

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

До захисту допущено:

Завідувач кафедри

_____ Олександр РОЛІК

«__» _____ 20__ р.

Дипломний проєкт

на здобуття ступеня бакалавра

за освітньо-професійною програмою «Інформаційні управляючі системи та технології»

спеціальності 126 «Інформаційні системи та технології»

на тему: «Мобільний застосунок для контролю проведеного часу активності та відпочинку»

Виконав (-ла):

студент (-ка) IV курсу, групи ІС-93

Сокольницька А.В.

Керівник:

старший викладач ІСТ

Анікін В.К.

Рецензент:

Доцент кафедри ІПІ, к.т.н

Ліхоузова Т.А.

Засвідчую, що у цьому дипломному проєкті немає запозичень з праць інших авторів без відповідних посилань.

Студент (-ка) _____

Київ – 2023 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 126 «Інформаційні системи та технології»

Освітньо-професійна програма «Інформаційні управляючі системи та технології»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Олександр РОЛІК

«___» _____ 20__ р.

ЗАВДАННЯ

на дипломний проєкт студенту
Сокольніцькій Анні Вікторівни

1. Тема проєкту «Мобільний застосунок для контролю проведенного часу активності та відпочинку», керівник проєкту Анікін Володимир Костянтинович, старший викладач, затверджені наказом по університету від «31» червня 2023 р. №2101-с
2. Термін подання студентом проєкту: 12 червня 2023 року.
3. Вихідні дані до проєкту: проблема із контролюванням часу проведеним за роботою та відпочинком.
4. Зміст пояснювальної записки:
 1. Загальні положення: основні визначення та терміни, опис предметного середовища, огляд ринку програмних продуктів, постановка задачі.
 2. Інформаційне забезпечення: вхідні дані, вихідні дані, опис структури бази даних
 3. Математичне забезпечення: змістовна та математична постановки задачі, обґрунтування та опис методу розв'язання.
 4. Програмне та технічне забезпечення: засоби розробки, вимоги до технічного забезпечення, архітектура програмного забезпечення, побудова звітів.
5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо) :
 - 1) Алгоритм роботи застосунку;

- 2) Взаємодія системи із користувачем;
 - 3) Діаграма класів;
 - 4) Діаграма послідовності процесів.
6. Дата видачі завдання: 1 березня 2023 року.

Календарний план

№ з/п	Назва етапів виконання дипломного проєкту	Термін виконання етапів проєкту	Примітка
1.	Вивчення рекомендованої літератури	20.03.2023	
2.	Аналіз існуючих методів розв'язання	26.03.2023	
3.	Постановка задачі	29.03.2023	
4.	Розробка інформаційного забезпечення	02.04.2023	
5.	Алгоритмізація задачі	03.04.2023	
6.	Розробка програмного забезпечення	21.04.2023	
7.	Налагодження програми	11.05.2023	
8.	Оформлення пояснювальної записки	19.05.2023	
9.	Подання ДП на попередній захист	05.06.2023	

Студент

Анна СОКОЛЬНИЦЬКА

Керівник

Володимир АНІКІН

АНОТАЦІЯ

Структура та обсяг роботи. Анотація до дипломного проєкту складається з п'яти розділів, містить 28 рисунків, 13 таблиць, 16 джерел.

Ключові слова: контроль часу роботи та відпочинку, мобільний застосунок для контролювання робочого часу та відпочинку.

Дипломний проєкт присвячений розробці додатку для контролювання проведеного часу за роботою та відпочинком. Метою створення даного додатку є підвищення продуктивності робочого дня та уникнення перевтомленості.

Для досягнення поставленої мети необхідно реалізувати такі задачі:

- реєстрація користувачів;
- створення таймеру;
- введення параметрів для таймеру;
- ведення параметрів для типу роботи;
- введення параметрів для повідомлень щодо відпочинку ;
- коригування параметрів таймеру та повідомлень;
- перегляд варіантів проведення відпочинку.

У розділі інформаційного забезпечення наведені вхідні та вихідні дані додатку.

Розділ алгоритм роботи застосунку присвячений вирішення основної задачі тобто роботи додатку та алгоритму його роботи.

У розділі програмного забезпечення обґрунтовано обрані засоби розробки: Android SDK, Java. Також наведено діаграму класів та опис кожного класу, а також детальний опис усіх функцій програми. Наведено діаграми послідовності та діаграми компонентів даного проєкту.

У технологічному розділі описано інструкцію користувача, де наведений перелік дій, які може виконувати користувач. Також вказані випробування системи, набори тестів, які дозволяють перевірити правильність роботи системи.

SUMMARY

Structure and scope of work. The explanatory note of the diploma project consists of five sections, contains 28 figures, 13 tables, and 16 sources.

Keywords: Time control, work and rest, mobile application for control of working time, interactive application for control of working time and rest.

The diploma project is devoted to the development of an application for monitoring the time spent at work and rest. The purpose of creating this application is to increase the productivity of the working day and avoid fatigue.

To achieve the set goal, the following tasks must be implemented:

- user registration;
- creating a timer;
- entering parameters for the timer;
- maintaining parameters for the type of work;
- entering parameters for notifications about rest;
- adjustment of timer parameters and messages;
- review options for recreation.

The input and output data of the application are listed in the information support section.

The application algorithm section is devoted to the solution of the main task, i.e. the operation of the application and the algorithm of its operation.

In the software section, the development tools are reasonably chosen: Android SDK, Java. Also provided is a class diagram and description of each class, as well as a detailed description of all program features. Sequence diagrams and component diagrams of this project are given.

The technological section describes the user manual, which lists the actions that the user can perform. System tests, test sets that allow you to check the correct operation of the system are also indicated.

[illegible]

**Пояснювальна записка
до дипломного проєкту
на тему: «Мобільний застосунок для контролю
проведеного часу активності та відпочинку»**

ЗМІСТ

ВСТУП.....	4
1 ЗАГАЛЬНІ ПОЛОЖЕННЯ	5
1.1 Опис предметного середовища	5
1.1.1 Опис процесу діяльності.....	7
1.1.2 Опис функціональної моделі.....	7
1.2 Огляд існуючих аналогів	9
1.3 Постановка задачі	16
2 ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ	18
2.1 Вхідні дані.....	18
2.2 Вихідні дані.....	18
2.3 Опис структури.....	24
3 АЛГОРИТМ РОБОТИ ЗАСТУСУНКУ	26
3.1 Змістовна постановка задачі.....	26
3.2 Математична постановка задачі.....	26
3.3 Обґрунтування алгоритму розв'язання.....	27
4 ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ	34
4.1 Засоби розробки.....	34
4.2 Вимоги до технічного забезпечення.....	36
4.3 Архітектура програмного забезпечення.....	37
4.3.1 Архітектура ПЗ	37
4.3.2 Діаграма класів	37
4.3.2 Діаграма послідовності	41
4.3.3 Діаграма компонентів	46
4.3.4 Специфікація функцій.....	47
5 ТЕСТУВАННЯ СИСТЕМИ.....	51
5.1 Керівництво користувача.....	51
5.2 Випробування програмного продукту.....	56

					ІС93.240БАК.004 ПЗ				
Зм.	Лист	№ докум.	Підп.						
		Сокольніцька А. В.			Мобільний застосунок для контролю проведеного часу активності та відпочинку. Пояснювальна записка		Літ.	Арк.	Аркушів
Перевір		Анікін В. К.					Т	2	64
							КПІ ім. Ігоря Сікорського Група ІС-93		
Затв.									

5.2.1 Мета випробувань та загальні положення випробувань	56
5.2.2 Результати випробувань	57
ВИСНОВКИ	62
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	63

					ІС93.240БАК.004 ПЗ	Арк.
						3
Зм.	Лист	№ докум.	Підпис	Дата		

ВСТУП

Сьогодні середньостатистична людина має дуже багато справ і дуже мало часу на відпочинок. Кожен день ми намагаємося зробити якомога більше, тому вміння контролювати свій час потрібно кожному із нас.

Але не всі можуть сліdkувати за співвідношенням між кількістю часу для роботи та відпочинку. Створення додатку для контролювання часу, проведеного за роботою та відпочинком, є важливим і корисним інструментом для багатьох людей. Такий додаток може нагадувати користувачеві про необхідність відпочивати та забезпечувати баланс між роботою і часом для себе. Програмний продукт має різні можливості, такі як інтерфейс, функціонал, вхід до вашого аккаунту, вибір діяльності та вид відпочинку.

Такі програми, як Motivate Clock, Toggl, ATracker пропонують свої послуги для контролю свого часу за роботою та відпочинком. Ми маємо можливість вказати час, який буде відведено на роботу, або побачити звіт по використанню тих чи інших програм за весь робочий день.

Для створення цього програмного продукту було вирішено на темі контролю проведеного часу за роботою та відпочинком. Метою створення даної програми є уникнення перевтоми в робочий час, уникнення погіршень здоров'я за довгим проведенням у сидячому положенні та навантаження на очі рахунок встановлення таймера на час роботи, вибору відведеного часу на відпочинок та вид відпочинку.

Практичне значення одержаних результатів це розроблене програмне забезпечення для контролювання проведеного часу за роботою та відпочинком за рахунок таймера та нагадування про відпочинок.

					ІС93.240БАК.004 ПЗ	Арк.
						4
Зм.	Лист	№ докум.	Підпис	Дата		

1 ЗАГАЛЬНІ ПОЛОЖЕННЯ

1.1 Опис предметного середовища

Погляньмо на зміну звичайного робочого дня на сьогоднішній. З кожним днем ми все більше і більше повинні проводити свій час за роботою, навчанням та навіть часом проведеним за комп'ютером, телефоном та іншими гаджетами або пристроями.

Проблема не контролювання часу проведеного за роботою і забування про відпочинок є поширеною серед багатьох людей. Відсутність ефективного управління часом може призвести до перенавантаження, втоми, стресу і негативного впливу на фізичне та психічне здоров'я.

Коли людина не контролює свій час, вона може занурюватися в безкінечні завдання, пропускаючи перерви і відпочинок. Це може призвести до втрати енергії, зниження продуктивності і збільшення ризику вигорання. Недостатній відпочинок може також вплинути на якість сну та загальний стан здоров'я.

Крім того, неефективне управління часом може вплинути на баланс між роботою і особистим життям. Людина може витратити більше часу на роботу, залишаючи мало часу для сім'ї, друзів і хобі. Це може призвести до відчуття відчуженості, незадоволеності та вплинути на стосунки з близькими людьми.

Не контролювати час за роботою і відпочинком означає втрату контролю над своїм життям. Людина може почуватися захопленою роботою, без змоги зосередитися на інших аспектах життя. Це може вплинути на особисті цілі, саморозвиток і загальне щастя.

Тому, важливо навчитися ефективно контролювати час і надавати достатньо уваги як роботі, так і відпочинку, забезпечуючи баланс і гармонію в своєму житті.

Ефективний таймер робочого процесу з можливістю нагадування про перерви є запорукою ефективного робочого дня та міцного здоров'я.

					ІС93.240БАК.004 ПЗ	Арк.
						5
Зм.	Лист	№ докум.	Підпис	Дата		

Додаток для контролювання часу за роботою та відпочинком може бути корисним інструментом для людини з декількома причинами.

Перш за все, в сучасному світі, де швидкість життя зростає, інформація безперервно надходить і відволікає, ефективне управління часом стає все

важливішим. Додаток для контролю часу допомагає створити баланс між роботою та відпочинком, що дозволяє бути більш продуктивним і уникнути перенавантаження.

Такий додаток може надавати можливості для створення розкладу робочого дня, встановлення цілей і трекінгу часу, що витрачається на різні завдання. Він нагадує про необхідні перерви, включаючи регулярні відпочинки, фізичні вправи або розслаблюючі паузи, що сприяє збереженню енергії та підвищенню продуктивності.

Контроль часу дозволяє краще організувати робочі процеси, пріоритети і завдання, допомагаючи уникнути втрати часу на непродуктивні справи. Він також допомагає зберегти баланс між роботою та особистим життям, що сприяє здоров'ю та знижує ризик вигорання.

Додаток може надавати аналітику та статистику про використання часу, що дозволяє користувачам оцінити свою продуктивність, виявити непродуктивні звички та вносити зміни для поліпшення робочого процесу.

В цілому, додаток для контролювання часу за роботою та відпочинком створює структуру, дисципліну та свідомість про використання часу, що допомагає досягти більшої продуктивності, ефективності та благополуччя у житті людини.

Отже, є потреба у розробці програми, що дозволить створювати всі комфортні умови для робочого дня із потрібною кількістю відпочинку.

Дане програмне забезпечення призначене для контролю робочого часу за роботою і відпочинком (тайм-трекер користувача з повідомленнями (нагадуваннями) про перерви за заданий проміжок часу). Метою є підвищення ефективності робочого дня та уникнення перенавантаження на

					ІС93.240БАК.004 ПЗ	Арк.
						6
Зм.	Лист	№ докум.	Підпис	Дата		

організм. Включення/виключення таймеру та вибір проміжку часу для нагадування про перерви обирає користувач, пропозиції варіантів проведення відпочинку (вправи для очей та всього тіла) забезпечуватиметься системою.

1.1.1 Опис процесу діяльності

Для створення таймеру та нагадувань про відпочинок спочатку користувач має увійти в систему. Для реєстрації і подальшої авторизації користувач має ввести логін, тобто електронну пошту та пароль (обирає сам користувач).

Після входу в облікову систему, користувач має вибір з двох кнопок Таймер та Відпочинок. Для включення таймеру, користувач повинен обрати Таймер, вид діяльності (робота, навчання або проведення часу за іграми), проміжок часу, через який будуть надходити повідомлення про відпочинок і кількість часу відведеного на відпочинок (5-30 хвилин). Після цього запускає таймер, йде відлік часу. Якщо робота була виконана раніше назначеного часу для відпочинку, є можливість зупинити таймер. У разі помилки або потреби в таймері ще, користувач може запустити таймер знову.

Якщо ж користувач отримав нагадування про відпочинок, він має можливість самостійно брати від відпочинку, або повернутись до головного меню, обрати Відпочинок, та переглянути запропоновані вправи для очей і тіла, що буде залежити від обраної типу роботи раніше. Після завершення відведеного часу на відпочинок, таймер продовжить відлік.

1.1.2 Опис функціональної моделі

Для опису функціональної моделі спочатку мають бути визначені дійові особи системи, тобто актори, та дії, які вони виконують. Виділимо актора системи: користувач.

					ІС93.240БАК.004 ПЗ	Арк.
						7
Зм.	Лист	№ докум.	Підпис	Дата		

Користувач: реєструється та авторизується в системі, переглядає меню програми, записує події на день, створює таймер, та вказує параметри, обирає тип діяльності, задає період часу для повідомлення та сам час відведений на відпочинок, обирає спосіб та переглядає надані програмою варіанти проведення відпочинку.

Функції користувача:

- реєстрація та авторизація в системі;
- запис подій на день;
- створення таймеру;
- корегування параметрів таймеру роботи;
- корегування роду діяльності;
- корегування параметрів таймеру відпочинку;
- перегляд варіантів відпочинку.

Усі функції, які виконує актор, мають пріоритети. Їх перелік наведено в таблиці 1.1

Таблиця 1.1 – Функції акторів та їх пріоритети

Актор	Функція	Пріоритет
Користувач	Реєстрація в системі	Високий
	Авторизація в системі	Високий
	Запис подій на день	Середній
	Створення таймеру	Високий
	Корегування параметрів таймеру роботи	Високий
	Корегування роду діяльності	Високий
	Корегування параметрів таймеру відпочинку	Високий
	Перегляд варіантів відпочинку	Середній

1.2 Огляд існуючих аналогів

Зараз є достатньо багато різноманітних програм та застосунків для планування свого часу. Більшість безкоштовних систем мають обмежений функціонал на відміну від платних. Але усі вони спрямовані на ефективну організацію робочого дня і чудово виконують свою задачу.

Для аналізу наявних аналогів розглянемо найбільш популярні:

- Toggl Track;
- Motivate Clock;
- ATracker;
- Everhour;
- RescueTime.

Розглянемо для початку додаток Toggl Track.

Toggl Track - це додаток для відстеження часу, який дозволяє користувачам точно виміряти і аналізувати час, витрачений на різні завдання та проекти. Ось кілька плюсів і недоліків цього додатку:

Плюси додатку:

- простота використання: Toggl Track має простий і інтуїтивно зрозумілий інтерфейс, що дозволяє швидко почати відстежувати час без складностей;
- гнучкість і налаштування: додаток дозволяє користувачам створювати різні проекти, завдання і теги, що дозволяє адаптувати його до індивідуальних потреб і вимог;
- інтеграція з іншими інструментами: Toggl Track підтримує інтеграцію з популярними інструментами та сервісами, такими як Asana, Trello, Google Calendar, що дозволяє легко синхронізувати дані і зручно використовувати додаток разом з іншими інструментами.

					IC93.240БАК.004 ПЗ	Арк.
						9
Зм.	Лист	№ докум.	Підпис	Дата		

Недоліки Toggl Track:

- обмежена безкоштовна версія: основні функції Toggl Track доступні в безкоштовній версії, але для отримання додаткових функцій потрібно підписатися на платний план;
- недостатня автоматизація: додаток вимагає вручного запуску таймера для початку відстеження часу, що може бути не так зручно для деяких користувачів;
- відсутність планування: Toggl Track сконцентрований на відстеженні витраченого часу, але не надає вбудованих інструментів для планування завдань і календарного управління.

Вигляд головного меню системи Toggl Track зображений на рисунку 1.1.

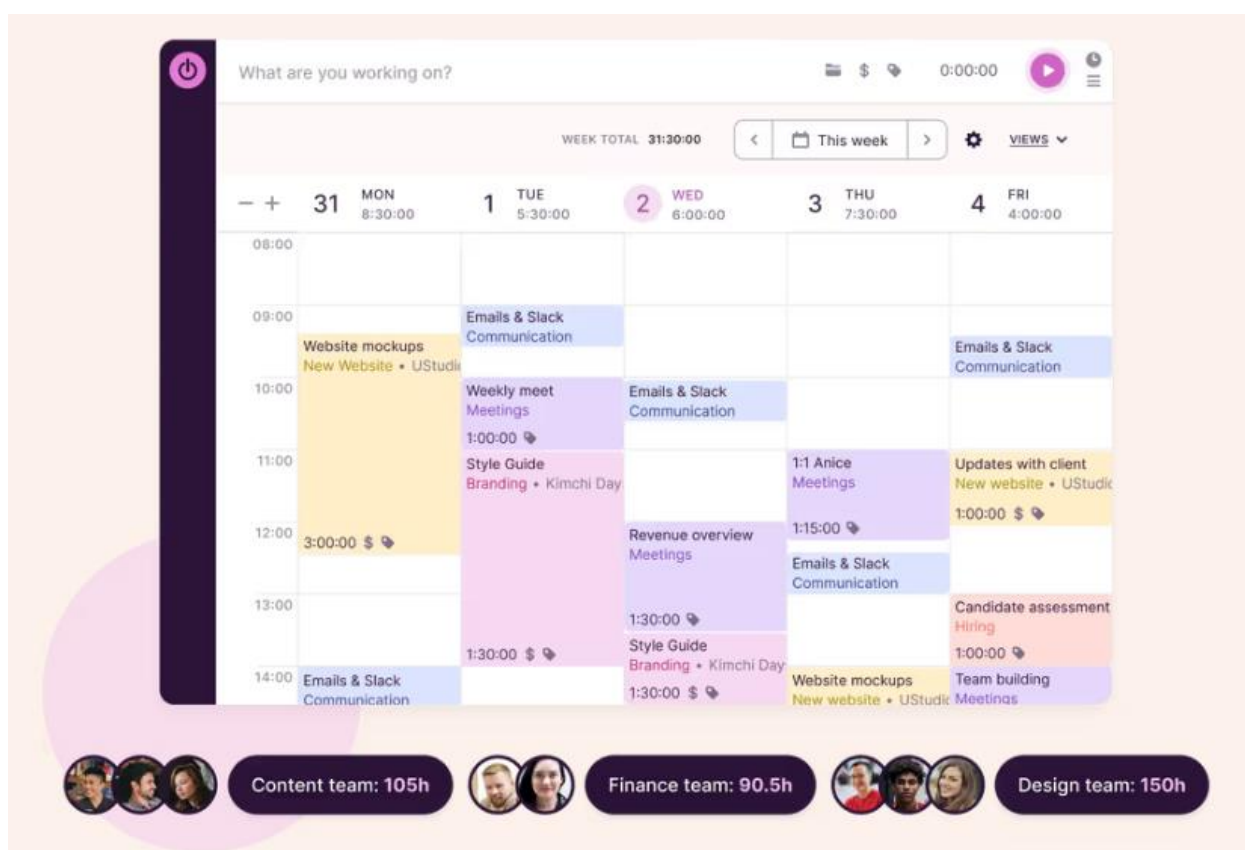


Рисунок 1.1 – Головне меню системи Toggl Track

Вигляд головного меню системи Toggl Track зображений на рисунку 1.1.

					ІС93.240БАК.004 ПЗ	Арк.
						10
Зм.	Лист	№ докум.	Підпис	Дата		

Розглянемо додаток Motivate Clock.

Motivate Clock - це додаток, який надає функціонал для контролю часу та мотивації користувачів. Ось кілька плюсів і недоліків цього додатку:

Плюси Motivate Clock:

- мотивація та нагадування: Motivate Clock надає мотивуючі повідомлення та нагадування, що допомагають користувачам залишатися фокусованими і виконувати свої завдання вчасно;

- гнучкість налаштування: додаток дозволяє налаштовувати цілі, пріоритети та тривалість завдань, а також встановлювати спеціальні параметри, такі як режими "продуктивності" або "відпочинку", щоб керувати своїм часом більш ефективно;

- статистика та аналітика: Motivate Clock надає користувачам детальну статистику та аналітику, що допомагає оцінити їх продуктивність, виявити тенденції та вдосконалити свої робочі звички.

Недоліки Motivate Clock:

- відсутність інтеграції з іншими інструментами: додаток може не підтримувати інтеграцію з популярними інструментами або сервісами, що може бути обмеженням для користувачів, які вже використовують інші інструменти для управління часом і завданнями;

- обмежений функціонал: у порівнянні з деякими іншими додатками, Motivate Clock може мати обмежені можливості та недостатній функціонал, особливо для вимогових користувачів;

- вплив на батарею та продуктивність пристрою: використання додатку може мати вплив на витрату заряду батареї та продуктивність пристрою, особливо якщо він постійно працює у фоновому режимі.

Вигляд головного меню системи Motivate Clock зображений на рисунку 1.2.

					ІС93.240БАК.004 ПЗ	Арк.
						11
Зм.	Лист	№ докум.	Підпис	Дата		



Рисунок 1.2 – Головне меню системи Motivate Clock

Розглянемо додаток Atracker.

Atracker - це додаток для відстеження часу, який допомагає користувачам контролювати свою продуктивність і ефективність. Ось кілька плюсів і недоліків цього додатку:

Плюси Atracker:

- простота використання: Atracker має простий і зручний інтерфейс, що дозволяє легко вимірювати і відстежувати час, витрачений на різні завдання та діяльності;
- гнучкість налаштування: додаток дозволяє користувачам налаштовувати власні категорії, теги та завдання, що допомагає адаптувати його до індивідуальних потреб і вимог;
- звіти і статистика: Atracker надає користувачам детальну статистику та звіти про витрачений час, що допомагає аналізувати продуктивність, виявляти тенденції і вдосконалювати робочі звички.

Недоліки Atracker:

- відсутність хмарного сховища: Atracker не має вбудованого хмарного сховища для синхронізації даних між пристроями, що може бути не зручним для користувачів, які використовують декілька пристроїв;
- відсутність інтеграції з іншими додатками: додаток може не підтримувати інтеграцію з іншими популярними інструментами або сервісами, що може бути обмеженням для користувачів, які вже використовують інші інструменти для управління часом і завданнями;
- відсутність автоматичного відстеження: додаток вимагає вручного запуску таймера для початку відстеження часу, що може бути не зручним для деяких користувачів, особливо якщо вони забувають включити часовий облік.



Рисунок 1.3 – Головне меню системи Atracker

Розглянемо додаток Everhour.

Everhour - це додаток для відстеження часу, який надає функціональність для контролю часу, витраченого на різні проекти і завдання. Ось кілька плюсів і недоліків цього додатку:

Плюси Everhour:

- простота використання: Everhour має інтуїтивно зрозумілий і дружній інтерфейс, що дозволяє користувачам швидко освоїти програму і почати відстежувати час;

- інтеграція з іншими інструментами: Everhour підтримує інтеграцію з популярними проектними менеджерами і інструментами, такими як Asana, Trello, Jira, Basecamp, що дозволяє легко і автоматично синхронізувати дані про час між різними платформами;

- гнучкість в налаштуванні: Everhour надає можливість налаштувати різні проекти, завдання, теги і клієнтів, що дозволяє користувачам адаптувати додаток до своїх конкретних потреб і вимог.

Недоліки Everhour:

- обмежена функціональність: У порівнянні з деякими іншими додатками для відстеження часу, Everhour може мати обмежені можливості і недостатній функціонал, особливо для великих команд і складних проектів;

- вартість: Everhour пропонує платні плани, які можуть бути дорогими для окремих користувачів або невеликих команд, особливо якщо вони не використовують всі доступні функції;

- залежність від Інтернету: Everhour - це хмарний сервіс, тому для використання додатку необхідне постійне підключення до Інтернету.

Враховуючи ці плюси і недоліки, Everhour може бути корисним інструментом для відстеження часу і управління проектами, але користувачам слід ретельно розглянути свої потреби та бюджет перед прийняттям рішення про використання цього додатку. Головне меню зображено на рисунку 1.4.

					IC93.240БАК.004 ПЗ	Арк.
						14
Зм.	Лист	№ докум.	Підпис	Дата		

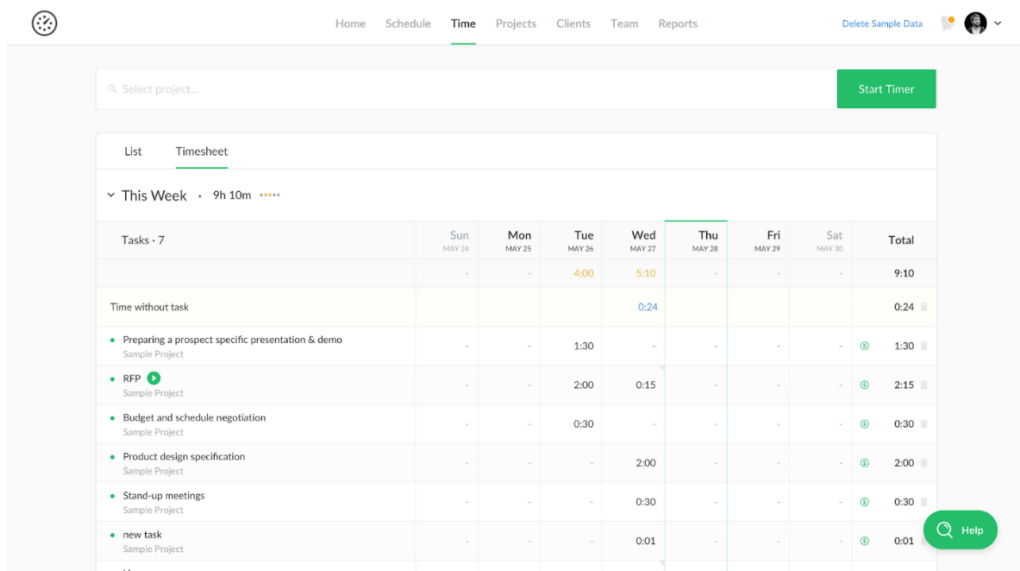


Рисунок 1.4 – Головне меню системи Everhour

Розглянемо додаток RescueTime.

RescueTime - це додаток для відстеження часу і аналізу продуктивності, який допомагає користувачам бути більш свідомими щодо витраченого часу на різні дії. Ось кілька плюсів і недоліків цього додатку:

Плюси RescueTime:

- автоматичне відстеження часу: RescueTime автоматично відстежує, як користувач витрачає час на різні програми, веб-сайти та діяльності, не потребуючи ручного введення даних. Це забезпечує точність і зручність відстеження;

- деталізована статистика: Додаток надає користувачеві детальну статистику про те, як вони витрачають свій час, включаючи час, витрачений на робочі додатки, веб-сайти, соціальні медіа і багато іншого;

- налаштування цілей і нагадувань: RescueTime дозволяє встановлювати цілі щодо витраченого часу на певні діяльності. Також можна налаштувати нагадування, які допоможуть зосередитися на пріоритетних завданнях і забезпечити баланс роботи та відпочинку.

Недоліки додатку RescueTime:

- потреба в постійному підключенні до Інтернету: RescueTime вимагає постійного підключення до Інтернету для збереження та синхронізації даних.

Це може бути проблемою в разі відсутності зв'язку або обмеженого доступу до Інтернету;

- обмежена точність відстеження: Хоча RescueTime старається автоматично відстежувати діяльність користувача, він може бути не завжди точним у визначенні, яку саме діяльність виконує користувач. Наприклад, він може помилково класифікувати продуктивну роботу як розваги або навпаки;

- відсутність детальної контролю: RescueTime надає загальний огляд часу, витраченого на різні діяльності, але не надає деталей про конкретні завдання або проекти. Це може ускладнити аналіз продуктивності на більш розгорнутому рівні.

На рисунку 1.5 зображено головне меню системи RescueTime

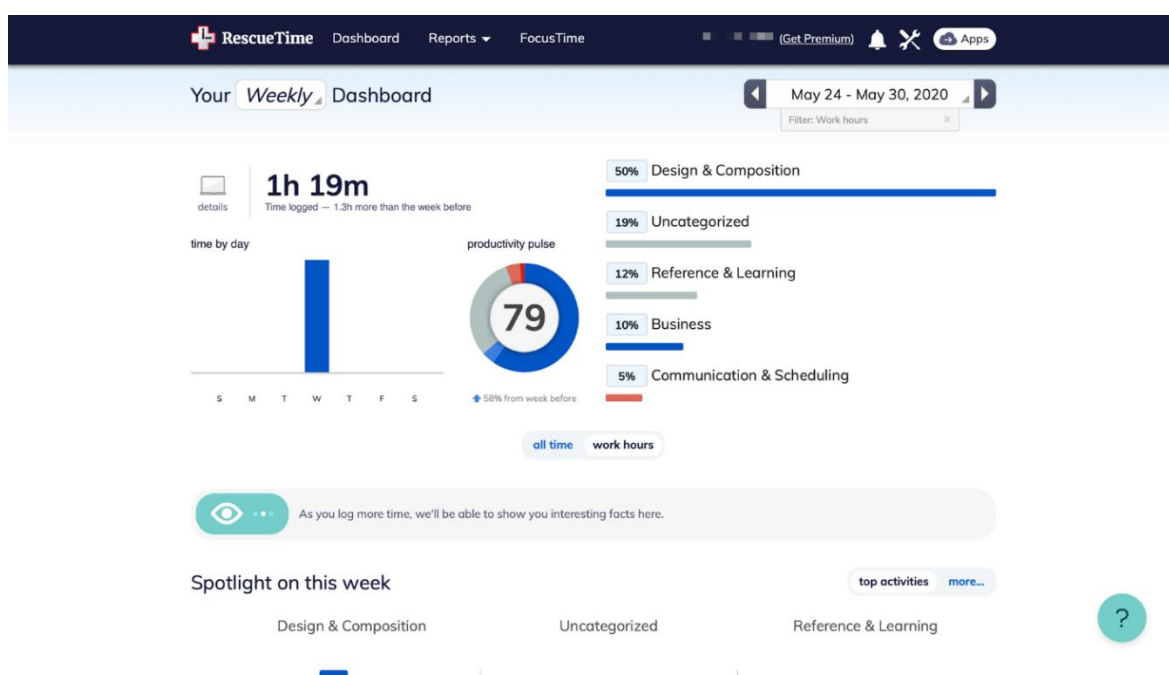


Рисунок 1.5 – Головне меню системи RescueTime

1.3 Постановка задачі

Система призначена для організації робочого дня та відпочинку за допомогою таймеру та нагадувань про відпочинок.

					ІС93.240БАК.004 ПЗ	Арк.
						16
Зм.	Лист	№ докум.	Підпис	Дата		

Метою створення такого застосунку є підвищення ефективності організації робочого дня за рахунок створення таймеру контролю робочого часу та відпочинку.

Для досягнення вказаної мети необхідно реалізувати такі задачі:

- реєстрація користувача;
- авторизація користувача;
- створення таймеру;
- задання параметрів таймеру;
- завдання параметрів повідомлень про відпочинок;
- перегляд варіантів відпочинку;
- запис завдань на день.

Висновок до розділу

У цьому розділі було виконано проектування додатку для контролювання проведеного часу за роботою та відпочинком. Описано предметне середовище та процеси діяльності. Визначено актора додатку та функції для функціональної моделі. Встановила пріоритети функцій.

Було переглянуто популярні аналоги даного додатка, визначила їх плюси та проаналізувала більшість їх недоліків. З цього було зроблено висновок що існують багато додатків для контролю роботи, але більшість із них контролюють саме час проведене за використанням тим чи іншим додатком, але жодне з них не дає змоги повідомлювати користувача про потрібний відпочинок та зберегти його від перевтоми. Зробивши припущення, можна сказати що існують схожі додатки у корпораціях та особистого користування, але про них інформації не було знайдено.

Було визначено призначення, цілі, описала задачі для реалізації та мету розробки такого додатку.

					IC93.240БАК.004 ПЗ	Арк.
						17
Зм.	Лист	№ докум.	Підпис	Дата		

2 ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

2.1 Вхідні дані

Вхідні дані для додатку надаються користувачем. На вході користувач вказує час для таймеру для роботи, кількість та тривалість таймеру для відпочинку.

Користувач вказує наступні дані:

- момент початку таймеру (час);
- тривалість таймеру (саме роботи, час);
- періодичність нагадування про відпочинок (час);
- тривалість таймеру (для відпочинку, час);

На рисунку зображений вигляд сторінки для введення вхідних даних системи.

2.2 Вихідні дані

Вихідними даними є створений таймер, який надає нагадування про відпочинок у заданий час і під час перерв надає можливість переглянути розминки, йогу або інше.

На рисунку 2.1 зображено вихідні дані після того як користувач зареєструвався або авторизувався у додатку. Меню являє собою 3 кнопки із вибором це «Повідомлення», «Таймер», «Контроль» чи «План завдань». Кожна кнопка задіяна класом та перенаправляє користувача до відповідної сторінки додатку. Кнопка «Таймер» направляє користувача до сторінки із налаштуванням параметрів для таймеру, «Повідомлення» направляє користувача до сторінки додатка, де є можливість налаштувати параметри повідомлень, «План завдань» перенаправляє користувача на сторінку-блокнот. Кнопка «Контроль» направляє користувача за сторінку із запуском таймеру та перегляду їх роботи.

					ІС93.240БАК.004 ПЗ	Арк.
						18
Зм.	Лист	№ докум.	Підпис	Дата		

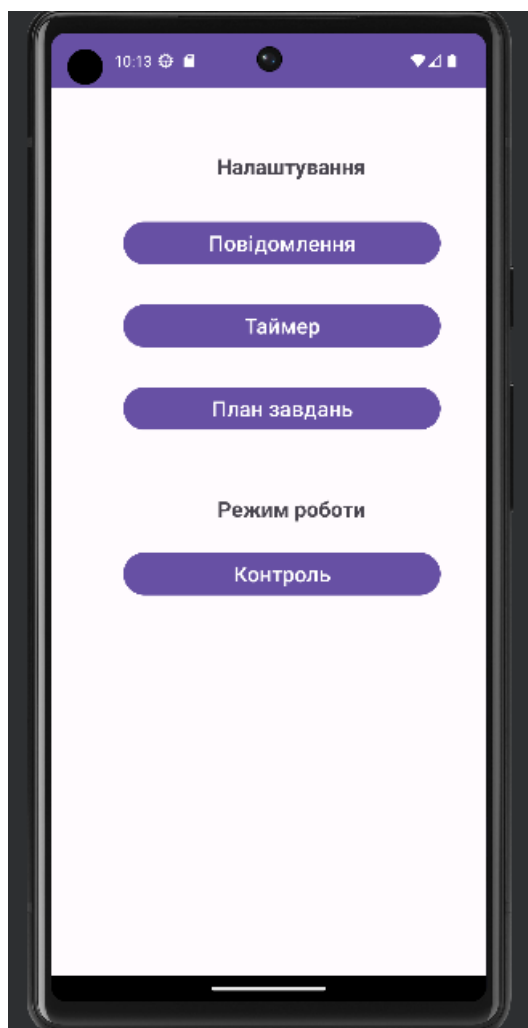


Рисунок 2.1 – Скріншот застосунка із вихідними даними

На рисунку 2.1 є три види параметрів, це «Періодичність», «Тривалість», та можливість змінювати текст повідомлення. Також є кнопка «Зберегти налаштування». Вихідні дані це обраний тип активності та інтервал з яким будуть надсилати повідомлення про відпочинок. Також дані із «Повідомлень» задіяні на рисунках 2.5 та 2.6.

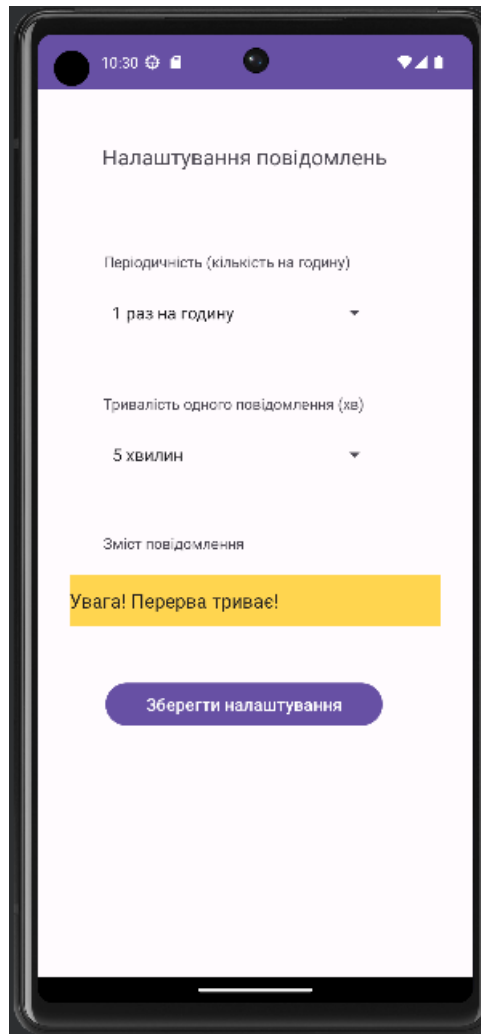


Рисунок 2.2 – Скріншот застосунка із вихідними даними

На рисунку 2.4, знаходяться налаштування таймеру, а саме перелік вибору «Кількість годин» та «Кількість хвилин». На даному етапі користувач задає параметри таймеру, а саме вказати час роботи застосунка, і вихідними даними є заданий параметр – час роботи таймеру.

					ІС93.240БАК.004 ПЗ	Арк.
						20
Зм.	Лист	№ докум.	Підпис	Дата		

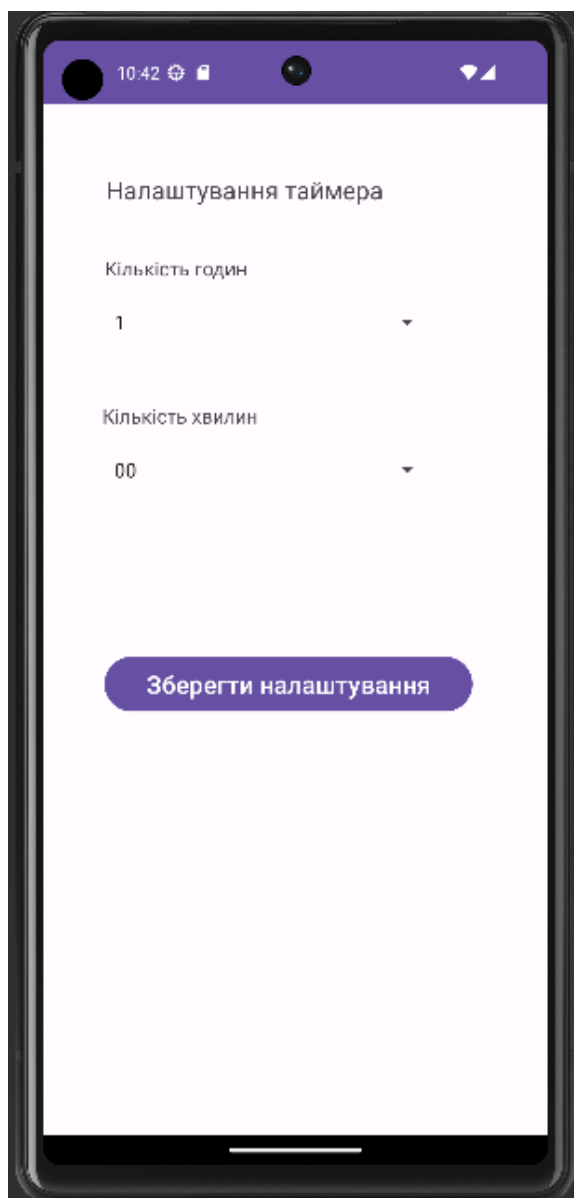


Рисунок 2.3 – Скріншот застосунка із вихідними даними

На рисунку 2.4, знаходяться показана робота двох таймерів, а саме . На даному етапі користувач задає параметри таймеру, а саме вказати час роботи застосунка, і вихідними даними є заданий параметр – час роботи таймеру.

					ІС93.240БАК.004 ПЗ	Арк.
						21
Зм.	Лист	№ докум.	Підпис	Дата		

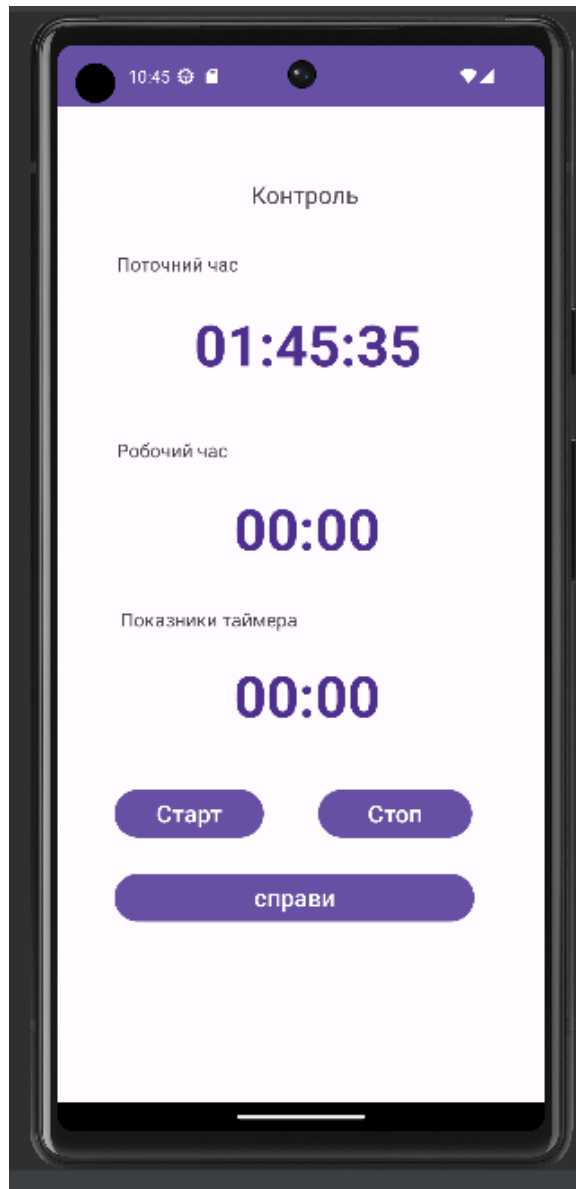


Рисунок 2.4 – Скріншот застосунка із вихідними даними

На рисунках 2.5 і 2.6 вказані вихідні дані, що може побачити користувач після вибору параметрів «Повідомлень». Вправи залежать від активності якою займається користувач. Вихідні дані – це фото вправ, підібрані під типи активностей таким чином, щоб користувач міг розім’ятись або навпаки, дати спокій своєму тілу та вберегти себе від перевтоми.

					ІС93.240БАК.004 ПЗ	Арк.
						22
Зм.	Лист	№ докум.	Підпис	Дата		



Рисунок 2.5 – Скріншот застосунка із вихідними даними

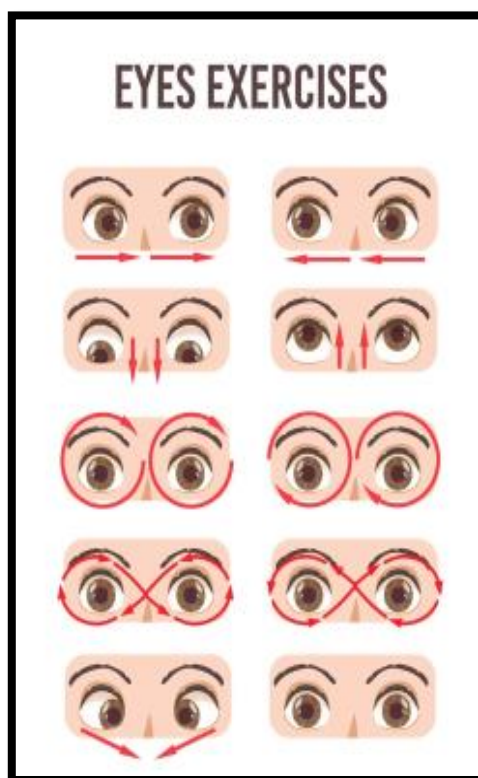


Рисунок 2.6 – Скріншот застосунка із вихідними даними

1.4 Опис структури бази даних

Структура показана в таблицях 2.1 – 2.3.

Таблиця 2.1 – Структура таблиці TimerActivity

Назва поля	Тип	Опис поля
countDownTimer	VAR	Відлік часу
timerRunning	VAR	Таймер
timeLeftInMillis	VAR	Показ скільки залишилось в мілісекундах

Таблиця 2.2 – Структура таблиці LoginActivity

Назва поля	Тип	Опис поля
email	VAR	Пошта користувача
password	VAR	Пароль користувача
id	INT	Первинний ключ
emailEditText	VAL	Можливе корегування пошти
passwordEditText	VAL	Можливе корегування паролю
intent	VAL	Надання доступу до додатку

Таблиця 2.3 – Структура таблиці BreakActivity

Назва поля	Тип	Опис поля
breakType	VAR	Вибір типу перерви
notificationInterval	VAR	Вибір інтервалу перерви

Висновок до розділу

У цьому розділі було визначено вхідні дані, їх джерело надходження та вихідні дані. Таким чином вхідні дані представлені параметрами, які вводять користувачі.

Для роботи додатку користувач повинен надати системі на вхід дані для створення таймеру, такі як час (години, хвилини або секунди). Для запуску отримання повідомлень користувач повинен обрати тип активності, яку він виконує під час роботи таймеру та інтервал, з яким будуть надсилатись повідомлення користувачеві.

Вихідними даними є створений та налагоджений таймер, де користувач може спостерігати його роботу, контролювати проведений час за роботою чи будь яким заняттям та має можливість провести відпочинок із розминкою чи простою зарядкою для очей, Також користувач отримує всі повідомлення за заданими раніше параметрами. Була розроблена структура бази даних, описані її поля із їх типами.

					ІС93.240БАК.004 ПЗ	Арк.
						25
Зм.	Лист	№ докум.	Підпис	Дата		

3 АЛГОРИТМ РОБОТИ ЗАСТУСУНКУ

3.1 Змістовна постановка задачі

Контроль часу і здатність відпочивати є ключовими для підтримання ефективності та збереження енергії протягом робочого дня. Головними критеріями для цього є планування робочого часу, встановлення проміжків між завданнями та встановлення нагадувань про відпочинок.

Користувач вказує наступні параметри для початку роботи:

- початок таймеру та кінець таймеру(час);
- тривалість перерви та інтервал її(час).

3.2 Математична постановка задачі

Математична постановка задачі додатку про контроль часу може мати наступну формулювання:

Позначення:

- T_{total} - загальний час робочого дня (у годинах)
- T_{break} - тривалість перерви (у годинах)
- T_{work} - тривалість робочого інтервалу (у годинах)
- T_{rest} - тривалість інтервалу відпочинку (у годинах)
- N_{work} - кількість робочих інтервалів протягом робочого дня
- N_{rest} - кількість інтервалів відпочинку протягом робочого дня

Треба знайти:

- Порядок та тривалість робочих і відпочинкових інтервалів, щоб уникнути перевтомленості.

Обмеження:

- Сума тривалостей всіх робочих і відпочинкових інтервалів повинна дорівнювати загальному часу робочого дня:

$$N_work * T_work + N_rest * T_rest = T_total - T_break$$

Мета:

– Мінімізувати перевтомленість шляхом ефективного розподілу робочих і відпочинкових інтервалів.

Одна з можливих стратегій розподілу робочих і відпочинкових інтервалів може бути встановлення фіксованих тривалостей роботи та відпочинку. Наприклад, користувач може встановити, що працюватиме T_work годин підряд, після чого візьме T_rest годин відпочинку перед наступним робочим інтервалом.

Ця стратегія передбачає, що загальний час робочого дня поділяється на рівні інтервали роботи та відпочинку. Тому кількість робочих і відпочинкових інтервалів можна визначити за формулою:

$$N_work = \lfloor (T_total - T_break) / (T_work + T_rest) \rfloor$$

$$N_rest = N_work - 1$$

Таким чином, знайдений порядок та тривалість робочих і відпочинкових інтервалів дозволить користувачеві контролювати свій робочий день та уникати перевтомлення.

3.3 Обґрунтування алгоритму розв'язання

Алгоритм додатку для контролювання часу роботи та відпочинку може бути наступним:

1. Вхідні дані:

– Встановити параметри робочого дня (T_total - загальний час робочого дня, T_break - тривалість перерви, T_work - тривалість робочого інтервалу, T_rest - тривалість інтервалу відпочинку).

2. Обчислення кількості робочих і відпочинкових інтервалів:

– $N_work = \lfloor (T_total - T_break) / (T_work + T_rest) \rfloor$ - визначити кількість робочих інтервалів, округлену вниз до найближчого цілого.

					ІС93.240БАК.004 ПЗ	Арк.
						27
Зм.	Лист	№ докум.	Підпис	Дата		

– $N_{rest} = N_{work} - 1$ - визначити кількість відпочинкових інтервалів, що на один менше за кількість робочих інтервалів.

3. Нагадування:

– Встановити нагадування перед кожним відпочинковим інтервалом, щоб нагадати користувачеві про необхідність відпочинку.

Цей алгоритм забезпечує ефективний розподіл часу між роботою та відпочинком, дотримуючись встановлених параметрів користувача.

3.4 Опис алгоритму

Опис алгоритму є важливим для розробки програмного забезпечення. Він допомагає розуміти послідовність дій, які потрібно виконати для досягнення певної мети. Це полегшує комунікацію між розробниками, сприяє аналізу та оцінці ефективності алгоритму. Опис алгоритму також може бути використаний для навчання, де він допомагає студентам розібратися в логіці та деталях роботи програми. Крім того, опис алгоритму є документацією, яка зберігається для майбутнього використання та посилення. В цілому, опис алгоритму відіграє важливу роль у процесі розробки програмного забезпечення та його розумінні. Алгоритм роботи усього додатку та взаємодію системи із користувачем можна переглянути в графічних матеріалах під шифром Д1 та Д2.

Користувач запускає додаток, та переглядає початкове меню застосунку: реєстрація та авторизація. Якщо користувач запускає додаток вперше, йому потрібно зареєструватися. Для цього він обирає “Sing Up”. Система направляє користувача на наступну сторінку додатку, де йому потрібно ввести дані про себе, такі як ім’я, електронну пошту, пароль та підтвердження паролю.

Система перевіряє чи є помилки при введенні пошти. Якщо є, система запрошує користувача повторно ввести данні для реєстрації якщо ж вона не знайшла помилок, система реєструє користувача і направляє на головне меню додатку.

Якщо користувач обирає «Sing In», система перенаправляє його на сторінку додатку із полями для вводу «E-mail» тобто електронна пошта та «Password», пароль який користувач вводив при реєстрації у додатку.

Після цього система перевіряє чи є помилки при введенні пошти та паролю. Якщо є, система запрошує користувача повторно ввести данні для реєстрації якщо ж вона не знайшла помилок, система авторизовує користувача і направляє на головне меню додатку. Схематично це показано на рисунку 3.1

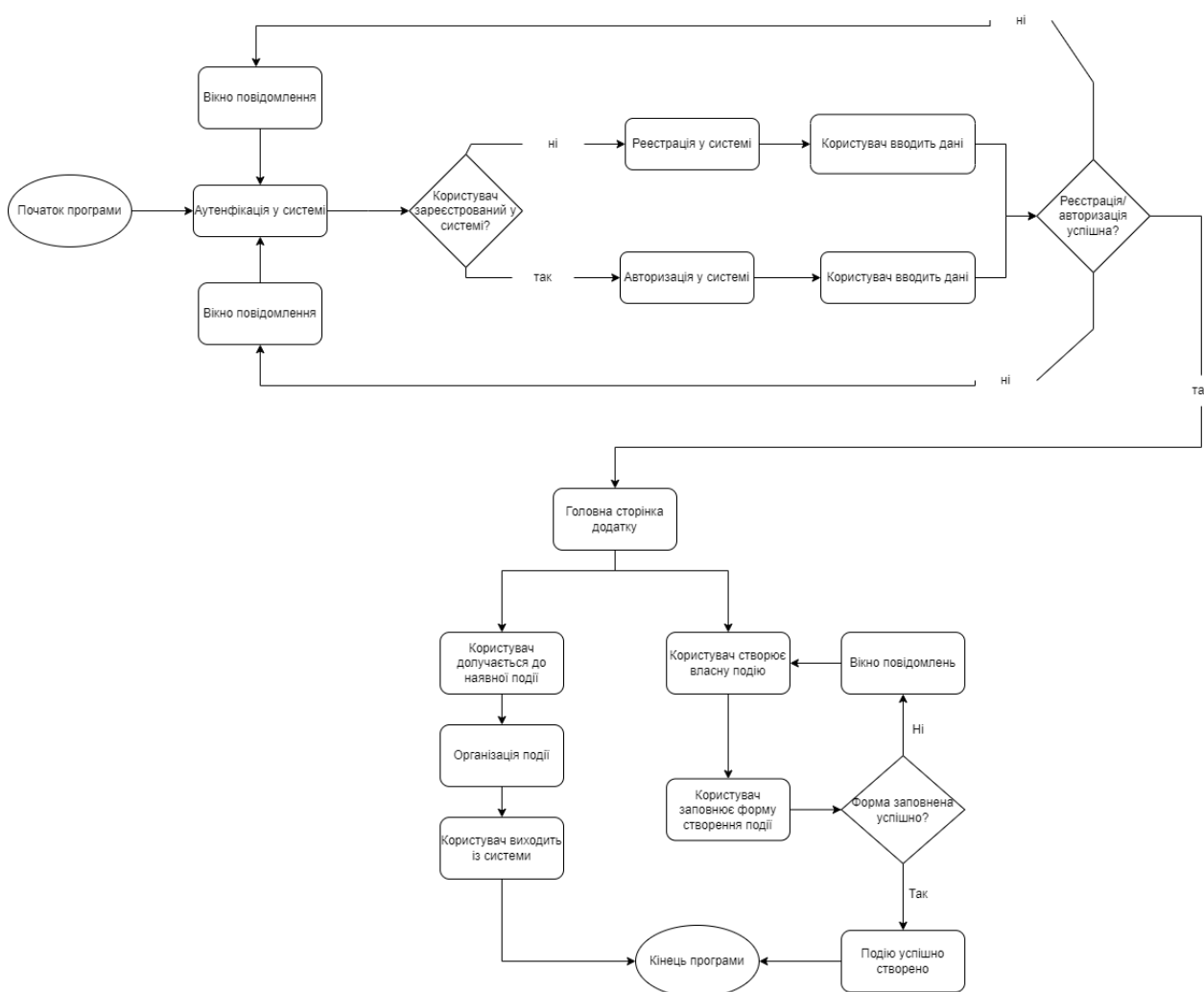


Рисунок 3.1 – Алгоритм системи для управління подіями.

Користувач, потрапивши на головне меню додатка, переглядає запропоновані функції, такі як «Timer», «Break», «Task Plan»,

Якщо користувач обрав «Timer», система направляє його на сторінку таймеру. В на даній сторінці користувачеві потрібно ввести параметри для налаштування роботи таймеру. Це обрати час, в годинах, хвилинах та секундах.

Далі система перевіряє чи є помилки при введенні параметрів, в даному випадку часу роботи таймеру. Якщо є, система запрошує користувача повторно ввести параметри, якщо ж вона не знайшла помилок, користувач має змогу натиснути кнопку «Start» та запустити роботу таймера. Схематично це показано на рисунку 3.2.

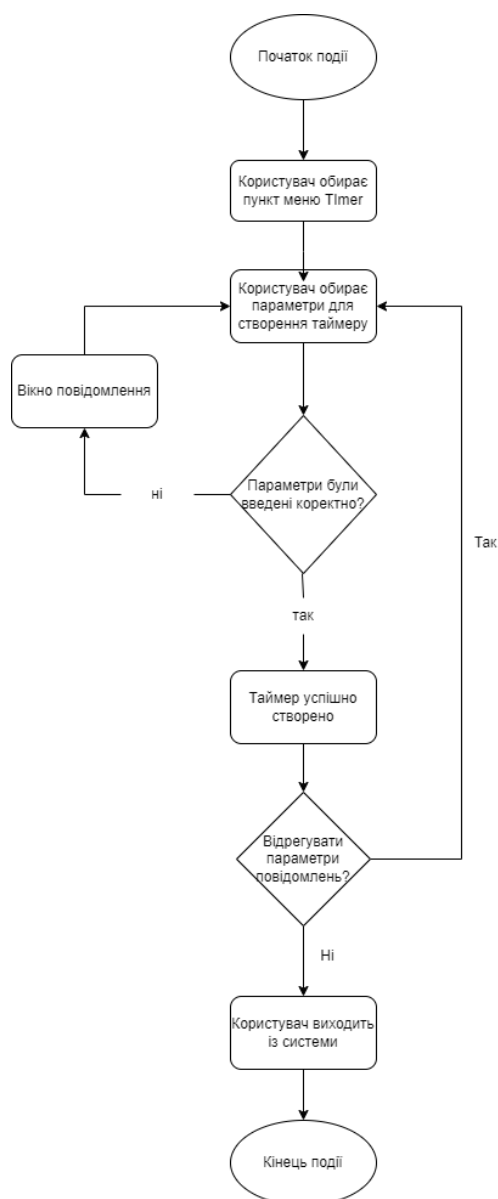


Рисунок 3.2 – Алгоритм запуску роботи таймера

Якщо користувач обрав «Break», система його направляє до сторінки додатку про повідомлення. Користувач повинен обрати параметри роботи, а саме отримання повідомлень про відпочинок. Для цього він обирає тип діяльності, тобто чим зайнятий користувач під час роботи таймеру. Це може бути як офісна робота, малювання, гра на комп'ютері або активна діяльність, як спорт. Також користувачу потрібно обрати інтервал, з яким будуть надходити повідомлення, за умовчанням це 30 хвилин, 1 година та 1,5 годин.

Після система перевіряє чи є помилки при введенні параметрів, в даному випадку вибір активності та інтервал отримання повідомлень. Якщо є, система запрошує користувача повторно ввести параметри, якщо ж вона не знайшла помилок, користувач має змогу натиснути кнопку «Start» та запустити роботу повідомлень. Схематично це показано на рисунку 3.3.

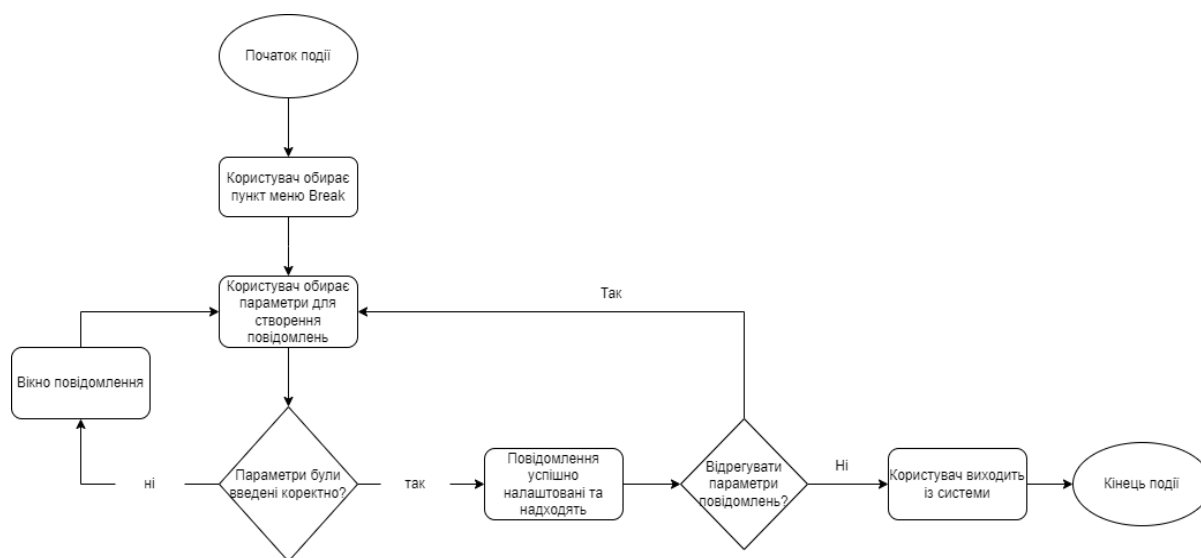


Рисунок 3.3 – Алгоритм запуску повідомлень

Якщо ж користувач обрав «Task Plan», система направляє його до сторінки із блокнот, де користувач має змогу ввести будь які записи, наприклад, те що користувач запланував на сьогодні, або наступні дні. Ці дані зберігаються в блокноті і користувач має змогу будь коли повернутися до цієї сторінки та змінити або ж видалити свої записи. Схематично це показано на рисунку 3.4.

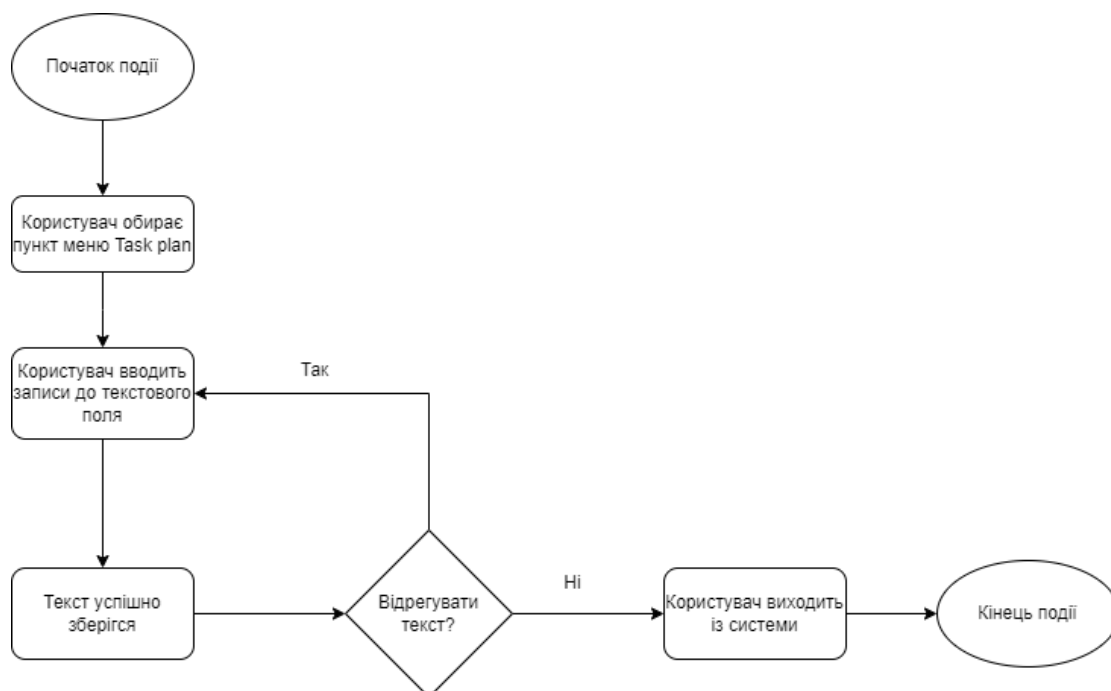


Рисунок 3.4 – Алгоритм роботи «Task Plan»

Саме головне, те що користувач має змогу провести свій відпочинок із користю. У застосунок було додана можливість зайнятися розминкою. Для цього користувач і вводив параметри у «Break» а саме про активність.

Якщо користувач обрав активність «Game», то під час відпочинку, система надасть змогу користувачеві перейти на сторінку додатку із правами для очей або розминкою тіла після довгого сидіння перед комп'ютером. Для цього, користувач обирає «Warm-up» і система перенаправляє користувача до сторінок застосунку із вправами.

На час відпочинку таймер зупиняє свою роботу та відновлює її після закінчення відпочинку. Ця дія повторюється до тих пір поки робота таймеру не закінчиться. Схематично це показано на рисунку 3.5.



Рисунок 3.5 – Алгоритм показу розминки

Висновок до розділу

У розділі було наведено змістовну та математичну постановки задачі. Також були вказані обґрунтування методів розв’язання та опис обраного методу вирішення задачі. Була вказана схема алгоритму роботи додатка.

Було описано алгоритм роботи всього додатку та опис як взаємодіє система із користувачем. Описано та наведено схематично алгоритм реєстрації та авторизації користувача, як система реагує на помилки при вводі (якщо вони наявні), алгоритм налаштування параметрів та запуску.

4 ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ

4.1 Засоби розробки

Інформаційна система було реалізована за допомогою наступного стеку технологій:

- Java;
- XML;
- JavaScript;
- Kotlin;
- Android Studio;
- Android SDK.

Java - це потужна і популярна мова програмування загального призначення. Вона відома своєю простотою, надійністю та портативністю, що робить її особливо популярною серед розробників програмного забезпечення. Ось кілька ключових рис Java:

- об'єктно-орієнтоване програмування: Java базується на об'єктно-орієнтованому підході, що дозволяє створювати модульні програми з використанням класів, об'єктів, спадкування, та інших об'єктно-орієнтованих концепцій. Це полегшує розробку та підтримку програмного забезпечення;

- портативність: одна з великих переваг Java - це його портативність. Код, написаний на Java, може бути виконаний на будь-якій платформі, яка має відповідний Java Virtual Machine (JVM). Це дозволяє розробникам писати один раз і виконувати на різних операційних системах, таких як Windows, macOS, Linux та інші;

- велика стандартна бібліотека: Java постачається з великою стандартною бібліотекою, яка містить багато готових класів і методів для різних завдань, таких як робота з рядками, мережами, файлами, базами даних.

XML – це розширювана мова розмітки, що використовується для представлення даних у структурованому форматі. XML використовується для

					IC93.240БАК.004 ПЗ	Арк.
						34
Зм.	Лист	№ докум.	Підпис	Дата		

зберігання та передачі інформації між різними системами, які можуть використовувати різні мови програмування та операційні системи.

JavaScript – це високорівнева, інтерпретована мова програмування, яка використовується для створення динамічних інтерактивних веб-сайтів. Вона дозволяє веб-розробникам додавати різноманітну функціональність до веб-сторінок, змінювати їх вміст та реагувати на дії користувача.

Android Studio - це інтегроване середовище розробки (Integrated Development Environment, IDE), призначене для створення програмного забезпечення для платформи Android. Воно розроблене компанією Google і надає розробникам потужні інструменти для створення, тестування та налагодження Android-додатків.

Ось кілька переваг Android Studio:

- інтегроване середовище розробки: Android Studio має вбудовану підтримку для всіх етапів розробки Android-додатків, включаючи створення інтерфейсу користувача, програмування, тестування та налагодження;
- потужні інструменти: Android Studio має широкий набір інструментів, які сприяють розробці Android-додатків;
- швидкість розробки: Android Studio надає широкий набір інструментів та шаблонів, які спрощують процес розробки Android-додатків. Це дозволяє розробникам ефективно працювати над своїми проектами та скорочує час, потрібний для створення додатків;
- підтримка Google Play Services: Android Studio має вбудовану підтримку Google Play Services, яка дозволяє розробникам легко інтегрувати функціональність, таку як аутентифікація користувач.

Android SDK - це набір розробничих інструментів, бібліотек, документації та ресурсів, які необхідні для розробки мобільних додатків для платформи Android. Він містить інструменти, які дозволяють створювати, тестувати, налагоджувати та розгортати додатки для Android.

Основні компоненти Android SDK:

- Android SDK Tools: Набір інструментів командного рядка, які включають Android Debug Bridge (ADB) для взаємодії з підключеними пристроями, а також інші утиліти для зборки, пакування та підпису додатків;
- Android Platform Tools: Включає такі інструменти, як adb (Android Debug Bridge), fastboot, systrace і інші. Ці інструменти дозволяють взаємодіяти з пристроями Android, виконувати налагодження, робити скріншоти тощо;
- Android Build Tools: Набір інструментів, які використовуються для збирання, пакування та оптимізації додатків для платформи Android. Включає компілятор Java, утиліти для збірки ресурсів, обробки анімацій, підпису додатків тощо;
- Android Emulator: Емулятор Android дозволяє запускати та тестувати додатки на віртуальних пристроях Android без фізичного пристрою.

4.2 Вимоги до технічного забезпечення

Даний програмний продукт представлений у вигляді мобільного додатку, який був роздобрений для платформи Android.

Мінімальні вимоги до технічного забезпечення для мобільного додатку на платформі Android:

1. Операційна система: Android 5.0 (Lollipop) або вище. Рекомендується використовувати останню стабільну версію Android для найкращої сумісності та безпеки.
2. Процесор: ARM або x86 процесор з тактовою частотою не менше 1,4 ГГц. Більш потужний процесор може бути необхідним для більш вимогливих додатків або ігор.
3. Оперативна пам'ять (RAM): Мінімум 1 ГБ оперативної пам'яті для базового функціонування додатків. Для більш складних або ресурсоємних додатків рекомендується більше оперативної пам'яті.

					IC93.240БАК.004 ПЗ	Арк.
						36
Зм.	Лист	№ докум.	Підпис	Дата		

4. Внутрішня пам'ять: Додаток вимагає 25 Мб.

5. Дисплей: Роздільна здатність екрану телефону або планшета повинна бути відповідна для коректного відображення інтерфейсу вашого додатку.

4.3 Архітектура програмного забезпечення

4.3.1 Архітектура ПЗ

Архітектура даного програмного забезпечення є доволі нескладною та чудово підходить для розробки мобільного застосунку. Структура програмного продукту зображена на рисунку схеми архітектури програмного забезпечення, який наведений у графічних матеріалах.

4.3.2 Діаграма класів

Діаграма класів зображує структурну будову програмного продукту і включає такі головні класи:

- TimerActivity;
- FirstActivity;
- MainActivity;
- BreakActivity;
- CleaningActivity;
- DrawingActivity;
- GameActivity;
- MenuActivity;
- RegisterActivity;
- LoginActivity;
- WorkActivity

Опис основних класів програмного застосунку наведено у таблиці 4.1

					IC93.240БАК.004 ПЗ	Арк.
						37
Зм.	Лист	№ докум.	Підпис	Дата		

Таблиця 4.1 – Опис основних класів програмного застосунку

Клас	Опис
TimerActivity;	Клас – фрагмент, де знаходиться сам таймер. У користувача є можливість змінювати таймер, запускати та зупиняти його, обнуляти, продовжувати. На пряму пов'язаний із BreakActivity, як тільки користувач отримує повідомлення про відпочинок, таймер припиняє роботи і відновлюється після завершення відпочинку.
FirstActivity;	Клас – активність, використовується для користувача при першому вході до застосунку.
MainActivity;	Головний клас активності всієї системи, де відбувається навігація по самому додатку. В ньому показано як будуть змінюватись фрагменти при натисканні кнопок при навігації.
BreakActivity;	Клас – фрагмент, де користувач отримуватиме повідомлення про відпочинок. За допомогою notificationManager користувач отримує повідомлення на екран про відпочинок, таймер припиняє роботи і відновлюється після завершення відпочинку. Дані отримаємо з BreakViewModel.

Продовження таблиці 4.1

CleaningActivity;	Клас – фрагмент, де задіяна діяльність користувача, отримує дані із ActivityViewmodel, та передає дані на BreakActivity, після направляє на CleaningViewModel, де користувач може переглядати відповідні вправи.
DrawingActivity;	Клас – фрагмент, де задіяна діяльність користувача, отримує дані із ActivityViewmodel, та передає дані на BreakActivity, після направляє на DrawingViewModel, де користувач може переглядати відповідні вправи.
GameActivity;	Клас – фрагмент, де задіяна діяльність користувача, отримує дані із ActivityViewmodel, та передає дані на BreakActivity, після направляє на GameViewModel, де користувач може переглядати відповідні вправи.
MenuActivity;	Клас – фрагмент, де відбуваються навігація по додатку, користувач робить вибір щодо таймеру, настройки повідомлень про відпочинок чи обирає вид діяльності/активності/роботи. Дані передаються до TimerActivity, BreakActivity.

Продовження таблиці 4.1

RegisterActivity;	Клас-активність, де користувач реєструється в системі. За допомогою findById, яка ініціалізується в даному класі, дані передаються до класу, який через інтерфейс передає їх на сервер. Далі через той самий view отримуємо відповідь від серверу і оновлюємо UI активності (показуючи помилки, такі як "адреса електронної пошти вже зареєстрована в системі", "Паролі не співпадають") або переходимо до MainActivity, якщо авторизація успішна.
LoginActivity;	Клас-активність, де користувач авторизується в системі. За допомогою FirstActivity, яка ініціалізується в даній активності дані передаються до класу через інтерфейс findViewById, Далі отримуємо відповідь від серверу та переходимо до MainActivity, або оновлюємо дані UI активності показуючи помилки, якщо вони є.
WorkActivity.	Клас – фрагмент, де задіяна діяльність користувача, отримує дані із ActivityViewmodel, та передає дані на BreakActivity, після направляє на WorkViewModel, де користувач може переглядати відповідні справи.

4.3.2 Діаграма послідовності

Опишемо послідовність виконання дій у системі. Усю діаграму послідовності процесів можна переглянути в графічних матеріалах зі шифром Д4.

Для успішної взаємодії із застосунком користувачу необхідно авторизуватися у системі. Послідовність процесів, що відбуваються у системі під час авторизації користувача наведені на рисунку 4.1

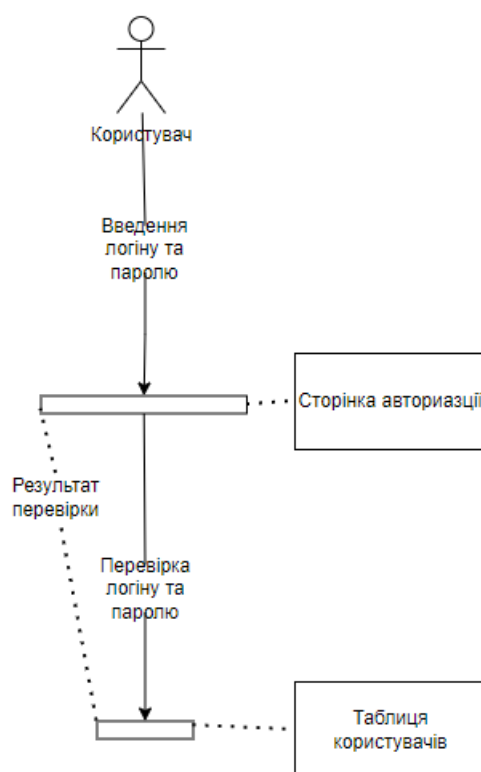


Рисунок 4.1 – Послідовність процесів під час авторизації користувача

Для створення нового таймеру, натискає на кнопку на графічному інтерфейсі. Сервер починає обробляти відправлений запит після його викликається метод класу, який виконує створення таймеру. Отримані результати зберігаються та відправляються на рівень представлення даних для відображення користувачеві.

Послідовність виконання дій для створення таймеру описана на діаграмі послідовності дій, яка зображена на рисунку 4.2.



Рисунок 4.2 – Послідовність виконання дій для запуску та зупинки таймера

Для налаштування повідомлень про відпочинок, користувач натискає на кнопку на графічному інтерфейсі. Сервер починає обробляти відправлений запит після його викликається метод класу, який виконує створення повідомлення про відпочинок. Отримані результати зберігаються та відправляються на рівень представлення даних для відображення користувачеві.

Послідовність виконання дій для налаштування повідомлень про відпочинок описана на діаграмі послідовності дій, яка зображена на рисунку 4.3.

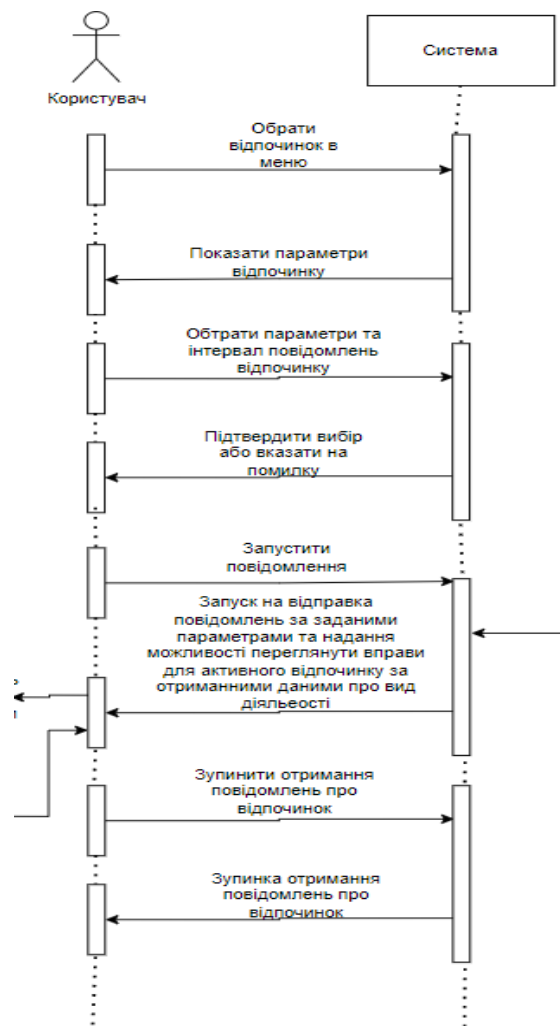


Рисунок 4.3 – Послідовність виконання дій для налаштування повідомлень про відпочинок

Для налаштування параметрів активності (вид діяльності, що користувач виконує протягом роботи) користувач натискає на кнопку на графічному інтерфейсі. Сервер починає обробляти відправлений запит після його викликається метод класу, який виконує налаштування параметрів діяльності. Отримані результати зберігаються та відправляються на рівень представлення даних для відображення користувачеві.

Послідовність виконання дій для налаштування параметрів діяльності описана на діаграмі послідовності дій, яка зображена на рисунку 4.4.

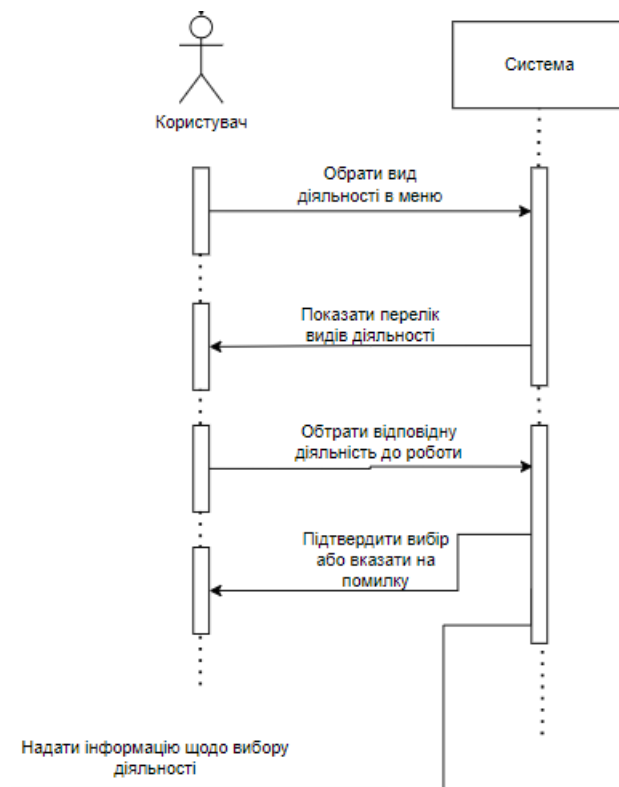


Рисунок 4.4 – Послідовність виконання дій для налаштування параметрів діяльності

На даному рисунку зображено користувача та систему і їх взаємодію. Описана послідовність при виборі діяльності, а саме те що користувач має перейти до меню діяльності, система йому покаже перелік видів діяльності на сторінці додатку. Наступним кроком користувач повинен обрати вид діяльності із запропонованих, а система повинна підтвердити або вивести помилку на екран користувачу. Також система зберігає та передає дані про вибір користувача на наступні методи для подальшої роботи додатку.

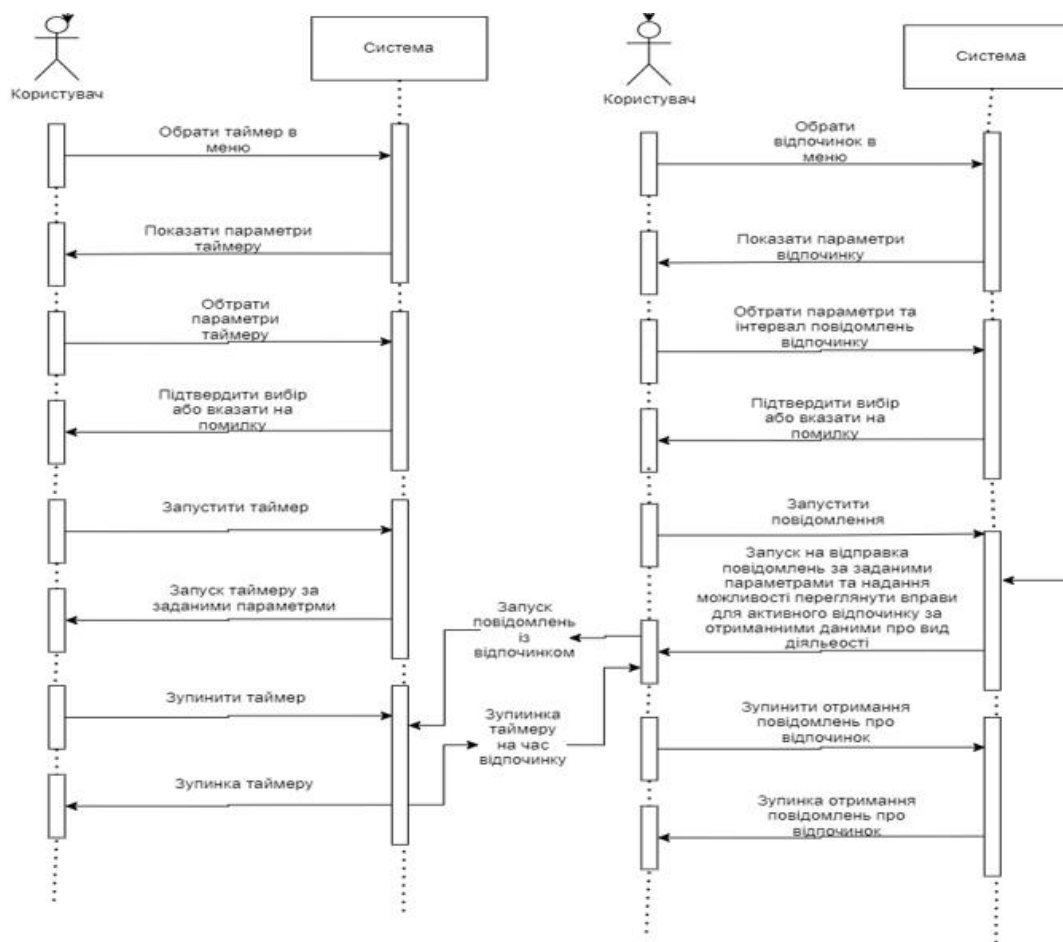


Рисунок 4.5 – Взаємозв’язок послідовності виконання дій для налаштування параметрів таймера та повідомлень про відпочинок.

Роздивимось взаємодію послідовностей дій таймеру та повідомлень про відпочинок. Щоб пов’язати методи користувачу потрібно перейти до таймеру, виставити параметри(час роботи) та запустити таймер, а на стороні повідомлень обрати параметри(інтервал та тривалість) для отримання повідомлень та запустити їх відправку. Як тільки заданий інтервал спрацює та надішле повідомлення користувачеві, система подасть сигнал до таймеру по почався відпочинок, і таймер на цей період часу зупиняє відлік часу. Після того як вийде час, відведений на відпочинок, система надсилає сигнал таймеру про те що потрібно відновитись та продовжити відлік до наступного повідомлення або до кінця роботи таймеру (якщо заданий час для таймеру менший ніж заданий інтервал для повідомлень).

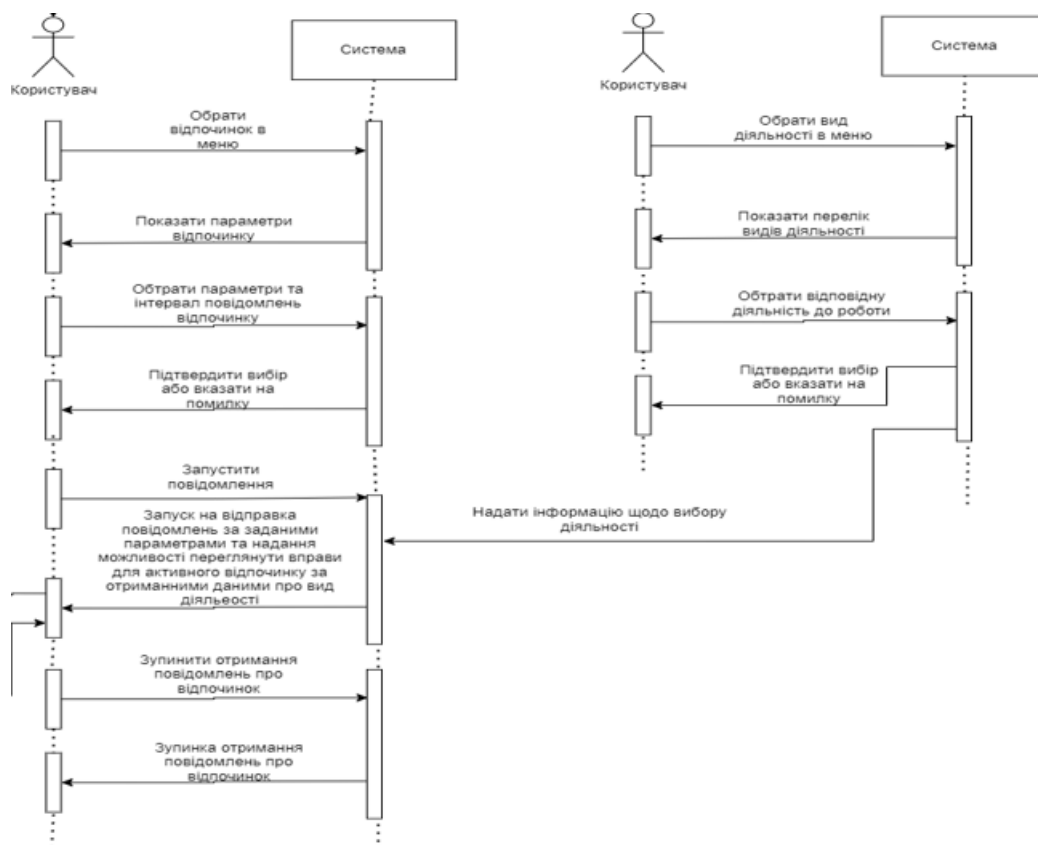


Рисунок 4.6 – Взаємозв’язок послідовності виконання дій для налаштування параметрів повідомлень про відпочинок та вибору діяльності.

Роздивимось взаємодію послідовностей дій відпочинку та вибору діяльності. Щоб пов’язати методи користувачу потрібно перейти до сторінки відпочинку, виставити параметри(інтервал та тривалість) для отримання повідомлень та запустити їх відправку, а на сторінці діяльності обрати певний вид діяльності який користувач планує виконувати. Після система надсилає до класу відпочинку інформацію про вибір користувача для подальшого використання інформації.

4.3.3 Діаграма компонентів

Даний додаток має такі компоненти, таких як: користувацький інтерфейс, сервер, база даних.

Інтерфейс забезпечує відображення усіх даних користувачеві, які обробляються сервером, так же всі дані зберігаються у базі даних. Сервер забезпечує передачу даних з бази даних до інтерфейсу та навпаки. Він обробляє запити, отримані з інтерфейсу, та відправляє інші запити до БД. Відповідь бази даних надсилається до інтерфейсу також через сервер. База даних забезпечує збереження даних та взаємодію із сервером шляхом запитів.

4.3.4 Специфікація функцій

У таблицях 4.2 – 4.6 наведений опис функцій класів системи.

Таблиця 4.2 – Опис специфікацій функцій для класу MainActivity

Клас	Назва функції	Опис функції
MainActivity	protected void	Метод, який є у кожному класі активності, що ініціалізує об'єкти, знаходячи їх по відповідному id в layout активності.
	public void Intent intent;	Метод, який ініціалізує навігацію програмного продукту, надає вибір певної вкладки меню, при якій змінює почний фрагмент контейнеру.

Таблиця 4.3 – Опис специфікацій функцій для класу TimerActivity

Клас	Назва функції	Опис функції
TimerActivity	public void override fun onCreate	Метод який викликає SetCountedView, та методи із FirstActivity викликає функцію запуску та зупинки таймеру.
	public void private fun startTimer fun onTick fun onFinish	Метод підтвердження запуску таймера. Запускає методи onTick та onFinish.
	public void private fun stopTimer	Метод підтвердження зупинки таймера.
	public void private fun updateCountdownText	Метод пов'язаний з View model і надає змогу користувачеві переглянути роботу таймера. Встановлює хвилини та секунди, виводить всі дані за допомогою countedTextView.

Таблиця 4.4 – Опис специфікацій функцій для класу FirstActivity

Клас	Назва функції	Опис функції
FirstActivity	public void Intent intent;	Метод для першого відвідування додатку. Воно передає дані такі дані як Sing In та Sing Up, накладає лістенери на вибір певної кнопки при вході в додаток.

Таблиця 4.5 – Опис специфікацій функцій для класу RegisterActivity

Клас	Назва функції	Опис функції
RegisterActivity	public void fun onCreate setOnClickListener	Метод для самого процесу авторизації у систему, накладає лістенери на вибір певного пункту вводу паролю чи логіну.

Таблиця 4.6 – Опис специфікацій функцій для класу BreakActivity

Клас	Назва функції	Опис функції
BreakActivity	private void notificationBuilder () timer = object notificationChanel()	Метод для повідомлень про відпочинок. Накладаємо лістенери, які при виборі певного інтервалу для повідомлень, що будуть відображати інструменти для взаємодії з ними.

Висновок до розділу

У розділі було описано засоби розробки та стек технологій, обрані для реалізації системи складання розкладу, а саме: Android Studio, Android SDK, Java, XML, JavaScript. Були наведені схеми архітектури програмного забезпечення та діаграми класів. Також були виділені та описані основні класи проєкту, та специфікації їх функцій. Для початку роботи потрібні класи які відповідають за реєстрацію та авторизацію користувача. Після задіяні ті які переправляють користувача на сторінку меню та вибір наступних функцій застосунку, таких як таймер, налаштування повідомлень та блокнот для запису планів на день.

Були перераховані функції всіх класів застосунку, їх взаємодії із іншими класами. Для правильного функціонування застосунку були визначені загальні вимоги до технічного забезпечення.

					ІС93.240БАК.004 ПЗ	Арк.
						50
Зм.	Лист	№ докум.	Підпис	Дата		

5 ТЕСТУВАННЯ СИСТЕМИ

5.1 Керівництво користувача

Коли користувач відкрив мобільний застосунок, він знаходиться на сторінці авторизації, яка зображена на рисунку 5.1. На цій сторінці користувач може обрати зареєструватися або увійти в систему якщо раніше вже був зареєстрований, ввівши логін і пароль та натиснувши кнопку “Enter”. Якщо система виявить помилку, вона повідомить про це і користувач повинен буде повторно ввести дані.

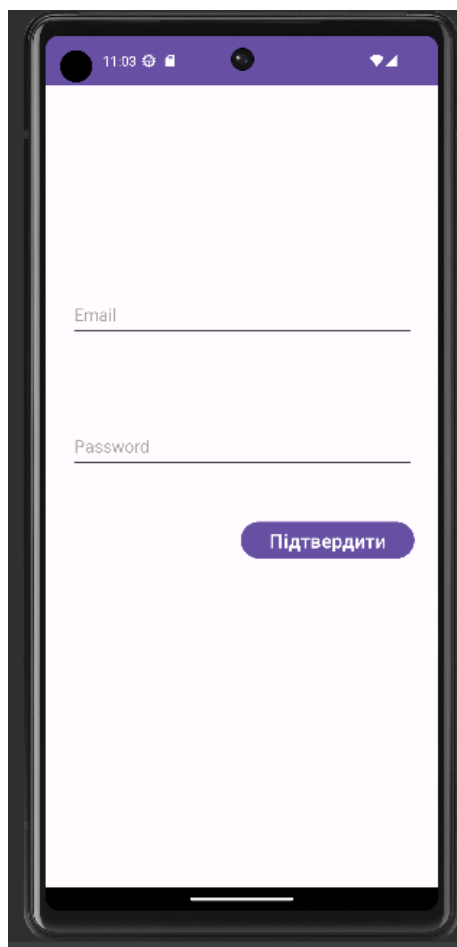


Рисунок 5.1 – Сторінка авторизації

Якщо користувач не зареєстрований у системі, він може створити аккаунт, натиснувши на кнопку “Підтвердити”. На сторінці реєстрації користувач вказує своє повне ім’я, адресу електронної пошти та пароль. Якщо система

					ІС93.240БАК.004 ПЗ	Арк.
						51
Зм.	Лист	№ докум.	Підпис	Дата		

виявить помилку, вона повідомить про це і користувач повинен буде повторно ввести дані, поки система не підтвердить вірність пошти. Це зображено на рисунку 5.2.

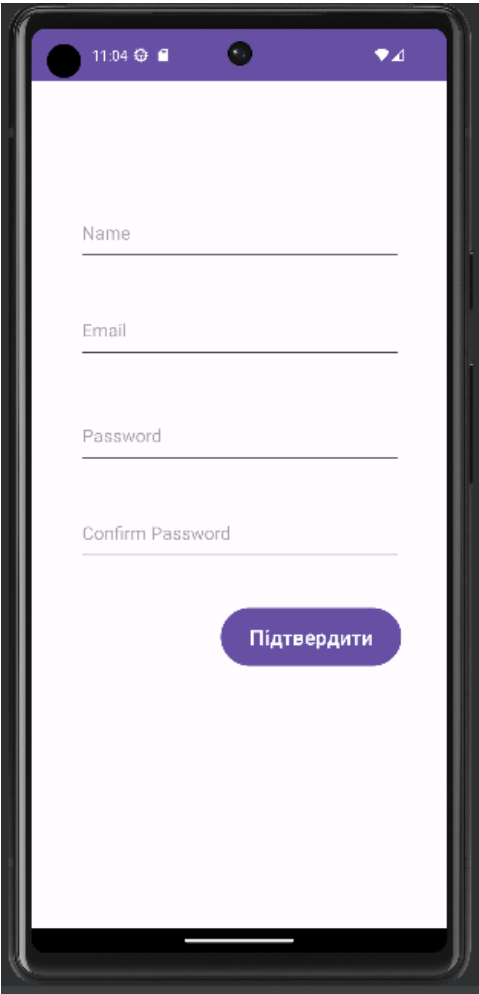


Рисунок 5.2 – Сторінка реєстрації

Після реєстрації чи авторизації система направляє користувач на головну сторінку додатку, де знаходиться саме меню. Меню являє собою 3 кнопки із вибором. Користувач може обрати «Повідомлення», «Таймер», «Контроль» чи «План завдань». Кнопка «Таймер» направляє користувача до сторінки із налаштуванням параметрів для таймеру, де його їй можна запустити. Кнопка «Повідомлення» направляє користувача до сторінки додатка, де є можливість налаштувати параметри повідомлень та запустити їх. Кнопка «План завдань» перенаправляє користувача на сторінку-блокнот, де він може записати будь що, наприклад плани на сьогоднішній день. Користувач може повернутись назад до головного меню в будь який час. Це все можна побачити на рисунку 5.3.

					ІС93.240БАК.004 ПЗ	Арк.
						52
Зм.	Лист	№ докум.	Підпис	Дата		

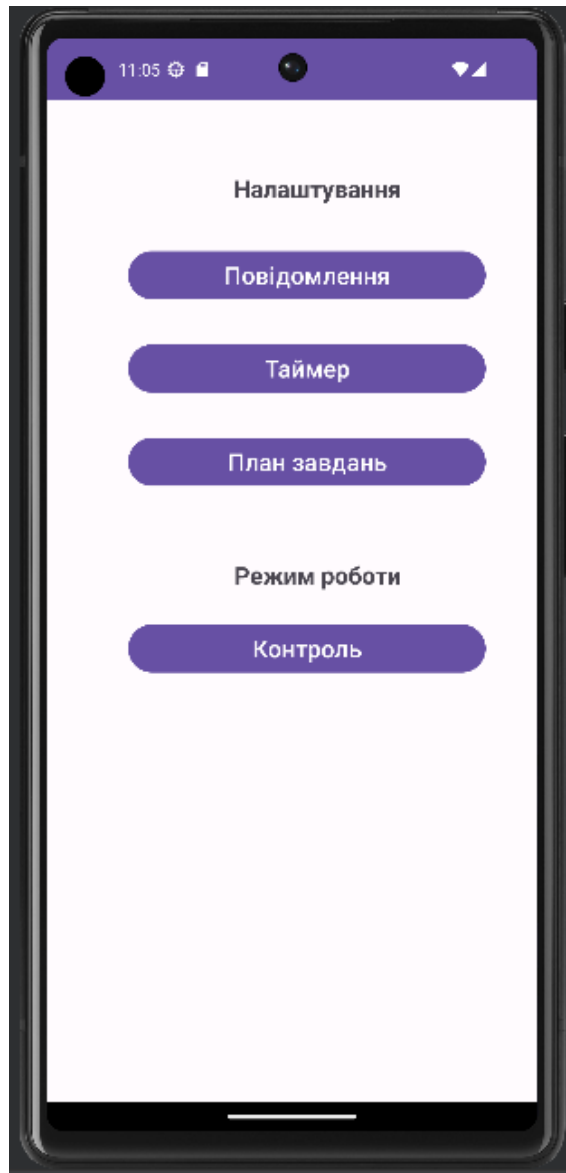


Рисунок 5.3 – Головна сторінка додатку

Якщо користувач обере «Таймер», його направле на наступну сторінку, показану на рисунку 5.4. На цій сторінці користувач баче цифри, а саме це майбутня робота таймеру, поле для вводу часу, та дві кнопки, це «Start» «Stop». На даному етапі користувач задає параметри таймеру, а саме вказати час роботи застосунка, та якщо система не виявить помилок, надасть йому можливість запустити таймер та зупинити. Якщо ж помилки будуть виявлені, система запитає користувача повторно ввести параметри.

					ІС93.240БАК.004 ПЗ	Арк.
						53
Зм.	Лист	№ докум.	Підпис	Дата		

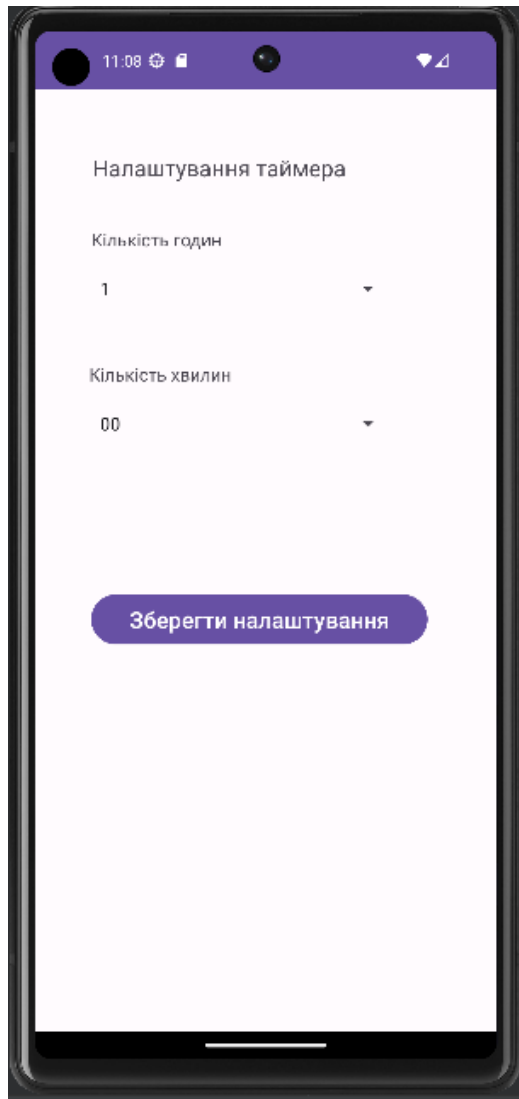


Рисунок 5.4 – Сторінка параметрів та роботи таймера

Якщо користувач обере «Повідомлення», система направле його до наступної сторінки додатку, яку зображено на рисунку 5.5. На цій сторінці, користувач може бачити два види параметри, це «Break Type» і «Interval», та дві кнопки «Start Notifications» і «Stop Notifications». Користувачу потрібно може обрати тип перерви, а саме активність яку він виконую під час роботи таймеру. Другий параметр це інтервал з яким система буде надсилати користувачі повідомлення.

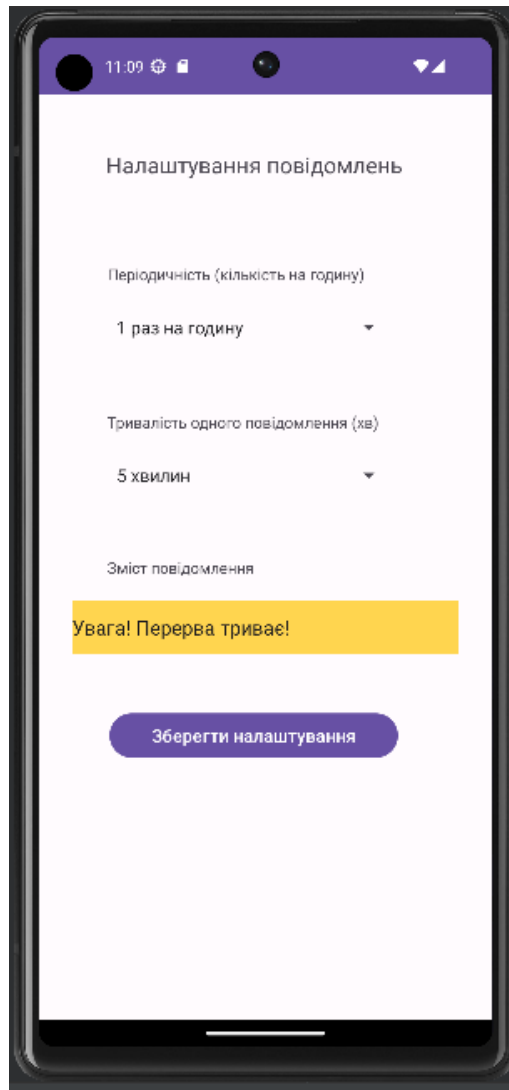


Рисунок 5.5 – Сторінка параметрів та роботи повідомлень-нагадувань про відпочинок

Одним із головних параметрів для налаштування повідомлень є «Break Туре», його сторінка показана на рисунку 5.6.

Перейшовши на цю сторінку користувач бачу 4 кнопки із «Заняття», «Гра», «Прибирання», «Малювання». Вибір залежить від того чим займається користувач під час роботи таймеру. Цей параметр надасть змогу користувачеві переглянути вправи для всього тіла або якоїсь частини, і це залежатиме від вибору активності.

Користувач може перейти на сторінку із вправами тільки після отримання повідомлення про відпочинок.

					ІС93.240БАК.004 ПЗ	Арк.
						55
Зм.	Лист	№ докум.	Підпис	Дата		

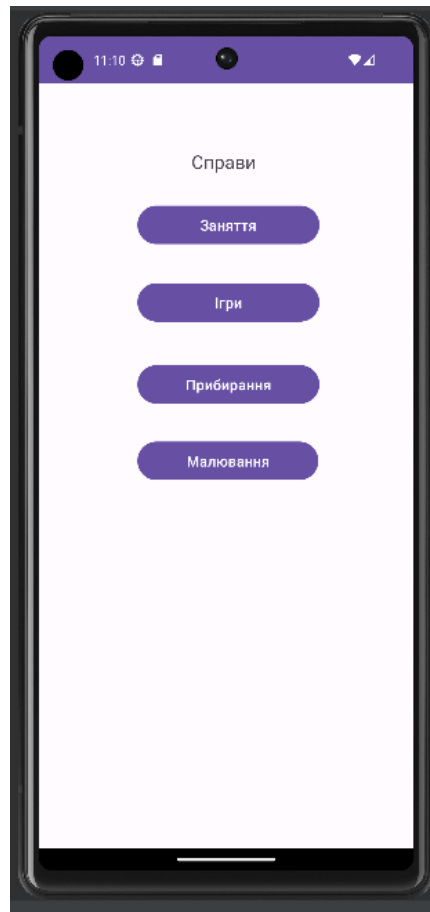


Рисунок 5.6 – Сторінка вибору активності

5.2 Випробування програмного продукту

Для перевірки працездатності додатку, та його відповідності до функціональним вимогам, які вказані у технічному завданні, було проведено тестування додатку для контролю проведеного часу за роботою та відпочинком. Цей розділ містить опис тест-кейсів з переліком дій для їх реалізації.

5.2.1 Мета випробувань та загальні положення випробувань

За мету випробувань було взято перевірку функціональності додатку для контролю проведеного часу за роботою та відпочинком.

Тестування програмного забезпечення (ПЗ) - це процес перевірки функцій, якості та надійності програмного продукту з метою виявлення помилок і забезпечення його відповідності вимогам та очікуванням користувачів.

					ІС93.240БАК.004 ПЗ	Арк.
						56
Зм.	Лист	№ докум.	Підпис	Дата		

Основні мети тестування програмного забезпечення це виявлення помилок, перевірка вимог, підтримка якості, впровадження програмного продукту. Тестування допомагає ідентифікувати помилки або дефекти в програмному продукті, дозволяє перевірити, чи відповідає програмне забезпечення встановленим вимогам, забезпечує високу якість програмного продукту і дозволяє оцінити готовність програмного продукту до впровадження на реальній апаратурі або середовищі.

Тестування програмного забезпечення може включати різні методи та підходи, такі як модульне тестування, інтеграційне тестування, системне тестування, регресійне тестування, валідація, верифікація, альфа- та бета-тестування та інші.

Тест-кейс (Test Case) - це детальний сценарій або набір інструкцій, які описують конкретні кроки для виконання певного тестувального випадку. Тест-кейс визначає передумови, кроки, вхідні дані, очікувані результати та критерії прийнятності для перевірки певного аспекту програмного продукту.

Основні елементи, які можуть бути включені до тест-кейсу, це номер тест-кейсу, назва, опис, передумови, кроки, вхідні дані, очікувані результати та критерії прийнятності.

Тест-кейси використовуються для організації тестування, документування результатів і стандартизації процесу тестування. Вони допомагають тестувальникам систематично перевіряти функціональність та якість програмного продукту.

5.2.2 Результати випробувань

Далі описані результати проведених експериментів системи, а саме перевірка основних функцій мобільного додатку. Результати випробувань наведені у таблицях 5.1 – 5.6.

					ІС93.240БАК.004 ПЗ	Арк.
						57
Зм.	Лист	№ докум.	Підпис	Дата		

Таблиця 5.1 – Тестування реєстрації користувача

Мета	Зареєструвати користувача
Передумова	Користувач повинен відкрити сторінку реєстрації
Вхідні дані	Користувач надає свої дані
Кроки	1. У полі «Name» ввести ім'я, по бажанню ще прізвище, англійською мовою. Наприклад «Anna Sokolnytska». 2. У полі «Email» ввести електронну пошту користувача англійською мовою. Наприклад «annasokolnytska@gmail.com» 3. У полі «Password» ввести пароль. 4. Натиснути кнопку «Sing up».
Очікувані результати	Система зареєструвала користувача.
Критерії прийнятості	Система перенаправляє користувача до головного меню додатку.

Таблиця 5.2 – Тестування функції запуску таймеру

Мета	Запустити таймер.
Передумова	Користувач повинен знаходитись на сторінці «Timer».
Вхідні дані	Параметри таймера.

Продовження таблиці 5.2

Кроки	1. Користувач обирає Таймер. 2. Користувач вводить час роботи таймеру та запускає його.
Очікувані результати	Система запустить таймер.
Критерії прийнятості	Користувач може спостерігати за роботою таймера а саме бачити цифри на сторінці «Timer» у додатку.

Таблиця 5.3 – Тестування функції повідомлення-нагадування

Мета	Система буде періодично надсилати користувачеві повідомлення-нагадування про відпочинок.
Передумова	Користувач знаходиться на сторінці додатку «Break».
Вхідні дані	Параметри повідомлень.
Кроки	1. Користувач у полі «Break Type» вказує тривалість відпочинку. 2. У полі «Interval» користувач вказує інтервал часу з яким будуть надсилатись повідомлення-нагадування. 3. Натискає кнопку «Start notifications».

Продовження таблиці 5.3

Очікувані результати	Система запускає функцію повідомлень і користувач отримує їх із заданими параметрами раніше.
Критерії прийнятості	Система запустила функцію повідомлень і користувач отримує/отримав їх із заданими параметрами раніше.

Таблиця 5.3 – Тестування функції вибір активності

Мета	Обрати вибір активності, відповідну до роботи.
Передумова	Користувач знаходиться на сторінці додатку «Activity».
Вхідні дані	Користувач задає тип активності.
Кроки	1. Користувач обирає будь який один тип активності.
Очікувані результати	Система прийме параметр, та направить користувача до сторінки із запропонованими вправами до відповідного типу активності/роботи.
Критерії прийнятості	Система прийняла параметр, та направила користувача до сторінки із запропонованими вправами до відповідного типу активності/роботи.

Висновки до розділу

У технічному розділі були наведенні інструкції користувача, описані дії для запуску роботи та завершення програми. Було проведене тестування системи і наведені тест-кейси, для визначення відповідності вимогам функціональних систем.

Були проведені такі тест-кейси, як тестування реєстрації користувача, функції запуску таймеру, функції вибір активності та функції повідомлення-нагадування із показом всіх кроків метою, передумовою, вхідними даними, очікуваними результатами та критеріями прийнятості, тобто те що користувач буде очікувати після виконання тих чи інших кроків у додатку.

					ІС93.240БАК.004 ПЗ	Арк.
						61
Зм.	Лист	№ докум.	Підпис	Дата		

ВИСНОВКИ

Цей дипломному проєкт присвячений розробці мобільного додатку для контролювання проведеного часу за роботою та відпочинком та уникнення перевтомлення користувача.

Метою створення цього мобільного додатку було підвищення ефективності робочого дня користувача за рахунок контролювання проведеного часу за роботою та відпочинком та уникнення перевтомлення.

Розділ загальних положень описує основні етапи проєктування системи, а саме: аналіз предметного середовища та процесу діяльності; визначення акторів, опис задач, які потрібно виконати для досягнення поставленої мети.

У розділі інформаційного забезпечення були визначені вхідні та вихідні дані додатку.

Розділ алгоритм роботи застосунку присвячений задачі роботи додатка. У ньому була наведена змістовна постановка задачі та описана математична постановка задачі. Було описано алгоритм розробки додатка.

Розділ програмного забезпечення описує обрані засоби розробки та стек технологій для реалізації інформаційної системи складання розкладу. Також була описана архітектура програмного забезпечення та наведені діаграми класів, послідовності та компонентів для кращого представлення архітектури системи. Були описані основні класи програмного забезпечення та специфікації їх функцій. Для успішного функціонування застосунку були визначені загальні вимоги до технічного забезпечення.

Тестувальний розділ присвячений керівництву користувача, була наведена послідовність дій, яку користувачу необхідно виконати для досягнення певної мети. Графічно були проілюстровані екранні форми мобільного застосунку та наведено короткий опис звітів.

					IC93.240БАК.004 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		62

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Топ-7 додатків які допоможуть сконцентруватися на завданнях
[Електронний ресурс] – Режим доступу до ресурсу:
<https://www.imena.ua/blog/the-7-best-apps-to-help-you-focus/>
2. Android studio [Електронний ресурс] –Режим доступу до ресурсу:
<https://developer.android.com/studio/features>
3. Android SDK[Електроннийресурс]–Режим доступу до ресурсу:
<https://www.techopedia.com/definition/4220/android-sdk>
4. What is Java? Definition, Meaning & Features of Java Platforms
[Електронний ресурс] – Режим доступу до ресурсу:
<https://www.guru99.com/java-platform.html>
- 5.XML ESSENTIALS[Електронний ресурс]–Режим доступу до ресурсу:
<https://www.w3.org/standards/xml/core>
- 6.Топ-7 додатків які допоможуть сконцентруватися на завданнях
[Електронний ресурс] – Режим доступу до ресурсу:
<https://www.imena.ua/blog/the-7-best-apps-to-help-you-focus/>
- 7.10 застосунків, які допоможуть повернути контроль над життям
[Електронний ресурс] – Режим доступу до ресурсу:
<https://suspilne.media/55352-10-zastosunkiv-aki-dopomozut-povernuti-kontrol-nad-zittam/>
8. Приклад використання Java Timer і TimerTask [Електронний ресурс]–
Режим доступу до ресурсу:
<https://javadevblog.com/primer-ispol-zovaniya-java-timer-i-timertask.html>
9. Android – Повідомлення [Електронний ресурс]–Режим доступу до ресурсу:
[https://coderlessons.com/tutorials/mobilnaia-razrabotka/uchitsia-android-ovedomleniia](https://coderlessons.com/tutorials/mobilnaia-razrabotka/uchitsia-android/android-ovedomleniia)

10. Увідомлення в Android. Notification + PendingIntent на прикладі

[Електронний ресурс]–Режим доступу до ресурсу:

<https://javadevblog.com/uvedomleniya-v-android-notification-pendingintent-na-primere.htm>

11. Уроки Android Studio [Електронний ресурс]–Режим доступу до ресурсу:

<https://itproger.com/ua/course/java-android>

12. Загальні відомості про бази даних. [Електронний ресурс]–Режим доступу до ресурсу:

<https://kppk.com.ua/ELLIB/ebook/Gorbenko/IKT/10/10.htm>

14. Пов'язування таблиць баз даних [Електронний ресурс]–Режим доступу до ресурсу:

https://kafinfo.org.ua/files/Informatyka_10_11/Glava_9_47.pdf

15. Програмування в Java [Електронний ресурс]–Режим доступу до ресурсу:

https://dut.edu.ua/uploads/l_1364_67332457.pdf

16. Частина 3: Взаємодія з користувачем [Електронний ресурс]–Режим доступу до ресурсу:

<https://code.makery.ch/uk/library/javafx-tutorial/part3/>

					IC93.240БАК.004 ПЗ	Арк.
Зм.	Лист	№ докум.	Підпис	Дата		64