**Министерство образования Республики Беларусь**

**Учреждение образования**

**«Международный государственный экологический   
институт имени А.Д. Сахарова Белорусского государственного университета»**

Факультет мониторинга окружающей среды

Кафедра экологических информационных систем

**Разработка мобильного приложения, информирующего население об индексе ультрафиолетового излучения в областных центрах республики Беларусь**

Дипломная работа студента IV курса

НЕСТЕРОВИЧА Константина Дмитриевича

\_\_\_\_\_\_\_\_\_\_К. Д. Нестерович

«Допустить к защите»

Зав. кафедрой ЭИС,

к.ф.-м.н., доцент

\_\_\_\_\_\_\_\_\_\_\_А. В. Иванюкович

Научный руководитель

к.ф.-м.н., доцент

\_\_\_\_\_\_\_\_\_\_\_\_\_А.Л. Карпей

\_\_\_\_\_\_\_\_\_\_\_\_\_А.Л. Карпей

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2019г.

Минск, 2019

**Министерство образования Республики Беларусь**

Белорусский государственный университет

Международный государственный экологический институт   
им. А.Д. Сахарова

Факультет мониторинга окружающей среды

Кафедра экологических информационных систем

**ЗАДАНИЕ НА ДИПЛОМНУЮ РАБОТУ**

Студент: Нестерович Константин Дмитриевич

студент 4 курса,

направление специальности:

«Информационные системы и технологии (в экологии)»

1. Тема: Разработка мобильного приложения, информирующего население об индексе ультрафиолетового излучения в областных центрах республики Беларусь

2 Срок представления дипломной работы   
к защите\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3 Исходные данные для научного исследования (проектирования): информационные web-ресурсы, предоставляющие информация по показателю индекса ультрафиолетового излучения

4 Содержание дипломной работы

4.1 Ультрафиолетовое излучение. Теоретические сведения.

4.2 Подбор инструментария для выполнения поставленной задачи

4.3 Проектирование и разработка приложения

Руководитель дипломной работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*подпись, дата инициалы, фамилия*

Задание принял к исполнению \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*подпись, дата*

**Реферат**

Дипломная работа 50 страниц, 20 рисунков, 9 источников, 7 приложений

УЛЬТРАФИОЛЕТОВОЕ ИЗЛУЧЕНИЕ, JAVA, УЛЬТРАФИОЛЕТОВЫЙ ИНДЕКС, РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ, ANDROID STUDIO.

Объект исследования – индекс ультрафиолетового излучения.

Предмет исследования – программное обеспечение для мобильных устройств

Цель работы – спроектировать и разработать приложение, которое позволит информировать пользователя об индексе ультрафиолетового излучения.

Задачи дипломной работы:

1. Обзор информации по теме ультрафиолетового излучения
2. Определить набор инструментов, необходимый и достаточный для реализации данного приложения
3. Спроектировать и разработать приложение

В результате проделанной работы спроектировано и разработано приложение, позволяющее информировать пользователей об индексе ультрафиолетового излучения в областных центрах республики Беларусь. Данное приложение позволяет определить максимально рекомендованное время нахождения под воздействием солнечного ультрафиолетового излучения без дополнительной защиты.

**Рэферат**

Дыпломная работа 50 старонак, 20 малюнкаў, 9 крыніц, 7 прыкладанняў

УЛЬТРАФІЯЛЕТАВАЕ ВЫПРАМЕНЬВАННЕ, УЛЬТРАФІЯЛЕТАВЫ ІНДЭКС, JAVA, РАСПРАЦОЎКА МАБІЛЬНАГА ДАДАТКА, ANDROID STUDIO.

Аб'ект даследавання - індэкс ультрафіялетавага выпраменьвання.

Прадмет даследавання - праграмнае забеспячэнне для мабільных прылад.

Мэта работы - спраектаваць і распрацаваць дадатак, які дазволіць інфармаваць карыстальніка аб індэксе ультрафіялетавага в ыпраменьвання.

Задачы дыпломнай работы:

1. Агляд інфармацыі па тэме ультрафіялетавага выпраменьвання
2. Вызначыць набор інструментаў, неабходны і дастатковы для рэалізацыі дадзенага дадатка
3. Спраектаваць і распрацаваць дадатак

У выніку праведзенай работы спраектован і распрацован дадатак, які дазваляе інфармаваць карыстальнікаў аб індэксе ультрафіялетавага выпраменьвання ў абласных цэнтрах рэспублікі Беларусь. Дадзенае прыкладанне дазваляе вызначыць максімальна рэкамендаванае час знаходжання пад уздзеяннем сонечнага ўльтрафіялетавага выпраменьвання без дадатковай абароны.

**Abstract**

Graduate work 50 pages, 20 figures, 9 sources, 7 applications

ULTRAVIOLET RADIATION, JAVA, ULTRAVIOLET INDEX, DEVELOPMENT OF A MOBILE APPLICATION, ANDROID STUDIO.

The object of research - the index of ultraviolet radiation.

The Subject of research - software for mobile devices.

The purpose of the work is to design and develop an application that will inform the user about the ultraviolet radiation index.

Tasks of the graduate work:

1. Overview of ultraviolet radiation information
2. Determine the set of tools necessary and sufficient to implement this application.
3. Design and develop an application

As a result of the work done, an application was developed and developed to inform users about the ultraviolet radiation index in the regional centers of the Republic of Belarus. This application allows you to determine the maximum recommended time spent under the influence of solar ultraviolet radiation without additional protection.

Оглавление

[ВВЕДЕНИЕ 8](#_Toc11053756)

[ГЛАВА 1. Ультрафиолетовое излучение 10](#_Toc11053757)

[1.1 Ультрафиолетовый индекс 12](#_Toc11053758)

[1.2 Экологические факторы, влияющие на показатель ультрафиолетового индекса 13](#_Toc11053759)

[1.3 Вычисление индекса ультрафиолетового излучения 14](#_Toc11053760)

[1.4 Типы кожи человека 14](#_Toc11053761)

[1.5 Минимальная эритемная доза 15](#_Toc11053762)

[ГЛАВА 2. Подбор инструментария для выполнения поставленной задачи 18](#_Toc11053763)

[2.1 Обоснование требований к среде разработки 18](#_Toc11053764)

[2.2 Apache Cordova 18](#_Toc11053765)

[2.3 Xamarin 20](#_Toc11053766)

[2.4 Android Studio 21](#_Toc11053767)

[2.5 Окончательный выбор среды разработки и обоснование выбора 22](#_Toc11053768)

[2.6 Модуль Andtroid Studio AVD 22](#_Toc11053769)

[2.7 Genymotion 24](#_Toc11053770)

[2.8 Система контроля версий – Git 24](#_Toc11053771)

[2.9 Объектно-ориентированный язык Java 26](#_Toc11053772)

[ГЛАВА 3. Проектирование и разработка приложения 27](#_Toc11053773)

[3.1 Проектирование и описание модулей приложения 27](#_Toc11053774)

[3.1.1 Выделение основных модулей приложения 27](#_Toc11053775)

[3.1.2 Модуль хранения данных 28](#_Toc11053776)

[3.1.3 Модуль интеграции иконок 30](#_Toc11053777)

[3.1.4 Модуль определения пространственных координат 32](#_Toc11053778)

[3.1.5 Модуль навигации в приложение вкладки 36](#_Toc11053779)

[3.1.6 Модуль парсинга страниц 37](#_Toc11053780)

[3.1.7 Модуль расчёта максимального времени пребывания под воздействием ультрафиолетового излучения 38](#_Toc11053781)

[3.2. Разработка алгоритмов модулей и их реализация 39](#_Toc11053782)

[3.2.1 Описание процесса реализации модулей приложения 39](#_Toc11053783)

[3.2.2 Алгоритм модуля определения пространственных координат. 39](#_Toc11053784)

[3.2.3 Алгоритм модуля парсинга страницы 40](#_Toc11053785)

[3.2.4 Алгоритм модуля расчёта максимального времени пребывания под воздействием ультрафиолетового излучения 42](#_Toc11053786)

[3.3 Описание работы пользователя с приложением 43](#_Toc11053787)

[3.3.1 Главная страница приложения 43](#_Toc11053788)

[3.3.2 Страница приложения с прогнозом ультрафиолетового индекса 44](#_Toc11053789)

[3.3.3 Страница с калькулятором 45](#_Toc11053790)

[3.3.4 Страница настроек пользователя 46](#_Toc11053791)

[3.3.5 Переход между страницами приложения 48](#_Toc11053792)

[3.3.6 Аппаратно-технические требования приложения 49](#_Toc11053793)

[ЗАКЛЮЧЕНИЕ 51](#_Toc11053794)

[СПИСОК ЛИТЕРАТУРЫ 52](#_Toc11053795)

[ПРИЛОЖЕНИЕ 53](#_Toc11053796)

ВВЕДЕНИЕ

Солнце является источником тепла и энергии, однако вместе с лучами солнца на планету воздействует ультрафиолетовое излучение, которое, в больших количествах, представляет опасность для живых организмов. Однако жизнь на поверхности планеты существует, а связанно это с озоновым слоем, который отражает большую часть вредоносного излучения обратно в космос. В результате чего только малая его часть достигает поверхности земли. Несмотря на то, что поверхности планеты достигает лишь малая часть ультрафиолетового излечения, для живых организмов длительное облучение представляет серьезную опасность. Связанно это с тем, что длительное воздействие ультрафиолета на поверхность кожи, вызывает ожоги, а также может способствовать злокачественному перерождению клеток кожи. Таким образом, пренебрежительное отношение к данному излучению может привести к негативным последствиям. Для того, чтобы следить за показателем ультрафиолетового излучения, сформулирована характеристика уровня воздействия ультрафиолетового излучения, достигающего поверхности земли, имеющая название – индекс ультрафиолетового излучения. Данная характеристика используется для определения максимально рекомендованного времени пребывания под воздействием ультрафиолета, что позволяет избежать пагубного воздействия на здоровье человека. Для того, чтобы такая информация была доступна населению, используются различные средства оповещения, такие как утренний прогноз погоды на государственных каналах, сайты с картами ультрафиолетового излучения, онлайн сервисы по предоставлению информации об ультрафиолетовом индексе. Для дальнейшего повышения уровня информированности граждан об ультрафиолетовом индексе можно использовать различные методы. Один из них – это создание приложение для мобильных устройств, которое в любой момент позволит определить ультрафиолетовый индекс в месте нахождения пользователя, а также позволит рассчитать максимальное время пребывания на открытой местности, находящейся под воздействием ультрафиолетового излучения. Пользователь данного приложения сможет получить информацию об индексе ультрафиолетового излучения в соседних городах, а также прогноз на последующие дни. Тем самым позволит оперативно среагировать на возможную опасность в связи с погодными условиями.

Цель данной дипломной работы – спроектировать и разработать приложение, которое позволит информировать пользователя о показателе индекса ультрафиолетового излучения.

Для достижения поставленной цели формируется список задач:

1) Изучение информации по теме ультрафиолетового излучения

2) Определить набор инструментов, необходимый и достаточный для реализации данного приложения

3) Спроектировать и разработать приложение

Данное приложение позволит увеличить уровень информированности населения об ультрафиолете. Пользователи данного приложения получат информацию об индексе ультрафиолетового излучения в областных центрах республики Беларусь. В перспективе данное приложение может быть полезно для курортных регионов, где отдыхающие большую часть времени находятся под воздействием ультрафиолетового излучения.

ГЛАВА 1. Ультрафиолетовое излучение

Ультрафиолетовое излучение является электромагнитным излучением, которое занимает спектральный диапазон между видимым и рентгеновским излучением. Длины волн УФ-излучения лежат в интервале от десяти до четырехсот нанометров. Термин происходит от лат. ultra – сверх, за пределами и фиолетовый (violet). В разговорной речи может использоваться также наименование «ультрафиолет» [1].

Ультрафиолет является неотъемлемой частью жизни на земле, так как его воздействие непрерывно. Большая часть ультрафиолетового излучения приходит на поверхность из космоса, в частности от солнца. Ультрафиолетовое излучение оказывает воздействие на организм человека. В малых доза ультрафиолетовое излучение полезно, в качестве профилактической меры, а именно укрепляет организм, способствуя выработке витамина D. Так же ультрафиолет позволяет обеззараживать воздух и воду. Однако большие дозы ультрафиолета оказывают негативное влияние не только на организм человека, но и на всю экосистему в целом.

Солнце является основным источником оптического излучения, поступающего на поверхность земли. От общего количества поступающего излучения, ультрафиолетовое излучение составляет 5%, в то время как на видимый спектр приходится порядка 39% , а на инфракрасный 56% (Рисунок 1.1).

Ультрафиолетовое излучение также считается биологически активным, так как оказывает влияние на живые организмы в большей степени, чем инфракрасное и видимое.

Ультрафиолетовое излучение подразделяется на 3 типа: УФА (коротковолновой диапазон), УФВ (средневолновой диапазон), УФС (длинноволновой диапазон).

УФА, с длиной волны 400-320нм, является коротковолновым излучением, хорошо проникает в кожу и практически не поглощается в атмосфере, поэтому большая его часть достигает поверхности земли.

УФВ, с длиной волны 320-280нм, является средневолновым излучение, большая часть которого поглощается стратосферным озоном, вследствие чего, только незначительная часть достигает поверхности земли.

УФС, с длиной волны 280-200нм, является коротковолновым излучением, обладает бактерицидными свойствами, однако полностью поглощается в стратосфере.

Количество УФ излучения, доходящего до поверхности земли, зависит непосредственно от толщины озонного слоя, облачности и положения солнца над поверхностью. Максимальное количество ультрафиолетового излучения, доходящего до поверхности земли, на территории республики Беларусь будет в момент полудня в районе 1300. Облачность также играет свою роль, уменьшая мощность ультрафиолетового излучения, доходящего до поверхности земли. Однако, даже при наличии густых облаков, по-прежнему может быть высокий показатель ультрафиолетового индекса.

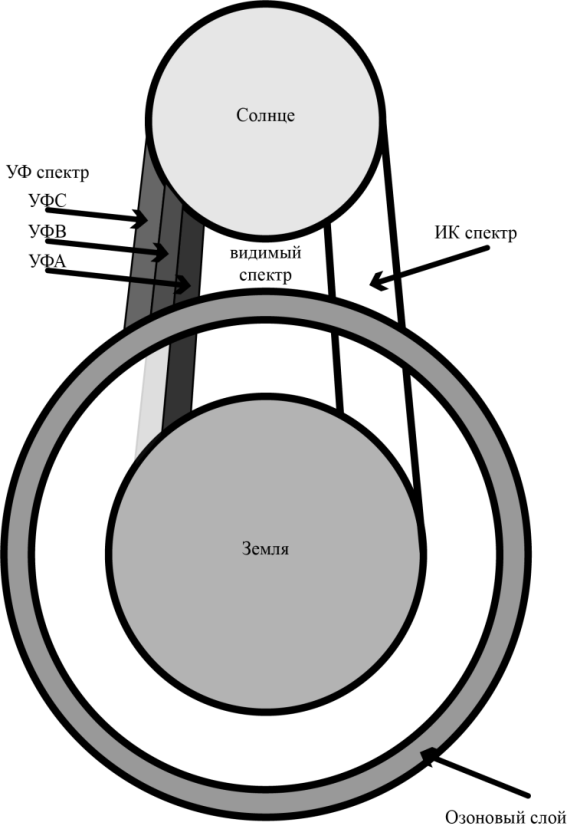


Рисунок 1.1*.* Солнечное излучение на поверхность земли

1.1 Ультрафиолетовый индекс

Ультрафиолетовый индекс демонстрирует, какое количество ультрафиолетового излучения достигает поверхности земли. В связи с тем, что ультрафиолетовое излучение в больших дозах способно навредить организму человека, необходимо относится к показателю ультрафиолетового индекса с максимальной внимательностью. Дополнительно, при высоком показателе ультрафиолетового индекса, необходимо применять меры по защите открытых участков кожи, а также глаз. Данный показатель солнечной активности не является распространенной информацией среди населения, так как большинство людей полагает, что показателем солнечной активности является температура воздуха. Индекс ультрафиолетового излучения принимает значения от нуля и выше. При этом, чем больше значение УФ-индекса, тем выше потенциальная опасность для кожи и глаз человека, и тем меньше времени, требуется для причинения вреда здоровью (Рисунок 1.2) [2]

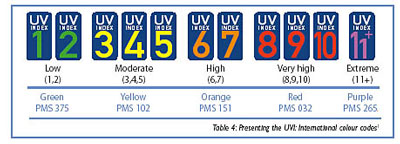


Рисунок 1.2. Шкала значений ультрафиолетового индекса

Показатель температуры воздуха используется населением при принятии мер по защите от пагубного воздействия солнца или, иначе, ультрафиолетового излучения солнца. Как бы то ни было показатель температуры воздуха не определяет вероятность получения ожога. Так как температура воздуха не зависит от количества ультрафиолетового излучения доходящего до поверхности земли. Таким образом, летом температура может быть не высокой из-за высокой облачности, однако показатель ультрафиолетового излучения будет высоким. В результате, человек совершит ошибку, опираясь на показатель температуры, при определении вероятности получения ожога.

1.2 Экологические факторы, влияющие на показатель ультрафиолетового индекса

Показатель ультрафиолетового индекса не является независимым параметром, так как существуют факторы, которые влияют на его значение. Одним из таких факторов является высота солнца. Чем выше находится солнце, тем выше показатель индекса ультрафиолетового излучения. Таким образом, количество ультрафиолетового излучения различно как в течение суток, так и в течение года, а максимальный показатель ультрафиолетового индекса в течение дня будет зафиксирован, когда солнце находится в зените, в районе часа дня на территории республики Беларусь [3].

Также важным фактором, влияющим на показатель ультрафиолетового индекса, является географическая широта. Чем ближе к экватору, тем выше показатель индекса ультрафиолетового излучения.

Состояние облачного покрова непосредственно влияет на показатель индекса ультрафиолетового излучения. Индекс принимает максимальные значения при безоблачном небе. При наличии облаков показатель индекса ультрафиолетового излучения уменьшается, но по-прежнему может принимать высокие значения. Так как ультрафиолетовые лучи рассеиваются молекулами воды и мельчайшими частицами в атмосфере, тем самым достигая поверхности земли через облачный покров в меньшем количестве.

Важным фактором, влияющим на показатель индекса ультрафиолетового излучения, является высота над уровнем моря. На значительной высоте атмосфера более разряженная и она легче пропускает ультрафиолетовое излучение. Таким образом, с увеличение показателя высоты над уровнем моря за каждую тысячу метров уровень ультрафиолетового излучения возрастает на десять, двенадцать процентов.

Озоновый слой так же является фактором, влияющим на показатель индекса ультрафиолетового излучения. Озоновый слой поглощает часть ультрафиолетовых лучей, которые могли бы достигнуть земной поверхности. Концентрация озона в атмосфере меняется как в течение суток, так и в течение года.

Способность поверхности отражать ультрафиолетовое излучение не что иное, как фактор, влияющий на показатель ультрафиолетового индекса. Количество отраженных лучей напрямую зависит от типа поверхности. Снежный покров позволяет отражать до восьмидесяти процентов пришедшего на поверхность ультрафиолетового излучения, в то время как сухой пляжный песок отражает пятнадцать процентов.

1.3 Вычисление индекса ультрафиолетового излучения

Ультрафиолетовый индекс определяется как дневной максимум биологически активной облученности или экспозиции, умноженной на 40. Его размерность – Вт/м2. Ниже приведена формула (1), по которой вычисляется УФ индекс (английская аббревиатура – UVI) [4]:

(1)

где E(λ) – интенсивность приземной солнечной радиации, К(λ) – спектр действия эритемы, утвержденный Международной комиссией по освещенности (CIE). Для безоблачного неба ультрафиолетовый индекс соответствует значению облученности в момент истинного полудня и характеризует максимально возможный риск ультрафиолетового облучения в течение дня. Коэффициент умножения 40 используется для более удобного представления значений ультрафиолетового индекса.

1.4 Типы кожи человека

Организм человека – это очень сложная система, состоящая из таких же сложных по структуре органов. В связи с тем, что не существует два полностью идентичных человека, применяют классификацию по признакам. Признаками можно считать отдельные внешние черты человека, такие как цвет кожи или цвет волос. Благодаря классификации можно выделять группы людей, у которых организм будет реагировать в равной степени на одни и те же раздражители.

Существует шесть основных типов кожи, отличающийся друг от друга чувствительностью к ультрафиолетовому излучению. Для республики Беларусь характерны четыре первых из шести возможных типов кожи (Рисунок 1.3) [5].

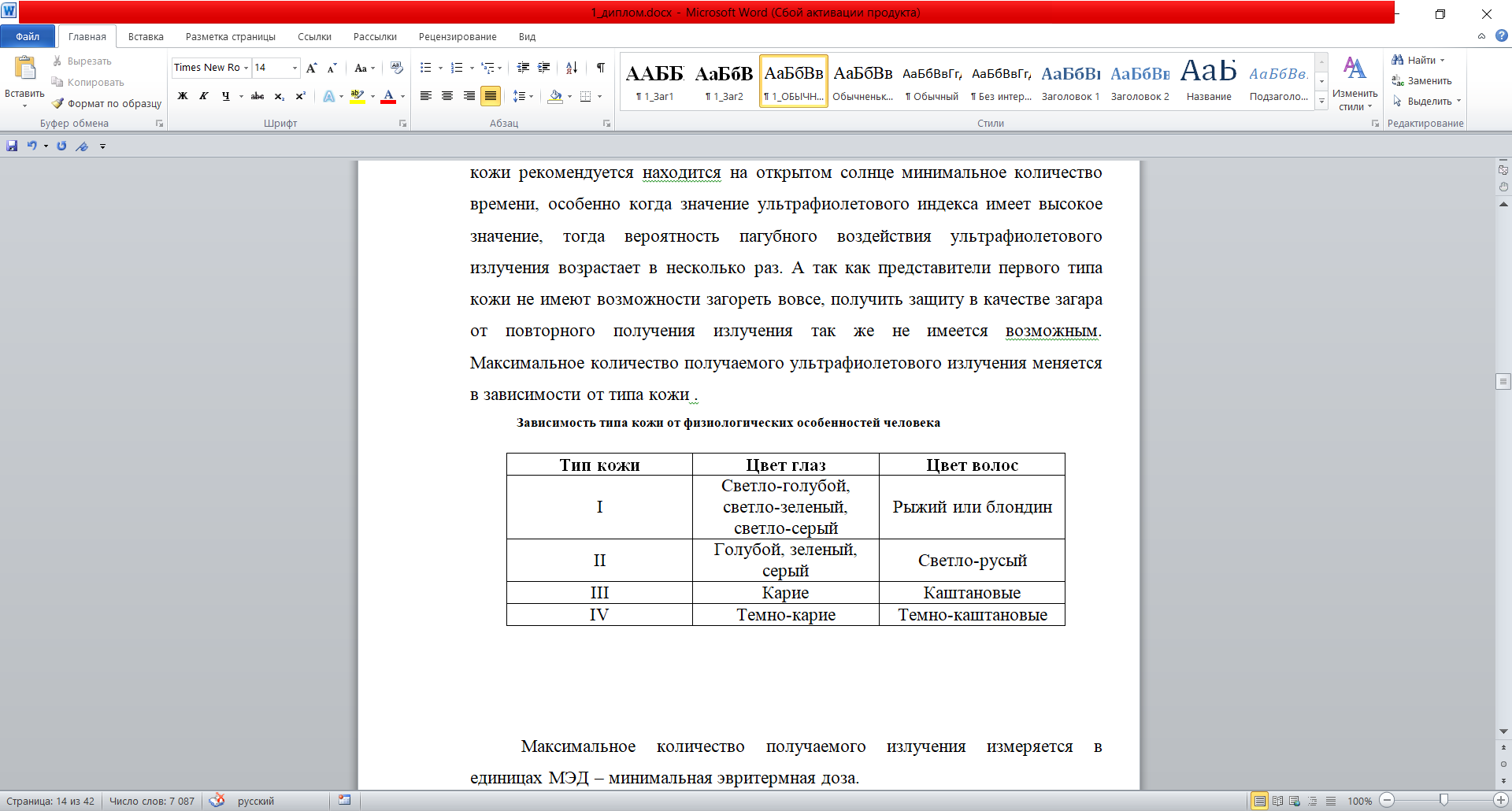


Рисунок 1.3. Типы кожи хараетрные для беларуси и физические парамтеры человека, описывающие их.

Каждый из типов кожи имеет уникальный набор свойств, который и позволяет определить чувствительность кожи конкретного человека к ультрафиолетовому излучению. Людей с белой кожей и наличием веснушек относят к представителям с первым типом кожи, которые являются наиболее чувствительными к ультрафиолетовому излучению. Людям с первым типом кожи рекомендуется находиться на открытом солнце минимальное количество времени, особенно, когда показатель ультрафиолетового индекса имеет высокое значение. Тогда пагубное воздействие ультрафиолетового излучения возрастает в несколько раз. Связано это также с тем, что представители первого типа кожи не имеют возможности загореть вовсе.

Каждый тип кожи определяет максимально безопасное количество получаемого излучения, которое измеряется в единицах МЭД – минимальная эритемная доза.

1.5 Минимальная эритемная доза

Минимальная эритемная доза измеряется в джоулях на квадратный метр и принимает значения в пределах от двухсот до двухсот пятидесяти джоулей на квадратный метр. Это количество ультрафиолетового излучения, которое приходится на поверхность, в данном случае на человеческий кожный покров. Эта величина измерена опытным путем с использованием искусственных источников ультрафиолетового излучения. Таким образом имеется придел получения дозы ультрафиолетового излучения, измеряемый в единицах минимальной эритемной дозы. Первому типу кожи соответствует 0,8 единиц минимальной эритемной дозы, второму, третьему и четвертому – 1, 1.4, 1.8 единиц соответственно. Данный показатель демонстрирует максимальное количество излучения, которое может принять без последствий представитель каждого типа кожи.

При планировании загара также необходимо учитывать, что при первом загаре требуется уменьшить значение минимальной эритемной дозы до значения, равного 40-ка процентам от начального значения минимальной эритемной дозы, так как организм не подготовлен к ультрафиолетовому излучению, не имея защиты в виде первичного загара. Так же необходимо учитывать, что представитель любого типа кожи может иметь особенности кожного покрова, такие как папилломы, родимые пятна больших размеров, шрамы и т.п. Данные участки кожи приставляют собой уязвимые места, так как структура кожи в этих местах имеет поврежденную защитную оболочку, а также слой кожного покрова в таких местах тоньше, следовательно, ультрафиолетовое излучение проникает вглубь кожных тканей и наносит вред кожному покрову в этих места куда сильнее.

Существует категория людей, которые не зависимо от типа кожи, имеют большую вероятность получения негативного воздействия ультрафиолетового излучения. К данной категории относятся дети, так как их кожный покров намного тоньше, чем у взрослого человека. В данном случае дети относят к представителям первой категории, когда ультрафиолетовое излучение может воздействовать на организм человека даже при небольших показателях ультрафиолетового индекса.

Международными экологическими организациями рекомендуется ограничивать ежегодное воздействие ультрафиолетового излучения шестьюдесятью единицами минимальной эритемной дозы. С учетом, что в сутки количество поглощённого ультрафиолетового излучения не будет превышать норму для соответствующего типа кожи [6].

ГЛАВА 2. Подбор инструментария для выполнения поставленной задачи

2.1 Обоснование требований к среде разработки

Для разработки приложения необходимо определить среду разработки. Главным аспектом, при выборе среды разработки, является возможность выполнения всех поставленных задач. Так же, в процессе анализа доступных средств разработки, необходимо уделять внимание такому параметру как доступность. Это значит, что платформа для разработки мобильных приложений должна быть в свободном доступе и иметь версию, не требующую крупных денежных инвестиций. Так же не мало важным параметром является время, необходимое для изучения данного инструмента разработки приложений. Так как время для достижения поставленной цели ограничено, то необходимо исключать среды с высоким порогом освоения интерфейса и принципов разработки. Еще один параметр – это обширный набор библиотек, находящихся так же в свободном доступе и имеющих актуальные версии, чтобы поставленные задачи были выполнены своевременно. Последним немаловажным параметром является наличие обучающего материала для данной среды разработки, так как процесс изучения нового инструмента должен проходить в максимально короткие сроки, чтобы хватило времени на выполнение поставленных задач. Следовательно, обучающий материал должен быть понятным и доступным для новых разработчиков. Результатом выбора среды разработки должен быть удобный инструмент, позволяющий выполнять поставленные задачи, а также изучение его основных возможностей не должно занимать много времени. Для более детального анализа были выбраны такие среды разработки как: Apache Cordova, Xamarin и Android Studio.

2.2 Apache Cordova

Apache Cordova позволяет использовать стандартные интернет технологии, такие как HTML5, CSS3 и JavaScript [7]. Данная среда позволяет создавать кросс платформенные приложения, не обращая внимания на языки, которые используются для каждой из платформ в отдельности, такие как Java для Android или C++ для IOS. Однако компиляция по-прежнему требует специальных программных и аппаратных средств, как для Android, так и для IOS. Это означает, что без доступа к устройству Apple и среде разработки для IOS Xcode получить готовый продукт для устройств с операционной системой IOS будет невозможно, соответственно возможность распространения готового продукта для устройств с операционной системой IOS будет также недоступна. Разработка под операционную систему Android является более простым процессом, она возможна без специальных аппаратных средств. Для успешной компиляции готового продукта необходимо получить набор инструментов разработчика Android SDK – Software Development Kit. Который позволит компилировать готовый продукт для распространения и тестирования на устройствах с операционной системой Android. Однако в последних версиях можно получить лишь основные элементы, которые включат последнюю доступную версию Android, это означает, что компиляция возможна только для последней версии Android, остальное версии и дополнительные инструменты для работы с возможностями устройств необходимо загрузить, используя среду разработки Android Studio. Таким образом, чтобы разрабатывать на данной платформе, необходимо понимать основы работы интерфейсов двух смежных сред разработки Xcode для IOS и Android Studio для Android. Так же для работы с Apache Cordova необходимо иметь среду-редактор, в которой возможно работать с проектами Apache Cordova. Такой платформой может стать Visual Studio, там имеется возможность установить Apache Cordova в качестве отдельного инструмента, для дальнейшей работы с ним. Данная среда разработки представляет большой интерес из-за своей кроссплатформенности и простоты в реализации задач. Однако количество времени, необходимого для изучения данной среды с целью получения необходимых результатов, превышает допустимое значение. Таким образом, данная среда после ознакомления не рассматривается как инструмент для реализации поставленной цели.

2.3 Xamarin

Xamarin – среда разработки, которая, как и Apache Cordova, позволяет разрабатывать кроссплатформенные приложения как под платформы с операционной системой IOS, так и под операционные системы Android. В среде разработки Xamarin используется объектно-ориентированный язык программирования C#. Данная среда позиционировалась не как инструмент, а как самостоятельная среда разработки, однако в 2016 коду ситуация изменилась, когда компания Microsoft выкупила компанию Xamarin и отныне, среда разработки Xamarin стала инструментом в среде разработки Microsoft Visual Studio [8]. Однако в отличие от Apache Cordova, Xamarin разделяет разработку под Android и IOS на отдельные части, он позволяет разрабатывать под операционные системы как IOS, так и Android, однако различными способами. Важным аргументом против Xamarin выступает тот факт, что работа с этим инструментом происходит в среде Visual Studio и таким образом подключение всех необходимых инструментов, таких как SDK – software development kit представляет собой сложный процесс, не всегда заканчивающийся успехом. В связи с чем польза от кроссплатформенности становится сомнительной, так как по-прежнему необходимо иметь доступ к оборудованию Apple, а также иметь лицензию Apple-разработчика для операционных систем IOS, дополнительно требуется установить среду разработки Microsoft Visual Studio для Mac и там произвести настройку для работы с инструментом Xamarin. Данная среда разработки после более детального анализа не представляет интереса в качестве среды для разработки данного приложения из-за сложной настройки и подготовки инструментов к рабочему процессу, а также в связи с тем, что доступ имеется только к разработке для операционных систем Android. Данный доступ осуществляется посредством подключения набора инструментов другой среды разработки, которая ориентирована под устройства с операционной системой Android.

2.4 Android Studio

Одной из сред разработки, с помощью которой можно реализовать мобильное приложение, является Android Studio. Рассмотрение данной среды разработки стало приемлемым в связи с тем, что большинство населения республики Беларусь используют устройства с операционной системой Android, а для разработки под операционную систему IOS, требуется специальные аппаратно-программные средства.

Android Studio, основанная на программном обеспечении IntelliJ IDEA от компании JetBrains, – официальное средство разработки Android приложений. Данная среда разработки доступна для Windows, OS X и Linux. 17 мая 2017, на ежегодной конференции Google I/O, Google анонсировал поддержку языка Kotlin, используемого в Android Studio, как официального языка программирования для платформы Android в добавление к Java и С++. Среда Android Studio имеет приятный и удобный интерфейс разработчика (Рисунок 2.1).

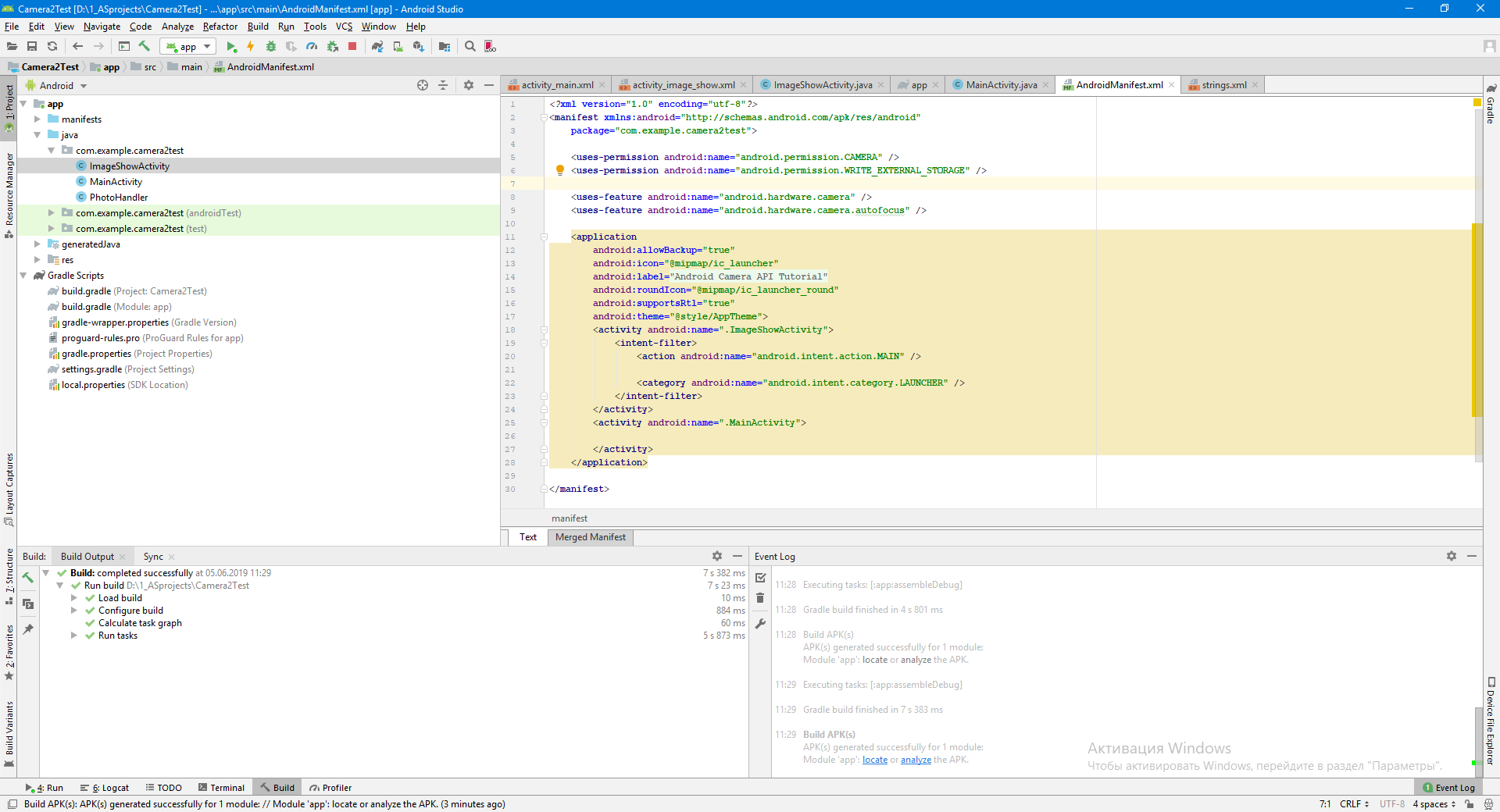


Рисунок 2.1. Интерфейс среды Android Studio

«Andtoid Studio» ­ это интегрированная среда разработки, позволяющая проектировать и разрабатывать мобильные приложения под устройства с операционной системой «Android». Данная среда позволяет использовать все возможности объектно-ориентированного языка программирования Java. А также позволяет совершать точную настройку элементов приложения, размещая их с использованием пикселей, не зависящих от плотности (Density-independent Pixel) – это относительная единица измерения, которая берет за основу плотность экрана. Примерно соответствует одному пикселю на устройстве с плотностью экрана 160 точек на дюйм. Отношение меняется в зависимости от плотности экрана.

2.5 Окончательный выбор среды разработки и обоснование выбора

В результате рассмотрения сред разработки, а также по причине истечения времени, отведенного на подбор инструментария, было принято решение разрабатывать приложение в среде разработки Android Studio. Данная среда позволяет разрабатывать приложения под операционную систему Android. Так же для данной среды существуют все необходимые для выполнения поставленных задач актуальные библиотеки. Так же существуют популярные библиотеки, для которых имеются обучающие материалы. Они позволяют быстрого изучить данные библиотеки для их дальнейшего внедрения в процесс разработки. В среде Android Studio используется базовый для операционной системы Android язык программирования Java, таким образом, некоторые функции упрощаются, а в результате разрабатываемые приложения становятся более оптимизированные в области использования, как операционной памяти, так и дискового пространства. Таким образом, среда разработки Android Studio полностью подходит для задачи разработки приложения под операционную систему Android. А доступные библиотеки позволяют решить поставленные задачи.

2.6 Модуль Andtroid Studio AVD

«Android Virtual Device» – система виртуальных устройств «Android», иначе эмулятор устройств с операционной системой «Android». Позволяет создавать виртуальные копии устройств с операционной системой Android для последующего их использования. При создании виртуального устройства имеется возможность выбрать модель эмулируемого устройства, размер и разрешение экрана, размер оперативной и видео памяти, которые выделяются из доступного резерва домашнего устройства, компьютера, на котором совершается эмуляция. Так же важной функцией эмулятора виртуальных мобильных устройств является то, что имеется возможность выбрать конкретную операционную систему Android. Так как при разработке очень важно знать, для каких устройств и с какой операционной системой данное приложение сможет работать, а с какими нет. Данная функция является очень важной при работе с последними элементами в Android API, требующие от устройства как раз последних версий операционной системы Android. Последние устройства выпускаются с версией «Android» 10. Чтобы протестировать будет ли работать данное приложение на устройстве с операционно системой «Android» 10, нет необходимости покупать новое устройства. Для этого достаточно создать виртуальное устройство с помощью встроенного в среду разработки «Andorid Stduio» эмулятора виртуальных устройств (Рисунок 2.2).

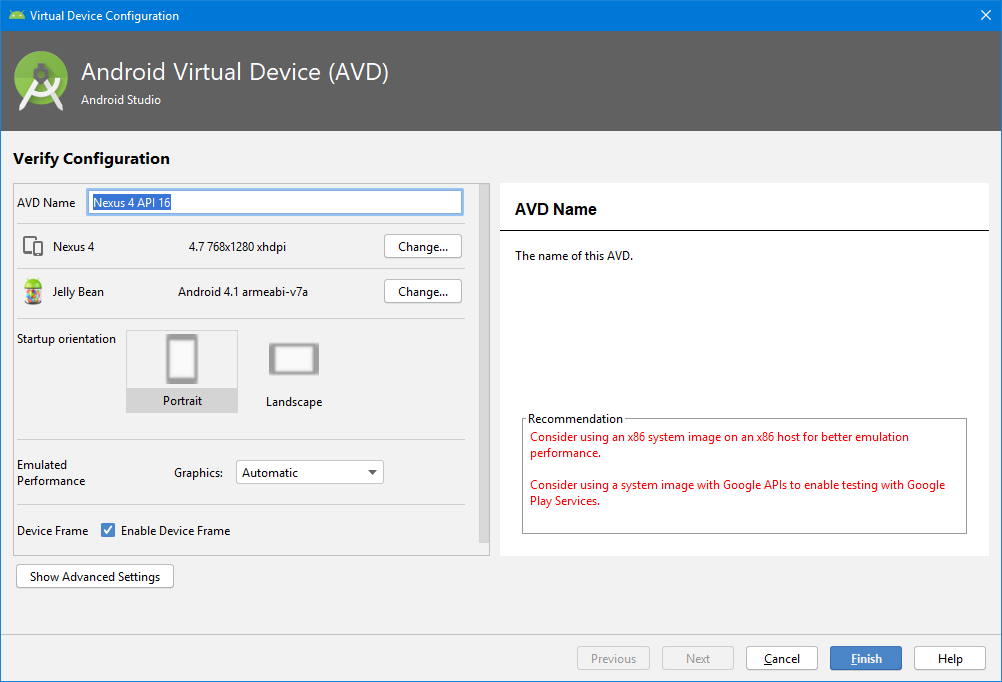


Рисунок 2.2. Окно настройки эмулятора виртуальных устройств

Так же данная функция важна для понимания минимальной границы приложения по уровню API. Все библиотеки, которые доступны разработчикам на Android Studio, сопровождаются необходимым минимум API, а также каждая функция этих библиотеки отдельно сопровождается таким же минимум API. Самый высокий показатель API из всех используемых библиотек и демонстрирует минимум API для приложения. Значение API показывает, сможет ли запустить данное приложение на данной версии операционной системе «Android». API - это отдельная шкала и она не является версией операционной системы «Android», а приставляет из себя шкалу уровня технологий «Android».

2.7 Genymotion

Приложение Genymotion создано для разработчиков мобильных приложений, ориентированных на устройства с операционной системой Android. Данное приложение позволяет создавать виртуальные устройства с операционной системой Android. Так же данное приложение позволяет создавать виртуальные устройства с собственными конфигурациями. В результате чего данное приложение позволяет производить тестирование приложения как на устройствах с передовыми характеристиками, так и на устройствах с конфигурацией старых моделей. Дополнительно данное приложение позволяет выставлять такой параметр для виртуальных устройств как версия операционной системы Android, что в процессе разработки мобильных приложений для Android является ключевым фактором. Так как приложения, разрабатываемые под последние версии операционной системы Android, не всегда имеют возможность быть установленными на устройствах со старой версией операционной системой Android. Так же данное приложение по принципу работы соответствует модулю Android Virtual Device, который встроен в среду разработки Android Studio. Однако приложение Genymotion интегрируется в среду разработки Android Studio и позволяет производить эмуляцию устройств на порядок быстрее, чем встроенный Android Virtual Device.

2.8 Система контроля версий – Git

Для реализации возможностей контроля версий существуют специальные системы. Система Git является одним из представителей систем управления версиями. Система Git позволяет хранить проекты на облачном хранилище, а также позволяет быстро и удобно работать с различными версиями проекта. В связи с тем, что данная система позволяет иметь копию существующего проекта на облачном хранилище, пользователь имеет возможность избежать потери данных из-за технических ошибок персонального оборудования. Также данная система позволяет производить манипуляции с проектом на любом устройстве, для этого необходимо скачать репозиторий проекта с сайта Git и проект будет доступен для модификации в локальном репозитории проекта. Результаты модификаций, при наличии соответствующих разрешений, отправляются обратно на облачное хранилище. Также имеется возможность просмотра репозитория на сайте Git, где представлены все файлы, входящие в репозиторий проекта (Рисунок 2.3).

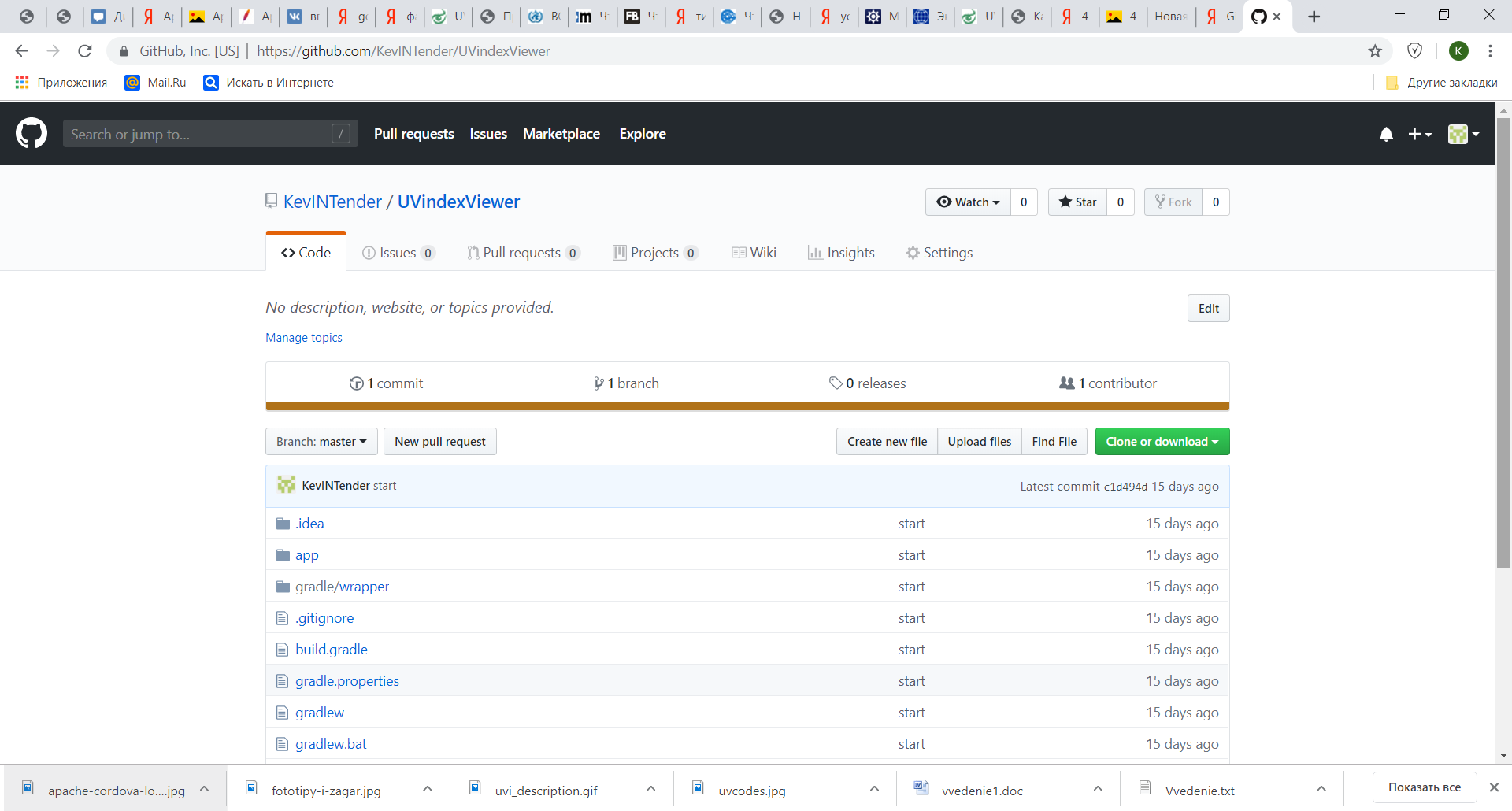


Рисунок 2.3. Репозиторий проекта, размещенный на GitHub

Таким образом, данная система позволяет разрабатывать проект группой разработчиков одновременно. При необходимости различные версии проекта можно объединять. В результате образуется новая конфигурация проекта, сочетающая в себе изменения обеих версий, которые не противоречат друг другу. Пользователю приходят оповещения о противоречиях, возникших при объединение имеющихся конфигураций проекта и способы их решения.

2.9 Объектно-ориентированный язык Java

Java **–** сильно типизированный объектно-ориентированный язык программирования, разработанный компанией Sun Microsystems (в последующем приобретённой компанией Oracle)

Данный язык программирования позволяет использовать все принципы объектно-ориентированного языка программирования, такие как инкапсуляция, полиморфизм, наследование, а также использовать их на практике для достижения поставленных целей.

Язык Java является основой операционной системы Android, поэтому при разработке приложений под данную операционную систему эффективнее всего использовать именно язык Java, так как это позволяет взаимодействовать с физическими элементами устройства. Такими элементами являются камера и память устройства, работа с которыми производиться напрямую, в результате чего не тратится время на перекомпиляцию исходного кода с других языков на Java. Данный подход используется при разработке мобильных приложений под операционную систему Android в среде разработки Android Studio.

Приложения Java обычно транслируются в специальный байт-код, поэтому они могут работать на любой компьютерной архитектуре с помощью виртуальной Java-машины.

Достоинством подобного способа выполнения программ является полная независимость байт-кода от [операционной системы](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) и [оборудования](https://ru.wikipedia.org/wiki/%D0%90%D0%BF%D0%BF%D0%B0%D1%80%D0%B0%D1%82%D0%BD%D0%B0%D1%8F_%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B0), что позволяет выполнять Java-приложения на любом устройстве, для которого существует соответствующая виртуальная машина. Другой важной особенностью технологии Java является гибкая система безопасности, в рамках которой исполнение программы полностью контролируется виртуальной машиной. Любые операции, которые превышают установленные полномочия программы (например, попытка несанкционированного доступа к данным или соединения с другим компьютером), вызывают немедленное прерывание.

ГЛАВА 3. Проектирование И РАЗРАБОТКА приложения

3.1 Проектирование и описание модулей приложения

3.1.1 Выделение основных модулей приложения

Для реализации данного приложения необходимо спроектировать основные модули. Ключевыми функциями данного приложения являются: предоставление пользователю актуальной информации о показатели ультрафиолетового индекса в областных центрах республики Беларусь, а также возможность расчёта максимального времени пребывания под воздействием ультрафиолетового излучения. Для реализации данных функций, необходимо спроектировать модуль, который позволит получать актуальную информацию о показателе ультрафиолетового индекса. Также необходимо спроектировать модуль, который позволит получать пространственные координаты пользователя и тем самым предоставлять данные, соответствующие местоположению пользователя. Также необходимо реализовать модуль расчёта максимального времени нахождения под воздействием ультрафиолетового излучения, используя полученные данные. Полученные данные также должны храниться, сохранять и загружать при необходимости. Данные функции буду реализованы с использованием модуля управления данными. В заключение, для реализованных модулей необходимо создать интерфейс, который должен включает в себя удобный и понятный модуль навигации. Структурная схема работы приложения представлена ниже (Рисунок 3.1). На данной схеме наглядно демонстрируется взаимодействие отдельных модулей приложения между собой, а также весь цикл работы приложения.

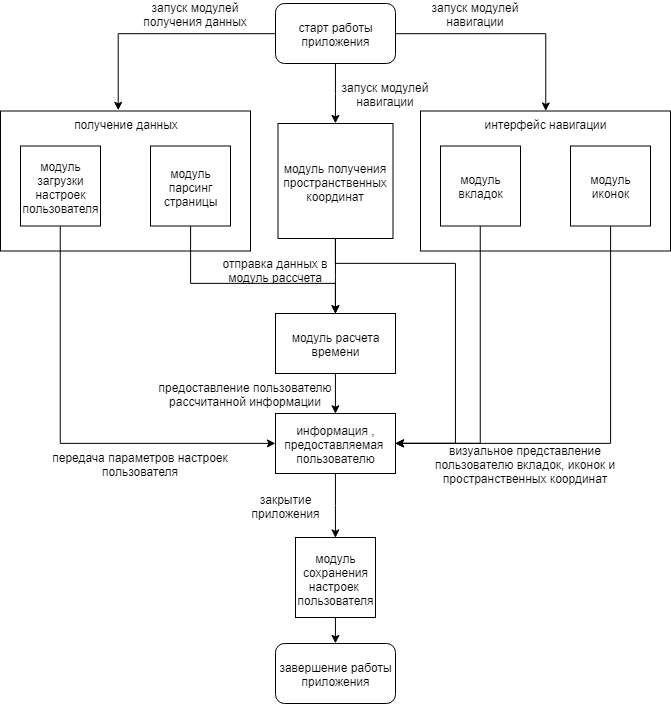


Рисунок 3.1. Структурная схема приложения

3.1.2 Модуль хранения данных

Для верного отображения информации, необходимой пользователю в момент запуска приложения, существует модуль хранения настроек пользователя. Данный модуль позволяет хранить заданные пользователем параметры работы приложения в локальной памяти приложения. Одним из параметров является тип кожи, где пользователь сравнивает свои физические параметры с имеющимся описанием четырех типов кожи и выбирает соответствующий тип кожи. Данный параметр выбирается пользователем в самом начале пользования приложения, а далее хранится в локальной памяти устройства. Данный параметр, как и остальные, могут быть изменены пользователем при необходимости в любой момент на странице настроек, перейдя на нее нажатием по вкладке настроек, а далее изменив необходимый параметр на значение, которое предоставляется пользователю на выбор. Данный модуль реализуется за счет встроенной в библиотеку android функции Preferences. Данная функция предназначена для хранения настроек пользователя во внутренней памяти приложения. Хранение во внутренней памяти представляет собой файл, создаваемый в скрытой части каталога приложения и который в дальнейшем будет доступен только данному приложению. Никакие сторонние приложения не буду иметь доступ к данному файлу, в том числе и для операции чтения, не только перезаписи. Данные свойства файла достигаются при настройке параметра MODE в функции Preferces, где устанавливается параметр PRIVATE что и означает доступность к этому файлу только приложению, создавшего его. Данная функция позволяет хранить информацию в основных типах данных, таких как строки – String , целочисленные значения – Integer, значения с плавающей точкой – Float , булевские переменные – Boollean , а так же имеется возможность сохранения списка строк String[]. Процесс сохранения данных осуществляется непосредственно после нажатия кнопки подтверждения сохранения настроек на странице настроек. Данная кнопка вызывает метод сохранения параметров пользователя, используя функцию библиотеки android Preferences.set() (Приложение E). Данный метод принимает два параметра по типу ключ-значение, где ключ - это имя переменной, по которому в дальнейшем функция загрузки сможет определить запрашиваемую переменную, а значение - это непосредственно значение, которое необходимо сохранить в память приложения. Данное значение должно иметь тот же тип данных, что и используется в методе set(). Если это setInteger(), то передаваемая переменная должна быть типа Integer, если setString(), то переменная должна быть типа String. Данные методы позволяю хранить различные типы данных, а также имеют ограничение лишь в объеме памяти устройства. Для реализации загрузки настроек пользователя, которая осуществляется при запуске приложения пользователем, используется соответствующий метод Prefernces.get() (Приложение F) куда передается также два параметра по типу ключ-начальное значение. Где ключ соответственно это ключ, который использовался при реализации сохранения параметров пользователя приложения, а начальное значение представляет значение, которое разработчик задает по умолчанию. Это позволяет избежать ошибки null при запросе значения, которое еще ни разу не было сохранено. Соответственно начальное значение имеет тот же тип данных, что предается в метод get(). Если это getString(), то параметр должен передаваться типа String. Данный метод, не смотря на свою ограниченность в разнообразие типов данных для хранения, может быть использован для хранения более сложных структур, таких как xml или json. Данные форматы данных могут быть конвертированы в строку, используя готовые библиотеки конвертации либо конвертеры, написанные разработчиком. Полученную в результате конвертации строку, имеющую тип данных String можно сохранить, используя метод setString(), во внутреннюю память устройства. При необходимости данную строку можно загрузить в оперативную память используя метод getString() и в последствие преобразовать с помощью конвертеров в необходимый структурный тип данных, такой как xml или json.

3.1.3 Модуль интеграции иконок

Иконки позволяют передавать информацию пользователю посредством визуального контакта с ними. Каждая иконка имеет визуальное представление какого-то обобщенного образа, хорошо известного любому человеку. Тем самым позволяя передавать необходимую информацию пользователям используя минимальные трудозатраты, а также не мало важно использование минимального пространства для передачи необходимой информации пользователю приложения. Данный метод передачи информации поддерживается разработчиками приложения во всем мире, так как данный способ удовлетворяет основным критериям, таким как информативность и доступность понимания у широких масс. Так же данный метод избегает такую проблему, как локализация, так как необходимости в переводе иконок нет, они понятны и едины на различных языках мира. Изображение солнца поймут, как в США, так и в Китае. Данная возможность позволит избежать перевода колоссального количества важной информации. Так же это позволит пользователю разбираться в новом приложение куда быстрее и при этом не уставая от чтения инструкций. (Рисунок 3.2)

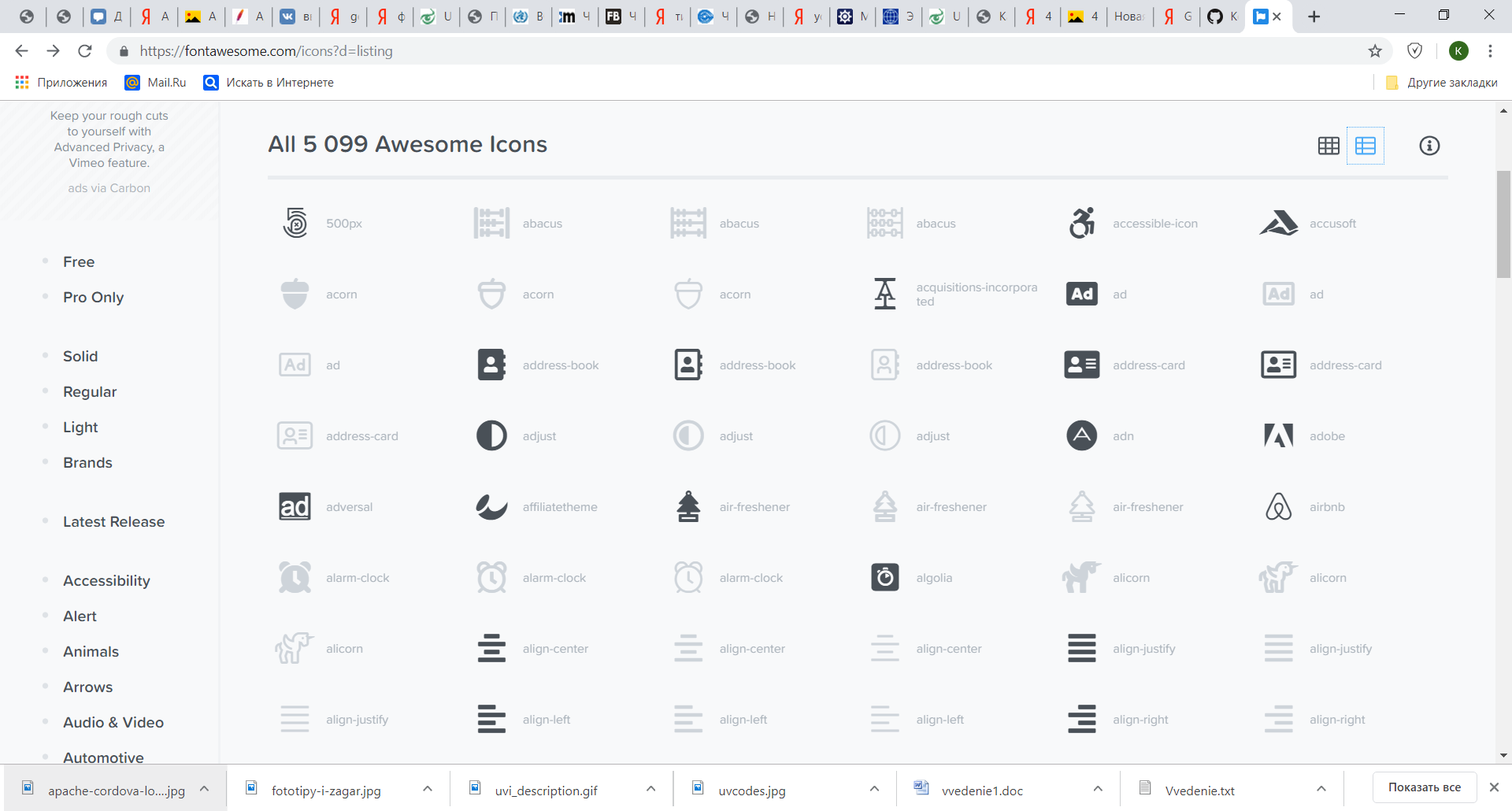


Рисунок 3.2. Иконки, доступные на сайте Font Awesome

Для реализация данного способа передачи информации в приложение используется ресурсы с сайта fontawesome.com, где имеется возможность получить в распоряжение огромное количество иконок в бесплатное пользование. На сайте имеется необходимый шрифт Solid, который устанавливается в среду разработки для дальнейшего отображения иконок в приложении пользователя. Каждая иконка в этом шрифте использует свой уникальный идентификатор или иначе уникальный код, который характеризует конкретную иконку. Данный код вставляется в текстовое преставление в приложение и при установки необходимого стиля текста, иначе называемого шрифт, отображает заданную под этим уникальным кодом иконку. Данная реализация преставление отличается от своего аналога простотой настройки и ее одноразовостью. Настройка стиля текста производится лишь раз, а далее можно неоднократно устанавливать необходимые иконки, меняя лишь уникальный идентификатор. Представление изображений с помощью растровых или векторных картинок является неэффективным аналогом иконок, где объем необходимой памяти увеличивается многократно, а также настройка каждой иконки превращается в отдельный процесс, который требует нахождения, либо создание соответствующей иконки и ее отдельной настройки, чтобы данное изображение отображалось на экране пользователя. Метод, использующий растровые или векторные изображения является устаревшим и неэффективным способом преставления информации. В данном приложение используются иконки из шрифтов, как эффективная в настройке и мало затратная в объеме ресурсов памяти альтернатива. Так же не мало важным плюсом иконок – это возможность их настройки как текста. Если имеется необходимость изменения цвета изображения иконки с одного на другой, то это выполняется достаточно просто, сменив цвет текста, что касательно растрового или векторного изображения, такая возможность минимальна. Имеется возможность изменить оттенок картинки, однако поменять цвет не имеется возможным. Необходимо отдельно создавать свое изображения, используя сторонние редакторы, такие как Photoshop, что занимает не мало времени, особенно на этапе дизайна, когда цвет выбирается и меняется десятки, если не более раз. Данный способ становится совершенно неприемлемым. Эффективным решением этих проблем как раз является использование иконок, как простого инструмента создание элементарных изображений для навигации, которые легко настраиваются и также легко интегрируются в среду. Коллекция иконок достаточно велика и практически на любую тему, какое бы приложение не разрабатывалось, будь это приложения для отеля или для автосалона, необходимые иконки найдутся по любой тематике, в том числе и для приложения по преставлению информации об ультрафиолетовом индексе в областных центрах республики Беларусь.

3.1.4 Модуль определения пространственных координат

Для определения местоположения в пространстве используются системы позиционирования. В мобильных устройствах предусмотрено два основных и один дополнительный способ получения пространственных координат. Эти системы позволяют определять местоположения пользователя устройства с различной степенью точности от города, в котором находится пользователь до конкретных координат с погрешностью в несколько метров. Используя данную систему в приложениях, имеется возможность предоставлять конечному пользователю более актуальную информацию, основанную на местоположение устройства. Таким образом, пользователь будет иметь в распоряжение данные, которые относятся непосредственно к конкретной ситуации пользователя. Актуальная информация представляет большую ценность для конечного пользователя. Поэтому возможность получения данных, соответствующих конкретной точке в пространстве является одной из ключевых возможностей в современной разработке мобильных приложений.

Одним из основных способов получения пространственных координат является GPS – система глобального позиционирования. Данная система позволяет получать местоположение пользователя, используя приемник устройства. На данный момент современные смартфоны оборудованы таким приемником, позволяющим использовать GPS. Данный способ позволяет получить местоположение устройства, не используя для данной задачи интернет, однако время получения пространственных координат устройства данным методом бывает достаточно велико, из-за отсутствия достаточного количества спутников на орбите. Данный способ позволяет определить пространственные координаты достаточно точно, так же данный метод может быть использован на удаление от населенных пунктов. Однако данный метод невозможно использовать в местах, куда не проникает сигнал со спутников. Таким образом, пользователь приложения не сможет получить пространственные координаты, если устройство будет находиться на крытой парковке, так как не будет иметь доступ к сигналам, которые посылают спутники.

Вторым основным методом является получения координат, используя сотовые вышки связи, а также источники сигнала WIFI. Данный метод позволяет использовать сотовые вышки, находящиеся поблизости, а также информацию об местоположение данных вышек в пространстве, что позволяет получить расстояние между ближайшими точками WIFI и сотовых вышек относительно устройства. А далее вычислить местоположение устройства относительно полученных координат источников сигнала и их расстояния до устройства пользователя. Точность конечного результата зависит от количества имеющихся источников. Также источником сигнала может служить другое устройство по соседству, которое имеет информацию о своих пространственных координатах. Данный способ совершенно не зависит от GPS, однако его точность достаточно нестабильна. Данный метод может работать, если поблизости имеются сотовые вышки, либо другой источник сигнала, имеющий информацию о координатах в данной точке пространства. Это означает, что пользователь, находясь в отдаление от источников сигнала, не сможет получить информацию о местоположение устройства. Данный метод затрудняет получение информации в отдаление от городов и деревень.

Данный метод позволяет получать пространственные координаты, опираясь на устройства, находящиеся поблизости и запрашивающие пространственные координаты первым или вторым методом. Приложение с помощью антенны устройства прослушивает пространство на наличие устройств, которые совершают запросы по определению пространственных координат. Информация, которая перехватывается приложением, позволяет определить пространственные координаты устройства. Данный метод не сможет работать без находящегося рядом устройства, способного получить информацию о пространственных координатах с использованием либо первого, либо второго способа.

Местоположение для пользователя мобильного устройства, а также пользователя мобильных приложений является крайне важным моментом, так как точность определения местоположения оказывает непосредственное влияние на точность полученной информации в конкретной местности, где находится пользователь. Не точная информация пользователю может только навредить, так как информация от места к месту может сильно отличаться и тем самым может повлиять на пользователя. Мобильные устройства позволяют получить координаты достаточно точно, чтобы определить со сто процентной вероятностью город, в котором находится пользователь. Данное приложение предусматривает возможность информировать пользователя об показатели ультрафиолетового индекса в областных центрах республики Беларусь. Для выполнения поставленной задачи достаточно информации, позволяющей определить конкретный город, в котором находится пользователь. Это реализуется посредством сопоставления координат пользователя, которые возвращает приложению устройство, с координатами областных центров республики Беларусь.

Для реализации модуля получения пространственных координат выбран второй метод, основанный на получение координат с использованием сигналов от сотовых вышек и других источников сигнала, таких как WIFI. Данный метод был выбран, так как актуальный показатель ультрафиолетового индекса получить без доступа к интернету в данном приложение не имеется возможны. В связи с тем, что данные получаются приложением посредством парсинга сайта, получение координат устройства в данном приложение без доступа к интернету не имеет необходимости. Так же данный метод позволяет получить информацию о местоположение пользователя за малый промежуток времени. Точность в данном случае так же не является главным критерием, так как необходимо определить положение пользователя относительно крупных городов, а значит точности второго метода достаточно, чтобы выполнить поставленную задачу для данного модуля.

Реализация данного метода осуществляется посредством выдачи приложению разрешений на использование возможностей устройства, связанных с получением пространственных координат. Данный способ реализуется посредством добавления соответствующего разрешение в файл Manifest (Приложение A). В данный файл необходимо дописать <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION"/>. Данная строка означает, что приложению выдаются разрешения на доступ ко всем методам получения пространственных координат, в том числе и метод, основанный на прослушивание источников сигнала. После получения доступа создается LocationManager – пространственный менеджер, который содержит информации о пространственных координатах и основывается он на методе запроса – requestLocationUpdates. Данный метод принимает на вход три парамтера: тип запроса NETWORK\_PROVIDER, который обозначает, что для данного запроса используется второй метод получения пространственных координат через провайдера. Минимальное время запроса, который означает, через какое минимальное время будет возвращен результат запроса слушателя. И последний параметр, который принимает метод запроса – это точность, которая определяет точность в метрах, где минимальное значение порядка 10 метров.

3.1.5 Модуль навигации в приложение вкладки

Для быстрого перемещения между страницами приложения необходимо создать удобный интерфейс навигации. Для реализации удобного интерфейса навигации, используется элемент «Вкладки». Который предоставляет пользователю возможность, в любой момент пользования приложением, переместиться на интересующею его страницу, посредством нажатия соответствующей вкладки на панели вкладок. Панель вкладок расположена в нижней части экрана устройства пользователя. В ней имеются вкладки для перехода на все имеющиеся в доступности у пользователя страницы. Для перехода на интересующую пользователя страницу достаточно нажать на соответствующую вкладку на панели вкладок. После перехода пользователя на другую страницу, панель вкладок остается в поле видимости пользователя и не меняет своего положения, по-прежнему находясь в нижней части экрана пользователя. Для демонстрации пользователю, на какой странице он находится в данный момент, используется подсвечивание соответствующей вкладки уникальным цветом, в тот же самый момент остальные вкладки имеют нейтральный цвет, означающий их неактивность. Вкладки не имеют текстового описания, однако они имеют изображения, которые описывают содержимое посредством элементарных иконок. Иконки позволяют определить содержимое, не имея, какое-либо текстовое описание. А также позволяют более опытным пользователям данного приложения ориентироваться в приложение куда более эффективно, не вчитываясь в текстовое описание вкладок. Так же местоположение и последовательность вкладок не меняет свое положение не зависимо от обстоятельств. В данном приложение нет необходимости вводить обязательное текстовое описание вкладок, так как любое действие пользователя не несет вреда приложению, что означает, какие бы действие не предпринял пользователь в самом приложение, они не повлияют на стабильность работы данного приложения.

3.1.6 Модуль парсинга страниц

Модуль парсинга страниц предназначен для получения информации с сайта, а также для последующей ее обработки. Данный модуль подразделяется на две основные функции. Первая функция заключается в открытие потока между приложением на устройстве пользователя и сайтом, который является целью для парсинга. Открытый поток позволяет получать данные с сайта, в данном случае вся страница сайта будет забираться в текстовом формате для дальнейшего анализа. Как только страница сайта в текстовом формате полностью прочтена и записана в память устройства, связь с сайтом прерывается функцией прерывания. Данные поступают во вторую функцию для обработки поступивших данных. Поступившие данные приставляют из себя код страницы в текстовом формате. В этих данных, среди всей информации имеются необходимые показатели индекса ультрафиолета. А также имеются показатели прогноза ультрафиолетового индекса. Для получения этих показателей в чистом виде, необходимо определить маркеры, на которые можно ориентироваться с целью определения местоположения необходимых значений и их последующего извлечения. Чтобы получить такой маркер, необходимо рассмотреть имеющийся код страницы и выделить уникальные элементы вблизи необходимых показателей. При анализе данного кода легко выявляется уникальный элемент "color" за которым следует код цвета, и дальше следует показатель ультрафиолетового индекса (Рисунок 3.3).

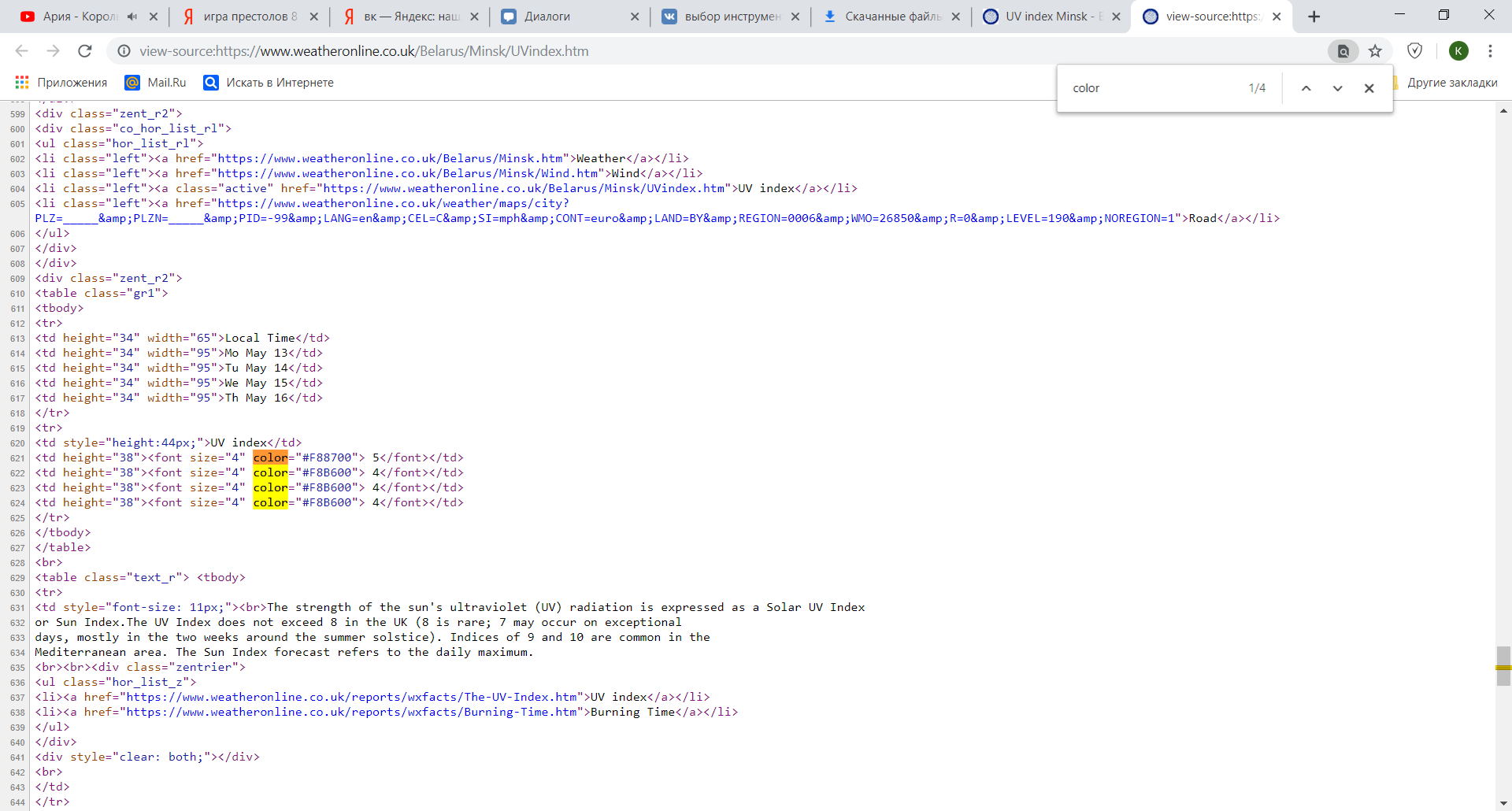


Рисунок 3.3. часть кода страницы с необходимыми показателями УФ индекса

Далее необходимо проверить, имеются ли совпадения с маркером в данном коде страницы. При поиске совпадений маркера по странице не было выявлено ни одного совпадения, кроме других показателей ультрафиолетового индекса. Таким образом, данный маркер максимально точно указывает на необходимые значения ультрафиолетового индекса.

3.1.7 Модуль расчёта максимального времени пребывания под воздействием ультрафиолетового излучения

Модуль расчёта максимального времени пребывания на открытом солнце без защиты от ультрафиолетового излучения позволяет определить время безопасного нахождения, на открытом солнце, используя показатели ультрафиолетового индекса, а также типа кожа, который выбрал пользователь приложения. Время, которое представляется пользователю, ориентировано на максимальное значение показателя ультрафиолетового индекса за сутки, определимого в промежуток времени между двенадцатью часами и одним часом после полудня. Значение безопасного времени нахождения на солнце несет рекомендательный характер. Так как данное значение зависит от множества факторов, в том числе от личных физических особенностей пользователя. Для достижения идеального правильного значения необходимо произвести индивидуальное медицинское обследование, а также опытным путем выявить значение минимальной эритемной дозы, только при таких условия имеется возможность расчёта более точного значения времени безопасного нахождения под воздействием ультрафиолетового излучения без защиты. Так как данная процедура является дорогостоящей, относительно результат, для данного приложения при расчётах не учитывается какая-либо защита кожи, а также берется максимальное значение показателя ультрафиолетового индекса в течение дня. Если не превышать данное время в течение суток, то вероятность неблагоприятных последствий снижается до минимального значения. Данные рекомендации так же не являются универсальными, так как в зависимости от физических особенностей пользователя они могут быть ошибочны. К таким особенностям относятся шрамы, папилломы, большие родимые пятна и другие физические особенности кожи.

3.2. Разработка алгоритмов модулей и их реализация

3.2.1 Описание процесса реализации модулей приложения

В результате процесса проектирования приложения были сформулированы текстовые описания работы пяти модулей, два из которых – модуль иконок и модуль влкадок, относятся к категориии дизайнерских решений, а так же четыре модуля, технически-програмного направления. Спроектированные модули необходимо реализовать, однако граммотным шагом будет составить блок-схемы модулей, что позволит выявить логические нестыковки в процессе их работы и тем самым избежать их в исходном коде. А когда блок-схемы буду реализованы, появится возможость без тривиальных ошибок перенести алгоритмы в исходный код.

3.2.2 Алгоритм модуля определения пространственных координат.

Первым модулем для моделрования блок-схемы полсужит модуль получения пространственных координат. Согласно описанию данный модуль получает данные посредством связи через интерент, в частности интернет провайдера (Рисунок 3.4).

Данная блок схема наглядно демонстрирует процесс получения пространственных координат. Перед началом работы модуля пространственных координат, где происходит проверка на наличие доступа к интернету. Это позволяет избежать ошибок в работе приложения, в случае отсутствия доступа к интернету.

Так же в момент работы модуля происходит проверка на наличие достаточного количества источников сигнала для определения пространственных координат пользователя. Если количество доступных сигналов недостаточно, то идет повторная проверка доступа к интернету, до тех пор, пока приложение не будет экстренно закрыто, либо будет найдено достаточное количество источников сигнала (Приложение D).

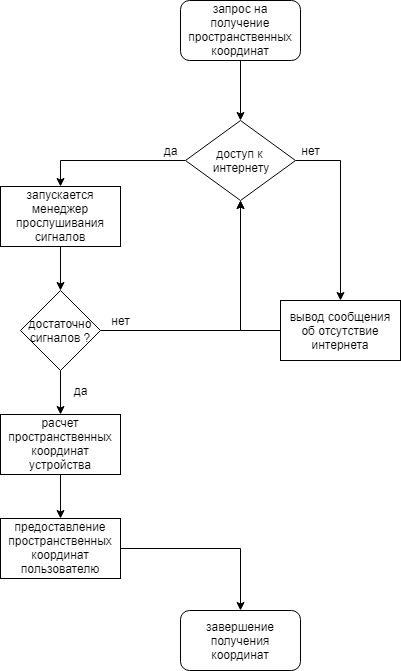


Рисунок 3.4 Блок-схема модуля определения пространственных координат

Далее полученные координаты сверяются с имеющимся набором точек, и выбирается точка, максимально приближенная к полученным координатам. Для этого используется метод DefineLocation() (Приложение C).

3.2.3 Алгоритм модуля парсинга страницы

Модуль парсинга страниц является ключевым функциональным элементом в приложение. Без получения данных, другие модули не буду иметь возможности выполнить поставленные перед ними задачи. Блок-схема данного модуля представлена ниже (Рисунок 3.5)

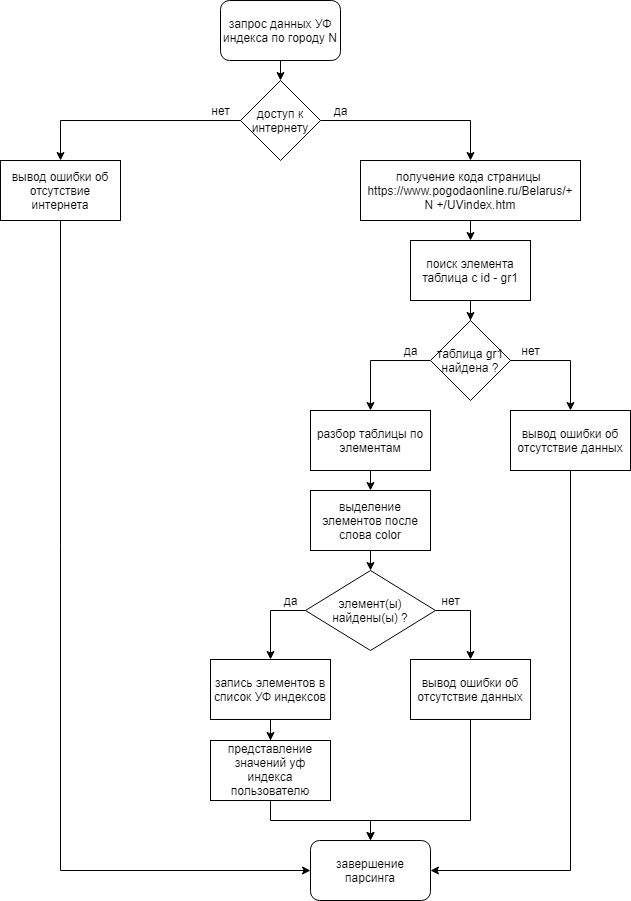


Рисунок 3.5 Блок-схема модуля парсинга страниц

Алгоритм данного модуля в отличие от остальных включает в себя несколько последовательных углублений в процессе решения поставленной задачи (Приложение B). В данном модуле, как и в модуле получения пространственных координат, перед началом работы происходит проверка на наличие доступа к интернету. После положительного результата проверки доступности интернета, происходит несколько ключевых событий. Первое событие – это получение кода страницы по указанному адресу в тестовом формате. В дальнейшем полученный код используется для поиска и получения необходимой информации. Получение информации происходит в два этапа. На первом этапе происходит поиск таблицы, в которой имеются данные, а во время второго этапа происходит поиск показателей ультрафиолетового индекса по найденной таблице. Результатом поиска является четыре числа. Первое это показатель ультрафиолетового индекса за сегодняшний день. А остальные числа являются прогнозируемыми показателями ультрафиолетового индекса на ближайшие трое суток.

3.2.4 Алгоритм модуля расчёта максимального времени пребывания под воздействием ультрафиолетового излучения

Данный модуль был спроектирован для того, чтобы предоставить пользователю возможность рассчитать максимальное время пребывания под воздействием ультрафиолетового излучения. В данной блок-схеме демонстрируется возможности данного модуля (Рисунок 3.6).

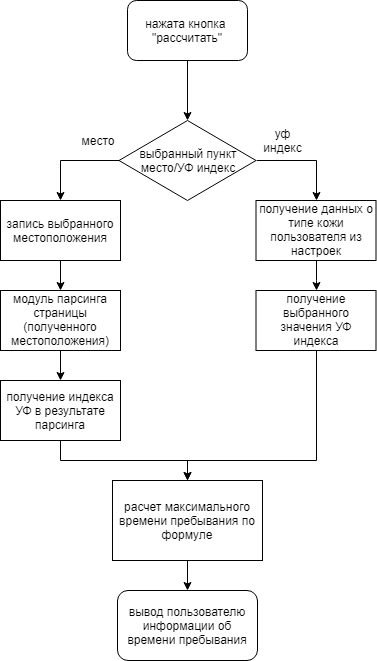
****

Рисунок 3.6 Блок-схема модуля расчёта максимального времени пребывания под воздействием УФ излучения

Данный модуль достаточно многозадачный. Он позволяет рассчитывать максимальное время пребывания как для конкретной местности, где находится пользователь, так и для любой местности, имеющиеся в списке доступных городов. Так же данный модуль позволяет рассчитать максимальное время пребывания и для любого из, предложенного в списке, показателя ультрафиолетового индекса.

Основной проверкой в алгоритме является предварительная проверка выбранного пункта. Если выбран пункт «местность», то расчёты буду вестись согласно выбранному пользователем населённого пункта. Если будет выбран пункт «УФ индекс», то на выбор пользователю будут даны показатели ультрафиолетового индекса, согласно которым будут производиться расчёты. Далее в обоих случаях из настроек принимается значение типа кожи пользователя, которое также участвует в окончательных расчетах времени пребывания под воздействием ультрафиолетового индекса. После расчётов пользователю демонтируется максимальное время пребывания под воздействием ультрафиолетового излучения.

3.3 Описание работы пользователя с приложением

3.3.1 Главная страница приложения

После запуска приложения пользователю демонстрируется главная страница приложения с информацией об ультрафиолетовом индексе в месте нахождения устройства. Каждая страница является результатом xml разметки (Приложение G). Снизу доступна панель навигации по приложению, реализованная в виде вкладок. Каждый элемент вкладки ведет на соответствующую страницу. На главной странице имеется лишь один функциональный элемент, и он должен показывать значение ультрафиолетового индекса (Рисунок 3.7).

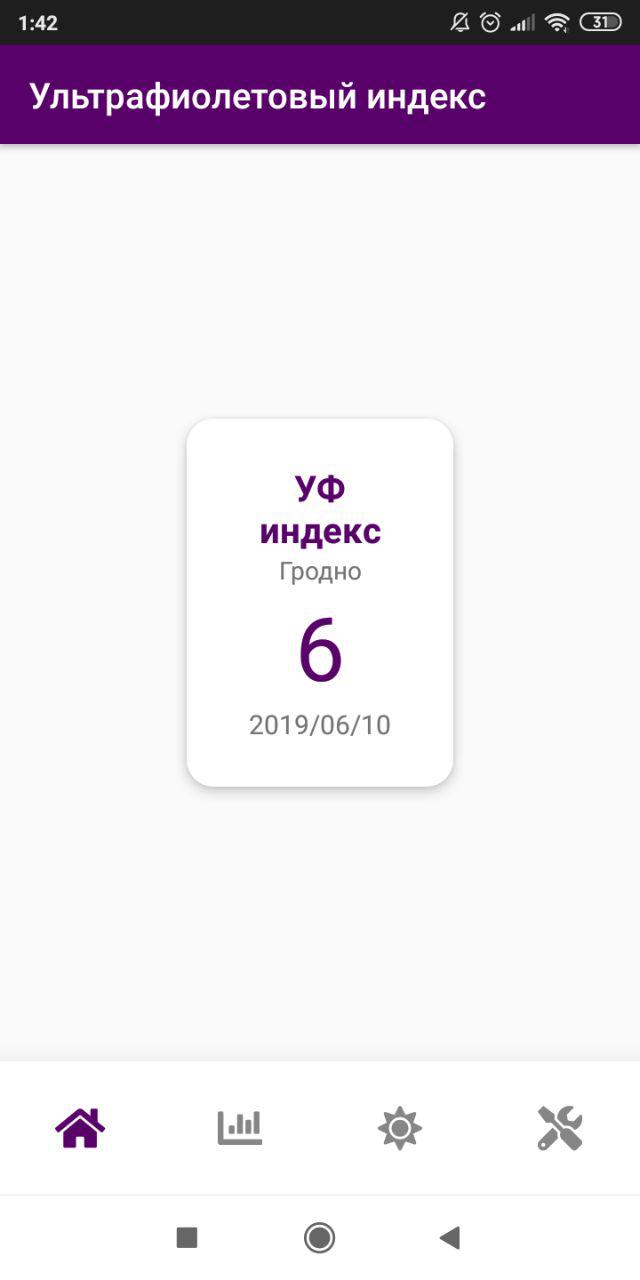


Рисунок 3.7. Главная страница приложения

О том, что пользователь находится на главной странице, оповещает подсвеченный элемент вкладки домашней страницы на нижней панели вкладок. Также на главной странице пользователю демонстрируется сегодняшняя дата, а также город, к которому относится ультрафиолетовый индекс. Также данный город соответствует городу, выбранному на панели настроек пользователя, либо самим пользователем, либо параметр по умолчанию.

3.3.2 Страница приложения с прогнозом ультрафиолетового индекса

На второй странице пользователю предоставляется прогноз показателя ультрафиолетового индекса, согласно выбранному местоположению, по умолчанию город Минск (Рисунок 3.8)

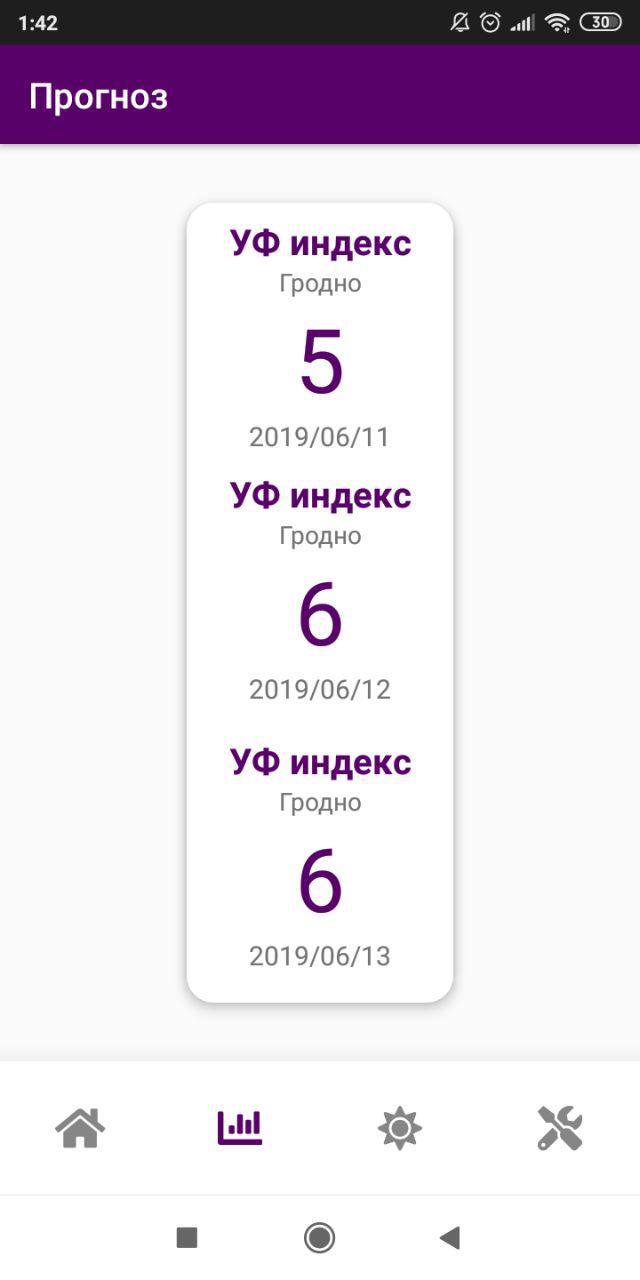


Рисунок 3.8. Страница приложения с прогнозами

Таким образом, пользователю предоставляется прогноз ультрафиолетового индекса на ближайшие трое суток. Согласно данной информации пользователь сможет делать самостоятельные выводы, опираясь на показатели индекса ультрафиолетового излучения в ближайшие дни. На странице прогноза пользователю демонстрируется прогноз ультрафиолетового индекса того же города, что и показатель ультрафиолетового индекса на главной странице приложения.

3.3.3 Страница с калькулятором

При переходе на третью вкладку пользователю становится доступна страница с калькулятором, позволяющем рассчитать максимальное время пребывания под воздействием ультрафиолетового излучения (Рисунок 3.9).

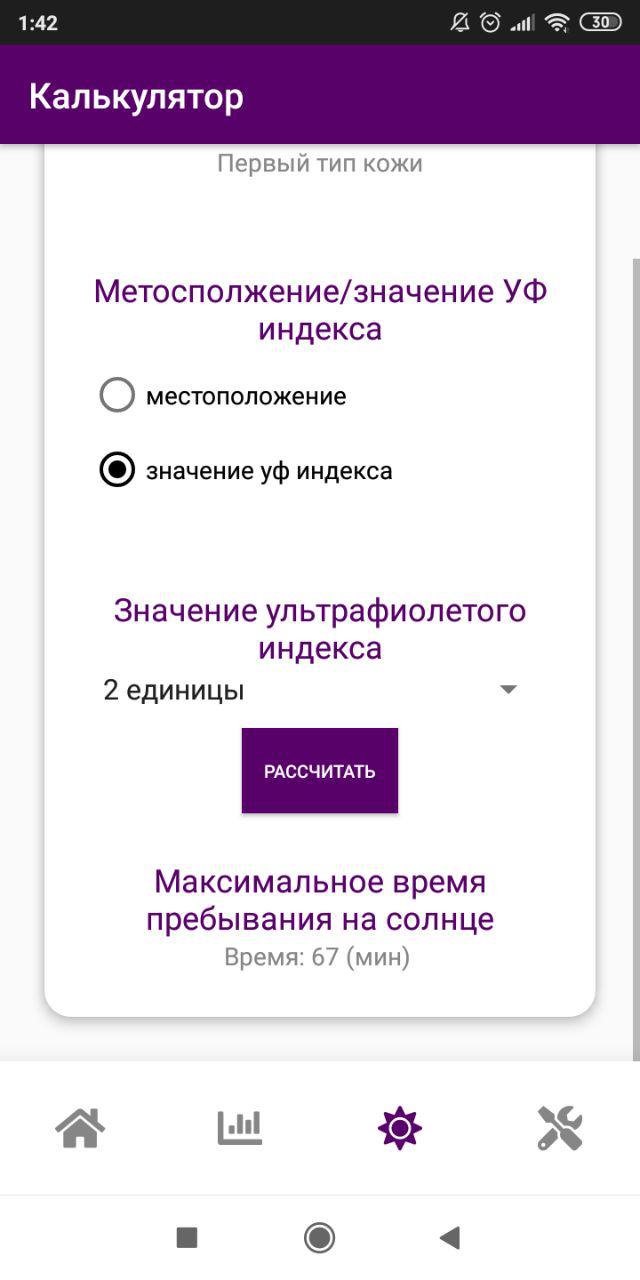


Рисунок 3.9. Страница приложения с калькулятором загара

Данная страница позволяет пользователю совершать расчёты по местоположению пользователя. Также на данной странице приложения, пользователь имеет возможность производить расчёты индекса в населенных пунктах, представленных в выпадающем списке местоположений. При необходимости можно производить расчёты без привязывания к местоположению пользователя, используя собственные значения ультрафиолетового индекса, таким образом, пользователь может производить расчёты со своими личными настройками. Пользователь может производить расчёты в двух режимах: по местоположению и показателю ультрафиолетового индекса.

3.3.4 Страница настроек пользователя

При переходе на четвертую вкладку пользователю становится доступна страница с настройками, позволяющая установить тип кожи пользователя, а также выбрать местоположение самому, либо определить местоположение автоматически посредством нажатия соответствующей кнопки (Рисунок 3.10).

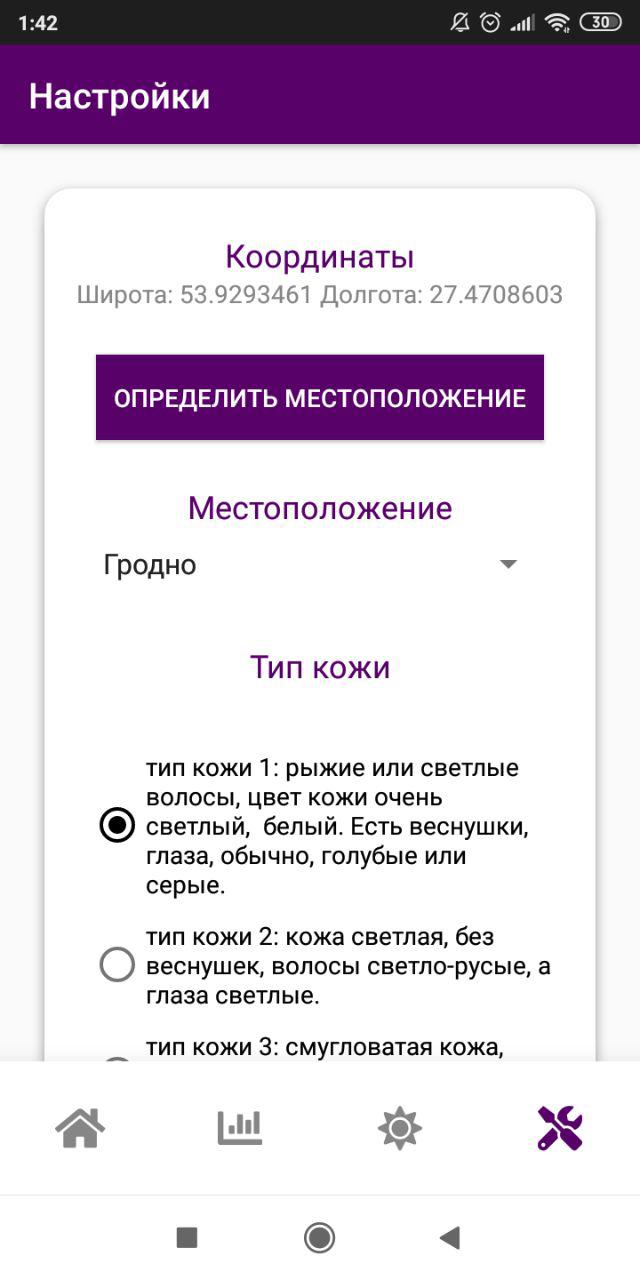


Рисунок 3.10. Определение местоположения пользователя

Также пользователю на странице настроек доступна возможность выбрать тип кожи. Данный параметр используется пользователем при проведении расчётов на странице калькулятора. Данный параметр определяется пользователем и автоматически при первом старте приложения определяется как первый тип кожи. При необходимости пользователь имеет возможность в любой момент изменить данный параметр, выбрав необходимый из списка предложенных вариантов. Также на данной странице имеется кнопка сохранения настроек. Обновление параметров приложения происходит только при нажатии кнопки сохранения настроек пользователя, иначе изменения не вступают в силу и у пользователя остаются предыдущие настройки. Если пользователь нажимает кнопку сохранения настроек пользователя, то приложение обновляет состояние на страницах, таким образом, при изменении местоположения и сохранение новых параметров, на главной странице отобразятся данные соответствующие новым настройкам пользователя (Рисунок 3.11).

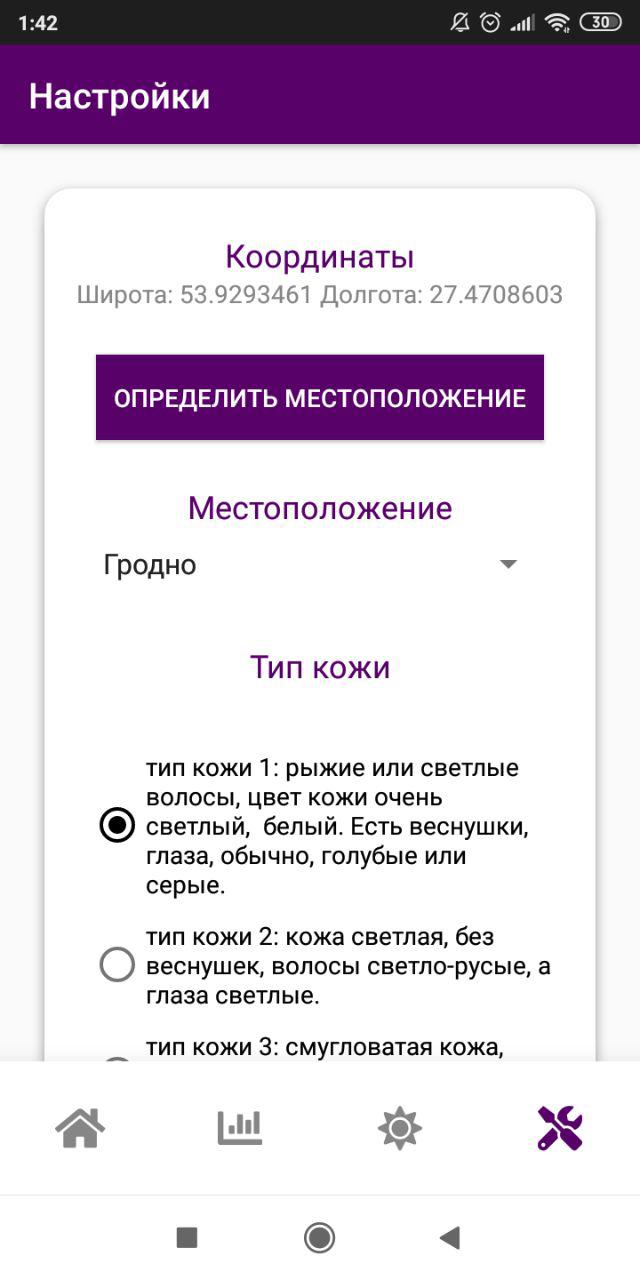
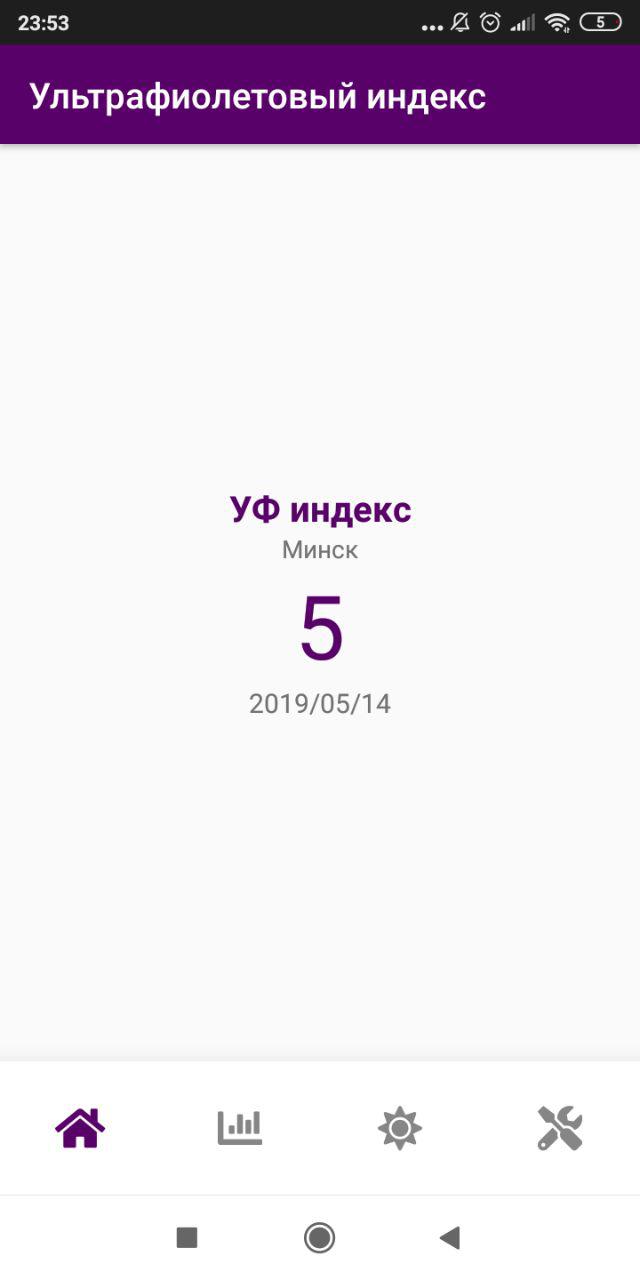


Рисунок 3.11. Страница приложения с настройками

3.3.5 Переход между страницами приложения

Переход между страницами приложения осуществляется посредством нажатия соответствующих вкладок страниц на панели вкладок (Рисунок 3.12).



3

1

2

4

Рисунок 3.12. Панель вкладок приложения 1–домашняя страница, 2 – страница с прогнозами 3 – страница с калькулятором, 4 – страница настроек

Данная панель позволят переходить по страницам приложения, при этом вкладка страницы, на которой находится пользователь, выделяется соответствующим цветом, в то время как остальные вкладки светло-серого цвета, что означает их неактивность в данный момент времени. Также на каждой вкладке изображена иконка, сообщающая о том, что находится на предполагаемой странице, таким образом, иконка дома сообщает пользователю, что это домашняя страница, а иконка инструментов на четвертой вкладке о том, что там страница настроек приложения. Данный элемент навигации позволяет быстро перемещаться между функциональными элементами приложения. При этом время, которое тратит пользователь на переход между страницами сводиться к минимуму, а значит, задача удобства навигации считается выполненной в полной мере.

3.3.6 Аппаратно-технические требования приложения

Данное приложение было протестировано на устройствах с операционной системой Android Studio 5.0 и Android Studio 8.0. В тестирование учувствовали такие модели устройств как: Prestigio Muze A7 и Xiaomi Readmi 6. В результате тестов критических ошибок выявлено не было. Таким образом, можно заключить, что для данных устройств работа приложения стабильна. Самой максимально требовательной библиотекой является jsonup API 16 и выше. Таким образом, для устройств с классической операционной системой Android 4.1 и выше (не изменённой) работоспособность приложения гарантирована. Однако операционная система может быть модифицированной и тогда тестирование необходимо проводить индивидуально, так как спецификация операционной системы может сильно отличаться от оригинала. Данное приложение может быть использовано большей частью населения республики Беларусь (Рисунок 3.13) [9].

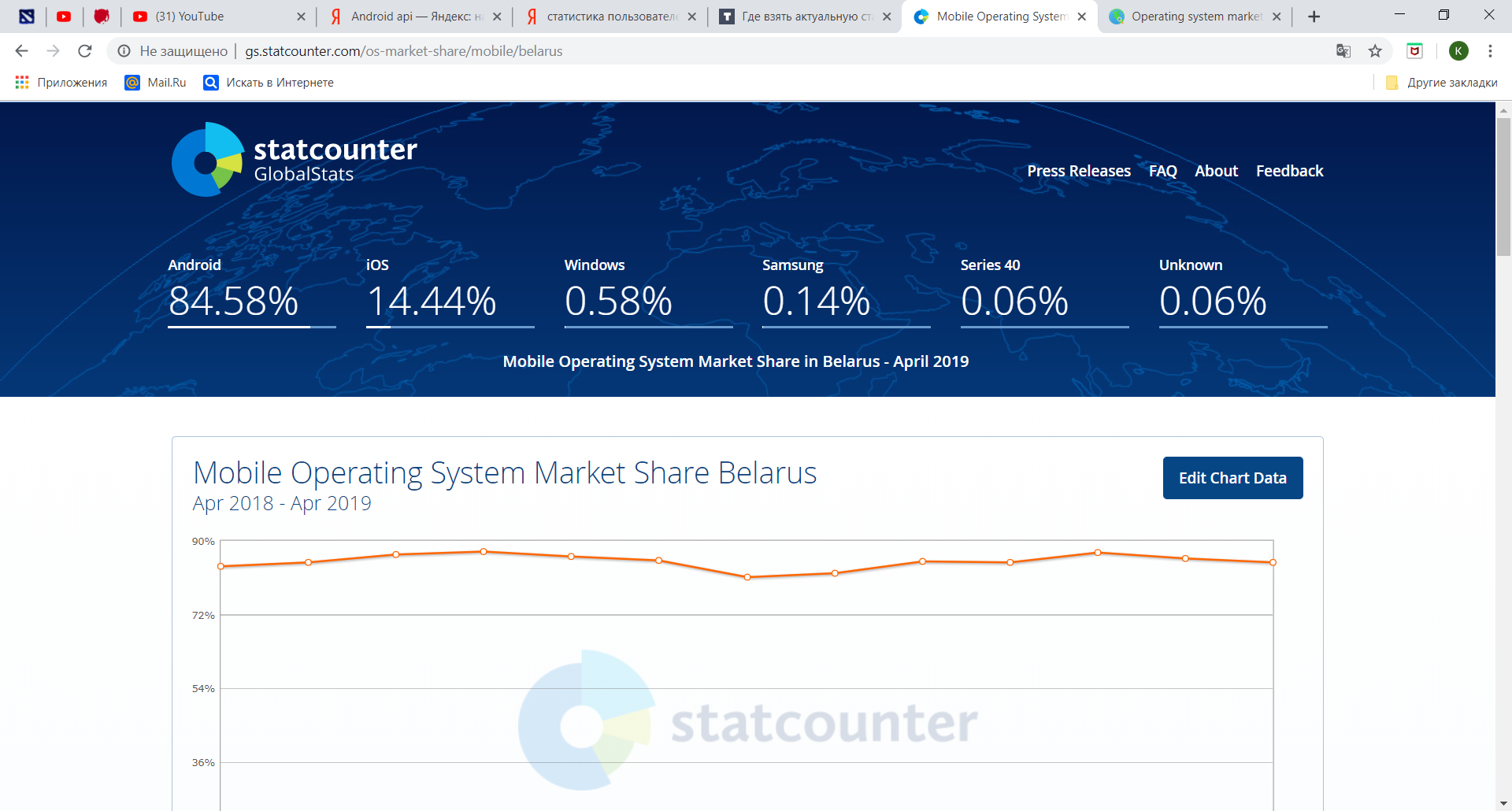


Рисунок 3.13 Статистика мобильных операционных систем в республики Беларусь за апрель 2019 года

Для того, чтобы приложение было успешно установлено, необходимо чтобы устройство имело операционную систему Android. Так же устройство должно иметь возможность получить доступ к сети WIFI либо 3G или 4G. Так же для установки данного приложения требуется место для самого apk-файла 1.8 МБ. Данный формат файла используется для установки приложений на устройства с операционной системой Android. Также необходимо место для самого приложения, который использует до 2.5 МБ цифрового пространства. На рисунке представлено окно свойств данного приложения на устройстве (Рисунок 3.14).

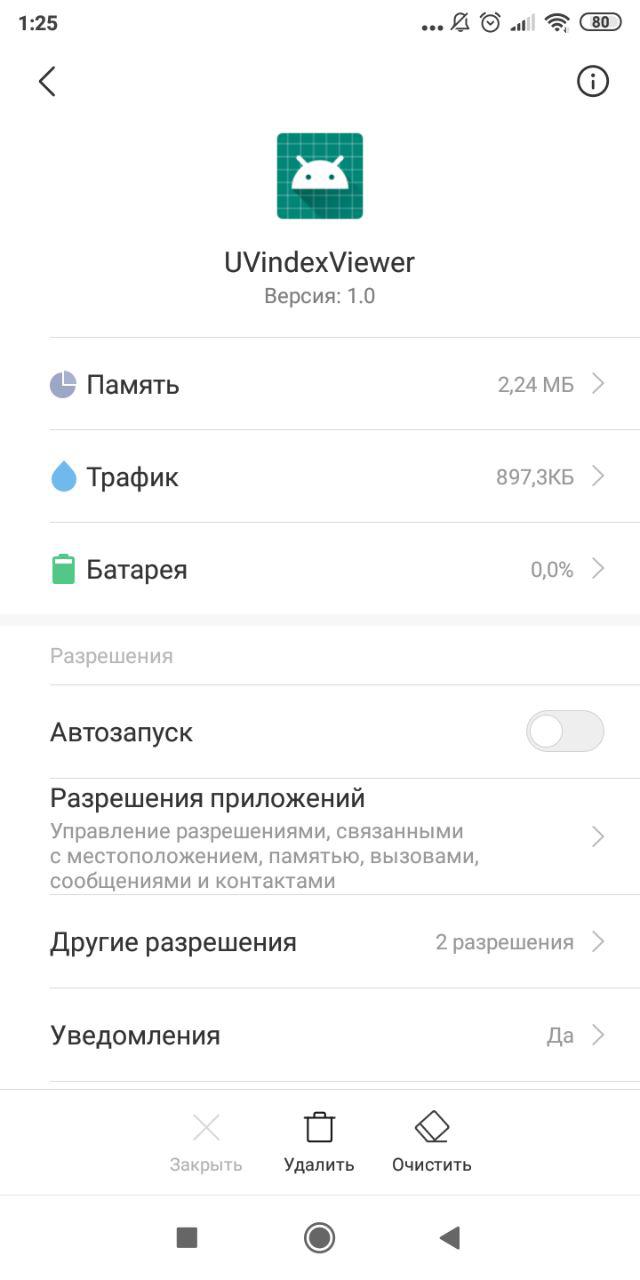


Рисунок 3.14 Окно свойств приложения на устройстве

Количество занимаемой памяти равно 2.24 МБ, однако это значение может отличаться в зависимости от версии операционной системы Android.

ЗАКЛЮЧЕНИЕ

В результате проделанной работы было разработано приложение, позволяющее информировать пользователей о значение индекса ультрафиолетового излучения в областных центрах республики Беларусь. Данное приложение позволяет определять местоположение пользователя и тем самым предоставлять актуальную информацию. Информация по ультрафиолетовому индексу предоставляется в соответствие с местоположением пользователя. Так же данное приложение позволяет определить максимально рекомендованное время нахождения под воздействием солнечного ультрафиолетового излучения без дополнительной защиты кожи.

Приложение было протестировано на физических устройствах с операционной системой Android 5.1 Lollipop и Android 8.0 Oreo. В результате тестирования не было выявлено критических ошибок для приложения. Производилось тестирование полного функционала приложения. Был протестирован модуль парсинга страниц, результатом которого было возвращение запроса с данными, которые соответствовали данным на сайте. Так же был проверен модуль сохранения настроек, все настройки были успешно сохранены в память устройства и загружены в приложение.

Также в процессе работы были получены навыки проектирования и разработки в среде Android Studio. Закреплены навыки работы с эмуляторами виртуальных устройств, таких как AVD – Android Virtual Device, а также сторонним приложением Genymotion. Изучен объектно-ориентированный язык программирования Java. Так же в процессе работы использовалась система контроля версий, с помощью которой процесс разработки приложения находился под полным контролем.

Конечным продуктом является удобное приложение, позволяющее узнать актуальную информацию об индексе ультрафиолетового излучения в областных центрах республики Беларусь.

СПИСОК ЛИТЕРАТУРЫ

1. Ультрафиолетовое излучение // Физическая энциклопедия / Большая Российская энциклопедия том пятый, 760 с, Гл. ред. А. М. Прохоров, 1998.
2. Сайт всемирной организации здравоохранения [электронный ресурс]

Режим доступа:

<https://www.who.int/uv/intersunprogramme/activities/uv_index/ru/>

Дата доступа – 24.03.2019

1. uvi.terrameteo.ru[электронный ресурс] – Режим доступа:

<http://www.uvi.terrameteo.ru/uvi_description.php>

Дата доступа – 27 03.2019

1. ozone.bsu.by[электронный ресурс] – Режим доступа:

<https://ozone.bsu.by/uv_brochure.html>

Дата доступа – 15 04 2019

1. zozhlife.ru[электронный ресурс] – Режим доступа:

<http://zozhlife.ru/profilaktika/kak-vybrat-solntsezashhitnyj-krem.html>

Дата доступа – 24 04 2019

1. uvinfo.bsmu.by[электронный ресурс] – Режим доступа:

<http://uvinfo.bsmu.by/med.html>

Дата доступа – 28 04 2019

1. cordova.apache.org[электронный ресурс] – Режим доступа:

<https://cordova.apache.org/docs/ru/latest/guide/overview/index.html>

Дата доступа – 06 05 2019

1. visualstudio.microsoft.com[электронный ресурс] – Режим доступа:

<https://visualstudio.microsoft.com/ru/xamarin/>

Дата доступа – 10 05 2019

1. [http://gs.statcounter.com](http://gs.statcounter.com/) [электронный ресурс] – Режим доступа:

<http://gs.statcounter.com/os-market-share/mobile/belarus>

Дата доступа – 20 05 2019

ПРИЛОЖЕНИЕ

Приложение A

//Manifest

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.user.uvindexviewer">  
  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/AppTheme">  
 <activity android:name=".TabsActivity"  
 android:screenOrientation="portrait">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 <activity  
 android:name=".ForecastActivity"  
 android:label="Прогноз"  
 android:theme="@style/ForecastTheme"  
 android:screenOrientation="portrait"/>  
 <activity  
 android:name=".HomeActivity"  
 android:label="Ультрафиолетовый индекс"  
 android:theme="@style/HomeTheme"  
 android:screenOrientation="portrait"/>  
 <activity  
 android:name=".SettingsActivity"  
 android:label="Настройки"  
 android:theme="@style/SettingsTheme"  
 android:screenOrientation="portrait"></activity>  
 <activity  
 android:name=".TanActivity"  
 android:label="Калькулятор"  
 android:theme="@style/TanTheme"  
 android:screenOrientation="portrait"></activity>  
 </application>  
 <uses-permission android:name="android.permission.INTERNET"/>  
 <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION"/>  
 <uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION"/>  
  
</manifest>

Приложение B

//Parsing.java парсинг страницы

class Parsing extends AsyncTask<String,String,String >  
{  
 public TextView info;  
 public String cityUrl;  
 public String list="";  
 public String[] uvIndexValue = new String[4];  
 @Override  
 protected String doInBackground(String... strings) {  
 try  
 {  
 //info.append("try2");  
 Document doc = Jsoup.*connect*("https://www.weatheronline.co.uk/Belarus/"+cityUrl+"/UVindex.htm").get();  
 //info.append("try3");  
 Element table = doc.select("table[class=gr1]").first();  
  
 Iterator<Element> ite = table.select("td[height=38]").iterator();  
 int i = 0;  
 while(ite.hasNext())  
 {  
 Element tablerow = ite.next();  
 Elements data = tablerow.select("td");  
 for (Element link : data) {  
 uvIndexValue[i]=link.text();  
 }  
 i++;  
 }  
  
  
  
 }  
 catch (Exception e)  
 {  
 Log.*e*("PARSING",e.toString());  
 }  
 return null;  
 }  
  
 @Override  
 protected void onPostExecute(String s) {  
 info.setText(uvIndexValue[0]);  
 super.onPostExecute(s);  
 }  
}

Приложение С

//Метод определения местоположения

public void DefineLocate()  
{  
 Double minValue=1000.0;  
 Integer minId=0;  
 Double testValue;  
 for(int i = 0; i<citiesLat.length; i++)  
 {  
 testValue = Math.*sqrt*(Math.*pow*(citiesLat[i]-latPostion,2)+ Math.*pow*(citiesLon[i]-lonPostion,2));  
 locationText.append("\n");  
 locationText.append("Citie: "+cities[i]+" distance: "+testValue);  
 if(testValue<minValue)  
 {  
 minValue=testValue;  
 minId=i;  
 }  
 }  
 spinner.setSelection(minId);  
 SaveSettings();  
}

Приложение D

private LocationListener listener = new LocationListener() {  
  
 @Override  
 public void onLocationChanged(Location location) {  
 Log.*i*("MYLOG","locLoad");  
 if (location!=null) {  
 latPostion = Double.*valueOf*(location.getLatitude()+"");  
 lonPostion = Double.*valueOf*(location.getLongitude()+"");  
 locationText.setText("Широта: " +String.*valueOf*(location.getLatitude())+" Долгота: "+String.*valueOf*(location.getLongitude()));  
 }  
 else{  
 locationText.setText("Координаты не получены");  
 }  
 }  
 @Override  
 public void onStatusChanged(String provider, int status, Bundle extras) {  
 }  
 @Override  
 public void onProviderEnabled(String provider) {  
 }  
 @Override  
 public void onProviderDisabled(String provider) {  
 }  
};

Приложение E

//Метод сохранения настроек пользователя

public void SaveSettings()  
{  
 SharedPreferences sPref = getSharedPreferences("Settings",Context.*MODE\_PRIVATE*);  
 SharedPreferences.Editor myEdit = sPref.edit();  
 myEdit.putInt("city",spinner.getSelectedItemPosition());  
 myEdit.putString("cityname",spinner.getSelectedItem().toString());  
 myEdit.putString("cityurl",citiesURL[spinner.getSelectedItemPosition()]);  
  
 int checkedIdP = skinTypeRadioG.getCheckedRadioButtonId();  
 RadioButton checkedRadioButtonP = (RadioButton)skinTypeRadioG.findViewById(checkedIdP);  
 int checkedIndexP = skinTypeRadioG.indexOfChild(checkedRadioButtonP);  
 myEdit.putInt("skinType",checkedIndexP);  
  
  
 myEdit.apply();  
 Toast.*makeText*(this,"Сохранено: город "+spinner.getSelectedItem().toString(),Toast.*LENGTH\_SHORT*).show();  
}

Приложение F

//Метод загрузки настроек пользователя

public void LoadSettings()  
{  
 SharedPreferences sPref = getSharedPreferences("Settings",Context.*MODE\_PRIVATE*);  
 spinner.setSelection(sPref.getInt("city",0));  
  
 Integer idp = sPref.getInt("pulldatalogs",0);  
 if(idp!=-1)  
 ((RadioButton)skinTypeRadioG.getChildAt((sPref.getInt("skinType",0)))).setChecked(true);  
  
 LocationManager manager = (LocationManager) getSystemService(Context.*LOCATION\_SERVICE*);  
 if (ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED*) {  
 // Проверка наличия разрешений  
 // Если нет разрешения на использование соответсвующих разркешений выполняем какие-то действия  
 Permissons.*Request\_FINE\_LOCATION*(SettingsActivity.this,2);  
 return;  
 }  
 manager.requestSingleUpdate (LocationManager.*NETWORK\_PROVIDER*, listener, null);  
}

Приложение G

//исходный код главной страницы приложения activity\_home.xml

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".HomeActivity">  
 <TextView  
 android:id="@+id/infotext"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent" />  
 <RelativeLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:gravity="center">  
 <TextView  
 android:id="@+id/homeUVIndexInfo"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="УФ индекс"  
 android:textSize="20dp"  
 android:textColor="@color/homeColor"  
 android:textStyle="bold"  
  
 android:gravity="center"/>  
 <TextView  
 android:id="@+id/cityName"  
 android:layout\_below="@+id/homeUVIndexInfo"  
 android:text="@string/city"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"/>  
 <TextView  
 android:layout\_below="@+id/cityName"  
 android:id="@+id/homeUVIndex"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textSize="50dp"  
 android:text="10"  
 android:gravity="center"  
 android:textColor="@color/homeColor" />  
 <TextView  
 android:layout\_below="@+id/homeUVIndex"  
 android:id="@+id/homeUVdate"  
 android:layout\_width="match\_parent"  
 android:gravity="center"  
 android:layout\_height="wrap\_content"  
 android:text="29.04.2019"  
 android:textSize="15dp" />  
 </RelativeLayout>  
  
  
</RelativeLayout>