Mobile Application of Information System for Smartphone Use Control

Yaroslav Neznaradko and Lyubomyr Demkiv

Lviv Polytechnic National University, S. Bandera Street, 12, Lviv, 79013, Ukraine

Abstract

The context of this paper is related to smartphone usage. In the current era, smartphones become our daily drivers. People use smartphones anywhere and anytime. This technology gives us endless possibilities, but at the same time, it gives us high responsibility for our time and our intentions. There are numbers of research papers that proved that smartphones and social media on these platforms have a high impact on our professional and personal lives. Smartphone usage monitoring and control is one of the priorities to be productive on work and have good mental health. This paper introduces detailed implementation and design of smartphone usage control method. The output of this work is a publically released application, which is available for iOS and Android and available on the corresponding mobile apps markets Google Play and App Store. A static and informative website with all required information about the project was implemented and deployed to https://acture.app. The information system was designed from scratch. Firstly the competitor analysis research was done. As a result, five existing applications were reviewed. The next step was system analysis. The system analysis included: analytics of goal tree hierarchy, conceptual model development using a series of UML diagrams, defining functional and non-functional system requirements. The tools and method review was conducted after the system analysis. A toolset was created that included programming languages, frameworks, IDEs, databases, UI/UX tools and analytics tools. And the final step included implementation details and results analysis. Implementation details contained the mobile app screenshots and a detailed explanation of technical solutions. Another part of the last step was the result analysis, including the number of downloads, users feedback and overall system effectiveness. Users accepted the system positively, and the project has more than 15 000 downloads with more than 200 pieces of feedbacks. The project is active, and a development roadmap is created.

Keywords 1

Screen time, social network, apple screen time, google digital wellbeing, user interface, google play, digital wellbeing, pairwise comparison, app store, useful link, non-functional requirement, google play console, android application, screen time monitoring, digital wellbeing application, quality criterion, foreground service, functional requirement, conceptual model, application development

1. Introduction

In 2020, the number of smartphone users was about 3.5 billion people, 44.87% of the world's population. The data in different countries will differ in percentage, but the figure of 3.5 billion is quite prominent. Almost every second person has a smartphone [1]. To set the research task, it is necessary to consider the smartphone as a technology from its beginning. The exact date when the smartphone appeared as such does not exist. After all, many consumer electronics companies, such as IBM, Motorola, Nokia, Apple, produced similar cell phones, which were functionally close to a smartphone [2]. But what distinguishes a smartphone from a regular phone? And why do most people associate the

MoMLeT+DS 2021: 3rd International Workshop on Modern Machine Learning Technologies and Data Science, June 5, 2021, Lviv-Shatsk, Ukraine

EMAIL: nezn27@gmail.com (Y. Neznaradko); demkivl@gmail.com(L. Demkiv)
ORCID: 0000-0001-5546-6631 (Y. Neznaradko); 0000-0002-2802-3461 (L. Demkiv)

advent of the smartphone with the release of the first iPhone? The answer is software and the development of cellular networks. After developing 2G, 3G, 4G cellular networks and creating them, in 2008, two and currently the most popular ways to distribute software for mobile devices were launched: Google Play and App Store [3-4]. From 2008 to 2010, the transition in countries from 3G to 4G began [5]. During these years, the number of smartphone sales on both iOS and Android platforms doubled and continues to grow today [6]. The smartphone, a technology that has emerged relatively recently within the history of technology, has captured 50% of the population in such a short period of 12 years. What brings a smartphone into people's lives that almost every second person had this electronic device in 2020? It is tough to list all the smartphone features now because there are many of them. The correct way to record the possibilities is the division into categories.

For example, consider the categories on Google Play.

Table 1
Categories of applications on the Google Play platform 2020

Category	Examples
Art and design	Tools for drawing, art, design, sketching and colouring
Cars and transport	Car shops, car insurance, car price comparisons, road safety, car reviews and news
Beauty	Makeup lessons, hairstyles, tools for creating an image, cosmetics stores, makeup simulators
Books and reference books	Book readers, reference books, textbooks, dictionaries, thesauri, wikis
Business	Applications for reading or editing documents, order tracking, remote management, e-mail management, job search
Comics	Games based on comics, comics
Communication	Messaging, chat and instant messaging, dealers, address books, browsers, call management
Dating	Finding a couple, romantic communication, building relationships, finding acquaintances, finding love
Education	Exam preparation, educational materials, dictionaries, educational games, language learning
Entertainment	Streaming videos, movies, TV series, interactive entertainment
Events	Tickets to the cinema, concerts and sporting events, purchase tickets from intermediaries
Finances	Banking, ATM search, financial news, insurance, taxes, asset portfolios and trade, commission calculators
Food and drink	Recipes, restaurants, food guides, wine tasting and search, drink recipes
Health and fitness	Personal training, physical activity tracking, diet and nutrition tips, health and safety, and more
House and farm	Search for houses and apartments, repair, interior design, mortgage, real estate
Libraries and demos	Software libraries, technical demos
Lifestyle	Style guides, planning weddings and parties, reference guides
Maps and navigation	Navigation tools, GPS, maps, vehicles, transport, public transport
Medicine	References for medicines and clinical trials, calculators, manuals for medical institutions, medical journals and news
Music and audio	Music services, radio, music players
News and magazines	Newspapers, news aggregators, magazines, blogs
Parenting	Pregnancy, care for children of different ages
Personalization	Wallpapers, animated wallpapers, home screen, lock screen, ringtones
Photo	Cameras, photo editing tools, photo management and sending

Productivity	Notepad, to-do list, keyboard, printing, calendar, backup, calculator, conversion
Shopping	Online shopping, auctions, coupons, price comparisons, product lists, product reviews
Social	Social networks, marks in places
Sport	Sports news and comments, tracking the results of competitions, managing a fictional team, coverage of sporting events
Tools	Tools for Android devices
Travel and local information	Travel booking tools, joint car trips, taxis, guides, information about local companies, travel planning tools, tour booking
Video players and editors	Video players, video editors, media storage
Weather	Weather forecast

You can also consider the categories of games on Google Play [7] as arcade, action, quizzes, puzzles, casinos, casual, card games, training, table, racing, adventure, role-playing, simulators, family, verbal, sports, strategy. From these categories, we can conclude that smartphones can be present in most human activities, perhaps even in those where you do not need an Internet connection. It is even possible to assume that you can pick up the phone in the morning, constantly interact with it throughout the day and put it only in the evening before bedtime. All areas of life, from entertainment to work - are covered by software that can be installed on a smartphone.

This large number of categories can be divided into two main categories: tools and services (which include means of consuming information and content or entertainment). From the first category, everything is clear. Users use services when they need them, for example, ordering a taxi through the application when you need to get home. It's just a clear and understandable use. But with the second category, everything is a little more complicated. The second category includes social networks (Instagram, Facebook, TikTok, Twitter, etc.), games, applications from news sites, forums. And as studies show, users spend the most time in this category of applications [8, 9]. Before considering the research results in more detail, it is necessary to introduce the concept of screen time, which is equal to the active time spent by the user on the screen. If my phone was functional for an hour a day, then my screen time for a day is 1 hour. Also, along with the screen time, consider the number of interactions with the phone (for example, the number of lifts or the number of screens unlocks). Usually, these are the applications that the user has used during this time, i.e. during user sessions. The description of the smartphone usage session should include the beginning of the interaction and its end (unlock time and lock time), i.e. timestamps, a list of applications that the user used during this session.

So, according to research in the world, the average screen time including all devices (PCs, laptops, smartphones, and smartwatches) is 6 hours 42 minutes, of which 3 hours 14 minutes - phone use. In addition, it should be noted that the user spends 144 minutes from this time on social networks. So, to sum up, during life, it is six years and eight months [8,9]. Interestingly, this time can be spent in different ways. And here, it is essential not a quantitative but qualitative assessment of this time, which is subjective, i.e. each user determines what is useful for him and what is not. Studies confirm that excessive use of social networks and their use on a smartphone can negatively impact a person's psychological health [10]. An objective assessment of service is the amount of information and its variability. For example, you can read a book or view photos on Instagram and read political tweets on Twitter. Of course, we can say that the book may not be helpful or unnecessary, but if the user reads the reader on a smartphone, he has a limited amount of information that he can consume. And in the case of Instagram, Twitter, TikTok, and Facebook - the situation is radically different.

Social networks in 2020 are similar to slot machines [11]. And all because of an exciting way of providing information. Information on popular social networks is presented in posts in an endless feed, which can be updated. Thanks to machine learning algorithms and artificial intelligence, the meal is created specifically for your interests. In addition, users have the opportunity to interact with posts, i.e. to respond to them in the form of comments, preferences, distributions, and the authors of these posts to receive information about interactions. The question arises: what are slot machines for here? Users often abuse social networks because there is everything they like, and the brain selects the format of

interaction. The brain secretes dopamine when viewing the Instagram feed or when there is a notification that the user has received a new message [12].

But why is this so? Are all social networks and entertainment applications addictive? Of course not, but the most popular ones, such as Facebook, have entire departments that deal with human psychology and create more interactive user interfaces, making applications and websites addictive.

Why does Facebook need this? Based on research and interviews with former employees of these companies, Facebook aims to collect user data and, of course, personalized advertising [13]. These are billions of businesses that make money on data collected from social networks. The capitalization of the advertising market in social networks is $ 23.68 billion [14].

Now you can go back to smartphones. Today, this device can easily fit in your pocket, and at any time, less than a second, we can be online. Considering the prevalence of smartphones and the results of research on smartphones in everyday life, and adding the coronavirus pandemic [15], the development of tools and analysis to control and monitor screen time is relevant.

The purpose of the task is to create a tool for planning intentions of the phone usage and monitoring these intentions. It will reduce screen time and increase the efficiency and relative usefulness of the time spent on the smartphone. The object of study is user behaviour during use smartphone and the purposes for which users use the smartphone. The subject of research is the process of beginning interaction with a smartphone, namely it's unlocking. The practical value of the work is to create an application that assists the user in the intended use of a smartphone in an era when the user's attention is a product for some companies. The relevance of the work has been confirmed by several scientific publications on the impact of modern social networks and the capabilities of smartphones on human health. The result is a developed application Actuflow, which is designed to plan using a smartphone, available on the two most common mobile platforms: iOS and Android. More than 15,000 users installed the application. Many users have reported a reduction in screen time of about 30% (while using the application). In addition to reducing time, users have reported a reduction in the impact of distractions to range in notifications and messages from other applications.

Approbation of the results was conducted on popular platforms for distribution of applications (ProductHunt, Betalist, HackerNews, Reddit) and after the marketing campaign collected user feedback, confirming the efficiency of the implemented method, accordingly, the information system. The application has repeatedly been recognized as one of the best applications of the month in 2020.

2. Related works

2.1. Existing approaches and methods of screen time control

After receiving the research results showing the direct impact of excessive use of social networks and other entertainment applications on users, various methods began to appear to help reduce this impact. Between 2010 and today, several publications have emerged on how to minimise the impact and how to use this technology properly. It is worth noting that most of these works apply not only to smartphones but in general to all devices that allow you to use the Internet. One of the first is the American journalist Nicholas Carr, who described in detail how the use of the Internet affects our brains [16]. Also worth noting is the work of Calvin Newport, an associate professor of computer science at Georgetown University. In his first book, he set himself the task of helping information workers (he calls them in his book "knowledge workers") to make their work process and work more productive and to reduce the number of distractions [17]. And in the second book, Calvin Newport collected ways and methods of controlling the use of devices in everyday life [18].

After reviewing all the methods, they can be divided into several categories: changing device settings to reduce the number of distractions, monitoring and controlling screen time, periodic restrictions on device use and work with usage habits (gaining useful patterns and getting rid of harmful ones). It should be noted that some of these methods may involve the use of specialized software, and some do not require the installation of additional software in the form of applications.

Before considering each of the methods, it is necessary to determine the range of smartphones in question. We are talking about smartphones on the platform iOS and Android because these platforms together occupy more than 90% of the smartphone market

2.2. Existing smartphone settings from the OS

Consider the first method – device settings changes to reduce distractions. Distractions, the source of which is a smartphone, are one of the ways to interact with a smartphone, namely the notification mechanism and the user interface itself.

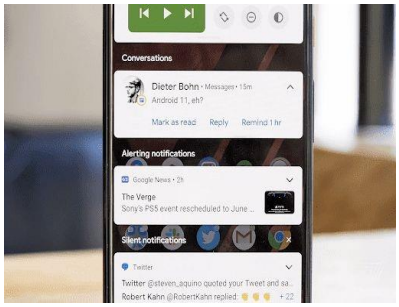


Figure 1: Notifications example on Android

Figure 2: Example of notification indicators on iOS

A notification is a sound signal and a vibrating alert about an event in an application. Each application can send messages at the right time. For example, it can be a notification about releasing a new video on your favourite channel on YouTube or information about a bank transaction after payment by card in the store. Notifications occur when the application decides. It is what creates a distraction or unwanted reminder. For example, a user is working at work and has granted permission to the YouTube application to notify him of new videos that may interest him. The YouTube application, in turn, does not take into account what the user is doing now and can send notifications when the user does not need this notification. When the information is triggered, the user does not know what application it is and is forced to be distracted because it can be an important notification. But the price of this distraction is 25 minutes. It takes precisely 25 minutes to return to work after a distraction, as determined by researchers at the University of Irvine [19]. Even before this study, operating systems set the so-called "Do Not Disturb" mode and complete control of permissions over notifications, which applications can send and which statements. Configuring these operating system features allows you to leave only the most critical messages. However, users often ignore this and allow all applications to send all kinds of notifications. In addition to notifications, the user interface itself has a reasonably high distraction. After all, in the user interface, especially on the iOS platform, you can see the number of missed notifications from the application [13]. They look like red round lights next to the app icon. It, in turn, gives signals to the brain that this is something important. When, for example, a user wants to route home in the navigation application, he may notice that one of the social networks has a missed notification. It may indicate a reaction from other users on the network to his message or post. Research says that the social reaction is a high priority for a person, and most likely the user will first view the reaction to his post on the social network and only then build a route [13]. Therefore, OS on both platforms can adjust colours and allow you to disable such indicators altogether. An intelligent way to reduce the interactivity of a smartphone is to put it into mode shades of grey. Users who have enabled this mode indicate a significant reduction in screen time.

2.3. Monitoring and control of indicators and temporary restriction of use

The screen time indicator and its derivatives, in the form of the number of interactions with the smartphone and the list of applications, are quantitative indicators that can be monitored. Surveillance can be carried out both with the help of special software and through planning outside the smartphone. There is special software that provides analytical data on-screen time in reports, charts and graphs. In addition, there is software that allows you to set usage limits, and after the limit reminds the user of the boundary or prohibits use. Some smartphone manufacturers, such as Google and Apple, have pre-installed monitoring and control applications. For example, Google has Digital Wellbeing on its smartphones, and Apple, in turn, has a Screen Time app on its iPhone.

The main functions of these two pre-installed applications are to provide the user with data about its use (screen time and several interactions) in the form of diagrams and historical data.

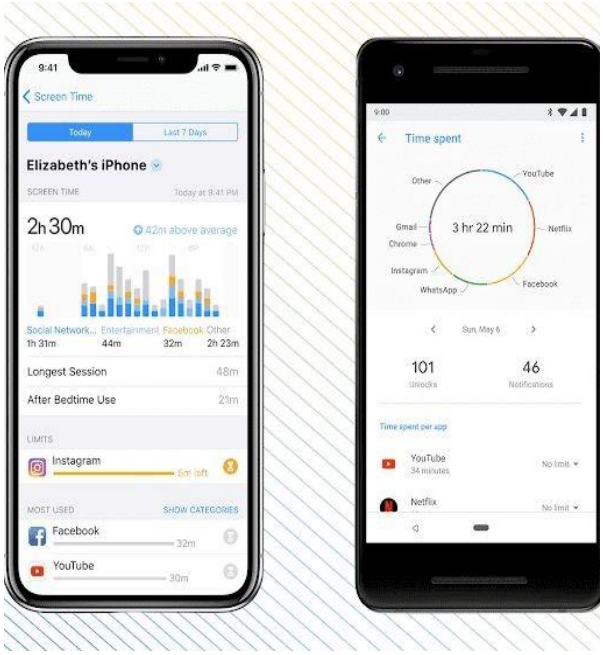


Figure 3: Apple Screen Time (left) and Google Digital Wellbeing (right)

Monitoring and control can be carried out without being based on quantitative indicators. The method of so-called digital detox is quite prevalent [21]. The essence of digital detoxification is to turn off all notifications and completely abandon devices and gadgets. This method has its results and works quite well for users who do not want to keep track of their screen time. Often users decide to perform a digital detox based on their feelings. Digital detoxification has a relatively positive effect, but it can be too radical for some users, as users often need to stay in touch. Of course, there is also special software that can help with this. For example, the well-known smartphone manufacturer OnePlus has pre-installed the so-called Zen Mode on its smartphones.

This mode disables all communications in smartphones for a short time, for a so-called digital detox. This kind of mobile application is not the know-how of One Plus because there are analogues. But when a manufacturer and OS developer manage such development, it has more authority than a third-party developer. But the presence of specialized software is not required for a digital detoxification.

2.4. Habits and patterns of use

Given the specifics of the human brain, a person adapts to a specific type of activity and performs it as a habit. It is exactly what happens when using a smartphone [16]. Usage habits create most behaviour patterns with a smartphone. These habits a person categorizes as beneficial and harmful, based on their feelings. But, users want to get rid of certain routines, and some, on the contrary, want to get. For example, many people experience the phenomenon of procrastination. That is, postponing the case for later. Historically, people have frequently been procrastinating, but recently a smartphone has been present in our procrastination. After all, the smartphone has entertainment that can be more interactive than, for example, paying bills. So many people want to get rid of procrastination on a smartphone. There are many exciting ways to get rid of this. For example, do not take your phone to work or use the Forest application. It is an application that helps to develop the habit of not taking your smartphone while working. The application offers to set a timer, which metaphorically means a planted tree. If the user exits the application by the end of the set timer, the tree will not grow. This simple metaphor is very effective, and it helps many Forest users [28].

Accordingly, the opposite example of a habit you want to get rid of Studies shows that users often use their phone before bed, which can affect sleep health [22]. So you cannot take the phone in bed and use a mechanical or electronic alarm clock. Or you can use the option of pre-installed applications from Google Digital Wellbeing and Apple Screen Time [25, 26]. Namely, Bedtime mode. For this mode, the

user selects the desired hours of sleep, and during these hours, the application disable all notifications and can also remind of this mode and make the screen black and white.

2.5. Specialized software for smartphone usage control

Methods of controlling the use of a smartphone were considered, some of which do not require additional software, and some include the use of other software. One of the aims of this study is to use specialized software for such needs. So it is worth considering this kind of software and compares existing solutions in more detail. It is necessary to determine the parameters of software comparison by which the comparative analysis will be performed. A small study of existing comparisons of software products in the categories of performance and screen time monitoring was conducted to select the parameters of comparisons.

Based on this study, applications will be compared according to the following parameters:

- Availability on platforms;
- What method of control is implemented;
- Basic functions;
- Additional features;
- Privacy and user data transactions;
- Usage price or monetization;
- Relative efficiency (based on feedback);
- Application design;
- The number of installations and user rating.

Applications to be considered:

- Google Digital Wellbeing [25];
- Apple Screen Time [26];
- Torque [27];
- Forest [28];
- Before launcher [29].

2.5.1. Google Digital Wellbeing

In August 2018, Google released the Digital Wellbeing application, which is exclusively pre-installed on Pixel smartphones sold by Google. The primary function of the application is to provide the user with detailed reports on his screen time. Google is not the first company to make such an application on Android. For example, in 2017, the StayFree application was released by developers from Turkey. But the reference function whose primary function is screen time monitoring is Digital Wellbeing because Google is the owner of the Android OS and has more authority to develop applications that can be installed on phones by default.

The application has many additional features, such as limiting the use of specific applications (Fig. 4), "concentration mode", and released in 2020 "bedtime". Limiting the time of use of a particular application is implemented as follows. The user in the Digital Wellbeing settings selects the application by how the application can be any, i.e. even the Camera or Phone and sets the time limit. After that, the application in the background counts the amount of time spent in the application and, upon the occurrence of the limit set by the user, notifies the user about exceeding the set limit. It is also possible to configure parental controls (control the child's device from the parent device).

The "focus mode" function works as follows. First, the user determines when he wants to enable the mode and selects a list of applications to which access will be blocked. The mode can be enabled on-demand or about the hours selected by the user, during the operation of the method, when opening applications from the list specified by the user, the Digital Wellbeing application reminds the user that the concentration mode is now and will allow you to enter the application urgently for a limited period of 5 minutes. The sleep timer feature was created to prevent the phone from being used at night and in the evening. It works as follows the user selects the desired hours of sleep, and during these hours, the Digital Wellbeing application will turn on the no disturb and change the screen mode to black and white.

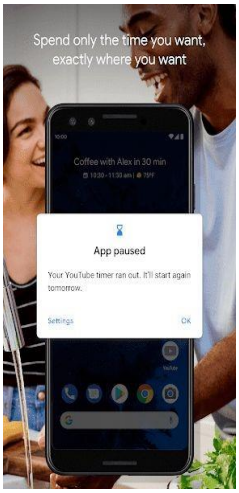


Figure 4: Operation of the limiting function

As mentioned above, the Digital Wellbeing application has the advantage that it is developed by the operating system's vendor, which provides additional development capabilities.

2.5.2. Apple Screen Time

Earlier than Google, Apple, together with the release of the first beta of iOS 12 in June 2018, releases a new standard application, Screen Time, which is part of the operating system (part of the settings). After the release of Screen Time, the problem of high screen time attracted the attention of the world community. After all, the application is pre-installed on all iPhones that have iOS at least 12 versions. The application's primary purpose is to provide users with data on their screen time and smartphone usage. Of the additional features, Apple Screen Time has a similar function as "sleep time" in Digital Wellbeing, called "Downtime", the ability to set limits on the use of specific applications and the process of "parental control". In contrast to Digital Wellbeing's "sleep time" the Downtime feature includes a do not disturb mode and allows notifications from the selected list of applications that the user has previously established for this model. But the limiting function and the "parental control" function have a slightly more straightforward configuration interface than in the case of Google Digital Wellbeing.

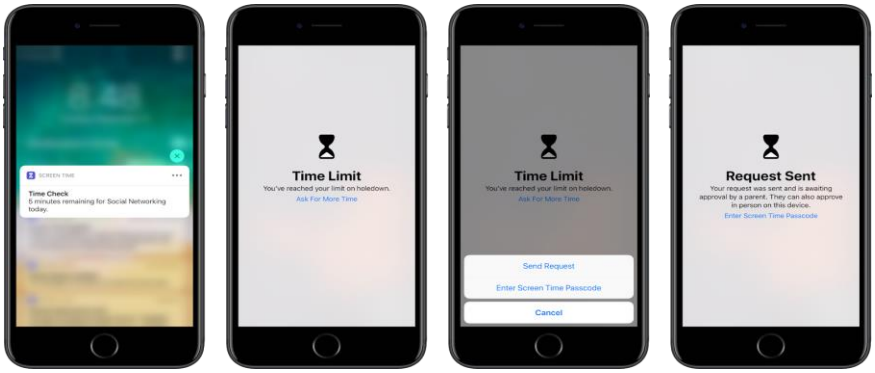


Figure 5: Interface of the parental control function from Apple Screen Time

2.5.3. Moment for iOS

Moment app for iOS and Moment Health Inc. from California, USA, were among the first to develop applications for screen time monitoring. And to be precise, this application was made by Covin Holesh, developer and designer. The moment was created much earlier than Apple Screen Time and Google Digital Wellbeing, namely in 2014 (i.e. even earlier than analogues on Android).

The primary function of Moment is to monitor and report the user's screen time. Thus, the Moment application can be called a kind of pioneer of screen time monitoring applications.

In addition to the monitoring function, Moment has many additional parts that Apple Screen Time and Google Digital Wellbeing do not even have. But this functionality is paid. The moment opens up other features for $ 4.99. These include daily limits on screen time, notifications of usage limits, an analogue of Google Digital Wellbeing's "concentration mode" - "Screen-Free time", and the ability to collaborate with friends or family. The ability to collaborate was a prevalent feature that helped to collectively monitor screen time, which increased the effectiveness of such monitoring.

Figure 6: Moment application interface for iOS

But one of the advantages of the Moment application is the presence of so-called coaching. It is an opportunity to purchase a training course, which in detail acts as an assistant in the application over time, which helps to acquire or get rid of a specific habit. In addition to classes, Moment has tips and advice on using a smartphone. There are free, and there are paid courses. The average price of the methods is $ 9.99. In addition to coaching, a promising feature turns the reduction of screen time into a collaboration between colleagues or family members, but this feature is also paid. We can say that Moment implements a hybrid model, i.e. monitoring and limiting and working with habits.

2.5.4. Forest

The Forest application belongs to a group of applications that focus on forming usage habits or the deprivation of usage habits. The Forest application is a product of the Taiwanese company Seekrtech. Like Moment, it specializes in health software. The Forest application has already been mentioned as an example. Still, it is worth considering it in more detail because Forest users report a reduction in screen time and improved productivity at work. The reason for this is that the Forest application implements the Pomodoro method, i.e. the method of interval work [23]. Due to the successful metaphor and visual aspect, the application is considered one of the benchmarks of performance applications. And all because of its simplicity and efficiency.

Figure 7: Forest application interface

The primary function is to set timers during which an imaginary tree grows in the application. If you start using a smartphone (exiting the application) before the timer ends, the mythical tree in the application will not succeed. Another application has a statistics function implemented in the garden format, which is created from these timers. Due to the implementation of the so-called gamification (English gamification) in the application are many free analogues. Still, thanks to the exceptional design and elements of the game, the Forest application is unique.

2.5.5. Before launcher

The latest application focuses on personalizing the Android operating system's interface to reduce screen time. A small group of enthusiasts Before Labs from Oregon, USA exclusively for Android, developed the application. The application's primary purpose is to replace the standard launcher of the Android operating system with a launcher, which is designed to be convenient and less distracting the user from his goal. Studies show that the system interface has a strong influence on user behaviour [13]. Therefore, Before the launcher changes the interface of the Android launcher so that it leaves only the names of applications on the desktop and a list with a search in which only the names of applications without icons will be displayed, in addition, with the pro version, the user can change the terms of applications, group them, hide applications from the list. An additional feature is the notification filter. Swipe left Before the launcher opens a window where you store filtered notifications from applications that the user has chosen to filter.

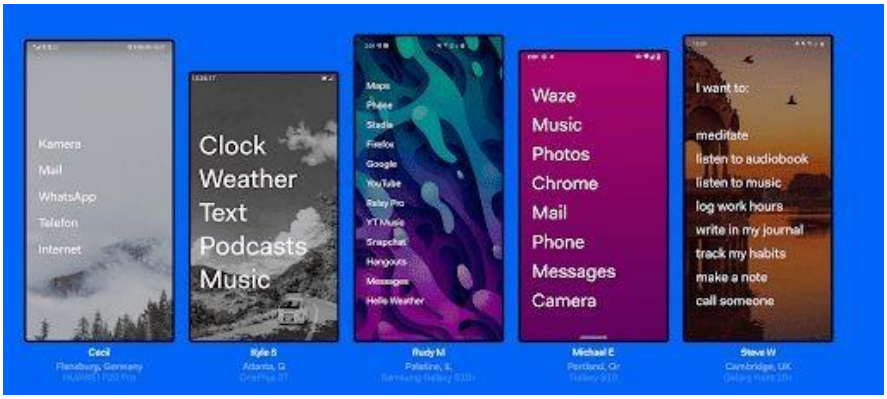


Figure 8: Interfaces are made using Before Launcher

Working on the visual aspect of the Before launcher allows you to reduce the impact of bright interfaces. And thanks to the notification filter, it will enable you to ignore unnecessary notifications without excluding them and in addition. Still, it is also essential to store filtered reports in one place.

Before, labs are not the first company to make and work for us minimalist launchers for Android. But it was they who managed to find a balance and leave what was necessary.

Before the launcher confirms the importance of the user interface and its impact on screen time and the habits that this interface creates.

2.5.6. Comparison of existing software

The results of the comparison of the predefined parameters are shown in the table.

Table 2

Comparison of existing applications to control the use of a smartphone

Comparison parameter	Google Digital Wellbeing	Apple Screen Time	Moment	Before launcher	Forest
Platform	Android	iOS iPadOS macOS	iOS	Android	iOS Android

Implemented method	limitation monitoring	limitation monitoring	habit limitation monitoring	habits	habits
Basic functions	Display on-screen time reports and limit application usage time	Display on-screen time reports and limit application usage time	Display on-screen time reports and limit application usage time	Personalization to reduce distractions from the interface	Set a timer and block access to the phone during the timer
Additional features	Parental Controls, Concentration Mode, Sleep Time	Parental Control, Downtime	"Screen free time", collaboration with friends or family, coaching	Notification filter	Interface statistics and customization
Privacy	Collects usage data	Collects usage data	collects personal data provided during registration and anonymized screen time data	collects anonymized analysis of the use of application functions.	collects personal data of users at registration and collects anonymized usage data
Price	Free	Free	Free Contains paid features from $ 4.99-$ 14.99	in-app purchases, namely the Pro version for $ 4.99	Free Android (with advertising) $ 1.99 iOS + in-app purchases
Relative efficiency	Users report a reduction in screen time almost immediately after setting application usage limits.	Users report a decrease in screen time almost immediately after setting application usage limits.	Thanks to the hybrid model, the application is effective not only in short term.	The user also reports a reduction in screen time over prolonged use.	Effective in the long run if you follow interval approaches to work.
Design	The design is made in the style of Material Design from Google.	The design is made in the style of iOS.	The design departs from the Apple Design Guidelines and is exceptional	Focused on minimalism	The design is similar to the game.
Number of installations and reviews	250,000 reviews	500 000 000	14,000 reviews 1,000,000 downloads	5,000 reviews 100,000 on one platform	250,000 reviews 10,000,000 on one platform

This section discusses methods and approaches to solving such problems, using special software and processes that do not require additional software. A comparative analysis of applications that solve the described problem was performed. It can be concluded that the availability and effectiveness of applications depend on the platform. It is possible to notice that the companies owning platforms that is Google and Apple also have the applications as a part of OS that means that the companies invest in an existing problem and urgency of a problem is high. But on both platforms, some applications implement one or another approach to solving the problem.

3. Material and methods

3.1. System analysis of the object of study

After reviewing the existing ways and methods of solving similar problems with screen time and excessive use of smartphones at work or in everyday life, a systematic analysis of study object and area subject was conducted.

3.1.1. The general purpose of operation

To comprehensively outline the essence of the system, it is recommended to describe the general purpose of the system and apply the goal tree method to specify the general sense of the system and conduct a technique of hierarchy analysis to determine the type of system, which can then be used for further decisions in system architecture. The general purpose of functioning is to create an information system to help users develop a habit of planning their use of the smartphone just before using it.

The tree of goals in Fig. 9 has three levels: the general-purpose, aspects of the general-purpose, their sub-aspects, and the criteria defining the system's quality of functioning.

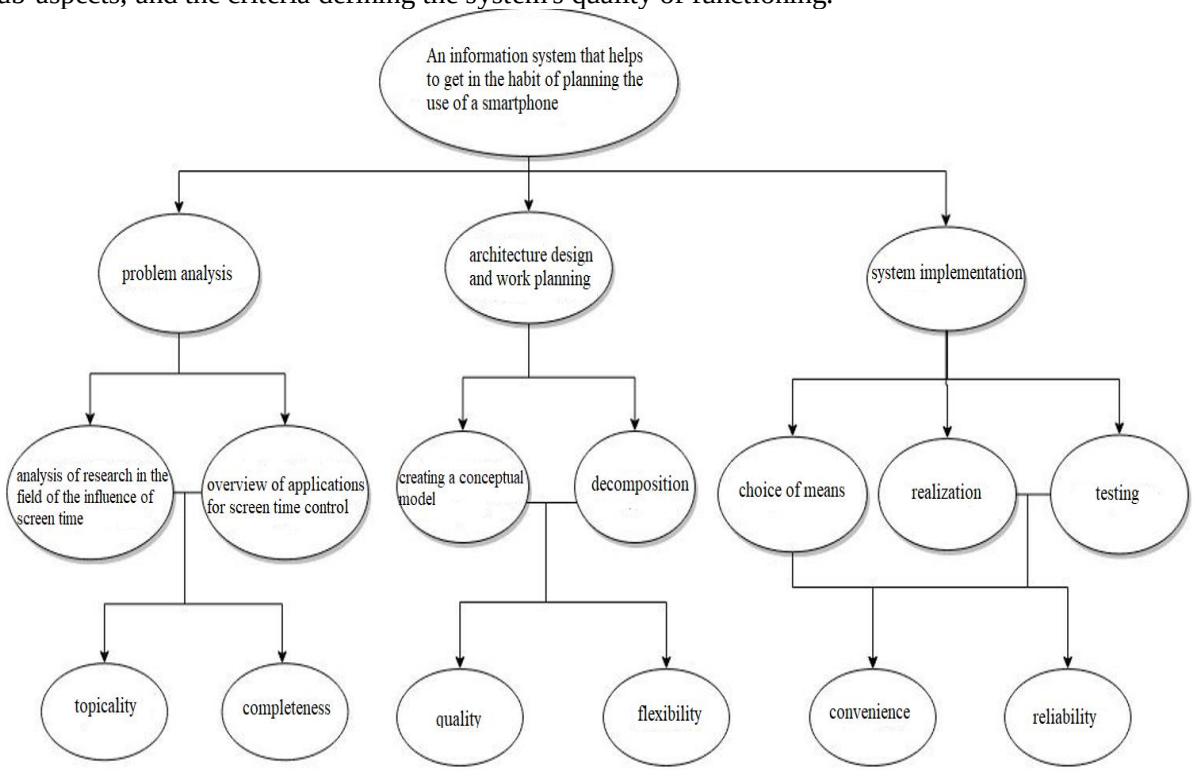


Figure 9: The goal tree

The main aspects of the general goal:

- Problem analysis.
- Architecture design.
- System implementation.

The problem analysis is a detailed study of existing software and research related to screening time and ways to interact with a smartphone. It is necessary to consider each aspect in more detail. The correct analysis of existing solutions and research can provide useful information that can be used in future decisions that will already relate directly to the system. Sub-aspects of the "Problem Analysis" aspect: analysis of research in the field of screen time influence, review of applications for screen time control. The criteria of the first aspect are relevance and completeness. Relevance is an important criterion because the development of performance applications involves creating applications for current methods to increase productivity and reduce screen time. No less critical parameter is completeness because it is a completeness that determines the extent of the developed system.

The next aspect is the design of architecture and works planning. This aspect is essential because it is during the application architecture design that important decisions are made, which then affect future development. And with proper work planning, you can avoid deadlocks during development, which arise due to incorrectly constructed dependencies on the problem and the unfair distribution of time.

It is divided into three sub-aspects: choice of tools, implementation, testing. Sub-aspects of this aspect are the decomposition of problems and the creation of a conceptual model. The criteria for this aspect are flexibility and quality. The last aspect is the implementation of the application, i.e. the system. The criteria of the latter aspect are reliability and convenience. There are the following types of systems:

- Information retrieval;
- Information reference;
- Information managers;
- Intelligent systems;
- Decision-making systems.

There are the following types of alternatives for the developed system:

- Information reference system (A1);
- Intelligent system (A2);
- Information control system (A3).

Accordingly, there are criteria for system quality:

- Relevance (K1);
- Completeness (K2);
- Quality (K3);
- Flexibility (K4);
- Ease of use (K5);
- Reliability (K6).

The following steps are to build a matrix of pairwise comparisons of criteria. And the calculation of eigenvalues (HF) and vectors (HV). Expert assessment is made using the relative importance scale of priorities from 1 to 9 (according to the scale of pairwise comparisons of T. Saaty).

Table 3

Matrix of pairwise comparisons of criteria

	K1	K2	K3	K4	K5	K6	HF	HV
K1	1	3	1	1/6	1/7	1/7	0.46	0.06
K2	1/3	1	1/4	1	1/6	1/4	0.38	0.05
K3	1	4	1	4	1	4	2	0.26
K4	6	1	1/4	1	1/6	4	1	0.13
K5	7	6	1	6	1	2	2.8	0.36
K6	7	4	1/4	1/4	1/2	1	0.97	0.12

To calculate the eigenvalues (HF), use the following formula:

$$HF=\sqrt[n]{\prod_{j=1}^{n} a_{ij}} \quad (1)$$

The following steps are to build matrices of pairwise comparisons of alternatives for each selected criterion and the primary goal.

Table 4
Matrix of pairwise comparisons of alternatives for the criterions

Relevance	A1	A2	A3	HF	HV
A1	1	6	1/6	1.0	0.23
A2	1/6	1	1/4	0.34	0.08
A3	6	4	1	2.88	0.68
Completeness	A1	A2	A3	HF	HV
A1	1	1/5	1/2	0.46	0.12
A2	5	1	1/5	1	0.27
A3	2	5	1	2.12	0.59
Quality	A1	A2	A3	HF	HV
A1	1	1/7	1/3	0.36	0.09
A2	7	1	1/3	1.32	0.35
A3	3	3	1	2.08	0.55
Flexibility	A1	A2	A3	HF	HV
A1	1	1/4	1/5	0.36	0.09
A2	4	1	1/3	1.10	0.27
A3	5	3	1	2.46	0.62
Ease of use	A1	A2	A3	HF	HV
A1	1	2	1/3	0.87	0.23
A2	1/2	1	1/4	0.5	0.13
A3	3	4	1	2.28	0.62
Reliability	A1	A2	A3	HF	HV
A1	1	2	1/3	0.87	0.24
A2	1/2	1	1/3	0.55	0.15
A3	3	3	1	2.08	0.59
Main goal	A1	A2	A3	HF	HV
A1	1	3	3	2.08	0.59
A2	1/3	1	1/2	0.55	0.15
A3	1/3	2	1	0.87	0.24

The last step is to build a matrix of comparisons of alternatives. Its result determines what type of information system is needed for implementation.

Table 5
Generalized priorities

Criteria	К1	К2	К3	К4	К5	К6	Generalized priorities
	0.06	0.05	0.26	0.13	0.36	0.12	
A1	0.23	0.12	0.09	0.09	0.23	0.24	0.16
A2	0.08	0.27	0.35	0.27	0.13	0.15	0.20
A3	0.68	0.59	0.55	0.62	0.62	0.59	0.51

After analysing the hierarchies, the following results were obtained:

- Information search system - 0.16.
- Intelligent system - 0.20.
- Information management system - 0.51.

Therefore, this system is an information management system.

3.1.2. System quality criteria

Quality criteria must be defined in such a way that they can be measured. But the added complexity of this system is that its quality criteria are subjective and, like the quality criteria of implemented systems, are difficult to calculate for performance or health. For example, reducing screen time does not guarantee system quality, and increasing screen time does not guarantee system inefficiency. The best criterion for the quality of the newly created system may be to reduce the number of cases when the user was distracted from the goal he set before using the smartphone, but this metric is difficult to calculate. Therefore, for simplicity, it is necessary to choose as a criterion of quality changes in screen time and user interaction with the system (how much the user follows the new method of using a smartphone).

3.1.3. Options for achieving the goal

There are several options for achieving the goal based on the resources and limitations of the platforms for which the system is designed. One of the options to achieve the goal is implementing a simple application through which the user would use as a scheduler of screen time and actions in the smartphone. But this option has a significant disadvantage, which is the complexity of use. After all, the main idea is to use an additional level or a step before you start using a smartphone. And forcing the user to go to the application every time you use a smartphone is not a good practice and a good idea in general. Another way to achieve the goal is to use the operating system to show an element of the application or the application itself immediately after unlocking the smartphone. The second option to achieve the goal seems acceptable, but it is worth considering the limitations of the OS.

Given that Apple has specific restrictions on opening applications and pop-ups in the system and restrictions on monitoring the events of unlocking the smartphone on Apple, it is necessary to implement another option with the application. But in the latest release of iOS 14, Apple announced desktop widgets that are ideal for this purpose. On Android, it is possible to show pop-ups immediately after unlocking. Given the limited number of options to achieve the goal and limit the OS, both options are implemented depending on the OS.

3.1.4. Limits of system functioning

The user himself plays an essential role in the proposed system. However, there are always difficulties in defining boundaries in health software development, as systems can only notify and provide a tool. Therefore, the user is forced to make decisions or follow recommendations and whether the indicators monitored by the system are acceptable to him. That is, the boundaries of functioning do not go beyond the assistant. The system only provides an interface that reminds the user of a slightly different way of using the smartphone and monitors its progress. The user decides how much this method offered by the system costs him.

3.1.5. Decomposition criteria

The new system is easily decomposed according to the criteria of the tasks formed from the tree of goals. When decomposing the system into elements, it is necessary to do the decomposition so that the system's components are as independent as possible and have a clear way of communication and interaction. Decomposition is relatively primitive:

- An element of the system responsible for assisting the user in setting goals immediately before using the smartphone;
- The system element that is responsible for historical data, i.e. the user's previous plans and their execution;
- Data visualization and representation element;
- Reminder item;

- Settings item;
- Element of education.

3.1.6. Data flow in the system

Given the simplicity of the goals in the system will be a primitive data flow, which the user must transparently manage. After all, the purpose of using a smartphone is the user's private information. The user's natural language is written in any form, which plays the role of a plan of use should be stored on the user's device and not transferred to third-party resources or stored elsewhere than on the user's device because one of the purposes is privacy and transparency. Only the application and the user have access to the data.

3.1.7. System analysis of the object of study and subject area

Given the existing software in the form of applications were discussed in section 2. It can be noted that it partly focuses on monitoring screen time and neighbouring indicators, and part focuses on the aspect of time that the user spends outside the smartphone. The existing systems are more focused on time outside the smartphone and create an association of "more = worse". But current applications do not cover the element of conscious and intentional use. The screen time indicator is conditionally seconds, but what actions the user does during this time is not covered by the screen time and the timers that need to be set to "lock" the smartphone. Monitoring of screen time, its limitation and setting interval timers for not using a smartphone during them do not consider the qualitative aspect of estimating the time spent behind the screen. It is the key to some users. Because for some users, the association of longer screen time does not mean worse time spent. And it is this shortcoming that the developed system is trying to correct by introducing an additional indicator and a slightly modified way of interacting with the smartphone, which is to plan the actions that the user will perform while using the smartphone. And this planning in the developed system takes place just before using the smartphone. So the purpose of the system is to provide the user with a way to qualitatively measure and plan their screen time, which existing analogues do not currently offer.

The proposed system should slightly modify, possibly complicating the existing way of using a smartphone. For example, when a user wants to use a smartphone function, he does not formalize or record his intention in any place (only in his head). And given the number of time users spend using a smartphone, it can't answer how many times a day it has been distracted from its original purpose because most of our interactions with a smartphone are based on habits.

The ideal place and time for planning is the time after unlocking the smartphone. At this point, the user has already started using the smartphone and intends to perform a list of actions. It is worth noting that at this point, the system should start functioning. To better understand all aspects and nuances of the developed product, it is necessary to consider it a system. To do this, use the methods of system analysis [30] and warehouse development [31-35]. Its essence is to consider the idea and the product as a system consisting of elements that interact with each other. To perform system analysis, it is necessary to define how the principles of the system are implemented in this project.

The principle of the ultimate goal is defined by defining the overall purpose and constructing a goal tree. Finally, the general sense is formed and will help to shape the habit of smartphone users.

Principles of unity consider the system as a whole and as a set of elements of the system. The habit-building application has only a few aspects of the system that follow a general goal. The aspect of assistance is intended to modify the smartphone user's experience. The aspect of notifications is intended to remind the user of his purpose. The element of analytics and reporting should show the user his progress. Each element has separate functions, but the general position is not inherent in any different part. The principle of connectivity indicates considering each part of the system along with the connections from its environment. For example, the analytics element is directly dependent on the assistance element, and the notification element is directly dependent on the analytics element. That is, the elements of the system are connected and exchange data or signals during the operation of the system. The principle of modular construction means the possibility of decomposition into separate

modules that can be used separately. Therefore, the system is divided into modules by elements. The system can only function with the assistance module, the analytics module and the notifications module can be used at the user's discretion.

The principle of hierarchy requires the system to have a hierarchical structure. That is, the elements of the system must be arranged in order of importance. Thus, for example, assistance is the essential element of the system, followed by analytics and the last component of notifications.

The principle of functionality is based on the principle of hierarchy, and the functional requirements for the system were built based on the built order. The principle of development indicates the need to consider the variability and expansion of the system, the possibility of growth or replacement of parts, the accumulation of information. This principle is implemented in the system by minimizing the responsibility of the elements of the system. Each component is responsible for one action. The system can be expanded simply by adding new elements. The principle of decentralization in the management of the system, the relationship between centralization and decentralization, is determined by the purpose and purpose of the system. The system tends to decentralize, but in the first implementation, the system is centralized, which gives its advantages. On the other hand, working with data in the system is isolated and can easily change or expand in the future. The system maintains a balance between centralization and decentralization through clear communication of elements and the absence of the core system.

The principle of uncertainty and randomness must be considered when determining the strategy and tactics of system development. By implementing a transparent process of working with data and a reasonably simple interface in indefinite and random leave, the system can change flexibly because its functions are essential in implementation.

3.2. Statement and substantiation of the problem

3.2.1. The purpose of development

They are analysing existing applications for screen time control, habit formation, productivity improvement. But after lengthy personal testing and based on market needs, which has been analysed based on the primary analysis of the latest literature, namely the work of Kel Newport [18]. It is evident that existing applications do not cover the planning and control of conscious and targeted use. Of course, existing applications are practical, but after personal experiments with minimal prototypes, it was found that combining all methods is much more effective. And the purpose of the development is to develop a simple application to increase productivity by scheduling sessions [36-38], which will be self-sufficient and an excellent additional component for screen time and performance monitoring systems that users already use. In addition, the specifics of existing applications were taken into account, namely the principles of data handling and data privacy and security. Most current applications abuse mid-application analytics and the collection of users' data, and the newly created system should cover the aspect of data privacy and security [39-50]. Finally, it is worth noting that the purpose of the development is not commercial but even more charitable, to create an accessible tool to control and draw attention to the abuse of smartphones and high screen time.

3.2.2. Description of purpose

The app system is intended for users interested in scheduling their usage time and users interested in reducing screen time [51-62], and users who wish to experiment with their screen time to determine their habits and analyse distractions while using a smartphone.

3.2.3. Place of application

The application to increase the smartphone's performance or reduce the screen time has a prominent application like most applications to improve performance. The central location is the scripts and places of use of ordinary users. Preferably it is either working or non-working time depending on the goals of the user and the nature of his activities.

3.2.4. The rationale for the development and implementation of the system

The problem of abusing a smartphone is relatively new, as the technology has emerged and spread relatively recently. Research confirms a direct link between mental health and the abuse of certain types of applications, and some statistics show that screen time is on the rise. The spread of the latest Internet networks and the high popularity of social networks and online entertainment services do not pose such a threat. Still, when abused, they hurt users of all ages. Using the smartphone's capabilities, there is an opportunity to slightly modify the user experience and thus draw his attention to potential problems that the user may overlook. Since the end of 2018, there has been some impression of relatively large products of corporations such as Facebook and Google, and it reads as follows: "If you do not pay for the service or product then you - the user is the product" and this impression is not unreasonable [13]. Contextual advertising, the creation of personalized news feeds that can influence user behaviour, all these events, and the modern design of some services are aimed at the user's attention. These factors mean that some software is created to distract the user. And given the relatively small number of applications, performance techniques that help control this impact, the development of this application system is entirely justified, even for charitable purposes.

3.2.5. Expected effects from the implementation of the system

Several results are expected from this system:

- The result of drawing attention to the issue;
- The impact of applying the method to users;
- The effect of motivating users to acknowledge the problem with screen time.

The effect of attracting attention is the application's ability to show obvious cases of user distraction when using a smartphone for secondary purposes, which are provoked by notifications from other applications or the interface of the operating system. Due to such a simple demonstration, this application can be easily used for experimental purposes for research. It is also attracting the attention of researchers and developers, and designers who can design slightly different user interfaces that will be more focused on creating a productive environment. Applying the method to users is to change the indicators of screen time and relative productivity of the user during operation. This effect directly depends on user feedback. The effect of motivating users to acknowledge the problem with screen time refers to users who did not recognize or did not think about high screen time. This type of user should get an answer to the question, "Am I distracted by non-priority things or things I didn't plan to do during the usage session?"

3.3. Development of a conceptual model of the system

Before you develop a system, you should create and specify the conceptual model of the system. The conceptual model should closely intersect with the requirements of the system. First, it is worth noting the vital root requirements for the system. The plan is to be implemented in the form of a mobile application (application), which must be available for both Android [63-64] and iOS [65]. The functions of the system must be implemented based on the limitations of the OS. That is, it is assumed that applications for different operating systems may differ. But in addition to this technical point, it should be noted that this system can be implemented without a specialized application. The system can exist without an application and can be implemented by alternative methods on other platforms. One of the plan's other goals is to disseminate the idea, so it is mandatory to create a module that explains and gives the concept behind the application.

3.3.1. Description of system functions and structure

One way to describe the functions of the system is user history [24]. User stories quite closely intersect with the needs of the user.

If the user stories are formed correctly, the developed system has fewer chances of fulfilling users' expectations.

1. As a user, I want to control when I use a new approach to using the system and when not in mode. As an example, flight mode.

2. As a user, I want to write down my goal of unlocking the smartphone immediately after opening it or after clicking on the appropriate user interface element.

2.1. As a user, I want to skip recording the purpose of unlocking the smartphone.

2.2. As a user, I want to view the time of missed fillings of unlocking goals.

2.3. As a user, I want to create a reminder about the original unlock goal and select a time interval when this reminder will be received.

2.4. As a user, I want my entries or record skips to have start timestamp and end of session timestamp (phone lock).

3. As a user, I want my entries to be logged.

3.1. As a user, I want my recordings to be stored exclusively on my device and not passed on to third parties.

4. As a user, I want to be able to view my records as magazine.

4.1. As a user, I want to be able to filter records by date.

4.2. As a user, I want to be able to update the bog entry.

4.3. As a user, I want to be able to view the timestamps of records.

4.4. As a user, I want to be able to delete entries.

4.5. As a user, I want to be able to mark records as successful and unsuccessful.

5. As a user, I want to have visual reports about my records or their absence.

6. As a user, I want to be able to disable skipping goal entry.

7. As a user, I want not to set a timer for unlocking purposes.

8. As a user, I want to get an answer to a question for frequent answers.

9. As a user, I want to know more about the project.

10. As a user, I want to leave a comment or share my experience.

3.3.2. Input and output data of the system and the main essence of the system

To implement the application, you should clearly describe the input and output data of the system. Regarding the user's requirements, the user's data is the text of any format in any language and optionally adds the desired time that he would like to use for the session. All input data comes into the system at the time of session planning using. You can use the XSD Schema to write input to the system, which you can use to describe the XML schema of a document using the Salami Slice method

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
 <xs:element name="name" type="xs:string"/>
 <xs:element name="estimatedTime" type="xs:short"/>
 <xs:element name="createdAt" type="xs:int"/>
 <xs:element name="root">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="name"/>
    <xs:element ref="estimatedTime"/>
    <xs:element ref="createdAt"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
</xs:schema>
```

Figure 10: Example figure

During a user's session, information about its purpose can be updated, namely by updating what is caused directly by the user (editing the text itself) and updating the system (adding the end time of the session or adding a label as arbitrary text). Subsequently, the system shows the user this data aggregated over some time and visualized in a log or graphs. Assuming that the schema will be stored in a relational database, we can describe the table's appearance in which the data will be written and, accordingly, as rows that will be the source data of the system.

```
CREATE TABLE `Action` (
                `name` TEXT NOT NULL,
                `id` BIGINT NOT NULL AUTO_INCREMENT,
                `estimatedTime` INT DEFAULT NULL,
                `createdAt` TIMESTAMP,
                `endAt` TIMESTAMP,
                `tag` TEXT );
```

Figure 11: Example figure

The main essence of the system is the intention or purpose that the user writes down.

3.3.3. Functional requirements

Functional requirements describe the actions of the system both inside and outside the system, i.e. what the system should do, in contrast to non-functional requirements, which describe what the system should be [24]. Thus, functional requirements intersect closely with user histories, but operational requirements are more specific, and they complement user histories with technical details.

The functional requirements for the application are as follows [66-68]:

1. Regardless of the OS, user input data in the arbitrary text must be stored in a database stored on the user's device.
2. Each time a user requests data about his session records, the application must make the necessary queries to the database and translate the response to the database into an internal type or class that will be described in a programming language.
3. When setting the timer by the user using the software interfaces of the OS, the application must schedule notifications through the notification scheduler for the time selected by the user.
4. The application should change the method of use only after explicit confirmation by the user to activate the mode.
5. If the user does not fill in the unlock goal (this is possible due to OS restrictions), the system recognizes how the user avoided the record and writes it to a record that does not contain the goal. That is, the system does not differentiate filled and unfilled sessions into different types.
6. If the user wants to add a label to the session, the system saves the label in a text box.
7. Specific requirements for iOS: clicking on the widget redirects to filling out the session plan.
8. Specific requirements for Android. After each unlocks of the smartphone, the application shows its interface in the foreground.

3.3.4. Non-functional requirements

Non-functional requirements describe the attributes of the system and not its actions.

Non-functional requirements are as follows:

1. A mini-portal should be developed with static information about the application in links to Google Play and App Store and information about the idea, frequently asked questions, and links to support the project. The developed site must have usage analytics connected.
2. The application must be available for download on the appropriate platforms Android - Google Play, iOS - App Store.
3. The minimum version of Android for the application on Android 7.0.
4. The minimum version of iOS for the application on iOS 11.0.
5. The size of the application should not exceed 10 MB.

6. The application should work without the Internet.

7. The application should use push notifications with options.

8. Specific requirements for iOS. The application should use the new features of iOS 14, namely Widgets.

9. Specific requirements for Android.

9. 1. The application should notify the operator of the mode that changes the Android Foreground Service method.

9.2. The application must use the Notification Tile for quick access

10. User interface requirements.

10. 1. The user interface should follow the platform recommendation in iOS-iOS Design Guidelines in the case of Android-Android Material design guidelines.

10.2. The interface should have the form of filling in the purpose of use which should contain the field of input of the goal, an interval of the reminder, the Save button and the Dismiss button.

Initial state, showing today's sessions. 10. 3. The application should contain a page representing the session log. The representation must be made in the style of a list. Each item in the list should contain the text of the goal, the date of creation, the scheduled time (if any) and the actual time spent on the session. The page should have a date filter that opens the filter date selection window, and after selecting the date, the log shows the sessions for the selected date.

10.4. Android. The application should have a page with settings and valuable links about the project. The page should contain two settings for disabling notifications and timer and disabling Dismiss button.

10.5. iOS.

10.5.1. The application should have a page with useful links to information about the project.

10.5.2. The application must have a page that reflects the current action that the user added last and the amount of time left for the selected time. The initial state of the page when there is no recent activity, i.e. if the user has completed it, the page displays a button that opens the form for creating an action.

10.5.3. The widget interface should contain text that reminds you of the purpose of unlocking and a button that opens the action creation form.

10.6. The application's website should contain the main screenshot of the application, namely the form of filling in the goal, a link to the application in the relevant application stores, a page of frequently asked questions, a page with a description of the project.

3.3.5. Uses and usage diagram

Given the specific purpose of the application and its desire to implement one method of using a smartphone, there are not many options for using the application.

The primary use case. The user who is interested in changing his method of using the smartphone includes an endless mode in which the interface changes after unlocking the smartphone, and its subsequent use will require filling the goal using the form of serving purposes or ignoring loading the goal using system navigation or a particular button, which will then be recorded in the activity log depending on how the user missed the goal completion. This user can when he needs to analyse his sessions using the application interface to view and explore their sessions. Alternative uses:

- An employee who decides to use the app to increase productivity turns on the app mode when he needs it. During his work, in case of lifting the smartphone, he analyses his goal and records it and continues using it if necessary. Otherwise, the user locks the smartphone and continues working. At the end of the work session, the user can turn off the mode.
- A researcher who has decided to conduct a study concerning the purposes of use or screen time provides a smartphone for the crucial period with the mode of the changed method of use already enabled, which he has previously activated. Alternatively, it prompts users to turn on the way at the appropriate time.

3.3.6. Primary sequences in the system and series diagram

The application has several direct lines and cycles, namely:

- Change the interface and record a reminder;
- View and analyse records;
- The process of ending a usage session.

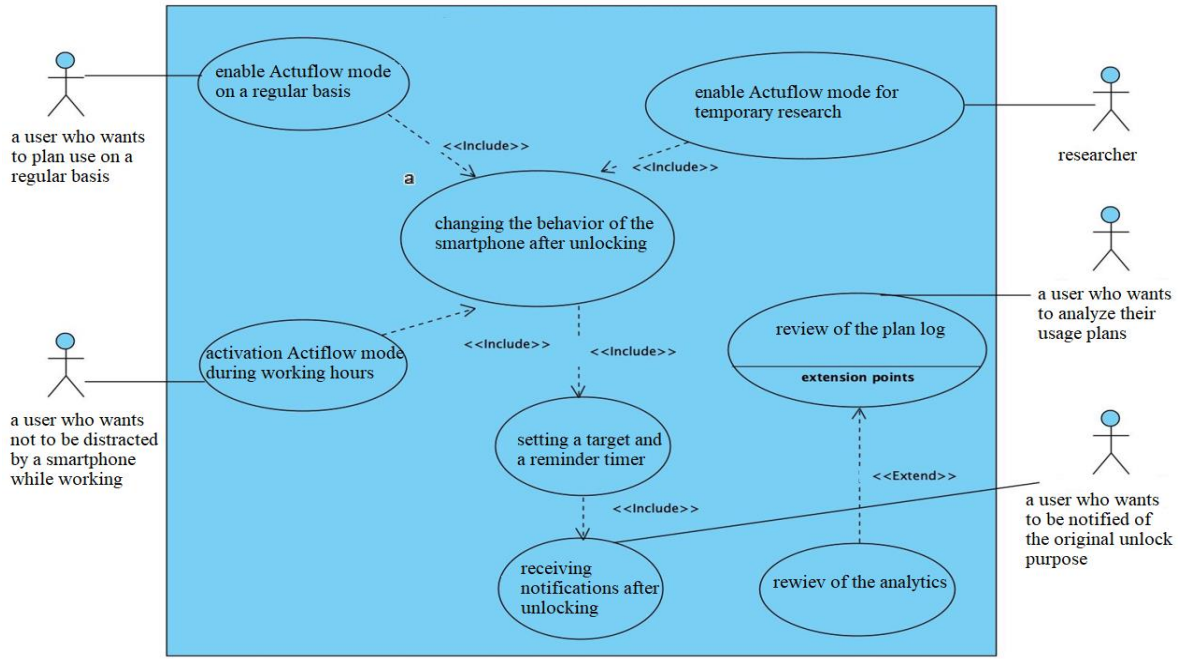


Figure 12: UML usage diagram

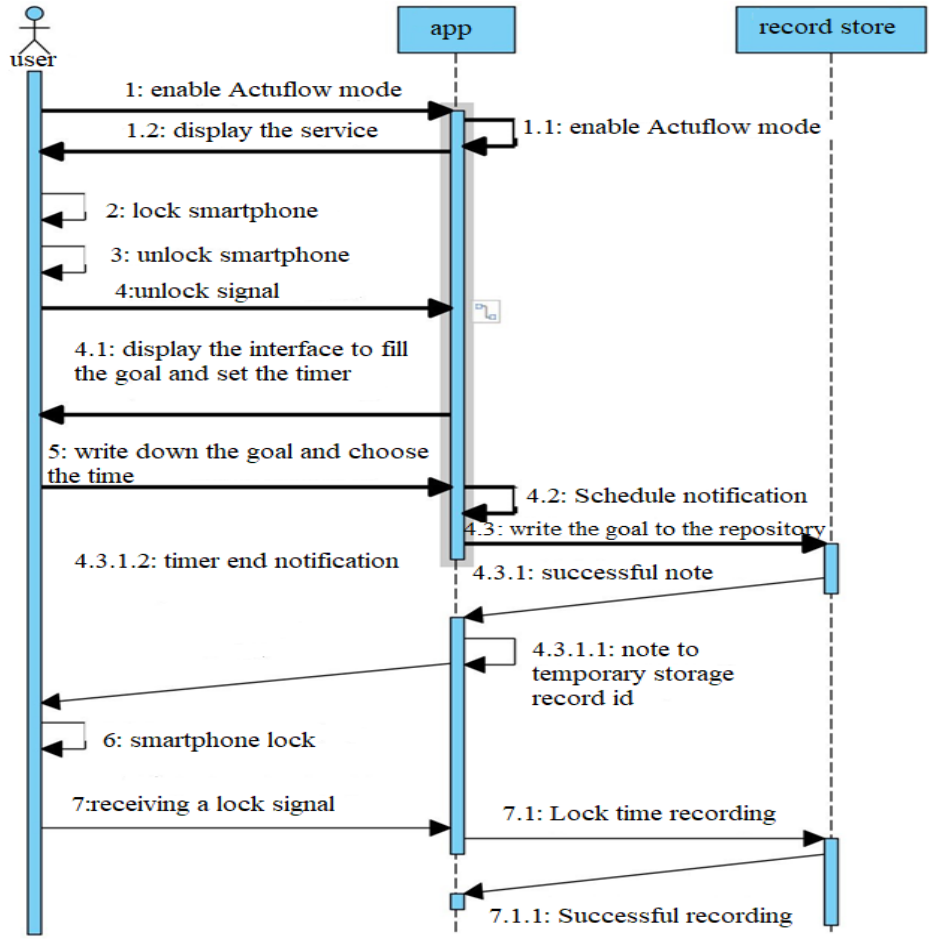


Figure 13: UML Diagram of the main sequence of Android

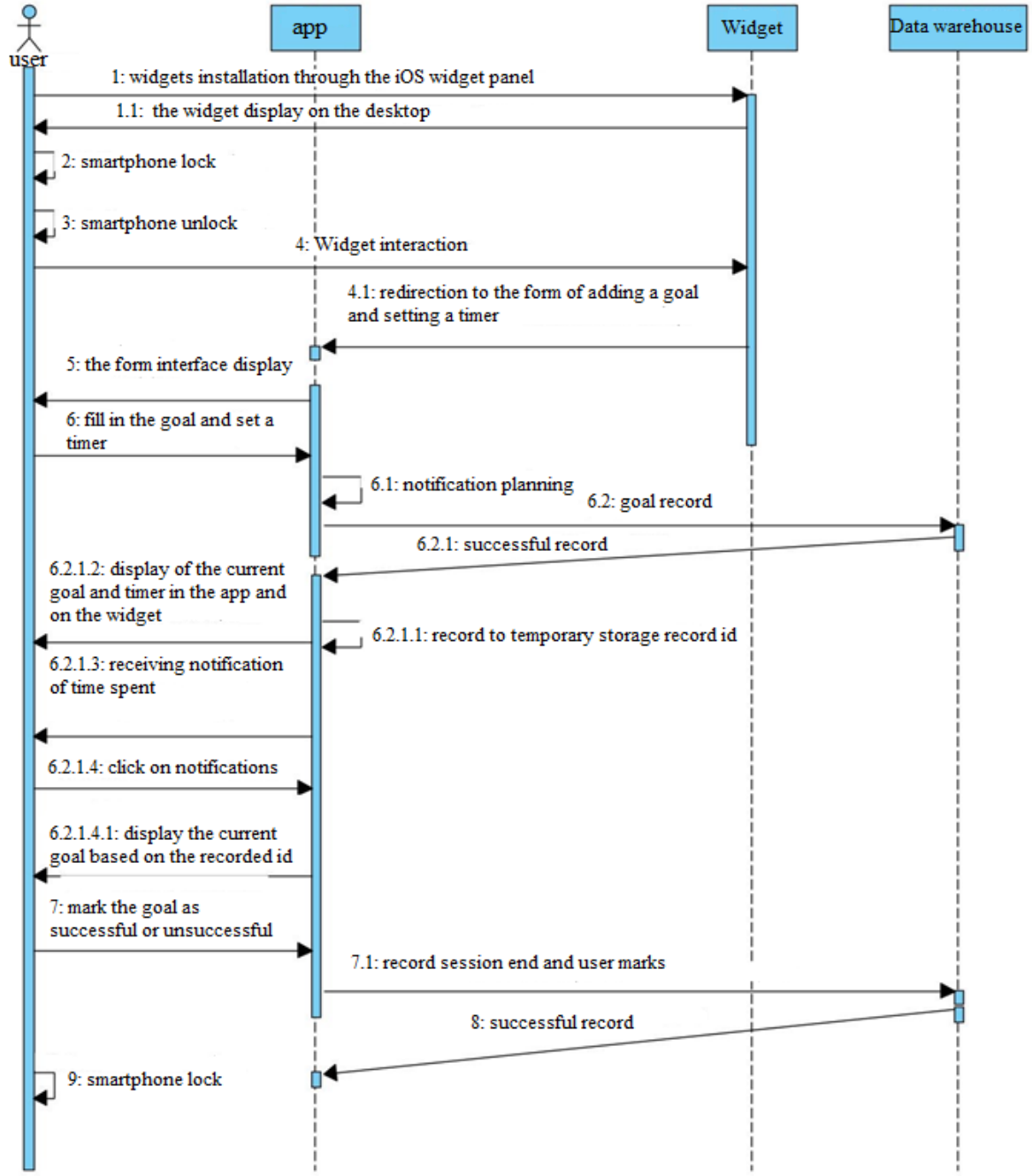


Figure 14: UML Diagram of the main sequence on iOS

Behaviour on different operating systems is slightly different given the limitations mentioned above.

Sequence on Android. Before considering the change of the interface, it is worth noting a short but essential line. It is the sequence of actions that turns on the change mode. After clicking one of the mode triggers (quick settings or in the application interface), the application registers the Foreground service, which accordingly monitors the unlocking events and associates this service with the notification. The interface is changed immediately after unlocking the smartphone and only if the user has enabled the mode offered by the application. Changing the interface involves launching an application window with a form to add an unlock goal. In pressing Save, the necessary instance of the fundamental essence of the system of intention is created. In case of clicking Dismiss or exiting the application window, the application records that this goal completion was missed.

Sequence on iOS. The series on iOS is a bit simpler. The user using the main page of the application or through the widget goes to the form and adds your goal and timer.

The sequence of viewing the log is the same on both operating systems. When you open the page of the magazine or analytics, the application receives information from the database and generates the appropriate report requested by the user. If it is a log, the application forms the required sample by date and generates and returns it to the application, generating a record. In analytics, the application creates aggregation queries to the database and sends them to the interface controller.

The end of the session differs on different operating systems, in the case of Android. Automatically when the user locks the phone, the Foreground service receives a notification of the lock. After receiving information from the temporary storage about the last active destination, the application records the lock time to the vibrant destination. In iOS, the user must go back to the application before unlocking and mark the goal as completed or not completed and then lock the smartphone.

3.3.7. Principal components and component diagram

The main components differ depending on the OS.

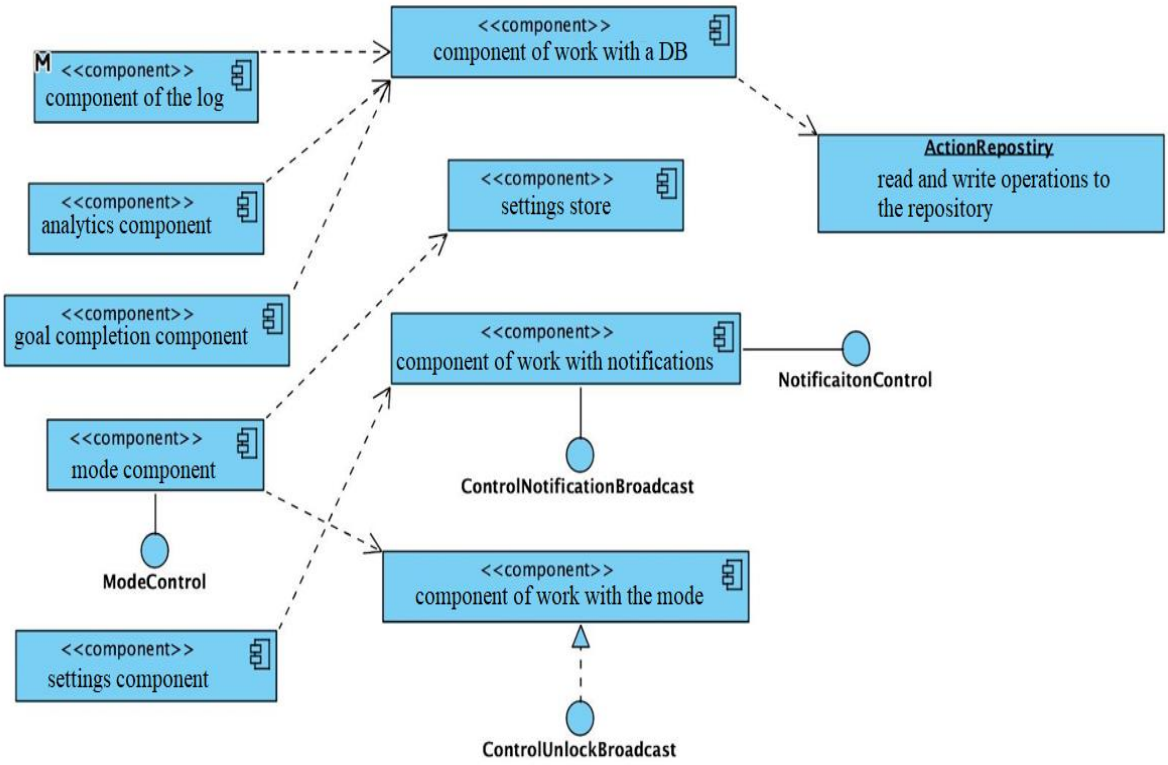


Figure 15: UML diagram of major components on Android

In the case of Android. The main components are:

- DB;
- Application settings repository;
- Database component;
- Notification component;
- Mode component;
- Log component;
- Analytics component;
- Goal Completion Component;
- Settings component.

In the case of iOS. The main components are:

- Application settings repository;
- Database component;

- Notification component;
- Widget component;
- Log component;
- The component of the current goal;
- The goal shape component.

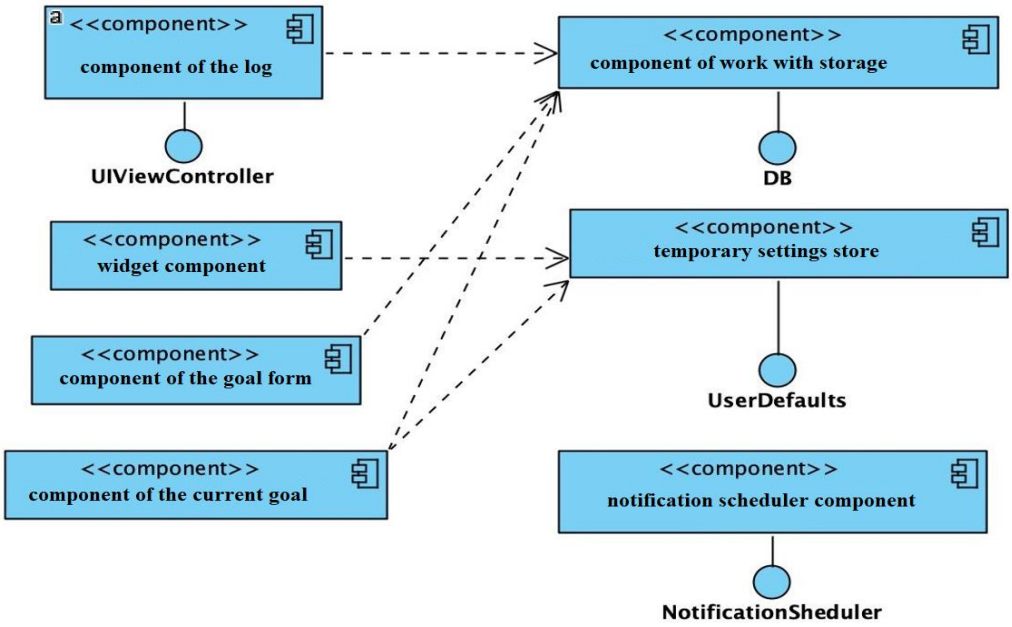


Figure 16: UML diagram of the main components of iOS

3.3.8. Introductory classes and class diagram

Classes may change during implementation, but when designing software using UML charts, you can create a framework of basic classes that can simplify certain issues at the beginning of development.

In application development, there are several class organization templates. Still, in application development, several approaches are leading, most of which are implemented in the frameworks and libraries used for development. In this case, it is MVC and MVVM.

Model-View-Controller (MVC) - This template involves dividing the system into three parts: an independent data model, a view, and a control module (which connects the idea and the data). It is used to separate data from the user interface so that changes to the user interface have a minimal impact on the data, and changes in the data model can be made without changes to the user interface. The approach is prevalent precisely because of its simplicity, but often in some implementations, the boundaries of Model View and Controller are pretty blurred. Model-View-ViewModel - MVVM facilitates the separation of interface development from business logic development, known as data interaction and business logic. The ViewModel is responsible for converting data for further support and use in View. This approach has a 'linking' of data and appearance in the ViewModel component, which is considered advantageous when developing systems with a simple display of data. The main classes, as well as the components, are expected to differ depending on the OS. In the case of iOS, the MVC approach is implemented. In the case of Android, it is MVVM. However, there are a few things in common in this approach. Interface classes are usually decomposed relative to interface pages. The data model class must have primitive attributes and must exist separately from business logic. But the implementation class of business logic and database access are somewhat different in these approaches.

Android. The Action class is the primary and only data class where the user's information and additional attributes necessary to implement sequence are stored.

The ActionDatabase class is a class that is responsible for accessing the database, namely migration, the location of the database file on the device. This class must use the ActionDao class. The ActionDao class is an implementation of the DAO approach. Data access object - a class that compares the methods of the class and actual queries to the database, the query language of the database.

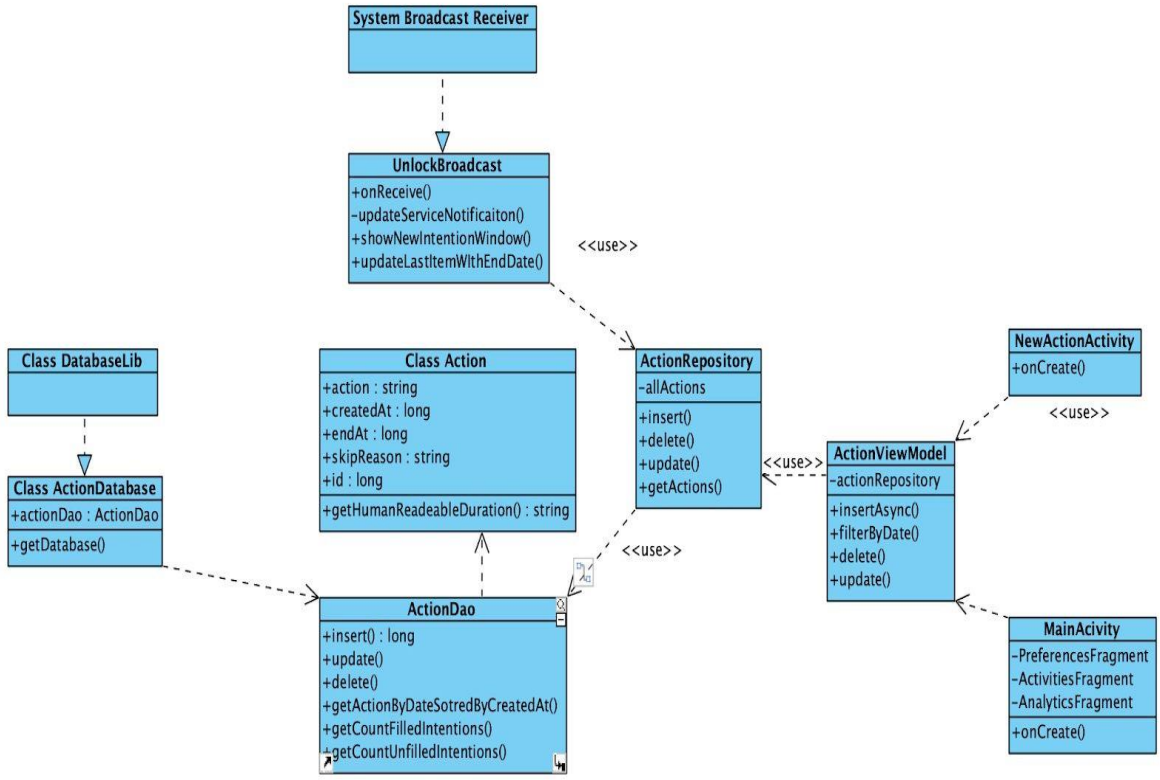


Figure 17: UML diagram of the main classes on the OS

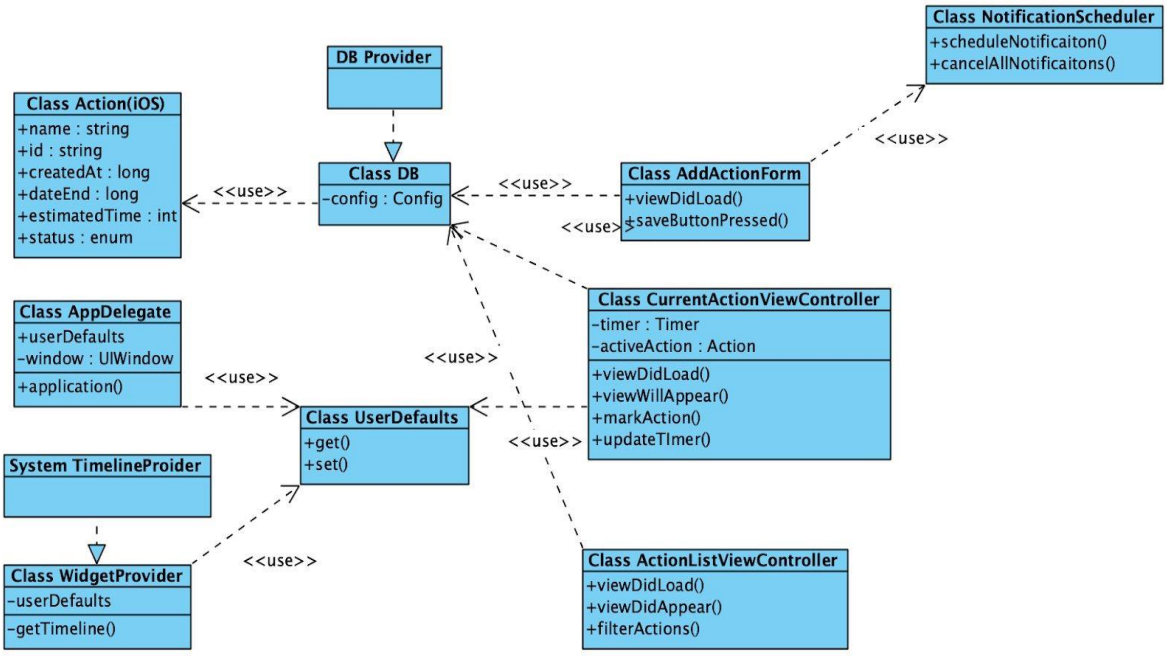


Figure 18: UML diagram of basic classes on iOS

The ActionRepository class is a class that MVVM is designed to use only ActionViewModel and generally only ViewModel classes. Therefore, the class must receive an instance of the ActionDao class to enter. The ActionViewModel class is already a class that contains the necessary methods for the essence of Action to work on the corresponding pages of the application. The attributes of the class are an instance of the ActionRepostiory class. This class will use the classes that are responsible for the interface pages. The MainActivity class is the root class of the Android application, which contains the implementation of the application's main window. In addition, this class acts as a controller of navigation between fragments (other pages) and the initialization of instances of other classes of the application in this system. The NewActionActivity class is the class responsible for the goal completion page. The UnlockBroadcast class is a class whose responsibility is to respond to unlock events and run iOS goal add formula activities.

The DB class is a class that is responsible for database access, database configuration, and migration.

The Action class is a data class of the main essence of the system. The AppDelegate class is the root class required by iOS to control the application. In this class, the application requests permissions from the user and responds to exiting the application. The AddActionForm class is a controller for adding a goal form. The main task is the validation of user input data and data storage in the database. This application also appeals to classes that schedule notifications based on user selection.

The CurrentActionViewController class is a current action controller. The main task is to display the current goal and the time remaining until the end of the timer and provide an interface to end the session. If the goal is not added, it shows a static page with a link to the form to add the goal.

The ActionsListViewController class is a class that is responsible for displaying a log page, filtering log activities, and editing activities. WidgetProvider class implements the interface of the new API introduced in iOS 14 - WidgetKit. It is required for the iOS 14 widget. The NotificationScheduler class is responsible for giving the application access to scheduling and cancelling notifications from other classes. Finally, the UserDefaults class is a class that provides access to a temporary primitive repository for an application provided by iOS, in which Apple recommends storing so-called "user settings."

3.3.9. Primary states and state transitions and state diagram

The state diagram of this system is primitive due to the simplicity and straightforwardness of the system. But on Android, there are some differences compared to iOS.

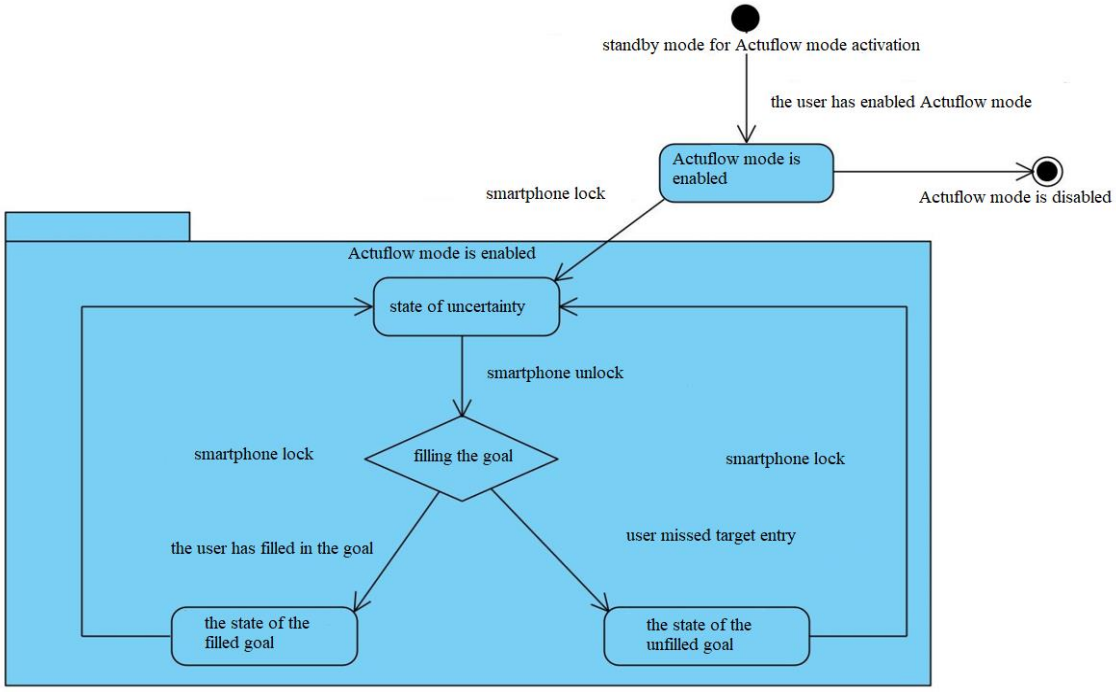


Figure 19: Status diagram on Android OS

Android. In the beginning, the system is in a state of waiting for the mode to turn on. After enabling the way, the application starts the necessary procedures to create the method, and the system goes into a state with active mode. The system has a completed goal state, an unfilled goal state, and an uncertainty state in a functional way. Accordingly, after the subsequent unlocking of the smartphone, depending on whether the user fills the goal, the system will go into the state of the served purpose. If not fill it will go into the state of the unfilled purpose. After locking the smartphone, the system will go into a state of uncertainty. The system acquires all these listed states only during the active mode.

iOS. The primary states on iOS are somewhat similar but still different. On iOS, the system does not depend on the mode, and the user fully controls the system's state. The user by the action of adding a goal puts the system in the form of a filled plan, and only after the end of the time allotted by the user to the plan, the user by his action puts the system in a state of an unfilled plan.

The states on the iOS version of the application are even more straightforward due to the lack of binding to the mode and the presence of another state of uncertainty. But there is a drawback because the state change trigger is the user, and accordingly, it is a bit more challenging to follow the method.

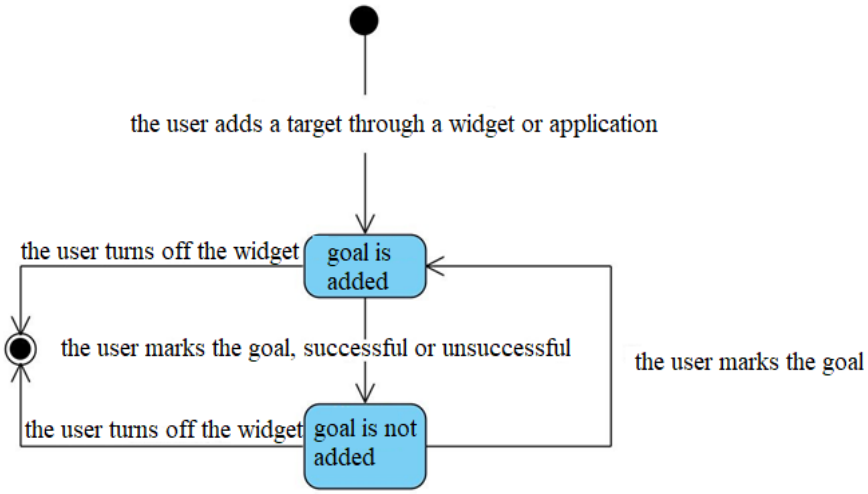


Figure 20: State diagram on iOS

3.3.10. Main activities and activity diagram

The activity diagram reflects the systems' capabilities, so a separate activity diagram was created for each OS. The main activities differ depending on the OS. But some activities are common to both operating systems, such as activity log viewing, analytics, and valuable links and settings pages.

The main activity of the user is to follow the sequence of filling his intentions. The exercise activates the application mode and fills out the form (which appears after each unlocks of the smartphone) during use until the user wants to follow the model. A minor activity is reviewing user goals in the journal and updating them. Viewing the analytics page is another little activity that partially overlaps with viewing the journal. And accordingly, setting up the system and viewing valuable links about the project is another user activity. It is from this activity that the user can get to the website of the application.

Activities can be divided into two categories: filling unlock goals and using additional features in the middle of the application. Log viewing and analytics, and useful links are activities that are present in the user's activity on iOS as well as on Android. The only difference is the main activity. The user's main activity begins either with the widget or with the page of the current goal, and the user has another action in the form of checking the following of their plans at the end of the timer, which the user set when filling out the form.

In this section, a systematic analysis of the issue was conducted, a tree of goals was built, and a general goal was formed. The problem was set out in detail and substantiated, and several requirements were developed using user history, functional and non-functional requirements and several UML diagrams. The result of this section is already used directly during application development.

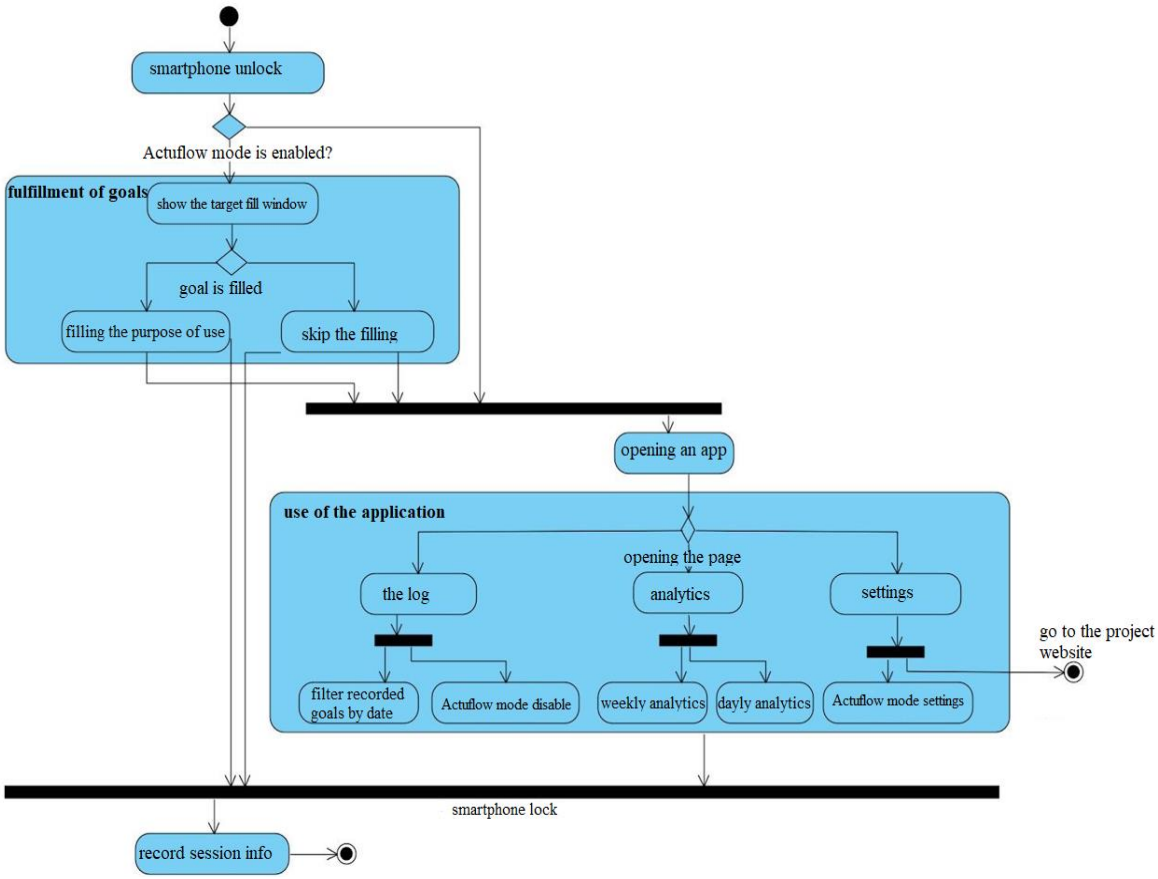


Figure 21: Activity chart for Android OS

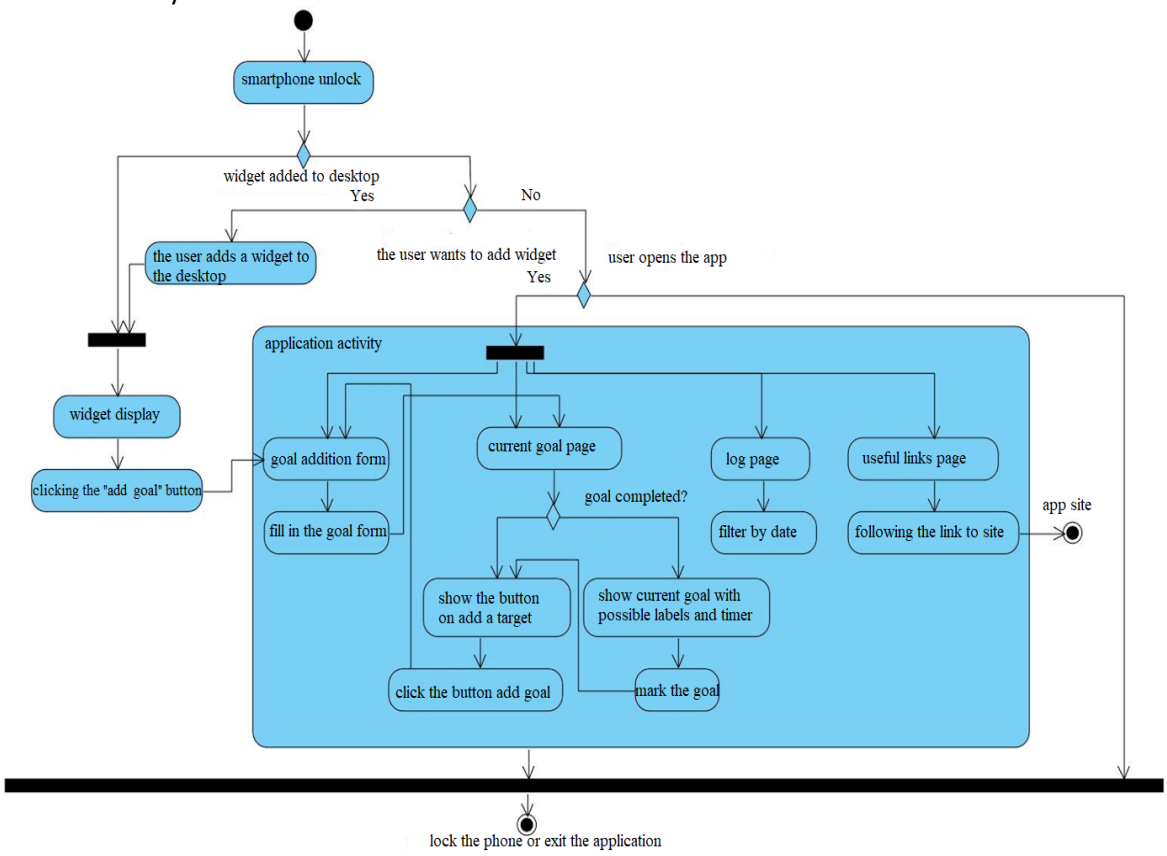


Figure 22: iOS activity diagram

3.4. Methods and means of solving the problem

3.4.1. Choosing and substantiation of methods of problem-solving

First of all, it should be borne in mind that the problem of screen time and abuse of smartphone use is complex, and one application cannot solve this problem, but it can provide a tool that will help gain control. In general, this is inherent in this particular problem and the issue of human productivity. After all, productivity and quality are difficult to measure without the context of the application.

The method implemented to solve the problem was proposed by the author of the book Digital Minimalism Kel Newport [18]. Going into the history of personal computers, the personal computer revolution came about because computer manufacturers stopped producing narrowly specialized computers to support a production line and began to make highly specialized computers. The bookseller had the opportunity to have his Apple II at work in his store for accounting, and after work, to deal with personal finance on the same computer. Since then, people have the impression that "broad purpose" = "productivity". The only problem with this impression is that there is no time in the formula. Comprehensive computers have the undoubted advantage that we do not need to have several computers for different purposes. Still, if we have a computer that can do everything, it does not mean that we can do more things at once. At that time, personal computers could not run more than one program at a time available to the user. The situation has changed somewhat, and the number of simultaneously open programs and programs running in the background has increased significantly. And users of modern computer devices very often change the active program, which seriously affects the quality of their work or leisure. And as a solution, the author proposes to "transform" their devices during their use into devices that can perform only one action at a time. Of course, the author does not mean to limit the capabilities of devices, namely to change the principle and method of use. The idea is that if you use the device as a source of entertainment, you do not need to use entertainment applications or social networks when you, for example, sent a message to a friend. The author himself emphasizes the irony of the method. Still, the author has researched people worldwide, and most people who have implemented this principle have reported increased productivity and reduced screen time.

The idea of the system, which was developed in the master's thesis, is implementing this method and its modification. The modification will be that the responsibility for implementing this method will be placed directly on the user. If the user does not follow the processor does not follow the instructions, the system will notify the user of his actions. Let's consider the diagram of activities in Fig. 21 and Fig. 22. It can be noted that the system provides a record of the user's actions even if his actions do not follow a conditionally ‘successful’ sequence in the system. This method has its advantages and disadvantages. The most significant benefit is the simplicity and conciseness, i.e. you do not need to implement too complex processes of control over user actions. Some other applications tried to use their functionality to control the user's actions strictly, but users responded that this interfered with the use of the device. And the biggest drawback is that the user has to want to fix the situation using a smartphone. That is, the user must be motivated to use the system. It significantly reduces the range of potentially interested users in using the system.

3.4.2. Selection and justification of means of solving the problem

Given the features of the OS for which the development of applications and limitations of these operating systems, it is necessary to consider each application separately (separately for Android, separately for iOS). There are hybrid development tools that work for both methods simultaneously, but given the specifics of applications and the requirement to work with screen unlock and lock events on Android, the possibility of using hybrid application development tools that generate code simultaneously for two systems was not considered.

3.4.3. Choosing tools for development

Android and a programming language. The Android application developer community has long used the Java programming language as the primary programming language. Using Java has its

advantages. Namely, language support is provided by large companies, the number of already developed software in Java is vast, which means that there are many resources for training and support. The language is also object-oriented, which provides the use of many patterns that simplify further support. Until recently, the Java programming language was the only official option for Android development. However, in 2017, Android officially began to support another programming language - Kotlin. Kotlin is a programming language developed by JetBrains, which is known for its IDE. Kotlin is a statically typed language that runs on top of the JVM. The main advantage of Kotlin over Java is its novelty and the introduction of a new syntax that may be familiar to programmers who have not worked with Java enough. That is why the Kotlin programming language was chosen because the experience of the system developer did not include the long-term use of the Java programming language.

Development environment. The undisputed leader in the Android Studio development environment provides all the necessary tools for development in the form of an emulator of different versions of Android, visualization of written interface templates, and more. This interactive development environment is recommended for developing applications on Android.

Additional libraries. The different libraries that will be used are:

- Android Room for working with databases, which helps to unify the work with the database;
- AppCompat to work on other versions of Android;
- ConstraintLayout to simplify the layout of layouts, using so-called constraints;
- MPAndroidChart for creating graphs.

All additional libraries (except MPAndroidChart) are standard and recommended by Google for a set of libraries for developing mobile applications.

iOS and a programming language. The situation with the main application programming languages for iOS is somewhat similar to the problem with languages for Android applications. Since the release of the first iOS applications have been written in the programming language Objective-C. It is a high-level object-oriented, general-purpose programming language developed as a set of standard C extensions. But Apple has decided to invest in the development and maintenance of its programming language. That's why Apple created Swift, a multi-paradigm compiled programming language developed by Apple to coexist with Objective C and be more resistant to incorrect code. Swift was presented at the WWDC 2014 developer conference. Swift language is focused on different levels of programmers, and therefore the basic concepts of the language are very primitive and simple.

That is why the Swift programming language was chosen.

Development environment. When choosing a development environment for iOS apps, there is no choice. Apple clearly states that there is only one officially approved application development tool, namely XCode. XCode is an integrated development environment (IDE) manufactured by Apple. Allows you to create software using technologies such as GCC, GDB, Java, etc. XCode is quite user-friendly in terms of development because Apple has made this development environment very narrowly specialized, so the whole set of tools is useful during development. Additional libraries:

- Realm is a library for working with the relational database Realm;
- SwiftUI is a framework for programming interfaces recommended by Apple;
- EurekaForms is a library for creating forms and validating them;
- KDCircularProgress is a library for visualizing progress bars.

3.4.4. Database selection

Given the need to have two applications for each operating system, it was decided not to use any specific database on both platforms but to use the more convenient database to run on the respective operating system. Also, the applications were developed at different times, and therefore the databases on both applications are not the same. But it is worth noting that both tools support the export of data in the desired format, and in the future, this difference is not a problem to switch to one device.

Android. For Android, a rather primitive but popular and recommended option was chosen, namely SQLLite. SQLite is a lightweight relational database management system. Embodied as a library, where many of the SQL-92 standards is implemented. The SQLite source code is distributed as a public domain, i.e. it can be used without restrictions and free of charge for any purpose. Financial support for

SQLite developers is provided by a specially created consortium, including Adobe, Oracle, Mozilla, Nokia, Bentley, and Bloomberg. The peculiarity of SQLite is that it does not use the client-server paradigm, i.e. the SQLite engine is not a separate process with which the application interacts but provides a library with which the program is compiled, and the machine becomes part of the program. Thus, function calls (APIs) of the SQLite library are used as an exchange protocol. Given the requirements for the database, SQLite is ideal, especially since it is one of the recommended options for organizing data storage in Android applications.

iOS. SQLite could also be used for an iOS application. But given some complexity with this database and the potential future transition to a standardized database for both applications, Realm was chosen for iOS. Realm is a relatively new development. It is even more correct to call it not a database but an object-store. Accordingly, this is not a relational database, and it may be better suited for this type of application than SQLite. The main advantage is the simplicity of work, the implementation of links through the so-called "links". Because each query result and each proxy object is a view of the base data, any changes made to the database are reflected in all objects that point to the same data. Realm commonly calls this behaviour a "zero-copy architecture" (along with the previously mentioned access to downloaded data).

3.4.5. Choosing tools for interface design

The sketch was used as a graphical editor and Zeplin to distribute the design to developers. The sketch is a paid vector graphic interface editor for Apple's macOS, developed by the Dutch company Bohemian Coding. He received the 2012 Apple Design Award. It is mainly used for user interface design and website and mobile application design and does not include printing design features. That is, the main focus of Sketch is the development of software design.

Zeplin is a tool for collaboration between designers and developers. It is a web or desktop application that allows designers to upload their interface design to the program, where developers will get all the necessary technical details of the plan. It provides an opportunity for software developers not to install what designers need or not to convert to intermediate formats such as pdf. Zeplin analyses the downloadable design uploaded by the designer and the platform for which the plan was developed and provides the developer with information such as colours, sizes and font types, indents, etc.

3.4.6. Choosing tools for website application development

The application site requirements do not require the site to be interactive [69-75]. That is, the application site must be purely informative. Because of this, it was decided to use the framework to create Gatsby websites and store content for the area along with code in Markdown format. To develop a website, Gatsby requires using the React library, which slightly changes the approach to writing templates. The developer writes the interface code using the React library, and in the code sets the CSS style, which is then displayed in the browser. Gatsby combines content that is written in Markdown format and provides an additional API that is available in the code to use that content. The output of Gatsby generates a so-called bundle, which already includes HTML and JS, which can be displayed in the browser. One of the benefits of creating websites this way is the ease of hosting. Particular web services allow you to download free static files generated by Gatsby and create a web page available via HTTPS on the service domain. Accordingly, by later purchasing a domain name, you can redirect the DNS service to this link and thus solve the problem of hosting without working with the server directly. One such service is Render, which has a free tariff and is ideal for site purposes.

3.4.7. Choosing tools for analytics

Application stores where you need to place applications also contain tools for analytics. In Android, this is the Google Play Console, and in the case of iOS - App Store Connect. Both platforms have similar analytics functionality, but the Google Play Console provides slightly more information than the App Store Connect. For example, the analytics panel offers the following information: number of

downloads, number of downloads per hour, information about the devices from which users downloaded the application, information about errors in the application, user feedback, and analytics by country. In addition, Google Analytics was used for the website, which is a viral website analytics tool and free.

3.4.8. Choosing tools for organizing support and creating a community

It is also necessary to provide a service where users could discuss ways to use the application or report bugs in the application and create a community of users. The Reddit platform was used for this purpose. Reddit is an entertainment, news online service [76-77], and online media, where registered users can add their content, such as text posts or direct links, and discuss them (BBS). Like many other similar sites, Reddit supports a system of voting for favourite messages - the most popular of them appear on the site's main page. Also, the most popular posts are displayed on the pages of the relevant categories first. After that, the content is organized with the help of interest cells called "subreddits". The section discusses methods and tools for developing a system that helps users plan their smartphone use and thus reduce their screen time. Development tools for both Android and iOS platforms and for the application's information website were described. Interface design and analytics tools were also described in the application and on the website. The programming language for the Android application is Kotlin, for the iOS application - Swift. The data warehouses will be SQLite on Android and Realm on iOS. Following section also described additional libraries that were used in mobile and web application development as in works [78-86].

4. Experiments, results and discussion

4.1. Description of the task implementation

As part of the work, an application for the Android OS and an application for the iOS OS were developed. On Android, the application requests special permission to block other applications and monitor the events of blocking and unlocking. The application awaits the possibilities of opening the smartphone and launching one of its activities immediately after unlocking. To run Activity, the application creates a service that runs in the background Foreground Service. Analytics and the journal are part of the application. When working with data, the application writes everything to the SQLite database. On iOS, the application is a bit simpler, and the user does all the actions through the application, and the windows do not appear automatically after unlocking. To improve the user experience, the application has support for iOS 14 widgets. When working with user records, the application records everything in the Realm object-store.

4.1.1. Description of the composition and structure of the database

Scheme of the table on Android and Schematic of the object on iOS:

Figure 23: Table in the Android application **Figure 24**: Schema of objects in the object storage on iOS

Field id - auto-incremented primary key. Action field - a text field where the text entered by the user after unlocking in the form of entering the goal will be written. The createdAt field was the timestamp

Unix epoch when the goal record was created. The endAt was the timestamp Unix epoch when the goal session was completed. Field skipReason - text label of the way to avoid filling the goal (exit via the dismiss button or use the system button "go back"). The skipReason and endAt fields can be NULL.

The id field is the unique identifier of the UUID object. The name field is a text field where the user's input text will be written. Field createdAt was timestamp Unix epoch when this record was created (click user SAVE button). The endAt field was the timestamp Unix epoch when the goal session was completed. The estimatedTime field is the user's selected reminder time in seconds. Finally, the status field - can be set to "successful" and "failed" is a label for the user. All fields are required.

4.1.2. User interface design description

The interface design was performed in the Sketch user interface design tool.

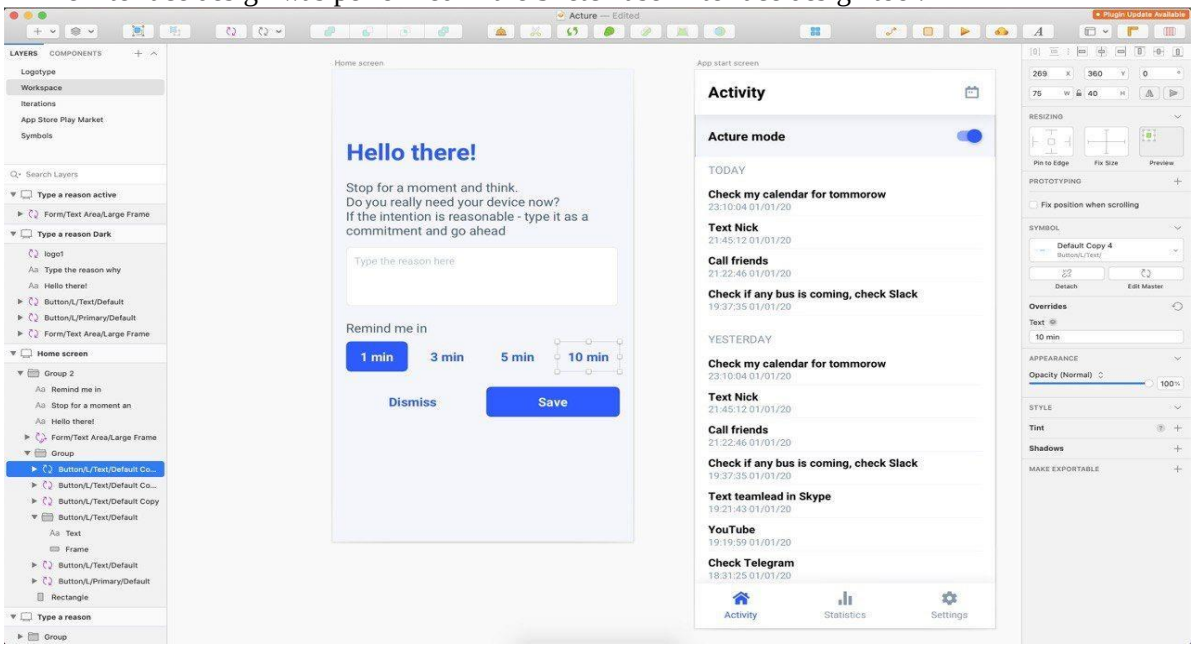


Figure 25: Interface design for the log page and goal fill page made in Sketch

It is also worth noting that Sketch has created a library of elements used on other screens. In addition, the design of all screens of the application with Sketch has been uploaded to Zeplin.

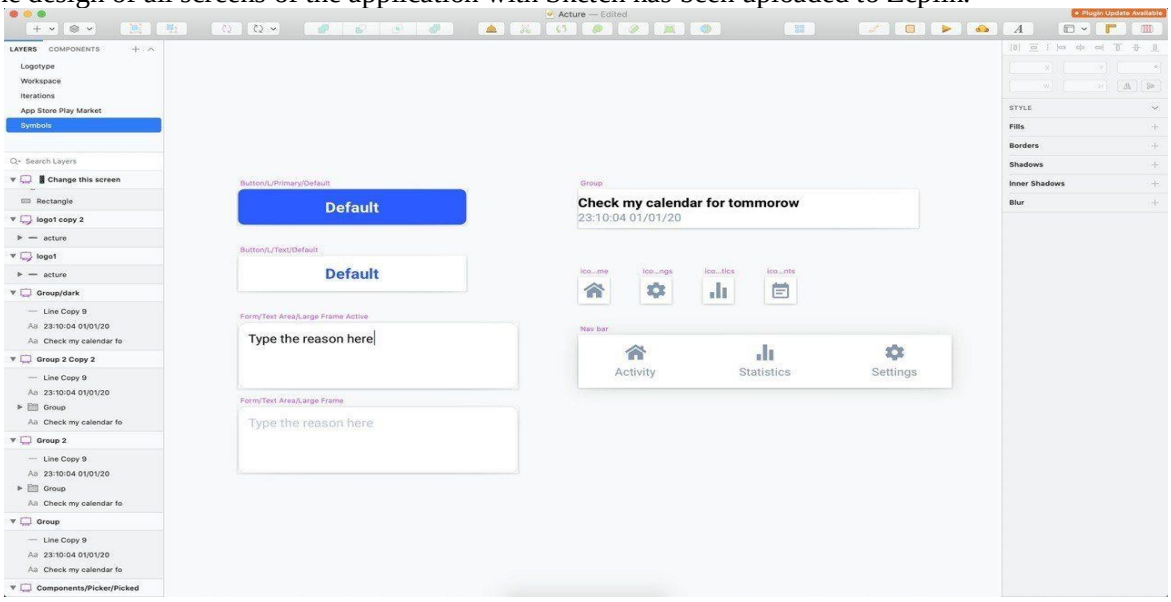


Figure 26: Design of basic elements made in Sketch

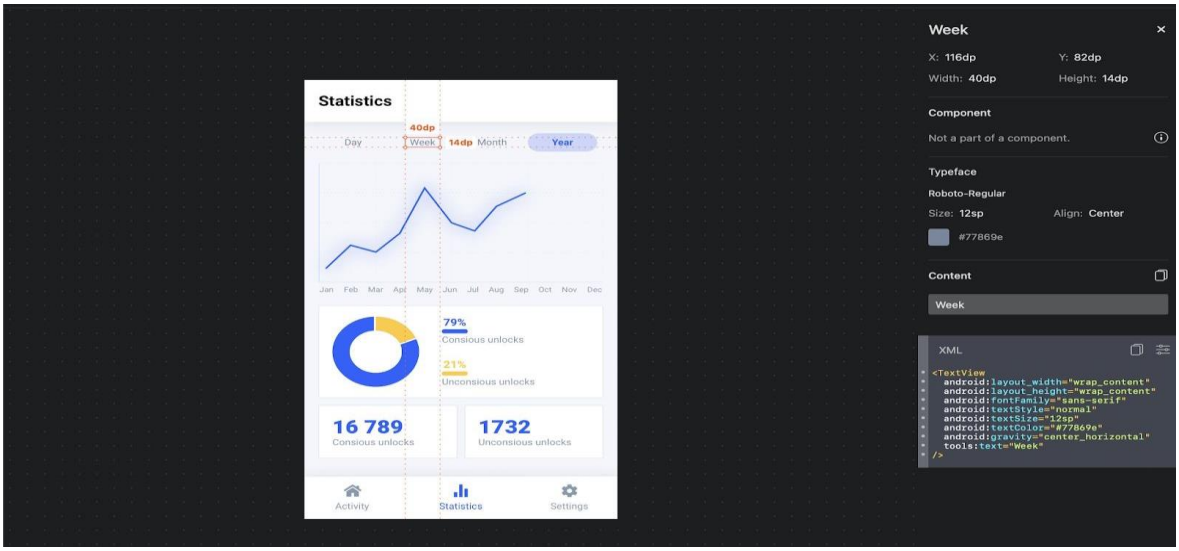


Figure 27: View of the Zeplin window for the analytics page (additional information for the developer is on the left)

4.1.3. Description of the implementation of the mechanism for adding goals

Adding goals is the essential element of the system, so the design and implementation of this sequence are as simple as possible because the user repeats this activity after each unlocks of the smartphone.

Android. First, the user must enable Actuflow mode in the application using the switch, as shown in Fig. 28.

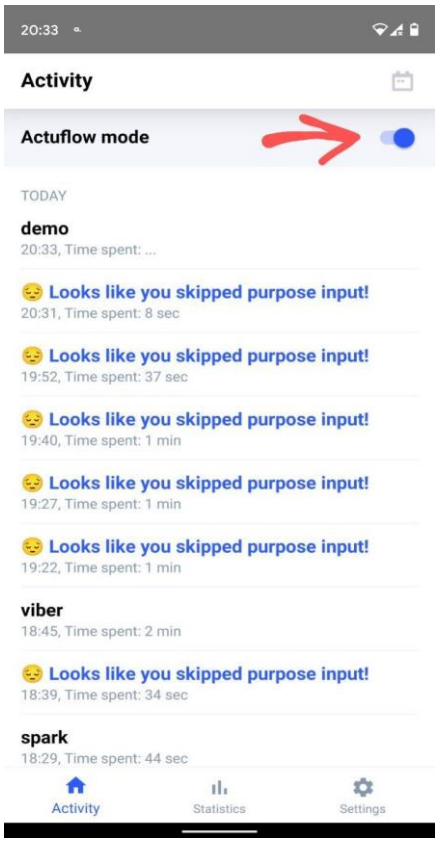


Figure 28: Enabling Actuflow mode in the application

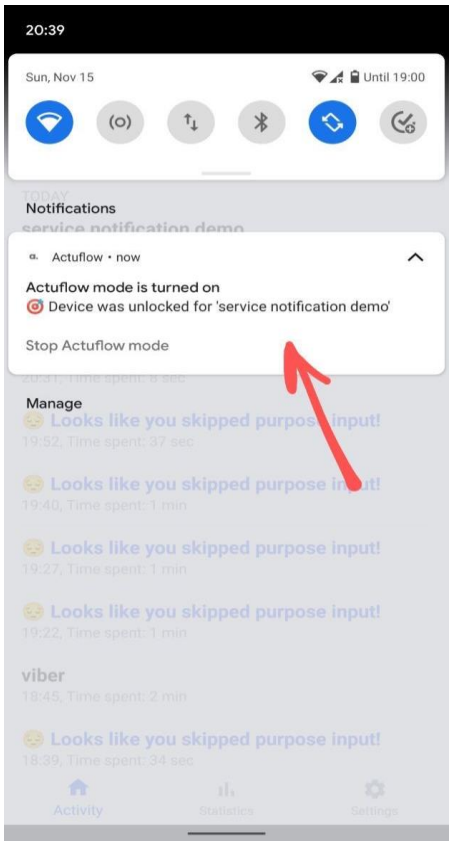
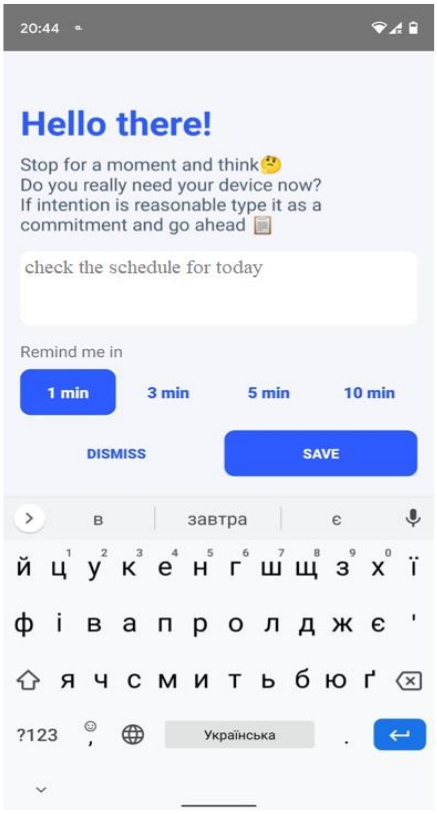


Figure 29: Actuflow mode background service notification

This switch causes the registration of Foreground Service - Device Tracking Foreground Service. Accordingly, in its onCreate method, it registers the corresponding BroadcastReceiver on the user's actions (blocking and unlocking the smartphone) and reports the necessary background notification, which should notify the user that the Actuflow application service is running.

Also, this notification (Fig. 29) will serve as a reminder of the goal, which the user will then write after unlocking the screen. Notification means a successful start of the mode. Now the user can lock the smartphone, and the next time the screen is unlocked, the goal creation window will be displayed. With the notification, the user can also turn off the mode without opening the application using the "Stop Actuflow mode". These are the initial user steps with the system that are required for the application to work. In the future, it is planned to add the mode planning function for specific hours because now the user is forced to turn on the mode.



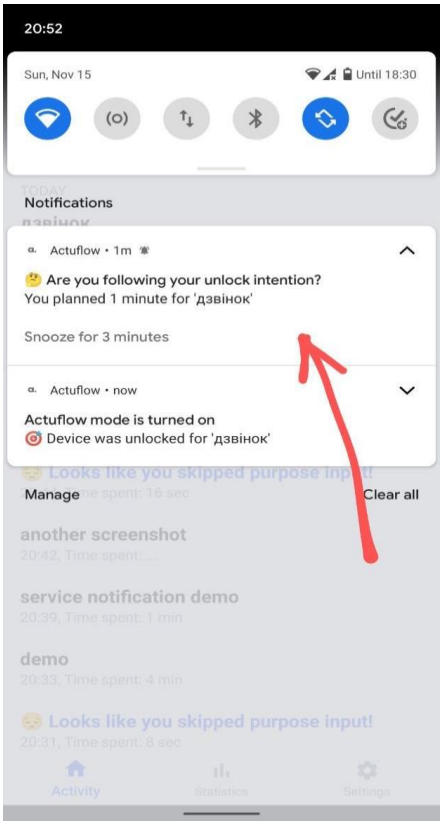


Figure 30: Window-form of filling the goal **Figure 31**: Notification that reminds the user of his purpose

After the next unlock, the application window appears (Fig. 30). You can configure the Dismiss button and the ability to create reminders on the settings page. When the user clicks the Save button using the ActonRepository class, namely the insert method, which uses ActionsDAO, the data is written to the SQLite database. A unique record ID is registered to the temporary data store, and then a reminder notification is planned. Suppose the user avoids recording the goal via the Dismiss button or using the system back button. In that case, the corresponding NewActionActivitiy activity class methods respond to these events and record the information in the skipReason field of the Action table.

After locking the phone, the application updates the unique identifier recorded in the temporary storage data about the time of the phone lock (end of session).

If the user continues to use the smartphone for more than the allotted time, the user receives a notification Fig. 31. In addition, the “snooze for 3 minutes” function is also implemented so that the user can snooze reminders. If the user has locked the phone before the notification, the notification will be cancelled, and the user will not receive an excessive number of messages.

iOS. Due to the limitations of the iOS system, the responsibility for fulfilling the goals is transferred to the user. In addition, the iOS app version is less interactive than the Android version. The user has two options: add a widget (Figure 32) through the system tools or go to the application and through the Home page to add your goal. You had to use the iOS Widget Kit to display the widget. More

specifically, create a separate extension for the application and implement the TimelineProvider interface. Using the tool for links between applications, the widget button "Add intention" through a link that has the format ‘actuflow: addIntention’ iOS redirects the user to the application and the application with the suffix addIntention processes this request and redirects the user to the form of adding a goal. Accordingly, when the user clicks Save, the Action object described above is created and stored in the Realm repository. Also, at this point, the notification is scheduled using UNUserNotificationCenter. The next step is to redirect the user to the current destination page.

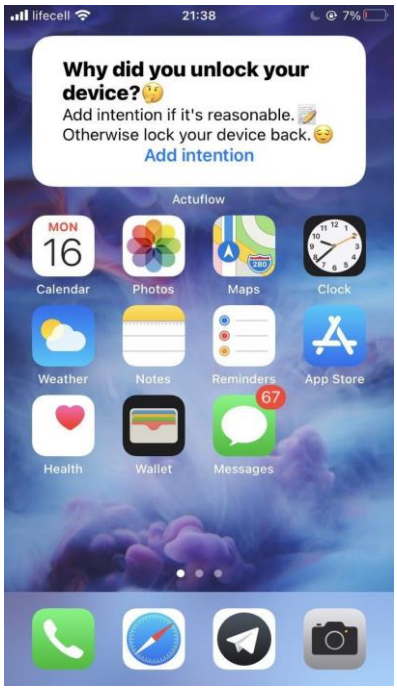


Figure 32: Actuflow iOS 14 widget

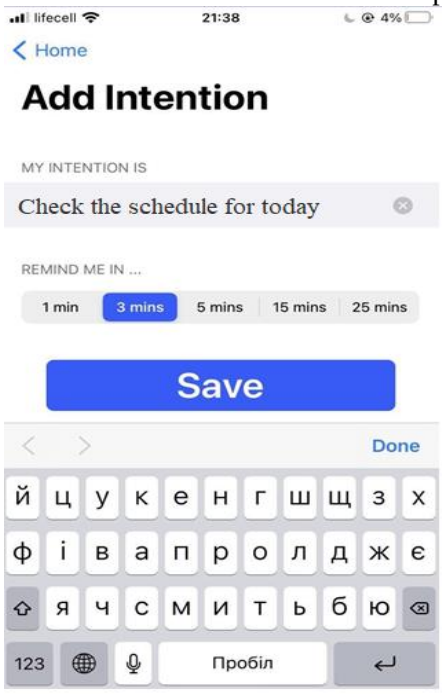


Figure 33: Form of adding a goal in the application for iOS

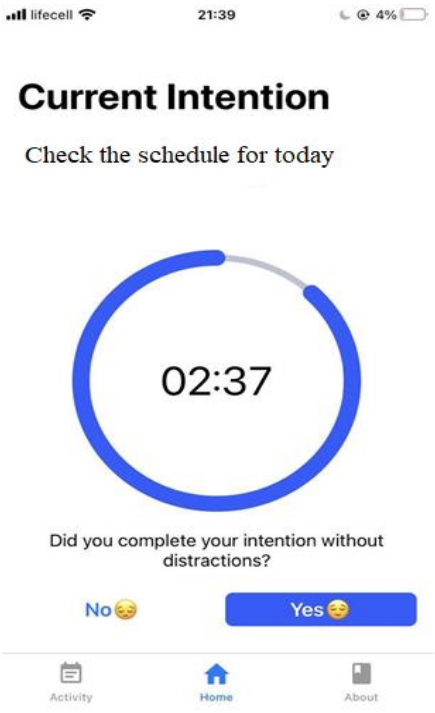


Figure 34: Current goal page in the iOS app

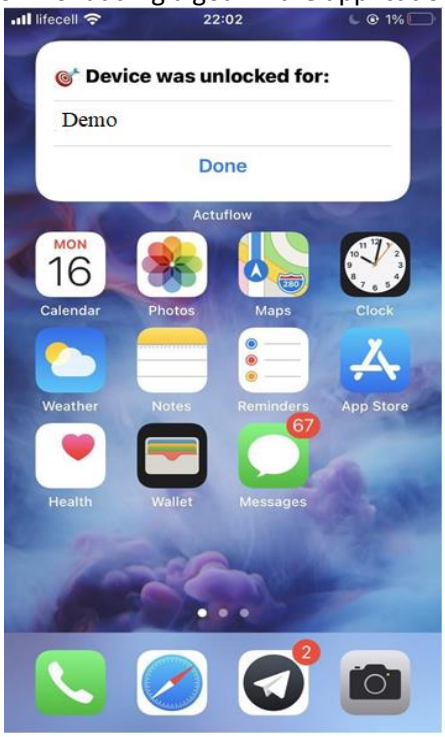


Figure 35: Changed widget status after adding a goal

The critical point is that after saving the goal in the temporary storage userDefaults record a unique goal ID, which then displays the page of the current plan. When the user exits the application, the widget

content will also be updated to see a unique ID in userDefaults. That is, the page of the current goal is dynamic relative to the state in which the system is at the moment.

4.1.4. Description of the log and analytics in the app

In the Android application, the analytics capabilities are slightly more comprehensive due to an analytics page where there is a visualization using bar and pie charts to display the proportion of filled and unfilled unlocks. On iOS, a page with such charts is planned to be added in future versions of the application.

Android. The log page has a date filter using the standard Android DatePicker element. Each element of the list has a creation time localized for the user according to his time zone (because in the database, the time is recorded relative to UTC), how much the user spent on the session. And also, the log contains the record, which testifies that the user missed filling in the whole. Other colours of the text mark them. The analytics page (Figure 37) contains a vertical bar chart hourly on the day's tab and by days on the week's tab. It also includes a pie chart and the total number of relevant types of unlocks (with a completed target or blank).

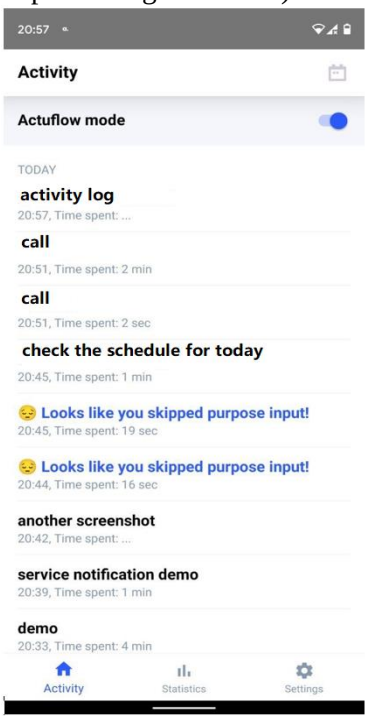


Figure 36: Activity Log Page

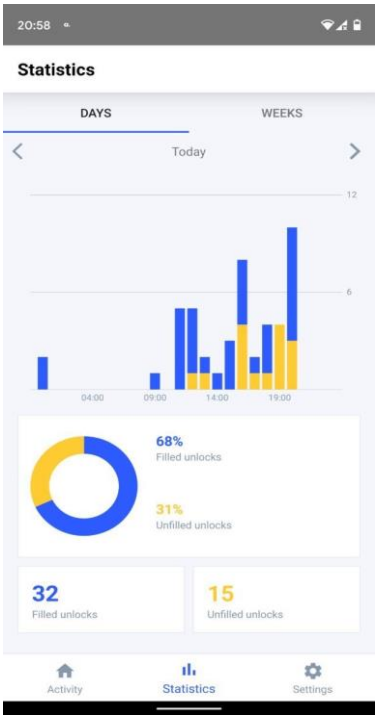


Figure 37: Analytics page

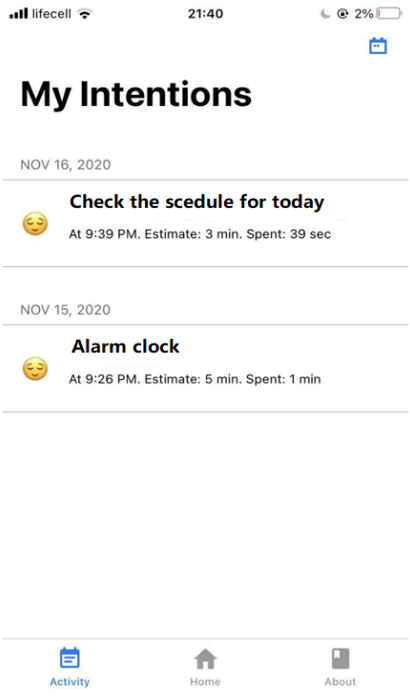


Figure 38: Log page in the iOS app

iOS. When a user accesses the log page (Figure 38), the ActionsListViewController controller uses the singleton instance class responsible for working with the database provided by the RealmSwift library. The controller sends a request to the repository Realm results = uiRealm.objects (Action.self) .sorted (byKeyPath: "createdAt", ascending: false) .filter ("createdAt BETWEEN% @", [dateStart, dateEnd]). Where dateStart and dateEnd are the default start and end of the current day, or when the user has selected the appropriate date filter using the UIDatePicker that the user called by tapping the icon.

4.1.5. Description of the implementation of the settings page and useful links

The settings page and valuable links play an essential role in terms of the UX user experience because if the user has a question, the application should have a place where the user can find the answer to this question.

Android. The settings and useful links page (Figure 39) contains the two settings of the target completion window that have already been mentioned (the ability to remove the Dismiss button and the ability to remove notifications) and accordingly useful links to the project site and the community created in Reddit and the project support page created on the Patreon service.

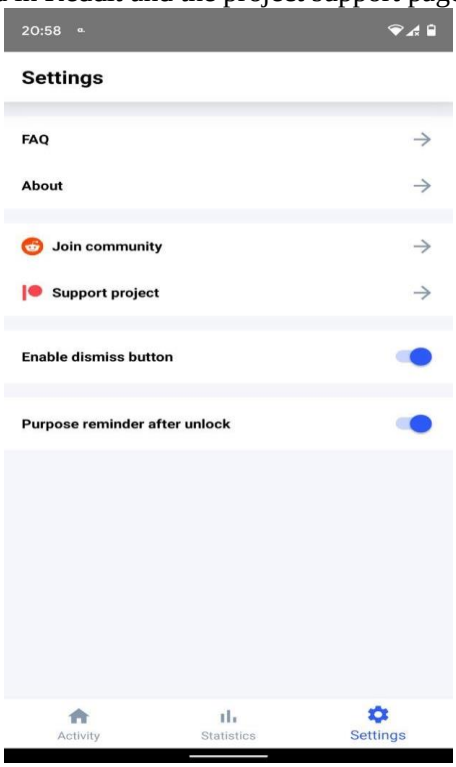


Figure 39: Settings screen and useful links

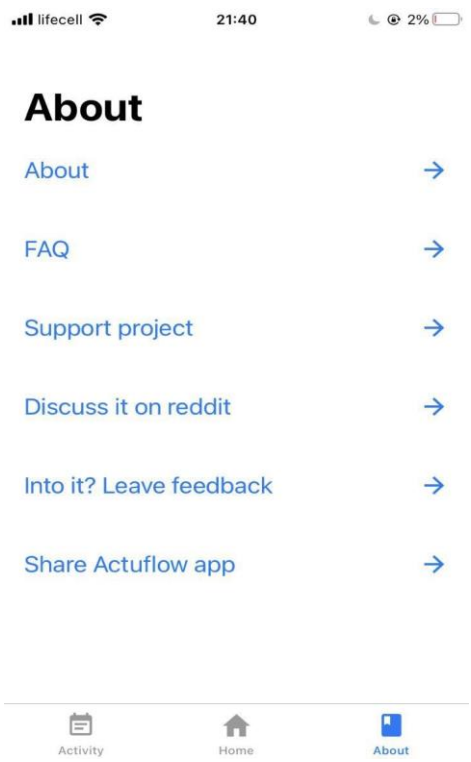


Figure 40: Useful links page

iOS. In the case of the application for iOS, there are no additional settings in the system, so the page (Figure 40) contains only useful links (Links to the site, Reddit, writing reviews).

4.1.6. Description of application distribution implementation

The distribution of applications was implemented according to the recommendations of Apple and Google. By purchasing the appropriate licenses from the Google Play Console, the Apple Developer Program application in beta has been published in the relevant application stores. When the application was in the Beta version for testing on Android, the platform for beta testing of the Google Play Beta Program was used on iOS TestFlight. And after the release of Beta, the application is available on both platforms officially in application stores. Starting with the Beta version of the application, on both platforms, both Apple and Google, the application underwent a mandatory inspection of each of the platforms for compliance with the store's requirements. The application is in the Productivity category. There are no age restrictions on the app, and the app is available for all ages.

The application has its own page with a description of the application with images of its functionality in each application store. On both platforms in the images of the functionality was used the basis of user activity, namely the process of filling the goal by the user.

4.1.7. Description of the website implementation

As mentioned above, the Gatsby tool was chosen as the website creation tool. All text on the website is stored in markdown files, and the page template is written in JSX files (which use this text). These files are converted by the Gatsby collector when deployed into minified JavaScript files that are perceived by modern browsers. Using the Render platform and connecting Render integration with the GitHub repository, the Render service launches a collector after each committee in the warehouse.

Then, it updates the website on its internal domain ending in onrender.com, to which you can redirect traffic from the purchased domain. After buying the acture.app domain on GoDaddy and making the necessary DNS settings, the application site is available at https://acture.app. Implemented applications have links to site pages with frequently asked questions and a description of the project. The site also has a privacy policy page required to publish applications to Google Play and App Store.

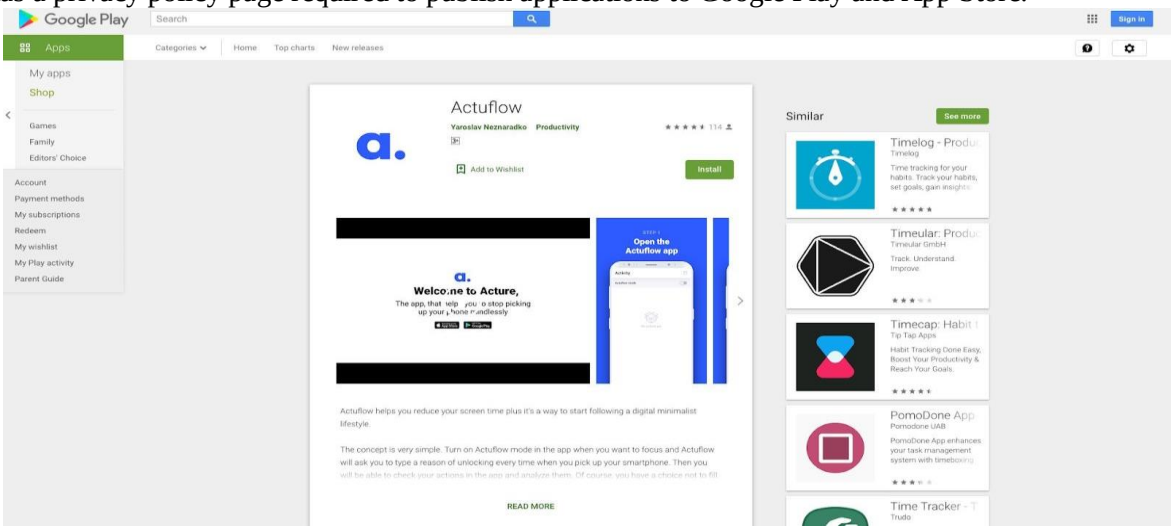


Figure 41: Appearance of the application page in the Google Play store

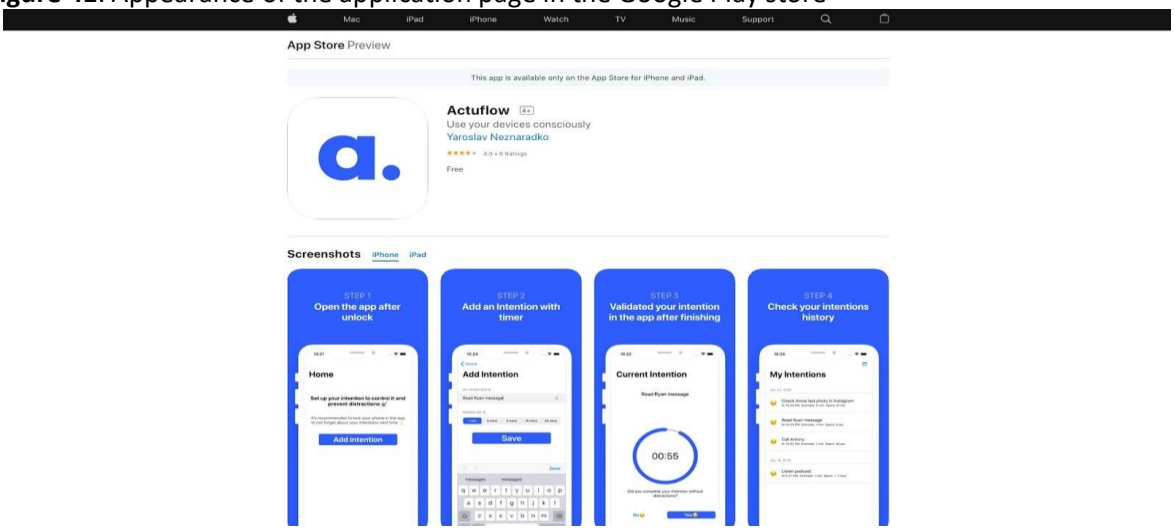


Figure 42: View of the application page in the App Store

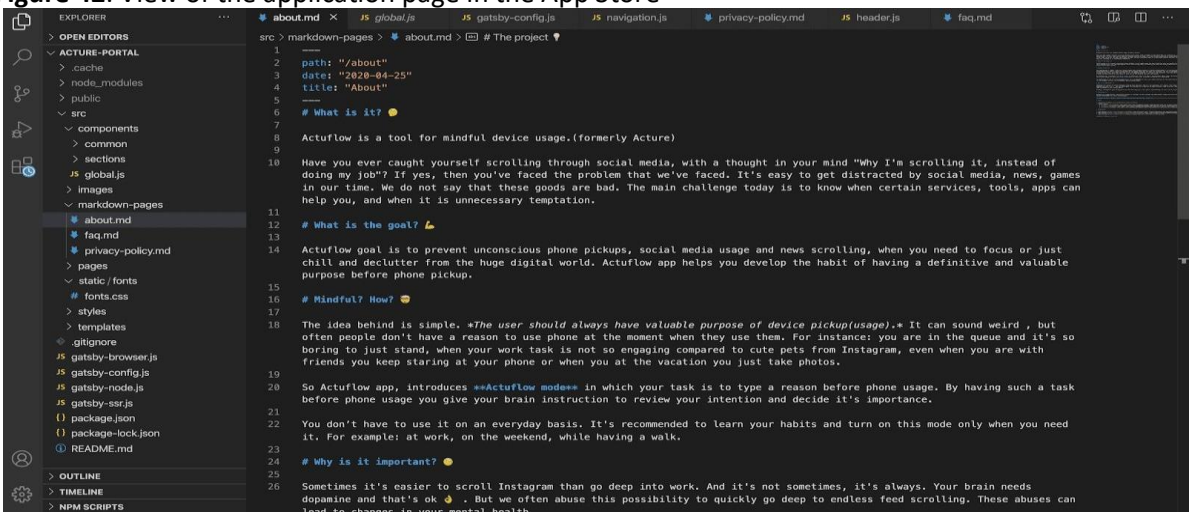


Figure 43: Application website structure (left) and sample content (right)

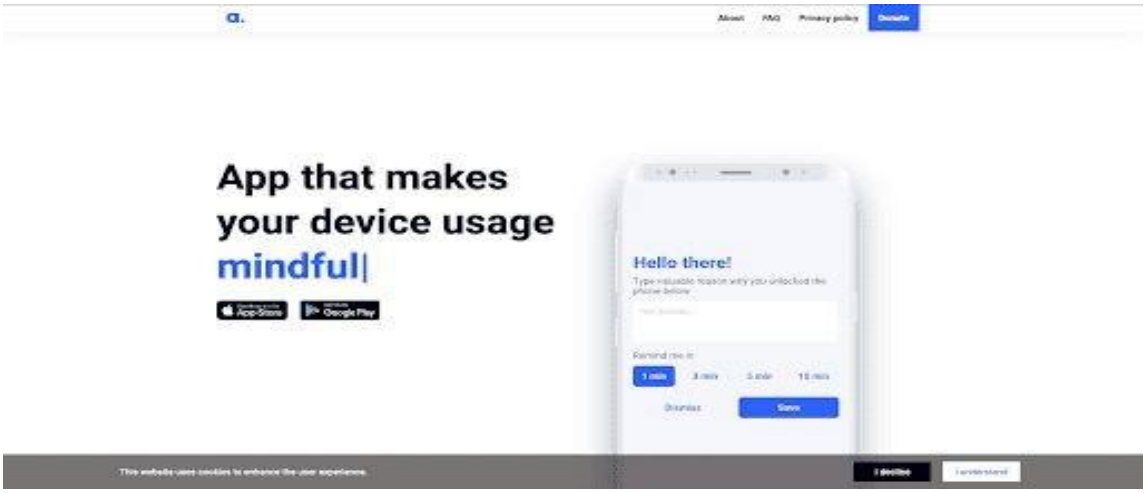


Figure 44: Appearance of the application website

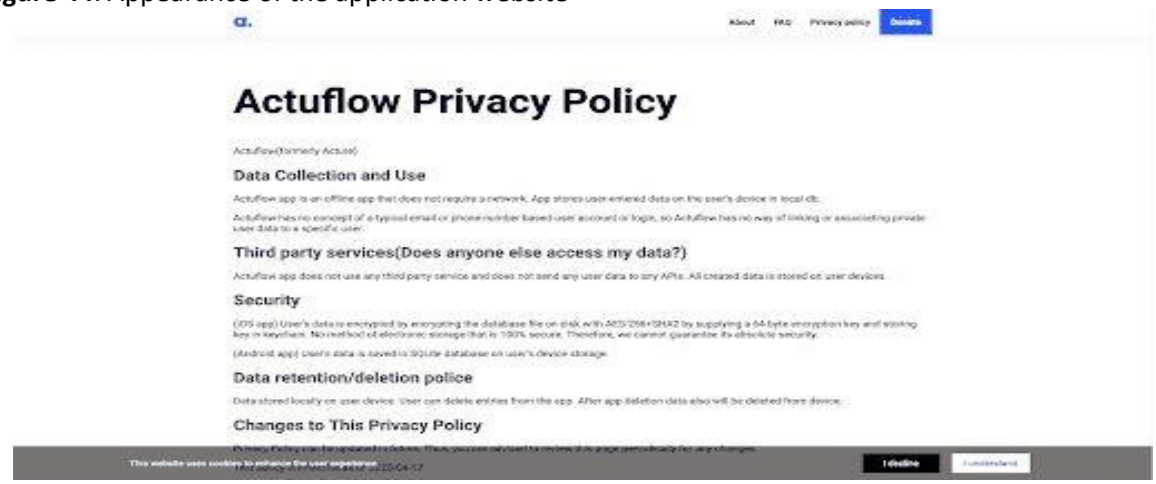


Figure 45: The privacy policy page view

4.1.8. Description of application and site analytics implementation

Analytics is an essential aspect in the development of the application because it provides information about users, based on which you can plan appropriate actions related to marketing [87-91]. But keep in mind the privacy of users and always notify users of the activities that your program or site, or application will perform with their data. Many companies and their products ignore this and create a false impression of specific categories of applications. That is why it was decided not to have additional analytics of user actions in the application in the early stages and use only the information provided by Google Play and App Store platforms and connect other analytics only to the website. The Google Play Console platform offers a more detailed analysis of applications compared to the App Store Connect. Still, the primary indicators in terms of the number of downloads, the number of reviews, traffic sources, the number of crashes are enough for the first versions of the application. Therefore, the Google Analytics service, a leader in website analytics, has been connected to the site.

4.2. Analysis of the obtained results

The Actuflow app on Android has been downloaded more than 10,000 times, on iOS about 8,000 times. In the Google Play app store, Actuflow has a rating of 4.6, in the App Store has a rating of 4.4. In total, about 130 text reviews were left in application stores on both platforms. During the promotion of the application and marketing, the application was recognized as one of the best applications of the month by well-known YouTube channels and the international edition of the Android Authority. Countries where the application is most popular USA, India, UK, Switzerland.

Figure 46: Google Play Console Analytics Bar

Figure 47: Analytics panel in the App Store Connect

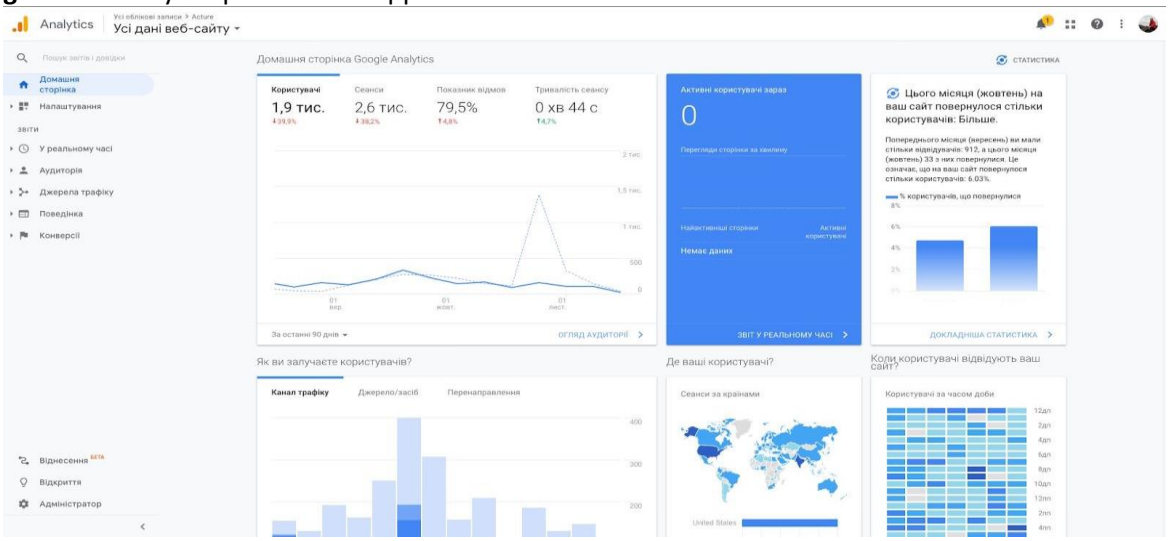


Figure 48: Google Analytics Dashboard

Due to its simplicity and relativity of various registrations and binding to the actual user, the application is widespread. The application is effortless and concise. Of course, this may change over time, but it is due to the availability of only the necessary features that users use the application as they need, based on their scripts. A study was conducted to measure screen time with some users of the application. Users used the application during the week on both systems and reported on their progress at the end of the week. On average, the screen time decreased by 30-40%. On the Reddit platform and in reviews of the app in stores and during beta testing, users reported and asked about new features, but at the same time emphasized that the app is self-sufficient even without new features.

Users asked about:

- Analytics (implemented on Android, iOS version in progress);
- Ability to schedule Actuflow mode for a specific hour (Android);
- Monitor user-opened applications;
- Ability to choose a goal instead of writing it.

The developed application entirely fulfilled the expected effects from its release. The most important result is to draw attention to the problem and motivation for further development. The application has become a kind of experiment that users can do with their smartphone use to understand and analyse their actions and usage habits. This chapter describes the practical implementation of the Actuflow application system. The principle and operation of the primary sequences of the system were demonstrated and explained in detail and with screenshots of the actual application. The implementation of additional tools such as analytics was described. The results of the performance were also analysed based on honest user feedback.

5. Conclusions

The master's thesis is an information management system to help users reduce distractions in smartphones. Google Play and App Store published this system. A website has also been created about the application and the methodology that the application implements.

An analysis of the issues and existing applications was conducted. Five applications were considered, which were compared by several qualitative and quantitative parameters (Availability on platforms, implemented methodology, essential functions, additional functions, privacy and transactions with user data, price of use or monetization, relative efficiency (how much to use the application and how to get the result), design, number of installations and user rating).

A system analysis was performed during which it was determined that the system is information control. Furthermore, according to the type of system the problem was posed and substantiated, the purpose of development, place and scope and expected effects from implementation were described.

A system's conceptual model has been developed using several UML diagrams (sequences, components, activity, states, and classes). Within the conceptual model, functional and non-functional requirements were formed. The conditions were also described in the user history format.

It was also justified to use the method of planning the use of a smartphone and using specific tools to implement the system. The last section described the implementation of the system. Finally, it analysed the results, which showed that the methodology works and that users positively received the implementation method, emphasising simplicity and conciseness worldwide.

6. References

[1] November 2020 Mobile User Statistics: Discover the Number of Phones in The World & Smartphone Penetration by Country or Region. URL: https://www.bankmycell.com/blog/how-many-phones-are-in-the-world

[2] Kevin Jackson A brief history of the smartphone. URL: https://sciencenode.org/feature/How%20did%20smartphones%20evolve.php

[3] Google Play. Google Play Store official website. URL: https://play.google.com/store.

[4] App Store. Official Apple App Store website. URL: https://www.apple.com/app-store/.

[5] Everything you need to know about 4G Network. Techspot. URL: https://www.techspot.com/guides/272-everything-about-4g/page3.html
[6] Number of smartphones sold to end users worldwide from 2007 to 2021 (in million units). URL: https://www.statista.com/statistics/263437/global-smartphone-sales-to-end-users-since-20 07/
[7] App and games categories on Google Play. Play Console Docs. URL: https://support.google.com/googleplay/android-developer/answer/9859673?hl=uk&visit_i d=637403592686389641-3668062092&rd=1
[8] Screen Time: UK vs US vs The Rest of the World Compared. URL: https://businessfibre.co.uk/screen-time/
[9] Average Time Spent Daily on Social Media (Latest 2020 Data). BroadbandSearch. URL: https://www.broadbandsearch.net/blog/average-daily-time-on-social-media
[10] Are Smartphones and Social Media Hurting Our Kids? Institute for family studies. URL: https://ifstudies.org/blog/are-smartphones-and-social-media-hurting-our-kids
[11] Social media copies gambling methods 'to create psychological cravings'. TheGuardian. URL: https://www.theguardian.com/technology/2018/may/08/social-media-copies-gambling-methods-to-create-psychological-cravings
[12] Dopamine, Smartphones & You: A battle for your time. Harvard University The Graduate School of Art. URL: http://sitn.hms.harvard.edu/flash/2018/dopamine-smartphones-battle-time
[13] The Social Dilemma. Official website. URL: https://www.thesocialdilemma.com/
[14] Social Network Ad Spending to Hit $23.68 Billion Worldwide in 2015. Insider intelligence. URL: https://www.emarketer.com/Article/Social-Network-Ad-Spending-Hit-2368-Billion-World wide-2015/1012357
[15] I Was a Screen–Time Expert. Then the Coronavirus Happened. New York Times. URL: https://www.nytimes.com/2020/07/27/parenting/children-screen-time-games-phones.html
[16] N. G. Carr, The Shallows: What the Internet Is Doing to Our Brains, 2010.
[17] C. Newport, Deep Work: Rules for Focused Success in a Distracted World By, 2016.
[18] C. Newport, Digital Minimalism: Choosing a Focused Life in a Noisy World By, 2019.
[19] G. Mark, D. Gudith, U. Klocke, The Cost of Interrupted Work: More Speed and Stress. URL: https://www.ics.uci.edu/~gmark/chi08-mark.pdf
[20] Will turning your phone to greyscale really do wonders for your attention? TheGuardian. URL: https://www.theguardian.com/technology/2017/jun/20/turning-smartphone-greyscale-atten tion-distraction-colour
[21] S. Trine, E. Gunn. Digital detox: Media resistance and the promise of authenticity. Convergence : journal, 2019, p. 1354856519847325. doi:10.1177/1354856519847325
[22] F. A. Alshobaili, N. A. AlYousefi, The effect of smartphone usage at bedtime on sleep quality among Saudi non- medical staff at King Saud University Medical City, volume 8(6) of Journal of family medicine and primary care, 2019, pp. 1953-1957. https://doi.org/10.4103/jfmpc.jfmpc_269_19
[23] Atsunori Ariga, Alejandro Lleras, Brief and rare mental 'breaks' keep you focused: Deactivation and reactivation of task goals preempt vigilance decrements, in: Cognition, 2011, DOI:10.1016/j.cognition.2010.12.007
[24] D. H. Steinberg, D. W. Palmer, Extreme Software Engineering, Pearson Education, Inc., ISBN 0-13-047381-2
[25] Google Digital Wellbeing. URL: https://wellbeing.google/
[26] Apple Screen Time. URL: https://support.apple.com/en-us/HT210387
[27] Moment iOS. URL: https://inthemoment.io/
[28] Forest app. URL: https://www.forestapp.cc/
[29] Before launcher. URL: https://www.beforelabs.com/
[30] Y. Dragan, M. Medykovskyy, L. Sikora, B. Yavorskyy, System analysis concepts and principles of construction mathematical model study object, in: International Conference - The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM, 2011, p. 34.
[31] T. Saaty, K. Cairns, Analytical Planning. Organization of systems. Moscow: Radio i svyaz, 1991.
[32] N. Schahovska, Datawarehouse and dataspace information base of decision support systemб in: 11th International Conference - The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM, 2011, pp. 170-173.

[33] R.P. Strubytskyi, N.B. Shakhovska, Analysis of approaches to modeling of cloud data warehouses, volume 149(11) of Actual Problems of Economics, 2013, pp. 263-269.
[34] N. Shakhovska, N. Boyko, P. Pukach, The Information Model of Cloud Data Warehouses, volume 871 of Advances in Intelligent Systems and Computing, 2019, pp. 182-191.
[35] N. Boyko, N. Shakhovska, Prospects for Using Cloud Data Warehouses in Information Systems, in: Conference on Computer Sciences and Information Technologies, CSIT, 2018, pp. 136-139.
[36] O. Veres, Allocation algorithms of basic resources attached to concluding of sparring Scheduling employments, in: The Experience of Designing and Application of CAD Systems in Microelectronics - Proceedings of the 6th International Conference, CADSM, 2001, pp. 263-264.
[37] O.O. Semenova, et al., Genetic ANFIS for scheduling in telecommunication networks Proceedings of SPIE - The International Society for Optical Engineering, 2018, 10808.
[38] D. Ruano-Ordás, V. Basto-Fernandes, I. Yevseyeva, J.R. Méndez, Evolutionary multi-objective scheduling for anti-spam filtering throughput optimization, volume 10334 of Lecture Notes in Computer Science, 2017, pp. 137-148.
[39] A. Gozhyj, I. Kalinina, V. Vysotska, S. Sachenko, R. Kovalchuk, Qualitative and Quantitative Characteristics Analysis for Information Security Risk Assessment in E-Commerce Systems, volume Vol- 2762 of CEUR Workshop Proceedings, 2020, pp. 177-190.
[40] O. Bisikalo, O. Kovtun, V. Kovtun, V. Vysotska, Research of Pareto-Optimal Schemes of Control of Availability of the Information System for Critical Use, volume Vol-2623 of CEUR Workshop Proceedings, 2020, pp. 174-193.
[41] M. Bublyk, V. Vysotska, L. Chyrun, V. Panasyuk, O. Brodyak, Assessing Security Risks Method in E-Commerce System for IT Portfolio Management, volume Vol-2853 of CEUR Workshop Proceedings, 2021, pp. 462-479.
[42] O. Garasym, L. Chyrun, N. Chernovol, A. Gozhyj, V. Gozhyj, I. Kalinina, B. Rusyn, L. Pohreliuk, M. Korobchynskyi, Network Security Analysis Based on Consolidated Threat Resources, volume Vol-2604 of CEUR workshop proceedings, 2020, pp. 1004-1018.
[43] M. Nazarkevych, I. Klyujnyk, H. Nazarkevych, Investigation the Ateb-Gabor Filter in Biometric Security Systems, in: Data Stream Mining & Processing (DSMP), 2018, pp. 580-583.
[44] M. Nazarkevych, I. Klyujnyk, I. Maslanych, B. Havrysh, H. Nazarkevych, Image filtration using the Ateb-Gabor filter in the biometric security systems, in: International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH), 2018, pp. 276-279.
[45] S. Sachenko, S. Rippa, I. Golyash, Improving the Information Security Audit of Enterprise Using XML Technologies, in: Inretnational Conference on Intelligent Data Acquisition and Advanced Computing Systems:Technology and Applications, 2011, pp. 933-937.
[46] O. Kehret, A. Walz, A. Sikora, Integration of Hardware Security Modules into a Deeply Embedded Tls Stack, volume 15(1) of International Journal of Computing, 2016, pp. 22-30.
[47] V. Kharchenko, Y. Ponochovnyi, A. Abdulmunem, A. Boyarchuk,Security And Availability Models For Smart Building Automation Systems, volume 16(4) of International Journal of Computing, 2017, pp. 194-202.
[48] A. Peleshchyshyn, V. Vus, S. Albota, O. Markovets, A Formal Approach to Modeling the Characteristics of Users of Social Networks Regarding Information Security Issues, volume 902 of Advances in Intelligent Systems and Computing, 2020, pp. 485-494.
[49] R. Hryshchuk, K. Molodetska, Y. Syerov, Method of improving the information security of virtual communities in social networking services, volume Vol-2392 of CEUR Workshop Proceedings, 2019, pp. 23-41.
[50] P. Zhezhnych, T. Burak, An approach to multiple security system development using database schemas, volume 1080 of Advances in Intelligent Systems and Computing, 2020, pp. 771-780.
[51] V. Lytvyn, V. Vysotska, I. Peleshchak, I. Rishnyak, R. Peleshchak, Time Dependence of the Output Signal Morphology for Nonlinear Oscillator Neuron Based on Van der Pol Model, volume 10 of International Journal of Intelligent Systems and Applications, 2018, pp. 8-17. DOI: 10.5815/ijisa.2018.04.02
[52] V. Tkachenko, O. Cherednichenko, M. Godlevskyi, The Concept of Device Meta-Model for Real-Time Communication in the Transboundary Environment Monitoring System, in: International Scientific-Practical Conference, PIC S&T, 2018, pp. 64-70.

[53] V. Lytvynenko, et al.: Hybrid Methods of GMDH-Neural Networks Synthesis and Training for Solving Problems of Time Series Forecasting, volume 1020 of Lecture Notes in Computational Intelligence and Decision Making, 2020, pp. 513-531.
[54] P. Pukach, On the unboundedness of a solution of the mixed problem for a nonlinear evolution equation at a finite time, volume 14(3) of Nonlinear oscillations, 2012, pp. 369-376.
[55] Z.M. Nytrebych, V.S. Ilkiv, P.Ya. Pukach, O.M. Malanchuk, Homogeneous Problem with Two-point Conditions in Time for Some Equations of Mathematical Physics, volume 7(2) of Azerbaijan Journal of Mathematics, 2017, pp. 180-196.
[56] P. Pukach, V. Il'kiv, Z. Nytrebych, M. Vovk, On nonexistence of global in time solution for a mixed problem for a nonlinear evolution equation with memory generalizing the Voigt – Kelvin rheological model, volume 37(5) of Opuscula Mathematica, 2017, pp. 735-753.
[57] O. Malanchuk, Z. Nytrebych, V. Il'kiv, P. Pukach, On the solvability of two-point in time problem for PDE, volume 38 of Italian Journal of Pure and Applied Mathematics, 2017, pp. 715-726.
[58] M. Melnyk, A. Kernyskyy, P. Pukach, Development of Subsystems for Reverberation Time Definition in Lecture Auditorium, in: The Experience of Designing and Application of CAD Systems in Microelectronics, (CADSM), 2017, pp. 354-356.
[59] H. Dmitriv, R. Kaminsky, Two algorithms median filtering to identify the time series trend, volume 512 of Advances in Intelligent Systems and Computing, 2017, pp. 283-292.
[60] A. Batyuk, V. Voityshyn, Apache storm based on topology for real-time processing of streaming data from social networks, in: Data Stream Mining and Processing, DSMP, 2016, pp. 345-349.
[61] P. Zhezhnych, A. Peleschychyn, Time aspects of information systems, in: The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM, 2007, pp. 530-533.
[62] R. Kaminsky, L. Mochurad, N. Shakhovska, N. Melnykova, Calculation of the Exact Value of the Fractal Dimension in the Time Series for the Box-Counting Method, in: 9th International Conference on Advanced Computer Information Technologies, ACIT, 2019, pp. 248-251.
[63] S. Bezobrazov, A. Sachenko, M. Komar, V. Rubanau, The Methods of Artificial Intelligence for Malicious Applications Detection in Android OS, volume 15(3) of International Journal of Computing, 2016, pp. 184-190.
[64] M. Schölzel, E. Eren, K. Detken, L. Schwenke, Monitoring Android Devices By Using Events And Metadata, volume 15(4) of International Journal of Computing, 2016, pp. 248-258.
[65] D. Ostrovka, V. Teslyuk, The Analysis of File Format Conversion Tools for Storing 3D Objects for the iOS Platform, volume Vol-2631 of CEUR Workshop Proceedings, 2020, pp. 341-350.
[66] N. Antonyuk, M. Medykovskyy, L. Chyrun, M. Dverii, O. Oborska, M. Krylyshyn, A. Vysotsky, N. Tsiura, O. Naum, Online Tourism System Development for Searching and Planning Trips with User's Requirements, volume 1080 of Advances in Intelligent Systems and Computing IV, Springer Nature Switzerland AG 2020, 2020, pp. 831-863.
[67] M. Medykovskyy, O. Shunevych, Optimal choice of wind turbines combination based on customer's requirements, in: Modern Problems of Radio Engineering, Telecommunications and Computer Science, TCSET, 2010, pp. 319.
[68] O. Cherednichenko, M. Vovk, O. Yanholenko, O. Yakovleva, Towards The Technology of Employers' Requirements Collection Development, in: Integrated Computer Technologies in Mechanical Engineering, 2020, pp. 228-239.
[69] T. Basyuk, The Popularization Problem of Websites and Analysis of Competitors, volume 689 of Advances in Intelligent Systems and Computing, Springer, Cham, 2018, pp. 54-65.
[70] T. Basyuk, Innerlinking website pages and weight of links, in: International Scientific and Technical Conference on Computer science and information technologies, 2017, pp. 12-15.
[71] T. Basyuk, Popularization of website and without anchor promotion, in: International Scientific and Technical Conference on Computer science and information technologies, 2016, pp. 193-195.
[72] O. Berezko, P. Zhezhnych, Rethinking the NGO website from the knowledge management perspective, in: Proceedings of the 12th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 2017, pp. 389-392.
[73] V. Senkivskyy, I. Pikh, S. Babichev, A. Kudriashova, N. Senkivska, Modeling of alternatives and defining the best options for websites design, vol. 2853 of CEUR Workshop Proceedings, 2021.

[74] P. Zhezhnych, O. Markiv, Linguistic comparison quality evaluation of web-site content with tourism documentation objects, volume 689 of Advances in Intelligent Systems and Computing, 2018, pp. 656-667.
[75] P. Zhezhnych, O. Markiv, A linguistic method of web-site content comparison with tourism documentation objects, in: International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 2017, pp. 340-343.
[76] S. Albota, Linguistically Manipulative, Disputable, Semantic Nature of the Community Reddit Feed Post, volume Vol-2870 of CEUR Workshop Proceedings, 2021, pp. 769-783.
[77] S. Albota, Resolving Conflict Situations in Reddit Community Driven Discussion Platform, volume Vol-2604 of CEUR workshop proceedings, 2020, pp. 215-226.
[78] B. Rusyn, L. Pohreliuk, A. Rzheuskyi, R. Kubik, Y. Ryshkovets, L. Chyrun, S. Chyrun, A. Vysotskyi, V. B. Fernandes, The Mobile Application Development Based on Online Music Library for Socializing in the World of Bard Songs and Scouts' Bonfires, volume 1080 of Advances in Intelligent Systems and Computing, 2020, pp. 734-756. DOI: 10.1007/978-3-030-33695-0_49
[79] A. Rzheuskyi, A. Gozhyj, A. Stefanchuk, O. Oborska, L. Chyrun, O. Lozynska, K. Mykich, T. Basyuk, Development of Mobile Application for Choreographic Productions Creation and Visualization, volume Vol-2386 of CEUR Workshop Proceedings, 2019, pp. 340-358.
[80] T. Teslyuk, V. Teslyuk, H. Lypak, N. Kunanets, N. Veretennikova, A Mobile Museum Guide Application, volume Vol-2631 of CEUR Workshop Proceedings, 2020, pp. 314-326.
[81] O. Artemenko, V. Pasichnyk, N. Kunanets, K. Shunevych, Using sentiment text analysis of user reviews in social media for e-tourism mobile recommender systems, volume Vol-2604 of CEUR workshop proceedings, 2020, pp. 259-271.
[82] N. Shakhovska, S. Fedushko, ml. M. Greguš, I. Shvorob, Yu. Syerova, Development of Mobile System for Medical Recommendations, in: The 15th International Conference on Mobile Systems and Pervasive Computing (MobiSPC), 2019, pp. 43-50.
[83] E., Vasilevskis, I., Dubyak, T., Basyuk, V., Pasichnyk, A.: Rzheuskyi, Mobile application for preliminary diagnosis of diseases, vol. 2255 of CEUR Workshop Proceedings, 2018, pp. 275-286.
[84] R. E. Hiromoto, A. Sachenko, V. Kochan, V. Koval, V. Turchenko, O. Roshchupkin, V. Yatskiv, K. Kovalok, Mobile Ad Hoc Wireless Networkfor Pre- and Post-Emergency Situations in Nuclear Power Plant, in: International Symposium on Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems, 2014, pp. 2-96.
[85] M. Davydov, O. Lozynska, Information system for translation into Ukrainian sign language on mobile devices, in: International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 2017, pp. 48-51.
[86] N. Shakhovska, S. Fedushko, M. Gregušml, I. Shvorob, Y. Syerov, Development of mobile system for medical recommendations, volume 155 of Procedia Computer Science, 2019, pp. 43-50.
[87] P. Kravets, The control agent with fuzzy logic, in: Perspective Technologies and Methods in MEMS Design, MEMSTECH, 2010, pp. 40-41.
[88] P. Kravets, R. Kyrkalo, Fuzzy logic controller for embedded systems, in: International Conference on Perspective Technologies and Methods in MEMS Design, MEMSTECH, 2009, pp. 58-59.
[89] O. Barmak, I. Krak, E. Manziuk, V. Lytvynenko, O. Kalyta, Classification technology based on hyperplanes for Visual analytics with implementations for different subject areas, volume 2623 of CEUR Workshop Proceedings, 2020, pp. 96-106.
[90] M.O. Medykovskyi, I.G. Tsmots, Y.V. Tsymbal, Information analytical system for energy efficiency management at enterprises in the city of Lviv (Ukraine), volume 175(1) of Actual Problems of Economics, 2016, pp. 379-384.
[91] L. Chyrun, P. Kravets, O. Garasym, A. Gozhyj, I. Kalinina, Cryptographic information protection algorithm selection optimization for electronic governance IT project management by the analytic hierarchy process based on nonlinear conclusion criteria, volume 2565 of CEUR Workshop Proceedings, 2020, pp. 205-220.