

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РТФ
Школа бакалавриата

ОТЧЕТ

По проекту
«Разработка системы электронного документооборота и администрирования
расписания занятий»
по дисциплине «Проектный практикум»

Заказчик: Насыров И. А.

Куратор: Насыров И. А.

Генеральный директор компании ООО «ЛАМАРТ»

Студенты команды Школы 1:

Хабибулин Тимур Раилевич РИ-320913

Амелин Александр Эдуардович РИ-320913

Аристов Михаил Николаевич РИ-320913

Еремина Ксения Сергеевна РИ-320913

Локтионов Николай Вячеславович РИ-320931

Екатеринбург, 2025

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Требования заказчика к продукту	4
2 Анализ	5
2.1 Определение проблемы.....	5
2.2 Анализ аналогов	6
3 Архитектура программного продукта	7
4 Ведение разработки и полученный результат	9
4.1 Итог работы за семестр и выводы	9
4.2 Результаты работы каждого участника	10
ЗАКЛЮЧЕНИЕ	12
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	13

ВВЕДЕНИЕ

Целью данного этапа разработки является реализация функциональности выгрузки квитанций из внешнего сервиса и предоставление возможности их просмотра и скачивания в мобильном приложении проекта «Открытые школы». Задача включает интеграцию с внешним API, загрузку PDF-файлов в базу данных и реализацию интерфейсов для операторов и конечных пользователей (родителей).

Актуальность разработки обусловлена необходимостью цифровизации документооборота и повышения удобства взаимодействия родителей с образовательной платформой. Автоматизация процесса получения квитанций позволяет значительно упростить получение информации о платежах и делает систему более прозрачной и доступной.

Результаты разработки применимы в рамках основного проекта «Открытые школы» и будут использоваться как составная часть мобильного и веб-приложений платформы.

По завершении текущей контрольной точки планируется:

- завершение архитектуры взаимодействия backend с внешним сервисом и клиентскими приложениями;
- реализация выгрузки квитанций в базу данных по команде оператора;
- обеспечение доступа родителей к квитанциям через мобильное приложение (просмотр и скачивание).

1 Требования заказчика к продукту

Требуемый результат: Система электронного документооборота в web-приложении и мобильном приложении родителя.

Итоговый продукт: Функциональность Электронный документооборот Android-приложение для родителя, которое позволяет получать, просматривать и скачивать квитанции за обучение детей в кружках дополнительного образования.

Более подробные требования к мобильному приложению родителя:

1. получение и сохранение PDF-квитанций из внешнего сервиса;
2. отображение информации о квитанциях в мобильном приложении;
3. возможность оператору вручную инициировать загрузку квитанций;
4. сохранение PDF-файлов в базу данных.

На основании требований был сформирован следующий план (backlog):

1. Настройка взаимодействия с внешним API;
2. Сохранение PDF-файлов в базу данных;
3. Реализация REST API для получения списка и содержимого квитанций;
4. Разработка пользовательского интерфейса диспетчера (администратора) в веб-приложении;
5. Интеграция с мобильным приложением для отображения и скачивания квитанций.
6. Интеграция с программным обеспечением Парус, для более удобного использования сервиса образовательными организациями.

2 Анализ

Целевая аудитория платформы «Открытые школы»: сотрудники организаций, оказывающих услуги дополнительного образования как для взрослых, так и для детей, учителя, предоставляющие свои образовательные услуги, ученики, которые хотят посещать кружки или секции и их родители, желающие записать на занятия своего ребенка.

Целевая аудитория мобильного приложения родителя: родители, которые хотят записать своего ребенка на кружки дополнительного образования.

2.1 Определение проблемы

Основные проблемы, выявленные в ходе проблемного интервью с представителем образовательной организации:

- Высокие временные затраты диспетчером (администратором) на сбор подписей и обработку бумажных данных;
- Высокая вероятность ошибки, связанная с возможными потерями и порчей документов родителями (или учителями)
- Неэффективное взаимодействие с преподавателем в связи с большим объемом бумажных данных и требованием его очного присутствия.
- В большинстве муниципальных организаций налажена работа с ПО Парус, которая объединяет всех участников образовательного процесса, однако эта работа с программным обеспечением налажена не самым удобным образом.

2.2 Анализ аналогов

Среди конкурентов онлайн-платформы «Открытые школы» достойных аналогов не много, есть такие сервисы как запись на программу дополнительного образования на «Портале Госуслуг» https://www.gosuslugi.ru/600316/1/form?_id=1733844974613.

Также среди аналогов продукта можно выделить CRM-системы, предназначенные для компаний, предоставляющих услуги в сфере дополнительного образования:

1. Учи.ру (включает онлайн-курсы)

<https://uchi.ru/>

2. Мой Класс

https://moyklass.com/crm-dlja-shkoly?utm_source=crmindex&utm_medium=catalog&utm_campaign=education ;

Функциональность интеграции с внешними сервисами для получения документов (например, в системах «Госуслуги», «Электронный дневник» и пр.) используется широко, однако большинство подобных решений не предоставляет интеграцию с существующими и принятыми в образовательных организациях системами автоматизации документооборота. А именно ПО Парус вряд ли интегрирован в какой-то из наших аналогов (мы не смогли найти такие аналоги).

3 Архитектура программного продукта

Обзор архитектуры программного продукта – мобильного приложения родителя:

Архитектура мобильного приложения базируется на принципах Clean Architecture и паттерне MVVM.

MVVM (Model-View-ViewModel) – шаблон проектирования архитектуры приложения. Позволяет отделить логику приложения от визуальной части (представления).

У MVVM есть целый ряд преимуществ по сравнению с другими архитектурными паттернами разработки программного обеспечения. Основные из них:

- Привязка данных: MVVM использует двунаправленную привязку, которая автоматически синхронизирует компоненты между собой. Это устраняет необходимость в явной манипуляции данными и делает код более простым и поддерживаемым.
- Разделение обязанностей: паттерн четко разделяет все три компонента. Это делает код модульным, а потому тестировать его становится легче.
- ау

Логика МП:

- Авторизация и регистрация. Регистрация происходит с отправкой смс-кода для подтверждения номера телефона. Для сохранения данных пользователя используется Room.

- Экран профиля. Включает в себя аватарку, ФИО, кнопки: “Семья”, “Приглашения в кружки”, “Расписание”, “Квитанции”, “Сообщить о проблеме”.

- Экран квитанции. Включает минималистичное отображение последних полученных квитанций (не оплаченные) по кружкам, на которые записан

ребенок в семье родителя, сумму начислений по всем квитанциям, а также кнопку скачивания всех квитанций в одном pdf файле.

- Функциональность «Уведомление», срабатывающая при загрузке квитанции в веб-приложение администратора.

Обзор архитектуры программного модуля – Django-приложения:

- Сервисный слой для обращения к внешнему API и обработки PDF-файлов;
- View-функции (контроллеры) для обработки запросов от веб и мобильных клиентов;
- Модель хранения квитанций, включающая бинарное поле для хранения PDF;
- Интерфейс диспетчера (администратора) для массовой и индивидуальной выгрузки квитанций;
- Мобильный API, предоставляющий список квитанций и возможность их скачивания.

В качестве архитектурного подхода использована модель с разделением ответственности на контроллеры (views), сервисы и модели, что упрощает масштабирование и тестирование компонентов.

4 Ведение разработки и полученный результат

Проект велся по гибкому итеративному подходу – по Agile. Внутри команды регулярно проводились синхронизации и распределение задач по приоритетам. Каждый компонент проходил ручное тестирование на этапе завершения реализации. Основные ошибки были связаны с кодировкой PDF-файлов и обработкой исключений при работе с внешним API, что было устранено путём дополнительной валидации и логирования.

А со стороны мобильного приложения, происходила длительное погружение в архитектуру проекта (был выбран и осваивался паттерн MVVM), а также длительное ожидание пуша бэкенда на стенд, из-за чего тестирование сетевых интерфейсов было отложено на более поздний срок. Написание всего нового кода сопровождалось распределением на слои, руководствуясь Clean Architecture.

Касательно работы дизайнера, нареканий и негативного опыта не было получено.

4.1 Итог работы за семестр и выводы

Не успели разработать mvw мобильного приложения к моменту третьей контрольной точки, т.к не была произведена интеграция ПО Парус,

Вывод: было ошибкой поздно начинать согласовывать интеграцию ПО Паруса (а именно в третьей части семестра) из-за длительного и усложненного процесса получения лицензии программного обеспечения. Из-за большого количества времени, потраченного на уточнение и анализ всех инструментов электронного документооборота, а также затянутое погружение в архитектуру

проекта, нам не удалось достигнуть поставленной в начале семестра цели – интеграции ПО Парус.

4.2 Результаты работы каждого участника

Хабибулин Тимур – тимлид и мобильный разработчик

Как тимлид – руководил командой, ставил задачи команде, следил и проводил регулярные созвоны для команды разработчиков: “загрузка” – получение задач на неделю, “демо” – отчетность за неделю по поставленным задачам и митапы (при возникновении важных вопросов), был прямым связующим с заказчиком проекта.

Как мобильный разработчик – создал архитектуру функциональности (и проекта), был ответственным за её реализацию, являлся ментором младшего мобильного разработчика.

Амелин Александр – мобильный разработчик

Пощенку сложности реализации выдвинутых идей по решению проблем, реализовал пару экранов и работу с сетевыми запросами в мобильном приложении родителя, а также написал логику обработки уведомлений о поступлении квитанций под наставничеством старшего мобильного разработчика.

Аристов Михаил – backend-разработчик

Разработал архитектуру передачи квитанций, реализовал взаимодействие с внешним API, загрузку PDF-файлов в базу данных, а также REST-интерфейсы для взаимодействия с веб и мобильными клиентами.

Еремина Ксения – дизайнер

Разработала пользовательский интерфейс для веб-приложения диспетчера (администратора), включая макеты выгрузки квитанций и взаимодействия с системой, а также принимала участие в корректировке макетов для мобильного приложения родителя, а именно наполнения экрана “Квитанции”.

Локтионов Николай – аналитик

Совместно с тимлидом и заказчиком продумала функционал и логику продукта, собрал требования от заказчика, присутствовал на встрече (проблемном интервью) с представителем образовательной организации, как результат встречи определил этапы текущей организации документооборота и учета расписания дополнительных кружков, подготовил описание бизнес-процесса и сформировал backlog для команды разработки.

ЗАКЛЮЧЕНИЕ

Реализованная функциональность почти полностью соответствует заявленным требованиям: обеспечена интеграция с внешним сервисом, PDF-файлы успешно загружаются в базу данных по команде диспетчера (администратора), а родители имеют возможность просматривать и скачивать квитанции через мобильное приложение.

Тестирование показало корректную работу всех ключевых компонентов. Основные сложности были связаны с особенностями формата данных, однако они были своевременно устранены. Идеальное решение поставленной задачи заключается в интеграции ПО Парус, однако основной помехой в данной интеграции является усложненное взаимодействие между организациями в связи с получением лицензии и дополнительных соглашений.

В качестве направления для дальнейшего развития предлагается:

1. автоматизация процесса выгрузки квитанций по расписанию;
2. расширение формата отображения данных в мобильном приложении (например, разбивка по месяцам);
3. интеграция в процесс выгрузки квитанций программного обеспечения Парус

Результаты текущей работы позволили заложить архитектурную основу для последующих доработок и стабильной интеграции в основной проект.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. <https://gitverse.ru/blog/articles/development/116-cto-takoe-mvvm-shablon-kak-rabotaet-preimushhestva-i-primery-ispolzovaniya>
2. <https://crminindex.ru/for/education>
3. <https://www.bitrix24.ru/journal/crm-v-detskom-tsentre-i-kak-ee-vnedrit/>
4. <https://parus.com/>
5. [https://parusnpc.ru/docs/1._%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B5_%D1%85%D0%B0%D1%80%D0%B0%D0%BA%D1%82%D0%B5%D1%80%D0%B8%D1%81%D1%82%D0%B8%D0%BA%D0%B8_\(%D1%82%D0%B5%D1%85%D0%BF%D0%B0%D1%81%D0%BF%D0%BE%D1%80%D1%82\)_%D0%9F%D0%B0%D1%80%D1%83%D1%81-%D0%91%D1%8E%D0%B4%D0%B6%D0%B5%D1%82_8.pdf](https://parusnpc.ru/docs/1._%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B5_%D1%85%D0%B0%D1%80%D0%B0%D0%BA%D1%82%D0%B5%D1%80%D0%B8%D1%81%D1%82%D0%B8%D0%BA%D0%B8_(%D1%82%D0%B5%D1%85%D0%BF%D0%B0%D1%81%D0%BF%D0%BE%D1%80%D1%82)_%D0%9F%D0%B0%D1%80%D1%83%D1%81-%D0%91%D1%8E%D0%B4%D0%B6%D0%B5%D1%82_8.pdf)