

Санкт-Петербургский государственный университет

Математика и компьютерные науки

Дисциплина: Виртуализация и облачные технологии

Лабораторная работа N3

Выполнил студент:

Ибрагимов Артем Рустамович

Преподаватель:

Кривоносов Егор Дмитриевич

Санкт-Петербург

2024 г.

Содержание

Проблема	3
Решение	3
Сервисы	4
Выбор провайдера	4
Список сервисов	5
Диаграмма компонент	6
Целевая нагрузка	6
Масштабирование	6
Диаграмма вариантов использования	8
Диаграмма базы данных	17
Реализация	18
Создание и настройка сервисов	18
Не реализованная функциональность	19
Вывод	20

Проблема

Существует множество инструментов для помощи в коллаборативной разработке, но они не расширяемы. То есть они все сделаны в рамках какой то одной популярной методологии разработки, например скрам. Очень сложно использовать эти инструменты в каких то больших масштабах. Например, невозможно построить управление вообще всей компанией через один такой инструмент. Плюс возможно мы не хотим делиться данными нашей компании со сторонними приложениями, даже при условии всяких соглашений о неразглашении. Все равно мы не можем быть уверены в безопасности. Поэтому хочется иметь какой то свой внутренний инструмент для помощи в разработке. Который можно будет расширять и которым удобно пользоваться настолько, что потом потенциально можно будет расширить этот инструмент для управления всей компании.

Решение

Мы напишем внутренний сервис для хранения задач, календаря встреч и информации по сотрудникам. Будет возможность архивировать задачи и сохранять их для накопления статистики. Также будет поддержка поэтапной разработки. Можно будет настроить какой то отрезок времени, сколько будет длиться один этап. И дальше в этот этап можно будет добавлять задачи. Когда этап закончится, все завершённые задачи будут архивироваться, а просроченные будут отмечаться как нибудь и переходить на следующий этап. У задачи будет много всяких атрибутов. Ее автор, исполнитель, описание, и так далее. Также можно будет писать комментарии к задаче. Но кто может писать комментарии, и вообще кто может редактировать задачу будет контролироваться системой прав доступа среди пользователей. Она будет контролировать безопасность даже внутри компании.

Еще будет возможность записывать заметки во время встречи и там же быстро назначать последующие встречи между сотрудниками, чтобы обсудить какие то вопросы поподробнее. Будет календарь, в котором можно будет отметить все встречи, дежурства и так далее. Также можно будет настроить систему поступления новых задач. Каждый сотрудник на протяжении всего этапа может занести задачу в систему. И она будет рассмотрена на планировании задач к следующему этапу. Большинство этих функций было бы очень сложно удобно реализовать с существующими инструментами.

Сервисы

Для хранения всех текущих задач и их архивации потребуется сервис базы данных. Также я хочу использовать виртуальную машину для развертки на ней сервера, который будет предоставлять API для работы с задачами. Еще для каких-то автономных, регулярных задач я хочу использовать функции как сервис.

Выбор провайдера

Основные критерии выбора - доступность из России, цена использования, легкость использования (документация) и многообразие предоставляемых сервисов.

Основной выбор я делал из четырех сервисов:

	Доступность	Цена, р/день	Документация, Простота
Yandex Cloud	Yes	108	5/5
Firebase	Yes	0	5/5
MTS Cloud	Yes	115	0/5
VK Cloud	Yes	167	4/5

Цена рассчитывалась по минимальной конфигурации, которая нужна для работы сервиса. Подробнее про каждое облако:

Так как я хочу написать мобильное приложение, один из очевидных и очень удобных облачных сервисов - это [Firebase](#). Он основан на Google Cloud и предоставляет очень много способов авторизации прямо из коробки, то есть очень легко сделать кучу надежных способов авторизации прямо в приложении, вообще при этом не прикладывая никаких усилий. Также он дает разные варианты

для базы данных. Основной минус в цене. Ну точнее в доступности, поскольку это Google Cloud, то оплатить из России нельзя. Есть бесплатный тариф, он включает ту конфигурацию, которая мне нужна. Также Firebase будет очень хорошо совместим с мобильным приложением.

Еще есть [Yandex Cloud](#). Он дает много бесплатных сервисов (до какой то большой квоты), но за некоторые все равно придется платить. Также у Yandex Cloud очень подробная документация + очень много сервисов, в том числе и бесплатных, поэтому очень легко подстраивать их под то, что мне нужно сделать. Yandex Cloud дает грант новым пользователям, с которого можно было бы оплатить эти 108 рублей в день. Но я этот грант уже потратил, поэтому если выбирать Yandex Cloud, то придется платить по-настоящему.

[MTS Cloud](#) еще один доступный из России провайдер. Но там гораздо сложнее разобраться и у него гораздо меньше сервисов, чем у Yandex Cloud. И даже если найти там ссылку на документация, первое что они говорят - это что их сервис Compute Cloud (тот что мне нужен) закрыт для новых пользователей. Поэтому для меня такой провайдер вообще не подходит.

По итогу на данный момент я выбираю сервис Firebase. Потому что это самое простое решение, как мне сейчас кажется. Для моих целей должно хватить бесплатного тарифа, но если все таки окажется, что его не хватит, то придется перейти на Yandex Cloud. Но это не должно быть сложно, потому что приложение у меня уже будет и надо будет просто написать небольшой сервер для его поддержки.

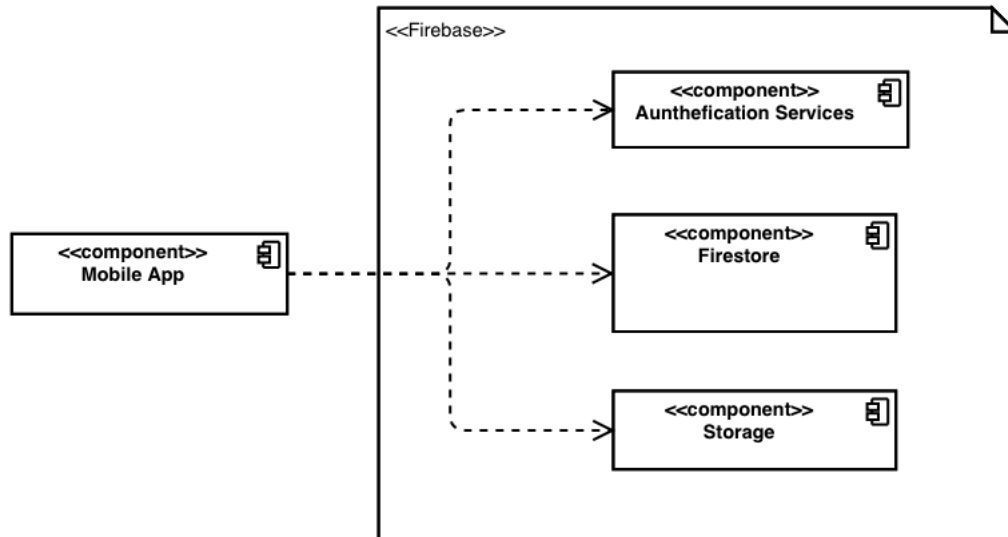
Список сервисов

Ну в первую очередь я буду использовать сервис авторизации. Нужно как-то отличать всех сотрудников компании, которые будут пользоваться приложением, по этому каждый будет регистрироваться по своему email.

Также нужно хранить где-то все созданные задачи и распределять их по разным категориям. Для этого я буду использовать Firestore - основную базу данных у Firebase.

Для третьего сервиса можно использовать хранилище медиафайлов. Чтобы сохранять аватарки пользователей и скриншоты.

Диаграмма компонент



Архитектура очень простая, на данный момент из моего кода будет только мобильное приложение, которое пользуется серверами Firebase чтобы хранить данные и авторизировать пользователей. Никакой бизнес логики поверх этого не будет тк пока что никакие функции приложения в этом не нуждаются.

Целевая нагрузка

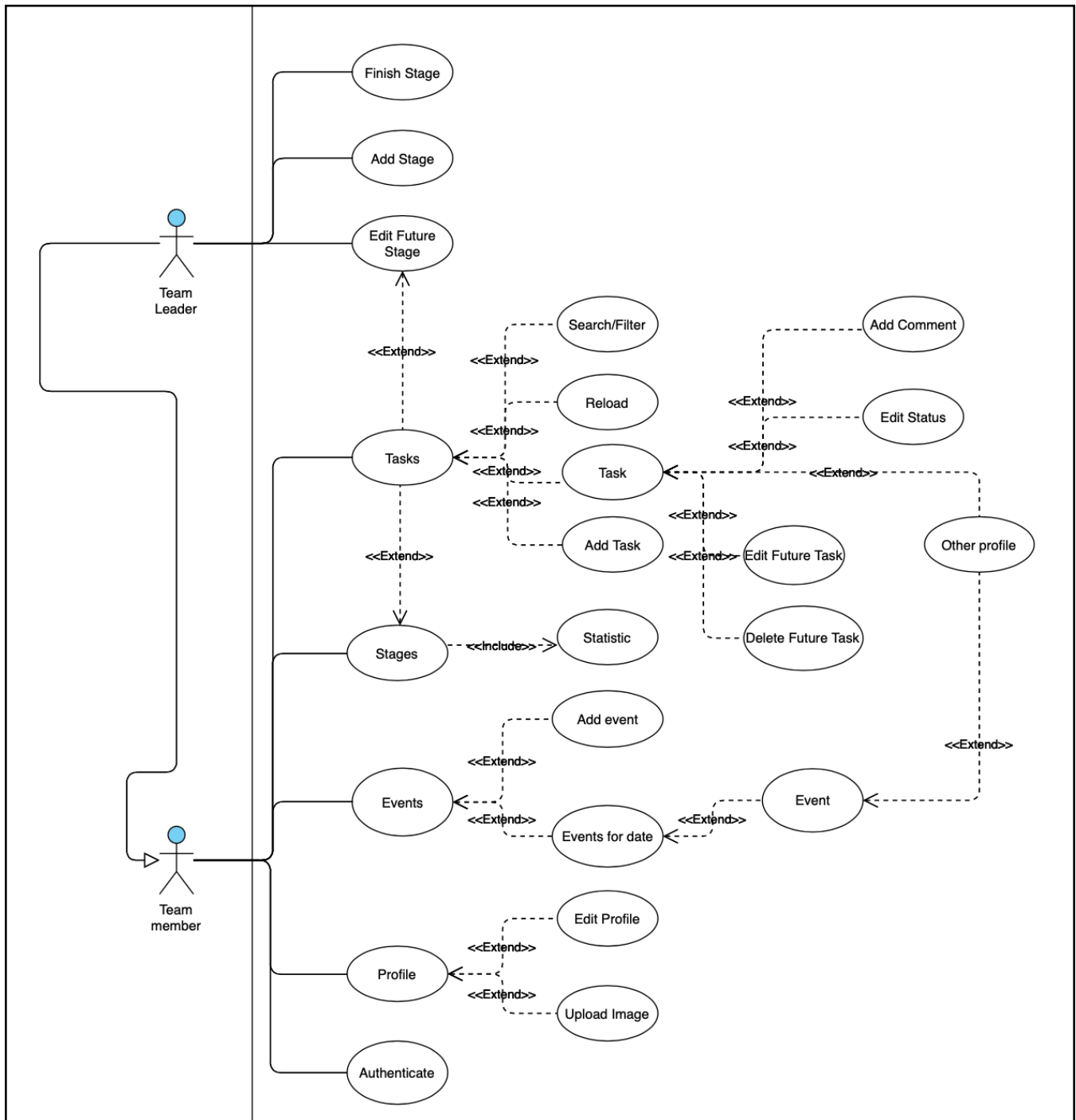
Пока что приложение будет рассчитано только на одну команду разработчиков, поэтому нагрузка будет очень небольшая, если команда состоит из 7 разработчиков, каждый из них просматривает свои задачи и встречи раз 10 - 12 в день то за неделю они сделают около 504 запроса (в среднем один заход считаем за 10 запросов к базе).

Масштабирование

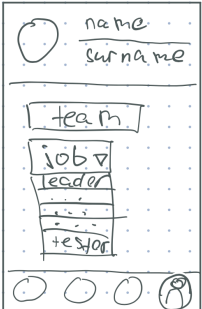
В дальнейшем подразумевается, что системой будет пользоваться вся компания для централизованного управления, в которой могут быть десятки тысяч сотрудников. Для этого надо будет сделать продвинутую систему прав доступа, и к каждой функциональности приписывать минимальные права, которые нужны

чтобы ей пользоваться. Также конечно нужны будут огромные объемы базы данных для хранения и архивирования всей информации. Можно сделать приложение автономным, то есть поставить уже свои сервера и хранилища - это будет более безопасно. Однако легче будет просто продолжать использовать Firebase. Потому что в своих [лимитах](#), они указывают максимум в 100 тысяч запросов в секунду и никаких лимитов на размер хранения данных, этого вполне достаточно даже для компании в 20 тысяч сотрудников. При таком варианте можно будет оставить систему безсерверной (без сервера...), если использовать дополнительно облачные функции - еще один сервис Firebase, для всего, что должна делать сама система, ведь ей не придется делать какие-то сложные вычисления. Конечно при этом цена возрастет, если выставить все нужные функции на максимум, то получается где-то 8000\$ в месяц. Но это вполне нормально для огромной компании.

Диаграмма вариантов использования



ID	Authenticate
Описание	Пользователь должен зарегистрироваться с выданным ему email и паролем.
Акторы	Любой пользователь
Основной поток	<ol style="list-style-type: none"> 1. Ввод правильного email 2. Ввод правильного пароля 3. Ура, мы вошли
Альтернативный поток	<ol style="list-style-type: none"> 1. Ввод правильного email 2. Ввод неправильного пароля 3. Получаем сообщение об ошибке
Постусловия	Пользователь вошел в систему.

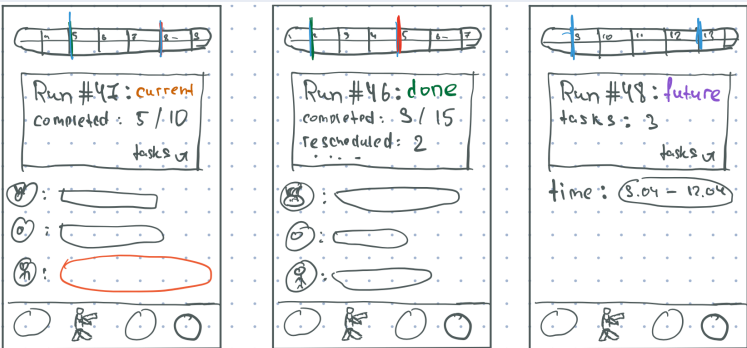
ID	Profile, Edit Profile, Upload Image
Описание	Можно посмотреть данные своего профиля, можно загрузить новое фото и можно отредактировать свою имя / фамилию / должность.
Акторы	Любой пользователь
Основной поток	<ol style="list-style-type: none"> 1. Перейти на вкладку профиль. 2. Extension: Загрузить картинку 3. Extension: Выбрать должность из списка 4. Подтвердить, нажав на кнопку confirm.
Альтернативный поток	<ol style="list-style-type: none"> 1. Перейти на вкладку профиль. 2. Extension: Загрузить картинку 3. Extension: Выбрать должность из списка 4. Подтвердить, нажав на кнопку confirm. 5. Ошибка подключения, выводится сообщение об ошибке
Предусловия	Пользователь вошел в систему
Постусловия	Данные о пользователе изменены
Интерфейс	

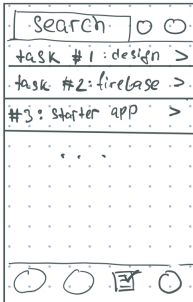

ID	Events, Events by date
Описание	Можно смотреть встречи во вкладке календарь. Можно нажать на дату и посмотреть список всех встреч, назначенных в эту дату
Акторы	Любой пользователь
Предусловия	Пользователь вошел в систему
Основной поток	<ol style="list-style-type: none"> 1. Перейти на вкладку календарь. 2. Extension: Add Event 3. Extension: Нажать на дату <ol style="list-style-type: none"> 2.1. Отобразится список встреч на эту дату 2.2. Extension: Event
Интерфейс	
ID	Event
Описание	Можно подробно посмотреть отдельную встречу: когда, ее участников, что будем обсуждать.
Акторы	Любой пользователь
Предусловия	Пользователь вошел в систему
Основной поток	<ol style="list-style-type: none"> 1. Выбрать встречу из списка 2. Отобразится информация по встрече 3. Extension: Other profile (нажать на профиль одного из участников)

ID	Add Event
Описание	Можно создать встречу, указав ее дату, время, участников и описание.
Акторы	Любой пользователь
Предусловия	Пользователь зарегистрирован
Основной поток	<ol style="list-style-type: none"> 1. На вкладке календарь, нажать на кнопку создать встречу 2. Корректно заполнить поля для встречи 3. Нажать на кнопку создать
Альтернативный поток	<ol style="list-style-type: none"> 1. На вкладке календарь, нажать на кнопку создать встречу 2. Некорректно заполнить поля для встречи (например время окончания будет раньше времени начала) 3. Кнопка создать заблокирована.
Постусловия	Добавлена новая встреча

ID	Other profile
Описание	Можно смотреть профили других пользователей
Акторы	Любой пользователь
Предусловия	Пользователь зарегистрирован
Основной поток	<ol style="list-style-type: none"> 1. Нажать на имя пользователя 2. Отобразится его профиль с картинкой, именем, фамилией и должностью

ID	Stages, Statistic
Описание	Можно посмотреть список всех этапов, которые пройдены, либо текущий этап, либо будущие этапы. Для выбранного этапа сверху показывается интервал времени, когда тот проходит.
Акторы	Любой пользователь
Предусловия	Пользователь зарегистрирован
Основной поток	<ol style="list-style-type: none"> 1. Перейти на вкладку этап 2. Отображается информация по текущему этапу (количество задач и тд) 3. Extension: Tasks (можно посмотреть список задач определенного этапа) 4. Includes: Statistic, на том же экране можно посмотреть статистику по каждому пользователю, кто сколько задач сделал за этот этап

Альтернативный поток	<ol style="list-style-type: none"> 1. Перейти на вкладку этап 2. Этапы еще не загрузились или их нет, отображается надпись, что этапов нет.
Интерфейс	
ID	Tasks, Search/Filter, Reload
Описание	Для любого этапа можно посмотреть список задач, а также отфильтровать их, поискать по названию, обновить из базы и посмотреть детали каждой задачи. Задачи для текущего этапа также отображаются на вкладке Задачи.
Акторы	Любой пользователь
Предусловия	Пользователь зарегистрирован
Основной поток	<ol style="list-style-type: none"> 1. Перейти на вкладку задачи или нажать на кнопку задачи в выбранном этапе. 2. Отображается список задач 3. Extension: Search <ol style="list-style-type: none"> 3.1. В поле для поиска вбить запрос. 3.2. Отобразятся только задачи по имени попадающие под запрос 4. Extension: Filter <ol style="list-style-type: none"> 4.1. Нажать на кнопку фильтра и выбрать тип фильтра из меню. 4.2. Отобразятся только задачи, которые подходят под фильтр 5. Extension: (Reload) <ol style="list-style-type: none"> 5.1. Потянуть вверх 5.2. Задачи перегрузятся из базы данных 6. Extension: Delete Future Task
Альтернативный поток	<ol style="list-style-type: none"> 1. Перейти на вкладку задачи или нажать на кнопку задачи в выбранном этапе. 2. Задачи еще не успели загрузиться или соединения нету 3. Вместо задач, показывается текст говорящий об этом

Интерфейс	
ID	Task, Edit Status
Описание	Можно посмотреть детали о задаче, ее описание, автора, исполнителя и тд
Акторы	Любой пользователь
Предусловия	Пользователь зарегистрирован, Edit Status: пользователь должен быть либо лидером; либо автором, либо исполнителем задачи
Основной поток	<ol style="list-style-type: none"> 1. Нажать на выбранную задачу 2. Отображается информация по задаче (название, описание, автор, исполнитель, статус и комментарии) 3. Extension: Edit Status <ol style="list-style-type: none"> 3.1. Исполнитель/Автор нажимает на статус задачи 3.2. И выбирает новый статус из вариантов 3.3. Задача имеет новый статус 4. Extension: Other profile, нажать на автора или исполнителя 5. Extension: Comment, нажать на кнопку комментировать
Альтернативный поток	<ol style="list-style-type: none"> 1. Нажать на выбранную задачу 2. Отображается информация по задаче (название, описание, автор, исполнитель, статус и комментарии) 3. Extension: Edit Status <ol style="list-style-type: none"> 3.1. Не Исполнитель/Автор нажимает на статус задачи 3.2. Ничего не происходит тк у него нет прав
Постусловия	EditStatus: статус задачи изменен
Интерфейс	

ID	Add Comment
Описание	Пользователь может прокомментировать задачу из незавершенного этапа
Акторы	Любой пользователь
Предусловия	Пользователь зарегистрирован, выбранная задача либо из текущего, либо из будущего этапа
Основной поток	<ol style="list-style-type: none"> 1. В деталях задачи нажать на кнопку комментировать 2. Открывается экран, где можно набить текст комментария 3. Вбить комментарий и нажать на кнопку готово 4. Комментарий добавился к задаче
Альтернативный поток	<ol style="list-style-type: none"> 1. В деталях задачи нажать на кнопку комментировать 2. Открывается экран, где можно набить текст комментария 3. Если оставить комментарий пустым, то нажать на кнопку готово нельзя
Постусловия	Добавлен новый комментарий к задаче

ID	Edit Future Task
Описание	Если задача из будущего этапа и у пользователя есть права, он может редактировать созданную задачу (описание, исполнителя и статус)
Акторы	Любой пользователь
Предусловия	Пользователь зарегистрирован и он либо лидер, либо автор задачи, которая сама из будущего этапа
Основной поток	<ol style="list-style-type: none"> 1. Перейти на экран деталей задачи 2. Нажать на кнопку Редактировать 3. Отредактировать описание задачи 4. Поменять исполнителя на другого пользователя 5. Нажать на кнопку Готово
Постусловия	Задача с новыми данными

ID	Add Task
Описание	Пользователь может добавлять задачи в будущий этап
Акторы	Любой пользователь
Предусловия	Пользователь зарегистрирован

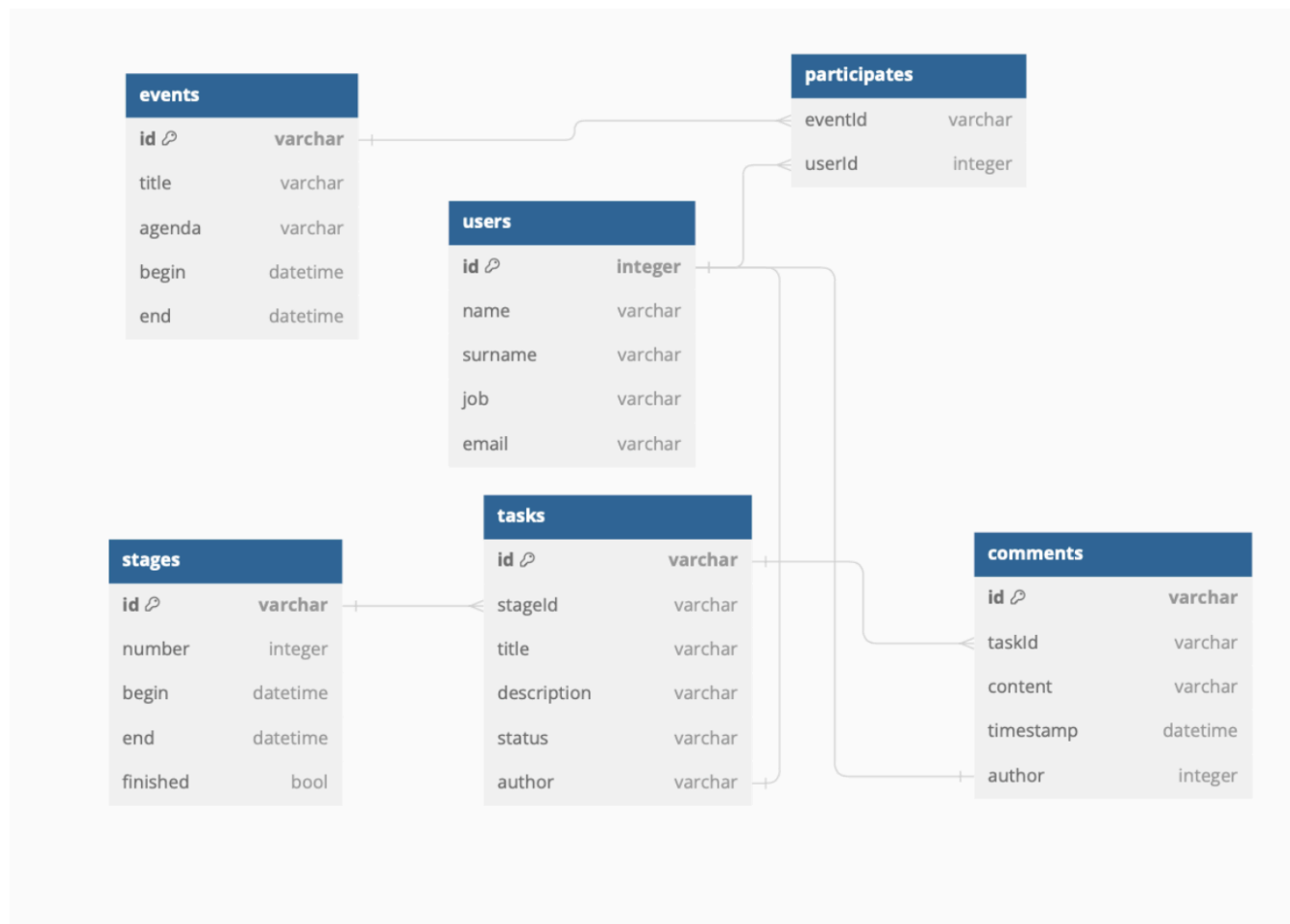
Основной поток	<ol style="list-style-type: none"> 1. На вкладке этап, выбрать будущий этап 2. Перейти в его задачи 3. Нажать на кнопку добавить 4. Открывается экран, где можно выбить всю информацию о задаче (название, описание, исполнитель и статус) 5. Нажать готово
Альтернативный поток	<ol style="list-style-type: none"> 1. На вкладке этап, выбрать будущий этап 2. Перейти в его задачи 3. Кнопки добавить задачу нету
Постусловия	Добавлена новая задача

ID	Delete Future Task
Описание	Если задача из будущего и пользователь правильный, то он может удалить задачу
Акторы	Любой пользователь
Предусловия	Пользователь зарегистрирован, он либо лидер, либо автор задачи, а сама задача из будущего этапа
Основной поток	<ol style="list-style-type: none"> 1. Открыть список задач для будущего этапа 2. Смахнуть вправо на выбранной задаче 3. Задача удалена
Альтернативный поток	<ol style="list-style-type: none"> 1. Открыть список задач для будущего этапа 2. Смахнуть вправо на выбранной задаче 3. Ошибка соединения, показывается сообщение об ошибке, задача не удаляется
Постусловия	Задача удалена

ID	Add Stage
Описание	Лидер может добавить новый этап, дав ему время начала и конца
Акторы	Лидер
Предусловия	Пользователь зарегистрирован как лидер
Основной поток	<ol style="list-style-type: none"> 1. Перейти на вкладку этап 2. Нажать кнопку Новый Этап 3. Ввести дату начала и дату конца, которая \geq даты начала, а также больше текущей даты 4. Нажать готово, этап создан

Альтернативный поток	<ul style="list-style-type: none"> 1. Перейти на вкладку этап 2. Нажать кнопку Новый Этап 3. Ввести неправильную дату начала или конца 4. Нельзя нажать готово
Постусловия	Добавлен новый этап
ID	Edit Future Stage
Описание	Лидер может также редактировать созданный этап, если он еще не начался.
Акторы	Лидер
Предусловия	Пользователь зарегистрирован как лидер
Основной поток	<ul style="list-style-type: none"> 1. Перейти на вкладку этап 2. Выбрать будущий этап 3. Нажать на кнопку редактировать 4. Изменить даты 5. Нажать готово
Альтернативный поток	<ul style="list-style-type: none"> 1. Перейти на вкладку этап 2. Выбрать текущий или прошлый этап 3. Нельзя нажать на кнопку редактировать
Постусловия	Этап отредактирован
ID	Finish Stage
Описание	Лидер может закончить текущий этап, при этом все задачи этого этапа которые были в активном статусе переходят в статус просрочено.
Акторы	Лидер
Предусловия	Пользователь зарегистрирован как лидер, есть этап который можно закончить.
Основной поток	<ul style="list-style-type: none"> 1. Перейти на вкладку этап 2. Выбрать текущий этап 3. Нажать на кнопку завершить
Постусловия	Этап завершен

Диаграмма базы данных



Реализация

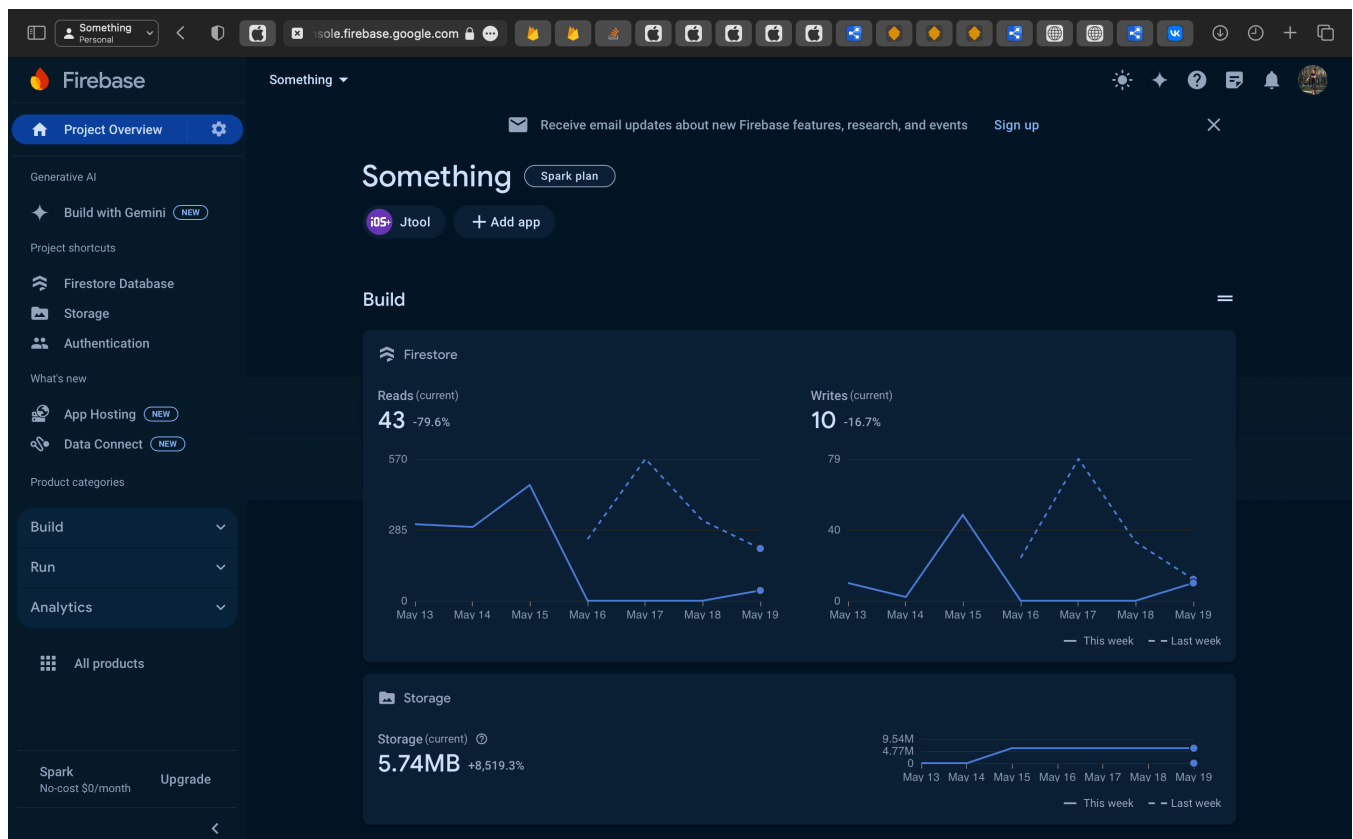
Создание и настройка сервисов

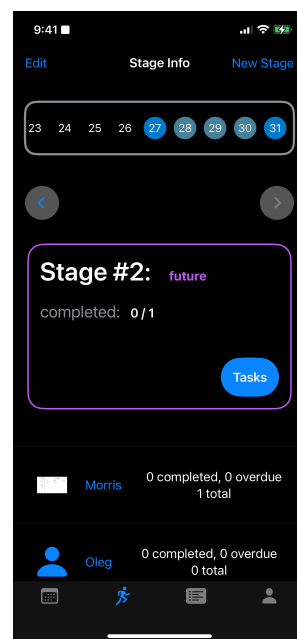
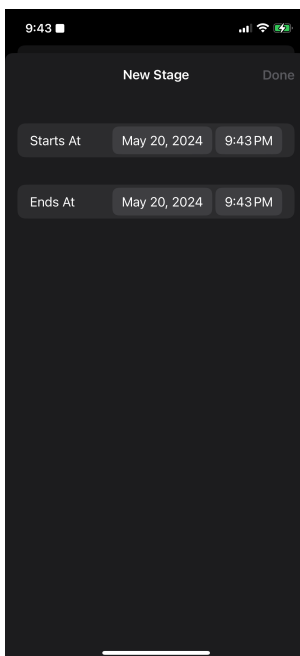
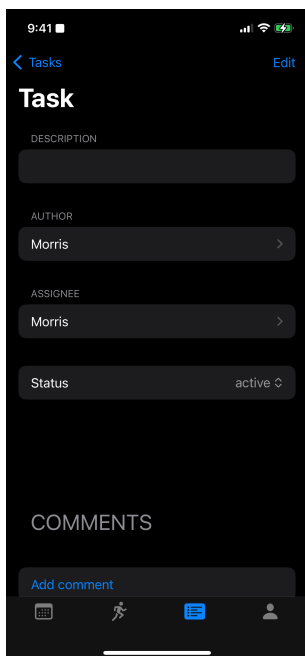
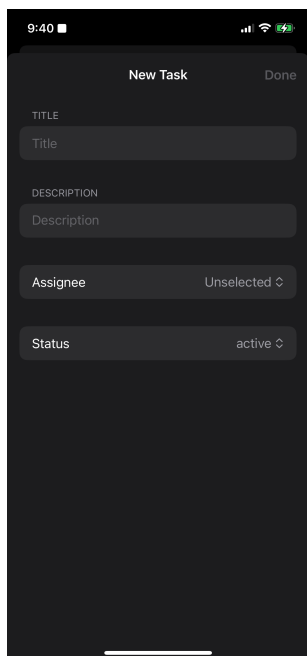
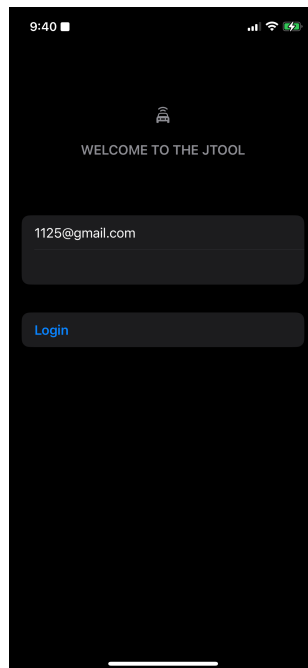
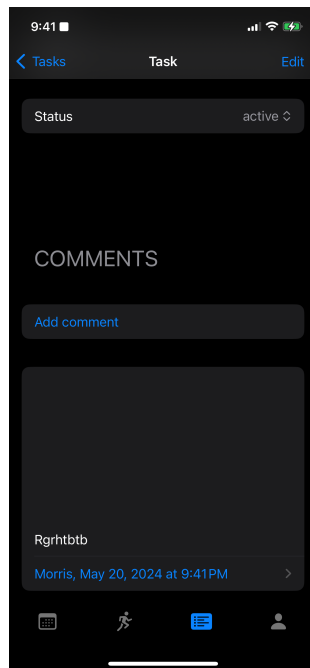
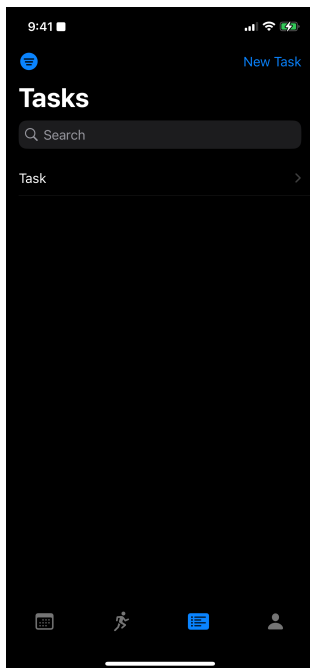
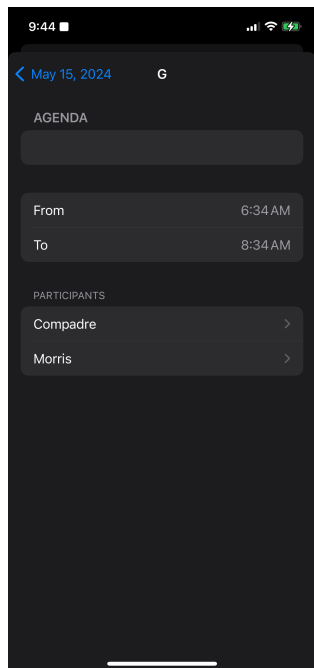
На сайте Firebase можно создать проект, куда уже можно подключать все нужные сервисы. Базу данных для удобного тестирования изначально запустим в тестовом режиме, то есть доступ к ней не будет требовать аутентификации пользователя. Далее к проекту можно подключить само IOS приложение по его идентификатору.

Первое, что необходимо настроить - это сервис аутентификации. Для тестирования я просто добавил нескольких пользователей с заранее заданными логинами и паролями. Теперь сервис аутентификации можно сразу использовать из приложения через FirebaseAPI для IOS.

Далее всех зарегистрированных пользователей нужно добавить в базу данных. Помимо этого нужно добавить в базу данных технические коллекции для статических данных приложения, такая как все доступные должности.

Сервис Хранилища - самый просто для настройки, достаточно просто запустить его в тестовом режиме (без отдельной аутентификации) теперь он доступен из приложения через FirebaseAPI.

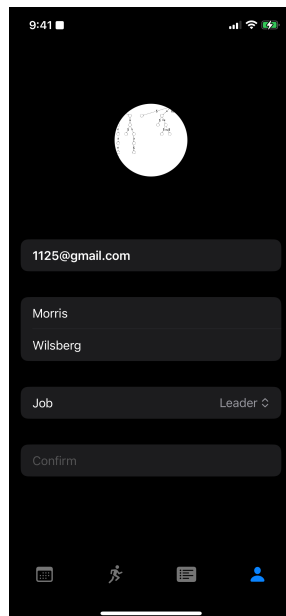
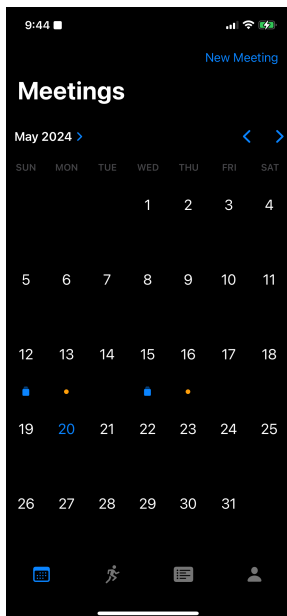




Не реализованная функциональность

Почти вся функциональность была реализована. Не было реализовано режима “интерактивных заметок”, когда во время встречи предполагалось, что лидер будет заполнять резюме встречи и отмечать, кто что сделал. Я посчитал, что эта функциональность не нужна для базового приложения, так как прогресс в задачах и так можно посмотреть через статус задачи, а ведение заметок во время

короткой встречи, может наоборот удлинить ее и мешать сосредоточиться на смысле. При желании, эту функциональность будет легко строить в существующее приложение.



Вывод

В итоге я реализовал приложение под IOS для помощи в разработке. Оно поддерживает список этапов, в каждом из которых находятся задачи. У каждой задачи свой автор, название, исполнитель, описание и статус. Пользователь может отфильтровать задачи и искать их. Также приложение поддерживает календарь встреч, с возможностью создавать и просматривать встречи. В качестве облака используется Firebase. Из-за serverless архитектуры мне не нужно поддерживать отдельный сервер для коммуникации, приложение общается напрямую с сервисами Firebase (аутентификация, база данных и хранилище). Как говорилось ранее, если потребуется выполнять какие-то регулярные задачи для поддержания базы данных, можно воспользоваться еще одним функционалом Firebase - функциями как сервис. Но для данной реализации, они не нужны. Есть много вариантов улучшения этого приложения, например добавление системы прав, чтобы каждая функциональность была доступна только

пользователю, обладающим определенными правами, эта доработка позволит в будущем использовать приложение для управления, не одной, а сразу несколькими командами.