

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Образовательная программа «Программная инженерия»

СОГЛАСОВАНО
Приглашенный преподаватель
департамента программной инженерии

УТВЕРЖДАЮ
Академический руководитель
Образовательной программы
«Программная инженерия»
Старший преподаватель департамента
программной инженерии

_____ Г.М. Сосновский
«__» _____ 2024 г.

_____ Н.А. Павлочев
«__» _____ 2024 г.

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	

**«Rewind» Приложение для
коллективного медиа-потребления**

Пояснительная записка

ЛИСТ УТВЕРЖДЕНИЯ

RU.17701729.12.20-01 81 01-1-ЛУ

Исполнитель
студент группы БПИ229
_____/ А. Д. Хрулева /
«__» _____ 2024 г.

УТВЕРЖДЕН
RU.17701729.12.20-01 81 01-1-ЛУ

**«Rewind» Приложение для
коллективного медиа-потребления**

Пояснительная записка

RU.17701729.12.20-01 81 01-1

Листов 20

Инв. № подл	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

СОДЕРЖАНИЕ

1.	ВВЕДЕНИЕ	4
1.1.	Наименование программы.....	4
1.2.	Документ, на основании которого ведется разработка	4
2.	НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ	5
2.1.	Назначение программы.....	5
2.1.1.	Функциональное назначение	5
2.1.2.	Эксплуатационное назначение	5
2.2.	Краткая характеристика области применения.....	5
3.	ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ.....	6
3.1.	Постановка задачи на разработку программы.....	6
3.2.	Описание алгоритма работы программы	9
3.2.1.	Архитектура приложения	9
3.2.2.	Реализация сетевого слоя	11
3.2.3.	Генерация изображения для цитаты.....	11
3.2.4.	Добавление участников в группу.....	12
3.2.5.	Шифрование ID группы.....	13
3.2.6.	Автоматическая генерация тегов для фотографий	13
3.3.	Описание и обоснование выбора метода организации входных и выходных данных.....	14
3.3.1.	Описание организации входных и выходных данных	14
3.3.2.	Обоснование выбора метода организации входных и выходных данных.....	14
3.4.	Описание и обоснование выбора состава технических и программных средств	14
3.4.1.	Состав технических и программных средств.....	14
3.4.2.	Обоснование выбора технических и программных средств.....	15
4.	ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ.....	16
4.1.	Ориентировочная экономическая эффективность	16
4.2.	Предполагаемая годовая потребность.....	16

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

- 4.3. Экономические преимущества разработки по сравнению с лучшими отечественными и
зарубежными образцами или аналогами.....16

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ19

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ20

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

1. ВВЕДЕНИЕ

1.1. Наименование программы

Наименование программы: «“Rewind” Приложение для коллективного медиа-потребления».

Наименование программы на английском языке: «“Rewind” Application for Collective Media Consumption».

Краткое наименование программы: «Rewind».

1.2. Документ, на основании которого ведется разработка

Учебный план подготовки бакалавров по направлению 09.03.04 «Программная инженерия» и утвержденная академическим руководителем программы тема курсового проекта.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

2.1. Назначение программы

2.1.1. Функциональное назначение

Приложение предназначено для создания групп и коллекционирования в них общих медиа-объектов в виде фотографий и цитат. Функционал приложения предоставляет возможность подбора случайного медиа-объекта по заданным фильтрам, добавления и удаления медиа-объекта, присваивания ему тегов.

2.1.2. Эксплуатационное назначение

Приложение предназначено для эксплуатации в целях создания групп и совместного накопления в них значимых моментов в форме фотографий и цитат.

Целевой аудиторией приложения являются знакомые между собой люди: друзья, члены семьи, коллеги и все те, кому важно сохранить и иметь возможность вспомнить общие моменты.

Задача приложения состоит в том, чтобы укрепить связь между людьми и привнести разнообразие в их повседневную жизнь. Просмотр случайных воспоминаний может стать интересным и увлекательным занятием для участников.

2.2. Краткая характеристика области применения

Приложение предназначено для создания и поддержания эмоциональной связи между людьми с помощью коллекционирования и просмотра общих ярких моментов в виде фотографий и цитат.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

3.1. Постановка задачи на разработку программы

Приложение должно предоставлять пользователю следующий набор функций:

1. Регистрация/авторизация пользователя:

1.1. При регистрации пользователю необходимо ввести:

- 1.1.1. Адрес электронной почты;
- 1.1.2. Код из 4-ех символов, отправляющийся на указанную ранее почту для ее подтверждения;
- 1.1.3. Пароль;
- 1.1.4. Имя пользователя;

1.2. При авторизации пользователю необходимо ввести:

- 1.2.1. Адрес электронной почты;
- 1.2.2. Пароль;

2. Главный экран со следующими возможностями:

- 2.1. Переход на экран выбранной группы при нажатии соответствующей кнопки;
- 2.2. Переход на экран аккаунта при нажатии соответствующей кнопки;
- 2.3. Быстрый выбор группы с помощью выпадающего списка;
- 2.4. Быстрое создание группы с помощью выпадающего списка;
- 2.5. Просмотр случайно подобранного медиа-объекта в середине экрана, который сменяется другим медиа-объектом при нажатии соответствующей кнопки;
- 2.6. Просмотр аватара имени пользователя, который добавил текущий медиа-объекта в галерею выбранной группы, вместе с датой добавления;
- 2.7. Переход на экран подробной информации конкретного медиа-объекта;
- 2.8. Фильтрация появляющихся медиа-объектов со следующими фильтрами:
 - 2.8.1. По фото;
 - 2.8.2. По цитатам;
 - 2.8.3. По избранным;
- 2.9. Сохранение текущего медиа-объекта на устройство при нажатии соответствующей кнопки;
- 2.10. Добавление медиа-объекта в избранные при нажатии соответствующей кнопки;

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

- 2.11. Переход на экран со всеми медиа-объектами, добавленными в текущую группу, по нажатию кнопки;
3. Экран аккаунта со следующими возможностями:
 - 3.1. Переход на экран всех групп, в которых состоит пользователь;
 - 3.2. Редактирование аватара пользователя;
 - 3.3. Редактирование имени пользователя;
 - 3.4. Редактирование адреса электронной почты;
 - 3.5. Редактирование пароля;
 - 3.6. Просмотр адреса электронной почты технической поддержки приложения для решения возникающих проблем;
 - 3.7. Отправка ссылки на сайт приложения;
 - 3.8. Выбор иконки приложения;
 - 3.9. Выход из аккаунта при дополнительном подтверждении от пользователя;
 - 3.10. Удаление аккаунта при дополнительном подтверждении от пользователя;
4. Экран всех групп со следующими возможностями:
 - 4.1. Поиск группы по ее названию;
 - 4.2. Добавление новой группы при нажатии на соответствующую кнопку;
 - 4.3. Просмотр полного списка групп, в которых состоит пользователь;
5. Экран выбранной группы со следующими возможностями:
 - 5.1. Переход на экран настроек группы;
 - 5.2. Отправка ссылки-приглашения в группу;
 - 5.3. Просмотр сокращенного списка участников группы;
 - 5.4. Переход на экран со всеми участниками группы;
 - 5.5. Просмотр небольшого количества миниатюр медиа-объектов из галереи группы;
 - 5.6. Переход на экран галереи группы;
6. Экран настроек выбранной группы со следующими возможностями:
 - 6.1. Редактирование аватара группы;
 - 6.2. Редактирование названия группы;
 - 6.3. Выход из группы при дополнительном подтверждении от пользователя;
 - 6.4. Удаление группы при дополнительном подтверждении от пользователя (эта возможность доступна только для создателя группы);
7. Экран со всеми участниками группы со следующими возможностями:

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

- 7.1. Поиск участника группы по имени пользователя;
- 7.2. Отправка ссылки-приглашения в группу при нажатии на соответствующую кнопку;
- 7.3. Просмотр полного списка участников группы;
- 8. Экран галереи группы со следующими возможностями:
 - 8.1. Просмотр полной коллекции медиа-объектов группы;
 - 8.2. Просмотр дополнительной информации о медиа-объекте при нажатии на него (п. 9);
 - 8.3. Добавление нового медиа-объекта по нажатию соответствующей кнопки;
 - 8.4. Переход на главный экран по нажатию соответствующей кнопки;
- 9. Просмотр дополнительной информации о медиа-объекте со следующими возможностями:
 - 9.1. Сохранение медиа-объекта на устройство по нажатию соответствующей кнопки;
 - 9.2. Переход на экран подробной информации о медиа-объекте по нажатию соответствующей кнопки;
 - 9.3. Просмотр аватара и имени пользователя, который добавил медиа-объекта в галерею группы, вместе с датой добавления;
 - 9.4. Поделиться медиа-объектом;
 - 9.5. Добавление медиа-объекта в избранные при нажатии соответствующей кнопки;
- 10. Экран подробной информации о медиа-объекте со следующими возможностями:
 - 10.1. Просмотр аватара и имени пользователя, который добавил медиа-объекта в галерею группы, вместе с датой добавления;
 - 10.2. Просмотр тегов со следующим функционалом:
 - 10.2.1. Удаление тега при нажатии соответствующей кнопки;
 - 10.2.2. Добавление тега при нажатии соответствующей кнопки;
 - 10.2.3. Автоматическая генерация тегов с использованием модели YOLOv3Int8LUT [1] при нажатии на соответствующую кнопку;
 - 10.3. Удаление медиа-объекта при дополнительном подтверждении от пользователя;
- 11. Добавление медиа-объекта в галерею группы:

При нажатии на кнопку добавления появляется выпадающий список со следующими надписями:

 - 11.1. «New photo» – добавление фото:
 - 11.1.1. Пользователю необходимо выбрать фото из галереи устройства;

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

11.1.2. Пользователь может сразу добавить теги к новому фото;

11.2. «New quote» – добавление цитаты:

11.2.1. Пользователю необходимо ввести содержание цитаты в окне, которое появляется при нажатии на соответствующую кнопку;

11.2.2. Пользователь может ввести имя автора цитаты;

11.2.3. Пользователю может настроить цвет заднего фона цитаты;

11.2.4. Пользователю может настроить цвет текста цитаты;

11.2.5. Пользователь может сразу добавить теги к новой цитате;

3.2. Описание алгоритма работы программы

3.2.1. Архитектура приложения

Выбор архитектуры приложения – важное решение, которое необходимо принять перед началом кодирования. Корректно выбранная архитектура должна облегчить разработку приложения, его поддержку и расширение в будущем.

Самыми распространёнными архитектурами для реализации iOS-приложения являются MVC (Model-View-Controller) [2], MVVM (Model-View-ViewModel) [3] и MVP (Model-View-Presenter) [4]. К менее распространенным, но также используемым, относятся Clean Swift [5], VIPER (View Interactor Presenter Entity Routing) [6] и TCA (The Composable Architecture) [7]. Все они были рассмотрены, прежде чем было принято финальное решение.

После изучения различных вариантов архитектур были сформулированы следующую критерии для отбора: архитектура должна в достаточной мере на взгляд разработчика декомпонировать код, архитектура должна быть удобной для работы над проектом в одиночку, архитектура должна быть нацелена на разработку с использованием UIKit [8]. В итоге была выбрана архитектура MVP+R (Model-View-Presenter + Router) [9] – она подходит под все перечисленные критерии и в ней достаточно легко разобраться.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

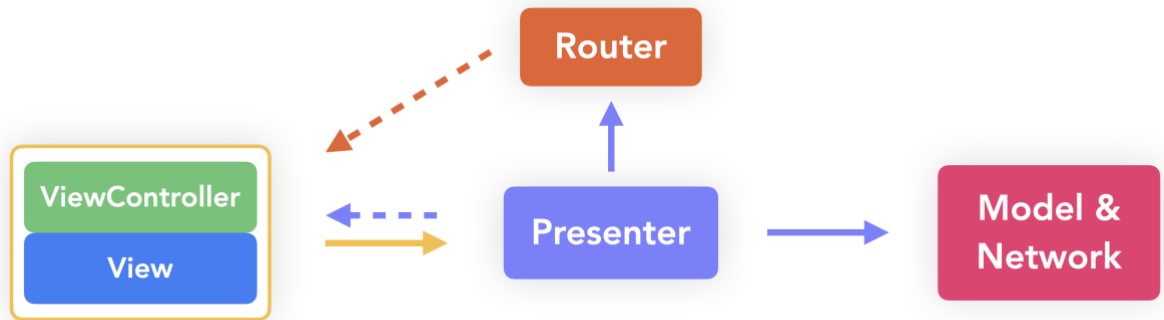


Рисунок 1 – Схема реализованной в проекте архитектуры MVP+R

В проекте архитектура реализована следующим образом: для каждого экрана написаны четыре файла: ViewController, Presenter, Router и Builder.

В файле ViewController описан интерфейс экрана, в нем также содержится сильная ссылка на объект типа Presenter, осуществляется вызов публичных методов данного объекта при взаимодействии пользователя с UI-элементами.

В файле Presenter обрабатываются действия пользователя и осуществляется вся бизнес-логика. Presenter содержит в себе слабую ссылку на объект типа ViewController. После обработки пользовательского действия Presenter вызывает публичные методы объекта типа ViewController для обновления отображаемого интерфейса. Presenter также содержит в себе сильную ссылку на объект типа Router. Presenter вызывает публичные методы данного объекта в случаях, когда необходимо осуществить навигацию на другой экран или презентацию другого экрана с текущего.

В файле Router описана логика навигации между текущим экраном и другими, а также логика презентации других экранов с текущего. Router содержит в себе слабую ссылку на объект типа ViewController для реализации логики навигации и презентации.

В файле Builder содержится единственный статический метод build, возвращающий объект типа ViewController. Данный метод нужен для инициализации вышеописанных компонент так, чтобы каждый элемент знал про все нужные ему.

Для решения данной задачи было написано более восьми тысяч строк кода.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

3.2.2. Реализация сетевого слоя

Сетевой слой является важной частью приложений, в которых необходимо осуществлять взаимодействие с удаленным сервером. Сетевой слой позволяет приложению отправлять данные на сервер, получать данные с сервера по запросу и осуществлять их первичную обработку.

Для решения данной задачи было обнаружено два подхода: писать запросы в тех файлах Presenter, в которых они потом вызываются, или определить отдельный класс и описывать в нем статические методы, которые будут осуществлять запросы. Для лучшего структурирования кода был выбран второй подход.

В проекте для реализации сетевого слоя написан класс `NetworkService`, в котором содержатся статические методы, которые осуществляют различные запросы к серверу. Также для реализации сетевого слоя была написана структура `NetworkResponse`, призванная унифицировать тип и структуру получаемого от сервера ответа. Помимо этого, из-за того, что сервер может возвращать в качестве ответа строку, JSON или JSON Array, в расширении класса `NetworkService` были написаны обработчики для каждого типа ответа: `processStringResponse`, `processJSONResponse`, `processJSONArrayResponse`.

3.2.3. Генерация изображения для цитаты

В функциональных требованиях к проекту указано, что пользователь должен иметь возможность добавлять в галерею текущей группы цитаты. Так как помимо цитат пользователи могут добавлять в галерею фотографии, то для более удобной работы с объектами было принято решение, что для цитат необходимо генерировать изображения. Построение таких изображений выполняется из задаваемого пользователем текста цитаты (обязательное поле) и автора (необязательное поле). Так же пользователю предоставляется возможность выбрать цвет текста и фона генерируемого изображения.

В ходе изучения информации по данной теме были обнаружены несколько возможных вариантов решения. В рамках Core Graphics framework [10] для реализации поставленной задачи могут быть использованы `CGContext`, получаемый из функции `UIGraphicsGetCurrentContext`, или `UIGraphicsRenderer`, получаемый из функции `UIGraphicsImageRenderer`. Другим вариантом является Core Text framework.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

В итоге для решения поставленной задачи был выбран вариант `UIGraphicsRenderer`, так как он предоставляет необходимый функционал и является достаточно простым в освоении.

В проекте данный вариант реализован в виде отдельного класса `QuoteImageGenerator`. В нем описан единственный статический метод – `convertToImage`, который возвращает изображение для цитаты, если его удалось сгенерировать.

3.2.4. Добавление участников в группу

Группы, объединяющие участников, являются ключевыми логическими ячейками приложения. Для создания групп из более чем одного участника – владельца группы – необходимо было реализовать возможность добавления других пользователей в группу.

В поисках решения задачи было рассмотрено два подхода: реализовать возможность поиска конкретного пользователя среди всех зарегистрированных и добавить возможность отправлять ему приглашение в группу; реализовать возможность поделиться пригласительной ссылкой в группу, после нажатия на которую пользователь автоматически переходит в приложение, происходит присоединение в группу и в случае успеха происходит переход на экран группы. Для решения задачи был выбран второй вариант, как более подходящий и не нагружающий интерфейс дополнительными кнопками и экранами.

Для генерации пригласительной ссылки было выявлено два способа: `Deep Link` и `Universal Link`. Вначале был реализован первый способ, однако быстро выявился существенный недостаток – сторонние приложения (не разработанные компанией Apple) не распознают `Deep Link` как ссылку, следовательно не делают её кликабельной. Эта особенность могла бы отрицательно сказаться на пользовательском опыте, поэтому решено было внедрять в проект `Universal Link`.

`Universal Link` предоставляет возможность переходить в приложение «Rewind» из всех сторонних мобильных приложений, распознающих URL схему `https://`.

Для генерации пригласительной ссылки в проекте был создан класс `JoinLinkService` и написан статический метод `createJoinLink`, который принимает ID группы и возвращает сгенерированную ссылку-приглашение, если шифрование ID прошло успешно.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

3.2.5. Шифрование ID группы

Для формирования ссылки-приглашения в неё обязательно нужно включить идентификатор группы. Например, в качестве параметра URL. Это необходимо, чтобы при переходе по ссылке приложение понимало, запрос на присоединение к какой группе необходимо отправить на сервер.

Формировать ссылку-приглашение таким образом, чтобы ID группы в ней отображался в оригинальном своем виде, то есть в виде числа, небезопасно. Так как тогда любой человек сможет подставлять в ссылку другие числа и присоединяться в группы, в которые его не приглашали. Таким образом, возникает необходимость шифрования ID группы.

Основная задача, которую необходимо было выполнить для достижения поставленной цели, заключалась в поиске метода шифрования, с помощью которого можно было бы зашифровать ID перед формированием ссылки, а затем расшифровывать при переходе по ссылке в приложение. То есть метод шифрования должен быть реверсивным.

В процессе поисков был обнаружен AES (Advanced Encryption Standard) [11] – симметричный алгоритм блочного шифрования. Симметричное шифрование означает, что для зашифрования и расшифрования используется один и тот же ключ. Блочное шифрование означает, что перед зашифрованием данные разбиваются на блоки фиксированного размера.

Реализация данного метода шифрования на языке Swift была взята из Интернета. Статические методы зашифрования и расшифрования были описаны в классе JoinLinkService.

3.2.6. Автоматическая генерация тегов для фотографий

Одним из пунктов в функциональных требованиях к проекту указана возможность автоматической генерации тегов для фотографий.

В процессе изучения данной темы выяснилось, что подходящим инструментом для решения данной задачи может быть Core ML framework [12]. Компания Apple предоставляет на выбор несколько вариантов предварительно обученных ML-моделей с различными акцентами. К примеру, MobileNetV2, SqueezeNet, Resnet50 – нейронные сети, которые классифицирует доминирующий объект на изображении. Или YOLOv3, YOLOv3-Tiny –

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

нейронные сети, классифицирующие 80 различных объектов, присутствующих на изображении. Все перечисленные ML-модели предоставляются в нескольких вариантах глубины и точности обучения.

В рамках данного проекта была выбрана предварительно обученная ML-модель YOLOv3Int8LUT, оптимизированная – по данным Apple – с использованием 8-битного квантования. Выбор именно этой модели обоснован её относительно небольшим размером в 62 Мб, что позволяет не сильно увеличивать размер всего приложения, а также даёт возможность разместить весь код, включая модель, в GitHub-репозитории без использования платных планов доступа.

Выбранное решение реализовано в проекте в виде класса YOLOv3Model, в котором описана функция, принимающая данные изображения и после их обработки вызывающая замыкание, в которое передается результат.

3.3. Описание и обоснование выбора метода организации входных и выходных данных

3.3.1. Описание организации входных и выходных данных

Входные данные приложения – это информация, получаемая от пользователя при взаимодействии с интерфейсом: почта, код подтверждения, пароль, имя пользователя, фотография пользователя, имя группы, фотография группы, фотография для добавления в галерею, тексты цитаты, текст автора, текста тега.

Выходные данные приложения – это отображаемые данные интерфейса, который конфигурируется на основе входных данных.

3.3.2. Обоснование выбора метода организации входных и выходных данных

Для организации входных и выходных данных был выбран метод, основанный на принципах удобства использования, эффективности хранения и передачи данных.

3.4. Описание и обоснование выбора состава технических и программных средств

3.4.1. Состав технических и программных средств

Для корректной работы клиентской части приложения необходимо мобильное устройство iPhone с операционной системой iOS 14.0 и выше, минимальным объемом

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

оперативной памяти – 2 Гб, минимальным объемом свободной памяти устройства – 200 Мб. Мобильное устройство должно иметь доступ к Интернету.

3.4.2. Обоснование выбора технических и программных средств

Выбор технических средств основан на анализе использования памяти устройства, который был проведен с помощью инструментов, предоставляемых средой разработки Xcode.

Выбор программных средств, в частности минимальной версии операционной системы, обоснован использованием в коде инструментов, которые требуют определенной версии операционной системы.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

4. ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

4.1. Ориентировочная экономическая эффективность

В рамках проекта расчет экономической эффективности программного продукта не производился.

4.2. Предполагаемая годовая потребность

Данный программный продукт будет востребована в кругу друзей, знакомых или коллег, которые хотят коллекционировать и вспоминать яркие моменты их общения.

4.3. Экономические преимущества разработки по сравнению с лучшими отечественными и зарубежными образцами или аналогами

Цель приложения «Rewind» – предоставить возможность создания групп людей и совместного накопления в них значимых моментов в форме фотографий и цитат.

В таблице 1 представлен сравнительный анализ приложения «Rewind» с лучшими аналогами.

Таблица 1 – Сравнительный анализ (прямые аналоги)

Категория	Критерии	Аналоги			Rewind
		Locket	Фотопленка iOS	TimeHop	
Доступность	Наличие платной подписки	+	-	-	-
Дизайн	Смена иконки приложения	-	-	-	+
Пользование приложением с друзьями	Возможность объединяться в группы	-	-	-	+
	Общая галерея медиа-объектов у нескольких пользователей	+	+	-	+
	Возможность видеть автора медиа-объекта	+	+	-	+

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

Продолжение таблицы 1

Пользование приложением с друзьями	Просмотр воспоминаний происходит по случайным датам	-	+	-	+
	Возможность удалить медиа-объект	-	+	-	+
Фильтрация медиа-объектов	По типу медиа-объекта	-	+	-	+
	По избранным	-	+	-	+
Особый функционал	Добавление цитаты с помощью генерации изображения с текстом цитаты и автором	-	-	-	+
	Автоматическая генерация тегов при нажатии на кнопку	-	-	-	+
	Возможность пометить медиа-объект избранным	-	+	-	+

В таблице 2 представлен сравнительный анализ «Rewind» с косвенными аналогами.

Таблица 2 – Сравнительный анализ (косвенные аналоги)

Категория	Критерии	Аналоги				Rewind
		Telegram	VK	Yandex Disk	Google Disk	
Доступность	Наличие платной подписки	+	+	+	+	-
Дизайн	Смена иконки приложения	+	+	-	-	+
Пользование приложением с друзьями	Возможность объединяться в группы	+	+	+	+	+

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

Продолжение таблицы 2

Пользование приложением с друзьями	Общая галерея медиа-объектов у нескольких пользователей	+	+	+	+	+
	Возможность видеть автора медиа-объекта	+	+	+	+	+
	Просмотр воспоминаний происходит по случайным датам	-	-	-	-	+
	Возможность удалить медиа-объект	+	-	+	+	+
Фильтрация медиа-объектов	По типу медиа-объекта	-	-	+	+	+
	По избранным	-	-	-	+	+
Особый функционал	Добавление цитаты в галерею с помощью генерации изображения с текстом цитаты и автором	-	-	-	-	+
	Автоматическая генерация тегов при нажатии на кнопку	-	-	-	-	+
	Возможность пометить медиа-объект избранным	-	+	+	+	+

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Apple Developer [Электронный ресурс] / Core ML Models. Режим доступа: <https://developer.apple.com/machine-learning/models/> (Дата обращения: 02.04.2024).
2. Википедия [Электронный ресурс] / Model-View-Controller. Режим доступа: <https://ru.wikipedia.org/wiki/Model-View-Controller> (Дата обращения: 02.04.2024).
3. Википедия [Электронный ресурс] / Model-View-ViewModel. Режим доступа: <https://ru.wikipedia.org/wiki/Model-View-ViewModel> (Дата обращения: 02.04.2024).
4. Википедия [Электронный ресурс] / Model-View-Presenter. Режим доступа: <https://ru.wikipedia.org/wiki/Model-View-Presenter> (Дата обращения: 02.04.2024).
5. Netguru [Электронный ресурс] / Clean swift (VIP) iOS Architecture Pattern. Режим доступа: <https://www.netguru.com/blog/clean-swift-ios-architecture-pattern> (Дата обращения: 02.04.2024).
6. Medium [Электронный ресурс] / Understanding VIPER Architecture. Режим доступа: <https://medium.com/@pinarkocak/understanding-viper-pattern-619fa9a0b1f1> (Дата обращения: 02.04.2024).
7. GitHub [Электронный ресурс] / swift-composable-architecture. Режим доступа: <https://github.com/pointfreeco/swift-composable-architecture> (Дата обращения: 02.04.2024).
8. Apple Developer [Электронный ресурс] / UIKit. Режим доступа: <https://developer.apple.com/documentation/uikit> (Дата обращения: 02.04.2024).
9. Habr [Электронный ресурс] / MVP with Router. Режим доступа: <https://habr.com/ru/articles/749700/> (Дата обращения: 02.04.2024).
10. Apple Developer [Электронный ресурс] / Core Graphics. Режим доступа: <https://developer.apple.com/documentation/coregraphics> (Дата обращения: 02.04.2024).
11. Википедия [Электронный ресурс] / AES (стандарт шифрования). Режим доступа: [https://ru.wikipedia.org/wiki/AES_\(%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82_%D1%88%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F\)](https://ru.wikipedia.org/wiki/AES_(%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82_%D1%88%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)) (Дата обращения 02.04.2024).
12. Apple Developer [Электронный ресурс] / Core ML framework. Режим доступа: <https://developer.apple.com/documentation/coreml> (Дата обращения: 02.04.2024).

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

[illegible]

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.12.20-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата