# Содержание

[**1. АННОТАЦИЯ 5**](#_Toc162906169)

[**1 ВВЕДЕНИЕ 7**](#_Toc162906170)

[**1.1 Наименование программы 7**](#_Toc162906171)

[**1.2 Краткая характеристика области применения 7**](#_Toc162906172)

[**1.3 Документ, на основании которого ведется разработка 7**](#_Toc162906173)

[**1.4 Наименование темы разработки 7**](#_Toc162906174)

[**2 НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ 8**](#_Toc162906175)

[**2.1 Назначение программы 8**](#_Toc162906176)

[**2.1.1 Функциональное назначение 8**](#_Toc162906177)

[**2.1.2 Эксплуатационное назначение 9**](#_Toc162906178)

[**2.1.3 Краткая характеристика области применения 9**](#_Toc162906179)

[**3 ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ 10**](#_Toc162906180)

[**3.1 Постановка задачи на разработку программы 10**](#_Toc162906181)

[**3.2 Описание и обоснование архитектуры программы 10**](#_Toc162906182)

[**3.2.1. Описание архитектуры программы 10**](#_Toc162906183)

[**3.2.1 Обоснование архитектуры программы 14**](#_Toc162906184)

[**3.3 Описание алгоритма работы программы 15**](#_Toc162906185)

[**3.4 Описание выбора метода организации входных и выходных данных 17**](#_Toc162906186)

[**3.4.1 Выбор метода организации входных данных 17**](#_Toc162906187)

[**3.4.2 Выбор метода организации выходных данных 17**](#_Toc162906188)

[**3.5 Описание и обоснование метода выбора технических и программных средств 18**](#_Toc162906189)

[**3.5.1 Описание метода выбора технических и программных средств 18**](#_Toc162906190)

[**3.5.2 Обоснование метода выбора технических и программных средств 18**](#_Toc162906191)

[**4. ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ 19**](#_Toc162906192)

[**4.2. Ориентировочная экономическая эффективность 19**](#_Toc162906193)

[**4.3. Предполагаемая потребность 19**](#_Toc162906194)

[**4.4. Экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами 19**](#_Toc162906195)

[**5. ИСТОЧНИКИ, ИСПОЛЬЗОВАННЫЕ ПРИ РАЗРАБОТКЕ 22**](#_Toc162906196)

[**ПРИЛОЖЕНИЕ 1 23**](#_Toc162906197)

[**ПРИЛОЖЕНИЕ 2 25**](#_Toc162906198)

[**ПРИЛОЖЕНИЕ 3 29**](#_Toc162906199)

[**ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ 30**](#_Toc162906200)

# АННОТАЦИЯ

Данный программный документ представляет собой пояснительную записку к программному проекту «Менеджер времени».

Настоящая пояснительная записка содержит следующие разделы: «Введение», «Назначение и область применения», «Технические характеристики», «Ожидаемые технико-экономические показатели» и приложения.

В разделе «Введение» представлены наименование и краткое наименование программы на русском и английском языках, а также документы, на основании который ведется разработка.

В разделе «Назначение и область применения» содержатся функциональное и эксплуатационное назначение программы и краткая характеристика области её применения.

В разделе «Технические характеристики» присутствуют следующие подразделы: постановка задачи на разработку программы, описание функционирования программы, описание и обоснование алгоритма работы программы, описание и обоснование математического алгоритма работы программы, описание выбора метода организации входных и выходных данных, описание работы с файловой системой, описание и обоснование выбора состава технических и программных средств.

В разделе «Ожидаемые технико-экономические показатели» указана предполагаемая потребность и экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами.

Программный документ разработан в соответствии с требованиями:

* 1. ГОСТ 19.101-77[1]: Виды программ и программных документов.
  2. ГОСТ 19.102-77[2]: Стадии разработки.
  3. ГОСТ 19.103-77[3]: Обозначения программ и программных документов.
  4. ГОСТ 19.104-78[4]: Основные надписи.
  5. ГОСТ 19.105-78 Общие требования к программным документам [5];
  6. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом [6];
  7. ГОСТ 19.404-79 Пояснительная записка. Требования к содержанию и оформлению [7].

Изменения к Пояснительной записке оформляются согласно ГОСТ 19.603-78 [8], ГОСТ 19.604-78 [9].

# ВВЕДЕНИЕ

## Наименование программы

Наименование программы на русском языке – «Менеджер времени».

Наименование программы на английском языке – «Time Manager».

Краткое наименование программы – «TManager».

## Краткая характеристика области применения

«Менеджер времени» - мобильное iOS-приложение, позволяющее пользователям взять контроль над своим временем. Оно представляет собой централизованный интерфейс, позволяющий с легкостью планировать задачи, отслеживать дедлайны и настраивать напоминания.

## Документ, на основании которого ведется разработка

Разработка ведется на основании учебного плана подготовки бакалавров по направлению 09.03.04 “Программная инженерия” Факультета компьютерных наук Национального исследовательского университета “Высшая школа экономики” и утвержденной академическим руководителем программы темы курсового проекта.

## Наименование темы разработки

Наименование темы разработки – «Менеджер времени».

Наименование темы разработки на английском языке – «Time Manager».

Условное обозначение темы разработки – «TManager».

# НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

## Назначение программы

### Функциональное назначение

Приложение "Time Manager" – это комплексный инструмент для удобного планирования времени, распределения задач и управления дедлайнами. Оно разработано для устройств на базе ОС iOS и обеспечивает широкий спектр функций для повышения продуктивности и организации.

Основные функции приложения:

* 1. Добавление задач: Пользователи могут добавлять задачи на конкретный день, указывая время, продолжительность и тег задачи (например, учёба, работа).
  2. Редактирование задач: Существующие задачи могут быть легко отредактированы, что позволяет изменять их параметры при необходимости.
  3. Удаление задач: Удаление существующих задач.
  4. Добавление общих задач: Приложение позволяет добавлять общие задачи. Для таких задач устанавливается только дата дедлайна (например, дедлайны домашних заданий, встреч).
  5. Создание списков задач: Пользователи могут создавать списки общих задач, классифицируя их по категориям для удобства навигации и организации.
  6. Отображение задач: Приложение удобно отображает конкретный день с задачами, списки общих задач по датам, отложенные задачи, что облегчает планирование и управление временем.
  7. Интеграция с календарем и напоминаниями: Возможность интегрировать напоминания о задачах в календарь и систему напоминаний iPhone, улучшая управление временем и напоминание о важных событиях.
  8. Поиск по тегам: Приложение позволяет удобно настроить фильтр по тегам.

Общее функциональное назначение "Менеджера времени" заключается в предоставлении мощного, но в то же время простого в использовании инструмента для управления временем и задачами, повышая продуктивность и помогая в организации повседневной жизни.

### Эксплуатационное назначение

Эксплуатационное назначение приложения "Менеджер времени" можно описать следующим образом:

Приложение разработано для широкого круга пользователей, которым необходим инструмент для планирования и управления временем. Основные категории пользователей включают:

* 1. Студенты и Учащиеся: для управления учебными заданиями и контроля сроков.
  2. Профессионалы: для координации рабочих задач, встреч и дедлайнов.
  3. Фрилансеры: для организации множественных проектов и соблюдения сроков.
  4. Личное Планирование: для управления личными задачами и обязанностями.
  5. Организаторы и Планировщики: для координации мероприятий и поддержания порядка в повседневных делах.

Приложение "Менеджер времени" разработано исключительно для пользователей устройств на платформе iOS, обеспечивая высокую производительность и оптимальное взаимодействие с системными функциями Apple. Оно представляет собой эффективное и удобное решение для пользователей, стремящихся повысить свою продуктивность и организованность в управлении временем и задачами.

### Краткая характеристика области применения

«Менеджер времени» - мобильное iOS-приложение, позволяющее пользователям взять контроль над своим временем. Оно представляет собой централизованный интерфейс, позволяющий с легкостью планировать задачи, отслеживать дедлайны и настраивать напоминания.

# ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

## Постановка задачи на разработку программы

Разрабатываемая программа должна предоставлять возможность выполнения следующих задач:

* 1. Добавление задач:
  2. Хранение данных;
  3. Очистка всех данных;
  4. Редактирование данных уже созданных задач;
  5. Создание тегов;
  6. Удаление тегов;
  7. Удаление созданных задач;
  8. Фильтрация по тегам;
  9. Понедельное отображение дней с конкретными задачами;
  10. Подробный просмотр всех видов задач;
  11. Отображение списка отложенных задач;
  12. Отображение списка общих задач;
  13. Интеграция c встроенным приложением “Календарь” iOS;
  14. Интеграция с встроенным приложением “Напоминания” iOS;
  15. Облачная синхронизация с iCloud ОС iOS.
  16. Настройки приложения;

## Описание и обоснование архитектуры программы

### Описание архитектуры программы

Архитектура приложения "Менеджер времени" основана на модели MVP (Model-View-Presenter), которая является деривативом классического паттерна MVC (Model-View-Controller). В этой модели View (экран) отвечает за отображение данных пользователю и передачу пользовательского ввода в Presenter. Presenter является посредником, который обрабатывает все бизнес-логику и управляет потоком данных в Model и обратно, обновляя View. Model представляет собой слой данных, который управляет бизнес-правилами и хранит состояние приложения.

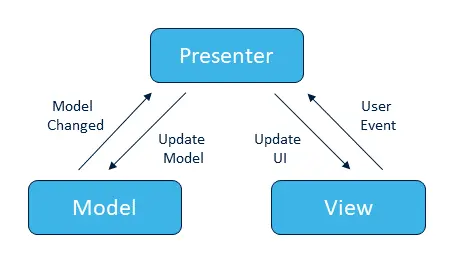


Рисунок 1. Схема MVP архитектуры.

Основные экраны:

* Экран конкретных задач
* Экран отложенных задач
* Экран общих задач
* Настройки

Они отображаются с помощью MainTabBarViewController, что позволяет обеспечить удобную навигацию по приложению и позволяет использовать данную навигационную панель:

A white background with black and white clouds

Description automatically generatedРисунок 2. Нижняя навигационная панель.

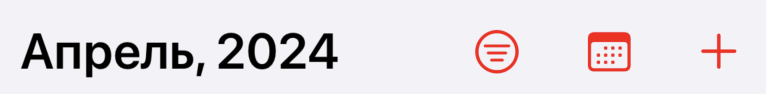
Причём для навигации к некотрым другим экранам каждый основной экран обернут в UINavigationController, которой помогает обеспечить верхнюю навигационную панель, например для экрана конкретных задач:

Рисунок 3. Верхняя навигационная панель экрана конкретных задач.

Для каждого экрана реализован соответсвующий MVP модуль. (см Приложение 1) Причём для некоторых экранов может быть не реализован соответсвующий Presenter, в связи с тем, что экран отвечает сугубо за отображение других экранов и не имеет никакой бизнес логики.

С учётом требований к функционалу, была разработан Model, соответсвующий следующей схеме классов:

A diagram of a task

Description automatically generated

Рисунок 4. Схема классов.

Соответсвующая схема используется в БД. Причём было установлена инверсивная связь между каждыми классами и правило удления Nullify, т.е. при удалении одного из объектов в связи, связанный объект не будет удаляться и хранить ссылеу на пустой объект вместо удаленного.

Репозиторий используется как абстракция над источниками данных, что позволяет Presenter работать с данными, не зная деталей их хранения. Это упрощает изменения в базе данных и обеспечивает гибкость при тестировании, поскольку источник данных может быть легко заменен моками или стабами. Интерфейсы репозиториев можно найти в [приложении 2](#пп2).

Для управления локальным хранением данных используется фреймворк Core Data, который позволяет эффективно работать с графом объектов и их сохранением в локальной базе данных на устройстве. Использование Core Data в связке с CloudKit предоставляет возможность синхронизации данных с облачным хранилищем iCloud, что обеспечивает сохранность данных даже при смене устройства.

Для упрощения доступа к данным и обеспечения единой точки взаимодействия с базой данных применяется паттерн синглетон для репозиториев, что позволяет легко получать доступ к ним из различных частей приложения.

Для интеграции с приложениями “Напоминания” и “Календарь” используются интерфейсы создания задач в этих приложениях:

protocol ReminderManaging {

func create(reminderModel: ReminderEventModel) -> Bool

}

protocol CalendarManaging {

func create(eventModel: CalendarEventModel) -> Bool

}

Где определены модели:

struct CalendarEventModel {

var title: String

var startDate: Date

var endDate: Date?

var note: String

}

struct ReminderEventModel {

var title: String

var startDate: Date?

var dueDate: Date

var note: String

}

При реализации данных интерфейсов используется фреймворк EventKit.

### Обоснование архитектуры программы

Выбор архитектуры MVP и использование репозиториев обоснован необходимостью обеспечения тестируемости компонентов, разделения ответственности и упрощения поддержки кода. MVP позволяет изолировать логику представления от логики бизнес-правил, что упрощает написание модульных тестов и поддержку кода.

Представленная реализация основных экранов помогает построить эффективную и интуитивно понятную навигационную систему.

Представленная схема классов помогает в переиспользовании и уменьшении кода при разработке, тем самым ускоряя разработку, упрощая тестирования и отладку.

Использование репозиториев позволяет централизованно управлять данными приложения, что обеспечивает удобство внесения изменений в модель данных без необходимости переписывания бизнес-логики, связанной с хранением данных.

Core Data в сочетании с CloudKit выбраны для локального хранения данных и их синхронизации с iCloud, что позволяет пользователям восстановить информацию при смене устройства или её потере. Это решение также позволяет избежать создания собственной системы регистрации и управления данными пользователей, снижая технический риск и повышая удобство использования приложения.

Представленная схема интеграции с приложениями “Напоминания” и “Календарь” позволяет обеспечить тестируемость и переиспользуемость кода.

Архитектура приложения "Менеджер времени" реализована таким образом, чтобы обеспечить гибкость, масштабируемость и поддержку, сосредоточив внимание на пользовательском опыте и надежности работы приложения.

## Описание алгоритма работы программы

* 1. Запуск приложения и инициализация данных:
  + Загрузка сохранённых данных из локальной базы данных.
  + Инициализация основных экранов и настройка навигационной панели.
  1. Добавление новой задачи:
  + Пользователь нажимает на кнопку добавления задачи.
  + Отображается форма для ввода данных о задаче.
  + Пользователь вводит данные и подтверждает создание задачи.
  + Данные сохраняются в локальную базу данных и опционально синхронизируются с iCloud.
  1. Редактирование задачи:
* Пользователь выбирает задачу из списка.
* Отображается форма с текущими данными задачи.
* Пользователь изменяет данные и подтверждает их сохранение.
* Обновленные данные сохраняются в базу данных.
  1. Удаление задачи:
* Пользователь выбирает задачу и инициирует процесс удаления.
  1. Фильтрация задач:
* Пользователь выбирает критерии фильтрации по тегам.
* Программа отображает только те задачи, которые соответствуют выбранным тегам.
  1. Просмотр задач:

Для конкретной задачи:

* Программа отображает задачи выбранного дня выбранной недели с учетом времени начала и окончания, а также отметками о выполнении.

Для общей задачи:

* Программа отображает задачи, дедлайны которых позже или равна выбранной дате.

Для отложенной задачи:

* Программа отображает все задачи.
  1. Просмотр и редактирование тегов:
* Пользователь может просматривать список тегов, добавлять новые или удалять ненужные.
  1. Интеграция с системными приложениями:
* Пользователь может отправить задачи в приложение "Календарь" или "Напоминания" для создания событий и напоминаний при создании задач.
  1. Синхронизация данных:
* Все изменения в данных автоматически синхронизируются с облачным хранилищем iCloud.

## Описание выбора метода организации входных и выходных данных

### Выбор метода организации входных данных

Входные данные в приложении "Менеджер времени" представлены информацией, вводимой пользователем через графический интерфейс, такой как названия задач, даты, описания и теги. Эффективная организация этих данных достигается путем использования интуитивно понятных форм ввода, которые минимизируют возможность ошибок пользователя и облегчают процесс добавления и редактирования задач.

Каждый элемент ввода в приложении имеет четко определенный формат и ограничения, которые позволяют сохранять консистентность данных в приложении и упрощают их обработку. Для организации данных используются стандартные компоненты UIKit, такие как UITextField и UIDatePicker, что обеспечивает единообразие с другими приложениями на платформе iOS и сокращает время на разработку интерфейса.

### Выбор метода организации выходных данных

Выходные данные приложения включают в себя отображение информации о задачах, фильтрацию и сортировку задач, а также экспорт данных в системные приложения и облачное хранилище. Отображение данных происходит с помощью таблиц (UITableView) и коллекций (UICollectionView), что позволяет представить информацию структурированно и удобно для пользователя.

Использование репозиториев для работы с данными позволяет презентерам запрашивать отфильтрованные и отсортированные данные без знания конкретных деталей реализации хранения данных, что упрощает процесс обновления и отображения информации в интерфейсе.

Синхронизация данных с iCloud осуществляется с помощью CloudKit, что обеспечивает надежное и безопасное хранение данных в облаке и их доступность на всех устройствах пользователя. Этот метод также позволяет автоматически обновлять данные в приложении в реальном времени при их изменении

## Описание и обоснование метода выбора технических и программных средств

### Описание метода выбора технических и программных средств

Для бесперебойной работы программного продукта требуется:

* 1. Мобильное устройство с сенсорным экраном
  2. 300 мб или более свободного дискового пространства
  3. 300 мб или более ОЗУ
  4. Стабильное подключение к сети Интернет (1 Мбит/c или выше)
  5. Операционная система iOS 17.0 или выше

### Обоснование метода выбора технических и программных средств

Выбор технических и программных средств для разработки и успешной эксплуатации мобильного приложения "Менеджер времени" был основан на необходимости обеспечения стабильной работы, высокой доступности функций и оптимизации пользовательского опыта.

# ОЖИДАЕМЫЕ ТЕХНИКО­-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

## Ориентировочная экономическая эффективность

В рамках проекта расчёт экономической эффективности программного продукта не производился.

## Предполагаемая потребность

Ожидается, что программа должна быть интересна для пользователей всех возрастов, желающих организовывать свое время.

## Экономические преимущества разработки по сравнению с отечественными и зарубежными образцами или аналогами

В таблице 1 приведена сравнительная характеристика базового функционала с прямыми аналогами. В таблице 2 приведена сравнительная характеристика c косвенными аналогами (см следующую страницу):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Критерий\Аналог** | **TManager** | Structured | Tappsk | Productive | TickTick | Focus To-Do |
| Возможность пользоваться бесплатно | ✅ | ✅ | ✅ | ✅ | ✅ | ✅ |
| Платная подписка | ❌ | ✅ | ✅ | ✅ | ✅ | ✅ |
| Кроссплатформенность | ❌ | ✅ | ✅ | ✅ | ✅ | ✅ |
| Установка тегов | ✅ | ❌ | ✅ | ✅ | ✅ | ✅ |
| Множественные теги | ✅ | ❌ | ❌ | ❌ | ❌ | ✅ |
| Возможность фильтрации по тегам | ✅ | ❌ | ✅ | ❌ | ❌ | ✅ |
| Отображение недельного календаря задач | ✅ | ✅ | ❌ | ✅ | ✅ | ✅ |
| Возможность импортировать задачи в “Календарь” | ✅ | ✅ | ❌ | ❌ | ✅ | ❌ |
| Возможность импортировать задачи в “Напоминания” (для iOS) | ✅ | ✅ | ❌ | ❌ | ✅ | ❌ |
| Статистика выполнений задач | ❌ | ❌ | ✅ | ✅ | ✅ | ✅ |
| Режим фокусировки | ❌ | ❌ | ❌ | ❌ | ❌ | ✅ |
| Облачная синхронизация | ✅ | ✅ | ✅ | ❌ | ❌ | ✅ |

Таблица 1 – Прямые аналоги

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Критерий\Аналог** | **TManager** | Apple Notes | Notion | Google Calendar | Google Sheets |
| Возможность пользоваться бесплатно | ✅ | ✅ | ✅ | ✅ | ✅ |
| Платная подписка | ❌ | ❌ | ✅ | ❌ | ❌ |
| Кроссплатформенность | ❌ | ❌ | ✅ | ✅ | ✅ |
| Установка тегов | ✅ | ✅ | ✅ | ✅ | ✅ |
| Множественные теги | ✅ | ❌ | ✅ | ❌ | ❌ |
| Возможность фильтрации по тегам | ✅ | ❌ | ✅ | ❌ | ❌ |
| Отображение недельного календаря задач | ✅ | ❌ | ✅ | ✅ | ❌ |
| Возможность импортировать задачи в “Календарь” | ✅ | ❌ | ❌ | ✅ | ❌ |
| Возможность импортировать задачи в “Напоминания” (для iOS) | ✅ | ❌ | ❌ | ❌ | ❌ |
| Статистика выполнений задач | ❌ | ❌ | ✅ | ✅ | ✅ |
| Облачная синхронизация | ✅ | ✅ | ✅ | ✅ | ✅ |

Таблица 2 – Косвенные аналоги

\*Зелёная заливка – только в платной подписке

“Менеджер времени” выделяется на фоне других приложений для управления задачами благодаря своей способности бесплатного использования, уникальным функциям импорта задач в приложения "Календарь" и "Напоминания" для пользователей iOS, а также возможностью множественного тегирования. Это делает его привлекательным выбором для тех, кто ищет гибкость и интеграцию с продуктами Apple.

# ИСТОЧНИКИ, ИСПОЛЬЗОВАННЫЕ ПРИ РАЗРАБОТКЕ

1. ГОСТ 19.101-77 Виды программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

2. ГОСТ 19.102-77 Стадии разработки. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

3. ГОСТ 19.103-77 Обозначения программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

4. ГОСТ 19.104-78 Основные надписи. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

5. ГОСТ 19.105-78 Общие требования к программным документам. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

6. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

7. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

8. ГОСТ 19.603-78 Общие правила внесения изменений. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

9. ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

10. ГОСТ 15150-69 Машины, приборы и другие технические изделия. Исполнения для различных климатических районов. Категории, условия эксплуатации, хранения и транспортирования в части воздействия климатических факторов внешней среды. – М.: Изд-во стандартов, 1997.

11. ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.

12. Swift Framework [Электронный ресурс]. Режим доступа: <https://developer.apple.com/documentation/swift>, свободный

13. Шаблон архитектуры MVP(Model View Prresenter) [Электронный ресурс]. Режим доступа - https://www.geeksforgeeks.org/mvp-model-view-presenter-architecture-pattern-in-android-with-example/, свободный

# ПРИЛОЖЕНИЕ 1

Cписок MVP модулей

|  |  |  |
| --- | --- | --- |
| **Названия View (экрана)** | **Название Presenter** | **Назначение** |
| SpecificTaskListViewController | SpecificTaskListPresenter | Экран отображения списка конкретных задач |
| AddSpecificTaskViewController | AddSpecificTaskPresenter | Экран для добавления конкретной/отложенной задачи |
| MainTabBarViewControllerController | - | Экран на котором отображаются основные экраны |
| SpecificTaskDescriptionViewController | - | Экран для отображения подробной информации по конкретной/отложенной задаче |
| AddTagViewController | AddTagPresenter | Экран для создания тега |
| DatePickerViewController | - | Экран для отображения календаря |
| GeneralTaskListViewController | GeneralTaskListPresenter | Отображение списка общих задач |
| AddGeneralTaskViewController | AddGeneralTaskPresenter | Экран для добавления общей задачи |
| GeneralTaskSearchViewController | GeneralTaskSearchPresenter | Экран для поиска общей задачи |
| DelayedSpecificTaskListViewController | DelayedSpecificTaskListPresenter | Экран отображения списка отложенных задач |
| GeneralTaskDescriptionViewController | GeneralTaskDescriptionPresenter | Экран для отображения подробной информации по общей задаче |
| TagFilterViewController | TagFilterViewController | Экран для выбора фильтра по тегам |
| SettingsViewController | SettingsPresenter | Экран настроек |
| TagSettingsViewController | TagSettingsViewController | Экран для удаления и добавления тегов |

# ПРИЛОЖЕНИЕ 2

Cписок интерфейсов репозиториев

1. Интерфейс для репозитория конкретных задач

|  |  |  |  |
| --- | --- | --- | --- |
| **Название метода** | **Аргументы** | **Тип возращаемого значения** | **Назначение** |
| getTasksByDate | date | [SpecificTask] | Получить все конкретный задачи, имеющие date |
| getTaskById | id | SpecificTask? | Получить конкретную задачу по id |
| createTask | id  name  isCompleted  taskDescription  tags  duration  scheduledDate  generalTask | Void | Создать конкретную задачу в БД |
| updateTask | id  name  isCompleted  taskDescription  tags  duration  scheduledDate  generalTask | Void | Обновить конкретную задачу, имеющую id, в БД |
| deleteTask | id | Void | Удалить конкретную задачу, имеющую id, из БД |
| getDelayedTasks | - | [SpecificTask] | Получить все отложенные задачи (у которых нет даты выполения, scheduledDate = nil) |
| deleteAllData | - | Void | Удалить все конкертные задачи из БД |

1. Интерфейс для репозитория общих задач

|  |  |  |  |
| --- | --- | --- | --- |
| **Название метода** | **Аргументы** | **Тип возращаемого значения** | **Назначение** |
| getTasksByDate | date | [GeneralTask] | Получить все общие задачи, имеющие date |
| getTaskById | id | GeneralTask? | Получить общую задачу по id |
| createTask | id  name  isCompleted  taskDescription  tags  deadlineDate  specificTasks | Void | Создать общую задачу в БД |
| updateTask | id  name  isCompleted  taskDescription  tags  deadlineDate  specificTasks | Void | Обновить общую задачу, имеющую id, в БД |
| deleteTask | id | Void | Удалить общую задачу, имеющую id, из БД |
| getExistingDates | - | [Date] | Получить множество уникальных дат дедлайнов всех общих задач в БД |
| deleteAllData | - | Void | Удалить все общие задачи из БД |
| getTasksByName | name | [GeneralTask] | Получить все общие задачи имена, которых начинаются с name, не учитывая регистр |

1. Интерфейс для репозитория тегов

|  |  |  |  |
| --- | --- | --- | --- |
| **Название метода** | **Аргументы** | **Тип возращаемого значения** | **Назначение** |
| getAllTags | - | [Tag] | Получить все теги из БД |
| getTagById | id | Tag? | Получить тег по id |
| deleteAllData | - | Void | Удалить все теги из БД |
| deleteTagById | id | Void | Удалить тег по id |

# ПРИЛОЖЕНИЕ 3

**ТЕРМИНОЛОГИЯ**

|  |  |
| --- | --- |
| **Термин** | **Определение** |
| Тег / тэг | Ассоциированное ключевое слово, относящееся к части информации с указанным цветом. Такие метаданные помогают описать информацию, классифицировать и быстро находить её через поисковый запрос. |
| БД | База данных |
| Дедлайн | Крайний срок выполнения задачи |
| ОС | Операционная система |
| Конкретная задача | Задача, выполнение которой описано для конкретной даты c конкретной продолжительностью |
| Общая задача | Задача, у которой установлена дата дедлайна и которая не имеет продолжительности, время выполнения не привязано ни к чему |
| Отложенная задача | Конкретная задача, дата выполнения которой пока не определена |
| Стабы | Объекты, также называемые заглушками, которые возвращают заранее определенные значения на определенные входные данные. |
| Моки(макеты) | Более сложные объекты, хранящие вызовы методов |
| MVP-модуль | Cовокупность model view presenter для 1 вида view |

# ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Номера листов (страниц) | | | | | Всего листов (страниц в докум.) | № документа | Входящий № сопроводительного докум. и дата | ППодпись | ДДата |
| Изм. | Измененных | Замененных | Новых | Аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |