1 АНАЛИТИЧЕСКАЯ ЧАСТЬ

1.1 Роль автоматизации бизнес-процессов в современном высшем учебном заведении

История управления документами неразрывно связана с развитием человечества. Первые методы фиксации данных появились тысячи лет назад, когда люди начали вести учет на глиняных табличках и папирусе. В Средние века, когда в Европе формировались первые университеты (XI–XII вв.), делопроизводство стало обязательной частью учебной жизни. Списки студентов, разрешения на преподавание и финансовые отчеты требовали строгого учета. В XX веке пишущие машинки и первые вычислительные машины ускорили работу клерков, но суть оставалась прежней — информация хранилась на физическом носителе.

Сегодня ситуация изменилась. В эпоху цифровых технологий университеты вынуждены уходить от старых методов. Вуз — это уже не просто место для лекций, а крупная организация, похожая на небольшое государство со своими законами, бюджетом и населением. Объем данных здесь растет с каждым годом, и старые методы управления перестают справляться с нагрузкой.

Деятельность современного вуза состоит из множества связанных действий. Эти действия называют бизнес-процессами. В отличие от заводов или торговых фирм, где главная цель — получение денег, вузы работают ради качества образования и развития науки.

Бизнес-процессы вуза делятся на три большие группы:

Основные процессы. Это действия, ради которых создан вуз: обучение студентов и научная работа. Сюда входят планирование учебных курсов, проведение лекций, прием экзаменов, защита дипломов и публикация научных статей. Эти процессы создают ценность для общества.

Обеспечивающие процессы. Они нужны, чтобы основные процессы не останавливались. Это ремонт зданий, закупка компьютеров, работа общежитий, уборка помещений, обеспечение безопасности и работа библиотеки. Без них учеба станет невозможной.

Процессы управления. Это «мозг» организации. Сюда относятся стратегическое планирование, наем сотрудников, распределение финансов и, что важно для этой работы, — документационное обеспечение (делопроизводство). Именно документы связывают все отделы в единое целое.

На рисунке 1 (схема процессов) видно, что каждый вид деятельности требует передачи информации. Например, утверждение нового учебного плана — это сложная цепочка. Документ должен пройти через кафедру, факультет, учебный отдел и ректорат. Выдача простой справки студенту требует проверки его оценок и оплаты. Без компьютеров даже простые задачи отнимают много времени и сил у сотрудников.



Рисунок 1.Сравнение традиционного и автоматизированного документооборота в ВУЗе

Управленческие процессы можно выполнять двумя способами: традиционным (на бумаге) и автоматизированным (в электронном виде).

В традиционном подходе носителем информации служит бумага. Документ физически перемещается из одного кабинета в другой. Этот метод имеет глубокие системные недостатки:

Потеря времени. Документ может лежать в папке на подпись днями или неделями, если руководитель занят или уехал. Процесс останавливается.

Риск потери. Бумажный документ существует в единственном экземпляре. Если его случайно выкинули или положили не в ту папку, восстановить данные трудно. При передаче между корпусами документы часто теряются.

Трудности поиска. Чтобы найти приказ пятилетней давности, нужно идти в архив и перебирать коробки. Это занимает часы рабочего времени.

Дублирование работы. Секретари часто печатают одни и те же данные (ФИО, должности) в разных бланках вручную, что ведет к ошибкам.

Отсутствие контроля. Автор документа (например, декан) не знает, где сейчас находится его служебная записка. Он вынужден звонить и спрашивать, подписана ли бумага.

На рисунке 2 показан путь документа. В бумажном виде — это длинная линия с множеством остановок. В электронном виде — это мгновенный доступ.



Рисунок 2.Структура бизнес-процессов ВУЗа

Программные системы меняют этот порядок. В электронном документообороте (СЭД) файл не перемещается физически. Он лежит в одной базе данных, а система меняет его статус (например, с «На согласовании» на «Подписано»).

Преимущества такого подхода очевидны:

Параллельная работа. Бумажный документ читают по очереди. Электронный документ могут открыть сразу три проректора и внести свои замечания одновременно. Это ускоряет процесс в разы.

Прозрачность. Система записывает каждое действие. Видно, кто создал документ, кто его открыл, сколько времени читал и что исправил. Сказать «я не получал документ» становится невозможно.

Единые стандарты. Система предлагает готовые шаблоны. Сотрудникам не нужно думать над оформлением и шрифтами, они просто заполняют поля. Это снижает число ошибок.

Экономия. Вуз тратит меньше денег на бумагу, картриджи для принтеров и обслуживание оргтехники. Также освобождается место, которое раньше занимали шкафы с архивами.

Однако переход на «цифру» — это не просто установка программы. Это сложная работа, которая встречает препятствия:

Психология людей. Сотрудники, особенно старшего возраста, привыкли к бумаге. Им трудно учиться новому, они боятся нажать «не ту кнопку».

Пересмотр правил. Нельзя просто перенести хаос с бумаги в компьютер. Перед запуском системы нужно упростить маршруты документов, убрать лишние согласования и переписать инструкции.

Технические вопросы. Нужны надежные серверы, защита от хакеров и регулярное сохранение копий данных, чтобы сбой электричества не уничтожил работу вуза.

Юридическая сила. Чтобы электронный файл стал документом, нужно внедрить систему входа по паролю или использовать цифровую подпись.

Для современного вуза автоматизация — это насущная потребность. Университет обслуживает тысячи людей. Поток заявлений, приказов и отчетов огромен. Ручная работа с таким объемом ведет к сбоям и снижает качество управления.

Переход к автоматизированным системам решает главную задачу: освобождает время людей. Преподаватели должны учить, а ученые — делать открытия, а не носить бумаги по коридорам. Хорошая информационная система работает как нервная система организма: она быстро и точно передает сигналы от руководства к исполнителям и обратно, делая вуз современным и эффективным.

1.2 Исследование предметной области. Классификация и анализ существующих систем электронного документооборота

Рынок программного обеспечения предлагает множество решений для автоматизации работы с документами. Чтобы выбрать верный путь разработки, необходимо изучить, какие системы уже существуют, как они работают и почему готовые продукты не всегда подходят для вуза.

Системы электронного документооборота (СЭД) — это класс программ, которые помогают создавать, изменять, хранить и передавать документы внутри организации.

Классификация систем

Существует несколько признаков, по которым делят эти системы. Понимание этой классификации помогает определить место создаваемой системы среди аналогов.

1. По функциональности:

Системы делопроизводства (EDMS). Их главная цель — строгий учет. Они фиксируют входящие и исходящие письма, приказы и распоряжения. Здесь важна жесткая карточка документа и регистрационный номер. (Пример: «1С:Документооборот»).

Системы управления контентом (ECM). Это большие хранилища файлов. Они позволяют хранить не только официальные документы, но и любые рабочие файлы, чертежи, видео и макеты. (Пример: Microsoft SharePoint).

Системы управления бизнес-процессами (BPM). Здесь акцент делается не на самом документе, а на маршруте его движения. Система следит за сроками, назначает задачи и пересылает данные от сотрудника к сотруднику.

2. По способу распространения:

Коммерческие (Proprietary). Программы с закрытым кодом. Организация платит за лицензию, поддержку и обновления.

С открытым кодом (Open Source). Бесплатные платформы, код которых можно менять. Они не требуют покупки лицензий, но требуют сильных программистов в штате для настройки.

Заказные (Custom). Системы, созданные специально для конкретной организации с учетом ее уникальных правил. Именно к этому типу относится система, разрабатываемая в данной работе.

На рисунке 3 представлена схема классификации систем электронного документооборота.



Рисунок 3 — Классификация систем электронного документооборота

Анализ популярных решений на рынке

Для сравнения были выбраны наиболее распространенные системы, используемые в странах СНГ и Казахстане.

1. «1С:Документооборот» Это самое популярное решение в нашем регионе. Система построена на платформе «1С:Предприятие».

Плюсы: Легкая интеграция с бухгалтерией и кадрами. Соответствие местным законам. Много специалистов по настройке.

Минусы: Высокая стоимость. Лицензирование происходит «по рабочим местам». Для вуза с 1000 сотрудников и 5000 студентов покупка лицензий обойдется слишком дорого. Интерфейс часто перегружен лишними функциями.

2. Directum / Directum RX Мощная корпоративная система для крупного бизнеса.

Плюсы: Готовые модули для канцелярии и договоров. Умный поиск и работа через веб-браузер.

Минусы: Система ориентирована на заводы и банки, а не на образование. В ней нет понятий «кафедра», «сессия» или «учебный план». Адаптация под вуз потребует долгой доработки.

3. Alfresco Community Edition Популярная западная система с открытым исходным кодом.

Плюсы: Бесплатная базовая версия. Гибкая архитектура. Работает на любых операционных системах.

Минусы: Сложность в установке. Англоязычная документация. Без платной поддержки настроить систему под нужды казахстанского вуза крайне трудно.

Проблемы использования готовых решений в ВУЗе

Анализ показывает, что универсальные системы плохо подходят для учебного заведения по трем причинам:

Экономический фактор. Большинство коммерческих систем требуют оплату за каждого пользователя. В вузе пользователями являются не только сотрудники администрации, но и преподаватели и студенты. Покупка тысяч лицензий делает внедрение невыгодным.

Избыточность функций. Крупные системы содержат сотни функций для бизнеса: продажи, склад, CRM. Вузу это не нужно. Лишние кнопки только путают пользователей.

Специфика процессов. Готовые системы не учитывают иерархию вуза (Факультет -> Кафедра -> Группа) и сезонность работы (приемная комиссия, сессия).

На рисунке 4 показано сравнение подходов к выбору программного обеспечения.



Рисунок 4 — Сравнение готовых коммерческих решений и собственной разработки

Анализ предметной области показал, что рынок насыщен универсальными системами. Однако для специфических задач высшего учебного заведения они оказываются либо слишком дорогими, либо неудобными.

Использование решений с открытым кодом требует серьезной переработки силами дорогих специалистов. Коммерческие продукты требуют постоянных платежей. В связи с этим, наиболее разумным решением является разработка собственной информационной системы.

Собственная разработка позволит:

1. Создать простой интерфейс, понятный преподавателям и студентам.
2. Учесть структуру вуза и типы учебных документов.
3. Избежать затрат на лицензии.
4. Использовать современные веб-технологии для удаленного доступа.

1.3 DDD и ограниченные контексты

После обоснования необходимости собственной разработки, следующим этапом является глубокое погружение в структуру данных, которые система будет обрабатывать. Понимание того, как движется информация внутри университета, — это фундамент для проектирования базы данных и интерфейсов.

Документопоток — это упорядоченное движение документов с момента их создания или получения до завершения исполнения, отправки или сдачи в архив.

Специфика вуза заключается в его сложной иерархии. Это не просто вертикаль «Начальник — Подчиненный». Это матричная структура, где взаимодействуют администрация (Ректорат), учебные подразделения (Деканаты, Кафедры) и обеспечивающие службы (Бухгалтерия, Отдел кадров, АХЧ). Количество участников процесса документооборота в среднем вузе исчисляется тысячами (с учетом студентов), а ежегодный объем документов — десятками тысяч.

Классификация документации по направлению движения

Весь массив документации традиционно делят на три основных потока. Однако для проектирования системы важно понимать не только направление, но и специфику обработки каждого потока.

1. Входящий поток (Поступление извне) Это документы, которые дают старт многим внутренним процессам.

Состав: Письма из Министерства науки и высшего образования, нормативные акты, коммерческие предложения, судебные запросы, жалобы родителей или студентов.

Маршрут: Канцелярия (регистрация) → Ректор (визирование) → Исполнитель (декан или нач. отдела).

Проблема: Часто теряется время на этапе «Ректор — Исполнитель». Бумажное письмо может лежать в приемной несколько дней.

2. Исходящий поток (Отправка во внешний мир) Реакция вуза на входящие запросы или инициативные письма.

Состав: Отчеты, справки для госорганов, ответы на жалобы, письма партнерам.

Маршрут: Исполнитель (составление) → Зав. кафедрой (согласование) → Проректор (подпись) → Канцелярия (отправка).

Проблема: Документ часто возвращают на доработку из-за ошибок в оформлении, что перезапускает весь круг согласования.

3. Внутренний поток (Циркуляция внутри вуза) Самый объемный сегмент, занимающий до 80% всего документооборота. Именно здесь происходит основная оперативная работа. Внутренние документы делятся на:

Организационно-распорядительные: Приказы ректора, распоряжения проректоров.

Информационно-справочные: Служебные записки, акты, протоколы заседаний, заявления сотрудников.

Учебно-методические: Учебные планы, ведомости, индивидуальные планы преподавателей.

На рисунке 5 представлена общая схема взаимодействия этих потоков.

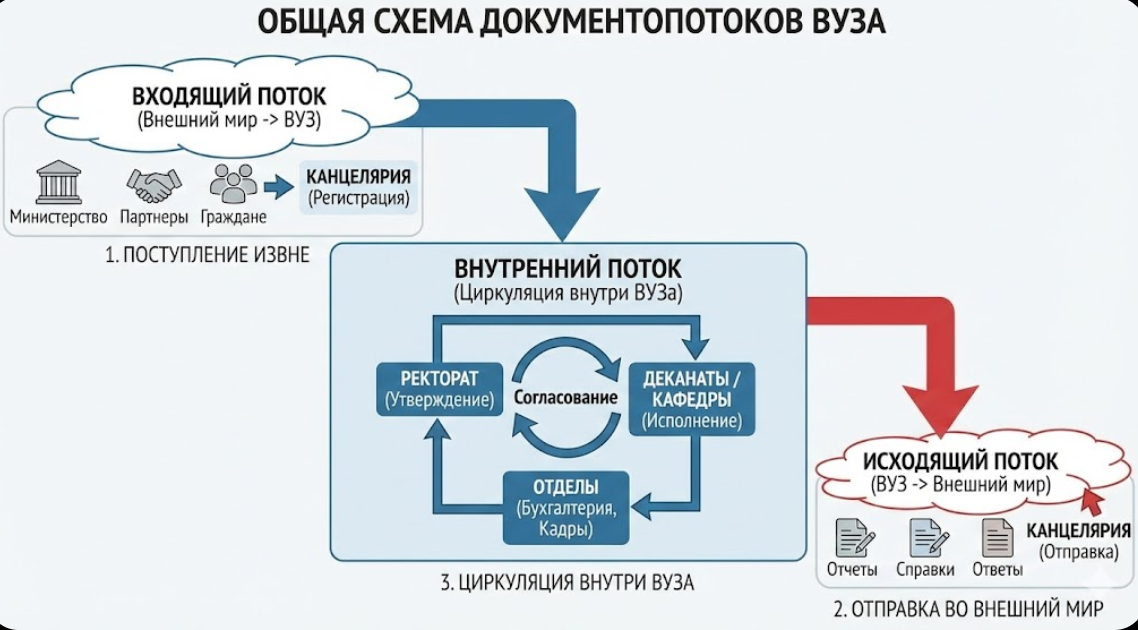


Рисунок 5 — Общая схема документопотоков ВУЗа

Жизненный цикл внутреннего документа

Для правильного проектирования системы (особенно модуля смены статусов) необходимо разобрать жизненный цикл документа. В условиях вуза любой документ проходит пять обязательных стадий.

Стадия 1: Инициация и создание проекта Автор (например, преподаватель) создает черновик документа.

Текущие проблемы: Отсутствие единых шаблонов. Преподаватели пишут заявления в свободной форме, часто забывая указать нужные реквизиты (дату, кому адресовано), из-за чего документ позже отклоняют.

Стадия 2: Согласование (Визирование) Самый длительный этап. Прежде чем документ попадет к лицу, принимающему решение, его должны одобрить заинтересованные стороны.

Пример: Заявление на покупку компьютера. Маршрут: Преподаватель → Зав. кафедрой → Декан → Главный бухгалтер → Проректор по АХЧ.

Текущие проблемы: Физический поиск подписантов. Если декана нет на месте, процесс встает. Часто документ теряется именно на этом этапе («Я оставлял папку у секретаря, ищите там»).

Стадия 3: Утверждение (Подписание) Финальная подпись руководителя (Ректора или Проректора), придающая документу юридическую силу.

Текущие проблемы: Накопление «стопок» на подпись. Руководитель подписывает документы пакетно, раз в несколько дней, что тормозит срочные процессы.

Стадия 4: Регистрация Присвоение уникального номера в журнале регистрации. Без этого документ не существует бюрократически.

Текущие проблемы: Ведение бумажных журналов или разрозненных Excel-файлов, что делает невозможным быстрый сквозной поиск.

Стадия 5: Исполнение и контроль Документ передается исполнителю для работы, а контролер следит за сроками.

Текущие проблемы: Отсутствие автоматических напоминаний. О поручении часто вспоминают только тогда, когда сроки уже прошли.

На рисунке 6 детально изображен этот маршрут на примере служебной записки.

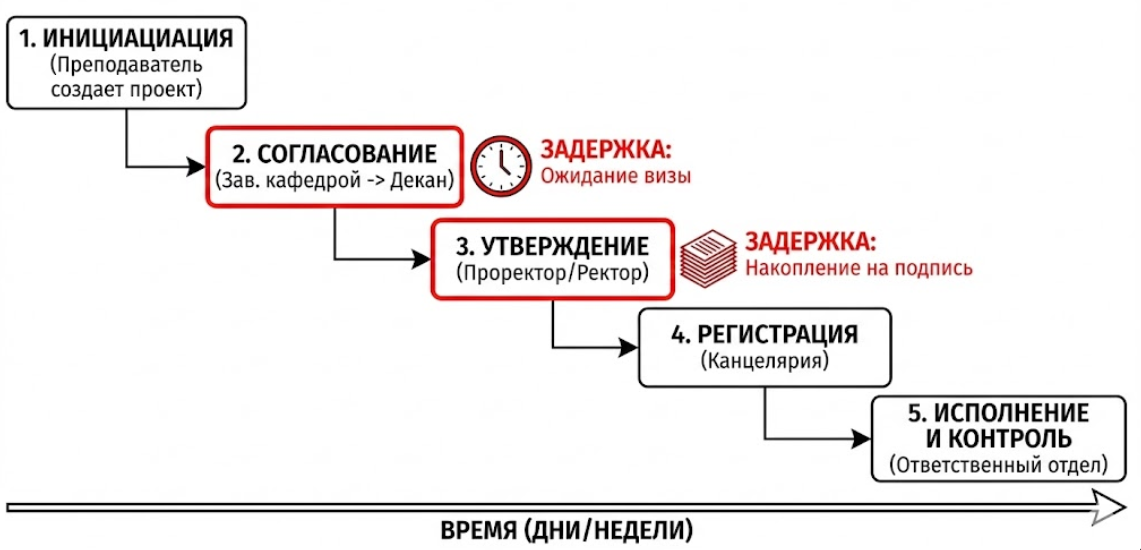


Рисунок 6 — Маршрут движения служебной записки и точки задержек

Анализ проблемных зон в текущей структуре

Анализ действующих маршрутов в типичном вузе позволяет выделить ключевые системные недостатки, которые должна устранить разрабатываемая ИС.

1. Дублирование функций и данных Одна и та же информация вводится многократно. Например, данные о студенте вводятся в приемной комиссии, затем вручную переписываются в деканате в журнал группы, затем — в библиотеке. Это порождает ошибки.

2. Разрыв информационных связей Часто подразделения не знают о действиях друг друга. Учебный отдел может планировать расписание для преподавателя, который уже подал заявление на увольнение в отдел кадров, но приказ еще не дошел до учебной части. Единая система должна устранить этот временной лаг.

3. Сложность контроля исполнительской дисциплины В бумажном потоке практически невозможно получить оперативную сводку: «Сколько поручений не выполнила кафедра ИТ за прошлый месяц?». Для такого отчета нужно вручную перебирать журналы. Руководство вуза принимает решения в условиях нехватки оперативной информации.

4. Высокие непроизводственные затраты времени Проведенный хронометраж показывает, что секретарь кафедры тратит до 30% рабочего времени на хождение по кабинетам для сбора подписей и передачу бумаг. В масштабах вуза это тысячи человеко-часов в год, оплачиваемых из бюджета, но не приносящих полезного результата.

Анализ структуры документопотоков показал, что управление современным вузом — это работа с огромным массивом информации, движущейся по сложным траекториям. Традиционные методы управления этими потоками достигли своего предела эффективности.

Ключевая задача разрабатываемой системы — не просто перевести текст с бумаги на экран, а оптимизировать маршруты:

Обеспечить параллельное согласование (все видят документ одновременно).

Внедрить жесткие шаблоны, исключающие ошибки на старте.

Создать единое информационное пространство, где статус документа виден всем участникам в реальном времени.

Это понимание процессов ложится в основу технического задания и проектирования базы данных в следующей главе.

1.4 Обзор современных технологий для разработки корпоративных информационных систем (Client-Server, Model View Controller)

Выбор правильной архитектуры — ключевой этап при создании любой информационной системы. От этого зависит, насколько быстро будет работать программа, легко ли будет ее обновлять и смогут ли ей пользоваться тысячи людей одновременно.

Для разработки системы документооборота ВУЗа наиболее подходящим является подход, основанный на веб-технологиях. Это позволяет отказаться от установки программ на компьютер каждого сотрудника. Рассмотрим две фундаментальные концепции, на которых базируются современные веб-системы: архитектуру «Клиент-Сервер» и шаблон проектирования MVC.

Трехзвенная архитектура «Клиент-Сервер»

Раньше программы устанавливались на каждый компьютер отдельно (так называемые «толстые клиенты»). Если выходила новая версия, системный администратор должен был обойти 500 компьютеров и обновить программу вручную. Для вуза такой подход неприемлем.

Современным стандартом является трехзвенная архитектура (Three-Tier Architecture). Она разделяет систему на три независимых уровня:

Уровень представления (Client Tier). Это то, что видит пользователь. В нашем случае это веб-браузер (Chrome, Firefox). Его задача — красиво показать кнопки, формы и таблицы, а затем отправить действия пользователя на сервер.

Уровень логики приложения (Application Tier). Это «мозг» системы, расположенный на сервере. Здесь работают программные скрипты (написанные, например, на Python или Java). Они проверяют права доступа, формируют документы и управляют маршрутами согласования.

Уровень данных (Data Tier). Это база данных (например, PostgreSQL или MySQL). Ее единственная задача — надежно хранить информацию и быстро выдавать ее по запросу сервера.

Преимущества такой схемы для ВУЗа:

Централизация. Обновление происходит только на одном сервере. У пользователей новая версия появляется мгновенно при обновлении страницы.

Доступность. Работать с документами можно не только с рабочего ПК, но и из дома или с телефона.

Низкие требования к «железу». Вся сложная работа выполняется мощным сервером. Компьютеры сотрудников могут быть старыми и слабыми, так как от них требуется только запустить браузер.

На рисунке 7 показана схема взаимодействия этих уровней.

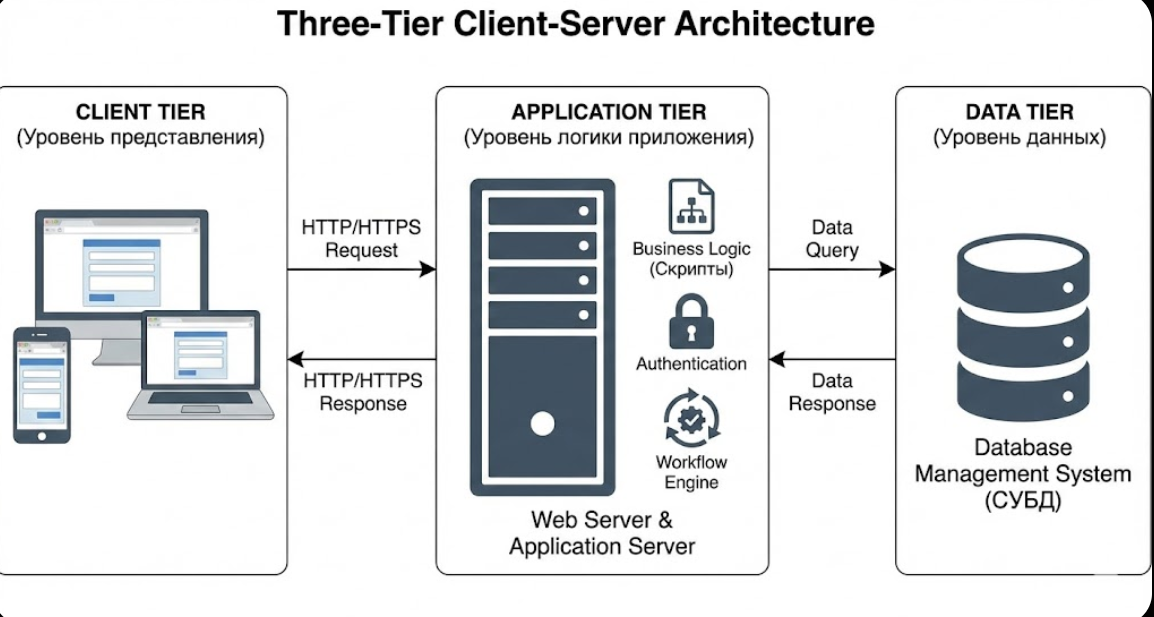


Рисунок 7 — Трехзвенная архитектура «Клиент-Сервер»

Шаблон проектирования MVC (Model-View-Controller)

Если архитектура «Клиент-Сервер» описывает физическое устройство системы, то MVC описывает, как организован программный код внутри неё.

При разработке крупных систем нельзя писать весь код в одном файле. Это приведет к путанице («спагетти-коду»), который невозможно поддерживать. Шаблон MVC предлагает разделить код на три компонента:

Модель (Model) — Данные. Этот компонент отвечает за структуру данных и работу с базой. Модель «знает», что у Студента есть ФИО и номер группы, а у Документа — дата и автор. Модель не знает, как эти данные будут показаны на экране.

Представление (View) — Внешний вид. Этот компонент отвечает за отображение данных пользователю (HTML-страницы). Он получает данные от Контроллера и рисует их. Представление не принимает решений и не делает расчетов.

Контроллер (Controller) — Управление. Это связующее звено. Когда пользователь нажимает кнопку «Создать приказ», сигнал идет в Контроллер. Контроллер проверяет права, обращается к Модели, чтобы записать данные, а затем говорит Представлению: «Покажи сообщение об успехе».

Почему MVC важен для магистерского проекта:

Разделение труда. Дизайнер может менять цвета кнопок в Представлении (View), не боясь сломать логику записи в базу данных (Model).

Легкость изменений. Если вуз решит изменить формат отображения приказа, нужно будет поправить только один файл шаблона, не трогая основной код программы.

На рисунке 8 наглядно представлен поток данных в паттерне MVC.

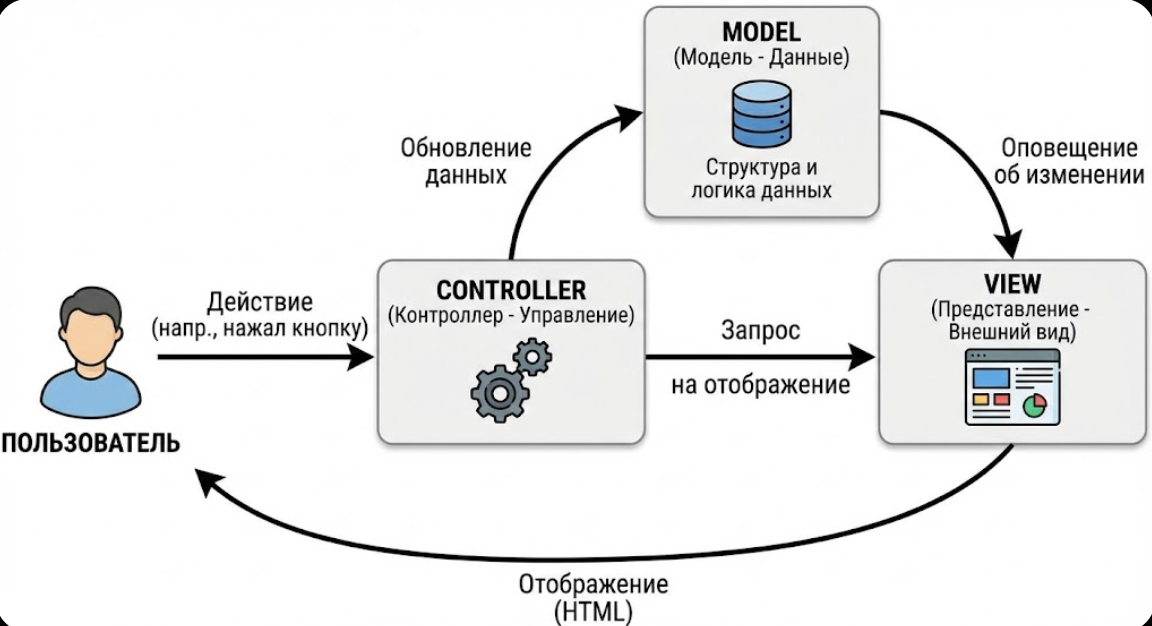


Рисунок 8 — Схема взаимодействия компонентов MVC

Технологии реализации (Веб-стек)

Основываясь на выбранной архитектуре, для реализации системы будет использоваться стандартный веб-стек. В качестве языка программирования на стороне сервера целесообразно использовать языки, поддерживающие MVC «из коробки» (например, Python с фреймворком Django или PHP с Laravel).

Для взаимодействия между клиентом и сервером используется протокол HTTP. Это универсальный язык общения в интернете. Клиент отправляет запрос (Request), сервер возвращает ответ (Response). Такой подход гарантирует, что система будет работать стабильно и предсказуемо, используя стандартные порты и протоколы безопасности.

Вывод по главе

Проведенный обзор технологий позволяет сформулировать технический фундамент будущей системы:

Система будет веб-ориентированной, что снимает проблему установки ПО на рабочие места.

Использование трехзвенной архитектуры обеспечит надежное хранение данных и централизованное управление.

Применение паттерна MVC позволит создать чистый, структурированный код, который легко поддерживать и дорабатывать в будущем.

1.5 Проблемы информационной безопасности и разграничения прав доступа в систему электронного документооборота

Внедрение любой информационной системы создает новые риски. Если раньше, чтобы украсть документ, нужно было взломать дверь кабинета, то теперь злоумышленник может находиться в любой точке мира.

В контексте высшего учебного заведения вопрос безопасности стоит особенно остро. Вуз — это не закрытый военный объект, а открытая среда, где тысячи студентов подключаются к сети со своих ноутбуков и телефонов. При этом в базах данных хранятся критически важные сведения: оценки, личные дела, данные о зарплатах преподавателей и приказы об отчислении.

Основные принципы безопасности (Триада CIA)

При проектировании защиты системы документооборота необходимо опираться на три фундаментальных свойства информации, которые принято называть «Триадой информационной безопасности» (CIA Triad):

Конфиденциальность (Confidentiality). Гарантия того, что информация доступна только тем, кто имеет на это право.

Пример нарушения: Студент получил доступ к ведомости другой группы или лаборант увидел размер зарплаты ректора.

Целостность (Integrity). Гарантия того, что информация не была изменена несанкционированным образом.

Пример нарушения: Взлом базы данных и изменение оценки за экзамен с «3» на «5».

Доступность (Availability). Гарантия того, что авторизованные пользователи могут получить доступ к информации в нужное время.

Пример нарушения: Система «упала» в день подачи заявлений на грант, и документы не были приняты вовремя.

На рисунке 9 показана схема этой триады. Нарушение любого из углов треугольника ведет к сбою всей системы безопасности.



Рисунок 9 — Триада информационной безопасности

Угрозы безопасности в университетской среде

В отличие от коммерческих фирм, в вузе основные угрозы часто исходят не от внешних хакеров, а изнутри. Классификация угроз выглядит следующим образом:

1. Внутренние угрозы (Инсайдеры):

Любопытство студентов. Технически грамотные студенты часто пытаются найти уязвимости в системе вуза ради интереса или повышения оценок (SQL-инъекции, перебор паролей).

Ошибки персонала. Секретарь может случайно удалить важный приказ или отправить документ не тому получателю.

Злоупотребление правами. Сотрудник, имеющий доступ к базе, может продавать информацию или менять данные по просьбе знакомых.

2. Внешние угрозы:

DDoS-атаки. Блокировка работы сервера путем отправки множества ложных запросов (например, во время приемной кампании).

Фишинг. Рассылка поддельных писем преподавателям с просьбой ввести пароль от системы.

3. Технические сбои:

Отказ жестких дисков, сбои электропитания, ошибки в программном коде.

Проблема разграничения прав доступа

Самая сложная задача при разработке университетской системы — это правильное распределение прав. В вузе сложная структура подчинения, и права доступа не всегда очевидны.

Классический метод «один пользователь — одни права» здесь не работает. В вузе один человек может совмещать несколько ролей: он может быть одновременно доцентом кафедры (читает лекции), куратором группы (видит личные дела студентов) и членом ученого совета (голосует за решения).

Для решения этой проблемы используется модель RBAC (Role-Based Access Control) — Управление доступом на основе ролей.

Суть метода:

Права доступа (читать, писать, удалять) назначаются не конкретному человеку (Иванову), а Роли (например, «Декан»).

Пользователю присваивается одна или несколько Ролей.

Проблемы реализации RBAC в вузе:

Динамика ролей. Студенты становятся выпускниками, преподаватели увольняются или меняют должности. Система должна уметь быстро отзывать права.

Иерархия объектов. Декан факультета «А» должен иметь полные права на документы своего факультета, но не должен видеть документы факультета «Б», хотя роль у него такая же — «Декан». Это требует настройки доступа не только к функциям, но и к конкретным данным.

На рисунке 10 показана схема работы ролевой модели.



Рисунок 10 — Схема ролевого управления доступом (RBAC)

Методы защиты информации в разрабатываемой системе

На основе анализа угроз можно сформировать требования к безопасности проектируемой системы:

Аутентификация. Вход только по логину и паролю. Пароли в базе данных не должны храниться в открытом виде — только в виде хеш-сумм (зашифрованных строк, которые нельзя расшифровать обратно).

Журналирование (Логирование). Система должна записывать каждое действие: «Пользователь X открыл документ Y в 14:00». Это позволит расследовать инциденты.

Разделение сессий. Если пользователь забыл выйти из системы на общественном компьютере в библиотеке, сессия должна автоматически закрываться через определенное время.

Резервное копирование (Бэкап). Ежедневное сохранение базы данных на отдельный носитель для защиты от потери данных.

Информационная безопасность — это не дополнительная опция, а фундамент системы. Анализ показал, что главная задача разработчика — защитить данные не столько от внешних хакеров, сколько от несанкционированного доступа внутри коллектива и случайных ошибок.

Применение модели RBAC (ролевого доступа) и современных методов шифрования паролей позволит создать надежную среду для работы с документами. Эти принципы будут реализованы в проектной части работы.

1.6 Методы аутентификации, авторизации и разграничения прав доступа (RBAC)

Для обеспечения информационной безопасности, принципы которой были описаны в предыдущей главе, системе необходимы четкие механизмы контроля входа и действий пользователей. Защита строится на двух последовательных процессах, которые часто путают, но которые имеют принципиальную разницу: аутентификация и авторизация.

Понимание разницы между ними критично для проектирования безопасной системы документооборота.

Аутентификация: Проверка подлинности

Аутентификация (Authentication) — это процесс ответа на вопрос системы: «Кто ты?». Это процедура проверки, действительно ли пользователь является тем, за кого себя выдает.

В контексте университетской СЭД применяются следующие методы:

Однофакторная аутентификация (Парольная защита). Самый распространенный и базовый метод. Пользователь вводит логин (например, студенческий билет или корпоративную почту) и секретный пароль.

Требование к разработке: Пароли ни в коем случае нельзя хранить в базе данных в открытом виде. Необходимо использовать хеширование (например, алгоритмы bcrypt или Argon2) с добавлением «соли» (случайной строки). Это гарантирует, что даже при краже базы данных злоумышленник не увидит реальные пароли пользователей.

Многофакторная аутентификация (2FA / MFA). Используется для защиты учетных записей с высокими привилегиями (Ректор, Главный бухгалтер, Администратор системы). Помимо пароля (то, что пользователь знает), требуется второй фактор — например, одноразовый код из SMS или мобильного приложения (то, что пользователь имеет).

Технология единого входа (Single Sign-On — SSO). В современном вузе уже есть множество систем (библиотечная, учебный портал, Wi-Fi). Заставлять пользователя помнить разные пароли для каждой — плохая практика. Технология SSO позволяет использовать одну учетную запись (обычно на базе протокола LDAP или Active Directory) для входа во все сервисы университета, включая разрабатываемую СЭД.

На рисунке 11 наглядно показана разница между процессами аутентификации и авторизации.



Рисунок 11 — Различие процессов аутентификации и авторизации

Авторизация (Authorization) — это процесс ответа на вопрос: «Что тебе можно делать?». Она происходит строго после успешной аутентификации. Система проверяет, имеет ли конкретный пользователь право выполнить запрошенное действие (например, «Подписать приказ» или «Удалить студента»).

Существует несколько моделей управления доступом. Выбор модели определяет логику работы всей системы.

1. Дискреционная модель (DAC — Discretionary Access Control). В этой модели владелец объекта сам решает, кому дать доступ.

Пример: Как в Google Документах — автор создает файл и сам «расшаривает» его коллегам.

Почему не подходит для ВУЗа: Слишком хаотично. Преподаватель не должен сам решать, кому показывать ведомость с оценками. Это должны определять правила университета.

2. Мандатная модель (MAC — Mandatory Access Control). Используется в военных и государственных структурах с высокой секретностью. Каждому объекту присваивается гриф секретности («Секретно», «Совершенно секретно»), а пользователю — уровень допуска.

Почему не подходит для ВУЗа: Слишком жестко и сложно в администрировании для открытой академической среды.

3. Ролевая модель (RBAC — Role-Based Access Control). Наиболее подходящая модель для корпоративных систем со сложной иерархией, которой и является вуз. О ней мы говорили в предыдущей главе, теперь рассмотрим ее структуру детальнее.

Детальная структура модели RBAC

RBAC устраняет хаос, отвязывая права доступа от конкретных людей. Вместо того чтобы назначать права Иванову, Петрову и Сидорову, администратор создает «роль» (виртуальную должность) и наделяет правами ее.

Основные компоненты модели RBAC для проектируемой системы:

Пользователи (Users): Реальные люди — сотрудники и студенты, имеющие учетную запись.

Роли (Roles): Именованные группы, отражающие должностные обязанности в вузе.

Примеры: «Студент», «Преподаватель», «Заведующий кафедрой», «Декан», «Сотрудник ректората», «Администратор».

Особенность: Один пользователь может иметь несколько ролей одновременно (например, «Преподаватель» и «Член ученого совета»).

Разрешения (Permissions) / Права: Конкретные атомарные действия, которые можно выполнять в системе.

Примеры: document.create (создать документ), document.read.own (читать свои), document.read.faculty (читать документы факультета), document.sign (подписать).

Объекты (Objects): То, к чему применяется действие (Типы документов, Справочники, Отчеты).

Преимущества RBAC для реализации в дипломном проекте:

Упрощение администрирования: Когда в вуз приходит новый декан, ему просто присваивается роль «Декан», и он мгновенно получает сотни необходимых прав. Не нужно настраивать доступ к каждому документу вручную.

Поддержка иерархии ролей: Можно настроить систему так, что роль «Заведующий кафедрой» автоматически включает в себя все права роли «Преподаватель».

Соответствие бизнес-логике: Роли в системе четко соответствуют штатному расписанию университета, что делает систему понятной для руководства.

На рисунке 12 представлена структурная схема взаимодействия компонентов RBAC, которая ляжет в основу проектирования базы данных.

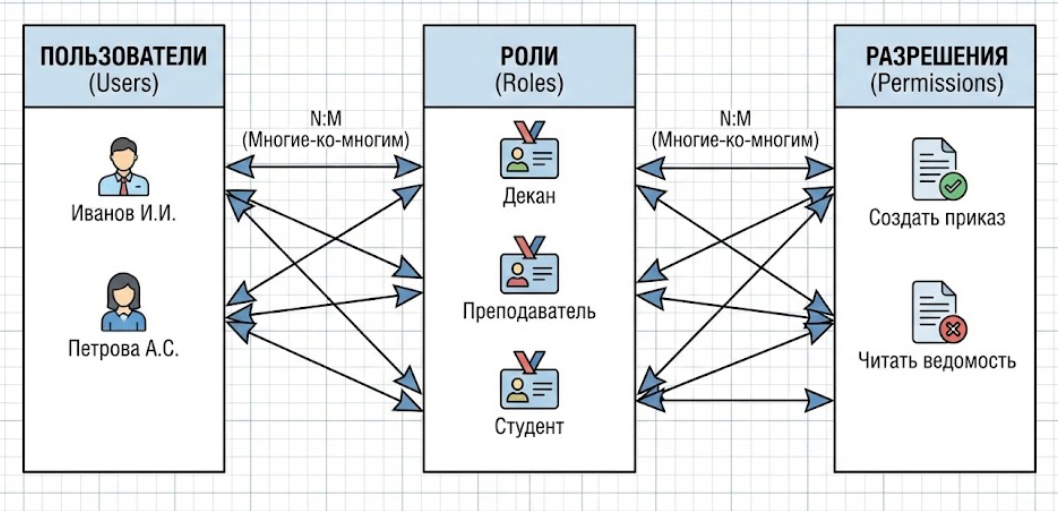


Рисунок 12 — Структурная схема компонентов модели RBAC

Для создания безопасной и управляемой системы электронного документооборота ВУЗа необходимо реализовать два ключевых механизма:

Аутентификацию через безопасное хранение паролей (хеширование) с возможностью интеграции с единой системой входа университета (SSO).

Авторизацию на основе ролевой модели (RBAC), которая позволит гибко управлять правами доступа тысяч пользователей в соответствии с их реальными должностными обязанностями.

Выбранные методы и технологии являются стандартом для современных корпоративных систем и обеспечивают необходимый баланс между безопасностью и удобством работы. Эти решения будут использованы в следующей главе при проектировании архитектуры системы.

1.7 Перспективы цифровизации образования в Республике Казахстан

Разработка информационной системы для вуза не может рассматриваться в отрыве от общей ситуации в стране. Казахстан активно движется по пути отказа от бумажных носителей во всех сферах — от медицины до налогообложения. Образование является одной из ключевых отраслей, где этот переход происходит наиболее заметно.

Государственная стратегия и программа «Цифровой Казахстан»

Основой для всех изменений служит государственная программа «Цифровой Казахстан». В сфере образования она ставит четкие цели: обеспечить равный доступ к знаниям и повысить качество управления учебными заведениями за счет технологий.

Если 10 лет назад цифровизация означала просто закупку компьютеров в классы, то сегодня фокус сместился на интеграцию данных.

Вузы Казахстана больше не являются изолированными островами. Они обязаны передавать данные в государственные системы:

НОБД (Национальная образовательная база данных). Сюда стекается статистика по студентам, педагогам и материальному оснащению.

eGov (Электронное правительство). Выдача дипломов, справок с места учебы и подача документов на поступление теперь происходят через портал eGov.kz.

ЕСУВО (Единая система управления высшим образованием). Платформа для мониторинга качества образования со стороны Министерства науки и высшего образования.

Разрабатываемая в данной диссертации система документооборота в перспективе может стать источником достоверных данных для этих государственных баз, исключая ручной ввод отчетов.

На рисунке 13 показана экосистема цифрового образования в РК.

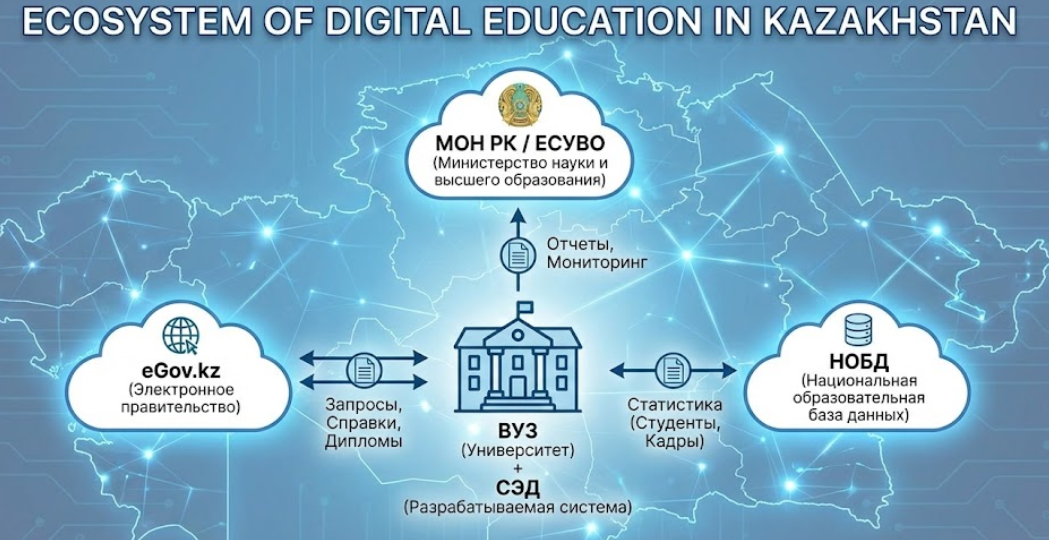


Рисунок 13 — Интеграция вуза в цифровую экосистему Казахстана

Концепция «Smart University» (Умный университет)

Современный тренд в Казахстане — трансформация классических вузов в «Умные университеты». Это не только умные доски или скоростной интернет. Это прежде всего умное управление.

Концепция Smart University стоит на трех «китах»:

Smart Learning (Умное обучение). Использование платформ типа Moodle, Platonus, Canvas для дистанционных лекций и тестов. В этой области казахстанские вузы уже достигли успеха, особенно после пандемии 2020 года.

Smart Campus (Умный кампус). Безопасность, турникеты с распознаванием лиц, умное общежитие, Wi-Fi зоны.

Smart Management (Умное управление). Именно к этой части относится тема данной диссертации. Это автоматизация приказов, служебных записок, бухгалтерии и кадров.

Проблема: В большинстве вузов РК хорошо развит первый пункт (обучение), но третий пункт (управление) часто остается на уровне бумажных журналов. Внедрение СЭД заполняет этот пробел, делая университет по-настоящему «цифровым».

На рисунке 14 показана структура Smart University, где система документооборота занимает центральное место в блоке управления.

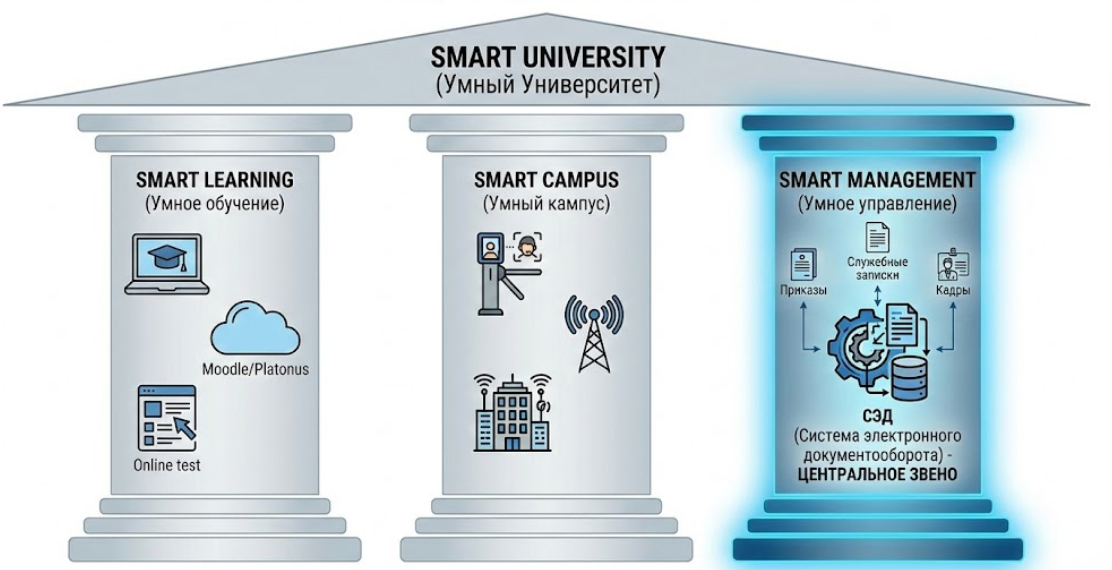


Рисунок 14 — Структура концепции «Smart University»

Тренды и вызовы цифровизации

Анализ ситуации в Казахстане позволяет выделить несколько ключевых трендов, которые будут определять развитие отрасли в ближайшие 5 лет:

Переход к сервисной модели. Вуз начинает работать как сервис. Студент заказывает справку в приложении, преподаватель подает заявление на отпуск с телефона. Это требует быстрых внутренних процессов, которые невозможны без СЭД.

Академическая свобода. Вузы получили больше самостоятельности от Министерства. Но свобода требует ответственности и четкого внутреннего контроля, который обеспечивает электронная система.

Аналитика на основе данных (Data-Driven Decision Making). Ректоры перестают принимать решения интуитивно. Им нужны точные цифры: «Какая кафедра работает хуже всех?», «Где застревают документы?». Бумажный документооборот не дает такой статистики, а электронный — формирует ее автоматически.

Существующие барьеры:

Цифровое неравенство. Не во всех регионах качество интернета позволяет работать с тяжелыми облачными системами. Поэтому разрабатываемая система должна быть легкой и быстрой.

Цифровая грамотность персонала. Педагоги старшего поколения — это золотой фонд вузов, но им бывает трудно осваивать новые программы. Интерфейс системы должен быть максимально простым, чтобы не оттолкнуть пользователей.

Цифровизация образования в Казахстане — это необратимый процесс, поддерживаемый государством. Разработка системы электронного документооборота полностью соответствует национальным интересам и концепции «Слышащего государства», обеспечивая прозрачность и скорость реакции вуза.

Общий вывод по Аналитической части (Глава 1): В ходе аналитического исследования было установлено:

Традиционные бумажные методы управления вузом неэффективны и ведут к потерям времени (п. 1.1).

Существующие на рынке готовые решения либо слишком дороги, либо не учитывают специфику учебного процесса (п. 1.2).

Внутренние потоки документов имеют сложную структуру, требующую оптимизации маршрутов (п. 1.3).

Для реализации системы выбрана надежная трехзвенная архитектура и веб-технологии (п. 1.4).

Вопросы безопасности будут решены через ролевую модель доступа RBAC (п. 1.5, 1.6).

Все это обосновывает целесообразность создания собственной информационной системы, проектирование которой будет описано во второй части работы.