Design and Development of a Mobile Application to Combat Digital Addiction and Dissociative States During Phone Usage

Dorian Lyons
Department of Computing&Informatics
Bournemouth University, Talbot Campus, Fern Barrow, BH12 5BB
Poole, United Kingdom
i7997197@bournemouth.ac.uk

Ceyda Kiyak
Department of Computing&Informatics
Bournemouth University, Talbot Campus, Fern Barrow, BH12 5BB
Poole, United Kingdom
ckiyak@bournemouth.ac.uk

Deniz Cetinkaya *
Department of Computing&Informatics
Bournemouth University, Talbot Campus, Fern Barrow, BH12 5BB
Poole, United Kingdom
dcetinkaya@bournemouth.ac.uk

Sarah Hodge
Department of Psychology
Bournemouth University, Talbot Campus, Fern Barrow, BH12 5BB
Poole, United Kingdom
shodge@bournemouth.ac.uk

John McAlaney
Department of Psychology
Bournemouth University, Talbot Campus, Fern Barrow, BH12 5BB
Poole, United Kingdom
jmcalaney@bournemouth.ac.uk

***Abstract*—Design features such as "infinite scrolling" and the state which keeps users engaged in this behaviour can be described as a timelessness or dissociative state and has links to the onset of digital addiction. As addiction/dissociation and the effect of these design choices could therefore occur automatically for the user, an active prevention technique is needed to combat this behaviour. In this paper, we present analysis, design and development of a proof-of-concept solution for the interruption of dissociative of users. A mobile application has been designed and developed for an android device which incorporates the prior research and provides features relevant to the prevention of the problem area. The prototype allows to set up interruption time period and to select the applications to interrupt users on a session-based usage viewpoint. An evaluation has been made through useful and constructive feedback received from user testing. Through evaluation of the prototype solution, the application has potential use not only as a product for the general user, but as a tool for research into other methods of digital dissociative state interruptions.**

Keywords— mobile application, digital addiction, dissociation

I. INTRODUCTION

Over the past two decades there has been a vast growth in adoption of mobile devices and technologies. This growth and advancement of the medium has brought along a varied array of applications and services, providing the user with tremendous functionality which is ubiquitously available. Many mobile applications are designed to increase the user's engagement and screen-time by using design techniques which are designed to maximise user retention [1]. These design techniques are found to be sometimes deliberately addictive and contribute to the formation of a digital addiction response [2]. Digital addiction is a researched form of addiction which has suggested to have many negative health and cognitive consequences [3]. As a health issue, digital addiction has a growing prevalence throughout the world and draws parallels to other addictive behaviours like gambling addiction and smoking [4].

Mobile application design principles like "infinite scrolling", which is prevalent on most modern media or social media websites, creates a continuous "infinite" stream of content for the user to consume [5]. This unrelenting onslaught of constant stimulation is typically absent of a stopping mechanism, which culminates in a repeating feedback loop and an eventual feeling of timelessness for the user [6]. Digital games or gambling applications are another avenue where these feedback loops are present and lead to an addictive activity which pushes the user into this timelessness state [7].

In the literature, there was a distinct absence of a software tool in which the user's state could be meaningfully interrupted on the app usage "session" level. New techniques to combat any form of digital addiction should be an increasingly important area of growth in software design and development. This paper proposes a solution of a mobile application which is designed to specifically combat this problem state. It allows the user to use a session-based approach to limit their device usage on individual apps. An interruption event will be used to bring the user out of the feedback loop and timelessness state which often leads to addictive behaviour. The aim of this research is to analyse, design, and develop a workable prototype solution which can act as a proof of concept for the interruption of problematic states of dissociation.

The next section presents literature review about the causes and possible solutions to screen addiction to provide the context as well as identifies existing techniques and related work. Section 3 and Section 4 present the system analysis and design respectively. Implementation details are explained in Section 5. After that, Section 6 discusses the evaluation. Finally, conclusions and future work are presented.

II. LITERATURE REVIEW

A. Screen Addiction Characteristics and Prevention

Screen addiction is a well-researched topic, especially in the realm of mobile phones and other smart devices whose application marketplaces contain millions of apps designed to compete for the user's attention. Xu et al. [8] argues that user activity and play time is associated with a service's revenue. Therefore, it would be a benefit to developers to consider design decisions to keep users invested for as long as possible. Increased activity and play time have been shown to be key in the formation of addiction behaviours.

Parallels can be drawn to long-held discussions on television addiction and related digital activities. Sussman and Moran [9] discuss screen addiction harmfulness levels, which can depend on what is being watched, how often, who is

watching, and whether the screen-time is communal. They also acknowledge some possible areas where screen addiction prevention can be practised. These include filters for certain types of media, more careful consideration of content within television programmes, screen-addiction informative advertising, and pop-up reminders of excessive screen-time on screen. Within the realm of gambling addiction, the content and timing of these on-screen pop-up messages can be factors in the effectiveness of the prevention. A more personal and engaging message displayed within a gambling session is shown to have more of an effect on a gambler's attitude and behaviour than a simple informative pop-up message at the beginning of a session [10].

Digital and online games are another realm where addictive practices are systematically used to keep a user's attention and maximise their playtime on the game. Ertemel and Aydin [11] suggest that the absence of a stop sign in most digital applications is a large contributor to screen addiction issues. Features like infinite scrolling and auto-play keeps the user in the application, constantly consuming digital media.

Another characteristic of screen addiction is the Fear of Missing Out (FoMo) phenomenon. "FoMo is characterized by feeling stressed thinking that something exciting or interesting might be happening elsewhere that the individual is missing." [12]. This by-product of the digital world leads to a behaviour of excessive checking of a user's device. Purohit et al. [13] explain this effect and the deliberate design incorporated into modern social media and digital media apps to achieve this FoMo trigger. A reduction of the visibility of stimulating notifications and pop-ups, as well as reducing access to overly frequented apps was shown to not only reduce the time spent on said apps, but also increase the user's enjoyment when using them.

Common prevention techniques designed to combat screen addiction tend to involve producing incentives for staying away from a device, or hard limits on a cumulative daily usage. Next section provides an analysis into the most popular Android app solutions. Other prevention techniques can be present within the application as an in-built addiction prevention feature. Online gambling services more frequently are including the ability for users to set limits on their gambling within the application or website itself. Also, pop-up reminders of these limits have been shown to have a positive effect on reducing the tendency for problematic behaviour [14].

B. Existing Solutions

Through research into applications and services designed to tackle screen addiction, specifically in the Android market, there are many solutions available under the categories of wellbeing and focus tools. Table 1 shows a selection of the top downloaded applications from the Google Play store. There is also an in-built Android native application called Digital Wellbeing which tracks the user's screen time on all installed applications in daily, weekly, and monthly filters. It also provides the ability to set limits on cumulative screen time for a day, which when the limit is reached sets that specific application to a greyscale appearance. Through looking into the most popular screen-time reducing apps available, it was apparent that the vast majority were performing the similar functionalities. For example, in the ActionDash app, daily time limits for individual applications can be set, which is a common feature among the listed apps.

TABLE I. ANALYSIS OF TOP DOWNLOADED RELEVANT APPS IN THE GOOGLE PLAY STORE

App Name	Features
Forest	- Staying off your phone will grow a virtual tree - Virtualised growing forest gamification - Reward scheme for growing real-life trees
StayFree	- Daily phone usage overview - Schedule application and website blocking - Compare usage with others online - Distracting website blocking - Detailed reports
ActionDash	- Daily phone usage overview - Focus mode - Individual application daily limits - Notification tracking (amount per application) - Compare statistics with others online
Freedom	- Create custom application blocklists - Schedule productivity sessions
AntiSocial	- Overview chart for: phone unlocks, application usage time, application launches - Three blocking modes: daily limit, scheduled, timer (system wide) - Application blacklist
Hold	- Earn points for not using your phone – gamification - Use accumulated points for real-world rewards - Compare points with others – leader board
Digital Wellbeing (Google)	- Overview chart for: total screen time, individual application screen time, number of unlocks, number of notifications - Set daily application usage timers for individual applications, resets at midnight - Bedtime mode - Focus mode

Another factor discovered, is that many of the most popular applications in this realm involve a form of incentive or gamification techniques to keep the user engaged and offer some positive reinforcement. Gamification is a design approach where certain tasks are incentivised using game-like techniques to make them more attractive to the user [15]. The app "Forest" has a unique user incentive feature of a virtual tree which grows when a user is not using their phone. It also is strong thematically with the rewards scheme of planting and growing real-life trees. "Hold" also includes a real-world rewards scheme and points for not using your phone. Other applications surveyed included more passive user incentive features for example, "ActionDash" allows for a phone usage statistics comparison with other users online.

A further look into the most popular applications of this type involved collecting user reviews. Common among most of the reviews analysed is the aspect of user choice and agency. Some of them had complaints of not having an easy way to stop or pause the functionality. Accuracy in screen time detection was another issue found in user reviews, e.g., there were reviews claiming where general screen time stats seemed to not be reflecting the user's experience. Overall, the main chosen method of screen-time prevention in the apps analysed was a block to the launching of either user selected apps, or all applications on the device. The threshold for blocking an application from launching was mostly a cumulative daily limit on time used. A method present in "ActionDash" is the ability to block app notifications, and "StayFree" allows the user to also block the launching of chosen websites.

A screen-time reducing method that was seemingly absent from Google Play store apps was functionality designed to interrupt the flow state during app use. The app "AntiSocial" has a feature of three modes of blocking app usage: a daily limit for individual apps, a scheduled device-wide block, and a device-wide daily timer. The device-wide daily timer is the closest to achieving that method, but it is not available for the scope of an individual application, or a session-based application.

C. Flow state, Dissociation, and Interruptions

Most research on the effects of having these devices constantly nearby target the distraction factor and therefore the tendency to interrupt the "flow" state. Csikszentmihalyi [16] states that "Flow refers to an optimal experience in which a person is full immersed in an activity due to a feeling of energized focus". Such a state can be valuable for productivity, so any interruption of said state is a negative by-product of the constantly connected modern world. Excessive smartphone use is one such interruption which has been found to have the effect of interrupting a productive flow state. Dittrich [17] concludes that "one's smartphone use does not leave flow states and productivity unaffected.". Therefore, it is to be believed that any interruption to this flow state must be negative.

The flow state may be useful for activities which can be categorised as productive, or task which are related to work and life goals; however, the flow state does not only occur in such circumstances and is a factor in addiction behaviours in digital gambling, gaming, and media consumption. The characteristics within this flow state can be a precursor for addiction [18]. The feeling of immersion and full concentration on one task or application is stimulating and can lead to the user having a tolerance to excessive use. As Khang et al [18] explains, "At the phase of tolerance development, media users tend to seek stronger stimuli, which this research assumes is correlated with the addiction phase". The flow state is accepted to be a positive and rewarding experience, and one that can be enjoyable and engaging; so, it is natural that such experiences present in smart devices could lead to addictive or obsessive behaviour [19].

Infinite scrolling is a technique used widely in modern software design. It involves the constant and continuous delivery of content as the user browses – usually by scrolling down through the page [5]. This technique has been found to be problematic and a contributor to users subconsciously spending longer on an app that includes it like Facebook or Twitter [20]. Infinite scrolling in software is designed to remove interruptions to browsing and therefore raise user engagement and immersion [21]. The removal of these interruptions in browsing can lead to a behaviour of mindless scrolling and a feeling of timelessness. Baughan et al. [22] explains this effect and how it is achieved through design, specifically in modern social media applications, and concludes that both internal and external interventions have a meaningful effect on reducing said behavioural pattern.

One of the external interventions analysed was a pop-up time limit dialog which the user was forced to acknowledge before resuming (or not resuming) use of the device. This type of interruption was found to be effective at bringing users out of a dissociative state. Interruptions can be either expected or unexpected. Expected interruptions have been shown to cause a natural break-off point for switching a task or behaviour [21]. A state of flow when interrupted can be difficult to recover. This is problematic in respect to productive activities but could be a useful method in curtailing harmful behaviour brought on by an unproductive, dissociative flow state.

III. ANALYSIS

The collection of requirements for the prototype solution is an important process which if done correctly will lead to a more defined and useful product at the end of development [23]. Initial requirement gathering sets the groundwork for the design and implementation stage of development and is a valuable process for further understanding the potential stakeholders and therefore ultimate scope of the project. Firstly, it was important to take into consideration the ultimate goals of the project. As discussed previously the prototype solution needs to provide the effect of managing and interrupting states of timelessness and dissociation which could be brought on by other installed apps on the device. Therefore, an initial list of functionalities was created to ensure that these goals were not missed.

Goals related to monitoring session-based, app-specific screen time are:

- Track which application is running on the screen,
- Know how long an application has been on the screen,
- Run in the background while other apps are used.

Goals related to interrupting app-usage session are:

- Provide an interruption event, and
- Keep an internal timer for an app-usage session limit.

Secondly, a stakeholder analysis was conducted. This was used in conjunction with the previously listed functionalities to provide a more user-centric scope for the elicitation of requirements.

The analysis was conducted in the form of coming up with user stories and use-case scenarios. User stories help determine possible goals for the software in an end-user centric view and are shown to have an effect in improved software quality and productivity [24]. Cohn's template of producing user stories was used to help produce succinct and real-world identifiable stories [25]. These are displayed below in Table 2. The user scenarios were analysed to come up with use-cases for the application and they were modelled in a use-case diagram.

Finally, a further analysis of current similar applications available was undertaken to determine if there were any additional features or functionality which could be included in the artefact. The gathered requirements were then prioritised considering the project time and resources. For this project, the FURPS+ (Functionality, Usability, Reliability, Performance, and Supportability) model is used for defining the non-functional requirements.

TABLE II. USER STORIES

As a/an	I want to…	So that…
Parent	Set a time limit for my child's gaming session	Excessive use does not hinder their schoolwork or mental health
University Student	Limit my time spent endlessly scrolling social media	I can focus more on my studies

Teenager	Be reminded when I have spent a certain amount of time playing a game on my phone	I can spend more time away from my device
Adult	Have a hard cut-off point when using online gambling apps	My habit of "just one more bet" can be prevented
Adult	Be interrupted from my social media scrolling in bed at night	I can get a better night's sleep

IV. DESIGN

User interface (UI) mock-ups were designed in accordance with Nielsen's usability heuristics [26]. Sample UI designs are displayed in Figure 1.

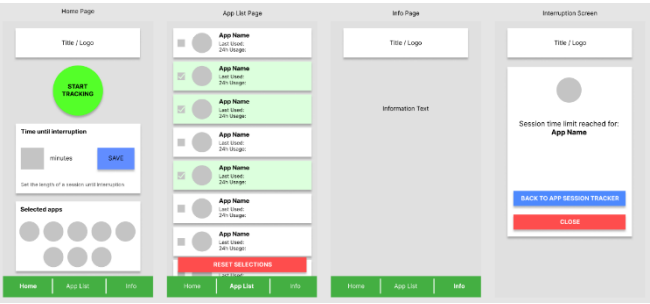

Fig. 1. User Interface designs

Below are a few examples of how these usability heuristics were considered in the design of the application.

Aesthetic and minimalist design: The user interface was designed to be as simplistic and easy to read as possible. The colour scheme and placement of functionality on separate "cards" helps with this minimalist design and prevents the application's interface from feeling too busy or confusing. Showing when the listed application was last used and the usage in the past 24 hours stat help the user gain context for whether said application requires tracking.

Visibility of system status: Feedback for user interactions needed to be intuitive and highly visible. Android developer documentation suggests that the use of a service in which the service's task should be noticeable by the user, must have a persistent status bar notification. The persistent notification will display updated information on which application is being tracked, and the active that is being run.

Match between system and the real world: Information and functionalities are displayed in a natural order depending on their importance. Buttons and interactable objects throughout the application are designed to make them look like they can be interacted with, and colouring of objects was made to fit the user's expectations of that interaction based upon prior conventions.

User centric design principles were applied to assure better user experience. This not only comes in the form of allowing the user to control when the tracking service is running, but also the manual choice of selecting a session length. Also, the functionality of having individual or multiple applications selected to be tracked by the user was key in providing maximum flexibility. Users with different thresholds for an acceptable session time limit, or whether the tracking occurs throughout the day or only at specific times are all catered for and reflected in the user centric design.

A. *Scope and Technical Design Considerations*

Due to the time limitation, certain design considerations and functionality which at first seemed natural to include were discarded. The first of these design choices was to not include an interface for the user to view their daily phone usage stats and total daily screen time data. This functionality is widely present in many other apps and is also available in Android pre-loaded application "Digital Wellbeing". It is not within the scope of this application to extend or imitate this feature, and not pursuing it allows for more focus into the core session-based tracking functionality.

Another design choice was to not include any user incentive or gamification techniques within the app. Although these may be useful avenues to expand the app in the future, for this project it was more important that the artefact produced was a tool demonstrating the project's goals. For the design of the application, it was important to closely follow the gathered requirements and ensure that the scope of the design was focused on the goals of the project. The requirements were further analysed to generate more of an understanding of the technical implementation solutions that the application will need. A UML Class diagram was also produced to detail the structure of the underlying systems and provide a visual overview of how all the systems interact.

V. IMPLEMENTATION

For this project, a Kanban approach was used. A service called Trello was used to create the Kanban board. Using this methodology allowed for following agile principles such as welcoming changing requirements, delivering working software frequently and promoting sustainable development. The prototyping methodology was chosen for the development of the solution. Producing multiple versions and deliverables allows for a more flexible development of features, with a clear cycle of implementation and then testing against the requirements and the goals. Android Studio Integrated Development Environment (IDE) is used as the development platform which provides access to in-built mobile device emulation. Through a detailed analysis it was decided to target the minimum Android API level of 9.0. This ensures that all the functionality of the artefact can be reached, and the market-share reach of the application is still not significantly affected.

The Developer Documentation for Android states that app permissions help privacy and security for the user by restricting access to both restricted data and restricted actions. Restricted data might involve sensitive user data, or system state and usage data. Permissions for such functionality need to be declared in the application's manifest file, with active runtime permissions needing their own user prompt when the app is loaded. The developed application does contain functionality which requires this type of permission declarations such as "QUERY_ALL_PACKAGES" to gain access to the installed apps, "FOREGROUND_SERVICE" to run functionality in the background separate from the app, "SYSTEM_ALERT_WINDOW" to allow for the full-screen interruption event notification, etc.

A. *Tracking Service Functionality*

The foundational logical functionality of the artefact requires the concurrent tracking of the length of time any specific chosen application has been on the screen of the device, and to know when that elapsed time has reached the user-chosen time limit value. Inherent in this functionality, the

tracking behaviour must run continuously in the background, unattached to the artefact application. Therefore, the screen-time tracking solution had to continue running both when the parent application was no longer active on the device's screen, and if the parent application had been closed completely by the user.

It was decided that a foreground service would be used, as it would allow persistent periodic execution of a task free from the parent application being active. The user can manually start and stop the service from a button in the parent application. As the foreground service is running, the persistent notification is utilised to provide an update on the status of the tracking functionality. Figure 2 left hand side shows how the tracking timer is reflected in this notification as it is running.

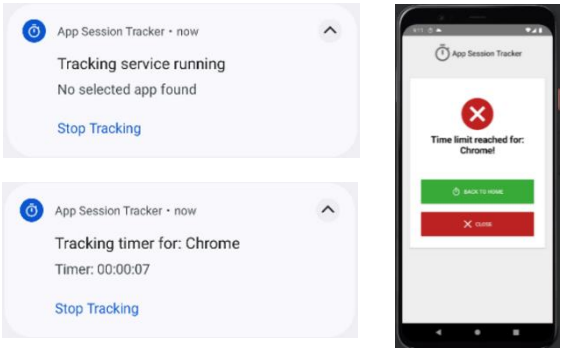

Fig. 2. Interruption event screen

The logic to provide the app tracking functionality requires information on when an application has appeared in the foreground on the device's screen, and when it is no longer in the foreground. The UsageStatsManager class in the Android API enabled the access to useful data on the running device's usage history and statistics. Within this class, UsageEvents objects can be queried to provide the application with necessary data on the switching of states of running applications. These UsageEvent objects come with a "type" attribute which describes whether the event pertains to an application being paused, resumed, or stopped. The tracking functionality used this information to determine when an app which was selected for tracking was active on the screen, and then a timer could be started until the application was closed/paused, or the time limit had been reached. This function would then run continuously every 1 second while the foreground service is running. Having the code executed every second is the longest interval required for the accurate timing of an app-session, any shorter interval would cause an unnecessary increase in resources used for the running device. The ability for the user to stop and start the running service manually also encourages use of the service when needed.

As this functionality was likely to be running for long stretches of time in the background, performance testing was done to ensure this functionality did not use more than enough resources. Through testing and debugging of this functionality in the earlier stages, using different types of apps as tracking targets, it became apparent that many apps through general use would sometimes produce a short pause UsageEvent quickly followed by a resume event. An example of this occurring is when tracking the concurrent app usage time on the "BBC News" app, loading a new article to read within the app would reset the app's timer back to 0 seconds. Similar issues occurred while navigating through apps like "Instagram" and "Twitter". As the resume event would come very shortly after the pause, the ultimate solution to this issue was to check whether the usage events were coming from the same currently tracked app and enforce a buffer in which the tracking timer's start time value would not be overridden. This small buffer also had the benefit of creating a more accurate reading of when an app's screen-time session had really ended for the user.

B. Interruption Functionality

One of the essential requirements is the need for an interruption event once the chosen time limit has been reached for a selected application. This requirement is necessary for demonstrating the effect previously mentioned of interrupting the flow state of an unproductive session on an application. To achieve this, the user's saved time limit is checked against the current application's session timer. Once the session timer reaches the time limit value, an interruption event is created. Initially this interruption event was using the Android Notification feature set, which allowed for a notification to be built which would display at the top of the screen and in the Android notifications drawer, much like when a text message or email is received. The device would make a notification sound if the user had notification sounds unmuted. This initial solution however was too easy for the user to circumvent or just ignore. The assumption with the tracking service and interruption event features are if the user has opted into having the tracking service running, then they are expecting an interruption which is more substantive.

The next iteration of the interruption event used a new application activity to display a full-screen window informing the user of the time limit being reached. This was done by creating a new activity within the parent application and preserving that activity's intent within the tracking service to have it ready to be displayed when needed as illustrated in Figure 2 right side.This newly created window was able to have data of the name of the application being interrupted sent to it to be displayed for the user. This second iteration of the interruption functionality was much more effective at creating the desired flow-state interruption behaviour. A full screen window over the top of the interrupted application does a better job of providing the intended effect of bringing the user out of the flow and feedback loops previously mentioned which may be present in the affected app.

VI. EVALUATION

The prototyping methodology used in the implementation of the artefact ensured that a working application prototype was available early in development. There were many different iterations attempted to develop and test the tracking functionality and the interruption functionality. To gather user feedback, a version of the application was made available to selected users for controlled user testing. A user feedback form in the style of a questionnaire was created using Google Forms, and the link to this form was provided to users who had access to the application. There were a total 9 participants. The participants chosen for feedback were not selected based upon any specific criteria, the only pre-requisite was that they had access to a mobile device which ran on the Android system. Users were asked to list the types of applications they typically chose for the tracking functionality. From this question the two main responses were for social media apps and games. This was not unexpected as from the prior research, these kinds of apps are likely to contain design

choices which initiate the dissociation response. This point is further illustrated by the overwhelming response to the question asking if the user had ever experienced the "timelessness" sensation with using an application. All of the participants answered that they had at some point experienced this, and for over half of them it was a regular occurrence.

The overall reception to the app and its usage was positive, with every respondent scoring the application at least 4 out of 5 and the vast majority scoring the maximum of 5. Participant views on the performance of the application were similarly high, and the intuitiveness of the user interface was more mixed but overall, largely positive. Every participant responded positively about the interruption events, answering with either a 4 or 5 out of 5 on the scale. Most participants from the questionnaire chose to manually switch the tracking service on and off throughout the day, with about a third of the participants keeping it running continuously. These results provide the question of whether users who kept the service running continuously generally felt more positive about the interruption helpfulness (answering 5 out of 5) and if the users who toggled the service throughout the day did so due to them having a lesser opinion of the interruption functionality.

VII. CONCLUSIONS AND FUTURE WORK

This paper presented a system analysis and design for a mobile solution for the interruption of problematic state of dissociation to combat digital addiction or excessive use. An evaluation has been made through useful and constructive feedback received from the user testing. The solution has potential use not only as a product for the general user, but as a tool for research into other methods of digital dissociative state interruptions [29]. Further user tests are planned for evaluation and to show the feasibility of the study in the future.

As a future work, more detailed statistics like "total count of interruptions for each app" or "previous session time" can be implemented. Similarly, providing more information about screen-addiction and dissociative states to the user within the app would add to the ability for users to gain knowledge on the subject matter, and therefore make more informed decisions on their app usage. From the survey results, useful considerations were offered for future development and possible avenues of improvement like a notification reminder for the user to switch the tracking service on or off, tracking service scheduling options, repeated or delayed interruptions, customisation such as choosing different interruption types, and accessibility considerations. Further improvements in creating a more interactive interruption event for the application would be a beneficial aspect to further develop the app. Incorporating current and future research in the problem domain into the app solution can lead to it being an increasingly useful tool in the future.

REFERENCES

[1] Neyman, C.J., 2017. A survey of addictive software design.

[2] Andersson, H., 2018. Social media apps are 'deliberately' addictive to users. [online] BBC News. Available at: https://www.bbc.co.uk/news/technology-44640959 [Accessed 11 March 2022].

[3] Small, G., Lee, J., Kaufman, A., Jalil, J., Siddarth, P., Gaddipati, H., Moody, T. and Bookheimer, S., 2020. Brain health consequences of digital technology use. Dialogues in Clinical Neuroscience, 22(2), pp.179-187.

[4] Cash, H., D. Rae, C., H. Steel, A. and Winkler, A., 2012. Internet Addiction: A Brief Summary of Research and Practice. Current Psychiatry Reviews, 8(4), pp.292-298.

[5] Loranger, H., 2014. Infinite Scrolling is Not for Every Website. [online] Nielsen Norman Group. Available at: <https://www.nngroup.com/articles/infinite-scrolling/> [Accessed 18 March 2022].

[6] Widyanto, L. and Griffiths, M., 2006. 'Internet Addiction': A Critical Review. International Journal of Mental Health and Addiction, 4(1), pp.31-51.

[7] Montag, C., Lachmann, B., Herrlich, M. and Zweig, K., 2019. Addictive Features of Social Media/Messenger Platforms and Freemium Games against the Background of Psychological and Economic Theories. International Journal of Environmental Research and Public Health, 16(14), p.2612.

[8] Xu, Z., Turel, O. and Yuan, Y., 2012. Online game addiction among adolescents: motivation and prevention factors. European Journal of Information Systems, 21(3), pp.321-340.

[9] Sussman, S. and Moran, M., 2013. Hidden addiction: Television. Journal of Behavioral Addictions, 2(3), pp.125-132.

[10] Haefeli, J., Lischer, S. and Schwarz, J., 2011. Early detection items and responsible gambling features for online gambling. International Gambling Studies, 11(3), pp.273-288.

[11] Ertemel, A. V., & Aydın, G., 2018. Technology addiction in the digital economy and suggested solutions. Addicta: The Turkish Journal on Addictions, 5, pp.665−690.

[12] Przybylski, A., Murayama, K., DeHaan, C. and Gladwell, V., 2013. Motivational, emotional, and behavioral correlates of fear of missing out. Computers in Human Behavior, 29(4), pp.1841-1848.

[13] Purohit, A., Barclay, L. and Holzer, A., 2020. Designing for Digital Detox: Making Social Media Less Addictive with Digital Nudges. Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems,.

[14] Auer, M., Hopfgartner, N. and Griffiths, M., 2018. The effect of loss-limit reminders on gambling behavior: A real-world study of Norwegian gamblers. Journal of Behavioral Addictions, 7(4), pp.1056-1067.

[15] García, F., Pedreira, O., Piattini, M., Cerdeira-Pena, A. and Penabad, M., 2017. A framework for gamification in software engineering. Journal of Systems and Software, 132, pp.21-40.

[16] Csikszentmihalyi, M., 1990. Flow: The psychology of optimal experience (Vol. 1990). New York: Harper & Row.

[17] Dittrich, S.M., 2020. Understanding the relationship between smartphone addiction, flow, and productivity (Bachelor's thesis, University of Twente).

[18] Khang, H., Kim, J. and Kim, Y., 2013. Self-traits and motivations as antecedents of digital media flow and addiction: The Internet, mobile phones, and video games. Computers in Human Behavior, 29(6), pp.2416-2424.

[19] Hull, D., Williams, G. and Griffiths, M., 2013. Video game characteristics, happiness and flow as predictors of addiction among video game players: A pilot study. Journal of Behavioral Addictions, 2(3), pp.145-152.

[20] Mildner, T. and Savino, G., 2021. Ethical User Interfaces: Exploring the Effects of Dark Patterns on Facebook. Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems,.

[21] Zhang, Y., Liu, L. and Ho, S., 2019. How do interruptions affect user contributions on social commerce?. Information Systems Journal, 30(3), pp.535-565.

[22] Baughan, A., Zhang, M., Rao, R., Lukoff, K., Schaadhardt, A., Butler, L. and Hiniker, A., 2022. "I Don't Even Remember What I Read": How Design Influences Dissociation on Social Media. CHI Conference on Human Factors in Computing Systems,.

[23] Zowghi, D. and Coulin, C., 2005. Requirements Elicitation: A Survey of Techniques, Approaches, and Tools. Engineering and Managing Software Requirements, pp.19-46.

[24] Lucassen, G., Dalpiaz, F., Werf, J. and Brinkkemper, S., 2016. The Use and Effectiveness of User Stories in Practice. Requirements Engineering: Foundation for Software Quality, pp.205-222.

[25] Cohn, M., 2008. Advantages of the "As a user, I want" user story template. [Blog] Mountain Goat Software, Available at: https://www.mountaingoatsoftware.com/blog/advantages-of-the-as-a-user-i-want-user-story-template [Accessed 9 April 2022].

[26] Nielsen, J., 2005. Ten usability heuristics.