

## СОДЕРЖАНИЕ

СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ .....	8
ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ .....	9
ВВЕДЕНИЕ .....	11
Глава 1. Анализ предметной области и обзор существующих решений .....	13
1.1 Анализ предметной области .....	13
1.2 Определение особенностей существующих решений .....	15
1.2.1 Особенности решения “Sleep as Android” .....	16
1.2.2 Особенности решения “Sleep Cycle” .....	16
1.2.3 Особенности решения “Pillow” .....	17
1.2.4 Особенности решения “Better Sleep” .....	18
1.2.5 Сравнение и анализ используемых технологий .....	18
1.2.6 Сравнение и анализ существующих решений .....	21
Глава 2. Определение требований к системе и разработка архитектуры приложения .....	23
2.1 Определение требований к системе .....	23
2.2 Разработка архитектуры базы данных .....	28
2.3 Разработка архитектуры приложения .....	32
2.4 Выбор технологий для реализации .....	35
2.5 Описание конечных точек сервисов .....	37
Глава 3. Реализация и тестирование системы .....	40
3.1 Реализация heart_rate сервиса .....	40
3.1.1 Неавторизованный пользователь .....	40
3.1.2 Авторизованный пользователь .....	41
3.2 Реализация ML сервиса .....	44
3.3 Реализация Eureka сервиса .....	46

3.4 Реализация Gateway сервиса.....	47
3.5 Реализация клиентской части .....	47
3.6 Тестирование сервисов Spring .....	55
3.7 Тестирование моделей машинного обучения .....	56
3.8 Деплой приложения .....	56
3.9 Подтверждение экспериментальной цели .....	56
ЗАКЛЮЧЕНИЕ .....	60
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	61

## СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

ID – идентификатор

ЧСС – частота сердечных сокращений

API – программный интерфейс приложения

IDE - Integrated Development Environment

UML - Unified Modeling Language

ML - Machine Learning

## ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Атрибут – характеристика сущности базы данных.

Соматотропный гормон – один из гормонов передней доли гипофиза, по-другому называется гормон роста.

Хронический недосып – это расстройство сна, от которого страдают миллионы людей во всем мире, характеризующееся трудностями с засыпанием, продолжением сна или пробуждением с ощущением отдохнувшего состояния.

Гигиена сна – это комплекс мероприятий, предназначенный для обеспечения полноценного здорового сна. Это понятие включает период для подготовки ко сну и правильно организованное спальное место.

Психоактивные вещества – вещества или смесь нескольких веществ, влияющие на функции центральной нервной системы и приводящие к изменению психического состояния, вплоть до измененного состояния сознания.

Антидепрессанты – психотропные лекарственные средства, применяемые прежде всего для терапии депрессии, оказывающие влияние на уровень нейромедиаторов, в частности серотонина, норадреналина и дофамина.

Кортикостероиды – общее собирательное название подкласса стероидных гормонов.

Стероидные гормоны – группа физиологически активных веществ, регулирующих процессы жизнедеятельности у животных и человека.

Бета-блокаторы – подавляют действие гормона стресса адреналина и вещества-посредника норадреналина. Это приводит к снижению частоты сердечных сокращений.

Антибиотики – группа препаратов, которые используются в лечении бактериальных инфекций.

Обезболивающие – лекарственное средство природного, полусинтетического и синтетического происхождения, предназначенное для снятия болевых ощущений.

REM или быстрый сон – фаза сна, характеризующаяся повышенной активностью головного мозга. Одним из признаков этой фазы являются быстрые движения глазных яблок.

NREM или длительный сон – это самая медленная фаза человеческого сна. Именно во время глубокого сна происходит полноценное восстановление организма. Она делится на 4 фазы: N1, N2, N3, N4.

N1 или поверхностный сон – промежуточная стадия между засыпанием и бодрствованием. Мышечная активность снижается, снижается частота дыхания и пульса, замедляется обмен веществ и понижается температура, глаза могут совершать медленные движения.

N2 или легкий сон – дальнейшее снижение тонической мышечной активности. Сердечный ритм замедляется, температура тела снижается, глаза неподвижны. Занимает в целом около 45—55% общего времени сна.

N3 или медленный сон – стадия классифицируется как третья, если дельта-колебания занимают менее 50% и как четвёртая — если дельта составляет более 50%.

Дельта-колебания – видимый на электроэнцефалограмме высокоамплитудный медленный ритм головного мозга в диапазоне 0,5–4 Гц.

N4 или глубокий сон – самый глубокий медленный дельта-сон. В это время человека разбудить очень сложно — именно на этой стадии возможны приступы лунатизма, ночные ужасы, разговоры во сне и энурез у детей.

Долг по сну – накопленный недостаток сна, который представляет собой совокупный эффект недостающего отдыха.

Апноэ – это приостановка дыхания.

## ВВЕДЕНИЕ

Полноценный сон необходим любому живому существу на планете, в том числе и человеку. Он является важным аспектом человеческой жизни, и играет решающую роль в физическом и психическом благополучии. Именно благодаря ему мы восстанавливаем ресурсы организма, которые затрачиваются в течение дня.

При всей важности такой базовой потребности, как сон, именно в наш век современных технологий большая часть населения Земли подвержена хроническому недосыпу. Множество факторов, таких как стресс, тревога, нерегулярный график сна, злоупотребление психоактивными веществами, возраст или лекарства, могут способствовать развитию бессонницы [1].

При этом, многие люди, столкнувшиеся с подобной проблемой, не обладают пониманием деталей, которые влияют на их качество сна. Любому взрослому человеку важно спать не только от 7 до 9 часов [2], соблюдая гигиену сна, но и организовать пробуждение во время REM или N1 [3] стадии сна, ведь именно такие факторы оказывают значительное влияние на то, как вы будете себя чувствовать на протяжении всего дня.

Следовательно, оптимальная продолжительность сна, соблюдение гигиены сна, и пробуждение на определенной фазе сна, имеют решающее значение для улучшения общего качества сна. Воспользовавшись этими правилами, любой больной человек может улучшить качество своего сна, начав со временем просыпаться, ощущая себя отдохнувшим и полностью восставшим энергию.

Сон представляет собой сложный процесс, в ходе которого в организме человека происходят многочисленные изменения биометрических показателей, связанных с метаболическими функциями. В этот период наблюдаются колебания таких параметров, как скорость и глубина дыхания, а также изменение сердечного ритма и артериального давления. Такие

изменения предоставляют возможность для мониторинга и анализа физиологических процессов, происходящих в организме во время сна.

Современный мир сделал огромный шаг в развитии технологий. Теперь почти каждый пользователь обладает не только компьютером, телефоном, но и умными часами. Благодаря такой распространенности этих компактных и многофункциональных электронных устройств становится возможно отслеживать биометрические данные человека в режиме реального времени. Это дает возможность точно определять физиологические процессы в организме и предоставлять индивидуальные советы пользователю, например для улучшения качества сна.

Таким образом, целью данной работы становится создание приложения по нормализации режима сна на основе частоты сердечных сокращений.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Исследовать предметную область для выявления факторов, влияющих на качество сна, чтобы лучше понять информацию о предмете исследования.
2. Выбрать несколько наиболее популярных существующих решений и проанализировать их достоинства и недостатки.
3. Разработать архитектуру базы данных для улучшения производительности и обеспечения целостности данных.
4. Разработать архитектуру приложения, для распределения функциональности между компонентами системы.
5. Реализовать приложение на основе ранее полученной информации.
6. Провести тестирование приложения для проверки функциональности и повышения качества продукта.

# Глава 1. Анализ предметной области и обзор существующих решений

## 1.1 Анализ предметной области

Сон является естественным физиологическим процессом, который необходим для нормального функционирования человеческого организма. В современном мире сон становится все более проблематичным для многих людей, что приводит к хроническому недосыпу и негативно влияет на здоровье и активность человека. В связи с этим, понимание предметной области сна является крайне важным для выяснения факторов, влияющих на качество сна и человеческий организм в целом.

Регулярный сон способствует отдыху и восстановлению, что положительно влияет на память, концентрацию и психическое состояние человека. Более того, сон может снизить риск развития различных заболеваний, таких как диабет, ожирение и сердечно-сосудистые заболевания [4].

Но польза сна не ограничивается только этими аспектами. Во время сна наш организм вырабатывает гормон - соматотропин, который является ключевым для активного питания клеток организма. Этот гормон ускоряет процесс деления клеток и помогает восстанавливаться костной ткани, мышцам и иммунной системе [5].

Кроме того, пробуждение в правильной фазе сна имеет множество преимуществ, оказывая значительное влияние на то, как вы будете себя чувствовать на протяжении всего дня. Просыпаясь во время более легких стадии сна, таких как REM или N1, вы избегаете сонливости и дезориентации, чувствуя себя более отдохнувшим и бодрым [3]. Именно это улучшить ваше настроение, уровень энергии и продуктивность на протяжении всего дня.

С другой стороны, когда мы пробуждаемся в более глубокой стадии сна, такой как N2, N3, N4, то наше тело еще находится в процессе восстановления и регенерации, поэтому прерывание этого процесса негативно сказывается на



нашем здоровье и настроении. Чувство вялости после такого пробуждения может сопровождаться головной болью, нарушением координации движений и снижением эффективности работы мозга [3]. Более того, такие пробуждения могут привести к усилению стресса и беспокойства, что может дополнительно ухудшить качество сна и привести к бессоннице.

Помимо этого, сон — это сложный феномен, сопровождающийся значительными изменениями биометрических показателей, которые тесно связаны с метаболическими процессами в организме человека. Во время сна наступают фазы, каждая из которых характеризуется своими геофизическими параметрами. Так во время медленного сна, дыхание становится более регулярным, и сердечный ритм замедляется, это время, когда тело активно восстанавливается. С другой стороны, во время фазы быстрого сна, наблюдаются более быстрый сердечный ритм, а дыхание может стать поверхностным и нерегулярным. Понимание изменений этих биометрических показателей на разных стадиях сна дает возможность для мониторинга и анализа физиологических процессов, происходящих в организме во время сна.

Как говорилось ранее, недостаточный или некачественный сон может быть связан с рядом проблем.

Нерегулярное время засыпания и пробуждение может нарушить внутренний цикл “сна-бодрствования” организма и привести к недосыпу. А также негативно сказаться на качестве сна - затрудняя возможность заснуть, продолжительность сна и приятное пробуждение. Нерегулярный график сна может быть вызван целым рядом факторов: занятия поздней ночью, смена часовых поясов, сменная работа или дневной сон. Лучшим решением будет установление времени отхода ко сну и пробуждения, стараясь придерживаться последовательного графика сна, даже по выходным. Также стоит избегать дремоты в течение дня, максимум 30 мин, и создать расслабляющие ритуалы перед сном и обстановку, благоприятствующую сну.

Режим сна может меняться с годами, и многие пожилые люди часто испытывают нарушение сна. Существует несколько возрастных факторов, которые могут способствовать нарушениям сна: лекарства, гормональные изменения, снижение физической активности, болезни, появляющиеся с возрастом. Решение в такой ситуации следующее: поддержка графика сна, регулярная физическая активность, избегание употребления психоактивных веществ или употребление снотворных препаратов.

Лекарства могут иметь побочные эффекты, которые приводят к бессоннице. Некоторые из наиболее распространенных лекарств, которые могут вызвать нарушение сна — это антидепрессанты, кортикостероиды, бета-блокаторы, антибиотики, обезболивающие. К сожалению, если у вас возникают проблемы со сном в связи с применением медикаментов, важно обсудить это с лечащим врачом.

В заключении сон играет критическую роль в поддержании здоровья и благополучия человека. Регулярный и качественный сон способствует улучшению памяти, концентрации, психического состояния, а также общего физического здоровья. Наряду с этим, важность правильного пробуждения не может быть недооценена, поскольку пробуждение в неподходящую фазу сна может негативно влиять на настроение, уровень энергии и производительность человека в течение дня. Учитывая все вышеизложенное, можно утверждать, что вопросы регулярности, длительности, в зависимости от возраста, и качества сна являются ключевыми в контексте поддержания здорового образа жизни.

## 1.2 Определение особенностей существующих решений

В текущий момент на рынке представлено несколько различных решений для нормализации режима сна. Рассмотрим несколько и выделим основные функциональные особенности каждого.

### 1.2.1 Особенности решения “Sleep as Android”

Приложение “Sleep as Android” имеет обширный и полезный функционал, который перечислен ниже:

- Наличие умного будильника
- Приложение производит проверку подъема пользователя с помощью повторного сигнала будильника
- Сбор статистики сна, а именно: общее время сна, время глубокого сна, эффективность сна, время храпа во время сна, частота дыхания, регулярность соблюдения режима
- Предоставление записей храпа во время сна
- Предоставление графика фаз сна
- Установка целей сна
- Возможность разбудить пользователя, если он храпит
- Возможность поставить колыбельную на ночь
- Резервное копирование данных приложения
- Кроссплатформенное приложение
- Отслеживание фаз сна с помощью акселерометра встроенного в телефон

### 1.2.2 Особенности решения “Sleep Cycle”

Приложение “Sleep Cycle” имеет обширный и полезный функционал, который перечислен ниже:

- Наличие умного будильника
- Возможно установить всего 2 будильника
- Приложение производит проверку подъема пользователя с помощью повторного сигнала будильника
- Анализ сна без будильника
- Сбор статистики сна, а именно: общее время сна, время глубокого сна, эффективность сна, время храпа во время сна, частота

дыхания, время засыпания, время подъема, регулярность соблюдения режима

- Предоставление графика фаз сна
- Предоставление записей храпа во время сна
- Установка целей сна
- Установка примечаний ко сну
- Резервное копирование настроек приложения
- Удаление записей сна спустя неделю
- Кроссплатформенное приложение
- Возможность поставить колыбельную на ночь
- Отслеживание фаз сна с помощью микрофона встроенного в телефон

### 1.2.3 Особенности решения “Pillow”

Приложение “Pillow” имеет обширный и полезный функционал, который перечислен ниже:

- Наличие умного будильника
- Сбор статистики сна, а именно: общее время сна, время этапов сна, эффективность сна, время храпа во время сна, частота дыхания, максимальная ЧСС, минимальная ЧСС, средняя ЧСС.
- Предоставление графика фаз сна
- Резервное копирование настроек приложения
- Резервное копирование статистики сна
- Предоставление информации по улучшению сна
- Только для IOS
- Отслеживание фаз сна с помощью микрофона и/или акселерометра, встроенного в телефон и/или пульсометра встроенного в часы

#### 1.2.4 Особенности решения “Better Sleep”

Приложение “Better Sleep” имеет обширный функционал, который перечислен ниже:

- Возможно установить всего 1 будильник
- Сбор статистики сна, а именно: общее время сна, время храпа во время сна, частота дыхания, лучшее время сна
- Предоставление записей храпа во время сна
- Возможность поставить колыбельную на ночь
- Возможность поставить историю на ночь
- Возможность поставить звуки медитации на ночь
- Возможность поставить музыку на ночь
- Возможность поставить фильм на ночь
- Возможность поставить дыхание на ночь
- Кроссплатформенное приложение

#### 1.2.5 Сравнение и анализ используемых технологий

В процессе изучения особенностей существующих решений мной были выделены 3 способа отслеживания фаз сна:

- Технология использования акселерометра, встроенного в телефон, работает по следующему принципу - когда вы спите, ваше тело проходит через различные фазы сна, каждая из которых характеризуется разными уровнями активности, так во время фазы REM, ваши глаза и дыхательные мышцы активно двигаются, в то время как большинство остальных мышц тела остаются парализованными. В других фазах сна уровень физической активности снижается [6]. Данная технология является одним из трех способов анализа фаз сна человека и обладает рядом достоинств:

- Почти каждый человек на планете обладает смартфоном, и нет необходимости в дополнительном оборудовании для отслеживания фаз сна.
- Любой современный телефон обладает акселерометром и получение информации с него требует минимальных затрат.
- Пользователю просто нужно положить телефон рядом с собой на кровать перед сном, заранее зайдя в приложение и ничего более не нужно делать.

Но также и обладает очень весомыми недостатками:

- Акселерометры в смартфонах не так точны, как специализированные устройства для отслеживания сна [6]. Они могут не точно определять фазы сна, особенно REM-фазу.
- Наличие смартфона в кровати во время сна может причинять ряд неудобств для некоторых людей.
- Телефон не предоставляет возможности отслеживать другие биометрические данные пользователя.
- Искажение данных из-за наличия любого другого живого существа в кровати.
- Технология использования микрофона, встроенного в телефон, работает по следующему принципу - различные фазы сна могут проявляться через разные звуковые профили. Так в фазе REM сна, человек может издавать различные звуки, это может включать в себя речь во сне, храп или даже внезапные крики или вздохи. В фазе N3, человек, как правило, не двигается и издает очень мало звуков. Храп может происходить в любой стадии сна, он часто усиливается во время REM сна из-за расслабления мышц горла и языка [7]. Данная технология является еще одним способом

анализа фаз сна человека и обладает рядом достоинств, аналогичных технологии использования акселерометра:

- Почти каждый человек на планете обладает смартфоном, и нет необходимости в дополнительном оборудовании.
- Пользователю просто нужно положить телефон рядом с кроватью перед сном, заранее зайдя в приложение.
- Приложение при помощи микрофона может отслеживать звуки дыхания и/или храпа во время сна, что может помочь выявить потенциальные проблемы с апноэ.

Но также и обладает весомыми недостатками:

- Микрофон смартфона не всегда предоставляет точную информацию. Это может быть связано с рядом факторов, таких как качество микрофона или чистота микрофонного отверстия.
  - Любые внешние шумы могут исказить данные, которые собирает микрофон.
  - Микрофон телефона должен располагаться точно на уровне головы пользователя и не должен менять положение со временем.
  - Телефон не предоставляет возможности отслеживать другие биометрические данные пользователя.
- Технология использования пульсометра, встроенного в часы, работает на основе изменений в сердечном ритме во время сна. Количество сердечных сокращений в минуту, может меняться в зависимости от фазы сна, так во время бодрствования и фазы сна REM, когда мозг наиболее активен, ваш сердечный ритм обычно будет выше [8]. Во время более глубоких фаз сна, таких как N3, сердечный ритм обычно снижается. Такая технология обладает рядом достоинств:

- Пульсометры в умных часах могут отслеживать частоту сердечных сокращений в режиме реального времени, что может обеспечить непрерывное отслеживание сна.
- Пульсометры в часах могут быть более точными, чем акселерометры и микрофоны в смартфонах, особенно при определении различных фаз сна.
- Умные часы также могут отслеживать другие биометрические данные, которые могут помочь пользователям лучше понять свои паттерны сна.

Потенциальными недостатками этого способа можно считать такие аспекты, как цена умных часов и необходимость их регулярной зарядки. Однако, эти факторы малозначительны и их можно не принимать во внимание.

Таким образом сделать вывод, что наиболее точным, удобным и доступным способом из перечисленных является метод отслеживания фаз сна с помощью умных часов.

#### 1.2.6 Сравнение и анализ существующих решений

Таким образом можно составить таблицу с преимуществами и недостатками каждого из существующих решений, которые рассматривались ранее (Таблица 1). Нашим оптимальным выбором будет приложение, которое предлагает наиболее точное отслеживание фаз сна, то есть через использование умных часов. Этот функционал представлен только в приложении Pillow. Однако, стоит отметить, что данное приложение работает исключительно на платформе IOS и совместимо только с умными часами Apple Watch, что ограничивает его возможности эксплуатации для владельцев, использующих устройства Android или другие умные часы.



Таблица 1, Сравнительный анализ существующих решений

	Sleep as Android	Sleep Cycle	Pillow	Better Sleep
Наличие умного будильника	✓	✓	✓	×
Кроссплатформенность	✓	✓	×	✓
Метод отслеживания фаз на основе акселерометра				
Доступность	✓	-	✓	-
Простота интеграции	✓	-	✓	-
Простота использования	✓	-	✓	-
Точность	×	-	×	-
Удобство использования	×	-	×	-
Возможность искажения данных	✓	-	✓	-
Ограниченное количество функционала	✓	-	✓	-
Метод отслеживания фаз на основе микрофона				
Доступность	-	✓	✓	-
Простота интеграции	-	✓	✓	-
Простота использования	-	✓	✓	-
Точность	-	×	×	-
Удобство использования	-	×	×	-
Возможность искажения данных	-	✓	✓	-
Ограниченное количество функционала	-	✓	✓	-
Метод отслеживания фаз на основе частоты сердечных сокращений				
Доступность	-	-	✓	-
Простота интеграции	-	-	×	-
Простота использования	-	-	✓	-
Точность	-	-	✓	-
Удобство использования	-	-	✓	-
Возможность искажения данных	-	-	✓	-
Ограниченное количество функционала	-	-	×	-

## Глава 2. Определение требований к системе и разработка архитектуры приложения

### 2.1 Определение требований к системе

После обзора существующих решений и перед разработкой системы по нормализации режима сна нужно сформировать список требований, которым должна будет отвечать разрабатываемая система.

Для начала создадим Use-case диаграмму, которая продемонстрирует набор функциональности, доступный каждому типу пользователей (Рисунок 1).



Рисунок 1, Use-case диаграмма приложения

Для неавторизованного пользователя будут представлены всего 3 функциональные возможности: регистрация в системе, вход в систему по имени и паролю, сброс старого пароля.

Для авторизованного пользователя будет предоставлен более обширный функционал: вход в гугл сервис для возможности считывания ЧСС, выход из гугл сервиса для возможности изменения аккаунта, изменение персональных данных, которые были указаны при регистрации, изменение языка локализации, оценка последнего сна пользователя, отправка письма в поддержку при возникновении каких-либо ошибок со стороны приложения, установка умного будильника и повторная проверка подъема пользователя, добавление и удаление друзей для возможности просмотра их промежутков сна, добавление, удаление и изменение данных о промежутке сна, если сон был не записан в связи со снятием часов на время сна.

Теперь следует провести оценку всех выявленных прецедентов, определив время, необходимое для реализации каждого требования, установив приоритеты и определив стабильность работы для каждого требования (Таблица 2, Таблица 3).

Для измерения объема работ была выбрана стандартная единица измерения - человеко-часы.

*Таблица 2, Описание функциональных требований к системе*

№	Требование	Приоритет	Трудоемкость	Стабильность
1	2	3	4	5
Функциональные требования (Неавторизованный пользователь)				
FR1	Система должна предоставлять возможность регистрации	Must have	8	Высокая
FR2	Система должна предоставлять возможность авторизоваться	Must have	3	Высокая
FR3	Система должна предоставлять возможность обновления пароля	Must have	4	Высокая

1	2	3	4	5
Функциональные требования (Авторизованный пользователь)				
FR4	Система должна предоставлять возможность выхода из аккаунта	Must have	2	Высокая
FR5	Система должна предоставлять возможность создания данных о промежутке сна	Must have	4	Высокая
FR6	Система должна предоставлять возможность редактирования данных о промежутке сна	Must have	2	Высокая
FR7	Система должна предоставлять возможность удаления данных о промежутке сна	Must have	2	Высокая
FR8	Система должна предоставлять возможность оценки пробуждения пользователя	Must have	3	Высокая
FR9	Система должна предоставлять возможность входа в аккаунт Google Fit	Must have	18	Средняя
FR10	Система должна предоставлять возможность выхода из аккаунта Google Fit	Must have	2	Средняя
FR11	Система должна предоставлять возможность корректировки информации о себе	Could have	2	Низкая

1	2	3	4	5
FR12	Система должна предоставлять возможность выбора языка для локализации	Could have	8	
FR13	Система должна предоставлять возможность отправить письмо технической поддержке с жалобой	Could have	2	Высокая
FR14	Система должна предоставлять возможность установки умного будильника для пользователя	Must have	18	Высокая
Функциональные требования (Авторизованный пользователь)				
FR15	Система должна предоставлять возможность отображение данных о сне на графиках и диаграммах для анализа	Could have	8	Средняя
FR16	Система должна предоставлять возможность просмотра прогресса по сну за последние 2 недели	Must have	2	Высокая

Таблица 3, Описание нефункциональных требований к системе

№	Требование	Приоритет	Трудоемкость (человеко- часы)	Стабильность
1	2	3	4	5
Нефункциональные требования				
NFR1	Интерфейс должен корректно работать во всех современных браузерах	Must have	-	Высокая
NFR2	Интерфейс должен корректно работать на всех современных телефонах	Must have	-	Высокая
NFR3	Интерфейс должен иметь адаптивную верстку для удобства пользования	Must have	-	Высокая
NFR4	Сайт должен корректно работать на пиковых нагрузках	Should have	-	Высокая
NFR5	Сайт должен быть локализован для английского и русского языка	Should have	-	Средняя
NFR6	Высокая производительность и быстрый отклик приложения	Could have	-	Высокая
NFR7	Легкий интерфейс пользователя и интуитивно понятная навигация	Won't have	-	Высокая

Проанализировав представленную выше таблицу, можно сделать вывод, что в первую версию приложения необходимо будет включить требования FR1 - FR10, FR14, FR16. Именно эти требования будут составлять каркас приложения по нормализации режима сна и позволять предоставить пользователю минимальный жизнеспособный продукт. Таким образом

создание первой версии приложения будет включать в себя 12 требований и займет 68 часа, что эквивалентно 9 дням работы.

Во 2 версию приложения войдут требования, способствующие оптимизации работы системы и повышению удобства ее использования, а именно требования FR11-FR13, FR15. Время необходимое на добавление данного функционала в уже работающий проект составит 20, что равносильно 3 рабочим дням.

Таким образом разработка самого проекта займет 88 часов. Также стоит добавить минимум 16 часов на тестирование данной системы и исправление ошибок, выявленных в процессе тестирования. Помимо этого, подсчет человеко-часов наивным методом является достаточно неточным способом, поэтому увеличение затрат по времени всего проекта на 25% вероятно даст более правдоподобные расходы, то есть 130 человеко-часа.

## 2.2 Разработка архитектуры базы данных

После проведения детального анализа всех требований, предъявляемых к системе, можно переходить к стадии формирования ее архитектуры. Первоочередным шагом в разработке приложения является конструирование архитектуры базы данных, потому что она служит ключевым стержнем для любого приложения. Этот процесс должен отличаться основательностью и глубоким проработанным подходом. На данном этапе определяются модели данных, необходимые для корректной работы приложения, и предоставляется готовый каркас для дальнейшей модификации.

В контексте разработки нашего приложения, стоит подчеркнуть, что требования к хранению данных не являются излишне обширными или сложными. Для эффективной работы приложения необходима лишь небольшая база данных, которая будет хранить данных в диапазоне от нескольких сотен до нескольких десятков тысяч записей (Рисунок 2).

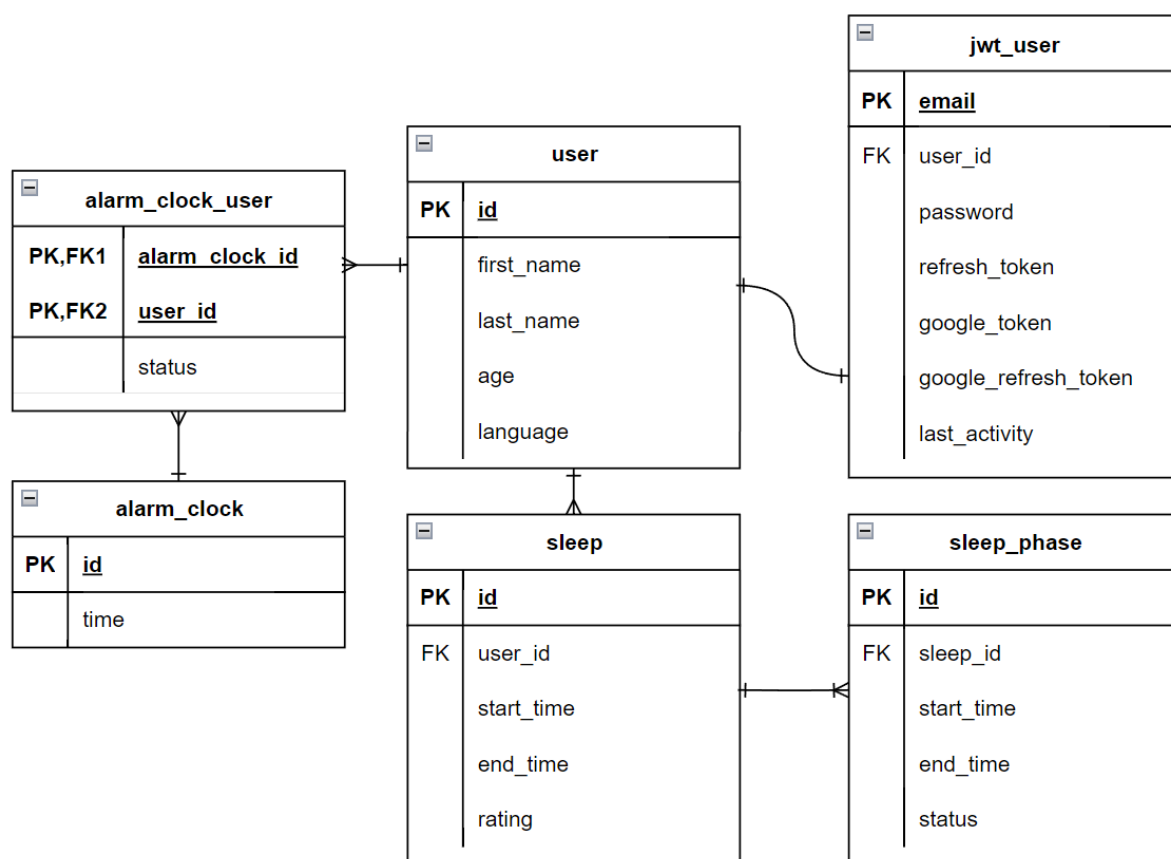


Рисунок 2, Архитектуры базы данных

Архитектура базы данных включает в себя 6 таблиц.

- `jwt_user` - таблица для хранения информации, необходимой для создания и обработки токенов приложения, а также взаимодействия с Google Fit API (Таблица 4).

Таблица 4, Структура таблицы `jwt_user`

Имя поля	Тип данных	Описание предназначения
1	2	3
email	varchar(255)	Первичный ключ
user_id	integer	Внешний ключ на первичный ключ таблицы user.
password	varchar(255)	Пароль пользователя от аккаунта

Продолжение таблицы 4



1	2	3
refresh_token	text	Токен обновления для приложения
google_token	text	Токен доступа для Google Fit API
google_refresh_token	text	Токен обновления для Google Fit API
last_activity	timestamp	Дата и время последнего входа в аккаунт

- user - таблица для хранения информации о пользователе приложения (Таблица 5).

Таблица 5, Структура таблицы user

Имя поля	Тип данных	Описание предназначения
1	2	3
id	serial	Первичный ключ
first_name	varchar(255)	Имя пользователя
last_name	varchar(255)	Фамилия пользователя
age	integer	Возраст пользователя
language	varchar(255)	Язык используемый для локализации

- sleep - таблица для хранения информации о сне пользователя (Таблица 6).

Таблица 6, Структура таблицы sleep

Имя поля	Тип данных	Описание предназначения
id	serial	Первичный ключ
user_id	integer	Внешний ключ на первичный ключ таблицы user.
start_time	timestamp	Дата и время начала сна пользователя
end_time	timestamp	Дата и время окончания сна пользователя
rating	integer	Оценка сна пользователем

- sleep\_phase - таблица для хранения информации о фазах сна (Таблица 7).

Таблица 7, Структура таблицы sleep\_phase

Имя поля	Тип данных	Описание предназначения
id	serial	Первичный ключ
sleep_id	integer	Внешний ключ на первичный ключ таблицы sleep.
start_time	timestamp	Дата и время начала фазы сна
end_time	timestamp	Дата и время окончания фазы сна
status	varchar(255)	Наименование фазы сна

- alarm\_clock\_user - ассоциативная таблица для таблиц user и alarm\_clock (Таблица 8).

Таблица 8, Структура таблицы alarm\_clock\_user

Имя поля	Тип данных	Описание предназначения
alarm_clock_id	integer	Первичный ключ. Внешний ключ на первичный ключ таблицы alarm_clock
user_id	integer	Первичный ключ. Внешний ключ на первичный ключ таблицы user
status	boolean	Статус будильника, который подразумевает значение включен или выключен

- alarm\_clock - таблица для хранения информации о времени пробуждения (Таблица 9).

Таблица 9, Структура таблицы alarm\_clock

Имя поля	Тип данных	Описание предназначения
id	serial	Первичный ключ
time	time	Время будильника

## 2.3 Разработка архитектуры приложения

После определения архитектуры базы данных можно приступить к созданию архитектуры самого приложения (Рисунок 4).

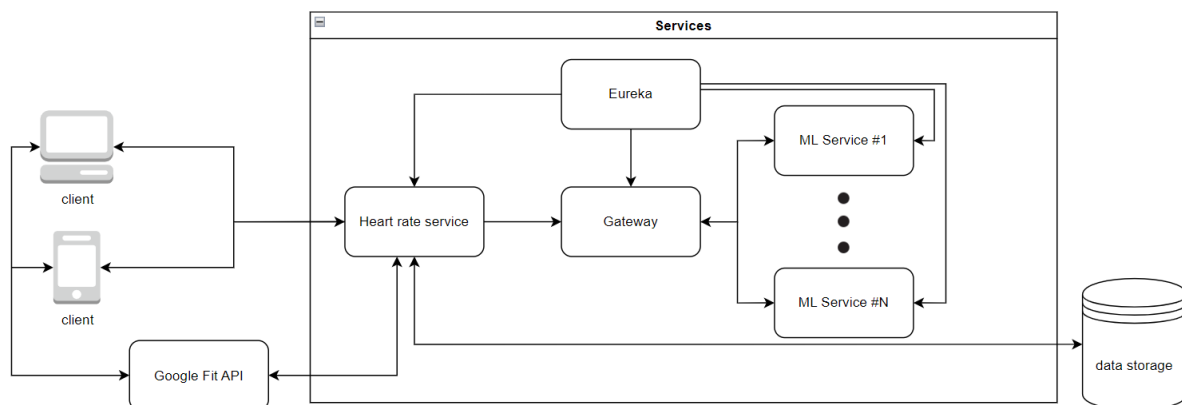


Рисунок 4, Схема приложения

Приложение имеет классическую клиент-серверную архитектуру.

В качестве клиента выступает веб-интерфейс, который может быть запущен на любом электронном устройстве с установленным браузером, будь то компьютер или телефон.

В центре архитектуры находятся 4 модуля:

- Модуль частоты сердечных сокращений
- Модуль Gateway
- Модуль машинного обучения
- Модуль Eureka

Большая часть функционала приложения зависит от модуля частоты сердечных сокращений, ведь он выполняет ключевые функции в приложении и играет связующую роль между клиентом, модулем машинного обучения и базой данных. Только этот модуль имеет доступ к базе данных, и поэтому он единственный может создавать, удалять и изменять данные в ней. Помимо этого, данный модуль отвечает за генерацию токенов для пользователя и их валидацию, а также регистрацию пользователя, авторизацию, проверку роли пользователя и его прав соответственно. Таким образом он играет ключевую

роль в функционировании приложения и позволяет обеспечить безопасность и защиту пользовательских данных.

Модуль Gateway перехватывает все входящие запросы и отправляет их через систему управления API. Он выполняет множество необходимых функций, таких как проксирование запросов и балансировка нагрузки. Данный модуль позволяет обрабатывать большой объем запросов без задержек и сбоев в работе. Благодаря этому пользователи могут получать доступ к сервисам приложения надежно и без проблем, а разработчики могут легко управлять входящими запросами и контролировать нагрузку на систему.

Eureka выполняет ключевую функцию автоматической регистрации сервисов в системе. Это дает нам возможность мониторинга всех экземпляров сервиса машинного обучения, а также обеспечивает отслеживание состояния сервисов сердечной активности и Gateway. При этом, в случае обнаружения отказов, Eureka способна управлять этими ситуациями, обеспечивая бесперебойную работу системы.

Модуль машинного обучения играет важную роль в определении фазы сна пользователя и необходимости его пробуждения в определенный момент времени. Этот сервис является неотъемлемой частью обработки информации о пульсе пользователя и вычислению значений, необходимых для работы модуля частоты сердечных сокращений. Кроме того, вычисление этих значений является достаточно трудоемкой задачей, поэтому изначально планируется создать два экземпляра модуля, которые будут масштабироваться при необходимости.

Работа по взаимодействию с API Google service будет происходить на стороне модуля частоты сердечных сокращений при наличии токена авторизации для сервиса Google. В противном случае будет произведено перенаправление пользователя на сервис Google, где ему необходимо будет войти в аккаунт Google и дать согласие на обработку и предоставление информации модулю частоты сердечных сокращений.

Сам API Google service предоставляет набор сервисов, предлагаемых Google, включая API Google Fit, который позволяет разработчикам внедрять функциональность по отслеживанию здоровья в свои приложения. Google Fit API позволяет приложениям собирать данные с датчиков на устройствах пользователей, таких как акселерометры, гироскопы, сердечные ритмы и датчики шагов.

## 2.4 Выбор технологий для реализации

Данное приложение будет состоять из четырех основных компонента: интерфейс, серверный компонент, база данных и стороннего сервиса, предоставляющего информацию о ЧСС.

Для разработки веб-интерфейса будет использоваться язык Dart с фреймворком Flutter, который имеет открытый исходный код. Данный фреймворк позволяет использовать один и тот же код для разработки мобильных, веб и десктопных приложений, что может сэкономить время и ресурсы при необходимости кроссплатформенного перехода. Кроме того, Flutter предоставляет широкий выбор виджетов, которые упрощают создание кастомного пользовательского интерфейса. Также приложения, созданные с помощью Flutter, обычно работают очень быстро и плавно благодаря использованию собственного движка рендеринга, который компилирует код в нативный.

Разработка серверного компонента будет проводиться на языке Java с использованием фреймворка Spring, благодаря его широким возможностям, описанным далее. Spring Core, который является частью фреймворка, предоставляет контейнер внедрения зависимостей, который автоматически создает и связывает экземпляры класса, помеченные специальными аннотациями. Библиотеки, входящие в состав фреймворка, предоставляют механизмы для решения типовых задач, что сокращает время разработки и позволяет сосредоточиться на более специфических задачах. Кроме того,

фреймворк предоставляет множество инструментов для масштабирования приложений, таких как Spring Cloud, что позволяет легко создавать и поддерживать приложения больших масштабов.

Для управления базами данных была выбрана PostgreSQL - бесплатная объектно-реляционная система управления базами данных, с открытым исходным кодом. Она не имеет ограничений на размер базы данных или количество записей в таблице, обеспечивает надежность транзакций и предоставляет множество различных типов данных для использования, а также имеет широкую поддержку сообщества и наличие большого количества документов, что облегчает процесс разработки и поддержки приложений.

Для обучения моделей машинного обучения будет использоваться библиотека scikit-learn, которая предлагает широкий спектр алгоритмов. Одной из ключевых особенностей scikit-learn является её простота использования. Она предоставляет единообразный интерфейс для взаимодействия со всеми ее алгоритмами обучения, что делает процесс изучения и применения этих алгоритмов более простым и эффективным.

Взаимодействия между клиентом и сервером будет осуществляться через REST API, потому что данный подход позволяет легко масштабировать приложение и дает возможность обрабатывать различные типы запросов и форматы данных, ведь принципы REST предполагают присвоение каждому доступному API объекту (называемому ресурсом) уникального URL-адреса и использование HTTP метода для определения выполняемого действия над ресурсом при отправке запроса.

Передача данных будет осуществляться в формате JSON. Данный формат обладает необходимой гибкостью, которая проявляется в поддержке вложенные структуры данных, что позволяет передавать сложные объекты и иерархии. Также он является очень компактный формат, что позволяет передавать большие объемы данных за короткое время, и имеет хорошую поддержку во многих языках программирования и фреймворках.

## 2.5 Описание конечных точек сервисов

Поскольку был выбран подход REST для организации взаимодействия между клиентом и сервером, то структура запросов к серверу определяется на основе HTTP-запросов.

GET-запросы:

- `/api/v1/sleep/{id}` - получение информации по определенному промежутку сна пользователя (параметр `id` - уникальный идентификатор промежутка сна пользователя);
- `/api/v1/sleep/filter?start_datetime=start_datetime_value&end_datetime=end_datetime_value` - получение списка промежутков сна пользователя с информацией о них (параметры: `start_datetime_value` - дата и время, с которых нужно искать промежутки сна, `end_datetime_value` - дата и время, по которым нужно искать промежутки сна);
- `/api/v1/authentication/logout` - выход пользователя из аккаунта;
- `/api/v1/alarm_clocks/{id}` - получение информации по определенному будильнику пользователя (параметр `id` - уникальный идентификатор будильника);
- `/api/v1/alarm_clocks/filter?after_time=after_time_value&before_time=before_time_value&status=status_value` - получение списка будильников (параметры: `after_time_value` - время будильника больше заданного параметра, `before_time_value` - время будильника меньше заданного параметра, `status_value` - статус будильника);
- `/api/v1/alarm_clocks/wake_up` - получение информации по необходимости пробуждения пользователя;
- `/api/v1/sleep/upgrade/rating` - получение информации по общему улучшению сна пользователя;



### POST-запросы:

- `/api/v1/authentication/register` - регистрация пользователя (параметры: `password` - пароль пользователя, `email` - почта пользователя);
- `/api/v1/authentication/login` - вход пользователя в аккаунт (параметры: `email` - почта пользователя, `password` - пароль пользователя);
- `/api/v1/authentication/google/login` - вход пользователя в аккаунт google (параметры: `google_token` - токен доступа для Google Fit API, `google_refresh_token` - токен обновления для Google Fit API);
- `/api/v1/refresh/token` - обновление токенов пользователя (параметр `refresh_token` - токен обновления пользователя);
- `/api/v1/sleep` - создание промежутка сна (параметры: `start_datetime` - время и дата начала сна, `end_datetime` - время и дата окончания сна, `rating` - оценка промежутка сна);
- `/api/v1/alarm_clocks` - создать умный будильник (параметр `time` - время будильника, `status` - состояние будильника);
- `/api/v1/inform` - отправить письмо технической поддержке с жалобой (параметры: `title` - заголовок жалобы, `description` - текст жалобы);
- `/api/v1/calculate/sleep` - рассчитать фазу сна пользователя (параметры: `avg_heart_rate` - среднее ЧСС, `currently_heart_rate` - список ЧСС с последнего момента активности пользователя, `age` - возраст пользователя, `long_sleep_today` - время, которое пользователь спал за последние 24 часа, `remain_sleep_time` - время до будильника);

### PUT-запросы:

- `/api/v1/reset/password` - сброс текущего пароля (параметр `new_password` - новый пароль пользователя);

- `/api/v1/sleep/{id}` - изменение информации о промежутке сна (параметры: `id` - уникальный идентификатор промежутка сна пользователя, `start_datetime` - время и дата начала сна, `end_datetime` - время и дата окончания сна, `rating` - оценка промежутка сна);
- `/api/v1/sleep/rating/{id}` - оценка промежутка сна (параметры: `id` - уникальный идентификатор промежутка сна пользователя, `rating` - оценка промежутка сна);
- `/api/v1/users/information` - изменение информации о пользователе (параметры: `first_name` - имя пользователя, `last_name` - фамилия пользователя, `age` - возраст пользователя);
- `/api/v1/users/localization` - изменение локализации приложения (параметр `localization` - новый язык локализации);
- `/api/v1/alarm_clocks/{id}` - изменение параметров умного будильника (параметры: `id` - уникальный идентификатор будильника, `status` - новое состояние будильника);

#### DELETE-запросы:

- `/api/v1/sleep/{id}` - удаление информации о промежутке сна (параметры: `id` - уникальный идентификатор промежутка сна пользователя);
- `/api/v1/authentication/google/logout` - выход пользователя из аккаунта google;
- `/api/v1/alarm_clocks/{id}` - удаление умного будильника (параметры: `id` - уникальный идентификатор будильника);

## Глава 3. Реализация системы

### 3.1 Реализация heart\_rate сервиса

#### 3.1.1 Неавторизованный пользователь

В основе возможностей авторизации и аутентификации нашего сервиса лежит модуль Spring Boot Starter Security [9], входящий в состав проекта Spring Boot [10].

Чтобы начать пользоваться сервисом, необходима регистрация в системе. Процесс регистрации максимально упрощен: вам потребуется указать свою почту и задать пароль. После успешного завершения регистрации, в таблицах 'user' и 'jwt\_user' будет создана новая запись, представляющая ваш аккаунт. Кроме того, система автоматически генерирует token, сроком действия 24 часа, и refresh\_token. Схема способа регистрации представлен на рисунке ниже (Рисунок 5).

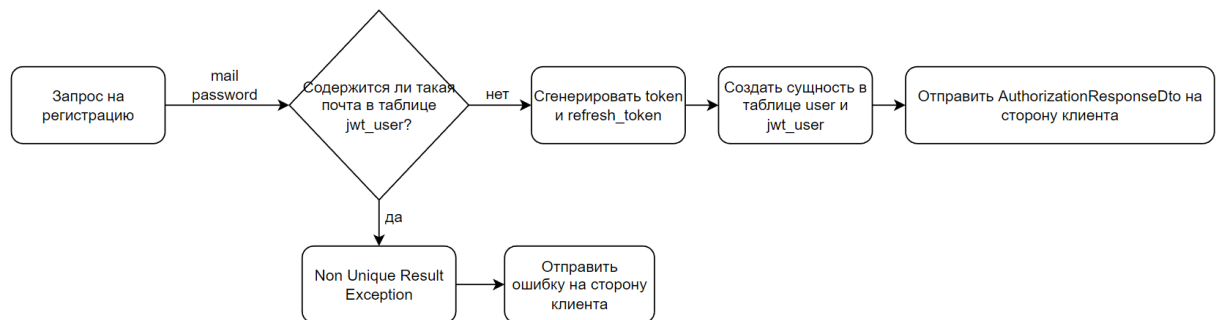


Рисунок 5, Схема регистрации пользователя

Если у пользователя уже есть зарегистрированный аккаунт, он может войти в систему. Это позволит системе идентифицировать пользователя и выдать ему token и refresh\_token для последующей аутентификации в сервисе.

В случае, если пользователь забывает свой пароль, система предлагает функцию восстановления доступа к аккаунту. Для этого пользователь должен предоставить почту, которую он использовал при регистрации. Эта информация проверяется на сервере, и в случае подтверждения существования указанного пользователя, на его почту отправляется ссылка с новым паролем.

Схема метода восстановления пароля представлен на рисунке ниже (Рисунок 6).

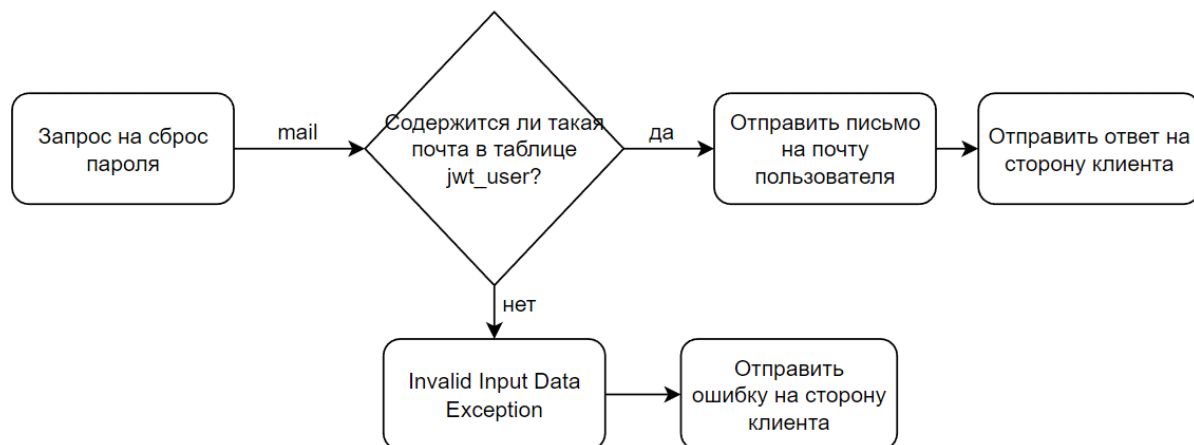


Рисунок 6, Схема сброса пароля пользователя

Процесс авторизации с использованием JWT токена работает по следующему принципу:

1. Сначала пользователь отправляет запрос на аутентификацию.
2. Затем сервер проверяет корректность введенных пользователем данных для входа (логин и пароль), и в случае их валидности генерирует токен.
3. С этого момента при каждом запросе к серверу, в заголовок этого запроса добавляется полученный на предыдущем шаге токен.
4. Получив запрос, сервер декодирует токен и на основе информации, содержащейся в нем, определяет, имеет ли пользователь права на выполнение запрашиваемой операции.

### 3.1.2 Авторизованный пользователь

После успешной авторизации пользователь получает доступ к большому числу функций, которые предоставляет приложение. Эти функции обеспечивают полноценное взаимодействие с системой, охватывая все основные аспекты работы. Остановимся и подробно рассмотрим алгоритмы

работы наиболее важных из этих функций, остальные функции представляют из себя не сложные CRUD операции.

Для корректной работы функции отслеживания сна и его фаз, пользователю потребуется авторизоваться в своем аккаунте Google Fit и предоставить нашему приложению разрешение на считывание данных о частоте сердечных сокращений. Важно понимать, что без принятия этого соглашения, приложение не сможет получать критически важную информацию для своей работы, поэтому пользователь будет перенаправляться на вкладку, где можно будет подключить Google Fit.

Процесс подключения включает перенаправление пользователя на сайт Google Fit, где он дает согласие на обработку своих биометрических данных. После успешного подтверждения клиентская часть получит `google_token` и `google_refresh_token` и отправит их на сервер. На стороне сервера они будут сохранены в таблице `jwt_user` и далее будут использоваться сервисом `heart_rate` для доступа к информации о частоте сердечных сокращений пользователя. Схема входа в аккаунт Google Fit представлен на рисунке ниже (Рисунок 7).

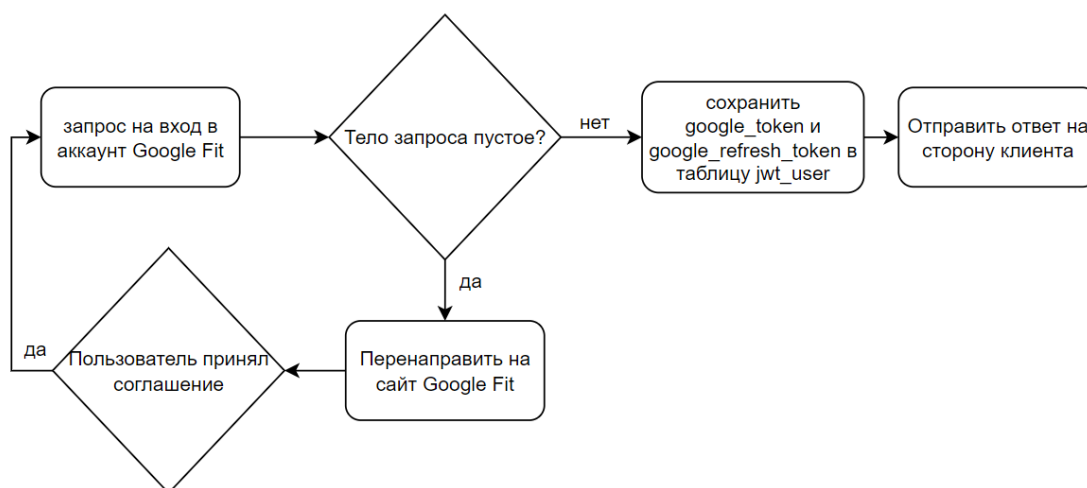


Рисунок 7, Схема входа пользователя в аккаунт Google Fit

Каждые три минуты в нашем приложении инициируется процесс, который определяет, есть ли необходимость в пробуждении пользователя. Этот процесс начинается с отправки запроса со стороны клиента на сервис

heart\_rate. Запрос обрабатывается и далее сервер пытается получить актуальные данные о частоте сердечных сокращений пользователя из Google Fit API, начиная с момента его последней активности. Сервер также получает информацию о средней частоте сердечных сокращений пользователя за весь день.

Собранные данные затем преобразуются в объект SleepRequestDto, который направляется в ML сервис. В свою очередь ML сервис анализирует каждую запись в списке частоты сердечных сокращений, используя для этого ранее обученную модель машинного обучения. В результате формируется SleepResponseDto, который отправляется обратно на сервис heart\_rate.

В heart\_rate сервисе данные сохраняются. В зависимости от значения в поле isNeedWakeUp, сервис может оповестить клиента о необходимости пробуждения пользователя. Схематичное изображение представлено на рисунке ниже (Рисунок 8).

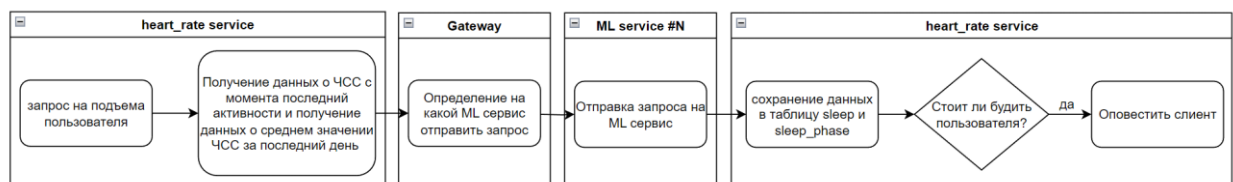


Рисунок 8, Схема необходимости пробуждения пользователя

Для более полного и наглядного представления об устройстве и функциональности разработанной системы, было решено дополнить текстовое описание визуальной составляющей. В этом нам поможет UML диаграмма классов, которая была построена с использованием IDE. Эта диаграмма позволит более детально и понятно ознакомиться с взаимосвязями и взаимодействием элементов нашей системы (Рисунок 9).

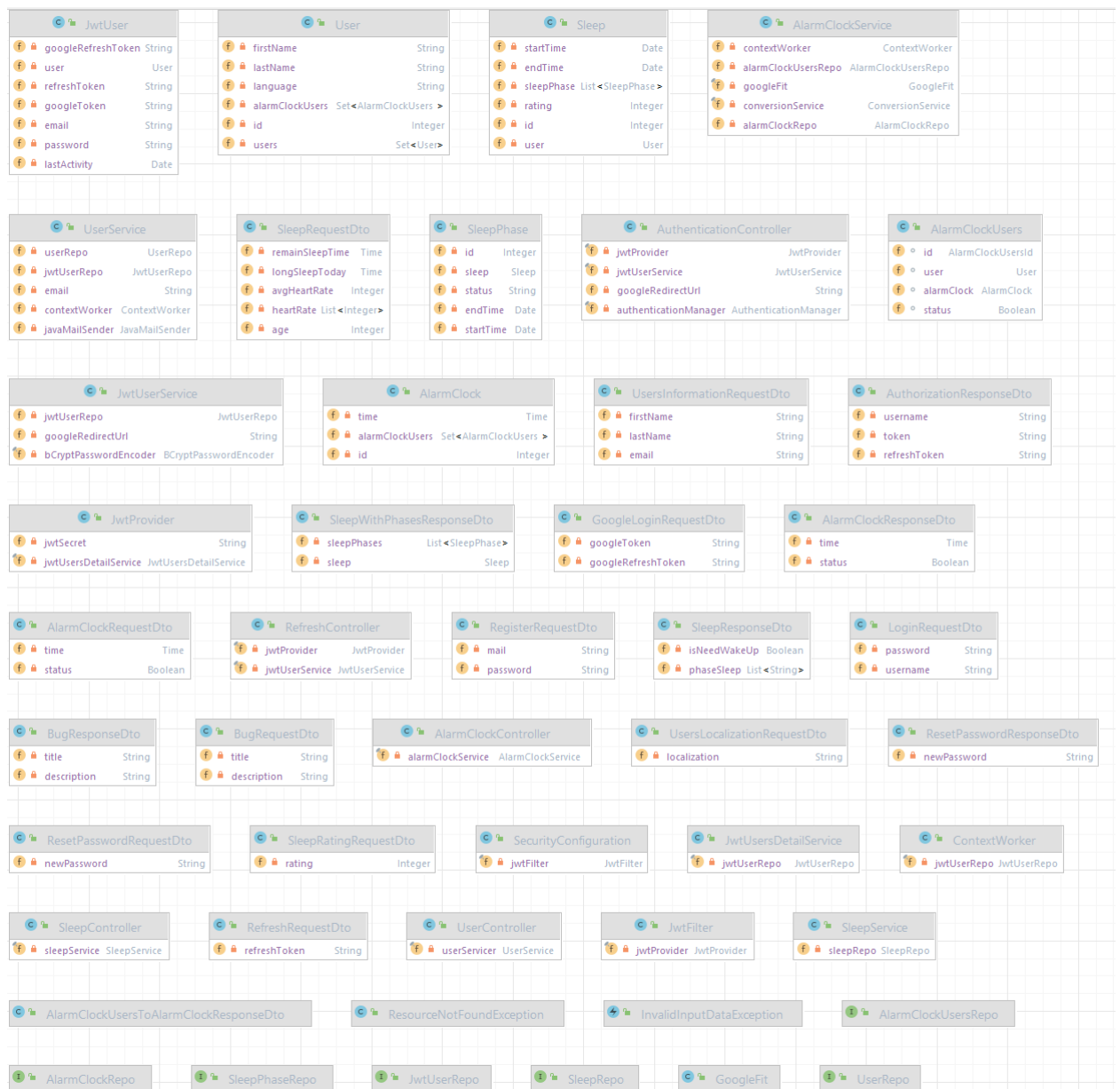


Рисунок 9, UML диаграмма классов сервиса heart\_rate

### 3.2 Реализация ML сервиса

Главной целью ML сервиса является анализ фаз сна пользователя и определение момента, когда его следует разбудить. Для выполнения данной задачи были обучены две модели машинного обучения.

Первая модель использует алгоритм машинного обучения “линейная регрессия”. Для обучения данной модели использовались данные пульса, его средние показатели за день и действующая фаза сна. Эти данные ранее были

собраны с помощью приложения Pillow. Все данные делятся в пропорции 60/20/20 для обучения, валидации и тестирования модели [11], сама модель была обучена с использованием библиотеки scikit-learn. После обучения модель была сериализована в файл linear\_regression.pt и добавлена в сервис ML для корректной работы приложения.

Вторая модель использует алгоритм машинного обучения под названием “логистическая регрессия”, для обучения данной модели использовались следующие данные: время начала сна, текущее время, время пробуждения и фаза сна, определенная при помощи прошлой модели. Распределение данных для обучения, валидации и тестирования соответствовало распределению, использованному для первой модели. Обучение также проводилось с помощью библиотеки scikit-learn и модель была сохранена в файл logistic\_regression.pt для дальнейшего использования сервисом ML.

Чтобы лучше понять структуру и функциональность разработанной системы хотелось бы предоставить описание не только в письменном виде, но и в визуальном, с этим нам поможет UML диаграмма классов построенная в IDE (Рисунок 10).



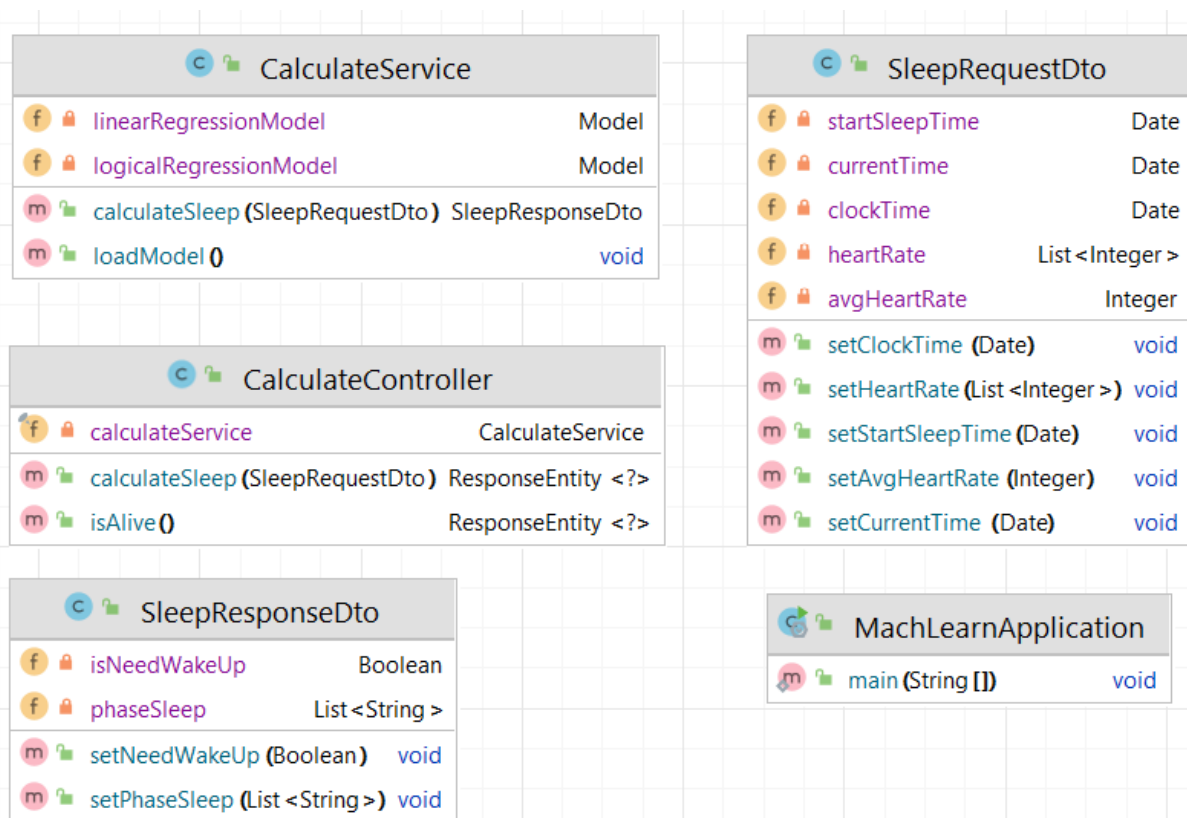


Рисунок 10, UML диаграмма классов сервиса ML

### 3.3 Реализация Eureka сервиса

Проект Spring Cloud [12] является основой для сервиса Eureka, который обеспечивает отслеживание всех экземпляров сервиса ML и контролирует стабильность работы сервиса heart\_rate и gateway. Благодаря Eureka, сервисы могут взаимодействовать без необходимости жесткого кодирования адресов хостов и портов. Еще одной преимущественной особенностью Eureka является её способность поддерживать функционирование сервисов, даже в случае недоступности самого Eureka-сервера — это достигается благодаря кэшированию информации на клиентской стороне. Более того, Eureka легко интегрируется с другими компонентами Spring Cloud.

В рамках моей работы был разработан отдельный модуль под названием Eureka, который представляет серверную часть service discovery. Этот модуль использует несколько зависимостей от Spring Cloud Netflix, и вся

конфигурация осуществляется через файл `bootstrap.yaml` и добавление аннотации `@EnableEurekaServer` в конфигурационный класс.

Для настройки клиентской стороны было необходимо выполнить несколько шагов. Во-первых, подключен модуль `Spring Cloud Netflix`. Затем, в файле `bootstrap.yaml`, обновлена соответствующая информация. Наконец, в конфигурационном классе использована аннотация `@EnableDiscoveryClient` для активации возможностей обнаружения сервиса.

### 3.4 Реализация Gateway сервиса

Проект `Spring Cloud` является основой для сервиса `Gateway`, который скрывает все сложности внутренней структуры ML сервиса и обеспечивает единую точку входа в систему, беря на себя необходимость маршрутизации.

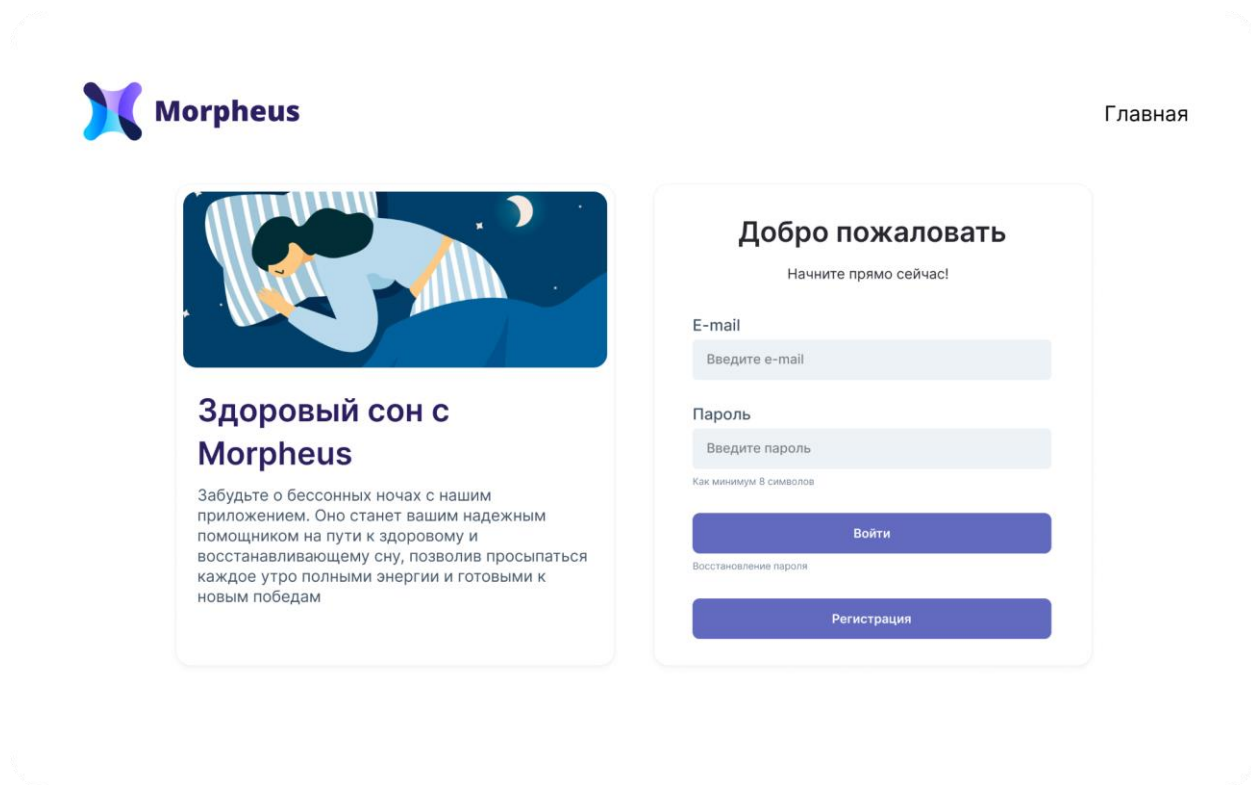
Одной из ключевых функций сервиса `Gateway` является балансировка нагрузки. В его основе лежит модуль `Spring Cloud Load Balancer`, который эффективно распределяет входящий трафик по различным экземплярам сервиса машинного обучения. Это гарантирует оптимальное использование ресурсов и повышает производительность системы.

В рамках выполнения данной работы, был создан отдельный модуль, который был назван `Gateway`. Для обеспечения его корректной работы, был подключен модуль `Spring Gateway` и проведена необходимая конфигурация через файл `application.yaml`.

### 3.5 Реализация клиентской части

Разработка удобного пользовательского интерфейса является ключевым моментом в обеспечении взаимодействия с серверной частью приложения. Фронтенд служит в качестве витрины для представления сложных данных, делая их понятными и доступными для пользователя. Такой подход позволяет обеспечить эффективное взаимодействие пользователя с приложением, упрощая восприятие информации.

Страница аутентификации является первой, с которой пользователь сталкивается при входе в приложение. У пользователя есть возможность войти в аккаунт, зарегистрироваться и сбросить пароль. Вы можете ознакомиться с ее образцом на приведенном ниже рисунке (Рисунок 11).



*Рисунок 11, Страница аутентификации*

При клике на кнопку “Регистрация” пользователя направляется на страницу регистрации. Вы можете ознакомиться с ее образцом на приведенном ниже рисунке (Рисунок 12).

### Создайте аккаунт

Это быстро и легко

E-mail


Пароль

Как минимум 8 символов

Зарегистрироваться

*Рисунок 12, Страница регистрации*

При клике на кнопку “Восстановления пароля” пользователя направляется на страницу восстановления пароля. Вы можете ознакомиться с ее образцом на приведенном ниже рисунке (Рисунок 13).



### Восстановление пароля

Введите e-mail и мы пришлем вам ссылку

E-mail

Восстановить пароль

*Рисунок 13, Страница восстановления пароля*

В случае успешной аутентификации пользователь перенаправляется на основную страницу приложения. Здесь собрана наиболее важная информация: установленные будильники, фазы последнего сна, изменение локализации, изменение темы интерфейса, а также оценка и длительность сна за каждый день недели. Вы можете ознакомиться с ее образцом на приведенном ниже рисунке (Рисунок 14).

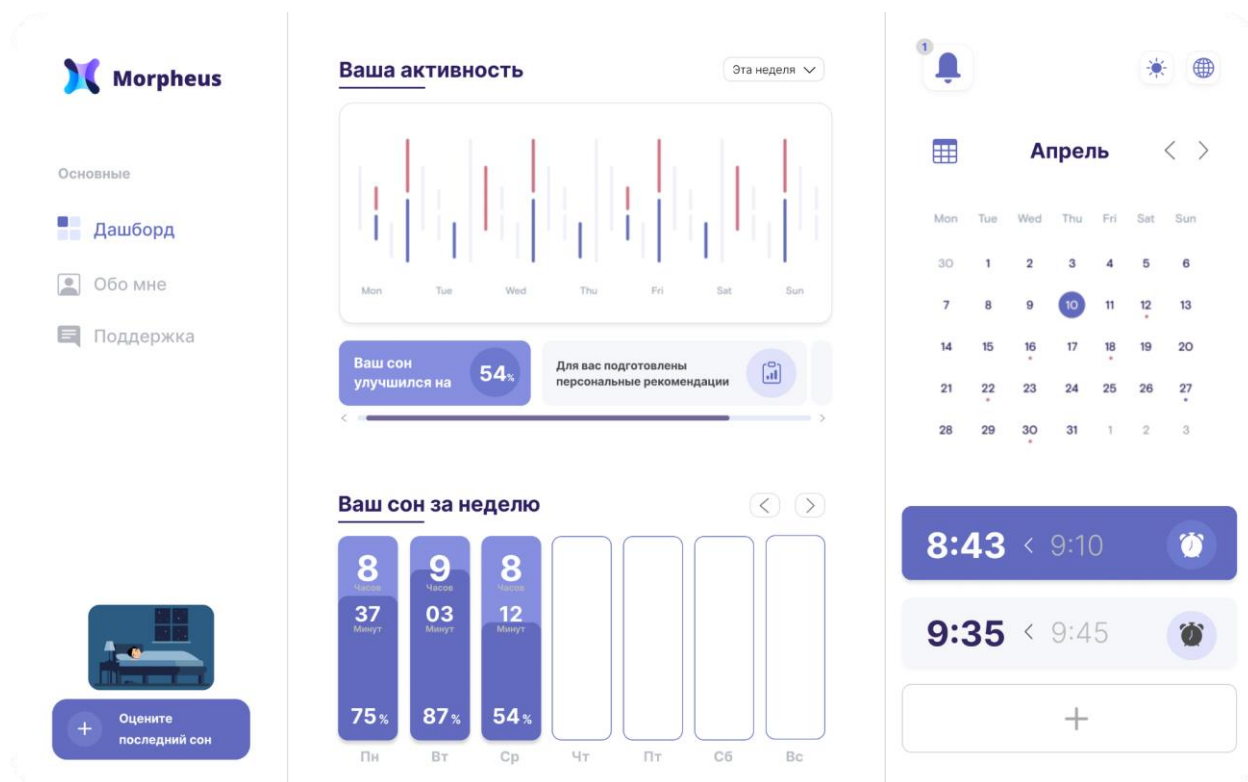


Рисунок 14, Основная страница приложения

При клике на кнопку изменения темы интерфейс пользователя инвертирует цвета, с примером вы можете ознакомиться на приведенном ниже рисунке (Рисунок 15).

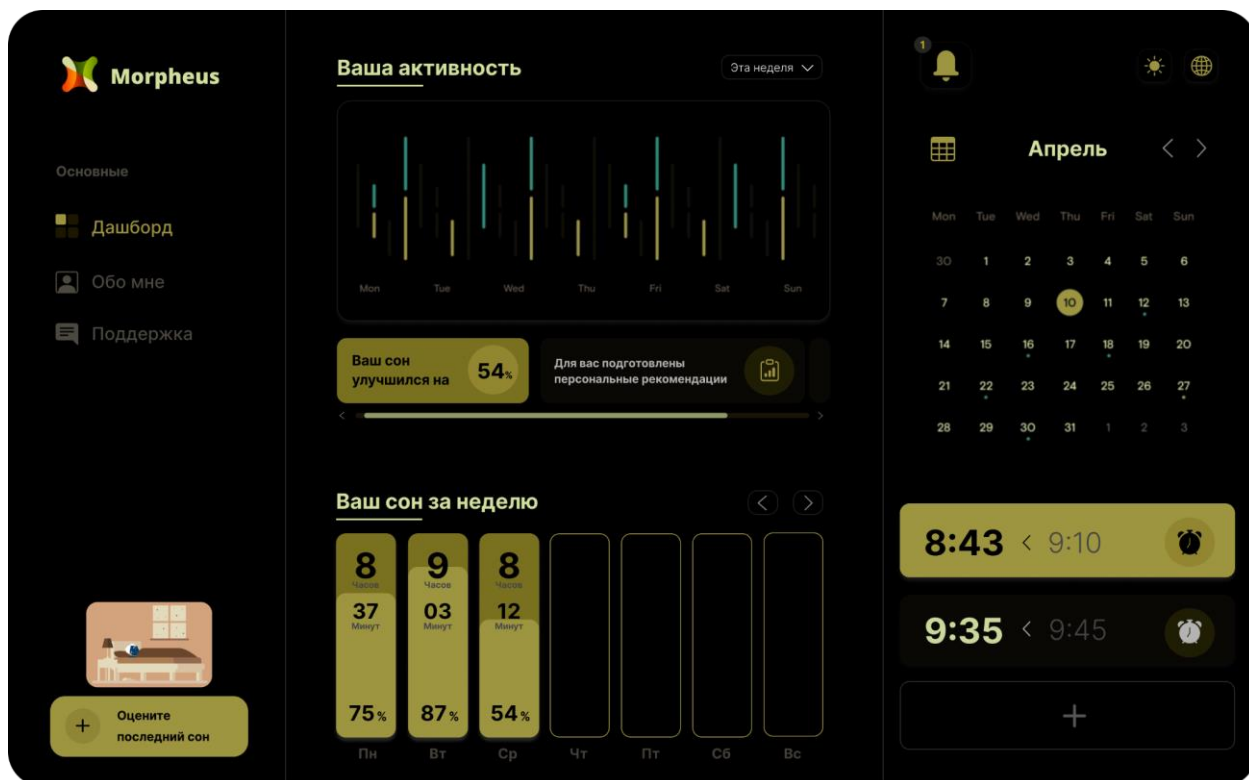


Рисунок 15, Основная страница приложения в стиле темной темы

При клике на кнопку “Оценить последний сон” у пользователя всплывает окно, с ним вы можете ознакомиться на приведенном ниже рисунке (Рисунок 16).

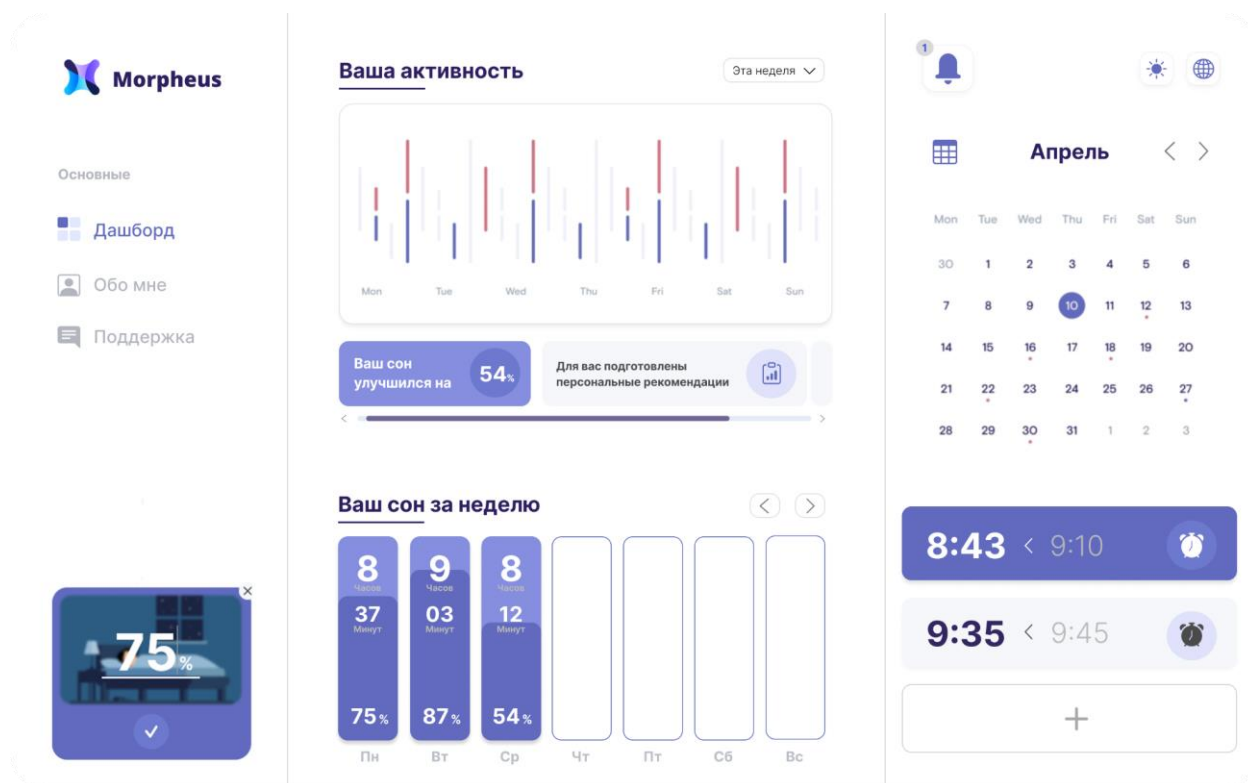


Рисунок 16, Основная страница приложения с открытым окном “Оцените последний сон”

При клике на кнопку добавления будильника у пользователя всплывает окно, с ним вы можете ознакомиться на приведенном ниже рисунке (Рисунок 17).

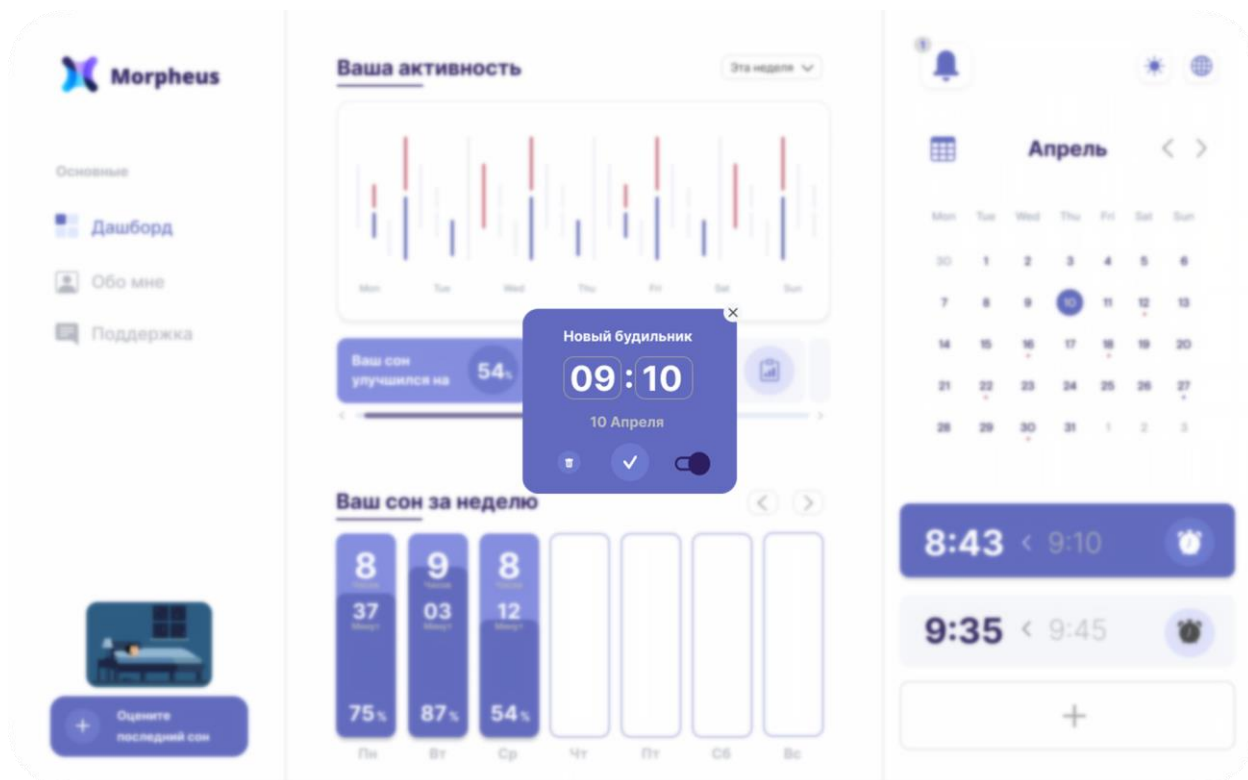


Рисунок 17, Основная страница приложения с открытым окном создания будильника

При двойном клике на элемент сна за неделю у пользователя всплывает окно, с ним вы можете ознакомиться на приведенном ниже рисунке (Рисунок 18).

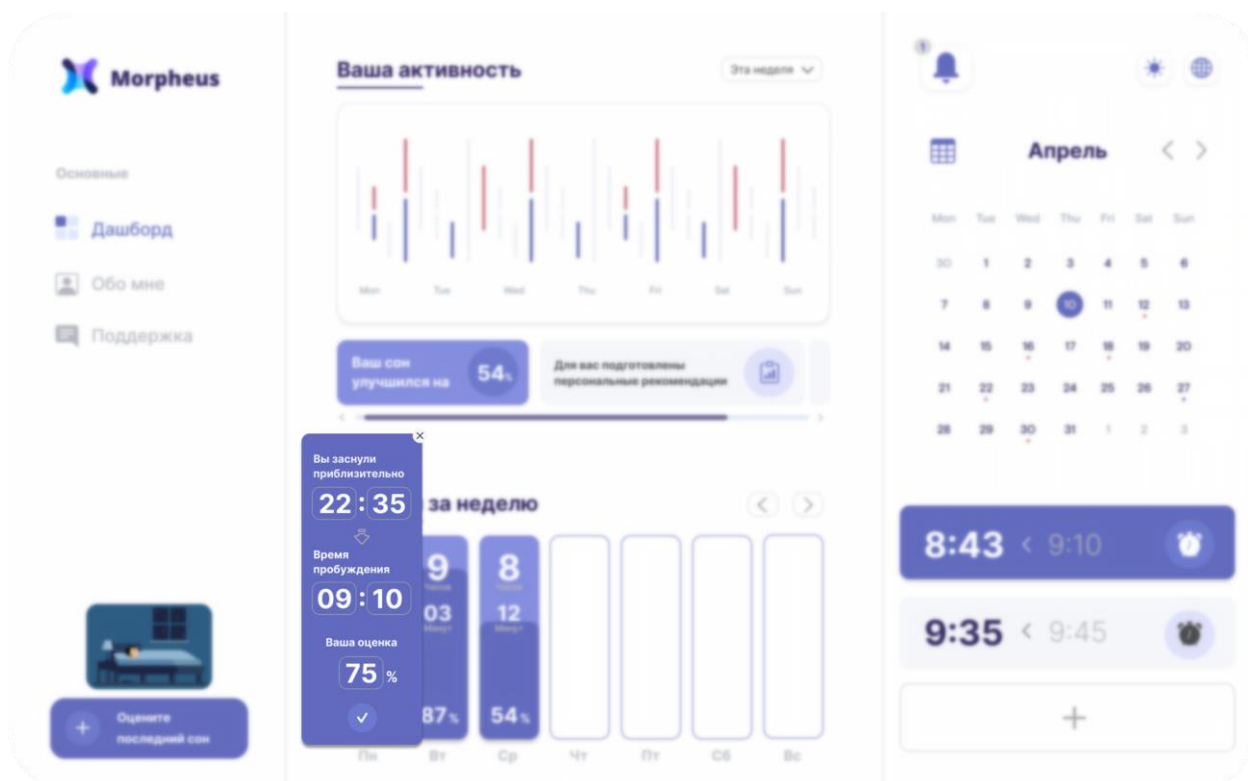
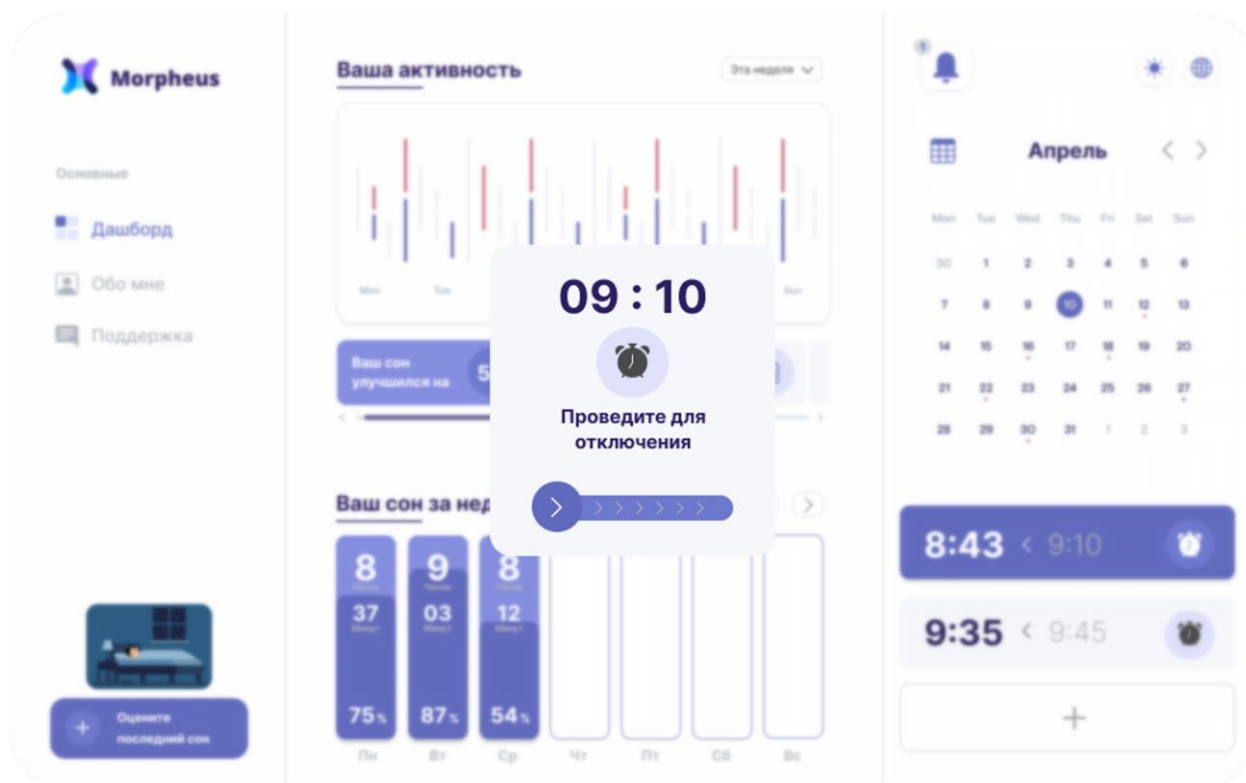


Рисунок 18, Основная страница приложения с открытым окном редактирования сна



При срабатывании будильника у пользователя всплывает окно, с ним вы можете ознакомиться на приведенном ниже рисунке (Рисунок 19).



*Рисунок 19, Основная страница приложения со сработавшим будильником*

При клике на кнопку “Обо мне” пользователя направляется на страницу с информацией о пользователе. На данной странице доступен вход в аккаунт Google Fit, изменение пользовательских данных и выход из аккаунта. Вы можете ознакомиться с ее образцом на приведенном ниже рисунке (Рисунок 20).

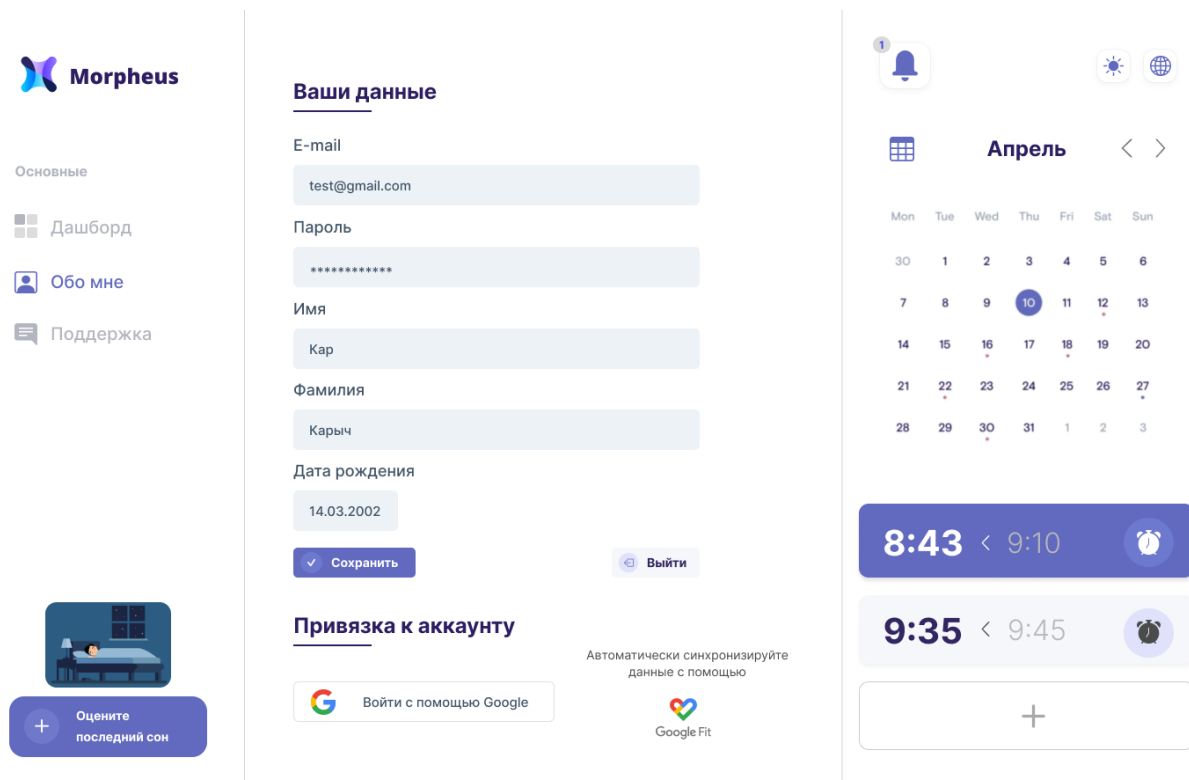


Рисунок 20, Страница с информацией о пользователе

### 3.6 Тестирование сервисов Spring

Используя инструменты Spring Framework, совместно с библиотекой JUnit, было выполнено модульное и интеграционное тестирование. Эти инструменты предоставляют удобные аннотации и функции для проведения тестирования.

В контексте тестирования сервисов heart\_rate и ML было разработано шесть отдельных классов для каждого из контроллеров, в каждом из которых тестируются конкретные конечные точки. Также каждый тестовый случай был помечен аннотацией @Test и @DisplayName для облегчения понимания их работы.

Тестирование включало в себя создание объектов, их отправку на соответствующие конечные точки и последующую проверку статуса возвращаемого ответа на корректность. При этом, для каждой конечной точки было создано от двух до четырех различных сценариев тестирования. В итоге

было разработано и успешно запустил 93 тестовых сценария, обеспечив необходимое тестовое покрытие проекта.

### 3.7 Тестирование моделей машинного обучения

Ранее было отмечено, что все данные, использованные для обучения алгоритмов машинного обучения, были разделены в соотношении 60/20/20, где последние 20% были предназначены для тестирования. Это тестирование проводится сразу же после завершения обучения, чтобы подтвердить правильность выбранного алгоритма.

Для прогнозирования результатов тестовых данных использовался метод `predict` из библиотеки `scikit-learn`. Затем предсказанные результаты были сопоставлены с реальными значениями для оценки точности модели.

При этом ошибка модели, основанной на алгоритме "линейной регрессии", составила всего 3,67%. В то время как для модели, обученной с использованием алгоритма "логистической регрессии", показатель ошибки составил 5,83%. Таким образом, можно утверждать, что оба алгоритма продемонстрировали достаточно высокую точность предсказаний.

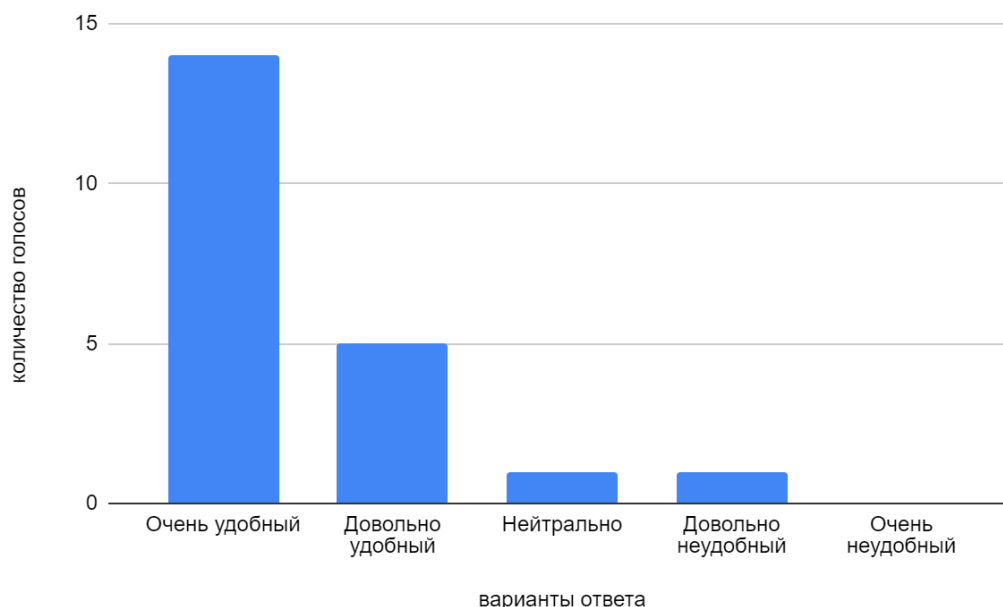
### 3.8 Деплой приложения

Приложение было развернуто на приватном сервере, используя набор серверного программного обеспечения Tomcat. Этот инструмент служит для испытаний, отладки и запуска веб-приложений, написанных на языке Java.

### 3.9 Подтверждение экспериментальной цели

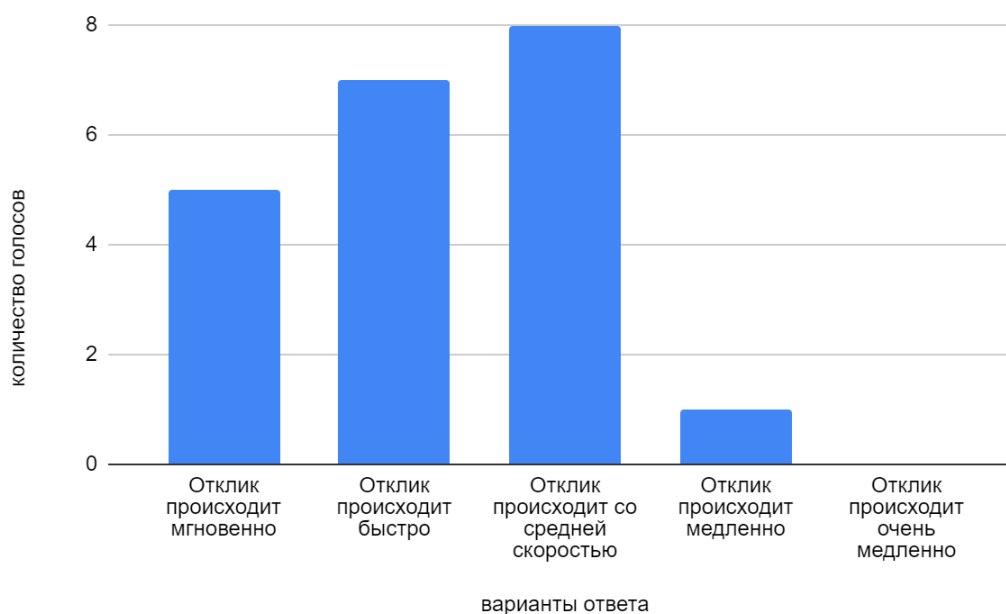
После двухнедельного использования приложения, был организован опрос среди пользователей, которые использовали систему для нормализации своего режима сна.

Результаты опроса удобства интерфейса показали следующие результаты (Рисунок 21).



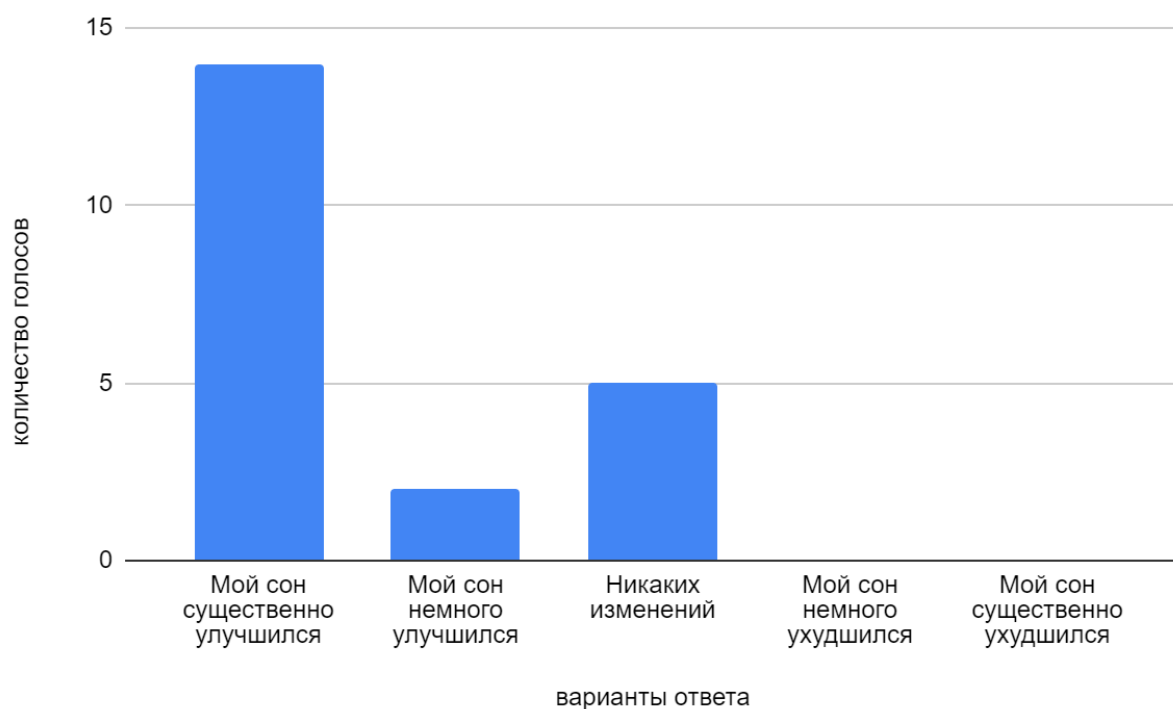
*Рисунок 21, Результаты опроса удобства использования интерфейса*

Результаты опроса скорости отклика приложения показали следующие результаты (Рисунок 22).



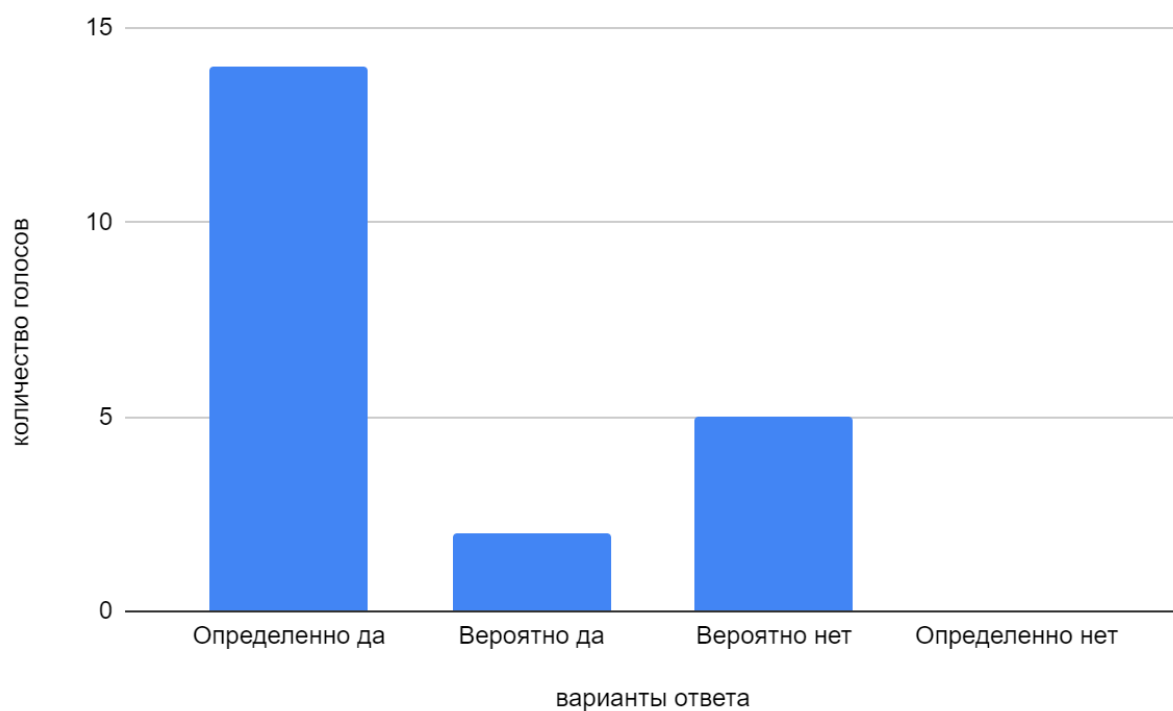
*Рисунок 22, Результаты опроса скорости отклика приложения*

Результаты опроса изменение своего сна после использования приложения показали следующие результаты (Рисунок 23).



*Рисунок 23, Результаты опроса изменения сна после использования приложения*

Результаты опроса «Считаете ли вы, что приложение помогло вам нормализовать режим сна?» показали следующие результаты (Рисунок 24).



*Рисунок 24, Результаты опроса помощи приложения в нормализации режима сна*

Результаты опроса «Будете ли вы продолжать использовать приложение и рекомендовать его друзьям и близким?» показали следующие результаты (Рисунок 25).

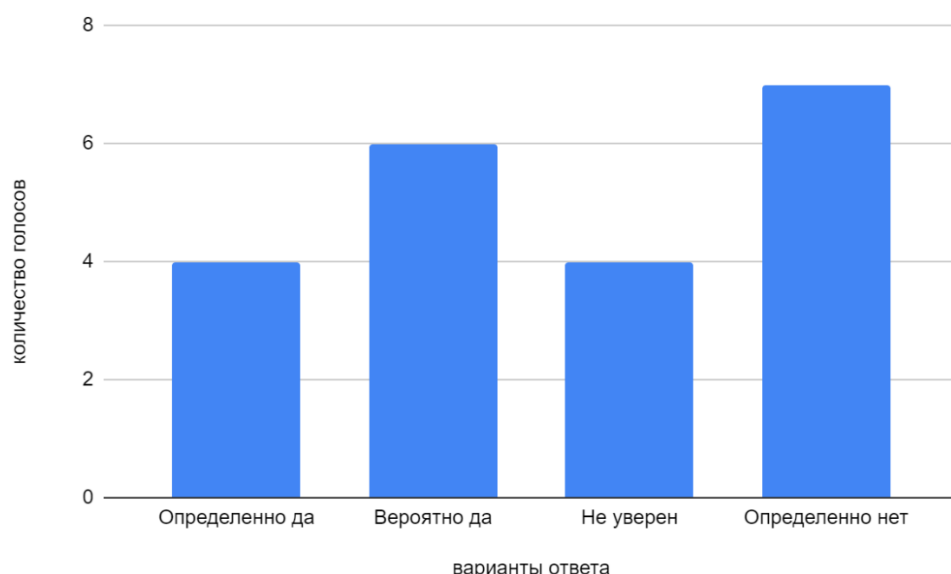


Рисунок 25, Результаты опроса по продолжению использования приложения

После периода эксплуатации приложения были собраны и обработаны данные о качестве сна, которые пользователи регулярно вносили в систему. На основании этих данных было выявлено, что приложение смогло нормализовать режим сна у 66% пользователей (Рисунок 26). Эти результаты полностью подтверждают решение поставленной задачи.

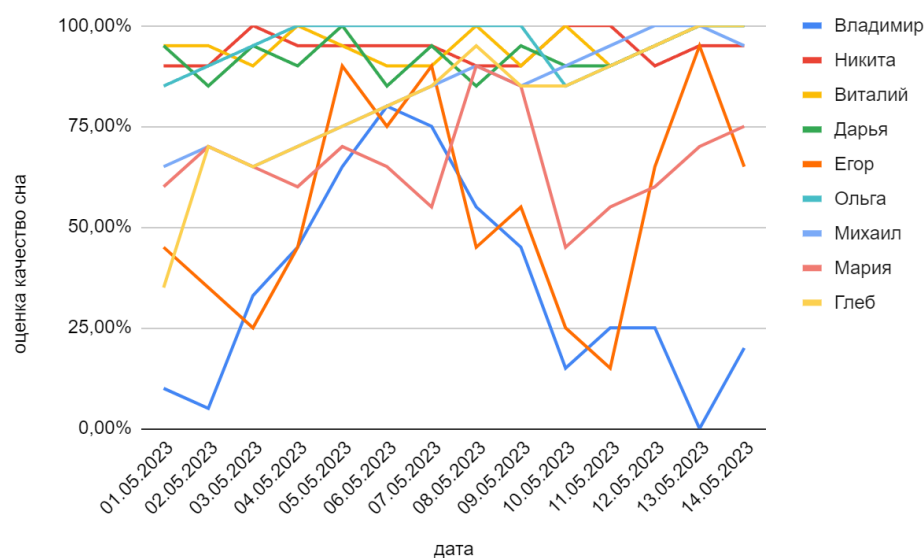


Рисунок 26, График зависимости качества сна от даты

## ЗАКЛЮЧЕНИЕ

В эпоху современных технологий, несмотря на первостепенную роль сна как фундаментальной потребности в обеспечении физического и психического благополучия, большинство людей на планете сталкиваются с проблемой хронического недосыпа.

Огромное число факторов таких как стресс, возраст, нерегулярное соблюдение графика сна способствуют ухудшению режима и самочувствия человека в целом.

Для достижения поставленной цели, были решены следующие задачи: исследование предметной области, анализ существующих решений с выявлением достоинств и недостатков, определены требования к системе. Также была разработана архитектура базы данных, проработана структура приложения, с последующей реализацией, тестированием и подтверждением поставленной цели.

По итогам выполнения выпускной квалификационной работы было разработано веб-приложение, которое помогает людям нормализовать свой режим сна, используя для этого их частоту сердечных сокращений. Система взаимодействует с Google Fit API для сбора информации и определяет текущую фазу сна пользователя, предсказывая оптимальное время для подъема. Такой подход делает пробуждение максимально комфортным, снижая стресс и стимулируя пользователя соблюдать здоровый режим сна.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Insomnia May Mediate the Relationship Between Stress and Anxiety: A Cross-Sectional Study in University Students / Md Dilshad Manzar, Mohammed Salahuddin, Seithikurippu R Pandi-Perumal, and Ahmed S Bahammam // Nat Sci Sleep. - №13. - 2021. - С. 31-38. - PMID: 33447116.
2. Sleep Duration and All-Cause Mortality: A Systematic Review and Meta-Analysis of Prospective Studies / Francesco P. Cappuccio, Lanfranco D'Elia, Pasquale Strazzullo, Michelle A. Miller // Sleep. - # 33. - 2010. - С. 585-592. - PMID: 20469800.
3. Aakash K. Patel. Physiology, Sleep Stages / Vamsi Reddy, Karlie R // JMIR Mhealth Uhealth. - № 1. - 2021. - PMID: 30252388.
4. Michiaki Nagai. Sleep Duration as a Risk Factor for Cardiovascular Disease- a Review of the Recent Literature / Satoshi Hoshida, Kazuomi Kario // Curr Cardiol Rev. - # 6. - 2010. - С. 54-61. - DOI: 10.2174/157340310790231635
5. Christina Southern Reh. Somatotropin in the treatment of growth hormone deficiency and Turner syndrome in pediatric patients: a review / Mitchell E Geffner // Clin Pharmacol. - # 2. - 2010. - С. 111-122. - DOI: 10.2147/CPAA.S6525.
6. Estimating sleep parameters using an accelerometer without sleep diary / Vincent Theodoor van Hees и др. // Sci Rep. - № 8. - 2018. - DOI: 10.1038/s41598-018-31266-z.
7. Automated sleep stage classification based on tracheal body sound and actigraphy / Christoph Kalkbrenner и др. // German medical science: GMS e-journal. - № 17. - 2019. - DOI: 10.3205/000268.
8. Dean J. Miller. A Validation of Six Wearable Devices for Estimating Sleep, Heart Rate and Heart Rate Variability in Healthy Adults / Charli Sargent, Gregory D. Roach // Sensors. - # 22. - 2022. - DOI:10.3390/s22166317.



9. Spring Project - Spring Security [Электронный ресурс] // URL:  
<https://spring.io/projects/spring-security> (дата обращения: 20.04.2023)
10. Spring Project - Spring Boot [Электронный ресурс] // URL:  
<https://spring.io/projects/spring-boot> (дата обращения: 20.04.2023)
11. Жизненный цикл модели машинного обучения // Викиконспекты  
[Электронный ресурс]. URL: <https://clck.ru/iJb3Y> (дата обращения:  
20.04.2023).
12. Spring Project - Spring Cloud [Электронный ресурс] // URL:  
<https://spring.io/projects/spring-cloud> (дата обращения: 22.04.2023)