizing player trajectories as P6 stated "The player map is awesome as it tells us exactly where the players went".

Interviewees did have their suggestions on how to improve the player trajectories. P7 mentions that the visuals can be very rigid when logged at larger time intervals and suggests the implementation of spline curves to round out the larger logged time intervals. P6 mentions that the custom events could be improved by clustering the close event icons into one numbered event for a more clear visual that depicts simple to read intensity values of certain in-game events.

Participants also found some problems with the visuals when comparing to more standard implementations of heatmapping. For example, P3 states "What I see is a pile of colorful spaghetti" when looking at all playthroughs at the same time. However, participants such as P10 stated "The lines are very helpful". P9 also finds the line renderings helpful as they stated "This shows more popular zones in the level".

4.3.3 Locations of Level Design Issues

The majority of participants could analyze the visuals and pinpoint specific locations of level design issues with only as little as seeing one participants playthrough. P3 commented in the interview "Vixen could help with understanding areas used in level" so I will explore this in more detail. P1, P4, P6, P10 describe that there is unused space within the games from the screenshots in Chapter 3. Without having seen the game before, P1 describes that the level shown in Figure 3.11 has large portions of unused map space. This participant’s hypothesis to this issue is that the camera may have an influence since the game appears to be isometric from the design of the level. There is a mutual agreement to P1’s statement as P6 states "There is a big space
of the level that is not being utilized" when pointing at the same level. However, the same participant can visually interpret the design intention as they stated that Vixen "Shows level design and game functionality". P4 references Figure 3.13 as they stated "Space in level not being fully utilized". Additionally, P10 felt there were instances of different levels that could be improved as they stated "The road in the car game could be better. The tanks game was tight and had limited navigation" and "The map design for the potato launcher promoted camping and thats no fun".

P3 stated "Level design issues can be quickly visualized" but P6 took the next step in analyze and proposed a hypothesis towards the unusual player behavior as they specifically stated "There could be issues with controls based on player movement". Another in depth look at the data had P9 mentioning "The outputted data shows a games over simplicity". This shows relevance to not only level design, but to user experience as well. Participants such as P6 and P9 have made comments that describe concerns of the player’s experience from the data they viewed. One participant in particular, P4 commented on the design for strategies as they stated "Level design may not provide opportune strategy". Experienced game designer participants took the visualizations presented by Vixen and explored the deeper meaning behind them.

The remaining participants mention that they would benefit from seeing more playthroughs of data before creating conclusions. P4 and P6 both mention the need for additional perspectives (filters) of the data to come to a more confident conclusion. These participants understand the importance of having a large sample size before coming to a decisive answer to potential level design issues. P5 specifically stated "I need to use the tool and work on the game to better locate level design issues". Having access to the tool and becoming familiar with how Vixen works will be explored in the second phase, study two.

Vixen can also verify if the design goals have been achieved. P2 stated "I can see that level design goals were fulfilled thanks to the visual data representation which shows the possibility of using Vixen to ensure the original design intentions were fulfilled. This argument can be found in P8’s interview as this interviewee stated "All the level designs look good". Each participant’s background experience has an effect on their opinion towards level design. Lastly, these potential level design issues can prompt discussion. This can be seen in P7’s comment as they proposed the question "What is deterring people from going under the spaceship?". Facilitating discussion will be looked at in more detail in the next section.
4.3.4 Facilitates the Discussion

Our interviewees mentioned they were from unique disciplines in game development, yet each one felt that Vixen was providing useful information that facilitates the discussion with peers and supervisors. P1, P2, P3, and P8 all mention the challenges with appropriately articulating the results of their findings to their supervisors. After reviewing Vixen’s visualizations, each one of these interviewees explained how these results can simplify the articulation of the data and promote discussion with internal development team members. As mentioned by P1 "I’m unsure of players getting stuck", and later mentioned that it would require a discussion with the game designers to understand if that is the intended method for navigation. A similar instance occurred with P8, where the interviewee asked "What caught the attention of the player?". Both participants mentioned that there is more to discuss when looking at the data and can be used when starting a discussion with other game designers.

There are opportunities for discussion to be facilitated when attempting to understand the story players go through. P6 understands this when they stated that Vixen "tells the story of the player within the context of the level". This can be especially important to discuss when developing a serious game as P4 explained "Position and what players are looking at is important for serious games to know if they are looking at the right places and learning the correct information". P8 also understood the importance of this information as they stated that Vixen "Shows what is interesting in the game or where the players are lost". Vixen also promotes the discussion of when it is time to retrieve more playtest data as P3 stated "Would like more participants to come to better conclusions".

If there was ever a discussion towards where players travel in an open world game, Vixen can assist with providing evidence to fuel the discussion. This is described by P4 as they stated "Great for nonlinear experiences to see where players go". It can help to have additional sources of data to verify points in a discussion and P5 understands this as they stated "Biometrics reinforce video recording output." These additional sources of data could potentially cause the tool to become cumbersome to setup but as it turns out, interviewees felt the opposite.

4.3.5 Fast and Simple Setup

P5 has published successful tools for Unity’s Asset Store and notes the importance of having a tool that is fast and simple to setup into any project because game development teams have purchased your work to save them time and development resources. Other interviewees (P2, P3, P4, P5, P9, P10) mention after viewing
the setup video for Vixen, they were pleased to see it requires minimal drag and
drops to have the tool prepared for recording and only one toggle switch to view the
recordings. P9 had stated "It can help make the testing phase much more smoother"
and then later followed up with "Feel confident leaving the room while testing".

Building on these comments, P3 stated that Vixen is a "Good tool for people
that don't have much time to build or understand user research methods". For those
who may be less experienced with user research, the implementation process is as P4
describes it "Intuitive to setup and use", or how P5 mentions "Setup looks easy, it
has drag and drop capabilities. It's easy to add new icons for the custom events". P7
explains part of the reason for Vixen's simplicity when they stated "Only a single
prefab needed for setup". This statement is similar to the one P3 mentioned as
they stated "Tool would be loadable very easily" when describing the process for
implementing Vixen into their own project. Not only did P2 find the visual telemetry
unique, they also stated "Visualized telemetry is unique and simple to setup".

One of the suggested benefits Vixen aims to achieve is to be a GUR tool that does
not require users to setup other tools to gather data. This is mentioned by P10 in the
interview when they stated "It records their face and speech with integration into the
genius". The interviewee mentioned this because in their previous experience, they
would need to use other software such as Open Broadcast Software to track webcam
and audio data. This tool automates that setup by having an integrated webcam and
audio recorder that automatically sets up the playback with additional interactive
options for filtering the data.

4.3.6 Modularity

As mentioned by P3 "Nice within Unity, well integrated into developer workflow",
participants felt that Vixen is not only fast and simple to setup, but also offers
modularity to accommodate for individual developer workflow. All participants had
cconcern about the modularity of Vixen as they each thought about their own games
and how Vixen would compliment their needs. There are additional uses for Vixen,
P7 stated that Vixen is "Pretty general purpose for Unity".

P1, P6, and P10 for example mention the desire for this tool as a useful software
for debugging purposes. These participants understand the alternative applications
Vixen has available. Vixen can assist with debugging for instances such as evaluating
AI behaviours or analyzing weapon projectile trajectories. During a round of inter-
views, P6 felt that the visuals are similar to a popular independent game as they
stated "Watch replays over again and watch multiple replays. 'Like super meat boy'".
The visuals Vixen renders has the potential to be used as graphical features in-game as well as a user researcher tool.

Additionally, P1 also discusses the topic of biases when conducting a playtest session. P1 stated "There is less observation bias because Vixen is non intrusive". Having a moderator standing over a player is a bad, yet common practice with inexperienced user researchers. P3 agrees in the interview as they stated "The process needs to be non intrusive".

P2, P3, P4 and P5 enjoyed the modularity to have complete customization of in-game event visuals and multiple ways to customize the player trajectories to better suit their game's needs. P4 had this to say on this topic "Custom events and colors are very modular". The same participant also felt it was important to mention "The tool appears modular for adding new components" incase the current systems in Vixen do not fully meet the requirements for a particular game's evaluation methods. P5 specifically stated "It's easy to add new icons for the custom events". Specifically, P2 felt that it was helpful to move around in the game world when viewing the data as they stated "Data looks raw because of linear interpolation, It is good to be able to fly around the world and look at data from multiple angles".

To summarize, P9 described Vixen in two words "Unified solution". Participants felt that Vixen is a unified tool because of the options and flexibility that are available.

4.4 Discussion

This tool has been developed with independent game developers as well as students developing at an academic institution in mind as we expect that these teams will have fewer members and may not have been working together long. New teams may not have dedicated a games user research group to evaluate their games being developed. To accommodate for this, the tool can be quickly implemented by any member of the team as it does not require much programming knowledge to implement. The tool can also remove the challenge of conducting test sessions and going through gameplay data as the developers can build an executable of their game with the tool built into the game and send if off to players. This saves time for developers as they do not need to run test sessions and they can still retrieve the data from players for later evaluation using the server upload and download features. If the team does not have access to a server, the data can still be retrieved manually from the participants.

The core mechanic of the tool, player trajectories recording, are constantly being improved to provide more accurate results for developers. From the visualization evaluation, improvements have already been made in these areas. Firstly, improving
the usability of Vixen to make it less complicated for developers to quickly integrate the tool in their development pipeline as we integrate new features. The tool is currently based on quantitative data collection as well as additional features for recording qualitative data. For example, the two features to record the user’s voice through microphone and their facial expressions via webcam. As developers review the playtesting data, these features enable further analysis of the emotional state of the user during the playtest. Qualitative data that would reflect the player’s level of engagement playing a game may provide helpful information to developers. The methods evaluated in this study were the webcam and audio as well as the biometrics. At the time the videos and screenshots were created, the physical distance from the screen feature was not accounted for due to the limited access to the Kinect camera. This feature could enhance understanding the physiological data visualized in the game world alongside the other qualitative data.

Once the Space Shooter game recordings were analyzed in the game, it was apparent that including tracking which locations on the screen the player’s character is looking at was not helpful as the character could not rotate. However, the position data showed interesting results as some players for an extended period of time, decided to not move up or down on the screen. This would indicate that the player requires a tutorial at the beginning of the game to learn that those are navigation options available to them. The style of visualization for the line renderings was most effective when viewed from a birds eye view as this was the view the camera had when the player was interacting with the game. This can be seen in Figure 3.9.

One of the biggest challenges with creating a tool to accommodate many games is ensuring a consistent result of data collection and visualization. For the Survival Shooter, when the in-game character dies, the game deletes the instance of the character and instantiates a new one for the next playthrough. This causes the link from the tool to the character to become broken. To address this specific issue, we adjusted the way Vixen links to character objects so they are able to record data even if the object is instantiated mid-game. However, there will be a need for each team to make slight adjustments to Vixen to ensure the tool appropriately meshes with their own game based on how they have created each of their own systems.

The immediate next step is to expand on the initial evaluations and invite game developers to use the tool in their development and participate in formal evaluation sessions. This will help us fine-tune information about the specifics of the tool (such as user interface (UI)), and data representation. This study can also provide needed insights on any usability issues.
4.5 Summary

The current state of this visualization tool is a complete Unity 3D plugin to help game developers and games user researchers to improve their usertesting process. The result from the visualization evaluation study with professional game developers suggests that the tool could support improvements being made with games through optimizing various aspects such as level designs or usability issues. For example, improved level designs will be able to aid game developers in creating a more enjoyable experience for their players. This tool has the potential to save developers time and resources in their production cycle through more intuitive data representation methods and simpler data collection styles.

The major contribution described in this chapter to the field of GUR, game development, and computer science is a tool that can provide developers with a new form of interactive visualization of playtesting data by providing a 3D interactive environment for game developers to analyze and explore playtesting data.

The next chapter will explain and display the results from phase 2 of Vixen’s evaluations. The second study, implementation evaluation, will explore the experience game developers have integrating Vixen into their own project and using Vixen for conducting playtest sessions.
Chapter 5

Study 2 - Evaluation Of The Tool’s Implementation

5.1 Introduction

The previous chapter focused on evaluating the latest versions of Vixen through conducting ten interviews with game developers who have worked for an academic institution. The interviews provided valuable insight into the needs and requirements for communicating GUR results for multidisciplinary teams. The interviews also provided a critique of key features and necessary improvements that were utilized when developing this next iteration of Vixen.

This chapter explores the refinements that were applied to Vixen that are influenced by the interview results described in the previous chapter. This iteration of Vixen utilizes different games that are developed by the participants in phase two of Vixen’s evaluation, the implementation evaluation. The participants will conduct user testing on their own game with Vixen implemented into their workflow and document their experience with Vixen. The results from the user tests will provide visualized data similar to what was seen in figures from Chapter 3. The participants were provided with time to implement changes to their games. Once the user testing was completed, participants completed a semi-structured interview similar to the one completed in Chapter 4. The chapter concludes with the results of the study focusing on how effective Vixen’s latest iteration communicated the results of each participant’s user testing sessions.
5.2 Continued Development of Vixen

This section will describe the development of Vixen’s latest iteration, highlighting important comments from the initial interviews that influenced the design. After completing the interviews, I developed eight key topics which encompassed 126 transcribed comments. After this development, the results were transferred into a spreadsheet that had the comments color coded for each key topic. This can be seen in Figure 4.2 from the previous chapter. Chapter 4 focused on the results of the interviews to better understand the overall visual presentation and workflow integration of Vixen across multiple different games with varying genres. By refining the comments into key points, it will be easier to highlight what worked and what did not. The following sections describe the results of Study 1, specifically what key features need to remain untouched and the necessary improvements mentioned from participants to influence the continued development of Vixen.

5.2.1 Key Features

The initial setup of Vixen was recognized by participants as a simple process for integration with a game made in Unity. This is an important feature that has been accurately achieved based on the results of Study 1 and therefore changes have not been made for the second study.

Participants mentioned that Vixen can provide developers with a variety of customizations to better suit their personal games and needs from the tool. As this was received well by participants in Study 1, this key feature will continue to receive improvements as there are always additional options for filtering data and providing manipulation options for the interface that developers interact with in Unity.

One of the most notable features within Vixen is the in-game 3D visualizations. During the interviews for Study 1, participants had mixed reviews on how the data was represented as the comments changed based on the quantity of playtesting sessions that were presented at a given time. When it came to viewing a single participant’s playthrough at a time, the interviewees were split on what they could achieve knowing this data as many described that one playthrough is not enough data for justified changes to be made. Others however described that one playthrough provides the developer with a storyline of a participant’s experience throughout the level. When looking at multiple playthroughs in one image, participants would continue to have mixed opinions towards this scenario because the data can be perceived as too "raw" or "noisy" where others perceived the visuals as a similarity to heatmaps. With mixed
feedback, it becomes increasingly difficult to create a solution that accommodates for each concern from the participants. In this case, the in-game visualizations have been left for now until further data has been collected in Study 2.

Having all of the necessary components to perform user-testing without the need for preparing multiple software was mentioned on multiple occasions as a feature that is appreciated by participants. For any further development, it will be critical for Vixen to continue future developments with this in mind as that has become a key feature of the tool.

5.2.2 Necessary Improvements

A consistent comment described by participants was the inherit time Vixen spent importing the playtesting data into Unity. After investigating the problem, it was clear that the cause for this problem was the video recordings data. This was discovered by reviewing the file sizes of the data being imported and the observation data was very large and caused importing to be far too long. To improve upon this, the tool will store the data outside the Unity assets folder to a folder close to the root harddrive. There will be a variable in the inspector window within Unity to allow for user’s to define where the data can be saved. This is important to have to divide the playtest data between potentially multiple games on the same computer. If this was not setup, user’s would see playtest sessions form different games within one game world. This can cause severe confusion as one game may have a character that can fly and the other only have a character that can walk, which would have drastically different visualizations of data. The data could also clip through geometry since in was not recorded from the game that it is being displayed in. Overall, the data storage had received necessary improvements to provide improved usability for the user.

Participants often mentioned that the amount of scripts required to perform some of the tasks within Vixen were too much and needed to be compressed to improve the usability of the features the tool had to offer. As cleaning up the interface that user’s interact with is an ongoing process, there were improvements that could be made here. When logging the player trajectories and their respective orientation, these tasks were handled separately with the idea of modularity in mind. The goal with having separate scripts handling these tasks is that the features the user wanted to be tracked would be attached as a component to the main Vixen game object. However, participants agree that this is not the ideal method since the main Vixen game object becomes cluttered when these scripts can be simply combined and use
a series of checkboxes to achieve similar level of modularity within Unity’s inspector window. This could lead to new challenges participants may face such as having difficulty finding where to toggle features since the visuals have been reduced from a script in the inspector down to a checkbox on a script within the inspector.

Another concern that participants described was the amount of information that is presented at once to the user attempting to analyze the data Vixen is visualizing. Some participants stated specifically that Vixen suffers from cognitive overload. It was evident that in both the visuals presented by Vixen as well as the GUI that users would use to work with the tool were both prone to cognitive overload. Reducing the cognitive overload that participants experience can be challenging to overcome for some reasons. First, the amount of information a user is comfortable can be subjective to each individual’s retention for information. Second, the data could be too reduced to accommodate for cognitive overload and important playtest data may be missed when interpreting the visuals. The approach taken to address the participants concerns regarding cognitive overload was to provide users with additional filtering options within the hierarchy window for the playtesting data. As for the GUI cognitive overload, the compression of scripts that hold the GUI elements will potentially aid users in quickly reading the interface and easily remember where the GUI elements are and what systems they interact with.

The interview results showed that participants found the limited platform support to be a concern. Participants frequently asked if this tool was available for Unreal or as a standalone to be compatible with in-house game engines. This has been seen as more of a future step than an immediate improvement that can be made. The other issue with platform support is that Vixen will only work with certain versions of Windows when using the heart rate monitor. Similarly, Windows has been the only platform Vixen has been evaluated on so the tool is not guaranteed to have similar results when used on Mac OS or Linux. Furthermore, Vixen is only compatible with Unity 5.4 and above because of the function calls used within the tool. This was improved slightly to have Vixen support Unity 5.0 and above versions. Other implementations of Vixen would be outside the scope of this thesis as it would require major rewrites of the tool to complete.

5.3 Phase 2: Goals and Approach

The second study aimed to demonstrate the effectiveness of Vixen by implementing Vixen into different games and documenting the changes made to the level design and game design by the developers of these games over a fixed period of time. For
this study, we recruited three game development teams to participate in this study. I recruited these participants at UOIT offering them Vixen for free as compensation for participating in the study. The participants were selected based on their experience, the genre of the current game in development, the game engine they are using, and the platform they are developing for.

Once a participant is recruited, they were orientated with the process for the study and signed the necessary documentation. The participants received Vixen as a Unity package and tasked with setting up Vixen into their existing game. The participants had us available for 1 hour for assistance with setting up Vixen. We only provided assistance upon request. During this time, we observed the developers setting up Vixen with their existing project.

After the 1 hour orientation was complete, each participant was responsible for completing playtests with Vixen gathering playtesting data. This must be completed within two weeks of receiving the orientation. Each participant then had two additional weeks to implement any changes they want to their game based on the data represented by Vixen alone. During the study, participants were not allowed to use other analytics tools or GUR methods to test their game. During these four weeks, participants were required to create diary entries discussing their experiences with using Vixen within their project.

After the participants completed the study, each participant was required to meet with me to discuss their experience with Vixen. This will take place in the form of a similar interview to Study 1 from Chapter 4 where we had questions to encourage discussion but the developers were open to discuss anything related to their experience with Vixen. Some of these questions included:

- How does Vixen differ from your previous methods of performing testing?
- Are there any concerns with the way the data is represented?
- After using Vixen, did you see any changes that need to be made in the level design?
- What limitations can you see with Vixen?
- Were you aware of the server and biometrics features?
- If there was a feature you were aware of but did not use, could you explain why?

After the focus group session is completed, the participants submitted their diary entries as well as a copy of their most current build of their game (and playtesting
data). At this point, we reviewed the games, diary entries, and feedback from the focus group discussion.

5.4 Results

After conducting the interviews with the participants, each one was transcribed similarly to the first study. Following this, the diary entries were collected along with the screenshots of the participants games with Vixen implemented. The results were broken down into three sections: Diary Entry results, Screenshots, and Interview results.

5.4.1 Diary Entries

Participants documented their experiences from the beginning of the study until they were interviewed. Each participant was free to document any experience they had related to Vixen. If participants struggled with writing notes regarding their experience, they were provided with this list of questions:

- What were the steps you took to setup Vixen?
- How did Vixen integrate into your workflow?
- How did you setup/conduct your user test sessions?
- What data did you analyze and how did you analyze the data?
- What changes did you make after iterating on the game?

Once the study was completed, these are the diary entries received from P1:

P1

- All files for the package should be under Toolbelt Tech. - All Vixen files should be in Toolbelt Tech/VIXEN/ to denote company’s product.
  - Importing package without upgrading == 4 error messages in Unity 5.6.1 having to do with the WWW class and movie textures. - These have been marked as obsolete and/or are deprecated. - I usually avoid auto-upgrades because Unity will often replace the old code with the easiest solution (and often the slowest), like with GetComponent calls in Update.
  - After running the API updater, Unity shows 21 warning messages. Some are obviously for variables to be used later, but others are legitimate warnings about old code. - Either way, you can use disable warnings in the code to prevent Unity from spitting these out.
First thing I did after a full, successful import was to look for the documentation. I found it under Toolbelt Tech/VIXEN/ as VixenDoc.pdf. This is good, it’s right where I expected it to be. You may want to consider renaming it to have an underscore at the start ("_VixenDoc.pdf") so it ends up at the top of the hierarchy. Not a big deal though.

A problem I found with the documentation: I’d prefer a small introduction as to what the package does. I think you can do more with "Tools for visualization of player experience". First thing after that I want to see is a quick-start guide. The simplest procedure to start using the tool (hopefully a drag and drop with a prefab?). Then go into the details. And follow that with the specific variables/classes/components and more information about them.

While looking for a quick-start prefab, I saw two prefabs folders. The one outside the Toolbelt Tech folder contained a particle system for confetti. Side note, the shader should be using alpha, not additive for this particle type (lel).

Couldn’t find a prefab, so I opted to check if there were sample scenes with a basic implementation to help me get started. I loaded up SampleProject.unity first. It defaulted to running in Observe mode. After stopping the scene, I saw the class had a default directory log of C:/Vixen, which I checked and found empty. Switched to Record Session mode and re-checked the folder and saw the data there now. Check documentation -> agent and saw which meant what. Switched back to Observe mode to see what it would look like with the no-clip camera.

Ready to import into my 2D game now. I repackaged everything with the "rogue" folders pushed into Toolbelt Tech and imported that.

Trying ExampleScene-Unity5_0 in my game project, I notice lots of errors at runtime. Also missing a script on that prefab (VIXAgent MK6 1). Opting to use the prefab used in SampleProject instead. Wait, no. I’ll use the MK7 because it looks like a more updated version.

Dragged prefab into scene of my game (there is only one scene in the game with a procedurally generated level that increases in size over time as the player moves through it). Glorious purple screen. Can’t see anything. After the first play and stop, the purple screen is gone.

Trying to log a jump event for player using "VIX-Agent.eventOccurred()", but it doesn’t exist? Noticing some minor grammatical errors in the docs, like "eventOccurred" on page 7. Taking a break. Will ask Brandon on what to do later.
- Intellisense must have been bugged or slow to catch up. – Nevermind, I was looking at a different page which had either VIX_EventManager (pg 4) or VIX_Agent (pg 7) as the class to call the static function from.

- Logging the jump event as a test from code now using "VIX_EventManager.eventOccurred("Jump", transform.position);", but I’m getting the wrong positions in Observe mode. – I see in the editor that "Jump" is a known event, and "print(eventsDict.ContainsKey("Jump"));" logs true, sooo... what’s up? It’s using the wrong icon and color, too.

- I found that some of the markers were hidden behind an object of mine that was rendered in a special way. It’s also the same reason why I was getting a purple screen - because the object was reddish and was mixing with the blue default background.

- Maybe consider changing the render order of all marker sprites so they’re always in front in Observe mode?

- I’m noticing that Vixen is logging the wrong values for the Z axis. When I do a transform.position print right after logging the event to the manager, I see what value should have been used (always printing 0, this is a 2D game), but while the x and y values are correct in the Vixen log, the Z value is going up and I don’t know why.

  - The markers are still at the wrong positions on all axes (x, y, and z).

**P2**

The results from P1 expressed details regarding confusion towards certain features within Vixen and explains their thought process towards how they overcome the problem. Diary entries were also collected from P2, these are the results from P2:

- It was easy and simple to setup and didn’t require a lot of time.

- I was not able to test out all of the features of Vixen such as collecting heart rate and the user’s physical distance away from the game since I did not have access to the Kinect or a heart rate sensor. However, I was still able to receive useful information from VIXEN when playtesting the game. The game I used was a platformer utilizing the leap motion for controls, where the player is jumping/rolling away from a shadow while collecting purple orbs to increase their score. The information I collected through VIXEN allowed me to tell if certain collectables where missed along the way meaning that the location of collectables should perhaps be changed and locations at which players died meaning level design should be reevaluated for balancing the difficulty. I was also able to note if the user’s got stuck trying to jump over obstacles. In addition, through the facial recording video, I learned when players where having difficulty performing gestures with the leap motion where I felt if I had
access to a heart rate sensor, that would provide even more information on stressful locations that need to be evaluated.

- Through the audio recording, I gathered any comments that were said by players which was useful on informing me on their level of engagement or issue they came across. The information gathers is important where it goes towards improving the gameplay and technical issues encountered by the game.

- I was not sure how to set additional events through VIXEN so I only implemented the collectables events which was sufficient. Perhaps for future additions, I feel depending on the game it might be useful to collect the time the player remained in a location which would help identify any issues as to why the player got stuck or was not able to move.

- For some reason, the icons (exclaim icon for events) did not appear on the screen but were instantiated correctly. It had something to do with the sprite renderer. When the component properties where copied form the arrow component and pasted on to the collectable’s icon, It showed up correctly.

- Opencv_ffmpeg2410.dll did not load for some reason but didn’t seem to cause any issue with recording.

5.4.2 Screenshots

Each participant was responsible for creating screenshots to show their game with the tool implemented. Ideally, participants would present multiple screenshots that would depict before and after for any level design changes that Vixen was the justification for. The results from P1 are found in Figure 5.1 and 5.2. The results from P2 are found in Figures 5.3, 5.4, 5.5, 5.6, and 5.7.

5.4.3 Interviews

The results from each participant’s interview are summarized into tables. Table 5.1 summarizes the results of the interview that asks participants about their experience with game development. Table 5.2 summarizes the results of the interview that asks participants about their experience with user experience. Tables 5.3 and 5.4 summarize each participant’s answers for the questions relating to their experience with Vixen.
Figure 5.1: P1’s 2D Game Similar To Flappy Bird
Figure 5.2: P1’s Game With Vixen Implemented

Figure 5.3: P2’s Game With Vixen’s Interface
Figure 5.4: P2's 3D Adventure Game With Vixen Implemented

Figure 5.5: P2’s Game With Another Participant
Figure 5.6: P2's Game With A Different Set Of Playtesting Data

Figure 5.7: P2's Game With More Participant Data Visualized
Table 5.1: Participant Demographics for Visualization Evaluation

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Tools Programmer</td>
<td>Game Designer</td>
<td>Game Programmer</td>
</tr>
<tr>
<td>Years of Experience</td>
<td>7</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>Genre of Games</td>
<td>Shooter, Puzzle</td>
<td>Serious, Platformer, Vr</td>
<td>Platformers, Shooters, Racing</td>
</tr>
<tr>
<td>Game Engines</td>
<td>Unity</td>
<td>Unity</td>
<td>Unity, Unreal</td>
</tr>
<tr>
<td>Team Size</td>
<td>10</td>
<td>6</td>
<td>7</td>
</tr>
</tbody>
</table>

Table 5.2: Participant User Experience Demographics for Visualization Evaluation

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of Testing Performed</td>
<td>Internal QA</td>
<td>Usability</td>
<td>Internal QA</td>
</tr>
<tr>
<td>Goal of the Testing</td>
<td>Evaluate Fun, Find Bugs</td>
<td>Usability, Fun and Engagement</td>
<td>Flow</td>
</tr>
<tr>
<td>Methods Performed</td>
<td>Observation</td>
<td>Questionnaires, Focus Groups</td>
<td>Observations, Think-aloud</td>
</tr>
<tr>
<td>Time Spent Testing (hours)</td>
<td>&lt;1 hour</td>
<td>8 hours</td>
<td>&lt;1 hour</td>
</tr>
<tr>
<td>Result Presentation Method</td>
<td>Casual Conversation</td>
<td>Report and Questionnaire graphs</td>
<td>Not Presented</td>
</tr>
<tr>
<td>Presentation Articulation</td>
<td>Devs could effective find bugs</td>
<td>Could be more effective if session was recorded</td>
<td>N/A</td>
</tr>
</tbody>
</table>
Table 5.3: Participant Answers

<table>
<thead>
<tr>
<th>Questions</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is Vixen showing</td>
<td>Icons for where the player jumped. This helped understand where changes needed to be. The icons were rendered behind another layer.</td>
<td>Position, orientation, facial recording, audio, length of the game, level progression</td>
<td>Lines, custom events, orientation but it was not needed, only there to prevent crashing</td>
</tr>
<tr>
<td>Difference from Previous Testing</td>
<td>Could see visually where players would go in the world.</td>
<td>Has a lot of tools all in one, speeds up playtesting process. Able to analyze everything at a glance. Depicts areas that need to be addressed. Can see webcam and audio at each timeframe. Shared visualisation of results. Synced data helps with understanding results. Can overlay the data from multiple sessions. It was clear that there were issues with level geometry and could be quickly fixed. Players die easily and it provides information for improving gradual difficulty. Study focuses on level design and asset placement.</td>
<td>Have an actual log of the game, previous would just be watching the player play. Vixen captures and saves it whereas we would just try to remember it.</td>
</tr>
<tr>
<td>Concerns With Data Representation</td>
<td>Events were logged correctly but did not match the recorded data and that was very frustrating. Vixen’s oiberver was conflicting with your camera. Could not see the camera because there was no webcam. Could improve workflow of Vixen by having everything under one folder. There is also no quick start documentation.</td>
<td>Issue with events not visually displaying in the game world at first but was easily fixed by fixing the components. It was unclear how to add another event.</td>
<td>The way death was handled, there were many sharp lines that were not needed. It was not easy to filter the play testing sessions so it made the results difficult to read. There were no timestamps for the events so it was difficult to tell when events happened.</td>
</tr>
<tr>
<td>Questions</td>
<td>P1</td>
<td>P2</td>
<td>P3</td>
</tr>
<tr>
<td>----------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Level Design Changes</strong></td>
<td>Level design changes did not get made because the gameplay was mostly finalized. Would make changes if there were more participants in the study.</td>
<td>Changing placement of assets, scaling assets, additional obstacles. Had concerns with potential random level generation.</td>
<td>Yes the level design changes became very obvious with Vixen. Checkpoints needed to be rearranged based on Vixen results.</td>
</tr>
<tr>
<td><strong>Previous Project</strong></td>
<td>Logging events, where players travel in the world. Would have been great to see player progression and back and forth moments. Would use on other projects.</td>
<td>Speed up time to track and analysis data. How to see player experience. Show results to team easily.</td>
<td>Logging player trajectories is valuable to any game where the player moves a character around. Not very helpful in a point and click menu based game.</td>
</tr>
<tr>
<td><strong>Change or Add To Vixen</strong></td>
<td>Simplify startup. More developed interface, not just components and instead a separate window.</td>
<td>Simulate gameplay in real time to see where enemies are. Did not know multiple objects could be tracked. Eye tracker to see UI used with a heatmap.</td>
<td>Display playtests menu. Timestamps for events. Filter data based on timestamps. Some sort of icon that moves along the path. Tooltips are needed for everything because it took a long time to know what things interacted with. Vixen is overly complex for simple tasks.</td>
</tr>
<tr>
<td><strong>Limitations</strong></td>
<td>Startup, interface, draw order, have an option to render based on depth or on the very top.</td>
<td>Hardware limitations. Screen recording is not available.</td>
<td>Same as changes or additions. Plus better error handling.</td>
</tr>
<tr>
<td><strong>Feature Awareness</strong></td>
<td>Was aware of and chose not to use Server because participants are local. Biometrics and Kinect were there and not needed for this specific game’s study design + didn’t have hardware.</td>
<td>Aware of Server but does not have a server. Biometrics and Kinect were there and not needed for this specific game’s study design + didn’t have hardware.</td>
<td>Was aware but didn’t need a server. Was aware but didn’t need Biometrics or Kinect Didn’t have the tools to record for it.</td>
</tr>
</tbody>
</table>
5.5 Discussion

Once the second study was completed and the results were compiled, there are key points that are important to discuss. Participants in this study were required to setup Vixen, which involved a workflow integration. After that, user test sessions were conducted by the participants. The user test sessions collected information which caused participants to analyze the data. After this point, participants would iterate on the games they developed. These are the key points that are important to discuss.

For participants, setting up Vixen into the project had unique challenges due to the nature of having unique games. Overall, once the participants had completed the setup task, each participant found the task to be achievable but sometimes unnecessarily challenging. For example, the documentation was not designed in such a manner as to provide participants with the fastest and simplest setup procedures for the core functionality of the tool. Bot P1 and P3 mention that the documentation would have greatly benefited from having an easy guide for quickly working with the basics of the tool, before going through the detailed documentation of all available features within Vixen.

After participants completed the setup and looked through the documentation, the next challenge that was encountered was developing a workflow integration into their games. This became a difficult challenge for each participant as each new feature they attempted to integrate from Vixen’s caused new problems. Starting with the player trajectories, P1 mentioned experiencing difficulties attempting to render lines because of the nature of the features already developed in their 2D games as the game implemented advanced graphical effects that caused the line renders to not be visible.

When it came time to perform user test sessions, one of the major challenges participants encountered did not directly come from working with Vixen as each participant found the task of recruiting participants very difficult. This however is relevant to Vixen as it revealed a gap in the current work presented in this thesis. Vixen currently does not have a feature to assist with recruitment of participants. The closest feature integrated into Vixen that can aid with this challenge users had was the Server Upload/Download feature as it provides users with the opportunity to recruit participants remotely. This would not directly gather participants for the user but will open the study the user is conducting to more participants than available in a local space.
During the testing sessions, users mention how they were responsible for distributing and collecting consent forms from participants as well as designing and conducting questionnaires. This is not something that Vixen supports the user with completing. Consent forms are an important step in the process of conducting a study, and questionnaires are commonly used within a user testing session. Vixen could assist with this portion of the process for conducting user testing sessions by providing templates and integrating both directly into the tool within Unity. Designing and distribution would become more simple and faster since it will be integrated into the same tool that is currently being used to handle the remaining portions of the process for conducting user testing sessions.

The user testing session was reported by users as simple to conduct since it only required running the game when the participant was ready to play and stopping the game when the test was completed or the participant was finished playing the game.

For P2 and P3, analyzing the data presented by Vixen within the Observe mode was a completable task. P1 showed no visuals because of a graphical bug within the game code. However, P1 used the text files that the data was stored within to analyze the data from the play test sessions. This was able to provide P1 with results as they were logging in-game events and wanted to know how many times an event was triggered as well as what the timestamp was for each event. As this was not an ideal method for analyzing the data Vixen presents, this was still an effective method for P1 to come to conclusions regarding the level design of the game.

Specifically for P3, the analysis of data was a challenging task to complete due to an issue with the game systems in place. The game systems caused there to be many unnecessary line renderings which clouded the visuals within the Observe mode. This would only happen when the in-game character died which allowed P3 to still be able to analyze the data and come to conclusions regarding the level design within the game.

As the data analysis is completed, participants were tasked with iterating on their games using the information provided by Vixen’s recordings. However, participants did not implement any changes to their games. Each participant discussed having a list of some possible changes to improve their games based on the visuals, which none of the participants chose to follow up on. This could have been caused by not having enough time to complete the task or possibly from not understanding the proper way to correctly implement the changes.

Another possible cause for this might have been participants not having these issues prioritized or proposed with possible suggestions for changes. These are cur-
rently not features that Vixen is capable of completing but it could be potentially in the distant future. If Vixen can provide the user with a list of potential fixes and prioritize them in a list, this might assist user’s more with following through with a critical step to the process.

5.6 Summary

The second study was discussed in this chapter. This also touched on the improvements that were implemented based on the results from the first study. These improvements provided a better experience for the participants in the second study. There were three participants, where one participant was also apart of the first study. For the second study, data from diary entries, screenshots, and interviews were collected from the participants. This chapter’s discussion touches on setup, workflow integration, user testing sessions, data analysis, and iterations. The next chapter is a discussion regarding the work presented in this thesis, future work that can be completed to further improve Vixen, and conclusions regarding the tool.
Chapter 6

Discussion, Conclusion, and Future Work

6.1 Summary

This thesis showcased visualization techniques that can assist with the facilitation of understanding and exploring the relationships among playtesting data. For example, combining data from players in-game movement with observation results and biometric data. However, this is not straightforward because the underlying data is often in different formats. Another challenge is to make these visualizations simple enough to provide a quick overview for producers, but also detailed enough to be usable and practical for gameplay programmers and game designers. Although various visualization techniques have already been introduced in this domain, most of these techniques focus on displaying large amounts of quantitative telemetry data without integrating qualitative or contextual data on players’ experience. Moreover, most of the current visualizations are static representations of playtesting data so they would not be customizable based on the users need (game producer, game programmer, game designer). Hence, there is a need for an interactive visualization tool that can adjust data representation based on the level of details required from different members of a development team. This thesis reported on the development efforts on a tool called Vixen that assists collection and provides dynamic and interactive representation of playtesting data. This thesis reported on two initial studies to evaluate the effectiveness of the tool with game developers to guide our future development.

This final chapter provides the summaries of discussions and contributions of each study and prototype development. This chapter revisits the research questions and
considers the contributions across the thesis. Finally, future work in the field is described and suggested.

6.1.1 Vixen Iterations

Initial designs of Vixen and the iterations were developed to create the current development build. Through exploring related GUR and analytics tools, the goal was to have a better understanding of how to effectively visualize data within the tool, Vixen. Additionally, by exploring the game development process in an academic institution, Vixen aimed to gain a better understanding of developing a positive user experience for software design tools.

The tool Vixen aimed to save time by automatically collecting important data regarding your user’s experience through various qualitative and quantitative data using mixed methods. A summary of the features in this tool include: recording player position, orientation, heart rate, user’s distance from screen, video of user, and audio. All of these features are presented to the developers with the goal of utilizing simple to understand methods. This recorded information is transferable from participant to developer over a server.

The work presented went through five versions with each focusing on iterating core features such as player position and observation data while developing new features such as in-game events. The first iteration built off of the previous work completed by Mirza-Babaei [33] to prove the concept could be possible in the Unity 3D game engine. The following iteration aimed to further improve the tool by including additional qualitative methods such as facial observations and audio recordings. The next iteration introduced new visualized data along with modified biometrics. The fourth iteration incorporated custom event based tracking of telemetry data as well as optimizations to the subsystems in place. The final iteration of Vixen builds off of the feedback provided in the first study conducted in this thesis, the visualization evaluation.

6.1.2 Study One

This study discussed the method for conducting the interviews depicting a two phase process. Following the method of this study, the results of the data were broken into key points that were divided into categories for better understanding the best approach to improve Vixen. This study continued with a description regarding how the results are relevant to the various roles within the game development industry.
The evaluations were split into two phases, the first phase consists of the visualization evaluation and the second phase is the implementation evaluation. The first phase describes the semi-structured interview design and implementation. The semi-structured interviews were conducted to evaluate the visuals presented by Vixen. The second phase is the implementation of Vixen into two games workflows. This second phase is explored in Study 2 of the thesis.

The goal of this research for Study 1 was to assess the needs game developers have for GUR in an academic setting. The tool was developed with the goal of potentially providing developers with cost-effective and easy-to-understand ways of collecting and analyzing data on people playing video games. This research advances previous work in two key areas. The first is to explore current approaches and needs of game designers to develop mixed methods that improve effectiveness and efficiency of qualitative and quantitative data collection and analysis. The second key area is to advance meaningful visualizations of player experience analysis, bringing together qualitative and quantitative data.

The ten rounds of interviews resulted in 126 individual comments which were condensed into 6 categories. The interview results suggest Vixen has advantages and possible improvements to be implemented in future work. When discussing interviewees experience with GUR and testing games, the results show that each participant has performed some form of testing on at least one of their games they have developed. The range of quality between all the participants stretched from internal QA to full studies with consistent, prepared procedures that required proper analysis of data and presentation of that data to upper management.

This study’s major contribution to the field of games user research, game development, and computer science is a tool that can provide developers with a new form of interactive visualization of playtesting data by providing a 3D interactive environment for game developers to analyze and explore playtesting data that fits the need of a variety of roles in game development such as a game programmer, producer, or game designer.

6.1.3 Study Two

This study explores the refinements that were applied to Vixen that are influenced by the interview results from the first study. This iteration of Vixen utilized different games that are developed by the participants in phase two of Vixen’s evaluation, the implementation evaluation. The participants conducted user testing on their own game with Vixen implemented into their workflow and document their experience
with Vixen. The results from the user tests provided visualized data similar to what was seen in figures from Chapter 3. The participants were provided with time to implement changes to their games if they felt it was necessary after exploring the data presented by Vixen. Once the user testing has been completed, participants completed a semi-structured interview to engage in a discussion with their experience integrating Vixen into their workflow and iterating on their game because of Vixen. The study concludes with the results focusing on the effectiveness Vixen’s latest iteration.

This study showed that certain systems of Vixen were faster and easier to integrate than others. Components such as player trajectory with orientation and in-game events were used heavily and with little confusion for implementation and understanding the results. Other components such as the biometrics and physical distance away from the screen were not explored by participants because of the hardware requirements they demand. These results showcased the key features of Vixen that provide meaningful insights for game developers. The results also describe the features that were underutilized by participants.

6.2 Thesis Discussion

This thesis began with exploration of design and evaluation methods covered in Chapter 2 where we discuss the previous work completed in the industry that created the inspiration for Vixen. Chapter 3 discussed the initial development of the iterations Vixen went through. Chapter 4 evaluates the iterations of Vixen though interviews with professional developers, who have developed games in an academic institution, resulting in eight categories that inspired the development of the latest iteration of Vixen. Chapter 5 begins with the development of the latest iteration and concludes with the evaluation where the results indicate that Vixen is potentially effective at communicating the results of a playtest. The final chapter in this thesis explores the suggested future development that can be implemented to Vixen.

This tool has been developed with independent game developers as well as students developing at an academic institution in mind as there is a degree of expectation that these teams will have fewer members and may not have been working together for extended periods of time. New teams may not have dedicated a GUR group to evaluate their games being developed. To accommodate for this, the tool, Vixen, can be quickly implemented by any member of the team as it does not require much programming knowledge to implement. The tool can also remove the challenge of conducting test sessions and going through gameplay data as the developers can build an executable of their game with the tool built into the game and send it.
to players. This saves time for developers as they do not need to run test sessions and they can still retrieve the data from players for later evaluation using the server upload and download features. If the team does not have access to a server, the data can still be retrieved manually from the participants.

This chapter explored multiple research methods that are often applied when studying interactions and player behavior across the gameplay experience. This chapter also explored multiple tools that are designed to gather and represent data relating to video games. A key area to understand is the effective methods of measuring the player experience.

This thesis showed an idea of the types of data that can be visualized to assist with interpretation of video game data and enhance the player experience. An effective tool for visualizing both qualitative and quantitative data would take the best aspects of work presented in this thesis and apply it to one tool. Some of the most common aspects would include using line renderings to visualize player location in the actual game world as detailed in Chapter 2. Another visualization for quantitative data would be for larger data sets and that is to heat map the data. When looking at unique in-game metrics, line graphs will easily visualize that information for the researcher.

The previous work presented in Chapter 2 has a couple limitations that need to be mentioned. First, the tools only work for one game and they are not dynamic for all or even most games. Another limitation is that the tools do not do a good job visualizing the user’s facial expression.

The current state of this GUR tool is a framework to design and develop new and innovative features to improve the current processes for completing user testing. There are already improvements being made with games through optimizing game developer’s level designs. These improved level designs will be able to aid game developers in creating a more enjoyable user experience for their players. This tool aimed to save developers time in their production cycle through more intuitive data representation methods and simpler data collection styles.

The core mechanic of the tool, player trajectories recording, are constantly being improved to provide more accurate results for developers. From the technical evaluation, there have already been improvements implemented in these areas. There is also work being done towards improving the usability of the tool to make it less complicated for developers to quickly integrate the tool in their development pipeline as new features are integrated. The tool is currently based on quantitative data collection as well as additional features for recording qualitative data. For example, the two
features to record the user’s voice through microphone and their facial expressions via webcam. As developers review the playtesting data, these features enable further analysis of the emotional state of the user during the playtest. Developers prefer to see qualitative data that would reflect the player’s level of engagement playing a game by measuring their physical distance away from the screen with a depth camera such as the one built into the Microsoft Kinect 2.0 [5]. This feature could enhance with understanding the physiological data as it is a different approach to gathering qualitative data from the player which can be easily quantified and visualized as an additional layer over the current visualizations Vixen has in place.

When exploring the space shooter game recordings that were analyzed by participants in Study 1, it was apparent that including tracking where the player’s character is looking at was not helpful as the character could not rotate. However, the position data showed interesting results as some players for an extended period of time decided to not move up or down on the screen. This would indicate that the player may require a tutorial at the beginning of the game to learn that those are navigation options available to them. The style of visualization for the line renderings was most effective when viewed from a birds eye view as this was the view the camera had when the player was interacting with the game. This can be seen in Figure 3.9.

One of the biggest challenges with creating a tool to accommodate many games is ensuring a consistent result of data collection and visualization. For the survival shooter game presented to developers in Study 1, when the in-game character dies, the game deletes the instance of the character and instantiates a new one for the next playthrough. This causes the link from the tool to the character to become broken. To address this specific issue, adjustments were made to the way Vixen links to character objects so they are able to record data even if the object is instantiated mid-game. However, there will be a need for each team to make slight adjustments to Vixen to ensure the tool appropriately meshes with their own game based on how they have created each of their own systems. This is just one example of cases that can be corrected with some minor programming knowledge.

Another example of this occurring is in Study 2, where a participant developed a side scrolling platformer and the system for death teleported the character back to the checkpoint. This did not break the recordings or visualizations, however it would show character trajectories that did not happen as the line rendering would link from the death position to the respawn position. These additional unnecessary line renderings create a less than desirable scene to explore when attempting to find problems in the level design within the game. A problem like this can be cleaned
up by allowing the participant to modify the rendering scripts to only render if the change in position does not exceed a certain threshold.

The immediate next step would be to expand on the evaluations and invite additional game developers to use the tool in their development and participate in formal evaluation sessions. This will assist with fine-tuning information about the specifics of the tool (such as UI), and data representation. This type of future work can also provide needed insights on any additional usability issues that have not already been mentioned through the results of the previous studies conducted.

6.3 Thesis Contributions

The main contributions of this thesis are: 1) A Unity plugin tool that will make 3D spatial visualization more accessible to indies (including academic developers) in comparison with tools/research that are often only accessible for AAA-developers, 2) A facilitation of both data collection/visualization, hence optimizing the user-test process, 3) A combination of both quantitative/qualitative data overlaying the game-world, with real-time filtering. The interactive 3D aspect helps developers to better explore the data and identify particular areas to explore further with qualitative data. The following section outlines each specific contribution:

6.3.1 A Unity Plugin Tool That Will Make 3D Spatial Visualisation More Accessible To Indies

After researching the available products and services for GUR on the Unity Asset Store and interviewing professionals who develop games for academic institutions, it was clear that there is a gap in the needs to be filled. With this approach, Vixen will fulfill the needs of game developers with limited resources to be able to incorporate GUR into their daily workflow.

6.3.2 A Facilitation Of Both Data Collection / Visualisation

Combining the data collection process with the visualization results allows for the overall process to be optimized since all aspects needed to perform GUR are combined into one tool. Having both aspects together allows for minimal friction when integrating into a developer's Unity based workflow. Bringing a GUR tool to independent developers will provide opportunities that were not previously available and potentially give them an advantage against the competition when taking their games to market.
6.3.3 A Combination Of Both Quantitative / Qualitative Data Overlaying The Game-World, With Real-Time Filtering

Various visualization techniques aim to provide a better understanding of player experience, however most of these techniques are based on players’ behavior or actions, and do not include players feeling resulting from those actions. By using a mixture of qualitative and quantitative approaches, Vixen is designed to use both groups to motivate scientific discussion and bring the field forward.

6.4 Limitations and Future Work

The next steps for this work would be to further integrate the user testing process into the tool. An example of this would be integrating a demographics survey into the beginning segment of each user testing session. Moreover, once a session is completed, a questionnaire system can be also integrated into the tool that the participant can fill in without diverting to another system. These improvements will allow for less portions of the process to be completed outside of the tool allowing for the developers to have additional time saved so those resources could be spent elsewhere [1].

6.4.1 Observation Data

One of the recording and visualization options is analyzing certain facial gestures, more specifically the player’s smiles or frown expressions. Previous research showed a possibility to infer player experiences using facial expressions [49]. As such, players smiling could be associated with them having a positive experience in a game. This can help provide useful information to developers as they now have strong evidence towards which portions of the game are creating a positive experience and can be kept in the game. The current version of Vixen records facial observation, and requires manual analyze. However, it is possible to analyze these recordings using smile detection algorithms. For example, Huang and Fuh [16] introduced a real-time, accurate, and robust smile detection system with the FG-NET face database with 88.5% detection rate for their methodologies. There are more reactions than smiles that players can experience during a gameplay session. Tan et al. [49] describe joy, anger, surprise and neutral as other possible expressions that can be interpreted for quantifiable data.

Another potential expansion to Vixen’s current implementation of facial observation would be to incorporate the in-game custom event based system to assist with visualizing the quantified data collected from the smile detection algorithms. This
step may increase simplicity of interpreting key points of interest in the video recordings since the data is similar to existing systems currently within Vixen.

Based on the information gathered in Study 1, the participants did not have many comments regarding the observation data. This might have been the case because many of the graphics used in Study 1 did not include facial observation view ports while observing the data. Many participants' comments were very passive regarding the facial observation data. Participants mention the facial observation data is present but lacked critical feedback on how to improve the facial observation system within Vixen.

As for Study 2, participants expressed many concerns with the setup regarding facial observation data. Firstly, participants did not find it clear how to begin recording data from the built in webcam. This caused the participants to not want to record the data and move onto the other recording options available within Vixen. To improve this system, the webcam recording should be more error proof. Many of these problems that arose were from errors that did not provide the user with adequate information on how to simply correct the mistakes. For example, information needed to be provided to inform the player on how to check if another software is currently using the computer's webcam. Providing the user with improved documentation that goes through common problems may enhance the overall user experience when setting up the facial observation system within Vixen. It is critical that features do not get underutilized as important information regarding the playtest session may be permanently lost.

### 6.4.2 Biometric Data

The possibility to have captured physiological data from players to interpret their emotional states during a gameplay session had several iterations over the course of developing Vixen. GSR could be recorded to measure a player's arousal state [23]. The data captured from a GSR recording device has been integrated in our tool, as currently the movement data is represented by rendered lines that show the path of the player, each line could be given a specific color based on the recorded GSR value [33]. These colors could range between green (lower level of arousal) and red (higher level of arousal) to provide a clear visual interpretation of the player's emotional state throughout their gameplay. They can also be set by the user of Vixen to personalize the experience to best suit their needs. Changes in GSR values may provide clues as to changes in players' emotional state at any given time during gameplay. Given that a low or high GSR can only indicate a possible change in emotions, it is important
to correlate this data with information presented by other data to more accurately estimate the emotional state of the user. Currently, a working feature for Vixen logs a timestamp for both the position and GSR readings, followed by syncing the timestamps together so the two sets of data line up.

However, as participants had access to all features of Vixen during Study 2, none of the participants logged any biometrics data and reported that the data was not accessible due to hardware limitations. This discovery has highlighted the importance of using accessible devices for measuring biometrics. A potential new hardware that Vixen can be compatible with for biometric data would be the Myo gesture control armband. This gesture controlled armband is superior to previous hardware used with Vixen in a number of perspectives. Firstly, the device is much less intrusive for the user as it is simply wrapped around the forearm. The Myo has well documented SDK’s and community support when developing video games. The Myo accurately logs EMG data, which is a measurement of muscle activity through electrodes on the skin. This kind of biometric recording has been used for emotion detection in the past by Cacioppo et al. [7].

Eye tracking is another feature that can be included into Vixen’s workflow. Eye tracking can further improve the exact areas in the game world that were observed by the player, but will require additional hardware [25]. Maurus et al. [25] present a novel practice for visualizing the gaze locations of a participant in a 3D world using heatmaps. This form of data representation can provide strong insight into what player’s are focusing on in the game world [25]. Eye tracking data could be a direct improvement from the previous feature of orientation arrows (See Player Trajectories Section In Chapter 3) rendered at scheduled timestamps along player trajectories.

Although, the eye tracking or GSR hardware may not be available to all developers, these forms of data could provide a new perspective into player experience that is currently difficult to capture and meaningfully represent to developers. Another possible advantage for including these additional data is to provide more filtering options for reviewing the gameplay videos, which means potentially faster and possibly more effective analysis of gathered data. For example, having a detection for smiles can allow game designers to scrub through the video to the timestamps where a smile is logged. This could potentially save time that evaluators do not have to watch the whole video anymore to find the highlights of the player’s experience [16].

88
6.4.3 Further Study into Contributions of Vixen in GUR

An experiment in which a comparison between Vixen, classical user testing methods, and no user testing methods would be a possible next step for a future study to reveal constructive differences between each approach. This experiment would build on the results presented in this thesis. Some key points for this potential future study would be:

- Have three development teams making a Unity game use Vixen with their game
- Allow a team to perform a heuristics evaluation
- Anticipating that Vixen saved time during data analysis process
- Two protocols with this study, the developers and the participants for the developers user testing sessions (this is a potential for extra data)

A more qualitative study is essential to better reveal the constructive differences of each approach. A similar study has been conducted by Mirza-Babaei\(^1\) [29]. In this study, it was found that BioSt will yield a higher quality game and a better gameplay experience. To achieve a similar result with Vixen is critical.

6.4.4 Further Study into Commercialization of Vixen

Another potential method for evaluating Vixen could be to release this work on the Unity Asset Store in order for game developers using Unity to use the tool. With this asset store, users of the store can post reviews of the products they purchase. With this experiment, the goal would be to evaluate the Unity Asset Store Post-Release Reviews that are posted in the store by users who have purchased Vixen. The key points for this potential future study would be:

- Analyze all reviews and ratings that are posted on the Unity Asset Store for Vixen
- Compare comments from developers with iterations made in Chapter 3
- Anticipating that the reviews are positive with high ratings and discuss the usability and overall experience when using Vixen

\(^1\)Chapter 6: Evaluating Biometric Storyboards p. 101
6.5 Conclusion

With the rise of new platforms such as virtual reality and constantly evolving consoles supported by increased distribution and crowdfunding services, the market is encountering fast changes and advancements in technology. Game development studios are under more pressure to create successful games due to the increase in competition and technologies. The need for GUR is becoming more important in order to make sure that the player experience is at the highest quality. GUR helps track and visualize experiences before they reach the mass market, ensuring the optimal player experience through rigorous iterative evaluation.

User researchers have been adopting HCI evaluation methods for GUR in order to provide a mixture of qualitative (observations, think aloud, interviews, focus groups, cognitive walkthroughs etc.) and quantitative (telemetry, questionnaires, physiological data) approaches to evaluate a game depending on their goal. However, finding the right mixture of approaches, delivering the user test results in a timely manner, and convincing designers on issues are some of the current challenges facing GUR.

To address this, the current state of this visualization tool is a complete Unity 3D plugin to help game developers and games user researchers to improve their usertesting process. The result from our interview study with professional game developers suggests that the tool could support improvements being made with games through optimizing various aspects such as level designs or usability issues. For example, improved level designs will be able to aid game developers in creating a more enjoyable experience for their players. This tool has the potential to save developers time and resources in their production cycle through more intuitive data representation methods and simpler data collection styles.

To summarize, the development of video games is highly complex involving several disciplines working to create an enjoyable player experience. GUR methods need to provide timely, specific and motivating reports that allow developers to make informed decisions. Vixen achieves this and as such, is the major contribution to the field of games user research and game development within this thesis. This thesis also achieved a tool that can provide developers with a new form of interactive visualization of playtesting data by providing a 3D interactive environment for game developers to analyze and explore playtesting data.
Bibliography


[45] Shewaga, R. A Comparison of Seated and Room-Scale Virtual Reality on Medical-Based Serious Games and Virtual Simulation by.


