**Требования**

1. Приложение должно работать на Android.
2. На главном экране приложения в шапке должны отображаться название текущего курса, кнопка для перехода к уведомлениям с индикатором непросмотренных уведомлений, кнопка для перехода в меню приложения.
3. На главном экране приложения должна отображаться таблица «Расписание», содержащая следующие данные: день недели и дата, расположенных в 1 строке по вертикали, время, ФИО педагога, информация о мероприятии (ссылки на сторонние ресурсы и другая информация при необходимости), расположенных в 1 столбце по горизонтали.
4. Расписание мероприятий в приложении должно отображать только текущую неделю: с понедельника по воскресенье.
5. На главной странице должны быть отображены ФИО и контакты менеджера и педагогов.
6. На главном экране приложения должна отображаться таблица «Успеваемость», содержащая следующие данные: номер раздела, название раздела, индикатор, отображающий завершённость раздела по данному курсу.
7. Таблица «Успеваемость» должна отображать все разделы текущего курса.
8. В приложении должно быть меню - каталог с кнопками (в указанном порядке): личный кабинет, мои курсы, мое портфолио, услуги, настройки приложения.
9. В приложении должна быть возможность просмотра и скачивания сертификатов за пройденные курсы.
10. В приложении должен отображаться список курсов обучающегося, разделенный на текущие и прошедшие.
11. Должна быть возможность оформлять покупки курсов приложения.
12. Регистрация и создание аккаунта в приложении должны осуществляться по почте и паролю пользователя.
13. Доступ к данным аккаунта в приложении должен быть только у самих пользователей.
14. Индикатор уведомлений должен быть яркого цвета.
15. В приложении к расписанию мероприятий должен быть доступ у всех пользователей.

**Функциональные:**

1. Сверху на главном экране приложения должны быть отображены название текущего курса, кнопка для перехода к уведомлениям, цвет которой показывает наличие или отсутствие непросмотренных уведомлений, и кнопка для перехода в меню приложения, отображенная в виде иконки.
2. На главном экране приложения должна отображаться таблица «Расписание», содержащая следующие данные (в указанном порядке):

а) день недели и дата, где дата в формате DD.MM, а день недели – аббревиатура из 2-х букв (например, «Пн, 28.02»).

б) время в формате hh.mm - hh.mm

в) ФИО педагога в формате Фамилия И.О.

г) информация о мероприятии (ссылки на сторонние ресурсы и другая информация при необходимости) в формате «текст».

1. На главной странице должны быть отображены ФИО менеджера и педагогов и кнопки перехода в личные чаты с ними.
2. На главном экране приложения должна отображаться таблица «Успеваемость» из 3-х столбцов, содержащая следующие данные:

а) номер раздела в формате «целое число»,

б) кнопка для перехода в раздел с материалами курса,

в) иконка в формате PNG, отображающий завершённость раздела по данному курсу.

1. В приложении должно быть меню - каталог с кнопками (в указанном порядке): личный кабинет, мои курсы, мое портфолио, услуги, настройки приложения.
2. В разделе меню «Услуги» отображаются действующие курсы приложения и информация о них.
3. Информация о неприобретенном курсе содержит:

а) название курса в формате строки,

б) цену за курс в формате «число с плавающей точкой, округленное до двух знаков после запятой»,

в) даты проведения в формате DD.MM.YYYY – DD.MM.YYYY,

г) кнопку «Купить».

1. В разделе «Мое портфолио» отображаются сертификаты в формате файла с расширением PNG и кнопки «Скачать».
2. В разделе «Личный Кабинет» содержится информация:

а) ФИО в формате Фамилия Имя Отчество,

б) номер телефона в формате строки,

в) почта в формате строки,

г) файл «Пользовательское соглашение» с расширением DOCX,

д) файл «Политика конфиденциальности» с расширением DOCX

1. В настройках приложения должна быть возможность регулировать звук уведомлений и других звуков приложения.
2. Информация о приобретенном курсе в разделе «Мои курсы» содержит:

а) название курса в формате строки,

б) даты проведения в формате DD.MM.YYYY – DD.MM.YYYY,

в) кнопка «Перейти»

1. Регистрация в приложении должны осуществляться по логину в формате «текст» и паролю пользователя. Менеджер и педагоги в качестве логина вводят свои ФИО в формате Фамилия И.О.

**Нефункциональные:**

1. Приложение должно работать на платформе Android 7.0 и новее.
2. Система должна иметь возможность увеличивать или уменьшать масштаб по мере необходимости.
3. Система должна быть доступна в любое время.
4. Название текущего курса на главной странице должно быть отображено жирным шрифтом.
5. Иконка кнопки «Уведомления» должна быть в виде точки яркого цвета. Если все уведомления просмотрены, иконка желтого цвета, если есть непросмотренные уведомления, то иконка зеленого цвета.
6. Иконка кнопки перехода в меню приложения должна быть в виде 4-x коротких параллельных прямых полосок, расположенных в столбик.
7. Название таблицы «Расписание» должно располагаться по центру сразу перед таблицей.
8. Расписание мероприятий в приложении должно отображать только текущую неделю: с понедельника по воскресенье, т.е. 7 дней. Строки «время», «педагог» должны быть обязательно заполнены при добавлении мероприятия в расписание. Если строка пустая - отображать строку. В таблице отображаются все мероприятия, которые стоят в расписании по текущему курсу.
9. В приложении к расписанию мероприятий курса должен быть доступ у всех пользователей данного курса.
10. На главной странице ФИО педагогов и менеджера должны быть выравнены по левому краю и отображены жирным шрифтом.
11. Кнопки перехода в чаты должны быть отображены в виде иконки письма.
12. Название таблицы «Уведомления» должно располагаться сразу перед таблицей.
13. Иконка в таблице «Успеваемость» должен отображаться в виде галочки. Если раздел не завершён, то иконка не отображается, иначе – отображается.
14. Таблица «Успеваемость» должна отображать все разделы текущего курса.
15. Кнопки в таблице «Успеваемость» должны содержать название раздела.
16. В разделе меню «Услуги» отображаются в виде списка. Отображаются только те курсы, которые пользователь не приобрел.
17. В период обучения на одном курсе нельзя параллельно обучаться на другом. Пока текущий курс не пройден, новый начать нельзя.
18. Информация о курсе отображается в виде небольшой ячейки, внутри которой содержатся информация о курсе и кнопка «Купить».
19. В разделе «Мое портфолио» файлы отображаются в виде списка, отсортированных по дате получения.
20. Если у пользователя отсутствует отчество, то в разделе «Личный кабинет» отображается только ФИ в формате Фамилия Имя.
21. В настройках приложения «регуляторы» звуков должны быть отображены в виде «ползунков».
22. Кнопка «Скачать» выровнена по левому краю и отображена в виде иконки скачивания.
23. В разделе «Мои курсы» курсы отображаются в виде списка. Текущий курс должен быть первый в списке, прошедшие курсы отсортированы по дате их завершения.
24. Доступ к данным аккаунта в приложении должен быть только у самих пользователей.
25. Если при регистрации введена почта в качестве логина, то в приложении пользователь будет идентифицироваться как «Ученик».
26. Система должна загружать приложение менее чем за 5 секунд даже при одновременном использовании 1000 пользователями.
27. Система должна выдавать не более 5 ошибочных запросов.
28. Система не должна противоречить закону о персональных данных.

**User story**

**Ученик**

Я, как ученик, хочу видеть свое расписание учебных мероприятий по текущему курсу.

Я, как ученик, хотел бы в приложении иметь возможность купить курс.

Я, как ученик, хотел бы иметь возможность связываться со своими преподавателями через приложение.

Я, как ученик, хотел бы следить за своей успеваемостью.

Я, как ученик, хотел бы иметь доступ к материалам всех моих курсов: текущих и пройденных.

Я, как ученик, хотел бы иметь возможность просматривать и скачивать сертификаты по своим завершенным курсам.

**Педагог**

Я, как педагог, хотел бы иметь возможность связываться с учениками своего курса.

Я, как педагог, хотел бы смотреть общую статистику успеваемости учеников курса.

Я, как педагог, хотел бы иметь возможность оценивать практические задания учеников курса.

Я, как педагог, хотел бы иметь доступ к просмотру успеваемости конкретных учеников курса.

Я, как педагог, хотел бы вносить изменения в расписание учеников своего курса.

Я, как педагог, хотел бы иметь доступ к просмотру заданий, выполненных моими учениками в рамках приложения.

**Менеджер**

Я, как менеджер, хотел бы иметь доступ к контактам всех пользователей приложения.

Я, как менеджер, хотел бы иметь возможность использовать фильтр при поиске определенного круга лиц среди пользователей приложения.

Я, как менеджер, хотел бы иметь доступ к редактированию информации и других данных, размещенной в приложении.

Я, как менеджер, хотел бы иметь возможность управлять деятельностью сотрудников в рамках приложения.

Я, как менеджер, хотел бы видеть статистику количества обучающихся, начинающих курс, и количества, закончивших этот курс в результате.

**Use Case**

1. Авторизация
2. Покупка курса
3. Просмотр материалов курса

Use Case:

|  |  |
| --- | --- |
| Название | Авторизация |
| Описание | Пользователь может авторизироваться в приложении, используя систему авторизации. |
| Акторы | Пользователь |
| Предусловия | Пользователь открыл приложение. |
| Основной поток событий | 1. Система запрашивает данные пользователя: логин, пароль  2. Пользователь вводит необходимую информацию  3. Нажать кнопку «Отправить» |
| Альтернативный поток событий | 1. На втором пункте введен логин, несуществующий в базе приложения  2. Система запрашивает заполнить необходимые поля: ФИО, «повторить пароль»  3. Пользователь вводит необходимую информацию  4. Нажать кнопку «Отправить» |
| Постусловия | Пользователь успешно прошёл авторизацию, и ему отображается его аккаунт. |
| Расширенные атрибуты | 1. Если клиент вводит неверные данные, система сообщает об ошибке и предлагает исправить их. |
| Диаграмма Use Case |  |
| Рекомендации по реализации | 1. Реализовать удобный и интуитивно понятный интерфейс для ввода данных учеником.  2. Отобразить в аккаунте всю информацию, введенную при регистрации, кроме пароля.  3. Предоставить пользователю удобный ввод данных для регистрации (например, в виде окошка с полями для ввода данных). |

Use Case:

|  |  |
| --- | --- |
| Название | Приобретение курса. |
| Описание | Ученик может выбрать интересующий курс и приобрести его, используя платежную систему. |
| Акторы | Ученик |
| Предусловия | Ученик открыл приложение. |
| Основной поток событий | 1. Ученик переходит в раздел «Услуги».  2. Система отображает список доступных к приобретению курсов. Система отображает название выбранного курса, сроки обучения и стоимость.  3. Ученик выбирает интересующий его курс и нажимает кнопку «Купить».  4. Система запрашивает у ученика данные для оплаты услуги.  5. Ученик вводит реквизиты банковской карты.  6. Ученик подтверждает оплату.  7. Система оповещает об успешно пройденной оплате. |
| Альтернативный поток событий | 1. На шаге 5 система оповещает ученика о некорректном вводе номера карты.  2. Ученик вводит новые данные карты.  3. Ученик подтверждает оплату.  4. Система оповещает об успешно пройденной оплате. |
| Постусловия | Ученик получил доступ к выбранному курсу. |
| Расширенные атрибуты | 1. Если у ученика недостаточно средств на карте, система предлагает пополнить баланс или использовать другую карту.  2. Ученик может выбрать доступный к приобретению курс.  3. Система принимает оплату за курс. |
| Диаграмма Use Case |  |
| Рекомендации по реализации | 1. Реализовать удобный и интуитивно понятный интерфейс для ввода данных учеником.  2. Отобразить доступные для покупки курсы в удобном для ученика виде (например, сортировать по цене или направлению). |

Use Case:

|  |  |
| --- | --- |
| Название | Просмотр материалов курса. |
| Описание | Пользователь может просматривать материалы курса и выполнять задания. |
| Акторы | Пользователь. |
| Предусловия | Пользователь открыл приложение. |
| Основной поток событий | 1. Пользователь перешел на главный экран приложения.  2. Система отображает расписание учебных мероприятий с приложенными ссылками на онлайн-занятия и успеваемость, предоставляет доступ к чатам с педагогами.  3. Пользователь открывает раздел курса в таблице «Успеваемость». |
| Альтернативный поток событий | 1. На шаге 3 пользователь открывает в меню раздел «Мои курсы». 2. Система отображает список текущих и пройденных курсов. 3. Пользователь выбирает интересующий курс и нажимает на кнопку «Перейти». |
| Постусловия | Система отобразила все материалы выбранного курса и предоставила доступ к выполнению заданий. |
| Расширенные атрибуты | 1. Пользователь может выполнить интересующее его задание. 2. Если у пользователя возникнут вопросы в процессе обучения, он может связаться с педагогом. |
| Диаграмма Use Case | IMG_256 |
| Рекомендации по реализации | 1. Предоставить пользователю возможность отслеживать, какие задания выполнены, путем отображения завершённости раздела по выбранному курсу в виде иконки (галочки). 2. Предоставление пользователю возможности подключиться к онлайн-занятию, перейдя на сторонний ресурс по ссылке, содержащейся в таблице «Расписание». 3. Предоставить пользователю возможность открывать материалы курса через раздел «Мои курсы» и через таблицу «Успеваемость». |

**Команда**

1. Бизнес-аналитик (1 человек)

-исследование актуальных программ обучения;

-мониторинг показателей эффективности;

-предлагает улучшения для оптимизации академии;

-анализ потребностей целевой аудитории;

-определяет требования к проекту;

-оптимизация маркетинга и продаж курсов;

-предложения по автоматизации процессов и внедрение новых технологий;

-работа с партнерами и инвесторами.

1. Проект-менеджер (1 человек)

-координация работ команды;

-управление разработкой и реализация проектов;

-планирование;

-контроль качества продукта, сроков, бюджета, различных рисков;

-обеспечение эффективной коммуникации между участниками проекта.

1. Системный аналитик (1 человек)

-оптимизация и интеграция различных механик с образовательным процессом;

-налаживание работы между программами и сервисами;

-разработка решений для оптимизации и автоматизации;

1. Backend-разработчик (3 человека: Middle \* 2 & Junior)

-разработка функционала для обработки данных (Middle & Junior);

-создание и поддержка сервера академии (Middle);

-интеграция внешних сервисов и баз данных (Middle & Junior);

-обеспечение безопасности системы (Junior);

-обеспечение стабильности системы (Middle & Junior \* 2);

1. Frontend (3 человека: Middle & Junior \* 2)

-реализация пользовательского интерфейса;

-адаптация под разные устройства;

1. Тестировщик (1 человек)

-фикс багов системы;

-фикс безопасности системы;

-фикс совместимости системы с разными устройствами;

1. UI/UX дизайнер (1 человек)

-очевидно разработка и отрисовка пользовательского интерфейса;

-а именно сделает нам интуитивно понятную и удобную систему для аудитории и визуально красивые дизайны решений;

1. Маркетолог (1 человек)

-создание контента для привлечения людей;

-продвижение курсов и образовательных программ академии;

-анализ целевой аудитории;

-разработка маркетинговых стратегий;

1. Менеджер (1 человек)

-управление курсами;

-планирование;

-решение административных вопросов;

-организация и координация образовательных процессов;

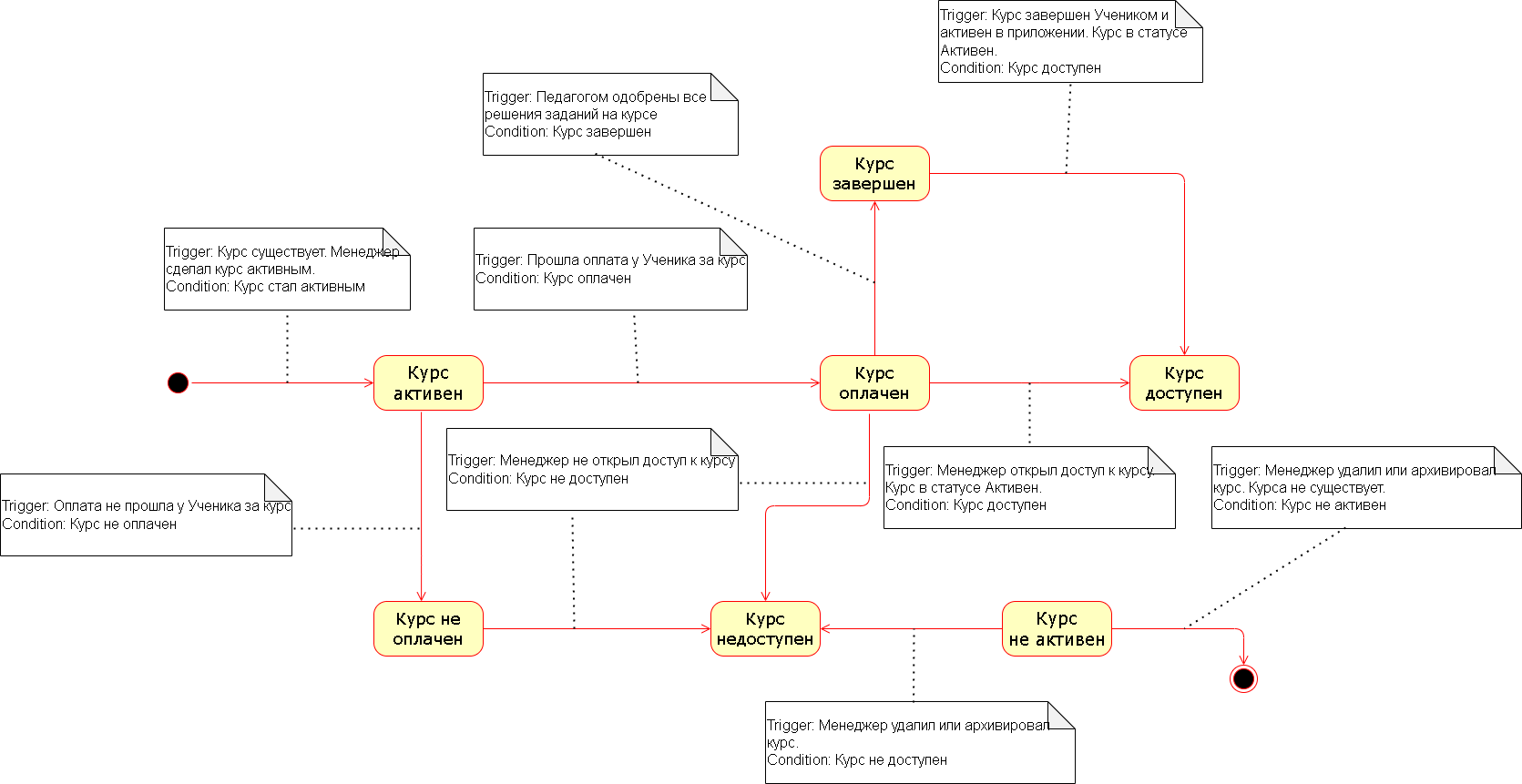
-взаимодействие с сотрудниками и всеми пользователями приложения;

В нашем проекте не понадобятся контент-менеджер, DevOps-инженеры, архитектор, продакт.

Текстовым содержимым продукта будет заниматься маркетолог. В задачи системного аналитика, разработчиков так же входят задачи DevOps-инженера, а именно настройка и автоматизация инфраструктуры приложения. Архитектура системы не предполагает каких-либо особенных сложностей, поэтому этой задачей будет заниматься системный аналитик и другие специалисты нашей команды. Так же и задачи продукт-менеджера будут делегированы между маркетологом, системным и бизнес- аналитиками.

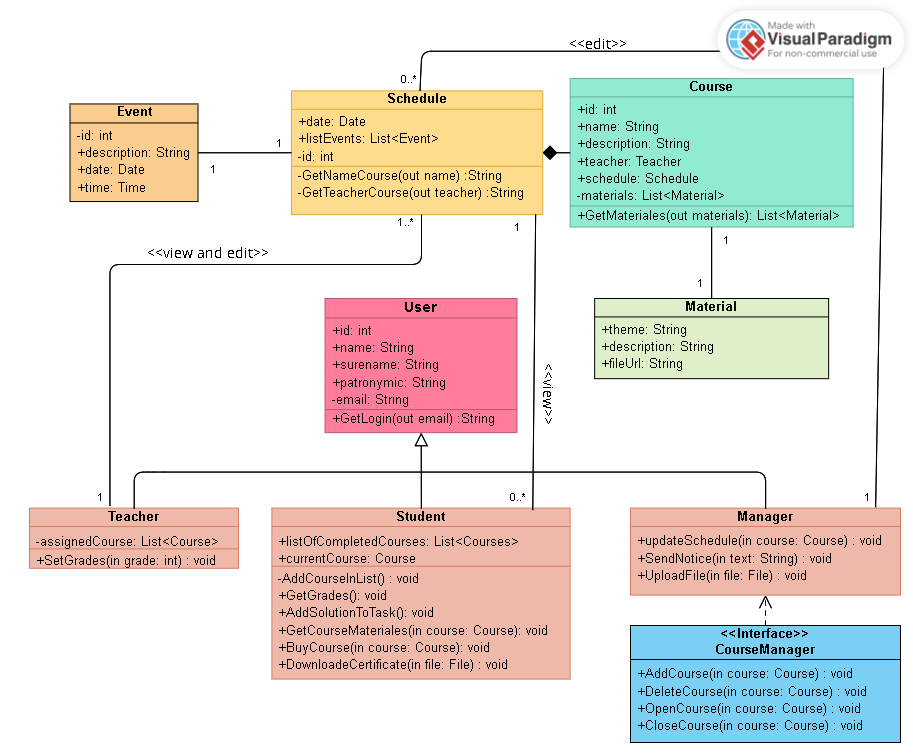
Система приложения не является чем-то новым или сложным в настоящее время. Все техники и механики подобных задач, которые предполагаются в приложении IT-академии, специалистам известны. По этой причине наш проект не нуждается в данных специалистах.

**Диаграмма состояний**



**Диаграмма последовательности**

**Диаграмма классов**



**Реализация**

Реализация проекта составит (73 дня) 584 часа или 8 спринтов.

Стоимость реализации проекта: 2123766 руб.

4) Описание структуры GitHub

В проекте на гит хабе есть 4 ветки: main, laboratory\_work, coursework и application. Суммарно сделано 20 коммитов. На ветке laboratory\_work лежат фотографии диаграммы классов, диаграммы последовательностей, диаграмма состояний, эксель-таблица со сметой и общей оценкой проекта, mind map, документы с описанием проекта, с требованиями и другие. В ветке coursework находится файл с курсовой работой. В ветке application должны располагаться проектные файлы для реализации приложения и apk. Также там располагается папка с дизайном приложения. Предполагается, наличие отдельных веток для разных страниц приложения, на которых будут лежать кодовые файлы и другие необходимые файлы.

