STRUCTURING A NEW INTERACTIVE FICTION MOBILE AUTHORING TOOL TAXONOMY IN ACCORDANCE WITH EVOLVING GAME DEVELOPER PREFERENCES

Tyler B. Wright and Raymond Francis

Central Michigan University, Mt. Pleasant, Michigan

ABSTRACT

This study describes the role of mobile devices in authoring interactive fiction (IF) games past, present, and future through conducting a survey of mobile apps and the unique characteristics that differentiate them. We attempt to answer the following research question: How might mobile IF authoring tools be classified and organized in accordance with current game developer preferences? This is accomplished through a three-step research methodology. First, a literature review was conducted to establish preliminary categories of mobile IF authoring tools and the apps which define them, analyzing scholarly and industry sources. After synthesizing findings from the literature into a rudimentary categorization of mobile tools, the focus then shifts to a qualitative research protocol which investigates the mobile authoring preferences of active IF game enthusiasts belonging to the the https://intfiction.org/ online discussion forum. Twenty-five unique mobile-accessible IF authoring tools were identified across the selected discussion threads and placed within five broad categories that emerged through an inductive qualitative data analysis: a) browser; b) compiler; c) interpreter; d) mobile; and e) text editor. Findings were then synthesized into a mobile tool taxonomy responsive to both IF practitioners and researchers alike, easing the tool selection process for users at every level of experience with writing in the genre.

KEYWORDS

Interactive fiction, text adventures, authoring systems, mobile interpreters, game compilers, e-book editors, text editors

1. INTRODUCTION

Interactive fiction (IF) is a nonlinear literary genre that presents the reader with opportunities to influence the story with their decisions, effectively turning stories into playable games. IF games provide textual descriptions of the player's world, obstacles, puzzles, non-playable characters, and tasks that must be accomplished to drive the story forward [1]. Digital IF started with *Colossal Cave Adventure*, a mainframe text adventure computer game which required that the player inputs text in a parser to explore the world and solve puzzles [2], however nonlinear stories go even further back.

Text adventures have come a long way since their earliest beginnings in printed format. The 1970s *Choose Your Own Adventure* (CYOA) book series is among the earliest examples of IF [3], eventually leading to fantasy gamebooks in the 1980s and a shift into text-based computer games [4]. With each shift onto a new medium (books to gamebooks to mainframe computers to personal computers), IF has undergone a transformation from being totally mobile in book form to being confined to a stationary computer device, with a particular emphasis placed on the

massive mainframe computers that marked a significant evolutionary step for IF. With the arrival of mobile computing and devices, it is natural to conclude that this beloved genre will find its way back into the palm of readers worldwide, if it hasn't already. Does the digital mobile phenomenon also apply to the process of authoring IF?

This study describes the role of mobile devices in IF authorship past, present, and future through conducting a survey of mobile apps and the unique characteristics that differentiate them. This study attempts to answer our research question: *How might mobile interactive fiction authoring tools be classified and organized in accordance with current game developer preferences?* By following this inquiry, we might get a good sense for the current state of mobile IF authorship as well as provide a glimpse into the future, relative to the interests of current authors convening online. Our resulting tool taxonomy is responsive to both IF practitioners and researchers alike, easing the tool selection process for users at every level of experience with writing in the genre.
This study provides a three-step research methodology to establish a survey of mobile IF authoring tools past and present, with a look into possibilities for the future. First, a literature review was conducted to establish preliminary categories of mobile IF authoring tools and the apps which define them, analyzing scholarly and industry sources. After synthesizing findings from the literature into a rudimentary categorization of mobile tools, the focus then shifts to a qualitative research protocol which investigates the mobile authoring preferences of active IF game enthusiasts belonging to the https://intfiction.org/ online discussion forum. Findings were then synthesized into a final mobile tool taxonomy. Apps described in the literature and others described by our sample population were arranged accordingly.

For the purposes of this research, "mobile IF authorship" and its equivalents refers to developing IF through handheld devices such as tablets and mobile phones but stops short of the personal computer or laptop device. It is also worth stating upfront that our focus is on the practice of *writing* IF through mobile devices rather than in *reading* it—a distinction which will likely become apparent by the latter half of the study.

2. LITERATURE REVIEW

A review of literature describing IF authorship from the context of mobile devices revealed four main tool categories: a) browser-based authoring systems; b) interpreters; c) mobile authoring systems; and d) e-book editors. This section details each of these categories, according to sources belonging to scholarly journals, mobile app repositories, and company website showcases.

2.1. Browser-Based Authoring Systems

One category of mobile IF authoring tools are those which may be accessed from a browser but are not necessarily designed for use on a handheld device. Kirilloff [5] describes the challenge that comes with authoring IF with browser-based authoring tools on mobile devices: "To teach Twine in the classroom, your students will also need access to computers with internet connections. It is possible to use Twine on a mobile device, but the interface is challenging" (p. 4). The authoring solutions that are particularly favorable for browser navigation are oftentimes limited on mobile devices [6-8], strengthening a demand for mobile apps designed specifically to handle nonlinear stories. While some tools navigated by browser might be accessed by mobile device, this is usually far from ideal, with notable exceptions such as *ChoiceScript* and *Undum* notwithstanding [7]. Despite these limitations, IF authoring systems such as *Twine, inklewriter,* and *Quest* represent some of the most popular authoring solutions on the market today (see Table 1).

Table 1. Browser-based interactive fiction authoring tools.

Authoring tools	Description
Twine	*Twine* can be accessed directly in the browser, but the user must download their game and import it back into the system to resume work on a later date. Performance varies between operating systems [5].
inklewriter	Inkle Studios offers a basic, non-programmer web-based IF authoring system called *inklewriter* which may be accessed straight from the company website [9].
Quest	An alternative to the downloadable *Quest* authoring software is a browser version which allows users to share their works more easily with a community of fellow developers [10].
ChoiceScript	*ChoiceScript* is celebrated for being easy to navigate on mobile phones and outputting games that are responsive to mobile environments and players whose preference is playing games on handheld devices [7].

2.2. Interpreters

IF written in specific languages must be run through the appropriate interpreter software, resulting in their compatibility across different platforms and operating systems [11]. The way the interpreter runs IF is analogous to how a word processor reads a digital word file [1]. Games written in popular IF languages such as TADS and Inform may be run on any operating system with a corresponding interpreter [1], negating the need for games to be rewritten in another language to broaden their compatibility across platforms. Interpreters often contain graphics capabilities that display a static image during source code renderings [12], a simple interface, or one that is more stylized [13]. Mobile interpreters of IF programs vary in terms of quality, fidelity, and supported languages.

Text Fiction is an Android port of Infocom's Z-machine engine, allowing users to run a vast collection of classic text adventure games from their smartphone [14]. Another mobile text adventure interpreter is *Fabularium*, which plays an array of formats including ADRIFT, Alan 2 and 3, Hugo, Glulx, Magnetic Scrolls, and Adventure Game Toolkit (AGT) [15-16]. *Fabularium* supports Unicode, allowing the user to interpret games outside of the English language, as well as provides accessibility settings that improves the user experience for those with disabilities [16]. *Twisty* is an Android mobile IF interpreter solution and *Frotz* is for iOS mobile devices [8]. *Twisty* not only handles Z-machine adventures, but also supports more modern games written in IF languages such as Inform 7 [17]. *Frotz* is another interpreter which handles Infocom games, developed originally by Stefan Jokisch in 1995-1997 and ported to Unix by Galen Hazelwood [18].

Mobile interpreters serve a classical IF parser game experience which requires textual input from the player. Styles et al. [8] suggests the limitations of parser game development on mobile devices and offers that hypertext is the future of mobile IF game development in the following:

> *The popularity of IF on handheld devices is surprising, considering the difficulty of entering lengthy commands onto a miniature keyboard. Given that a typical hypertext fiction story requires little or no keyboard input on the part of the user, we believe hypertext fiction is an even better fit for handhelds than IF is. (pp. 2-3)*

Authors and players seeking a classic parser gameplay experience (as in the works of early computer text adventure games) have interpreters—however another category of mobile authoring tools must be envisioned which serves a userbase that prefers a more modern, hypertext gaming experience. Furthermore, interpreters are principally built to *run* source code, but not necessarily *write* in it; for that, an additional editor or word processor may be required. The

emergence of simplified mobile solutions for writing interactive branching narratives was surely inevitable.

2.3. Mobile authoring systems

A smaller grouping of authoring tools represents solutions that are specifically designed for writing in the genre without concern for the author's geographical location. Works of digital IF may be downloaded and installed on mobile phones, offering a substantial multimodality range of illustrations, sounds, games, and puzzles [19]; however, this study is not concerned with mobile-friendly outputted projects but rather is concerned with the tools employed for their development. Where browser-based authoring tools aren't always designed with mobile compatibility in mind and mobile language interpreters exist for running the source code of games authored with classical languages, mobile authoring systems are designed to serve the interests of developers who desire a complex authoring solution from the palm of their hand. Another advantage to authoring IF from mobile tools is that they consume fewer resources [2]. Below are a few notable examples of mobile IF authoring system apps that are currently on the market:

- The *Interactive Story* app is one such mobile IF authoring tool designed specifically for the author who wants to get straight to work and share their final works with an online community [20].
- The drag-and-drop navigational features of *Texture* make it a viable solution for authoring short IF games on mobile platforms which are then saved on a server [21].
- *Dorian* is a no-code IF developer mobile app with a mobile game store that allows developers to make a profit from their works, sharing any earnings with the platform [22]. The monetization scheme is such that story-routes exist behind a paywall and *Dorian* game developers are issued a portion of microtransactions, while also owning their game's intellectual property [23].
- One system, *iStory*, employs a hypertextual layout which resembles classic CYOA novels and boasts 20 built-in stories as examples [8].

This category of mobile IF authoring tool appears to command a smaller reach than browser-based authoring systems, suggesting that there is a greater demand for hybrid systems that are more favorable to desktop computer development pursuits, albeit with some mobile development capabilities. Even still, Styles et al. [8] defines the development of handheld hypertext storytelling applications as "the most exciting area of future work" (p. 5); a bold statement to conclude their study of IF. Given the shortage of true mobile IF authoring solutions, it may be safe to suggest that this option presently serves a small niche within the overall ecosystem of mobile authoring tools, much like interpreters and digital e-book editors, the latter of which is to be explored in the next and final tool categorization.

2.4. E-book editors

Another category of mobile IF authoring tools are those which aid in the development of hypertextual CYOA-styled works, or classic "gamebooks," as they are often referred. Gamebooks are stories which jump forward or backward through the pages based on choices made by the reader, resulting in a variety of story outcomes [24-25]. Mobile e-book editors and readers operate in the same manner as word processors and EPUB/PDF readers and could simply characterize the clever adaptation of generalized tools to serve the aims of nonlinear storytelling projects. Most if not all mobile word processors and reading apps provide hypertext functionality as one of their core features, making them appropriate options for authoring works in the CYOA and gamebook traditions.

Below is a non-exhaustive miscellany of tools which fall into this category of mobile IF authoring solutions:

- *Stretchtext's* hypertext editing possibilities extend digital books in a manner conducive to IF [26].
- The critical annotation features of two mobile EPUB reader apps (*ILSAditor* and *@note*) are described within the context of extending digital e-books in ways that transform them into works of IF, revealing collaborative possibilities unique to the medium [27].

This category of mobile IF authoring solution is concerned with enriching digital books in ways that involves the reader beyond what a standard linear text could provide, making them suitable for a variety of digital storytelling activities and applications (to include IF). IF development through e-books is appealing not only for e-book readers, but also for the developer seeking a low-complexity and low-cost mobile solution to nonlinear storytelling [7]. The irony of IF's mobile history is that it started in handheld book form and returned to handheld digital book form, demonstrating a clear appeal for the hypertextual e-book baked into the genre's minutiae.

3. METHODS

New digital tool channels in games development results in an evolving ecosystem of technologies which requires a robust multimodal analysis [28]. Our study begins to structure the dynamic ecosystem of mobile applications and their uses within the realm of IF authorship. Instead of simply reporting on the tools available to the developer, we attempted to categorize them in a manner which clearly differentiates them in accordance with key characteristics that emerge through an inductive qualitative data analysis of documented use.

The target population for this study are IF enthusiasts belonging to an active online community. The public https://intfiction.org/ online discussion forum provided a window into the preferences of real-world IF practitioners, exhibiting varied levels of experience and familiarity with mobile IF authoring tools.

Inclusion criteria accepted discussion threads published from any year, providing a substantial range of interactions between authors describing IF mobile tools. A loose forum search criteria combining words "mobile," "author," and their close synonyms was employed to isolate threads relating to the scope of this study. Each discussion thread from 2013-onward was reviewed manually, and any threads which specifically engaged in conversation evaluating mobile authoring tools were included in our analysis.

Six forum threads on https://intfiction.org/ were selected for this qualitative analysis (see Table 2). Substantive threads were isolated, and a data extraction method was applied which only made light omissions for personal identifiers, duplicate data, and comments of an "off topic" nature (i.e.: not pertaining to the topic under review). Raw qualitative data were extracted and added into a loose digital text (TXT) file until all discussion threads were thoroughly data-mined, approximating to 7,000 words.

Care was taken to omit discussions concerned with running or publishing mobile games as, while they might initially appear germane, are describing very different phenomena. Included data sources were those which focused exclusively on developing IF within mobile environments, but discussions where the focus is on playing IF games on mobile devices were excluded. The difference between *reading* versus *writing* IF emerged frequently within our data extraction

protocol, and this level of dissonance was even felt by forum contributors as the lines often blurred, requiring some level of periodic clarification to refocus discussions.

Table 2. Discussion threads selected for review.

Year	Discussion Thread
2013	Android Interpreters: State of the Art in 2013?
2014	Mobile Development Environment
2014	mobile developement [sic] system
2016	Interactive Fiction Editor for Mobiles!!!
2016	Using mobile devices to create IF
2020	Mobile IF makers

The resulting dataset was organized and readied for a thematic qualitative data analysis, resulting in a holistic and adaptable tool taxonomy. Each of the tools described in this paper's literature review were investigated within a lens produced by study findings and placed within this new framework, also incorporating tools mentioned in discussions but falling outside of the available literature. As a result, mobile tools described within a body of literature were applied within an improved tool taxonomy responsive to actual insights offered by real-world IF practitioners.

4. RESULTS

Twenty-five unique mobile-accessible IF authoring tools were identified across the selected discussion threads (see Table 3) and placed within five broad categories that emerged through an inductive qualitative data analysis: a) browser; b) compiler; c) interpreter; d) mobile; and e) text editor. A couple of tools belonged to multiple tool categories, however most satisfied only one. Participants' opinions are included within this breakdown, as conveyed verbatim in the discussions.

Table 3. Results of qualitative analysis.

Authoring tool	Category	Supporting quote(s)
Quest	browser	*It isn't ideal, some of the tabs and buttons on the user interface become too small for practical purposes as the only platform you plan to build it on.*
Borogove	browser	*Borogove is a web app that lets you develop Inform 6, Inform 7, Dialog, and Hugo. I've experienced difficulty using the either the standard Google keyboard or SwiftKey (my usual keyboard) with the editor though.*
inklewriter	browser	*Inklewriter would be incredibly easy to program on a phone or tablet, but it simply does not scroll with finger swipes, which is odd, because the input is almost exactly like the output.*
Seltani	browser	*Speaking of Web-based editors, I wrote a bunch of experimental code in Seltani from my phone. It was a nice experience. My phone had a hardware keyboard at the time, which probably helped.*
Undum	browser	*The more versatile is anything that runs on a browser:*
fi.js	browser	*The more versatile is anything that runs on a browser:*
ngPAWS	browser	*The more versatile is anything that runs on a browser:*
Quixe	browser	*The more versatile is anything that runs on a browser:*
Inform 7 (beta)	compiler	*I'm using Inform 7 on an Android pad. It's a beta version that seems to have stalled at beta. The compiler and the release mechanism work fine, but the IDE is flaky. You have to*

		generate a z8 output and run it on a z interpreter.
Playfic	compiler, browser	*Thankfully, one such compiler for Inform is available online: playfic.com.; I just started using Playfic.com and everything seems to be going swimmingly on my Android phone.*
Fabularium	interpreter	*Fabularium 1 is an Interpreter for Android but it also lets you create Inform 6, Glulx, and TADS 3 games. However, the editor is very bare bones and a bit difficult to work with since you can't adjust the font size.*
ZMPP	interpreter	*I use ZMPP for my mobile playing. It seems to be the most recently updated of the 4. I prefer not to play on web based interpreters as I am not always in range.*
Frotz	interpreter	*Much more convenient is the fact that iOS Frotz can open up z and glulx files from pretty much any other app, including Safari, Chrome, and DropBox.*
LegacyASL	interpreter	*…one is called LegacyASL and handles games written for Quest 1-4…*
WorldModel	interpreter	*…the other is called WorldModel and handles games written for Quest 5…*
Twisty	interpreter	*Twisty and JFrotz were pretty plain-jane.*
JFrotz	interpreter	*…only JFrotz could actually play all the games I threw at it (however crudely) so I stuck with that for my regular playing (and I play primarily when mobile, so that means JFrotz is what Interactive Fiction mostly looks like to me).*
Wunderverse	mobile	*Wunderverse on iPad.*
Texture	mobile	*If Texture ever comes out of alpha - that system was designed specifically for authoring on touchscreen devices.*
Lowres Coder	mobile	*Especially Lowrescoder is a lot of fun, if you're into 8bit-retro-stuff and have fun to write some simple code it's worth to take a look.*
Novello	mobile	*I am working on the app for creating interactive stories on mobile devices without coding. The app is called Novello.*
Interactive Story	mobile, browser	*They even have an app for android and the online version works great on mobile devices.; There's Interactive Story on the Android, but it's bloody horrendous because it doesn't allow enough user control, IMHO.*
vi/vim	text editor	*Touch keyboard with a suitable editor like vi/vim works surprisingly well even for coding, but it can never compete with a dual-monitor desktop setup, real keyboard and a well-configured emacs (or vim or sublime if you prefer).*
WriteRoom	text editor	*I use WriteRoom on iOS (and any old text editor on Android) to update text files into DropBox. On my desktop (on all the time) I have Twine in auto-build mode to update my .html output.*
Sadako	text editor	*Sadako is my own engine and it creates hypertext fiction in HTML, like Ink or Twine. It's just a couple javascript files so you can easily create games using any text editor on your phone/tablet. There's also a method for converting your source to JSON that only needs Termux 3 and node.js installed.*

Beyond our exhaustive survey of mobile-friendly tools, a few nuances into authoring IF on handheld devices were uncovered and they are described here. Their inclusion is intended to establish some parameters and framing for our subsequent tool taxonomy and its categorizations.

One user hints at a broadening palette of tools available to authors who make use of mobile software that remotes them into a desktop device or virtual machine: *"If you don't mean natively*

and absolutely anywhere, you have a few more options. SSH'ing into a terminal, or a remote desktop app for example." While certainly an intriguing datapoint worthy of inclusion in this discussion of the many solutions available to the modern IF author, we must draw the line somewhere—this wellspring of tools is in no way designed for mobile use even though they may be adapted for this purpose with a few workarounds. Clever solutions to accessing software remotely by smartphones are, nevertheless, a possibility for those with a very specific need that cannot be satisfied by more accessible browser or mobile app options.

Several participants in the sample population tend to favor text editors for handling projects, such as one user's solution to resolve a common scrolling challenge with software on mobile devices: *"…if you have to deal with an IDE that doesn't scroll, any kind of text editor should support scrolling. Look for one that can "share" the text with other apps - this will be a shortcut to get the source code into the IDE."* Text editors were not featured during a review of scholarly sources but are prominent within IF forum discussions, suggesting that this community is comfortable developing games with classic programming languages that require an interpreter or compiler to execute code.

The tendency for members of this online IF community to push the boundaries of mobile apps to realize their game vision does not end there; IF developers have found ways to create works from mobile devices by establishing their own unique developer workflows, such as this user: *"My dev route is: Droidedit, Inform, copy the output.z8 to Download, Text Fiction, delete previous output, import new output.z8 to Text Fiction, test… return to droidedit."* A tendency for members to push the boundaries of mobile apps reveals some elegance to the art of mobile IF-writing. The creativity involved in utilizing a decentralized suite of tools to produce a game might even indicate limitations present in the available tech, thereby revealing opportunities to solve demands presently unmet by handheld devices.

5. CONCLUSION

A new mobile IF tool taxonomy was developed through a synthesis of scholarly findings and results from a qualitative analysis of online author discussions. Mobile IF tools are now classified in accordance with four main categories: a) browser tools; b) mobile authoring systems; c) text editors, interpreters, and compilers (TE/I/C); and d) e-book editors. An additional category "other" was added so that any tools which fall outside of these four categories might still be accounted for. Their inclusion could prompt subsequent revisions of this taxonomy to better reflect the full spectrum of mobile IF authoring tools, adding new categories as they are uncovered.

After establishing tool classifications, all the tools described within this study were then grouped within this new taxonomy (see Table 4), providing a simplified representation of mobile IF authoring solutions in one place.

Table 4. A new responsive mobile interactive fiction tool taxonomy.

Browser	**Mobile**	**TE/I/C**	**E-book editors**	**Other**
Twine, inklewriter, Quest, ChoiceScript, Undum, Borogove, Seltani, fi.js, ngPAWS, Quixe, Playfic, Interactive Story	Interactive Story, Texture, Dorian, iStory, Wunderverse, Lowres Coder, Novello	Fabularium, Text Fiction, Twisty, Frotz, ZMPP, LegacyASL, WorldModel, JFrotz, Inform 7 (beta), Playfic, vi/vim, WriteRoom, Sadako	Stretchtext, ILSAditor, @note	N/A

This new classification system is relatively unchanged from the four main categories which emerged during the literature review, however there is one notable exception. One persistent trend in the https://intfiction.org/ forum discussion threads was observed where several authors toggle between multiple tools to develop their works, comprising a sort of "triad" of complementary apps belonging to text editors, interpreters, and compilers. Interpreters are typically only concerned with running a game (with some extensions beyond the core functionality), revealing an entire other dimension to the game development process missing from our original framework. As a result, the original "interpreter" category has been expanded into "text editors, interpreters, and compilers" or "TE/I/C" for short. While some interpreters and compilers contain built-in editors (e.g.: *Fabularium* and *Playfic*), it is quite common for developers to have a suite of tools on-hand to perform individual functions like writing and debugging in a text editor and playtesting on an interpreter. This category belongs to works developed in programming languages such as Inform 7, Zork Interpretive Language (ZIL), or TADS.

E-book editors are missing from forum discussions and appear to be more of an experimentation delivered within academic contexts. The nature of these tools could be more centered on transforming existing published digital works instead of generating wholly new content, making them quite niche and potentially limited in appeal, but they are nevertheless distinctive. Are e-book editors a subcategory of a broader, generalized multimedia software category? We have included this category in the framework for now, however future insights might identify a parent category for which these mobile tools belong.

This adaptive mobile IF tool taxonomy could benefit the IF community in a few ways. Firstly, it provides a lens with which the most immediate characteristics of mobile IF apps are conveyed to authors seeking the right solution for their project scope (e.g.: a game written in *Inform 7* might require a text editor and interpreter). Secondly, this taxonomy in its current form challenges practitioners and scholars to think through the full spectrum of mobile tools, inviting periodic revision in accordance with new technologies and practices that emerge from the IF space. Finally, navigating through this multi-step qualitative research methodology welcomes insights into the IF authorship domain beyond the immediate goals of this study (classifying and organizing mobile tools), uncovering practice and theory that would otherwise remain unexplored or as tacit practitioner knowledge locked in the consciousness of a broader userbase.

One forum participant puzzles over a lack of complete mobile IF authoring tools available on the market:

"It strikes me as odd that an activity requiring only reading and writing has not that many options for mobile. Certainly can't be the myth of no-keyboard: I can write perfectly well and fluidly with swype, but I could just as well connect it to a bluetooth keyboard. I also have plenty of text editors available. We just need a compiler."

Another participant in a separate thread provides a clear appeal for IF authoring solutions accessible by laptop, highlighting the limitations of smaller devices: *"Tablets and phones both support Bluetooth keyboards, though, at that point, you might as well use a laptop."* While our study does not include laptops in our working definition of "mobile IF authorship," their inclusion would significantly broaden the scope of tools, including solutions that Android and iOS handheld devices would be unable to handle under normal circumstances. A more liberal scope of research would likely dilute the concept of mobility to the point of superfluity, though one could certainly make a reasonable case for the laptop's inclusion given its portability.

5.1. Implications

For now, browser-based authoring tools appear to dominate the mobile market despite their obvious limitations [5-8], however there is still much promise and demand for mobile-exclusive authoring systems into the future as their full potential is still arguably unrealized [8]. With mobile devices and smartphones being such a permanent fixture on society, combined with IF's gamebooks origin, IF authoring tools which specifically addresses the idea of a game developer "on the go" will likely proliferate into the future. One thing is for certain when consulting the past: IF tools have evolved, producing works on different media and in a variety of rich formats. It is therefore reasonable to assume that the tools of the trade will continue to adapt to new user preferences with mobile devices uniquely positioned to facilitate the genre's creative continuity.

5.2. Future Research

While this early attempt at making sense of the handheld tools available to IF game developers help to set a foundational structure, there will be a likely demand for future surveys of the total tool ecosystem as new technologies and practices enter the space. The natural progression of mobile tools in response to new developer preferences could require periodic refreshing of this or a similar scholarly inquiry.

It is worth noting that this is only one taxonomy configuration of many possible arrangements. Future research might identify specific mobile tool characteristics and features that can be structured in a similar manner, according to user preferences and the "wishlist" features that are commonly sought-out in a tool. One classification system could, for instance, organize mobile tools according to operating systems or compatibility with specific devices. As such, researchers are encouraged to adopt a similar approach in their own constructions of mobile tool classification systems and enrich this research methodology as the need arises.

Finally, our attempts to make sense of the richness of IF tools make it easier to identify gaps in practice and respond to the evolving needs of its creative base, thus broadening the reach and universal appeal of the artform. With the presence of handheld devices operating in so many living contexts, IF authorship will surely find its way into the pockets of newer generations who marvel at the power of nonlinear storytelling in the same manner as generations prior.

REFERENCES

[1] Eve, E. (2007). All Hope Abandon: Biblical text and interactive fiction. *Digital Humanities Quarterly*, *1*(2).

[2] Emms, S. (2020, September 18). *Write and play interactive fiction with open source software*. LinuxLinks. https://www.linuxlinks.com/write-and-play-interactive-fiction-with-open-source-software/

[3] Tyndale, E., & Ramsoomair, F. (2016). Keys to successful interactive storytelling: A study of the booming "choose-your-own-adventure" video game industry. *i-Manager's Journal of Educational Technology*, *13*(3), 28–34. https://doi.org/10.26634/jet.13.3.8318

[4] Flynn, S., & Hardman, M. (2019). The use of interactive fiction to promote conceptual change in science. *Science & Education*, *28*(1-2), 127–152. https://doi.org/10.1007/s11191-019-00032-6

[5] Kirilloff, G. (2021). Interactive fiction in the humanities classroom: How to create interactive text games using Twine. *Programming Historian*, (10). https://doi.org/10.46430/phen0095

[6] Gómez-Albarrán, M., Sarasa-Cabezuelo, A., Sierra-Rodríguez, J.-L., & Temprado-Battad, B. (2021). Authoring and playing interactive fiction with conventional web technologies. *Multimedia Tools and Applications*, *81*(11), 14705–14747. https://doi.org/10.1007/s11042-021-11316-x

[7] Montfort, N, & Short, E. (2012). Interactive fiction communities: From preservation through promotion and beyond . *Dichtung Digital. Journal Für Kunst Und Kultur Digitaler Medien*, 1–14.

[8] Styles, M. G., Jung, S.-Y., Eto, C., & Draper, G. M. (2012). An approach to hypertext fiction for mobile devices. *Proceedings of the 2nd Workshop on Narrative and Hypertext*, 1–5. https://doi.org/10.1145/2310076.2310078

[9] Baker, S. (2021). Choose your own educational resource: Developing an interactive OER using the ink scripting language. *Code4Lib*, *51*.

[10] Granberry, T., Ramage, R., Blackburn, J., & Kuehny, J. (2020). Choose your own adventure: Using interactive fiction software to facilitate the case study instructional technique in a postsecondary agriculture course. *Conference: American Association for Agricultural Education National Conference*.

[11] Montfort, N. (2013). Riddle machines: The history and nature of interactive fiction. *A Companion to Digital Literary Studies*, 267–282. https://doi.org/10.1002/9781405177504.ch14

[12] Fitzgerald, T. (2013). GUInform: Interactive fiction for GUI prototyping. *Computer Science Faculty Publications and Presentations*.

[13] Sargsyan, M., & Madyarov, I. (2020). 6 English language learners reading interactive fiction: A case study. *The European Journal of Applied Linguistics and TEFL*, *9*(1), 109–130.

[14] Ahlbrecht, P. (2014, September 17). *Text fiction the Z-machine interpreter for Android*. Text Fiction - The best text adventure app for Android. https://textfiction.onyxbits.de/

[15] Cadogan-Cowper, T. (2018, November 18). *Fabularium: Interactive Fictio - apps on Google Play*. Fabularium: Interactive Fictio. https://play.google.com/store/apps/details?id=com.luxlunae.fabularium&hl=en&gl=US

[16] Cadogan-Cowper, T. (2019, October 3). *Fabularium: F-droid - free and Open source android app repository*. F. https://f-droid.org/en/packages/io.davidar.fabularium/

[17] Collins-Sussman, B. (2017, July 9). *Twisty, a text-adventure interpreter for Android*. GitHub. https://github.com/sussman/twisty

[18] Griffith, D. (2020, November 30). *Frotz - A Portable Z-Machine Interpreter*. Frotz. https://davidgriffith.gitlab.io/frotz/

[19] Harshitha, H. (2022). Choose Your Own Adventure: The evolution of digital interactive fiction and its use in language pedagogy. *Fortell*, (45), 44–55.

[20] Balint, A. (n.d.). *Interactive Story*. Interactive story. https://www.interactive-story.com/is/

[21] Fernandez-Vara, C. (2018, January 26). *Tools to make narrative games*. Vagrant Cursor. https://clarafv.com/2018/01/02/tools-to-make-narrative-games/

[22] Silberling, A. (2022, March 17). *Dorian's no-code, interactive storytelling app turns fiction writers into game developers*. TechCrunch. https://techcrunch.com/2022/03/17/dorian-interactive-storytelling-fan-fiction/

[23] Harkin, S. (2023). Vampires, drama, and romance: Examining the in-game interactive fiction tools in Episode and Dorian. *Proceedings of DiGRA Australia 2023*, 1–3.

[24] Kerthyayana Manuaba, I. B. (2017). Text-based games as potential media for improving reading behaviour in Indonesia. *Procedia Computer Science*, *116*, 214–221. https://doi.org/10.1016/j.procs.2017.10.041

[25] Paiva, A. B. (2018). Gender indeterminacy in translation: The case of R. L. Stine's Give Yourself Goosebumps gamebooks via Portuguese translation. *Revista De Estudos Anglo-Portugueses/Journal of Anglo-Portuguese Studies*, (27), 329–345. https://doi.org/10.34134/reap.1991.208.2710

[26] Sarasa-Cabezuelo, A., Sierra Rodriguez, J. L., & Diez Sanmartin, C. (2018). A tool for the digital edition of interactive fiction using Stretchtext. *2018 22nd International Conference Information Visualisation (IV)*, 356–361. https://doi.org/10.1109/iv.2018.00067

[27] Diez-Sanmartín, C., Gayoso-Cabada, J., & Sarasa-Cabezuelo, A. (2020). Use of critical annotation and interactive fiction for the creation of digital educational content. *International Journal of Emerging Technologies in Learning (iJET)*, *15*(09), 231–244. https://doi.org/10.3991/ijet.v15i09.12377

[28] Lim, H. (2016). Exploring ecosystem approach for games development and connected learning. *Computer Game Development and Education: An International Journal (CGDEIJ)*, *1*(1), 21–31.

AUTHORS

Tyler Wright holds a Doctor of Educational Technology (DET) with Central Michigan University, blending scholarly research into game development pursuits. His research interests include digital interactive fiction and game development pedagogies, which are at the center of his dissertation entitled *Digital inroads into nonlinear storytelling*. Tyler is an independent educational software developer and scholar with a passion for game design and development as a powerful multidisciplinary approach to instruction.

Dr. Ray Francis is a tenured graduate faculty at Central Michigan University (CMU). Dr. Francis currently teaches courses in evaluation and measurement, research methods, and research capstone seminars at the MA level. Dr. Francis' current research interests include aspects of student motivation in blended and online learning, concept mapping, prior learning assessment, authentic assessment, #MessyData, and the use of global field experiences.