

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”

Факультет компьютерных наук
Кафедра технологий обработки и защиты информации

Приложение «Мой ФКН»
Курсовой проект
по дисциплине
Технологии программирования

09.03.21 Информационные системы и технологии
Обработка информации и машинное обучение

6 семестр 2022/2023 учебного года

Зав. Кафедрой	_____ д. т. н., профессор А. А. Сирота
Обучающийся	_____ Н. В. Мерзляков, 3 курс, д/о
Обучающийся	_____ А. В. Сиваков, 3 курс, д/о
Обучающийся	_____ Д. И. Кулинченко, 3 курс, д/о
Обучающийся	_____ И.М. Кудинов, 3 курс, д/о
Руководитель	_____ В.С. Тарасов, ст. преподаватель ___. ___.20__.

Воронеж 2023

Содержание

Содержание	2
Введение	4
1 Используемые определения	5
2 Постановка задачи	7
3 Анализ предметной области	10
3.1 Анализ существующих решений	10
3.1.1 FlipTable	10
3.1.2 Университет в кармане.	11
3.1.3 BlackBoard	13
3.1.4 SDU Informer	15
3.1.5 Мобильное приложение «Цифровой университет МГЮА»	16
3.1.6 SUAI Pocket: Расписание ГУАП	17
3.2 Итог анализа	18
3.3 Анализ потребности	19
4 Графическое описание работы системы	20
4.1 Диаграмма IDEF0	20
4.2 Диаграмма активности	21
4.3 Диаграммы прецедентов	22
4.3.1 Диаграмма прецедентов (admin)	22
4.3.2 Диаграмма прецедентов (student)	23
4.3.3 Диаграмма прецедентов (teacher)	24
4.3.4 Диаграмма прецедентов (unauthorized user)	25
4.4 Диаграмма развёртывания	25
4.5 Диаграммы состояний	26
4.5.1 Диаграмма состояний (mobile app)	27
4.5.2 Диаграмма состояний (user)	28
4.6 Диаграммы сотрудничества	28
4.7 Диаграмма последовательности	32
4.8 Диаграмма классов	36
4.9 Диаграмма объектов	36
4.10 ER-диаграмма	38
4.11 Физическая модель базы данных	39

Список использованных источников	40
--	----

Введение

В наше время во время обучения в университете очень сложно обойтись без применения информационных технологий. ФКН стремится идти в ногу со временем и поэтому неудивительно, что всё новые и новые технологии находят свое применение в образовательном процессе. Тем не менее, за время обучения у авторов этой работы накопилось некоторое количество идей, реализация которых могла бы способствовать автоматизации и упрощению образовательного процесса как для студентов, так и для преподавателей. За основу была взята идея доработки уже имеющихся технологических решений. Основной идеей данной работы является упрощение использования уже имеющихся функций информационных систем, применяемых на факультете и добавление новых, которых не хватает.

Внедрение данных разработок поможет упростить ряд задач, неизбежно возникающих при реализации образовательных задач факультета, что в конечном счёте будет способствовать улучшению качества образовательного процесса.

Как уже отмечалось выше, за основу были взяты уже имеющиеся решения. В процессе разработки приложения одной из задач стало сохранение преимуществ этих решений и исправление недостатков. Был сделан упор на функциональность и простоту использования: приложение должно решать ряд задач, но при этом быть простым, интуитивно понятным для всех пользователей и не перегруженным лишними функциями.

Как показывают многочисленные исследования [1], одной из главных проблем российских ВУЗов является плохая коммуникация между студентами, преподавателями и деканатами. Именно поэтому факультету важно иметь мобильное приложение, чтобы решить проблему с устаревшими и неэффективными способами коммуникации.

1 Используемые определения

Таблица 1. Используемые определения

Термин	Определение термина
Аватар	Изображение, используемое в учетной записи для персонализации пользователя.
Авторизация	Предоставление определённого лицу прав на выполнение определённых действий; а также процесс проверки (подтверждения) данных прав при попытке выполнения этих действий.
Авторизованный пользователь	Пользователь прошедший процесс авторизации
Агрегатор (приложение-агрегатор)	Приложение, объединяющее в себе услуги нескольких компаний, данные из нескольких источников и/или функции нескольких приложений или сайтов.
Администратор	Человек, имеющий доступ к расширенному функционалу веб-сервиса, имеющий знания о формате приема статей.
Боковое меню (сайд меню)	Меню, которое представляет собой панель, которая находится (или открывается, при помощи, каких-либо кнопок или жестов) снизу, слева или справа от области основного контента приложения, содержащая вертикальную, независимую от основного контента приложения прокрутку, и служит основным инструментом навигации в приложении.
Веб-сервис	Идентифицируемая уникальным веб-адресом (URL-адресом) программная система со стандартизированными интерфейсами, а также HTML-документ сайта, отображаемый браузером пользователя.
Виртуальный ассистент	Программный агент, который может выполнять задачи для пользователя на основе информации, введенной пользователем, данных о его местонахождении, а также информации, полученной из различных интернет-ресурсов.
Искусственный интеллект (ИИ)	Искусственный интеллект — свойство искусственных интеллектуальных систем (в

Термин	Определение термина
	том числе информационных) выполнять творческие функции.
Неавторизованный пользователь	Пользователь, не прошедший процесс аутентификации
Ошибка (Bug)	Общий термин, используемый для обозначения непредвиденной ошибки или дефекта в аппаратном или программном обеспечении, что приводит к его неисправности.
Пользователь	Лицо, которое использует действующую систему для выполнения конкретной функции.
Профиль (в веб-приложении)	Учетная запись пользователя в веб-приложении, вход в которую осуществляется с помощью логина / номера телефона / e-mail и пароля. В учетной записи содержится информация о пользователе.
СУБД	Система управления базами данных. Комплекс программ, позволяющих создать базу данных (БД) и манипулировать данными.
Фреймворк	Программные продукты, которые упрощают создание и поддержку технически сложных или нагруженных проектов. Фреймворк, как правило, содержит только базовые программные модули.
Чат	Средство обмена сообщениями по компьютерной сети в режиме реального времени.
Android	Операционная система для мобильных устройств.
API	Программный интерфейс приложения. Описание способов, которыми одна компьютерная программа может взаимодействовать с другой программой
CSS	Формальный язык, служащий для описания оформления внешнего вида документа, созданного с использованием языка разметки (HTML, XHTML, XML).
Django	Свободный фреймворк для веб-приложений на языке Python, использующий шаблон проектирования MVC.
Django ORM	Инструмент фреймворка Django, который позволяет взаимодействовать с базами данных,

Термин	Определение термина
	используя высокоуровневые методы Python, а не SQL-запросы.
Flutter	Комплект средств разработки и фреймворк с открытым исходным кодом для создания мобильных приложений под Android и iOS.
Front-end	Пользовательский интерфейс компьютера или любого устройства.
HTML	Стандартизированный язык разметки веб-страниц во Всемирной паутине.
iOS	Мобильная операционная система для смартфонов, электронных планшетов, носимых проигрывателей, разрабатываемая и выпускаемая американской компанией Apple
JavaScript	Мультипарадигменный язык программирования, используется как встраиваемый язык для программного доступа к объектам приложений.
MVC	Схема разделения данных приложения и управляющей логики на три отдельных компонента: модель, представление и контроллер.
PostgreSQL	Свободная объектно-реляционная система управления базами данных.
Python	Высокоуровневый язык программирования общего назначения.
REST	Архитектурный стиль взаимодействия компонентов распределенного приложения в сети.
WebView	Системный компонент, которое отвечает за открытие веб-страниц в рамках другого приложения.

2 Постановка задачи

Целью данного курсового проекта является разработка самостоятельного приложения для упрощения образовательного процесса на факультете ФКН. Приложение не должно полностью дублировать функции, уже реализованные при помощи других технических решений, в частности, платформы moodle, но должно содержать функции, которых еще нет, или которыми неудобно пользоваться в текущем виде.

Для достижения данных целей приложение должно отвечать следующим требованиям:

1. Приложение должно обладать простым, интуитивно понятным и не перегруженным лишними деталями интерфейсом.
2. Приложение должно выполнять ряд основных функциональных задач:
 - Получение информации о новостях факультета. К каждой из новостей может быть привязано событие с определенным временем;
 - Возможность просмотра карты факультета по каждому из зданий по этажам;
 - Возможность просмотра календаря с событиями, каждое из событий будет иметь свой тип и соответствующий цвет;
 - Возможность общения между пользователями при помощи чатов. Способ поиска собеседника - его имя пользователя (логин);
 - Возможность общаться с ассистентом с искусственным интеллектом. Ассистент при необходимости должен предоставлять информацию о ближайших занятиях, событиях и новостях;
 - Возможность работы с Moodle и BRS из самого приложения;
 - Возможность просмотра расписания для конкретной группы студента. Преподаватель также должен иметь свое расписание, основанное на расписаниях студентов.

Для выполнения данных требований необходимо выполнить следующие задачи:

1. Проектирование веб-сервиса средствами языка Python;
2. Разработка back-end части, включающая в себя:
 - 2.1 Реализация ролей:

- Администратор;
- Студент;
- Преподаватель;
- Незарегистрированный пользователь (гость);

2.2 Реализация функциональных возможностей ролей;

2.3 Подключение внешнего модуля для хранения данных;

2.4 Разработка базы данных;

2.5 Разработка бизнес-логики приложения;

3. Разработка front-end части, включающей в себя:

3.1 Создание макета дизайна;

3.2 Реализация мобильного приложения при помощи фреймворка Flutter;

3.3 Добавление соединения с API сервера;

4. Проведение тестирования проекта.

3 Анализ предметной области

3.1 Анализ существующих решений

Анализ существующих решений будем проводить на основе данных сервисов:

Таблица 2. Примеры существующих решений

FlipTable	https://fliptable.ru/
Университет в кармане	http://moyuniver.ru/
Blackboard	https://www.blackboard.com/
SDU Informer	https://sdu2.software.informer.com/
Мобильное приложение «Цифровой университет МГЮА»	https://msal.ru/content/tsifrovoy-universitet/studentam/mobilnoe-prilozhenie/
SUAI Pocket: Расписание ГУАП	https://vk.com/suainav

3.1.1 FlipTable

Российская платформа, облегчающая создание и просмотр расписания. Может работать в браузере без установки каких-либо приложений. Обладает рядом полезных возможностей: экспорт расписания, многопользовательский доступ, импорт данных и т.д. Пользователи системы имеют возможность просматривать расписание в любом удобном для них формате: xls, *pdf, *iCal, в

онлайн-формате или мобильном приложении. По данным разработчиков, данная система успешно используется в 10 учебных заведениях России.

Преимущества:

- Использует свободно распространяемое ПО;
- Поддерживает разнообразные форматы;
- Упрощает создание расписания: система автоматически обнаруживает «накладки»;
- Предоставляет инструменты для аналитики;
- Простая в использовании.
- Недостатки:
- Платная: в бесплатной версии есть реклама и нет некоторых возможностей, например, экспорта в iCal и приложение «Расписание ВУЗов»;
- Практически невозможно найти отзывы на сторонних ресурсах. Их фактически нет после того, как приложение было удалено из Play Market. Разработчики также предоставляют только общую информацию о продукте. Следовательно, до начала использования невозможно оценить риск возможных проблем и наличие недостатков.

3.1.2 Университет в кармане.

Представляет из себя онлайн-платформу и экосистему мобильных приложений для решения любых учебных задач. Содержит свободно распространяемую информацию как для студентов, так и для преподавателей. Является не одним конкретным приложением, а целой экосистемой. Проект был спонсирован компанией Microsoft. На сайте можно задавать вопросы и искать учебные и методические материалы, что может помочь в образовательном

процессе. Сайт обладает простым и понятным интерфейсом. Однако можно отметить слишком минималистичный и устаревший дизайн.



Рис. 1 - Лента сообщений на сайте «Университет в кармане»

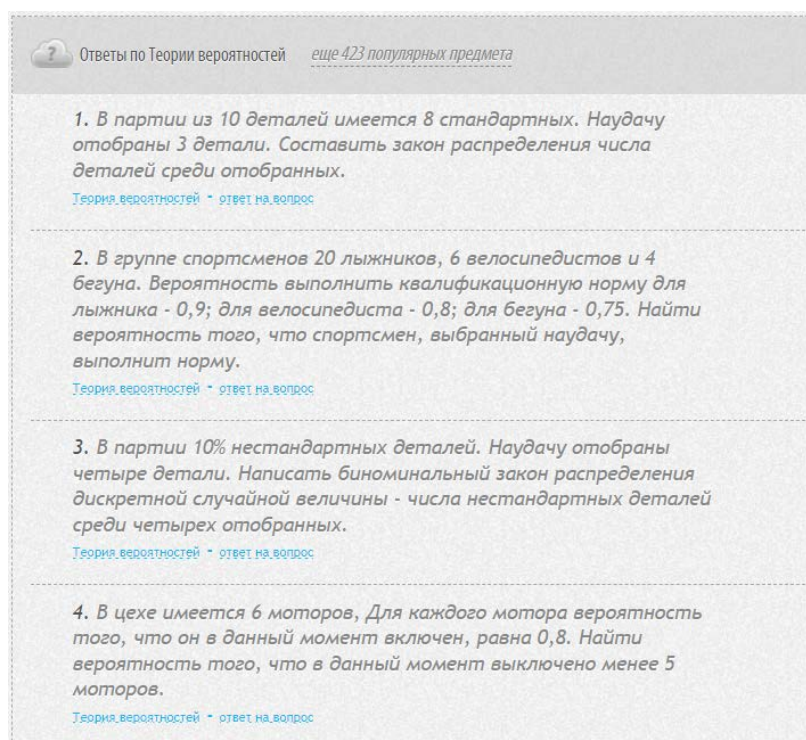


Рис. 2 - Ответы на сайте «Университет в кармане»

Преимущества:

- Может применяться как студентами, так и преподавателями;
- Наличие экосистемы из приложений, которые могут удобно взаимодействовать друг с другом;
- Простой интерфейс и наличие поиска по сайту.

Недостатки:

- Ограниченность функций: сервис не имеет никаких функций, связанных, например, с расписанием. Фактически, его цель – лишь помощь в обмене данными и освоении учебного материала, а для преподавателей – помощь в распространении методических материалов;
- Сервис направлен скорее не на упрощение процесса образования, а на обман образовательной системы. Так, например, на нём можно найти генераторы текста для сочинений и предложения о выполнении учебных работ за деньги. Таким образом, внедрение подобной системы на факультете будет несколько неуместным и даже вредным действием;
- Устаревший дизайн;
- На момент написания данной работы могут возникать проблемы при попытке воспользоваться данным сервисом без VPN.

3.1.3 BlackBoard

Зарубежный коммерчески успешный продукт, используемый во многих учебных заведениях, например, США. Представляет из себя не полноценное приложение, а своеобразный конструктор, позволяющий каждому ВУЗу собрать своё приложение. BlackBoard предоставляет ряд возможностей: удобную

загрузку новостных лент, карты университетов, поиск контактов, возможность подключения DropBox, календарь событий и многое другое.



Рис. 3 - Иллюстрация внешнего вида сайтов, созданных при помощи BlackBoard



Рис. 4 - Интерфейс приложения Florida State University, основанного на BlackBoard



Рис. 5 - Интерфейс приложения College of St. Scholastica, основанного на Blackboard.

Преимущества:

- Простота создания приложений;
- Большое количество разнообразных функций;
- Большинство необходимых функций уже реализовано – не надо ничего создавать самостоятельно.

Недостатки:

- Является конструктором, а не полноценным приложением. Следовательно, требуются дополнительные ресурсы на разработку;
- Является коммерческим и, следовательно, платным продуктом;
- Некоторые функции, например, фотогалерея, реализованы плохо;
- Не предусмотрен вывод практически никакой персонализированной информации для студентов.

3.1.4 SDU Informer.

Система, разработанная казахстанских университетом им. С. Демиреля. Обладает продвинутой лентой новостей, постоянно держит пользователя в курсе

событий. Также приложение содержит обновленную информацию об университете, факультетах, студенческих клубах, контактах, последних новостях и библиотеку с учебниками в формате e-pub, авторами которых являются преподаватели ВУЗа.

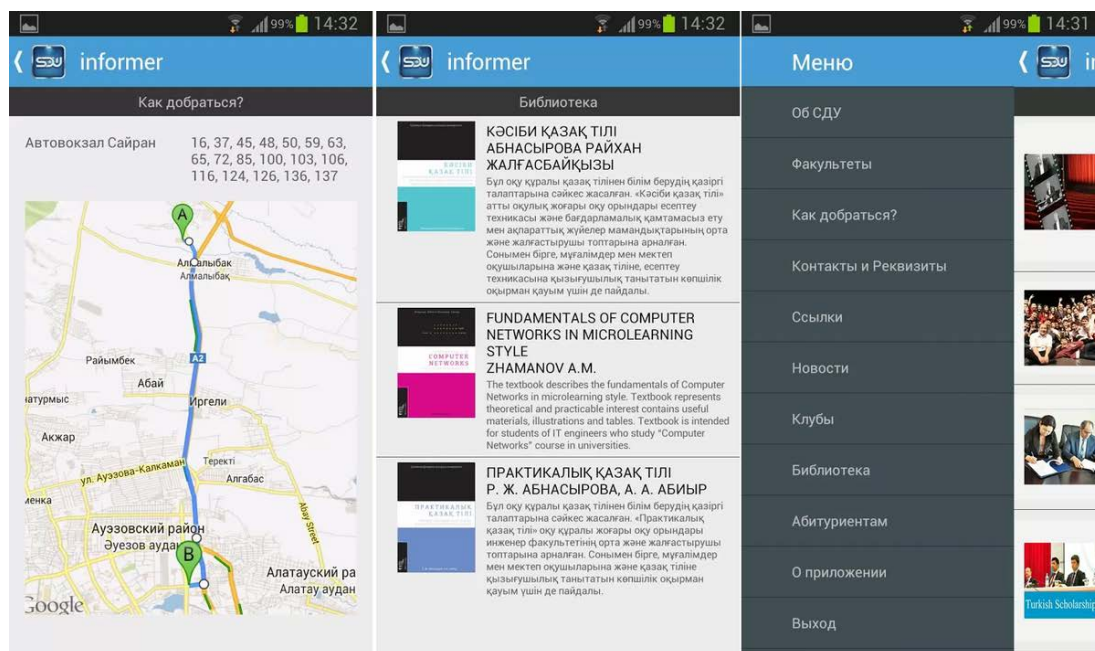


Рис. 6 - Интерфейс SDU Informer.

Преимущества:

- Удобный интерфейс
- Современный дизайн
- Функциональность

Недостатки:

- Пользователи отмечают нестабильную работу

3.1.5 Мобильное приложение «Цифровой университет МГЮА»

Приложение, используемое Московским Государственным Юридическим Университетом. В приложении публикуется личная информация по расписанию занятий, новостям и мероприятиям. Имеет простой интерфейс. Личные кабинеты студентов и преподавателей отличаются и в зависимости от роли приложение предлагает разные функции.

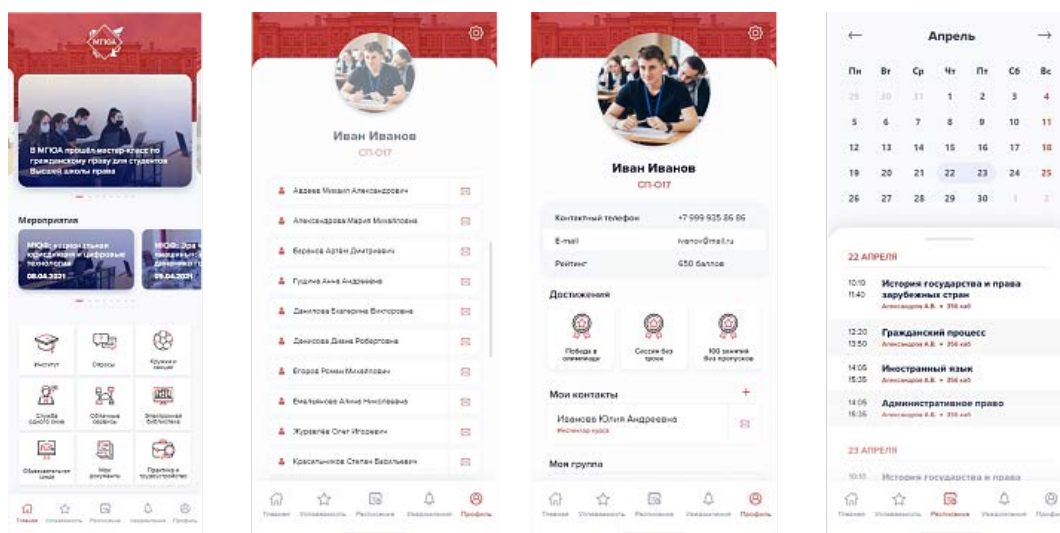


Рис. 7 - Интерфейс приложения «Цифровой университет МГЮА»

Преимущества:

— Простой, понятный, не перегруженный интерфейс;

Недостатки:

— Низкая функциональность;

— Проблемы со входом в учётную запись и нестабильная работа;

3.1.6 SUAI Rocket: Расписание ГУАП

Приложение, используемое Санкт-Петербургским государственным университетом аэрокосмического приборостроения. Имеет ряд полезных функций для студентов. Отдельно стоит отметить удобный просмотр расписания. Также есть полезная функция просмотра заданий с приближающимся сроком сдачи. Также есть возможность просмотра полезной информации об университете и преподавателей. В отличие от большинства аналогов, имеет функцию загрузки отчётов по заданиям. При этом преподаватели имеют возможность прямо в приложении дать отзыв на задание.

Преимущества:

— Современный дизайн

— Удобный интерфейс

- Множество полезных функций
- Недостатки:
- Перегруженность функциями, необходимость которых сомнительна.

3.2Итог анализа

В процессе анализа предметной области было установлено, что при разработке приложения следует придерживаться следующих аспектов:

- Чат для удобной коммуникации;
- Простой и понятный интерфейс;
- Наличие хорошей ленты новостей и календаря событий;
- Стабильность работы;
- Возможность использования как студентами, так и преподавателями;
- Удобный просмотр расписания;
- Современный дизайн;
- Самостоятельность приложения;
- Наличие карты факультета;
- Своевременное обновление информации и информирование о происходящих событиях;

Также было установлено, что необходимо избегать таких ошибок, как:

- Нестабильная работа
- Устаревший дизайн и неудобный интерфейс
- Низкая, ограниченная функциональность

- Недостаток возможностей для коммуникации
- Создание «конструктора» вместо полноценного приложения

3.3 Анализ потребности

Можно заключить, что факультет нуждается в простом, но при этом функциональном и стабильном мобильном приложении. Имеющиеся на рынке решения либо недоступны для приобретения факультетом, либо обладают определёнными недостатками. Существуют альтернативы разрабатываемого приложения, например, такие как уже используемая платформа moodle. Однако, их недостатки негативно сказываются на удобстве образовательного процесса. Поэтому приложение, разрабатываемое в ходе данного проекта, является лучшим решением, чем имеющиеся на данный момент сервисы.

4 **Графическое описание работы системы**

4.1 Диаграмма IDEF0

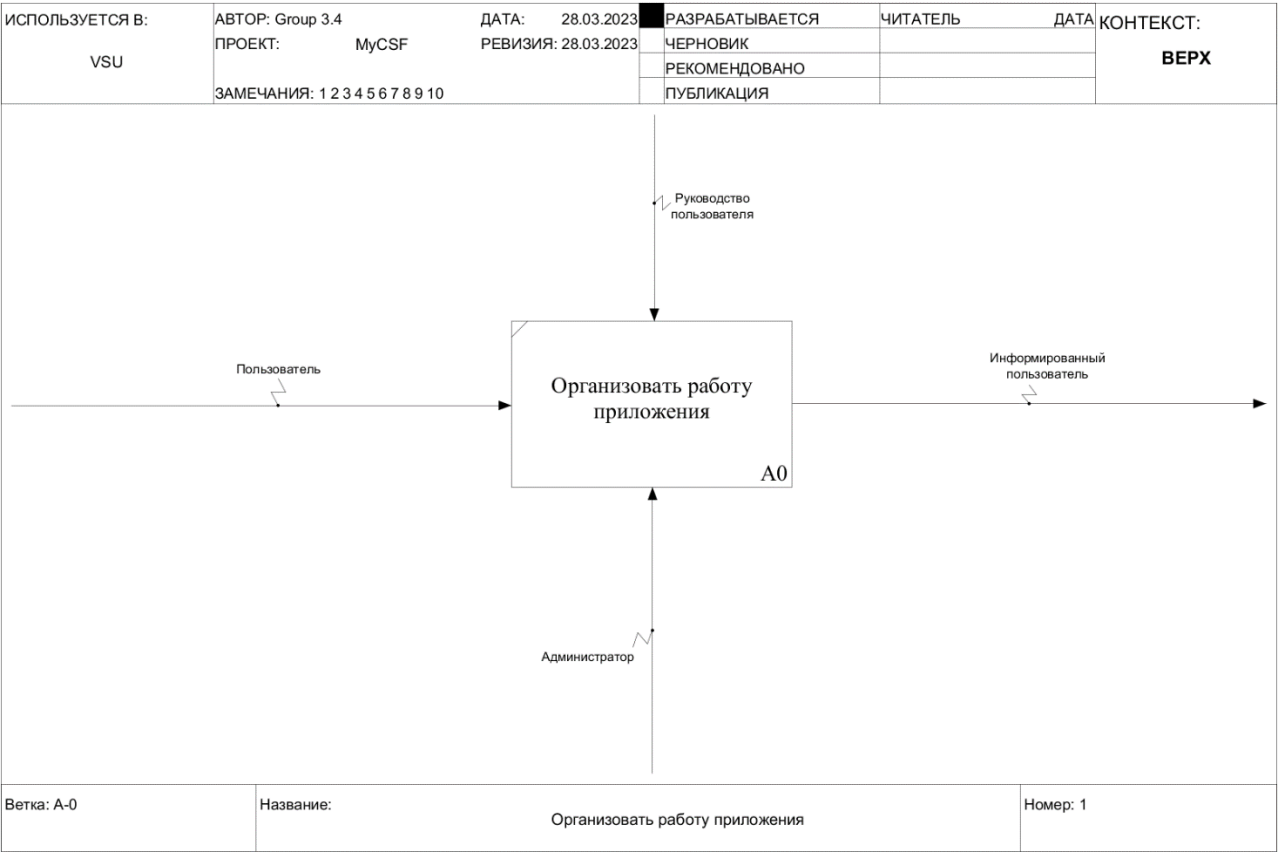


Рис. 8 - диаграмма IDEF0

На рисунке 8 представлена диаграмма IDEF0. Данная диаграмма используется для создания функциональной модели, отображающей структуру и функции системы, а также потоки информации и материальных объектов, связывающих эти функции.

4.2 Диаграмма активности

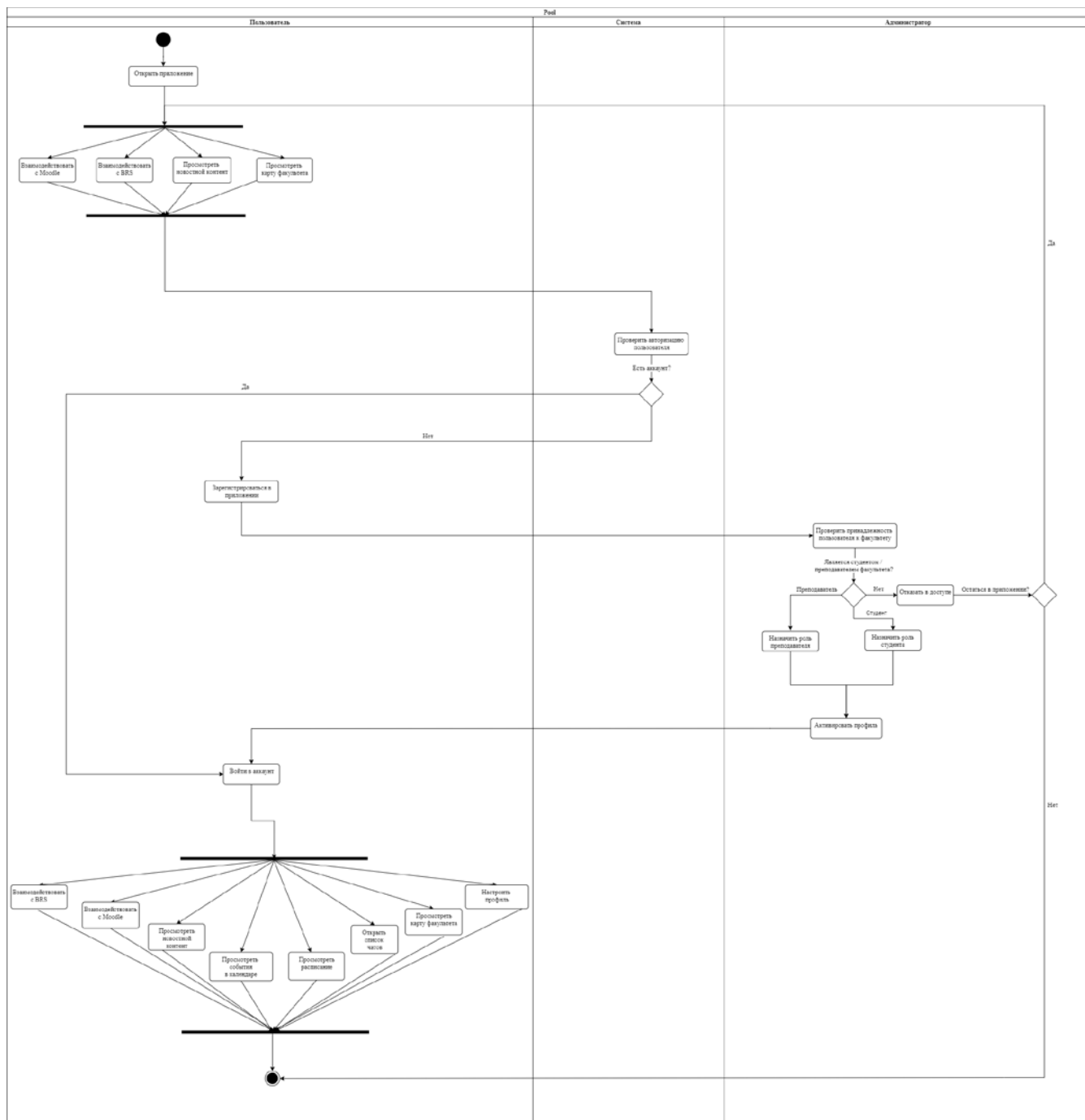


Рис. 9 - Диаграмма активностей

На рисунке 9 приведена диаграмма активностей, используемая для моделирования происходящих процессов и выполняемых последовательностей действий.

4.3 Диаграммы прецедентов

Далее будут приведены диаграммы Use-Case (диаграммы прецедентов) для разных пользователей. Эти диаграммы демонстрируют различные сценарии, возникающие при использовании приложения.

4.3.1 Диаграмма прецедентов (admin)

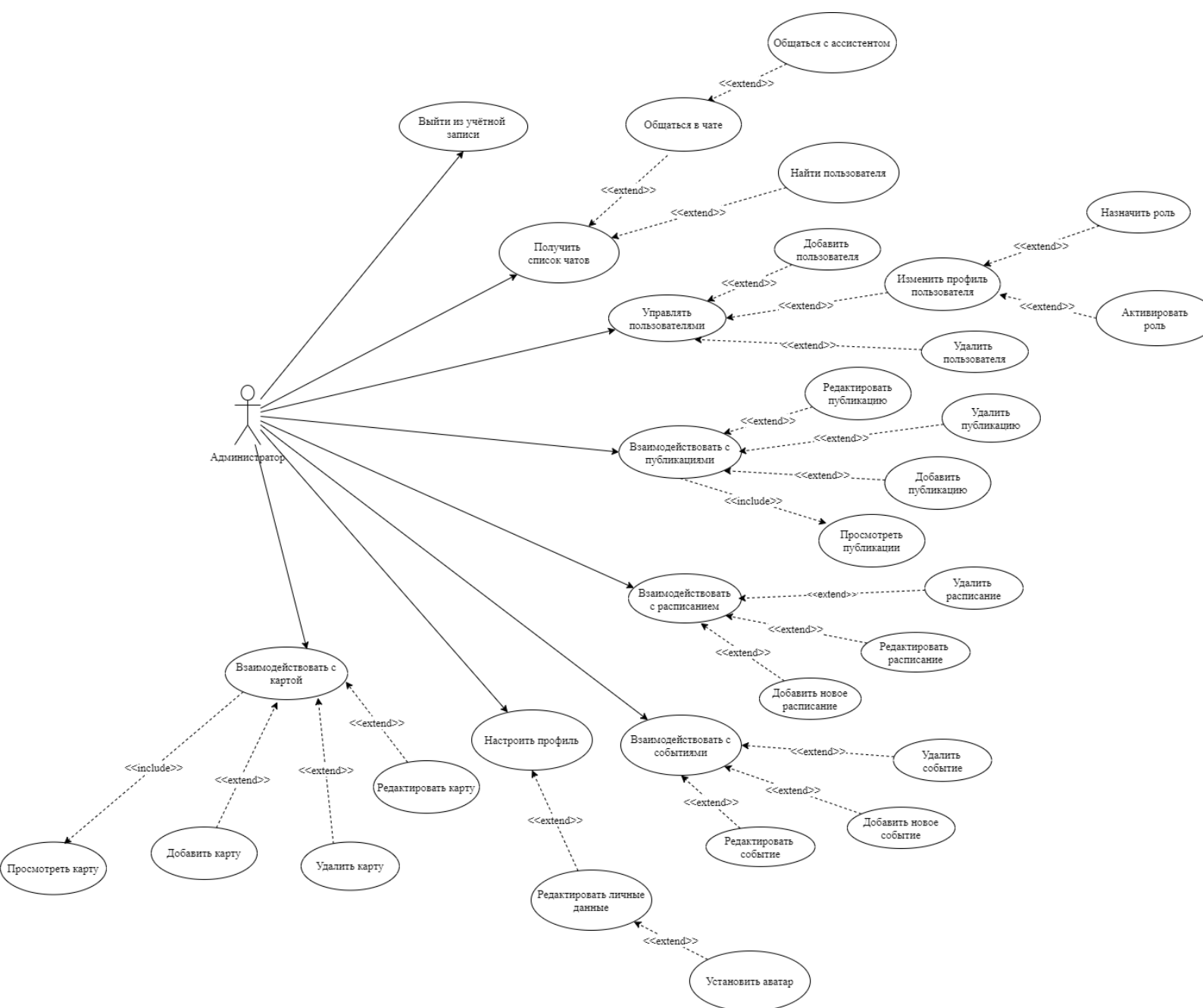


Рис. 10 - Диаграмма прецедентов (admin)

4.3.2 Диаграмма прецедентов (student)



Рис. 11 - Диаграмма прецедентов (student)

4.3.3 Диаграмма прецедентов (teacher)

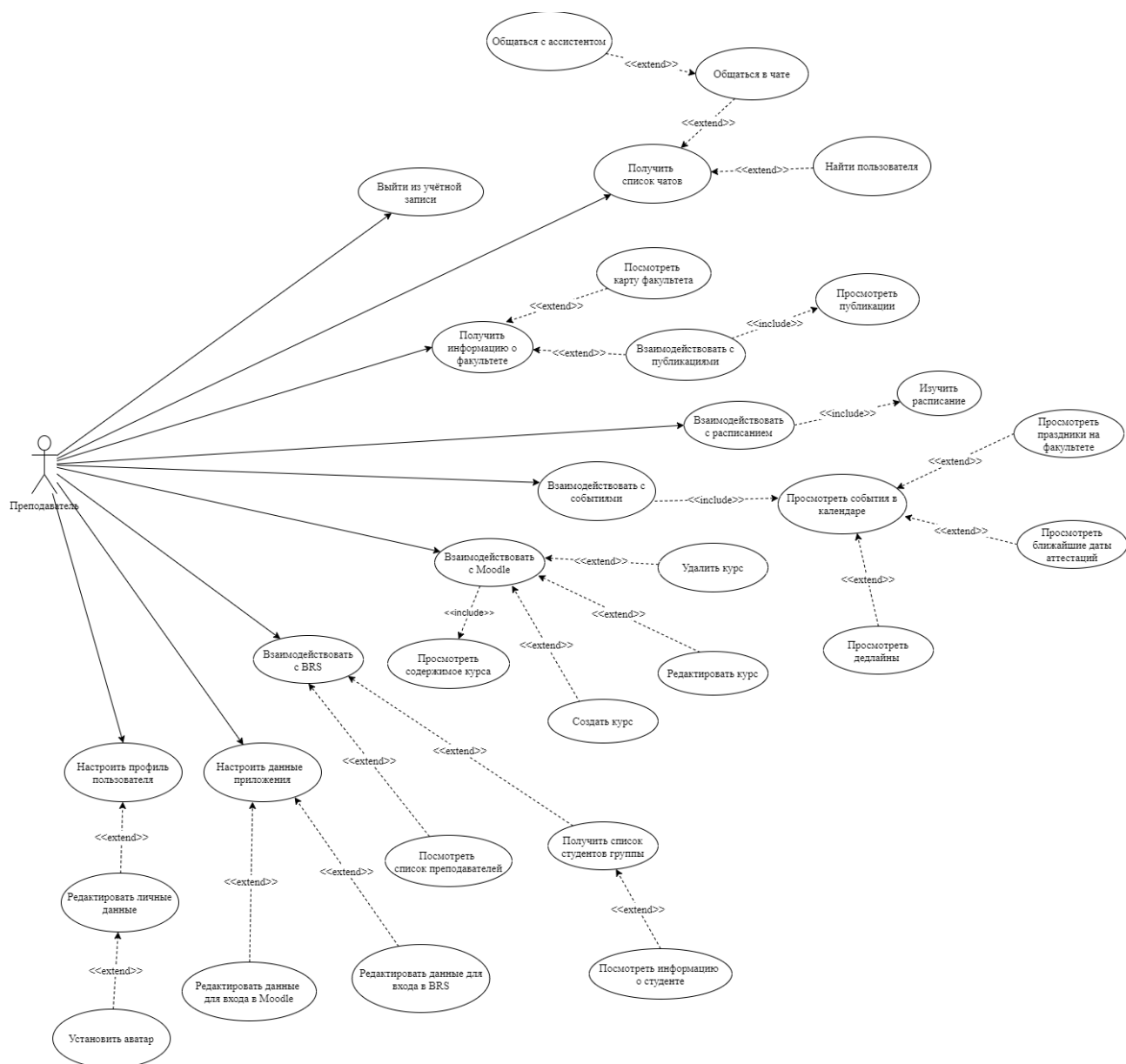


Рис. 12 - диаграмма прецедентов (teacher)

4.3.4 Диаграмма прецедентов (unauthorized user)

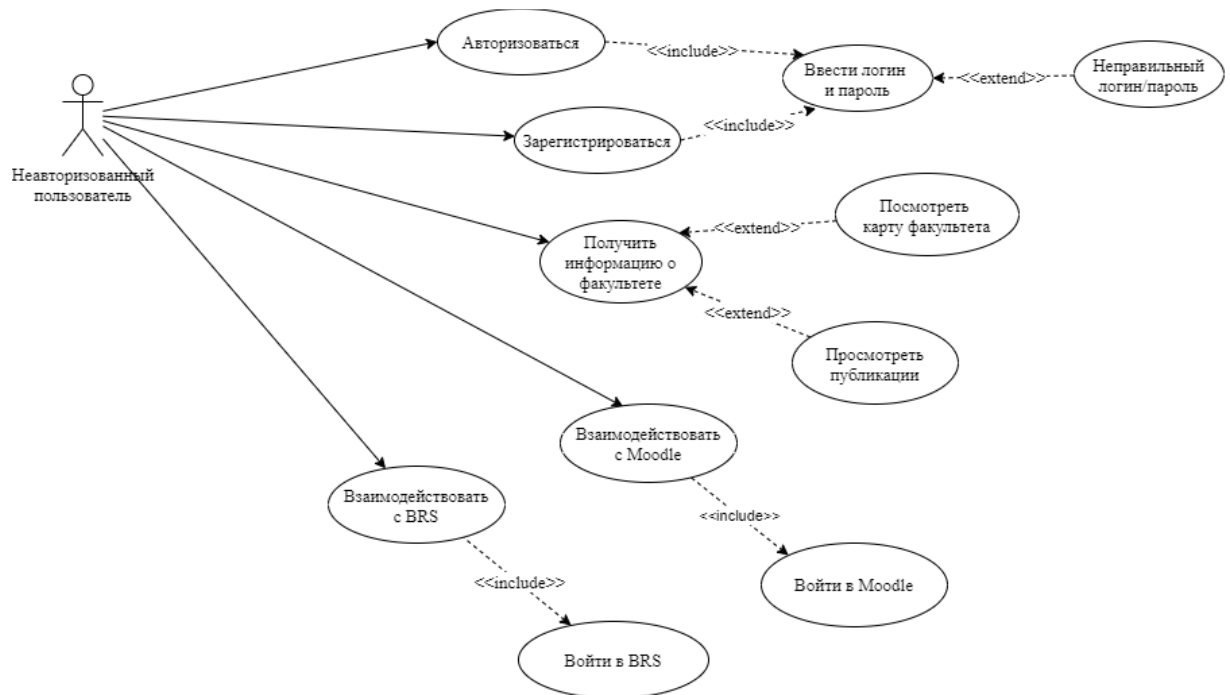


Рис. 13 - Диаграмма прецедентов (unauthorized user)

4.4 Диаграмма развёртывания

Далее приведена диаграмма развёртывания. Она используется для моделирования архитектуры системы, ее компонентов и их размещения на устройствах. Она показывает физическую структуру системы, то есть как ее компоненты располагаются на устройствах и как они взаимодействуют между собой.

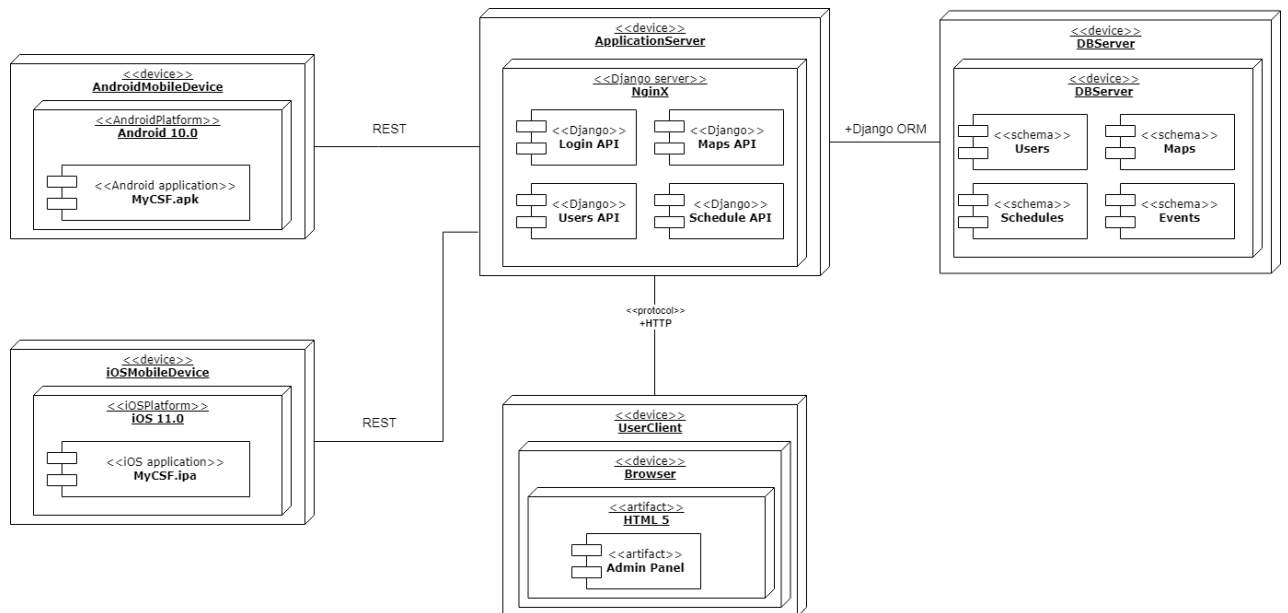


Рис. 14 - Диаграмма развёртывания

4.5 Диаграммы состояний

Далее приведены диаграммы состояний. Они используются для моделирования поведения системы в различных состояниях. Эти диаграммы показывают, как система реагирует на внешние события и как она изменяет свое состояние в ответ на эти события.

4.5.1 Диаграмма состояний (mobile app)

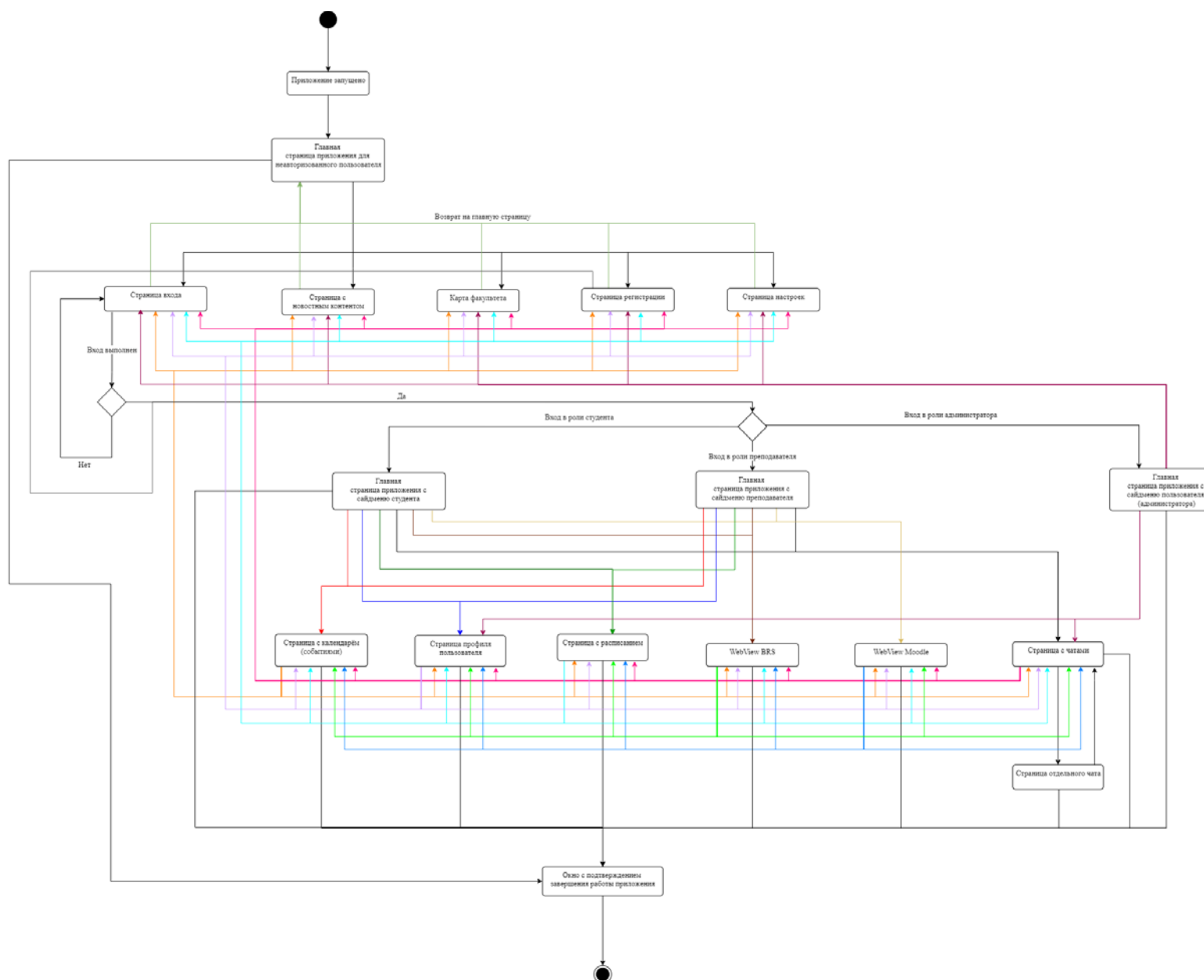


Рис. 15 - Диаграмма состояний (mobile app)

4.5.2 Диаграмма состояний (user)

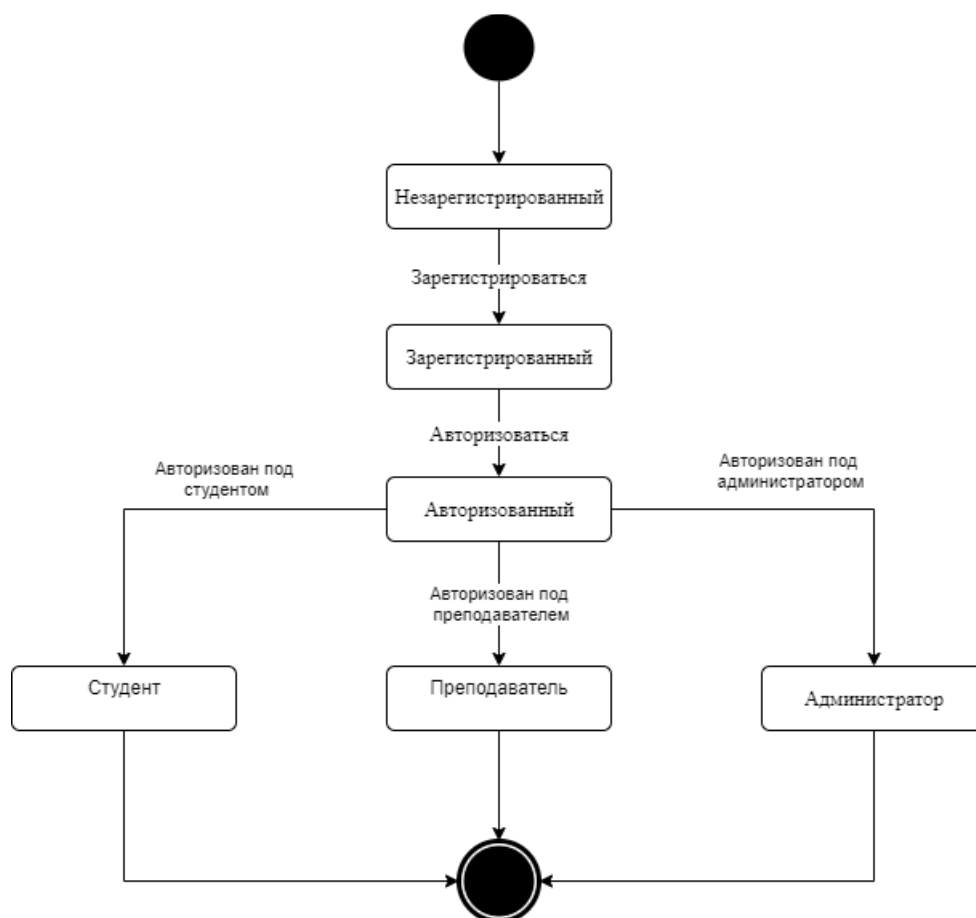


Рис. 16 - Диаграмма состояний (user)

4.6 Диаграммы сотрудничества

Далее будут приведены диаграммы сотрудничества. Они используются для моделирования взаимодействия между объектами в системе и показывают, как объекты обмениваются сообщениями и как они взаимодействуют друг с другом для выполнения определенной функции.



Рис. 17 - Авторизация пользователя

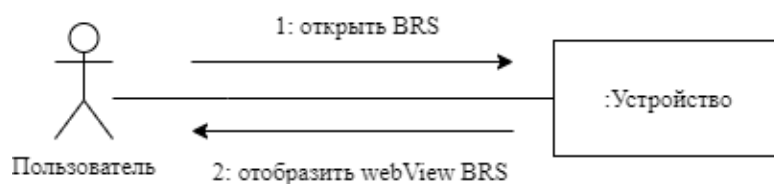


Рис. 18 - Взаимодействие с BRS

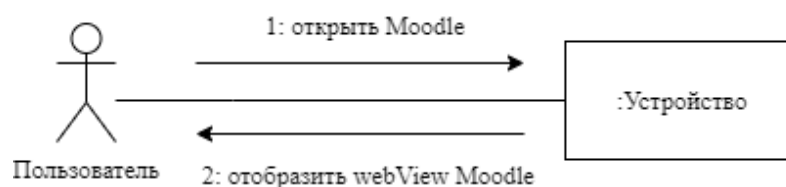


Рис. 19 - Взаимодействие с Moodle



Рис. 20 - Общение с пользователями



Рис. 21 - Просмотр карты факультета



Рис. 22 - Просмотр расписания



Рис. 23 - Просмотр события



Рис. 24 - Регистрация пользователя



Рис. 25 - Редактирование профиля пользователя

4.7 Диаграмма последовательности

Далее приведена диаграмма последовательности, позволяющая описать последовательность происходящих событий при взаимодействии пользователя с системой.

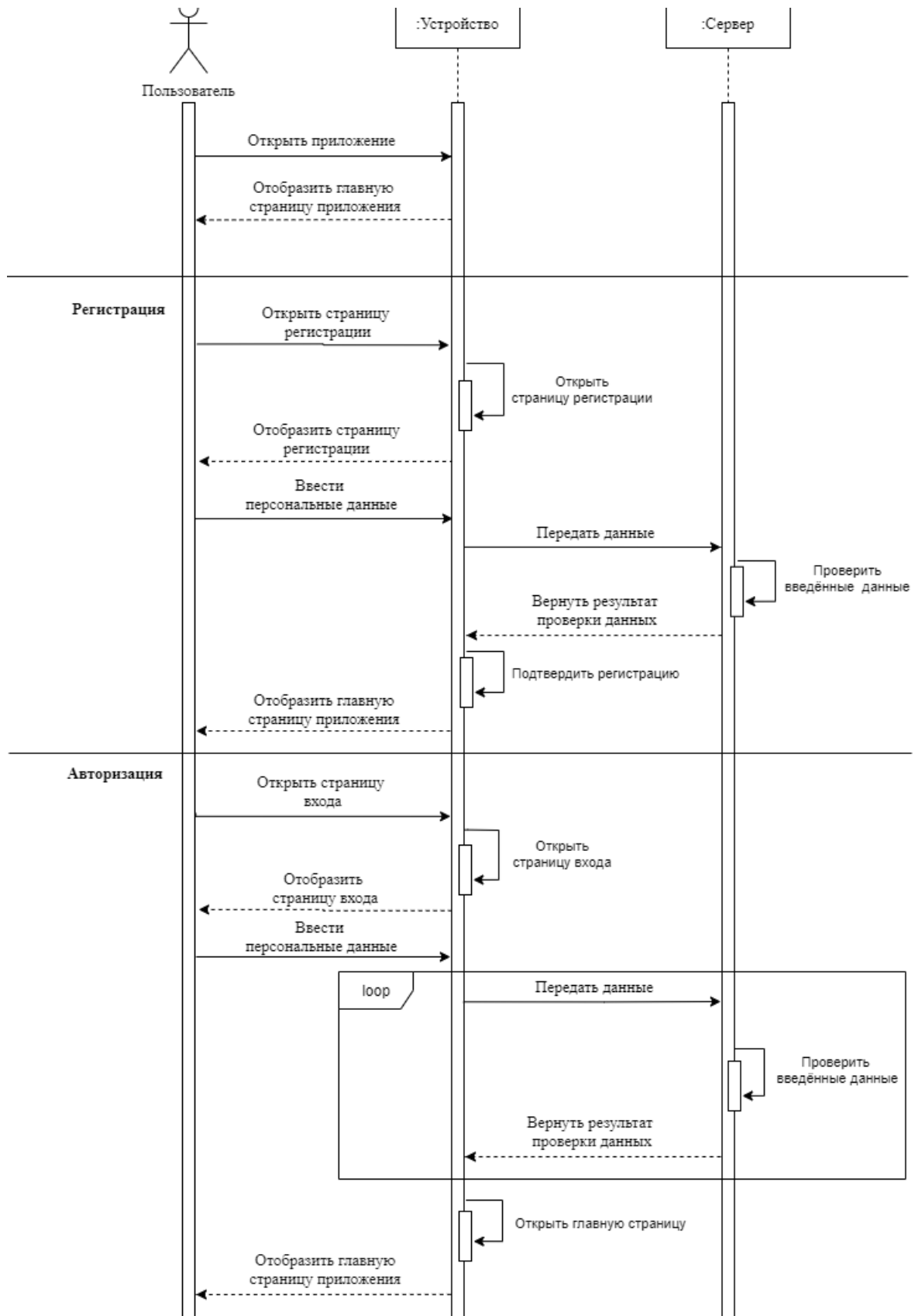


Рис. 26 - Диаграмма последовательности (часть1)

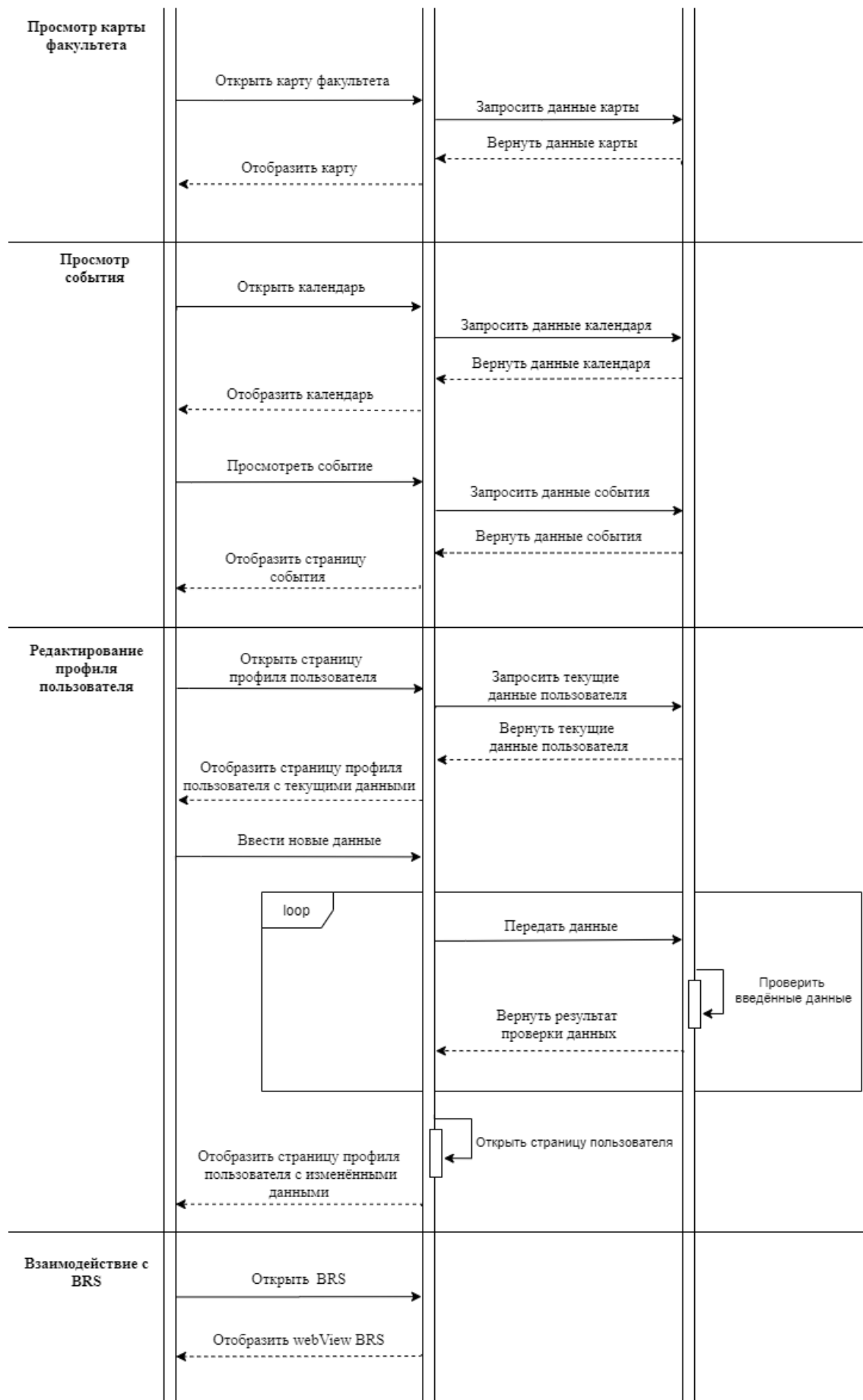


Рис. 27 - Диаграмма последовательности (часть 2)

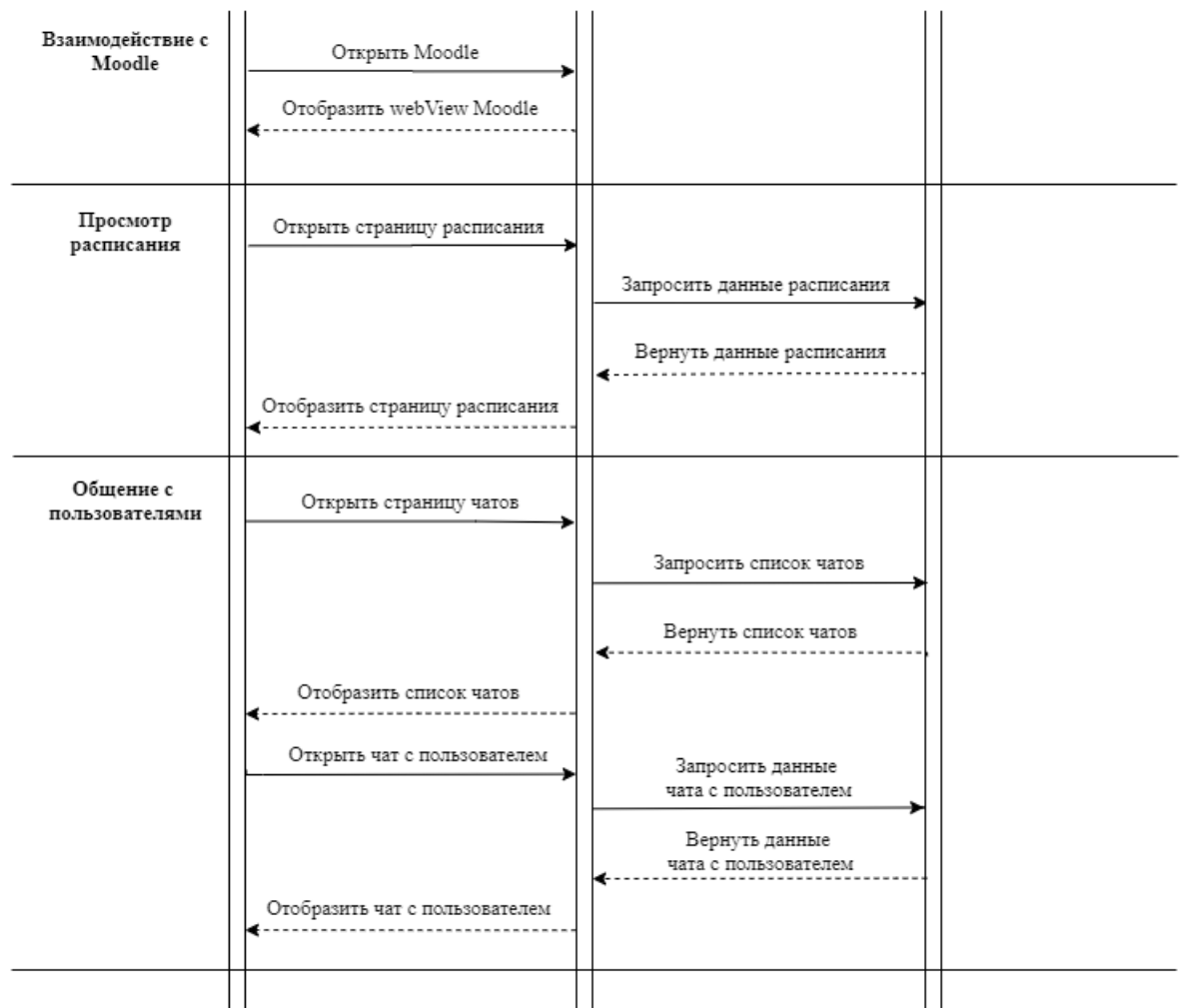


Рис. 28 - Диаграмма последовательности (часть 3)

4.8 Диаграмма классов

Диаграмма классов используется для моделирования структуры классов и их отношений между собой в системе.

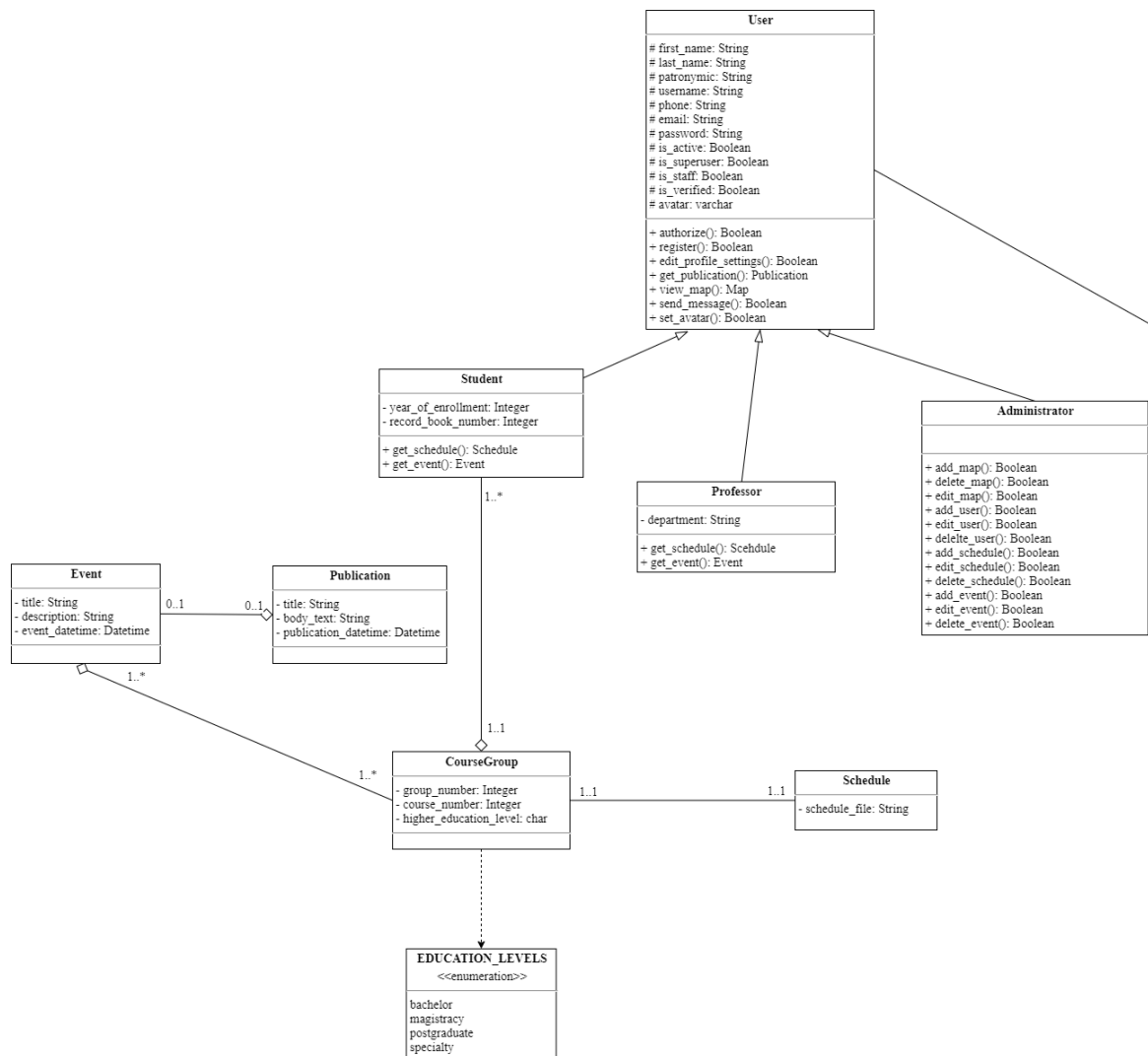


Рис. 29 - Диаграмма классов

4.9 Диаграмма объектов

Диаграмма объектов используется для моделирования конкретных объектов и их отношений в системе. Она показывает экземпляры классов, их атрибуты и связи между ними.

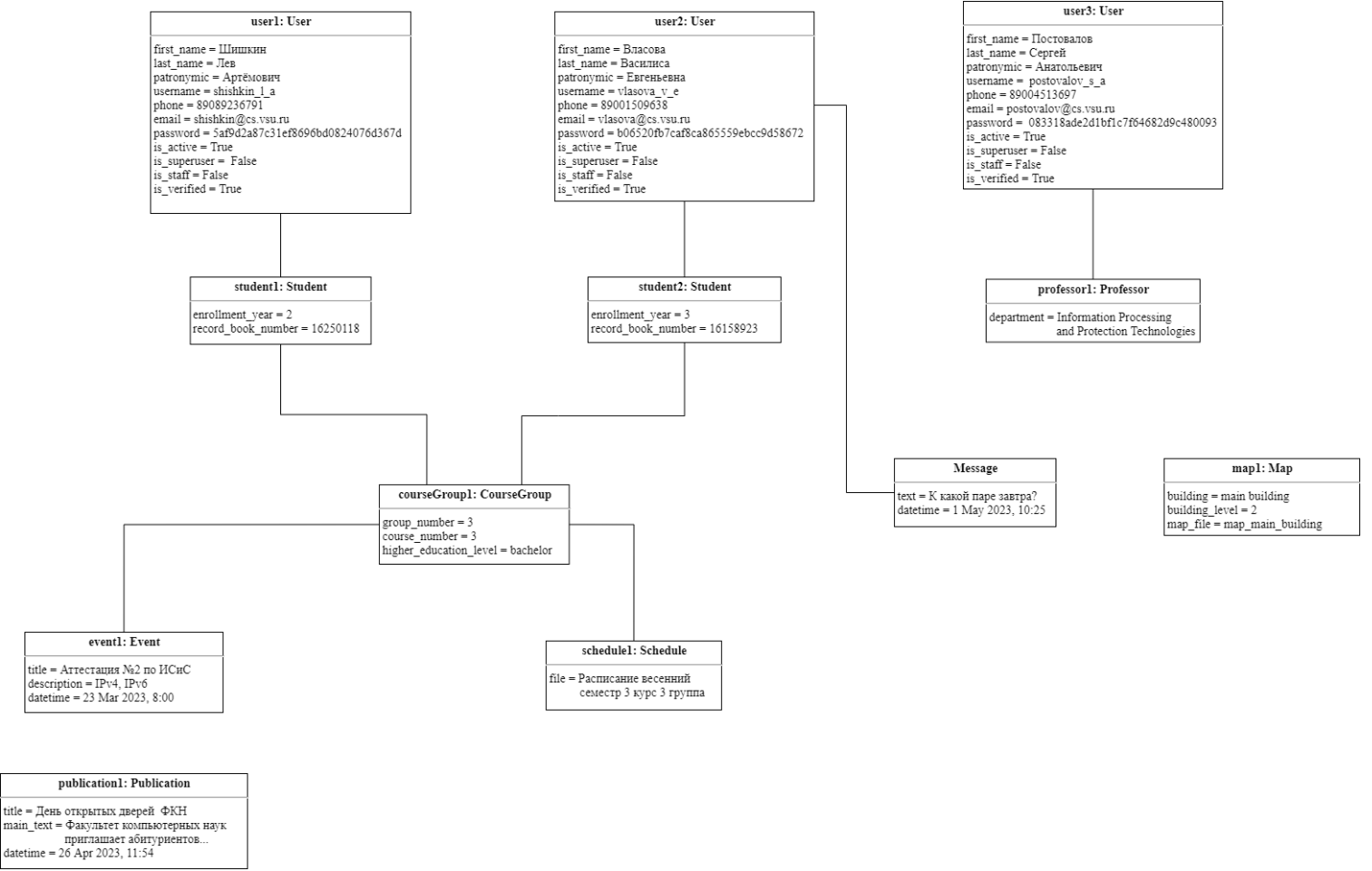


Рис. 30 - Диаграмма объектов

4.10 ER-диаграмма

ER (Entity-Relationship) диаграмма - это графическое представление структуры базы данных, которое используется для моделирования сущностей, их атрибутов и отношений между ними.

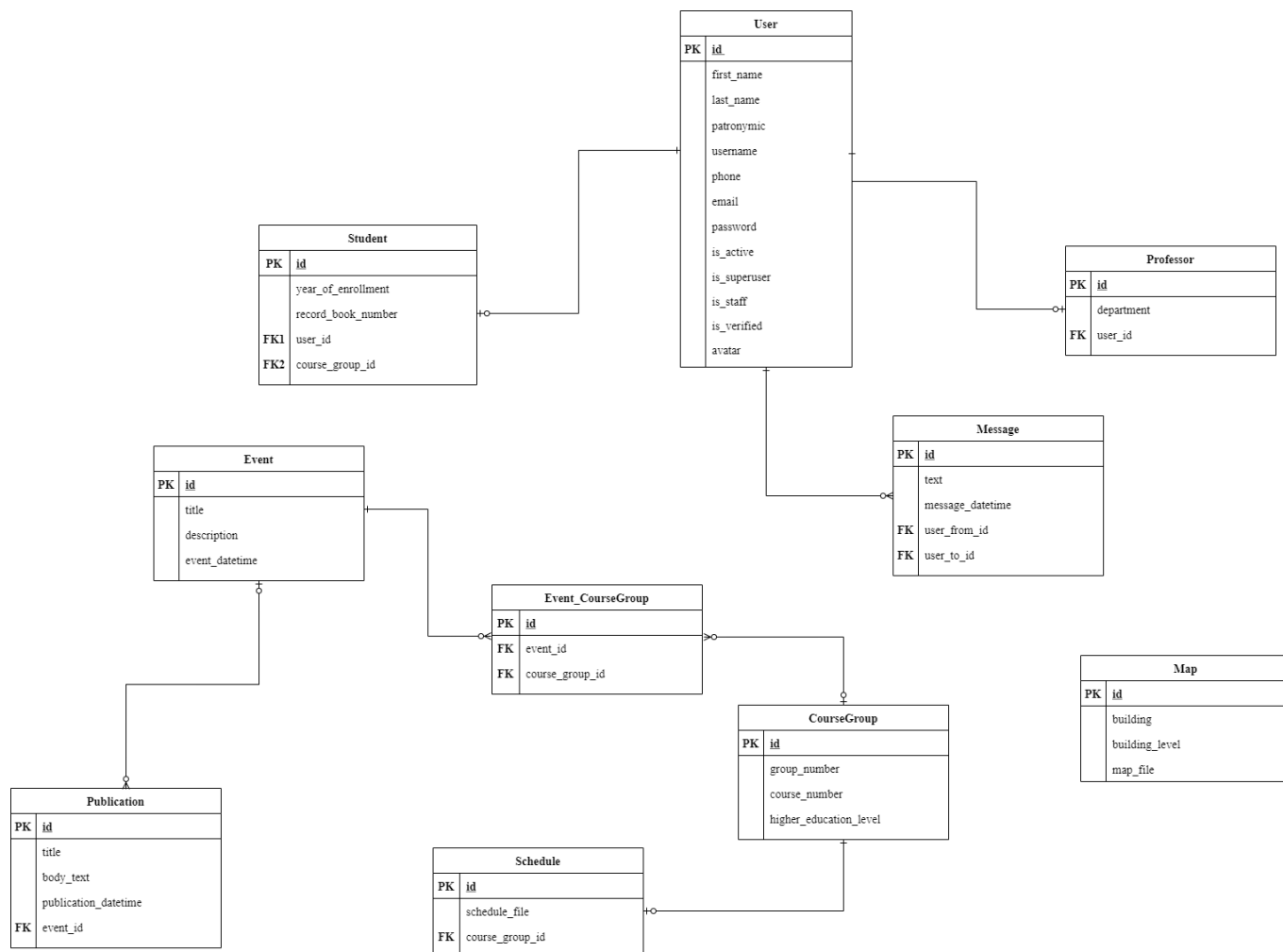


Рис. 31 - ER-диаграмма

4.11 Физическая модель базы данных

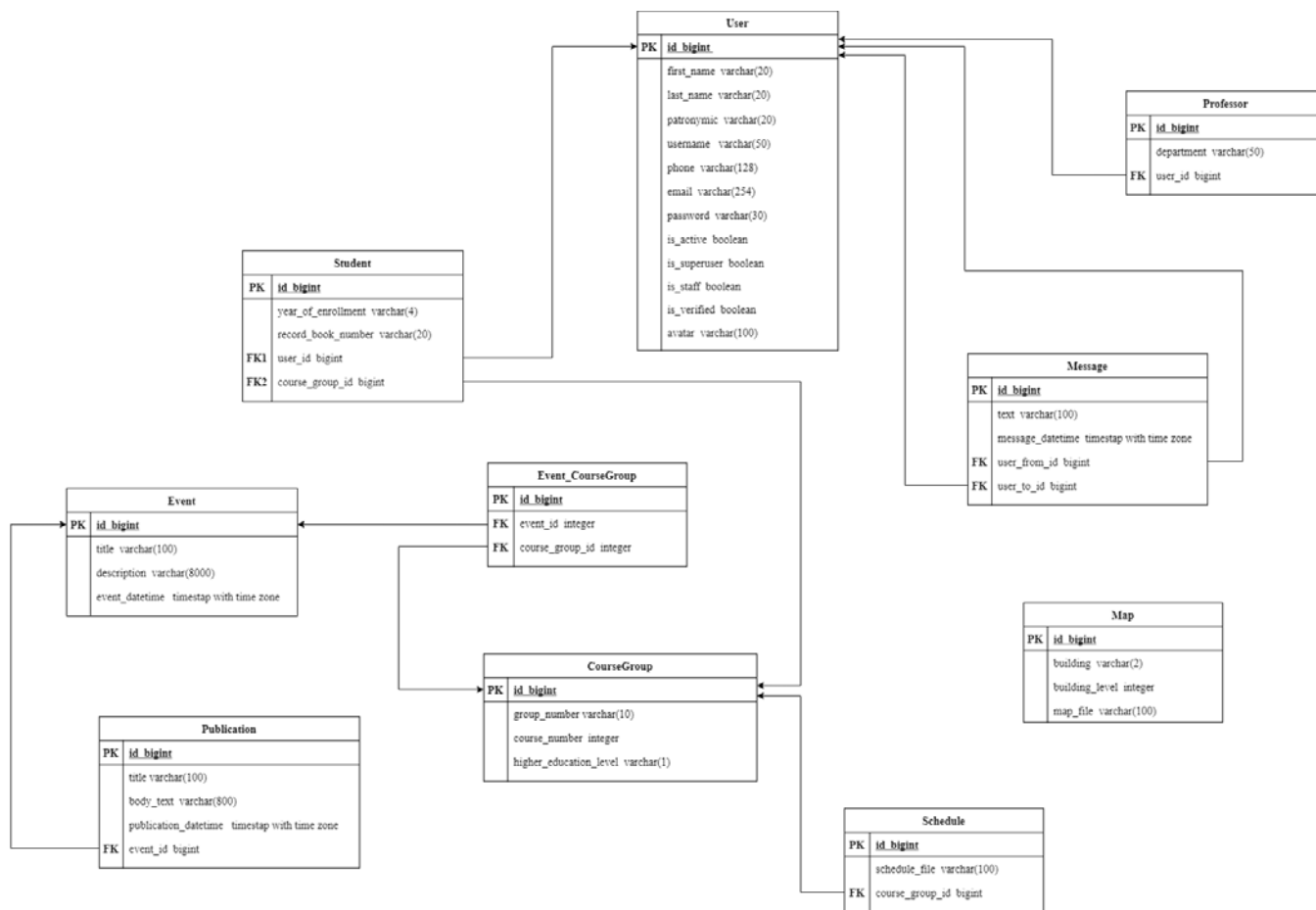


Рис. 32 - Физическая модель базы данных

Список использованных источников

- 1 Чем недовольны российские студенты [Электронный ресурс]. – Режим доступа: <https://student-app.ru/blog9>. – Заглавие с экрана. – (Дата обращения: 16.03.2023).