**Содержание**

[Введение 3](#_Toc512288279)

[1 ОБЩАЯ ХАРАКТЕРИСТИКА ПРЕДПРИЯТИЯ 4](#_Toc512288280)

[1.1 Структура и функционирование организации 4](#_Toc512288281)

[1.2 Изучение области и специфики решаемых в организации задач 7](#_Toc512288282)

[1.3 Изучение новых программных разработок и компьютерно-информационных технологий, применяемых в организации 8](#_Toc512288283)

[2 ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ 11](#_Toc512288284)

[2.2 Обоснование актуальности выбранной темы 14](#_Toc512288285)

[2.3 Поиск, обзор, и сравнение возможных аналогов по теме дипломного проекта. 15](#_Toc512288286)

[2.4 .Разработка структуры системы с указанием параметров и выбранных технических характеристик. 17](#_Toc512288287)

[ЗАКЛЮЧЕНИЕ 18](#_Toc512288288)

# Введение

Преддипломная практика проходила с 23.03.2018 по 19.04.2018 на предприятии ИООО “ЭПАМ Системз”.

Основными целями преддипломной практики является систематизация и применение в профессиональной работе полученных знаний в области автоматизации объектов и систем.

Задачей практики является получение опыта практического программирования для решения конкретных задач.

План работы по преддипломной практике:

1) Изучение организационной и функциональной структуры предприятия;

2) Изучение области и специфики решаемых в организации задач;

3) Изучение новых программных разработок и компьютерно-информационных технологий, применяемых в организации;

4) Выполнение поставленных заданий;

5) Оформление отчета по преддипломной практике.

# 1 ОБЩАЯ ХАРАКТЕРИСТИКА ПРЕДПРИЯТИЯ

## 1.1 Структура и функционирование организации

EPAM Systems — американская ИТ-компания основанная в 1993 году, производитель заказного программного обеспечения, специалист по консалтингу, резидент Белорусского парка высоких технологий. Штаб-квартира компании расположена в Ньютауне (США, штат Пенсильвания).

С 1993 года EPAM помогает мировым лидерам придумывать, проектировать, разрабатывать и внедрять программное обеспечение, которое изменяет мир. Сегодня компания уже больше, чем просто разработчики. Это эксперты, которые помогают вывести бизнес на новый уровень, используя самые современные методы программирования.

EPAM Systems неизменно остается признанным лидером в таких областях, как:

− разработка, тестирование, сопровождение и поддержка заказного программного обеспечения и бизнес-приложений;

− интеграция приложений на базе продуктов SAP, Oracle, IBM, Microsoft;

− создание выделенных центров разработки (центров компетенции),центров тестирования и контроля качества программного обеспечения;

− ИТ-консалтинг с учетом отраслевой специфики бизнеса.

Сегодня EPAM является ведущим глобальным поставщиком услуг по разработке цифровых платформ и программного обеспечения. Используя свой многолетний технологический опыт и компетенции в сфере консалтинга, дизайна и инновационных стратегий, EPAM тесно сотрудничает со своими клиентами для создания передовых решений, которые превращают сложные бизнес-задачи в реальные бизнес-возможности. Команды профессионалов EPAM работают с заказчиками в более чем 25 странах в Северной Америке, Европе, Азии и Австралии. EPAM занимает 12 место в списке Forbes «25 cамых быстрорастущих публичных технологических компаний», 1 место среди поставщиков ИТ-услуг в списке Fortune «100 самых быстрорастущих компаний». Результаты работы в 2017 году (выручка 1,45 млрд. долларов) подтверждают репутацию EPAM как ведущего мирового игрока. Способность масштабировать соответствующие мировым стандартам возможности по разработке сложных технологических решений и сочетать их с услугами отраслевого и цифрового консалтинга обуславливает преимущество предложения компании для клиентов. Руководство компании рассчитывает, что выбранный ими подход к созданию команд из дополняющих друг друга специалистов в области стратегии, дизайна, программирования и автоматизации будет способствовать дальнейшему росту и поможет расширить возможности для сотрудников EPAM по всему миру.

У EPAM есть опыт разработки для Microsoft, IBM и других известных технологических лидеров и производителей программных продуктов. Фактически компания начинала как Developer for Developers, но затем стала помогать клиентам из IT-индустрии не только разрабатывать их продукты, но и внедрять их у конечных заказчиков и кастомизировать их под потребности бизнеса. Тем самым компания постепенно приобрела бизнес-опыт в различных отраслях. Первоначально это были индустрия путешествий и развлечений, ритейл и медиа, а затем — финансовый сектор и страхование. В 2016 году больше 25% бизнеса компании, оборот которого превышает миллиард долларов, пришелся на заказчиков из сферы финансов, среди которых Barclays, UBS, Citibank, Группа ВТБ, Сбербанк, Альфа-Банк и другие

. EPAM, развиваясь как эволюционно, так и путем слияний и поглощений, приобрел консультантов, дизайнеров, визионеров, которые способны определить ценность идеи на уровне концепта. При этом компания смотрит на внедрение инноваций как на процесс, которым можно и нужно управлять, с определенной методологией и набором различных инструментов. Они помогают определить лучшие идеи и довести их до создания «Most Valuable Product» и запуска в эксплуатацию. Главное — сделать это быстро, чтобы заказчик смог предложить своим клиентам что-то, что отвечает их постоянно меняющимся ожиданиями.

EPAM показывает ежегодный стабильный рост прибили в 20-25%. Компания демонстрирует, что качество и разнообразие предоставляемых решений и сервисов позволяет компании продолжать значительный рост. В условиях очень высокой конкуренции на мировом рынке технологических услуг EPAM удерживает лидирующие позиции в качестве ключевого партнера для крупнейших мировых компаний из списка Global 2000, помогая им успешно реализовывать проекты по цифровой трансформации бизнеса

Административная структура компании EPAM Systems представлена на рис.1.1.

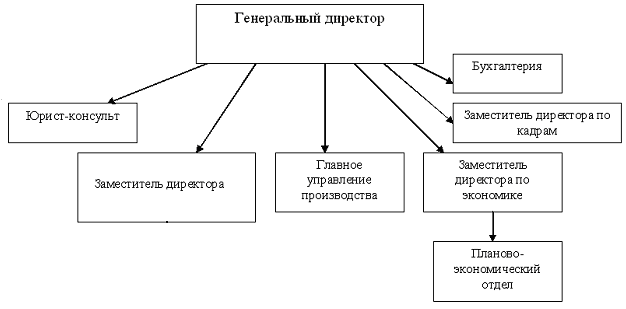


Рисунок 1.1 – Административная структура компании EPAM Systems

Во главе предприятия находится директор. Он организует всю работу предприятия и несет полную ответственность за результат производственно-хозяйственной деятельности EPAM Systems. Директор представляет предприятие во всех учреждениях и организациях, заключает договора, издает приказы по предприятию, открывает в банках счета предприятия и выполняет целый ряд других функций.

В непосредственном подчинении директора предприятия находятся три заместителя: по маркетингу, по экономике и по кадрам, а также главный бухгалтер и юрист-консульт.

Заместитель директора по кадрам возглавляет работу по обеспечению предприятия кадрами рабочих, служащих и инженерно-технических работников требуемых профессий в соответствии с уровнем и профилем полученной подготовки, заботится о создании положительного мнения общественности о предприятии. Он также подготавливает документы для начисления пенсий, осуществляет контроль за правильностью оформления документов по начислению зарплаты.

Заместитель директора по маркетингу и перспективному планированию является первым заместителем директора предприятия, в обязанности которого входит целый ряд наиболее важных и ответственных функций. Первый заместитель директора предприятия организует и руководит деятельностью службы, занятой проведением комплекса маркетинговых мероприятий, направленных на повышение конкурентного уровня качества выпускаемой продукции; рекламирование продукции и услуг предприятия. Он участвует в принятии решений по претензиям потребителей продукции и услуг; руководит мероприятиями, направленными на сбыт выпускаемой продукции.

Заместитель директора по экономике руководит и координирует работу планово-экономического отдела. Планово-экономический отдел подготавливает необходимые материалы для рассмотрения проектов планов, итогов деятельности предприятия, организует работу по нормированию и ценообразованию на предприятии, проводит экономический анализ производственно-хозяйственной деятельности.

Юрист-консульт подчиняется непосредственно директору предприятия и осуществляет проверку соответствия действующему законодательству приказов, распоряжений, соглашений и других актов правового характера, участие в заключении коллективных договоров и т.д.

Общая характеристика предприятия EPAM Systems показала, что на современном этапе развития данная организация является одним из лидеров по разработкам программного обеспечения. Показатели результативной деятельности предприятия являются удовлетворительными и свидетельствуют о большем потенциале организации.

## 1.2 Изучение области и специфики решаемых в организации задач

Средства информационных технологий являются неотъемлемой и наиболее существенной их составляющей. Они призваны выполнять ту же роль, что и средства производства в процессе трудовой деятельности.

Средства информационных технологий - это совокупность средств деятельности человека, которые создаются и используются с целью осуществления процессов производства и удовлетворения непроизводственных потребностей общества. К средствам информационных технологий относят разнообразную компьютерную технику, которая облегчает и повышает уровень эффективности трудовых усилий человека, расширяет его возможности в процессе трудовой деятельности и освобождает человека частично или полностью от работы.

В состав средств информационных технологий входят приспособления и инструменты, машины, механизмы, автоматические устройства. Также, их можно классифицировать на средства организационной, коммуникационной и вычислительной техники. В состав организационной техники входят средства, которые облегчают и обеспечивают офисную и инженрно-техническую работу, копировальное и проектное оборудование (компьютеры, принтеры, проекторы, локальная и глобальная сети).

К коммуникационной техники входят телефоны, радиосвязь, факс и другие предметы, с помощью которых осуществляется передача информации. В состав компьютерной техники входят автоматизированные средства обработки данных и информации.

Средства информационных технологий позволяют сделать процесс обработки информации и данных наиболее легче, удобнее и быстрее.

## 1.3 Изучение новых программных разработок и компьютерно-информационных технологий, применяемых в организации

На предприятии я изучал новую технологию uwp. Научился работать с данными в формате Jason. Познакомился трехуровневой архитектурой веб-приложения.

UWP (Universal Windows Platform) представляет собой унифицированную платформу для создания и запуска приложений в Windows 10 и Windows 10 Mobile.

UWP стала результатом фолюции более ранних технологий. Так, с выходом Windows 8 была внедрена новая архитектурная платформа для приложений - Windows Runtime (WinRT), которая позволяла запускать приложения в так называемом режиме Modern (Metro) на десктопах, планшетах. Затем с выходом Windows 8.1 и Windows Phone 8.1 эта технология получила развитие - появились "универсальные приложения", которые можно было запускать сразу Windows 8.1 и WP8.1. И в июле 2015 года официально вышла новая ОС Windows 10. Она использует платформу UWP, которая представляет собой развитие Windows Runtime.

Как подсказывает название платформы, она является универсальной - универсальной для всех устройств экосистемы Windows 10. А это обычные дестопы, планшеты, мобильные устройства, устройства IoT (интернет вещей), Xbox, устройства Surface Hub. И приложение UWP может одинаково работать на всех этих платформах, если на них установлена Windows 10.

**Трёхуровневая архитектура** – архитектура клиент-серверных приложений, которая разбивает приложения на 3 уровня(как показано на рисунке 3.1):

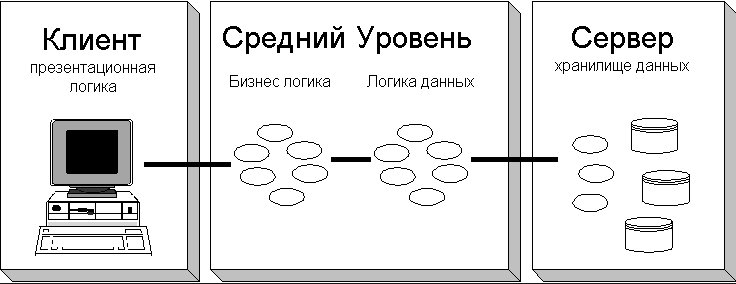


Рисунок 2.4 – Трехуровневая архитектура

1. Представление(клиент)

2. Логика

3. Данные(БД)

Каждый уровень приложения должен быть независимым и закрытым, то есть не должен показывать зависимости, которые относятся к реализации данного уровня. Все слои должны быть соединены между собой, значит несоединённые слои не должны иметь возможности взаимодейстовать.

Уровень представления предоставляет пользовательский интерфейс, управляет взаимодействием с пользователем и не должен содержать никакой бизнес-логики или доступа к данным.

Уровень логики содержит в себе набор правил для обработки информации, может работать с несколькими пользователями и не должен содержать в себе уровень доступа и данных.

Уровень данных содержит в себе физическое хранилище данных, обрабатывает запросы к базе данных и не должен содержать в себе к уровню представления и логики.

# 2 ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

Тема дипломного проекта – разработка клиентского приложения MoneySpliter для перераспределения финансовых средств между пользователями.

**2.1 Изучение литературы и стандартов по теме дипломного проектирования. Определение технологий реализации.**

Для разработки приложения я выбрал язык С# и технологию uwp, которая описана в 1.3 пункте отчета. Программирование под UWP несет ряд преимуществ:

1)Широта распространения. На текущий момент (апрель 2017) Windows 10 установлена уже более чем на 400 миллионах устройств. На десктопах Windows 10 уже опередила Windows 8/8.1.

2)Поддержка широкого круга устройств. Десктопы, планшеты, смартфоны, большие планшеты Surface Hub, различные IoT-устройства, в перспективе устройства виртуальной реальности HoloLens - круг устрйоств, на которых может работать Windows 10 действительно широк.

3)Поддержка разных языков и технологий программирования. UWP-приложения можно создавать с помощью таких языков, как Visual C++, C#, Visual Basic, JavaScript. В качестве технологии для создания графического интерфейса Visual C++, C# и Visual Basic используют XAML, JavaScript применяет HTML. Кроме того, С++ может вместо XAML использовать DirectX. То есть достаточно распространенные и и знакомые многим технологии.

4)Магазин приложений и удобство распространения. Windows Store представляет собой прекрасное место для распространения UWP-приложений, как платных, так и бесплатных. Сами возможности платформы и магазина Windows Store позволяют использовать разные способы монетизации. Например, можно интегрировать в приложения блоки для показа рекламы через различные SDK. Можно распространять за определенную плату, причем оплату можно гибко настраивать. При необходимости можно встроить предоставление ознакомительной версии, после использования которой пользователь может решить, покупать приложение или нет. И также можно монетизировать по модели freemium, при которой приложение условно бесплатное, а отдельные услуги внутри приложения предоставляются за определенную плату. Причем все эти возможности монетизации обесечиваются встроенными инструментами SDK.

5)Богатые возможности платформы. UWP многое наследует от Windows Runtime из Windows 8.1 и в то же время предоставляет много новых функциональности, как, более богатые возможности по интеграции с облаком, использование Cortana, системы уведомлений в Win10 и многое другое.

Перед как разрабатывать приложение необходимо изучить литературу и техническую документацию. Я познакомился с архитектурой MVVM, научился работать с данными в формате Jason. Изучил принципы работы фреймворка Caliburn и познакомился с разметкой XAML.

**Архитектура MVVM:**

Шаблон Model-View-ViewModel (MVVM) — применяется при проектировании архитектуры приложения. Первоначально был представлен сообществу Джоном Госсманом (John Gossman) в 2005 году как модификация шаблона Presentation Model. MVVM ориентирован на современные платформы разработки, такие как Windows Presentation Foundation, Silverlight от компании Microsoft[1], ZK framework.

MVVM используется для разделения модели и её представления, что необходимо для изменения их отдельно друг от друга. Например, разработчик задает логику работы с данными, а дизайнер соответственно работает с пользовательским интерфейсом.

Шаблон MVVM делится на три части:

1) Модель (англ. Model), так же, как в классической MVC, Модель представляет собой логику работы с данными и описание фундаментальных данных, необходимых для работы приложения.

2) Представление (англ. View) — это графический интерфейс, то есть окно, кнопки и т. п. Представление является подписчиком на событие изменения значений свойств или команд, предоставляемых Моделью Представления. В случае, если в Модели Представления изменилось какое-либо свойство, то она оповещает всех подписчиков об этом, и Представление, в свою очередь, запрашивает обновленное значение свойства из Модели Представления. В случае, если пользователь воздействует на какой-либо элемент интерфейса, Представление вызывает соответствующую команду, предоставленную Моделью Представления.

3) Модель Представления (англ. ViewModel) является, с одной стороны, абстракцией Представления, а с другой, предоставляет обёртку данных из Модели, которые подлежат связыванию. То есть, она содержит Модель, которая преобразована к Представлению, а также содержит в себе команды, которыми может пользоваться Представление, чтобы влиять на Модель.

**Формат данных Jason:**

JSON (JavaScript Object Notation) - простой формат обмена данными, удобный для чтения и написания как человеком, так и компьютером. Он основан на подмножестве языка программирования JavaScript, определенного в стандарте ECMA-262 3rd Edition - December 1999. JSON - текстовый формат, полностью независимый от языка реализации, но он использует соглашения, знакомые программистам C-подобных языков, таких как C, C++, C#, Java, JavaScript, Perl, Python и многих других. Эти свойства делают JSON идеальным языком обмена данными.

JSON основан на двух структурах данных:

Коллекция пар ключ/значение. В разных языках, эта концепция реализована как объект, запись, структура, словарь, хэш, именованный список или ассоциативный массив.

Упорядоченный список значений. В большинстве языков это реализовано как массив, вектор, список или последовательность.

Это универсальные структуры данных. Почти все современные языки программирования поддерживают их в какой-либо форме. Логично предположить, что формат данных, независимый от языка программирования, должен быть основан на этих структурах.

В нотации JSON это выглядит так:

Объект - неупорядоченный набор пар ключ/значение. Объект начинается с { (открывающей фигурной скобки) и заканчивается } (закрывающей фигурной скобкой). Каждое имя сопровождается : (двоеточием), пары ключ/значение разделяются , (запятой).

**XAML:**

XAML (англ. eXtensible Application Markup Language) — расширяемый язык разметки для приложений (произносится [замл] или [зэмл]) — основанный на XML язык разметки для декларативного программирования приложений, разработанный Microsoft.

Модель приложений Vista включает объект Application. Его набор свойств, методов и событий позволяет объединить веб-документы в связанное приложение. Объект Application контролирует выполнение программы и генерирует события для пользовательского кода. Документы приложения пишутся на XAML. Впрочем, с помощью XAML описывается, прежде всего, пользовательский интерфейс. Логика приложения по-прежнему управляется процедурным кодом (С#, VB, JavaScript и т. д.). XAML может использоваться как для браузер-базированных приложений, так и для настольных приложений.

XAML включает основные четыре категории элементов: панели, элементы управления, элементы, связанные с документом и графические фигуры. Заявлено 7 классов панелей, которые задают принципы отображения вложенных в них элементов. Для задания положения элементов относительно границ родительской панели используются атрибуты на манер свойств в объектно-ориентированных языках. Подобный синтаксис не совсем соответствует рекомендациям CSS, но является привычным для программистов настольных приложений.

Приложения, объявленные в XAML, могут включать множество страниц. Элемент управления PageViewer позволяет разбивать содержание на страницы и обеспечивает навигацию по ним. Элемент ContextMenu помогает в создании навигационных меню приложения. Код процедурного языка может быть размещён непосредственно в файле XAML или же назначен при сборке проекта.

**Caliburn:**

Caliburn.Micro — это фреймворк, сильно упрощающий описание View и ViewModel. По сути, он сам создаёт байндинги на основании соглашений об именах, тем самым избавляя разработчика от написания их вручную и делая код меньше и чище.

## 2.2 Обоснование актуальности выбранной темы

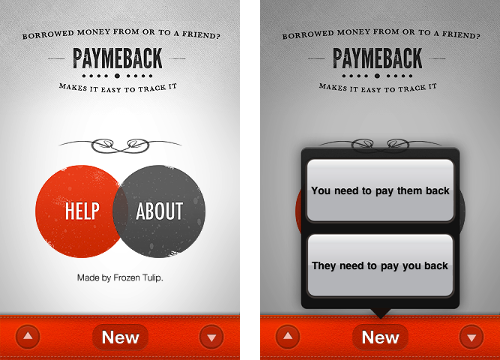
Денежный долг - это достаточно интересная сфера отношений. Мы сами даем в долг, у нас просят дать в долг. Все, кто хоть раз участвовал в этом финансовом упражнении, могут отметить одно правило – взять в долг гораздо легче, чем отдать его.

Практически все из нас рано или поздно сталкиваются с необходимостью взять деньги в долг или одолжить их кому-либо. Ситуаций может быть много, это и друг забывший дома кошелек, и ситуация, когда один человек покупает подарок, а остальные позже с ним рассчитываются. Или мероприятие, где друзья платят друг за друга. Но что делать если иногда небольшие суммы забываются? Конечно, можно вести записи на бумажке, но а можно использовать данное приложение.

## 2.3 Поиск, обзор, и сравнение возможных аналогов по теме дипломного проекта.

В интернете имеется следующие аналоги:

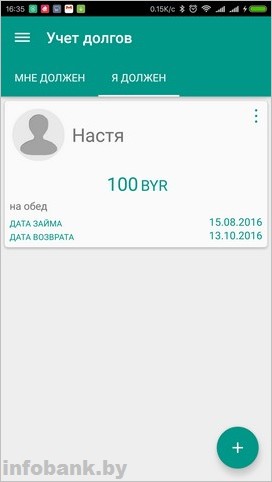
**PayMeBack** — очень простое приложение, несущее всего лишь одну функцию. Как вы уже могли догадаться, мы будем пользоваться им для того, чтобы записать, когда вы берете в долг или одалживаете кому-либо.



**Учет долгов.**

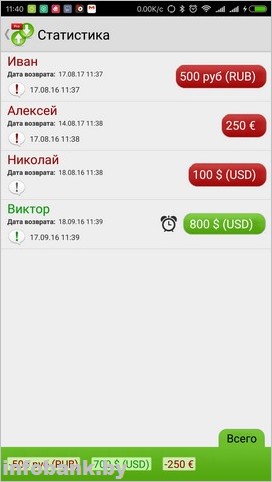
Это приложение очень простое, есть всего два раздела: Мне должен и Я должен.

Программа также считает баланс долгов – сколько вы должны и сколько вам должны по каждой валюте



**Должники PRO**.

Есть наглядная статистика, причем долги выделены разными цветами – красный это я должен, а зеленый – мне должны.



К видим приложения в основном мобильные под android и ios. И работают по принципу записной книги. Пользователь делает какие-то записи, пометки, и он в любое время может посмотреть долги. Мое же приложение работает на устройствах с операционной системой Windows 10. Это может быть планшет, пк, ноут-бук, мольное устройство. Но главное отличие не в этом. Мое приложение работает с реальными пользователями, а не просто с записями. Каждый пользователь регистрируется и заключает сделки с другими пользователями .

## 2.4 .Разработка структуры системы с указанием параметров и выбранных технических характеристик.

Тема дипломного проекта – разработка клиентского приложения MoneySpliter для перераспределения финансовых средств между пользователями.

Для начало пользователь должен зарегистрироваться в приложении нажав на кнопку регистрация. Или войти если он уже зарегистрирован. На главной странице находится информация пользователя, а также информация о текущих транзакциях. В меню есть поиск, где пользователь может искать других людей и добавлять в друзья. А друзей можно добавлять в группы и вместе скидываться на какой-нибудь праздник. Ты пишешь сумму и эта сумма делится на всех. И каждому приходит запрос согласен ли он на эту сделку. Пользователь должен дать свое согласие. Тот кто создавал сделку, у него появляются записи, кто ему должен. И у должников появляется запись что они должны. Когда должник закрывает сделку, отдает деньги, пользователю приходит запрос действительно ли это так, если да то сделка закрывается. Можно будет выставлять дату, на которую даются деньги.

Можно будет расширить функционал и долг давать предметы книжку. Без приложение вы можете забыть и нет у вас книги.

Если нет интернета вы не можете заключать сделки и закрывать, а только просматривать их.

# ЗАКЛЮЧЕНИЕ

Был проведён анализ организационной и функциональной структуры предприятия, структуры и характеристик информационных потоков, аппаратного и программного обеспечения средств автоматизированной обработки информации.

По итогам преддипломной практики я окончательно определилась с тематикой дипломного проекта, которая будет наиболее актуальна на месте работы. Кроме того, полученные знания и навыки будут использованы при написании дипломного проекта.

Преддипломная практика повысила практические навыки в области работы над большими проектами,командной разработке, углубила знания в области объектно-ориентированного программирования и проектирования. Немалую роль играет практика работы в команде, общения с иностранными коллегами. Также была получена практика разработки по различным методологиям разработки(Scrum, Canban).

Условия практики соответствовали требованиям. Опыт, полученный на преддипломной практике, будет использован в моей дальнейшей профессиональной деятельности.