

Мобильное приложение и алгоритм для составления индивидуального плана вакцинации детей

В.А. Кузнецова, Д.Ю. Ильин, Т.В. Беляева

Аннотация—Педиатры, руководствуясь инструкциями к вакцинам, национальным календарем прививок и другими нормативно-правовыми документами, регламентирующими вакцинопрофилактику в Российской Федерации, вручную составляют индивидуальные планы вакцинации детей, которые обеспечивают эффективную защиту от опасных инфекций. На составление корректного индивидуального плана вакцинации требуется много времени врача, ввиду чего рациональна разработка инструмента, который позволял бы автоматизировать составление таких расписаний, так как анализ существующих решений показал, что имеющиеся приложения не решают данную задачу.

В статье рассматривается мобильное приложение, разработанное на языке программирования Kotlin, которое позволяет составлять индивидуальный план вакцинации ребенка за приемлемое время. Предложено алгоритмическое решение для составления расписания на основе гибридного подхода. В качестве входных данных используются дата рождения ребенка, введенные ранее вакцины (при наличии), дни работы прививочного кабинета, производственный календарь РФ и правила, по которым необходимо комбинировать вакцины.

Представлены результаты проектирования мобильного приложения. Хранение данных о характеристиках вакцин и правилах осуществляется в формате JSON. Приведены примеры составленных индивидуальных планов вакцинации при разных входных данных, а также скриншоты пользовательского интерфейса приложения.

Ключевые слова—мобильное приложение, Android, Kotlin, расписание, гибридный алгоритм составления расписания, вакцинация, индивидуальный план вакцинации.

I. ВВЕДЕНИЕ

Одним из важнейших способов поддержания здоровья является вакцинация, определенная Всемирной организацией здравоохранения (ВОЗ) как эффективный и безопасный способ защиты общества от инфекционных заболеваний [1].

Особое внимание уделяется вакцинации детей,

Статья получена 07 апреля 2023.

В. А. Кузнецова, МИРЭА – Российский технологический университет, Москва, Россия (e-mail: kuznetsova.valeria.7@gmail.com).

Д. Ю. Ильин, МИРЭА – Российский технологический университет, Москва, Россия (e-mail: i@dmityilin.com).

Т. В. Беляева, Российская медицинская академия непрерывного профессионального образования, Москва, Россия (e-mail: docbelyaeva@gmail.com)

которая регламентируется нормативно-правовыми документами, основным из которых является национальный календарь прививок (НКП) [2]. Кроме регламентирующих документов для составления индивидуального плана вакцинации (ИПВ) педиатру необходимо учитывать различные факторы, такие как возраст ребенка, текущее состояние его здоровья, данные о ранее проведенной вакцинации. Авторы [3] отмечают, что составленные ИПВ показывают свою эффективность, но на составление соответствующего требованиям плана уходит много времени. Таким образом, автоматизация процесса могла бы ускорить его составление и освободить время педиатра для других задач.

На рабочий компьютер врача в поликлинике может не быть возможности произвести установку стороннего программного обеспечения, по этой причине целесообразно рассмотреть существующие решения для мобильных устройств. На сегодняшний день представлено несколько приложений:

1. Приложение «Прививки - личный календарь» (АО «НПО «Микроген»») позволяет составлять план вакцинации людей старше 18 лет на основании национального календаря прививок РФ. Расписание меняется в зависимости от того, когда были введены прививки.
2. Приложение «Child Vaccination Schedule» (Nilesh Harde) помогает составлять ИПВ детей, но оно ориентировано на другую страну и не позволяет редактировать план.
3. Приложение «CDC Vaccine Schedules» (Centers for Disease Control and Prevention) содержит информацию только о национальном календаре прививок и не подходит для пользователей из России.
4. Приложение «Vaccination Chart For Children» (Brain 2016) не может составлять ИПВ на основе даты рождения ребенка и содержит информацию о национальном календаре прививок другой страны.

С помощью большинства рассмотренных решений невозможно составить ИПВ ребенка. Если в приложении есть такая функция, то отредактировать расписание на основе дат уже введенных прививок не удастся.

В связи с этим, целью статьи является разработка

программного обеспечения автоматизации составления индивидуальных планов вакцинации детей.

В соответствии с поставленной целью необходимо решить такие задачи, как:

1. провести анализ предметной области и существующих решений;
2. определить требования к программному обеспечению;
3. выполнить проектирование программного обеспечения;
4. осуществить разработку и тестирование программного обеспечения.

II. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

По данным StatCounter [4] по состоянию на февраль 2023 года более 73% всех мобильных устройств в России работают на базе операционной системы (ОС) Android, в то время как доля устройств с iOS составляет около 26,5%. На рисунке 1 приведена статистика использования мобильных операционных систем. В связи с этим разработка приложения для ОС Android является приоритетной.

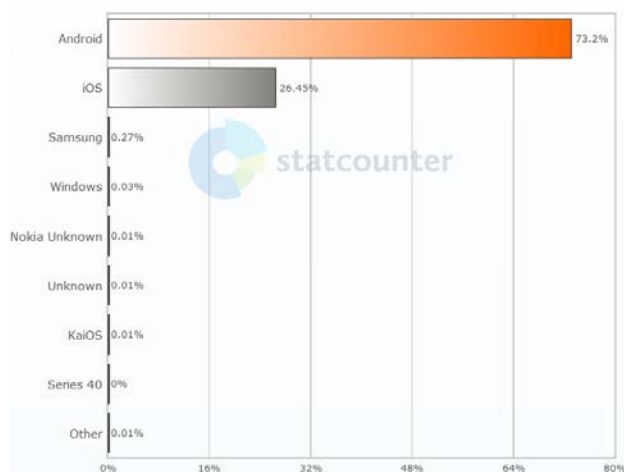


Рисунок 1 – Статистика использования мобильных операционных систем в России

Методы календарного планирования рассматриваются в теории расписаний. Теория расписаний – это область исследования операций, в которой строятся и анализируются математические модели календарного планирования различных целенаправленных действий с учетом целевой функции и различных ограничений [5]. Исследованные задачи из данной области в общем случае являются NP-полными, но при этом для них необходимо находить решения [6]. Существует ряд подходов [7], позволяющих найти необходимое расписание, а некоторые из них позволяют найти субоптимальное решение за приемлемое время.

1. Точные методы решения NP-полных задач. В их основе лежит полный перебор всех возможных решений, ввиду чего такие методы имеют высокую ресурсоемкость.
2. Эвристические методы [8]. Они основаны на алгоритмах, которые используют логические

предположения, не подкрепленные математическим обоснованием. Решения, сформированные с помощью эвристик (правил), в большинстве случаев являются субоптимальными.

3. Динамическое программирование. Предполагает разбиение задачи на более простые подзадачи [9]. Алгоритм на базе динамического программирования не всегда может быть эффективным, так как существуют задачи, для которых нет разумных способов сведения их к подзадачам, хотя такой способ решения может быть менее ресурсоемким в сравнении с полным перебором.
4. Программирование в ограничениях. Составление расписания рассматривается как задача удовлетворения некоторых ограничений [10].
5. Целочисленное линейное программирование. Заключается в оптимизации линейной функции с заданным набором линейных ограничений на переменных, являющихся целочисленными [11].
6. Гибридные алгоритмы. Основаны на комбинировании различных методов составления расписания таким образом, что недостатки одних способов компенсировались бы преимуществами других [12].

Авторы [13] и [14] изучали область теории расписаний в рамках задачи построения ИПВ и разработали мобильное приложение с алгоритмом, основанным на динамическом программировании, с помощью которого можно составлять план догоняющей вакцинации.

III. ТРЕБОВАНИЯ К РАЗРАБАТЫВАЕМОМУ МОБИЛЬНОМУ ПРИЛОЖЕНИЮ

На основе результатов исследования предметной области, а также обзора существующих решений, сформулированы требования к разрабатываемому мобильному приложению.

A. Функциональные требования

Диаграмма прецедентов, визуализирующая функциональные требования к разрабатываемому приложению, представлена на рисунке 2.

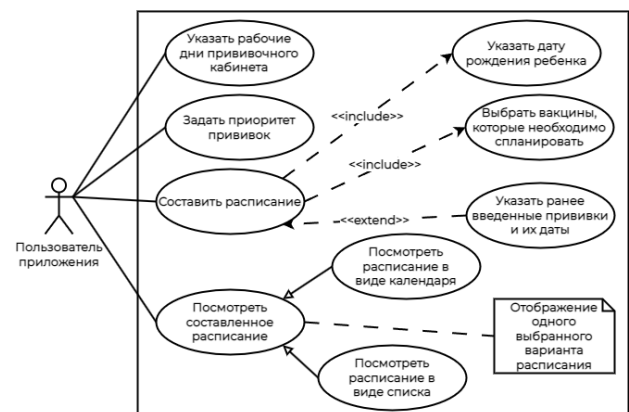


Рисунок 2 – Диаграмма прецедентов

Приложение для составления ИПВ должно соответствовать следующим функциональным требованиям и предоставлять такие возможности, как:

1. указание даты рождения ребенка;
2. указание рабочих дней недели прививочного кабинета;
3. выбор перечня вакцин, на основе которых необходимо составить ИПВ;
4. указание даты введения для предоставленных ранее вакцин;
5. построение ИПВ по указанной дате рождения ребенка, учитывая при этом рабочие дни прививочного кабинета и выходные дни по производственному календарю Российской Федерации.

В. Нефункциональные требования

Основные нефункциональные требования к разрабатываемой системе можно сформулировать следующим образом:

1. решение должно быть разработано для ОС Android;
2. приложение должно быть ориентировано на использование людьми с медицинским образованием (педиатрами);
3. мобильное приложение должно осуществлять планирование на срок в один год;
4. система не должна осуществлять хранение персональных данных.

Также, для того чтобы пользоваться приложением, не должно быть необходимости в предварительной регистрации. Так как с системой взаимодействует только один пользователь, информация должна храниться локально.

IV. ПОСТАНОВКА ЗАДАЧИ СОСТАВЛЕНИЯ ИНДИВИДУАЛЬНОГО ПЛАНА ВАКЦИНАЦИИ

Пусть задано множество всех доступных вакцин для детей $V_{all} = \{v_1, v_2, \dots, v_k\}$, из которого формируется множество предполагаемых к введению вакцин $V_{proposed}$, такое, что $\forall V_{proposed} \subseteq V_{all}$. Также пусть задано множество дней $d \in D$, $D = \overline{\{1, 365\}}$, в которые возможно произвести введение вакцин. Пусть задана такая функция $f : (V_{proposed}, D) \mapsto V_{scheduled}^0$, которая ставит каждой вакцине из множества $V_{proposed}$ в соответствие день d из множества D с учетом правил и ограничений, заданных предметной областью и формирует множество пар $V_{scheduled}^0$.

Тогда базовое расписание вакцинации $V_{scheduled}^0 = \{s_i = (v_i, d) : v_i \in V_{proposed}, d \in D, i = \overline{1, n}\}$,

где $|V_{scheduled}^0| = |V_{proposed}| = n$. Вакцины v_i при этом уникальны и не повторяются в пределах множества $V_{scheduled}^0$, тогда как дни d для введения вакцины могут повторяться.

Пусть задана функция $g : (V_{scheduled}^0) \mapsto V_{scheduled}^*$, которая модифицирует базовое расписание. При этом для каждой пары $s_i^* \in V_{scheduled}^*$ в ходе модификации допустимо изменение дня введения вакцины d , но не самой вакцины v_i , то есть:

$$\forall V_{scheduled}^* : \{proj_1(s_i) \mid s_i \in V_{scheduled}^0, i = \overline{1, n}\} = \{proj_1(s_i^*) \mid s_i^* \in V_{scheduled}^*, i = \overline{1, n}\}, \quad (1)$$

где функция $proj_1(s_i)$ означает получение первого элемента из кортежа s_i , а $proj_1(s_i^*)$ – получение первого элемента из кортежа s_i^* соответственно.

Оптимальность расписания определяется по двум критериям:

$$J_1 = \min_g \left(\sum_{i=1}^n proj_2(s_i^*) \right), \quad (2)$$

$$J_2 = \min_g \left(\left| \{proj_2(s_i^*) \mid s_i^* \in V_{scheduled}^*, i = \overline{1, n}\} \right| \right). \quad (3)$$

Полученные оценки расписания принадлежат множеству целых неотрицательных чисел.

V. ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО РЕШЕНИЯ

А. Архитектура приложения

Приложение должно состоять из нескольких компонентов.

1. Модуль пользовательского интерфейса. Отвечает за отображение информации пользователю и помогает взаимодействовать с ним. Для отображения данных необходима информация из хранилищ данных о вакцинах и о настройках.
2. Модуль с парсером производственного календаря. Данный компонент обращается ко внешнему сервису данных о производственном календаре, получает необходимую информацию и преобразовывает ее в вид, необходимый для работы приложения.
3. Модуль построения календаря. Часть мобильного приложения, участвующая в составлении ИПВ. Источниками данных для данного компонента являются несколько хранилищ: данные о вакцинах, настройки, данные производственного календаря.

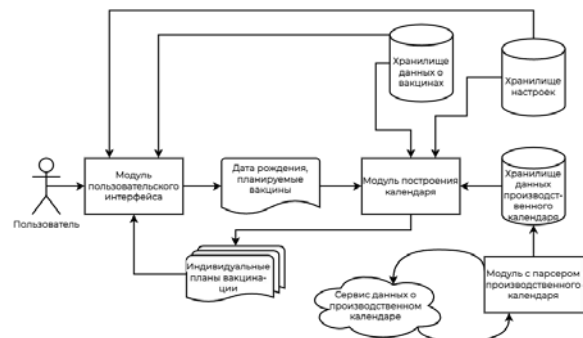


Рисунок 3 – Компоненты мобильного приложения
На рисунке 3 представлены компоненты

разрабатываемого мобильного приложения.

При проектировании мобильного приложения используются принципы Clean Architecture с применением архитектурного шаблона Model-View-ViewModel (MVVM), помогающего отделить пользовательский интерфейс от логики приложения.

В качестве языка программирования (ЯП) был выбран Kotlin [15], так как он признан предпочтительным для создания приложений под Android [16].

В. Порядок работы системы

Для более полного понимания работы системы необходимо определить порядок ее функционирования. На рисунке 4 представлена схема работы системы проектируемого приложения.

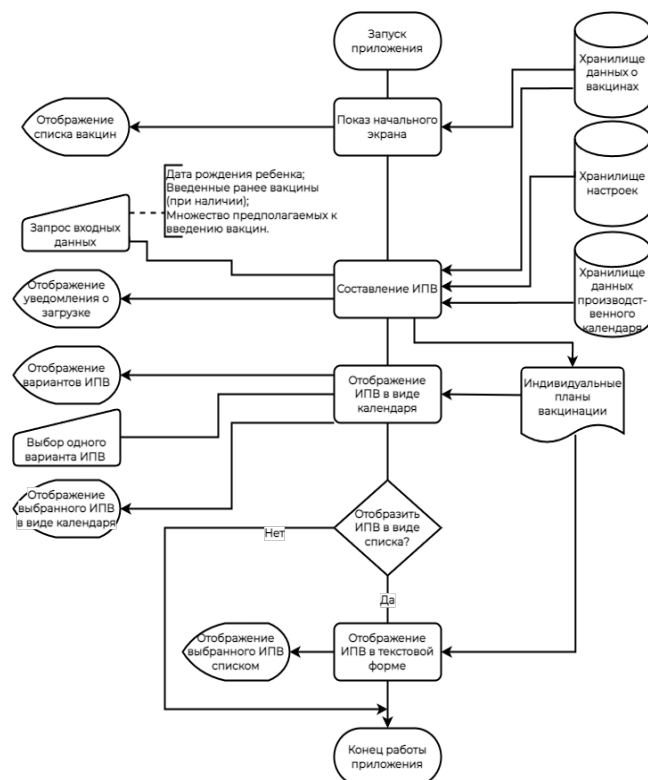


Рисунок 4 – Схема работы системы для проектируемого приложения

После запуска приложения пользователю отображается начальный (главный) экран со списком вакцин, на котором он может внести все необходимые данные. После того, как пользователь введет все данные и нажмет кнопку «Составить календарь», начнется составление ИПВ в соответствии со всеми необходимыми правилами и ограничениями, а ему отобразится уведомление о загрузке.

После того, как индивидуальные планы вакцинации будут составлены, пользователю отобразится экран с представлением одного из ИПВ в виде календаря. На этом экране пользователь сможет выбрать, какой вариант ИПВ он хочет посмотреть. Также, по нажатию кнопки «Получить расписание текстом», пользователю отобразится в виде списка (в текстовой форме) тот план вакцинации, который он выбрал.

С. Способ хранения данных

В ОС Android предусмотрено хранение данных различными способами, каждый из которых имеет свои достоинства и недостатки. Как правило, разные способы хранения используют одновременно, повышая тем самым эффективность работы с данными.

Для разрабатываемой системы рациональным способом хранения данных являются файлы в формате JSON [17], предназначенные для хранения слабоструктурированных данных. Файлы такого типа удобны как для машинной обработки, так и для редактирования вручную. Формат широко распространен, и практически во всех ЯП реализованы механизмы взаимодействия с JSON-файлами.

На рисунке 5 показан пример файла в формате JSON, используемого в мобильном приложении. В файлах представленной структуры осуществляется хранение данных о вакцинах и сопутствующих правилах введения.

```
{
  "name": "Фенатит В",
  "drugName": "Перевак",
  "stage": "RV1",
  "isAlive": false,
  "insertionSite": "верхне-наружная поверхность средней части бедра",
  "daysAfterBirthMin": 183,
  "maxYearsForVaccineInDays": 6569,
  "daysAfterAnyVaccination": 0,
  "thisVaccinationAfterAnyVaccinationWithoutObservingAStrictInterval": null,
  "daysBeforeAnyVaccination": 0,
  "thisVaccinationBeforeAnyVaccinationWithoutObservingAStrictInterval": null,
  "previousVaccinationAgainstTheSameDisease": {
    "namePreviousVaccination": "Фенатит В",
    "namePreviousDrug": "Перевак",
    "stagePreviousVaccination": "V2",
    "minDelayAfterPreviousVaccination": 183,
    "maxDelayAfterPreviousVaccination": 547
  },
  "alreadyGotTheVaccine": false
}
```

Рисунок 5 – Файл с описанием характеристик вакцины в формате JSON

Настройки приложения целесообразно хранить в Shared Preferences (SP) – простом хранилище типа «ключ-значение».

Д. Алгоритм составления ИПВ

Гибридные алгоритмы являются эффективным способом решения задач, так как они способны комбинировать преимущества различных классических алгоритмов. В связи с этим в основе предлагаемого решения используется гибридный подход для составления ИПВ.

Разработанный алгоритм составления расписаний состоит из трех стадий.

На первой стадии составляется базовое расписание $V_{\text{scheduled}}^0$ с учетом выходных и праздничных дней по производственному календарю РФ, в соответствии с инструкциями к препаратам, а также со строгими правилами, регламентированными НКП.

На вход поступают:

1. «Days» – массив D , каждый элемент которого содержит информацию о конкретном дне;
2. «ProposedVaccines» – массив рассматриваемых к введению вакцин V_{proposed} , в каждом элементе которого содержится информация о планируемой вакцине;

3. «AdministeredVaccines» – массив уже введенных вакцин $V_{administered}$, где каждый элемент содержит информацию о вакцине и дату ее введения.

Происходит проверка по каждому дню из массива D : если день $D[i]$ является рабочим, рассматривается каждая планируемая вакцина $V_{proposed}[j]$ из массива $V_{proposed}$. Сначала осуществляется проверка возможности введения вакцины $V_{proposed}[j]$ в выбранный день $D[i]$ (проверяются такие ограничения вакцины, как: соблюдение всех необходимых интервалов для постановки вакцины; минимальный и максимальный возраст введения; проставлены ли необходимые для введения этой вакцины прививки; правила из НКП по постановке живых и неживых вакцин). Если вакцину возможно ввести в этот день, происходит проверка на совместимость планируемой вакцины $V_{proposed}[j]$ с каждой уже введенной вакциной $V_{administered}[k]$ из массива $V_{administered}$ (проверяется соблюдение всех необходимых интервалов, так как у уже введенных вакцин могут быть какие-либо свои специфические ограничения). Если планируемая вакцина $V_{proposed}[j]$ совместима со всеми вакцинами из $V_{administered}$, проверяются ограничения конкретного дня $D[i]$ (осуществляется проверка на максимально допустимое количество прививок за один раз). С помощью алгоритма, представленного на рисунке 6, можно построить базовое расписание $V_{scheduled}^0$.

На второй стадии созданное базовое расписание $V_{scheduled}^0$ преобразовывается во множество вариантов путем передвижения вакцин.

Базовое расписание $V_{scheduled}^0$ подвергается изменению заданное количество раз ($L = 1000$): каждый раз берется начальное расписание $V_{scheduled}^0$, в котором по очереди происходит перестановка каждой вакцины $V_{scheduled}^0[t]$ с вероятностью 0,5. При передвижении вакцин обращается внимание на их объединение для сокращения числа посещений поликлиники. Если вакцину возможно передвинуть так, чтобы расписание оставалось корректным и удовлетворяло всем условиям, также сдвигаются все последующие этапы этой вакцины, чтобы не нарушались правила введения препарата. Благодаря таким передвижениям вакцин появляется возможность составлять разнообразные расписания $V_{scheduled}^*$, основываясь на одном базовом варианте.

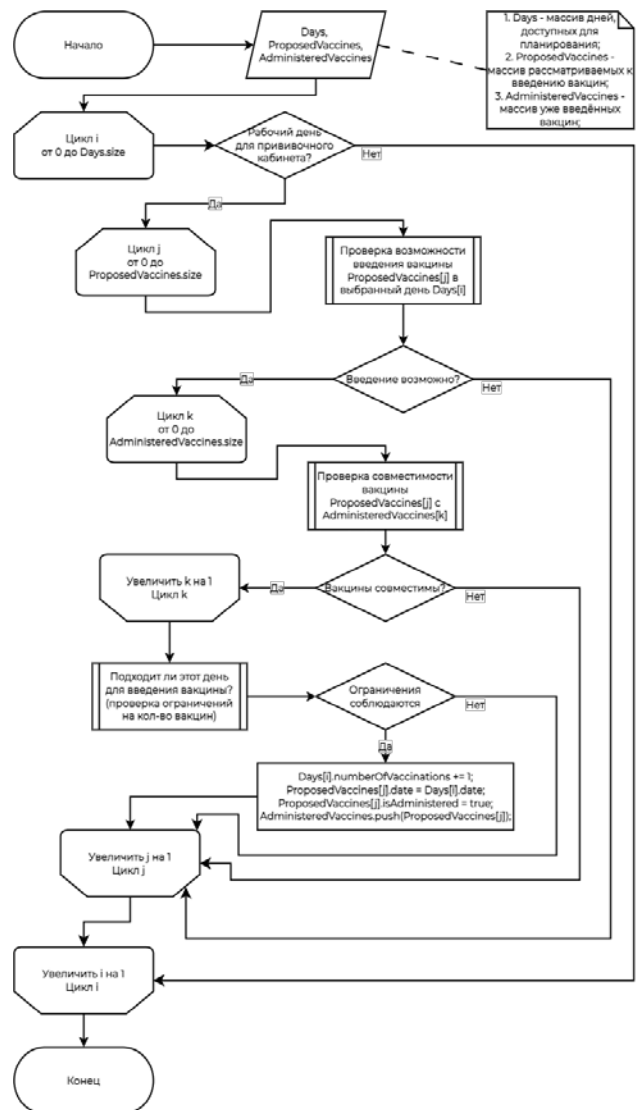


Рисунок 6 – Схема алгоритма составления базового расписания

На рисунке 7 представлена схема алгоритма перестановки одной вакцины.

На вход поступают:

1. «AdministeredVaccines» – массив $V_{administered_shift}$ уже введенных вакцин, каждый элемент которого содержит информацию о вакцине и дате ее введения;
2. «ProposedVaccine» – вакцина $V_{proposed_shift} = V_{scheduled}^0[t]$, рассматриваемая к передвижению.

Для каждой уже введенной вакцины $V_{administered_shift}[i]$ из $V_{administered_shift}$: начальная дата $D_{proposed}$ вакцины $V_{proposed_shift}$, полученная из элемента базового расписания $V_{scheduled}^0[t]$, сравнивается с датой $D_{administered}$ вакцины $V_{administered_shift}[i]$. Если разница между ними не более двух недель, то происходит проверка условий на совместимость этих двух вакцин в один день $D_{administered}$ (для этого необходимо заново спланировать вакцину $V_{proposed_shift}$, как на первой стадии, при условиях, что массив «ProposedVaccines» состоит из одной вакцины

$V_{proposed_shift}$, следующих этапов вакцины $V_{proposed_shift}$ нет в массиве «AdministeredVaccines», а массив «Days» состоит из одного дня – даты $D_{administered}$). Если введение вакцины $V_{proposed_shift}$ в день $D_{administered}$ возможно, то необходимо переставить все следующие этапы $V_{proposed_shift}$ (перестановка следующих этапов вакцины происходит как на первой стадии, при условии, что массив «ProposedVaccines» состоит только из следующих этапов вакцины $V_{proposed_shift}$, а массив «AdministeredVaccines» состоит из всех остальных введенных вакцин $V_{administered_shift}$, а также содержит $V_{proposed_shift}$).

На третьей стадии из множества составленных ИПВ $V_{scheduled}^*$ выбираются Парето-оптимальные решения [18] по двум критериям J_1 и J_2 , представленным в формулах (2) и (3):

1. суммарное количество дней, за которое ребенок приобретет иммунитет;
2. количество посещений поликлиники.

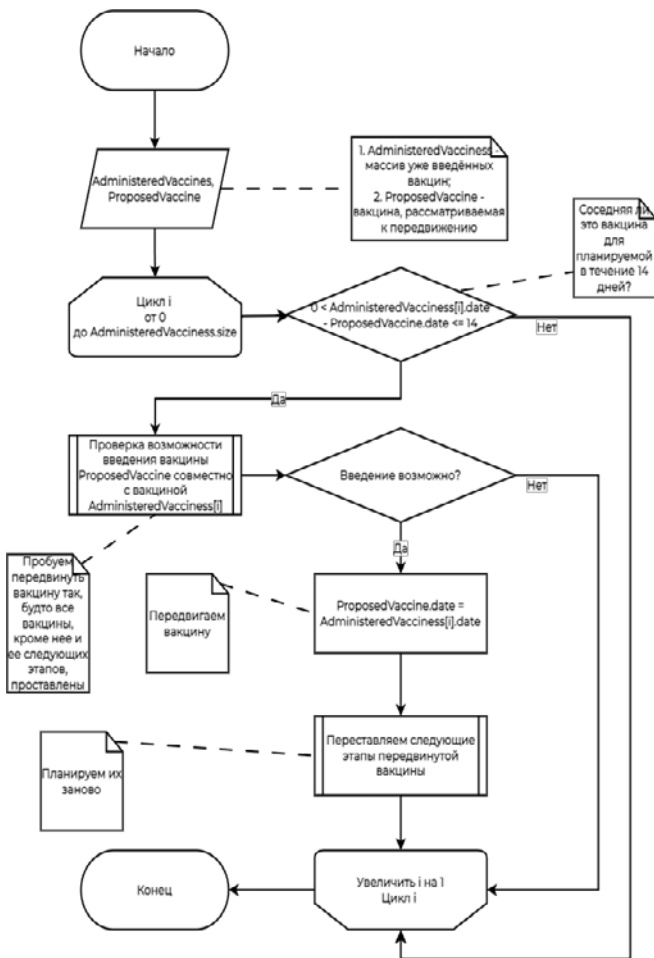


Рисунок 7 – Схема алгоритма передвижения одной вакцины

На рисунке 8 изображено графическое представление Парето-фронта. Подходящими решениями являются те для которых нет альтернативы, которая была бы эффективнее сразу по двум критериям.

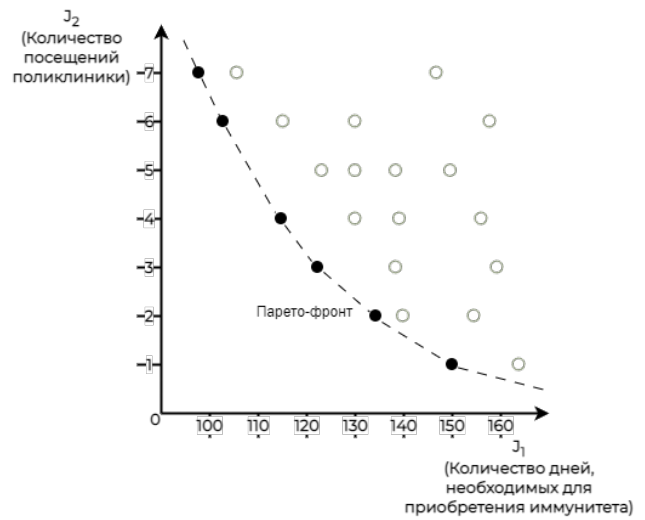


Рисунок 8 – Графическое представление Парето-фронта

Отбор Парето-оптимальных решений происходит с помощью алгоритма, изображенного на рисунке 9.

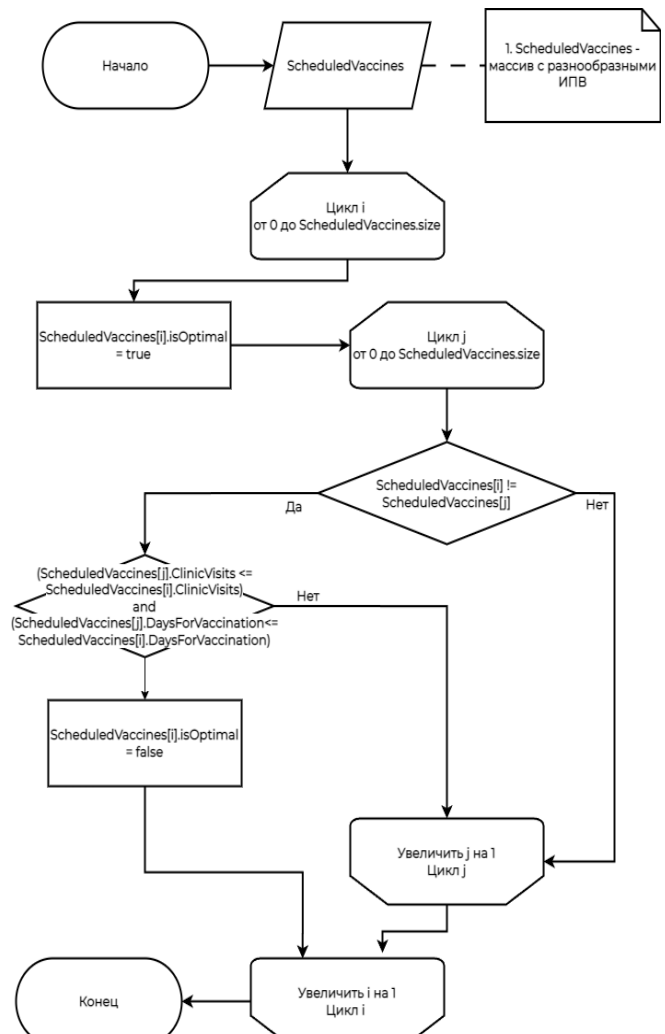


Рисунок 9 – Схема алгоритма выбора Парето-оптимальных решений

На вход поступает массив со всеми возможными вариантами разнообразных расписаний $V_{scheduled}^*$. Необходимо поочередно сравнить каждый элемент массива $V_{scheduled}^*$ с каждым, чтобы найти Парето-

оптимальные решения. Если существует ИПВ, который является более эффективным сразу по двум критериям в сравнении с рассматриваемым вариантом, то такой ИПВ не принадлежит множеству Парето-фронта и не подходит для дальнейшего рассмотрения.

VI. ДЕМОНСТРАЦИЯ РАЗРАБОТАННОГО МОБИЛЬНОГО ПРИЛОЖЕНИЯ

А. Пользовательский интерфейс

На рисунке 10 представлен пользовательский интерфейс разработанного мобильного приложения.

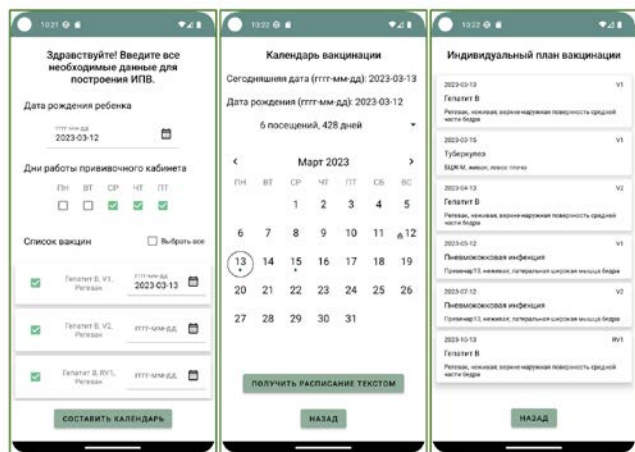


Рисунок 10 – Пользовательский интерфейс разработанного мобильного приложения

Приложение рассчитано на пользователей из России, оно имеет интерфейс на русском языке и работает на операционной системе Android (версия 8.0 и выше). Экраны мобильного приложения выполнены в едином стиле (Material Design [19]) и единой цветовой гамме. В рамках прототипа представлен ограниченный перечень препаратов для проведения вакцинации.

В. Результаты составления ИПВ

Далее приведены четыре сценария составления ИПВ, полученные для двух разных детей с различными входными данными. Расчеты производились 14.03.2023. Для разработанного прототипа распределение вакцин по календарю занимает от 5 до 7 секунд на мобильном устройстве с процессором Hisilicon Kirin 710.

Случай 1. Необходимо составить ИПВ на три месяца для новорожденного ребенка (14.03.2023), еще не получившего ни одной прививки. Рабочие дни недели прививочного кабинета – вторник, четверг, пятница. Заболевания, от которых необходимо ввести прививки: гепатит В (этапы V1, V2); туберкулез (V1); пневмококковая инфекция (V1); полиомиелит (V1); гемофильная инфекция типа b (V1).

Приложение составило два расписания:

1. 5 посещений поликлиники и 283 дня на приобретение иммунитета: гепатит В (V1) – 14.03.2023; туберкулез (V1) – 17.03.2023; гепатит В (V2) – 14.04.2023; пневмококковая инфекция (V1) – 16.05.2023; полиомиелит (V1)

– 15.06.2023; гемофильная инфекция типа b (V1) – 15.06.2023.

2. 6 посещений поликлиники и 281 день на приобретение иммунитета: гепатит В (V1) – 14.03.2023; туберкулез (V1) – 17.03.2023; гепатит В (V2) – 14.04.2023; пневмококковая инфекция (V1) – 16.05.2023; полиомиелит (V1) – 13.06.2023; гемофильная инфекция типа b (V1) – 15.06.2023.

Случай 2. Трехмесячный ребенок (12.12.2022), получивший не все необходимые прививки к своему возрасту (гепатит В (V1) – 12.12.2022; туберкулез (V1) – 16.12.2022; пневмококковая инфекция (V1) – 13.02.2023; полиомиелит (V1) – 13.03.2023). Необходимо ввести недостающие прививки (гепатит В (V2); гемофильная инфекция типа b (V1)), а также спланировать следующие: пневмококковая инфекция (V2); гемофильная инфекция типа b (V2, V3); полиомиелит (V2). Рабочие дни недели прививочного кабинета – понедельник, среда, пятница.

Приложение составило два расписания (распределялись только вакцины с неуказанными датами, а даты ранее введенных прививок остались прежними):

1. 3 посещения поликлиники и 205 дней на приобретение иммунитета: гепатит В (V2) – 15.03.2023; гемофильная инфекция типа b (V1) – 15.03.2023; полиомиелит (V2) – 17.04.2023; пневмококковая инфекция (V2) – 17.04.2023; гемофильная инфекция типа b (V2) – 17.04.2023; гемофильная инфекция типа b (V3) – 14.06.2023.
2. 4 посещения поликлиники и 202 дня на приобретение иммунитета: гепатит В (V2) – 15.03.2023; гемофильная инфекция типа b (V1) – 15.03.2023; полиомиелит (V2) – 14.04.2023; пневмококковая инфекция (V2) – 17.04.2023; гемофильная инфекция типа b (V2) – 17.04.2023; гемофильная инфекция типа b (V3) – 14.06.2023.

VII. ЗАКЛЮЧЕНИЕ

В данной работе была рассмотрена возможность автоматизации составления индивидуальных планов вакцинации детей.

Было определено, что существующие приложения не решают задачу построения ИПВ детей либо не учитывают нормативно-правовые документы, регламентирующие вакцинопрофилактику в Российской Федерации, ввиду чего сделан вывод о целесообразности разработки программного обеспечения для автоматизации составления индивидуальных планов вакцинации детей.

Сформулирован ряд требований к программному обеспечению, на основе которых спроектировано и разработано мобильное приложение для составления ИПВ, состоящее из нескольких компонентов. Была представлена схема работы системы и было предложено алгоритмическое решение для составления ИПВ на

основе гибридного подхода. Мобильное приложение было разработано на языке программирования Kotlin для операционной системы Android.

С помощью готового решения составление ИПВ можно производить быстрее, чем вручную. Благодаря автоматизации такой рутинной работы возможно освобождение рабочего времени врачей педиатров.

Приложение является прототипом, в будущем планируется его доработка и добавление нового функционала. Необходимо расширение списка доступных вакцин и усовершенствование алгоритма составления ИПВ таким образом, чтобы была возможность планирования разных этапов вакцинации разными препаратами. Планируется также добавить догоняющий вид вакцинации.

БИБЛИОГРАФИЯ

- [1] C. M. C. Rodrigues and S. A. Plotkin, "Impact of Vaccines; Health, Economic and Social Perspectives," *Frontiers in Microbiology*, vol. 11, p. 1526, Jul. 2020, doi: 10.3389/fmicb.2020.01526.
- [2] Российская Федерация, Министерство здравоохранения, "Об утверждении национального календаря профилактических прививок, календаря профилактических прививок по эпидемическим показаниям и порядка проведения профилактических прививок" : Приказ Министерства здравоохранения РФ от 6 декабря 2021 г. № 1122н, URL: <http://edu.garant.ru/> (дата обращения: 25.09.2022).
- [3] M. W. Kreuter, C. A. Caburnay, J. J. Chen, and M. J. Donlin, "Effectiveness of Individually Tailored Calendars in Promoting Childhood Immunization in Urban Public Health Centers," *American Journal of Public Health*, vol. 94, no. 1, pp. 122–127, Jan. 2004, doi: 10.2105/ajph.94.1.122.
- [4] "Mobile Operating System Market Share Russian Federation," *StatCounter Global Stats*, URL: <https://gs.statcounter.com/os-market-share/mobile/russian-federation/> (accessed Mar. 15, 2023).
- [5] А. А. Лазарев, *Теория расписаний. Методы и алгоритмы*. Москва: ИПУ РАН, 2019.
- [6] Т. Рафгарден, *Совершенный алгоритм. Алгоритмы для NP-трудных задач*. Санкт-Петербург: Издательский дом «Питер», 2021.
- [7] Х. Пападимитриу, У. Вазирани, С. Дасгупта, *Алгоритмы*, издание 3-е, стереотипное. Москва: МЦНИП, 2023.
- [8] G. Blokdyk, *Heuristic (computer science)*, 2nd ed. London: 5STARCOoks, 2018.
- [9] Т. Рафгарден, *Совершенный алгоритм. Жадные алгоритмы и динамическое программирование*. Санкт-Петербург: Издательский дом «Питер», 2020.
- [10] И. Устинов, И. Набродова, "Анализ существующих алгоритмов для составления расписания занятий," *Известия Тульского Государственного Университета Технические Науки*, № 9, 2022.
- [11] Л. Заозерская, В. Ильев, Т. Леванова, *Методы оптимизации. Целочисленное линейное программирование*. Омск: Издательство Омского государственного университета, 2020.
- [12] Е. А. Кумагина, Е. А. Неймарк, "Разработка гибридного алгоритма решения задачи упорядочения," *Труды НГТУ Им П Е Алексева*, № 3 (126), с. 17–23, 2019.
- [13] F. G. Engineer, P. Keskinocak, and L. K. Pickering, "Catch-Up Scheduling for Childhood Vaccination," *Operations Research*, vol. 57, no. 6, pp. 1307–1319, 2009, doi: 10.1287/opre.1090.0756.
- [14] H. K. Smalley, P. Keskinocak, F. G. Engineer, and L. K. Pickering, "Universal Tool for Vaccine Scheduling: Applications for Children and Adults," *Interfaces*, vol. 41, no. 5, pp. 436–454, 2011, doi: 10.1287/inte.1110.0583.
- [15] L. Ardito, R. Coppola, G. Malnati, and M. Torchiano, "Effectiveness of Kotlin vs. Java in android app development tasks," *Information and Software Technology*, vol. 127, 2020, doi: 10.1016/j.infsof.2020.106374.
- [16] M. Martinez and B. Gois Mateus, "Why Did Developers Migrate Android Applications From Java to Kotlin?," *IEEE Transactions on Software Engineering*, vol. 48, no. 11, pp. 4521–4534, 2022, doi: 10.1109/TSE.2021.3120367.
- [17] P. Bourhis, J. L. Reutter, and D. Vrgoč, "JSON: Data model and query languages," *Information Systems*, vol. 89, 2020, doi: 10.1016/j.is.2019.101478.
- [18] V. Roostapour, A. Neumann, F. Neumann, and T. Friedrich, "Pareto optimization for subset selection with dynamic cost constraints," *Artificial Intelligence*, vol. 302, 2022, doi: 10.1016/j.artint.2021.103597.
- [19] "Material Design," *Material Design*, URL: <https://m2.material.io/> (accessed Jan. 15, 2023).

Mobile application and algorithm for individual vaccination schedule tailoring for children

V.A. Kuznetsova, D.Y. Ilin, T.V. Belyaeva

Abstract—Pediatricians, guided by the instructions for vaccines, the national vaccination calendar and other legal documents regulating vaccination in the Russian Federation, tailor up individual vaccination schedules for children to ensure the effective protection against dangerous infections. It takes a lot of time to tailor up a correct individual vaccination schedule manually, so it is rational to develop a tool that would automate the process, since an analysis of existing solutions has shown that they do not address this task.

The article discusses a mobile application developed in the Kotlin programming language, which allows one to make an individual vaccination schedule for a child in a reasonable time. An algorithmic solution for scheduling based on a hybrid approach is proposed. The date of birth of the child, the list of previously administered vaccines (if any), the days of operation of the vaccination room, the working days calendar of the Russian Federation and the rules by which vaccines can be combined are used as an input data.

The results of designing a mobile application are presented. Vaccine specifications and regulations are stored in JSON format. Examples of produced individual vaccination schedules for different input data are given, as well as screenshots of the application's user interface.

Keywords—mobile application, Android, Kotlin, schedule, hybrid scheduling algorithm, vaccination, individual vaccination schedule.

REFERENCES

- [1] C. M. C. Rodrigues and S. A. Plotkin, "Impact of Vaccines; Health, Economic and Social Perspectives," *Frontiers in Microbiology*, vol. 11, p. 1526, Jul. 2020, doi: 10.3389/fmicb.2020.01526.
- [2] Russian Federation, Ministry of Health, "On approval of the national calendar of preventive vaccinations, the calendar of preventive vaccinations according to epidemic indications and the procedure for conducting preventive vaccinations" : Order of the Ministry of Health of the Russian Federation of December 6, 2021, no. 1122n, URL: <http://edu.garant.ru/> (accessed Sep. 25, 2023).
- [3] M. W. Kreuter, C. A. Caburnay, J. J. Chen, and M. J. Donlin, "Effectiveness of Individually Tailored Calendars in Promoting Childhood Immunization in Urban Public Health Centers," *American Journal of Public Health*, vol. 94, no. 1, pp. 122–127, Jan. 2004, doi: 10.2105/ajph.94.1.122.
- [4] "Mobile Operating System Market Share Russian Federation," *StatCounter Global Stats*, URL: <https://gs.statcounter.com/os-market-share/mobile/russian-federation/> (accessed Mar. 15, 2023).
- [5] A. A. Lazarev, *Schedule theory. Methods and algorithms*. Moscow: ICS RAS, 2019.
- [6] T. Roughgarden, *Algorithms Illuminated. Algorithms for NP-Hard Problems*. Saint Petersburg: Piter Publishing House, 2021.
- [7] C. Papadimitriou, U. Vazirani, and S. Dasgupta, *Algorithms*, 3rd, stereotypical ed. Moscow: MCNIP, 2023.
- [8] G. Blokyk, *Heuristic (computer science)*, 2nd ed. London: 5STARCOoks, 2018.
- [9] T. Roughgarden, *Algorithms Illuminated. Greedy Algorithms and Dynamic Programming*. Saint Petersburg: Piter Publishing House, 2020.
- [10] I. Ustinov and I. Nabrodova, "Analysis of existing algorithms for scheduling classes," *Izvestiya Tula State University Technical sciences*, no. 9, 2022.
- [11] L. Zaozerskaya, V. Ilyev, and T. Levanova, *Optimization Methods. Integer linear programming*. Omsk: Omsk State University Press, 2020.
- [12] E.A. Kumagina and E.A. Neumark, "A hybrid algorithm for permutation flowshop scheduling," *Transactions of NNSTU n. a. R. E. Alekseev*, no. 3 (126), pp. 17–23, 2019.
- [13] F. G. Engineer, P. Keskinocak, and L. K. Pickering, "Catch-Up Scheduling for Childhood Vaccination," *Operations Research*, vol. 57, no. 6, pp. 1307–1319, 2009, doi: 10.1287/opre.1090.0756.
- [14] H. K. Smalley, P. Keskinocak, F. G. Engineer, and L. K. Pickering, "Universal Tool for Vaccine Scheduling: Applications for Children and Adults," *Interfaces*, vol. 41, no. 5, pp. 436–454, 2011, doi: 10.1287/inte.1110.0583.
- [15] L. Ardito, R. Coppola, G. Malnati, and M. Torchiano, "Effectiveness of Kotlin vs. Java in android app development tasks," *Information and Software Technology*, vol. 127, 2020, doi: 10.1016/j.infsof.2020.106374.
- [16] M. Martinez and B. Gois Mateus, "Why Did Developers Migrate Android Applications From Java to Kotlin?," *IEEE Transactions on Software Engineering*, vol. 48, no. 11, pp. 4521–4534, 2022, doi: 10.1109/TSE.2021.3120367.
- [17] P. Bourhis, J. L. Reutter, and D. Vrgoč, "JSON: Data model and query languages," *Information Systems*, vol. 89, 2020, doi: 10.1016/j.is.2019.101478.
- [18] V. Roostapour, A. Neumann, F. Neumann, and T. Friedrich, "Pareto optimization for subset selection with dynamic cost constraints," *Artificial Intelligence*, vol. 302, 2022, doi: 10.1016/j.artint.2021.103597.
- [19] "Material Design," *Material Design*, URL: <https://m2.material.io/> (accessed Jan. 15, 2023).

About authors

Kuznetsova Valeria Andreevna, student at the department of Data processing digital technologies, MIREA – Russian technological university, Moscow, Russian Federation

Ilin Dmitry Yurievich, Ph.D. (Tech.), associate professor at the department of Data processing digital technologies, MIREA – Russian technological university, Moscow, Russian Federation

Belyaeva Tatyana Vadimovna, resident physician at the department of Allergology and Immunology of Russian Medical Academy for Continuous Medical Education, Moscow, Russian Federation