

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное
образовательное учреждение высшего образования
МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
(МОСКОВСКИЙ ПОЛИТЕХ)

ДОПУЩЕН К ЗАЩИТЕ

Заведующий кафедрой

_____ / Пухова Е. А. /

Руководитель образовательной программы

_____ / Даньшина М. В. /

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

по теме:

ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ КОЛЛЕКТИВНОГО ОБМЕНА ЗНАНИЯМИ СТУДЕНТОВ

по направлению 09.03.01 Информатика и вычислительная техника

Образовательная программа (профиль) «Веб-технологии»

Студент: _____ / Петров Александр Игоревич, 221–321 /
подпись *ФИО*

Руководитель ВКР: _____ / Клейменова Лариса Маясовна, к.п.н. /
подпись *ФИО, уч. звание и степень*

Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное

образовательное учреждение высшего образования

МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

(МОСКОВСКИЙ ПОЛИТЕХ)

ЗАДАНИЕ НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ

по направлению 09.03.01 Информатика и вычислительная техника Образовательная
программа (профиль) «Веб-технологии»

Тема ВКР	Веб-приложение для коллективного обмена знаниями студентов.
ПРАКТИЧЕСКИЙ РЕЗУЛЬТАТ	
Назначение	Веб-приложение предназначено для организации и совместного использования учебных материалов студентами в рамках образовательных программ. Система позволяет централизованно хранить, структурировать и находить материалы по различным дисциплинам и вузам. Пользователи могут добавлять собственные материалы, просматривать и сохранять материалы других пользователей, формируя персональную базу знаний. Приложение направлено на упрощение доступа к учебной информации, повышение эффективности самостоятельного обучения и взаимодействия между студентами.
Основные функции	<ol style="list-style-type: none">Осуществление регистрации и авторизации пользователей.Управление профилем пользователя.CRUD материалов.Поиск и фильтрация.Система избранного.
Используемые технологии и платформы	Next.js, TypeScript, Tailwind CSS, Zustand, Node.js, NestJS, REST API, PostgreSQL, Prisma, React Query

ВЫПОЛНЕНИЕ РАБОТЫ	
Решаемые задачи	<ol style="list-style-type: none"> 1. Провести анализ предметной области веб-приложений для обмена знаниями. 2. Выполнить анализ существующих аналогичных решений и определить их преимущества и недостатки. 3. Определить целевую аудиторию и пользовательские роли. 4. Сформировать функциональные требования к системе. 5. Разработать пользовательские сценарии (поиск, добавление, сохранение в избранное, просмотр материала). 6. Спроектировать структуру и навигацию пользовательского интерфейса (прототипы/макеты страниц). 7. Спроектировать архитектуру клиент-серверного взаимодействия и REST API. 8. Спроектировать схему базы данных (пользователи, вузы, дисциплины, материалы, избранное). 9. Реализовать серверную часть веб-приложения на NestJS/Node.js. 10. Реализовать клиентскую часть веб-приложения на Next.js/TypeScript с использованием Zustand и Tailwind CSS. 11. Реализовать функции поиска, фильтрации и категоризации материалов. 12. Реализовать загрузку/хранение файлов материалов и отображение вложений. 13. Провести тестирование веб-приложения и устранить выявленные ошибки.
Состав технической документации	<ol style="list-style-type: none"> 1. Техническое задание. 2. Пояснительная записка.
Состав графической части	<ol style="list-style-type: none"> 1. Презентация: 1 экз. 2. Диаграмма IDEF0 AS-IS (анализ существующего процесса): 1 экз. 3. Диаграмма IDEF0 TO-BE (проектируемая система): 1 экз. 4. ERD-диаграмма базы данных: 1 экз. 5. Схема архитектуры веб-приложения (frontend/backend): 1 экз. 6. Схема взаимодействия компонентов системы (API, клиент, БД): 1 экз. 7. Пользовательские сценарии (Use Case / User Flow): 1 экз. 8. Макеты интерфейса веб-приложения (Figma): 1 экз.

ПЛАН РАБОТЫ НАД ВКР

Этапы	Недели семестра																	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Провести анализ предметной области.																		
Сравнить существующие аналогичные решения.																		
Провести анализ целевой аудитории веб-приложения.																		
Определить функциональные требования к веб-приложению.																		
Разработать пользовательские сценарии.																		
Спроектировать архитектуру веб-приложения.																		
Разработать дизайн-макеты страниц и компонентов веб-приложения.																		
Спроектировать схему базы данных.																		
Разработать серверную часть веб-приложения.																		
Разработать клиентскую часть веб-приложения.																		
Разработать руководство по использованию.																		
Провести различные виды тестирования веб-приложения.																		
Обеспечить информационную безопасность веб-приложения.																		

РУКОВОДИТЕЛЬ ОП:

«___» ____ 2026, _____ / Даньшина Марина Владимировна /
подпись *ФИО, уч. звание и степень*

РУКОВОДИТЕЛЬ ВКР:

«___» ____ 2026, _____ / Клейменова Лариса Маясовна, к.п.н. /
подпись *ФИО, уч. звание и степень*

СТУДЕНТ:

«___» ____ 2026, _____ / Петров Александр Игоревич, 221–321 /
подпись *ФИО, группа*

АННОТАЦИЯ

Наименование работы: веб-приложение для коллективного обмена знаниями студентов.

Цель работы: разработать веб-приложение, предназначенное для централизованного хранения, структурирования и поиска учебных материалов по вузам и дисциплинам, а также для формирования персональной базы знаний пользователей с использованием механизма избранного.

Объект исследования: веб-приложение, обеспечивающее студентам инструменты для публикации и просмотра учебных материалов, их классификации по образовательным организациям и дисциплинам, поиска и фильтрации, а также сохранения материалов в избранное для быстрого доступа.

Предмет исследования: процесс организации коллективного обмена знаниями в студенческой среде и автоматизация управления знаниями в формате веб-платформы.

Работа состоит из Введения, трех глав, Заключения, Списка использованных источников и Приложений. Общий объем работы составляет X страниц, включая Y страниц Приложений. В работе содержится A рисунков, B таблиц и Z листингов кода. Библиография включает N источников.

Во Введении изложены цель, задачи, объект и предмет исследования, актуальность, новизна и практическая значимость работы. Первая глава посвящена анализу предметной области систем обмена знаниями, выявлению основных проблем, обзору существующих аналогов и определению требований к системе. Вторая глава описывает проектирование и реализацию веб-приложения: архитектуру клиент-серверного взаимодействия, разработку REST API, проектирование базы данных и пользовательского интерфейса, а также реализацию ключевых модулей (регистрация и авторизация, профиль пользователя, CRUD материалов, поиск и фильтрация, избранное и просмотр материалов). Третья глава посвящена тестированию системы, вопросам информационной безопасности и авторизации, оценке удобства использования, а также возможностям масштабирования платформы (поддержка нескольких вузов, расширение дисциплин и рост базы материалов). В Заключении представлены выводы по выполненной работе и перспективы дальнейшего развития системы.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	8
1 ПРЕДМЕТНАЯ ОБЛАСТЬ И ТЕХНОЛОГИИ	10
1.1 Анализ предметной и проблемной области	10
1.2 Анализ процесса загрузки учебного материала	12
1.3 Анализ процессов поиска и фильтрации учебных материалов	16
1.4 Обзор аналогов и существующих решений	21

ВВЕДЕНИЕ

Современная образовательная среда активно переходит к цифровым форматам обучения и совместной работы. При этом ключевая проблема для студентов сохраняется: учебные материалы часто распределены по разным источникам (чаты, облачные диски, личные заметки, файлы преподавателей), не имеют единой структуры и быстро теряются. В результате возрастают времена на поиск нужной информации, снижается регулярность самостоятельной подготовки и усложняется обмен знаниями внутри учебной группы.

Традиционно обмен знаниями происходит через неформальные каналы: мессенджеры, социальные сети, общие папки и форумы. Данный подход неудобен из-за отсутствия единого каталога и механизмов классификации, ограниченных возможностей поиска, а также недостаточной прозрачности: сложно понять актуальность файла, к какой дисциплине он относится и насколько он полезен. Отдельной проблемой является отсутствие персонализации — даже если материал найден, его невозможно быстро сохранить «в свою базу» и вернуться к нему позже в несколько кликов.

Актуальность данной работы обусловлена необходимостью разработки веб-приложения, обеспечивающего централизованное хранение, структурирование и быстрый доступ к учебным материалам. Создание системы коллективного обмена знаниями позволит упорядочить материалы по вузам и дисциплинам, повысить скорость поиска и повторного использования информации, а также снизить потери учебных ресурсов при смене учебных групп или формата обучения. Кроме того, наличие механизма избранного позволит пользователям формировать персональную базу знаний и поддерживать систематичность подготовки.

Целью данной работы является разработка веб-приложения для коллективного обмена учебными знаниями студентов с возможностью структурирования, поиска и сохранения материалов в избранное. Для достижения поставленной цели необходимо решить следующие задачи:

1. Провести обзор аналогов систем обмена учебными знаниями и определить их преимущества и недостатки.
2. Выполнить анализ предметной области и выделить основные сценарии использования (поиск, публикация, просмотр, сохранение в избранное).
3. Определить целевую аудиторию веб-приложения и пользовательские роли.
4. Сформировать функциональные и нефункциональные требования к системе.

5. Спроектировать структуру и навигацию пользовательского интерфейса, подготовить макеты ключевых экранов.
6. Спроектировать архитектуру веб-приложения и клиент-серверное взаимодействие.
7. Разработать модель данных и спроектировать схему базы данных.
8. Реализовать серверную часть веб-приложения (REST API, авторизация, работа с знаниями и избранным).
9. Реализовать клиентскую часть веб-приложения (каталог материалов, поиск и фильтрация, карточки, просмотр, профиль, избранное).
10. Провести тестирование разработанной системы и устраниТЬ выявленные недостатки.
11. Описать возможности масштабирования и дальнейшего развития системы (расширение классификаторов, модерация, рекомендации и др.).

Объект исследования — веб-приложение, предназначенное для организации коллективного обмена учебными знаниями.

Предмет исследования — процесс структурирования, хранения и поиска учебных материалов и поддержка совместного использования знаний студентами в цифровой среде.

Таким образом, разработка веб-приложения позволит централизовать учебные материалы, обеспечить удобную классификацию по вузам и дисциплинам, ускорить поиск и доступ к информации, а также повысить эффективность самостоятельной подготовки за счет формирования персональной базы знаний через механизм избранного. Система может быть востребована в вузах и учебных сообществах, где важно быстрое распространение актуальных материалов и поддержка регулярной учебной практики.

1 ПРЕДМЕТНАЯ ОБЛАСТЬ И ТЕХНОЛОГИИ

1.1 Анализ предметной и проблемной области

Предметная область данного исследования охватывает разработку веб-приложений в сфере цифрового образования, ориентированных на организацию, хранение и распространение учебных материалов между студентами. Данное направление относится к образовательным технологиям (EdTech) и связано с цифровизацией учебного процесса, развитием онлайн-форматов и ростом потребности в удобных инструментах доступа к знаниям.

В реальных условиях учебные материалы часто распределены по множеству источников: мессенджеры, социальные сети, облачные хранилища, личные файлы студентов и преподавателей. Такая фрагментация приводит к потере материалов, отсутствию единой структуры и сложностям в поиске. Особенно остро проблема проявляется при подготовке к экзаменам и зачётам, когда требуется быстро восстановить доступ к конспектам, методическим указаниям, лабораторным работам и типовым заданиям по конкретной дисциплине.

Сложившаяся практика обмена знаниями через неформальные каналы имеет ряд ограничений. Во-первых, отсутствует удобная классификация: файлы редко систематизируются по вузам и дисциплинам, а названия и содержимое не стандартизированы. Во-вторых, возможности поиска ограничены: даже при наличии общего облачного диска пользователь вынужден просматривать множество папок и файлов. В-третьих, отсутствует персонализация: студент не может быстро сформировать собственный набор полезных материалов и возвращаться к нему без повторного поиска.

Основной проблемой предметной области является отсутствие единой платформы, обеспечивающей структурированное хранение учебных материалов, быстрый поиск по атрибутам (вуз, дисциплина, тип материала) и удобные механизмы повторного доступа к найденным материалам. При ручной организации материалов (через папки и чаты) неизбежно возникает дублирование, теряется актуальность, снижается качество навигации и возрастает время на подготовку. Это негативно влияет на эффективность самостоятельного обучения и обмена знаниями внутри учебного сообщества.

Современные веб-технологии позволяют решать перечисленные задачи за счёт создания централизованных систем управления контентом. Использование клиент-серверной архитектуры, механизмов авторизации, базы данных и инструментов поиска обеспечивает удобный доступ к материалам, поддержку добавления и редактирования контента, а также формирование персональных подборок через механизм избранного. В результате

снижается “стоимость поиска” информации, а доступ к знаниям становится быстрее и более предсказуемым.

Перспективы развития подобных платформ связаны с масштабированием классификаторов (поддержка нескольких вузов, расширение дисциплин и типов материалов), внедрением модерации контента, системой тегов и рекомендаций, а также анализом популярности материалов. Такие улучшения повышают качество базы знаний и делают обмен знаниями более безопасным и удобным для пользователей.

Таким образом, разработка веб-приложения как системы коллективного обмена учебными знаниями направлена на решение актуальной задачи: обеспечение структурированного хранения знаний, удобного поиска и быстрого повторного доступа к материалам. Это способствует повышению эффективности самостоятельной подготовки, улучшению взаимодействия студентов и упорядочиванию учебной информации в цифровой среде.

1.2 Анализ процесса загрузки учебного материала

Процесс загрузки учебного материала в системе представляет собой последовательность действий пользователя, направленных на добавление нового информационного ресурса в общую базу знаний с последующей классификацией, проверкой корректности данных и обеспечением удобного доступа для других пользователей. В отличие от неформального обмена файлами (через мессенджеры и облачные диски), загрузка в систему предполагает обязательное описание материала метаданными, что позволяет выполнять поиск, фильтрацию и построение структурированного каталога.

В рамках предметной области можно выделить два типовых сценария добавления материалов: (1) загрузка файла (например, PDF, DOCX, презентация) и (2) добавление материала в виде ссылки на внешний ресурс (облачное хранилище, видеолекция, статья). Оба сценария объединяет общая логика: пользователь описывает материал, относит его к вузу и дисциплине, выбирает тип и при необходимости дополняет тегами.

1.2.1 Участники процесса и роли

В процессе загрузки и публикации учебных материалов участвуют следующие роли:

- 1.Студент (основной пользователь) — добавляет материалы, описывает их, публикует, просматривает материалы других пользователей и сохраняет в избранное.
- 2.Модератор/администратор (при наличии роли) — контролирует соблюдение правил публикации, может отклонять материалы, редактировать метаданные и удалять нарушения.
- 3.Система (веб-приложение) — обеспечивает интерфейс загрузки, валидацию данных, сохранение файлов/ссылок, создание записей в базе данных и индексацию для поиска.

1.2.2 Входные данные и выходные результаты процесса

Входными данными процесса загрузки являются:

- 1.Контент материала: файл (PDF/DOCX/PPTX/изображение/архив) или ссылка на внешний ресурс.
- 2.Метаданные: название, вуз, дисциплина, тип материала, описание, теги, дополнительные параметры (например, семестр/курс, преподаватель — если предусмотрено).
- 3.Служебные данные: автор материала (пользователь), дата публикации, состояние материала (черновик/опубликован/на модерации).

Выходными результатами процесса являются:

- 1.Новая запись в каталоге материалов с корректно заполненными метаданными.

2. Доступность материала для поиска и фильтрации по вузу/дисциплине/типу/тегам.

3. Возможность открытия/скачивания материала и добавления в избранное другими пользователями.

1.2.3 Этап 1. Инициация загрузки материала

Процесс начинается с того, что пользователь выбирает действие «Добавить материал».

Данное действие может быть доступно из каталога материалов (кнопка в интерфейсе), из личного профиля или из контекстного меню. На этом этапе система выполняет проверку авторизации: если пользователь не вошёл в аккаунт, он перенаправляется на страницу входа/регистрации. После успешной авторизации пользователь возвращается к форме добавления материала.

На этапе инициации система подготавливает форму ввода, подтягивает справочники (список вузов, перечень дисциплин для выбранного вуза, доступные типы материалов) и отображает подсказки по заполнению полей. Это снижает количество ошибок при вводе и повышает качество каталога.

1.2.4 Этап 2. Заполнение метаданных и классификация

Ключевым отличием системы от обмена файлами через чат является обязательное

1. Ввод названия материала. Название должно быть информативным и отражать содержание (например: «Лекции 1–4 по дискретной математике»).

2. Выбор вуза. Вуз определяет верхний уровень классификации и используется в фильтрации каталога.

3. Выбор дисциплины. Дисциплина привязывает материал к конкретному учебному курсу и обеспечивает тематическую группировку.

4. Выбор типа материала (конспект, лекция, методичка, лабораторная, практикум, шпаргалка и др.). Тип помогает пользователям быстрее отфильтровать нужный формат.

5. Добавление краткого описания (по желанию или обязательно, в зависимости от требований). Описание уточняет, что именно находится в материале, и помогает оценить его полезность без открытия файла.

6. Указание тегов. Теги применяются для дополнительной семантической классификации (например: «теория графов», «комбинаторика», «экзамен», «примеры»).

Для повышения качества данных система должна выполнять валидацию: проверку длины названия, запрет пустых полей, корректность выбранных значений из справочников, а также

нормализацию тегов (например, удаление повторов и лишних пробелов). При ошибках форма отображает сообщения и подсвечивает проблемные поля.

1.2.5 Этап 3. Добавление контента (файл или ссылка)

После классификации пользователь добавляет содержимое материала. Возможны два варианта.

Вариант А — загрузка файла. Пользователь выбирает файл с устройства. Система:

- 1.Проверяет размер файла и допустимые расширения.
- 2.При необходимости вычисляет контрольную сумму (для выявления дублей).
- 3.Выполняет загрузку на сервер или в файловое хранилище.
- 4.Сохраняет метаданные о файле: имя, тип, размер, путь хранения.

Вариант В — добавление ссылки. Пользователь вставляет URL. Система:

- 1.Проверяет корректность формата ссылки.
- 2.При необходимости проверяет доступность ресурса (например, что ссылка не пустая и корректно открывается).
- 3.Сохраняет ссылку как часть записи материала.

В обоих вариантах важно обеспечить единый интерфейс просмотра: пользователь должен понимать, что именно будет открываться (файл или внешний ресурс), и какие действия доступны (скачать, открыть в новой вкладке).

1.2.6 Этап 4. Предпросмотр и проверка перед публикацией

Перед финальной публикацией пользователь должен иметь возможность проверить корректность данных. На этом шаге система отображает сводку: название, выбранный вуз и дисциплину, тип материала, теги, описание и прикреплённый файл/ссылку. Пользователь подтверждает публикацию либо возвращается к редактированию.

Этот этап снижает число ошибок и повышает качество общего каталога, так как исправление на стадии загрузки проще, чем последующее редактирование или удаление материала.

1.2.7 Этап 5. Публикация и сохранение в базе данных

После подтверждения система создаёт запись материала в базе данных. На этом этапе фиксируются:

- 1.Автор (идентификатор пользователя).
- 2.Дата и время публикации.
- 3.Метаданные (вуз, дисциплина, тип, теги, описание).
- 4.Ссылка на файл или URL внешнего ресурса.

5.Служебный статус (например: «опубликован», «на модерации», «скрыт»).

Если в системе предусмотрена модерация, материал может получать статус «на проверке», а доступ для других пользователей может быть ограничен до подтверждения. Если модерации нет — материал становится видимым сразу.

1.2.8 Этап 6. Интеграция в каталог и обеспечение доступа

После публикации материал должен стать доступным пользователям через основные сценарии:

- 1.Поиск по названию и ключевым словам.
- 2.Фильтрация по вузу, дисциплине, типу и тегам.
- 3.Открытие карточки материала с просмотром метаданных.
- 4.Переход к файлу/ссылке и загрузка (если доступно).
- 5.Добавление в избранное для быстрого доступа в будущем.

На данном этапе важным является обеспечение быстрого отклика интерфейса и удобства навигации: карточки материалов, понятные бейджи (вуз/дисциплина/тип), предсказуемые кнопки действий.

1.2.9 Этап 7. Обработка ошибок и исключительных ситуаций

В процессе загрузки возможны ошибки, которые должны быть предусмотрены и обработаны:

- 1.Отказ в доступе (пользователь не авторизован) — перенаправление на вход.
- 2.Ошибки валидации полей (пустые поля, слишком длинные значения) — подсказки в форме.
- 3.Превышение допустимого размера файла или неподдерживаемый формат — сообщение об ограничениях.
- 4.Сбой загрузки файла (обрыв сети, ошибка сервера) — повторная попытка и сохранение введённых данных.
- 5.Дублирование материала — предупреждение и предложение изменить метаданные или подтвердить публикацию.

Наличие детальной обработки ошибок повышает надёжность системы и снижает число неуспешных публикаций.

1.2.10 Итоговое описание процесса

Таким образом, процесс загрузки учебного материала в состоит из последовательных этапов: инициация добавления, заполнение метаданных и классификация, добавление

контента (файл/ссылка), предпросмотр, публикация и интеграция в каталог с последующим обеспечением поиска и доступа. Структурирование через метаданные и единый интерфейс публикации отличают систему от неформального обмена файлами и позволяют сформировать масштабируемую базу знаний, пригодную для регулярного использования и коллективного пополнения.

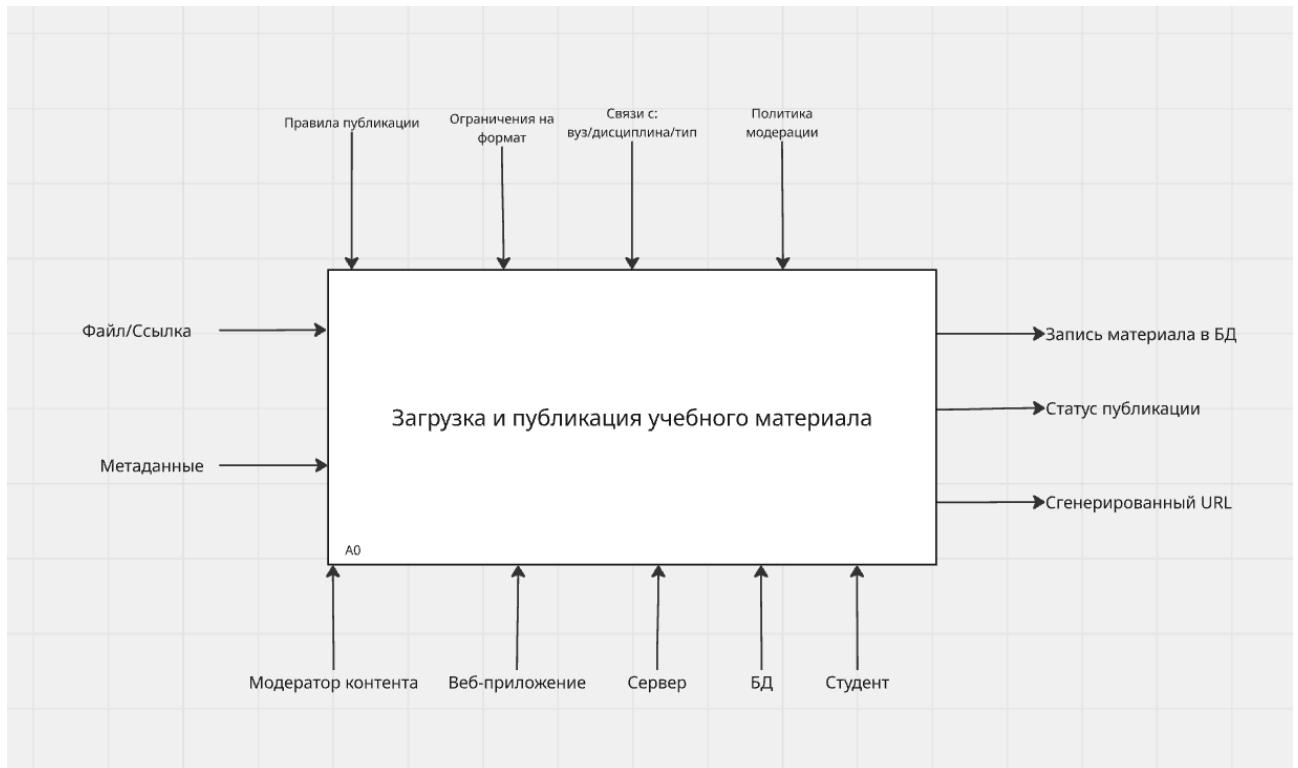


Рисунок 1: Диаграмма загрузки материала IDEF0 (as is)

1.3 Анализ процессов поиска и фильтрации учебных материалов

Процесс поиска и фильтрации учебных материалов в системе направлен на быстрое выявление релевантных ресурсов в общем каталоге на основе текстового запроса и метаданных (вуз, дисциплина, тип, теги и др.). В отличие от неструктурированного поиска по чатам или папкам в облаке, система использует классификацию материалов и единый интерфейс выдачи, что позволяет пользователю уточнять запрос, уменьшать информационный шум и находить нужные материалы за минимальное число действий.

Поиск и фильтрация реализуются как последовательность шагов: ввод запроса и базовых параметров, формирование условий выборки, получение результатов, уточнение через фильтры/сортировку и переход к просмотру конкретного материала. Процесс должен быть стабильным и предсказуемым: изменения фильтров должны приводить к ожидаемому изменению выдачи без потери контекста.

1.3.1 Участники процесса и роли

В процессе поиска и фильтрации участвуют следующие роли:

- 1.Студент (пользователь) — формулирует запрос, выбирает вуз/дисциплину и фильтры, просматривает выдачу, открывает карточки материалов и сохраняет результаты в избранное.
- 2.Система (веб-приложение) — принимает запрос, применяет фильтры, выполняет поиск по индексу/базе данных, формирует ранжированную выдачу, обеспечивает пагинацию и отображение результатов.
- 3.Администратор/модератор (опционально) — определяет справочники и правила видимости (скрытые/на модерации материалы), что влияет на состав выдачи.

1.3.2 Входные данные и выходные результаты процесса

Входными данными процесса являются:

- 1.Поисковый запрос: ключевые слова (например, «дискретка графы», «лаба 2 SQL», «экзамен вопросы»).
- 2.Параметры фильтрации: вуз, дисциплина, тип материала, теги, семестр/курс и др. (в зависимости от модели данных).
- 3.Параметры сортировки: по релевантности, по дате добавления, по популярности (если предусмотрено), по алфавиту.
- 4.Параметры пагинации: номер страницы/лимит, либо «загрузить ещё».
- 5.Контекст пользователя: авторизация, список избранного, ограничения видимости (например, скрытые/на модерации).

Выходными результатами процесса являются:

- 1.Список релевантных материалов (выдача) с краткой информацией: название, вуз, дисциплина, тип, теги, дата публикации и автор (при наличии).
- 2.Возможность уточнения запроса и фильтров без потери контекста (сохранение параметров поиска).
- 3.Переход к карточке материала и выполнение действий: открыть, скачать, перейти по ссылке, добавить в избранное.

1.3.3 Этап 1. Инициация поиска

Поиск может начинаться из нескольких точек интерфейса: главная страница каталога, строка поиска в хедере, раздел конкретного вуза или дисциплины. Пользователь вводит

ключевые слова и/или выбирает базовые параметры (вуз, дисциплина), после чего система формирует начальный запрос на выдачу.

На этапе инициации система также отображает подсказки: автодополнение по вузам/дисциплинам, популярные теги, историю запросов (если предусмотрено). Это снижает нагрузку на пользователя и помогает избежать ошибок ввода.

1.3.4 Этап 2. Формирование условий выборки и валидация параметров

После ввода данных система формирует условия выборки:

- 1.Нормализует текст запроса: приводит к одному регистру, убирает лишние пробелы, может применять стемминг/лемматизацию (при наличии).
- 2.Проверяет корректность фильтров: что выбранный вуз существует в справочнике, дисциплина относится к выбранному вузу, тип материала входит в перечень допустимых.
- 3.Определяет видимость: исключает материалы со статусом «скрыт» или «на модерации» (если это правило применяется для обычных пользователей).
- 4.Собирает итоговый запрос в поисковый модуль/БД с учётом выбранных параметров.

При некорректных параметрах интерфейс должен давать понятную обратную связь (например, сброс дисциплины при смене вуза или сообщение о недоступном фильтре).

1.3.5 Этап 3. Выполнение поиска и получение первичной выдачи

Система выполняет поиск и формирует выдачу. В зависимости от реализации поиск может строиться:

- 1.По метаданным (фильтрация по вузу/дисциплине/типу/тегам) и частичному совпадению названия/описания.
- 2.По индексу полнотекстового поиска (если используется), что позволяет искать по описанию и расширенным полям.
- 3.По комбинации методов: сначала фильтры по метаданным, затем ранжирование по текстовой релевантности.

Результат выдачи должен включать:

- 1.Карточки материалов (краткая карточка): название, дисциплина, тип, теги, дата.
- 2.Общее количество найденных материалов (если вычисляется) и элементы навигации по страницам.
- 3.Состояния интерфейса: «загрузка», «ничего не найдено», «ошибка запроса».

1.3.6 Этап 4. Уточнение через фильтры и сортировку

Пользователь уточняет выдачу, применяя дополнительные фильтры и сортировку. Система должна обеспечивать мгновенную и предсказуемую реакцию:

1. Применение фильтра по вузу/дисциплине ограничивает каталог до заданного учебного контекста.
2. Фильтрация по типу материала позволяет отбирать формат (лекции, методички, лабораторные, шпаргалки и др.).
3. Фильтрация по тегам помогает выделять тематические подмножества (например, «экзамен», «примеры», «теория»).
4. Сортировка по дате позволяет быстро находить актуальные материалы, по релевантности — точнее отвечать на текстовый запрос.

Важно сохранять выбранные параметры при переходах по страницам, чтобы пользователь не терял контекст поиска.

1.3.7 Этап 5. Просмотр результатов и переход к карточке материала

После формирования подходящей выдачи пользователь выбирает конкретный материал и переходит на страницу просмотра. На этом этапе система передаёт контекст:

1. Сохраняет параметры поиска, чтобы кнопка «назад» возвращала к той же выдаче.
2. Отображает в карточке материала метаданные и доступные действия (открыть, скачать, перейти по ссылке).
3. Позволяет добавить материал в избранное непосредственно из выдачи или из карточки.

1.3.8 Этап 6. Пагинация и подгрузка результатов

Если найдено много материалов, система использует пагинацию или «ленивую» подгрузку.

На этом этапе важно:

1. Обеспечить стабильный порядок выдачи при одинаковых параметрах (чтобы результаты не «прыгали»).
2. Сохранять фильтры/сортировку при переходе между страницами.
3. Отображать понятные индикаторы загрузки и ошибки подгрузки.

1.3.9 Этап 7. Обработка ошибок и исключительных ситуаций

В процессе поиска возможны ошибки, которые должны быть предусмотрены:

1. Пустой запрос — выдача по умолчанию (например, популярные/последние материалы) либо требование ввести ключевые слова.

2. Несогласованные фильтры (дисциплина не относится к выбранному вузу) — автоматический сброс/подсказка.
3. Ошибка сети или сервера — сообщение пользователю и возможность повторить запрос.
4. Слишком много результатов — предложение уточнить фильтры и подсказки по популярным тегам.
5. Ничего не найдено — рекомендации: изменить формулировку, убрать часть фильтров, выбрать близкую дисциплину.

1.3.10 Итоговое описание процесса

Таким образом, процесс поиска и фильтрации учебных материалов включает: инициацию запроса, формирование и валидацию условий, выполнение поиска с получением выдачи, уточнение результатов через фильтры и сортировку, переход к карточке материала и навигацию по результатам (пагинация). Использование метаданных (вуз/дисциплина/тип/теги) совместно с текстовым поиском позволяет обеспечить структурированную выдачу и быстрый доступ к релевантным материалам, что является ключевым требованием платформы коллективного обмена учебными ресурсами.

1.4 Обзор аналогов и существующих решений

Современный рынок образовательных технологий активно развивается, расширяя спектр цифровых решений для обучения и самообразования. При этом всё большее значение приобретают сервисы, ориентированные не только на проведение занятий, но и на удобную организацию учебных материалов: их структурирование, быстрый поиск и повторный доступ. Для студентов особенно актуальны инструменты, которые помогают снизить время на “поиск по чатам и папкам” и превращают разрозненные файлы в упорядоченную базу знаний.

Для определения оптимального набора функций и ключевых аспектов, требующих особого внимания при разработке и проектировании веб-приложения, необходимо проанализировать существующие решения, представленные на рынке. В рамках исследования были отобраны наиболее распространённые платформы, используемые студентами для обмена материалами, а также системы, применяемые образовательными организациями для размещения учебного контента. Рассматриваемые решения различаются по назначению и модели использования: от публичных библиотек конспектов и файловых каталогов до корпоративных систем, где материалы размещаются внутри конкретных курсов и доступны ограниченному кругу пользователей.

Анализ аналогов позволяет выявить наиболее удачные практики: каталогизацию материалов по дисциплинам и учебным организациям, развитые механизмы поиска и фильтрации, наличие персональных подборок (избранного), а также возможности управления контентом и контроля качества. Одновременно становятся заметны типовые ограничения: фрагментарность источников, отсутствие единого стандарта описания материалов, сложности с оценкой актуальности и полезности, а также недостаточная персонализация доступа. Таким образом, обзор существующих решений является необходимым этапом для формирования требований к разрабатываемой системе и обоснования проектных решений.

- 1) Notion — универсальная платформа для ведения заметок и построения баз знаний в формате “рабочего пространства”. По сути это конструктор страниц, где можно хранить текст, таблицы, файлы и собирать всё в базы данных. Студенты обычно делают в Notion “рабочую тетрадь” на семестр: страницы по дисциплинам, конспекты лекций, чек-листы, ссылки на материалы, иногда — базы “Дисциплины/Темы/Материалы” с фильтрами.

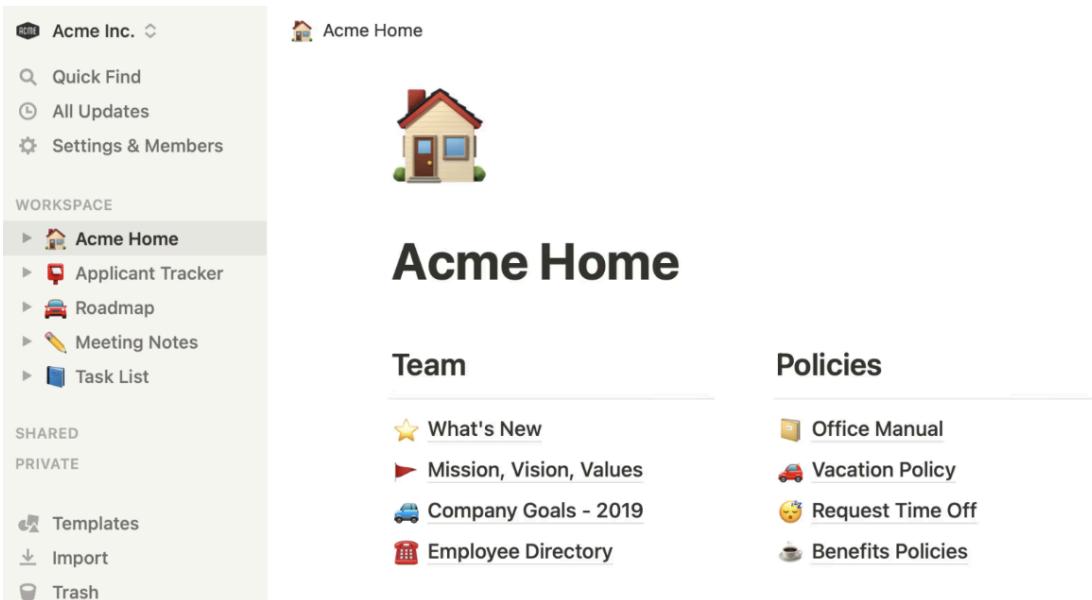


Рисунок 2: Интерфейс приложения Notion

Релевантные функции для предметной области

1. **Структурирование:** древовидные страницы и базы данных (таблица/доска/календарь) с настраиваемыми полями (теги, курс, преподаватель и т.п.).
2. **Поиск:** глобальный поиск по workspace для быстрого нахождения материалов по ключевым словам.
3. **Фильтры и представления:** возможность создавать разные представления (например, “Экзамен”, “Лабы”, “По курсу”) и комбинировать условия фильтрации.
4. **Совместная работа:** общий доступ по ссылке/приглашению, комментарии, упоминания пользователей.
5. **Хранение контента:** вложения, ссылки и встраивания (при этом сервис не является “медиахранилищем” в классическом смысле).
6. **Шаблоны:** стандартизация карточки материала (например: “Название → дисциплина → тип → ссылка → описание”).

Сильные стороны

1. **Гибкая структура и кастомизация:** позволяет быстро собрать “базу материалов” под свои потребности без разработки.
2. **Удобный UX базы знаний:** страницы, теги, фильтры, шаблоны и быстрый поиск упрощают навигацию и поддержку порядка.
3. **Поддержка коллективной работы:** комментарии и механизмы шаринга приближают Notion к сценарию “коллективного обмена”.

Ограничения как платформы материалов для студентов

- 1. Нет доменной модели “вуз → дисциплина → материал” из коробки:** структуру приходится вручную строить через поля и базы.
- 2. Не публичный каталог по умолчанию:** Notion ориентирован на workspace, а не на витрину материалов для широкой аудитории студентов.
- 3. Нет контроля качества и актуальности:** отсутствуют встроенные механизмы модерации, стандарты публикации и единая “карточка материала”, как у специализированных сервисов.
- 4. Слабее сценарии массового поиска:** сервис эффективен внутри одного пространства, но плохо подходит для каталога по множеству вузов/дисциплин.
- 5. Избранное ограничено уровнем страниц/закладок:** это не всегда эквивалентно “избранному материалов” как отдельной функции каталога.

Выводы для требований к системе

- 1. Стоит перенять:** карточную модель материала, теги, фильтры, шаблонность, быстрый поиск и аккуратный UX базы знаний.
 - 2. Важно реализовать отличия:** строгую классификацию (вуз/дисциплина/тип), публичный или полупубличный каталог, “избранное” как отдельный механизм, а при необходимости — модерацию и правила публикации.
- 2) YoNote — онлайн-платформа для ведения учебных конспектов и организации материалов в цифровом виде. Сервис ориентирован на создание заметок, их структурирование по темам и быстрый доступ к информации в процессе обучения. В типичном сценарии студент использует YoNote как “электронную тетрадь”: создаёт конспекты по дисциплинам, закрепляет важные фрагменты, прикрепляет ссылки/файлы, ведёт списки задач и подготовку к зачётам/экзаменам. Материалы обычно группируются по предметам и темам, что упрощает повторение и поиск.

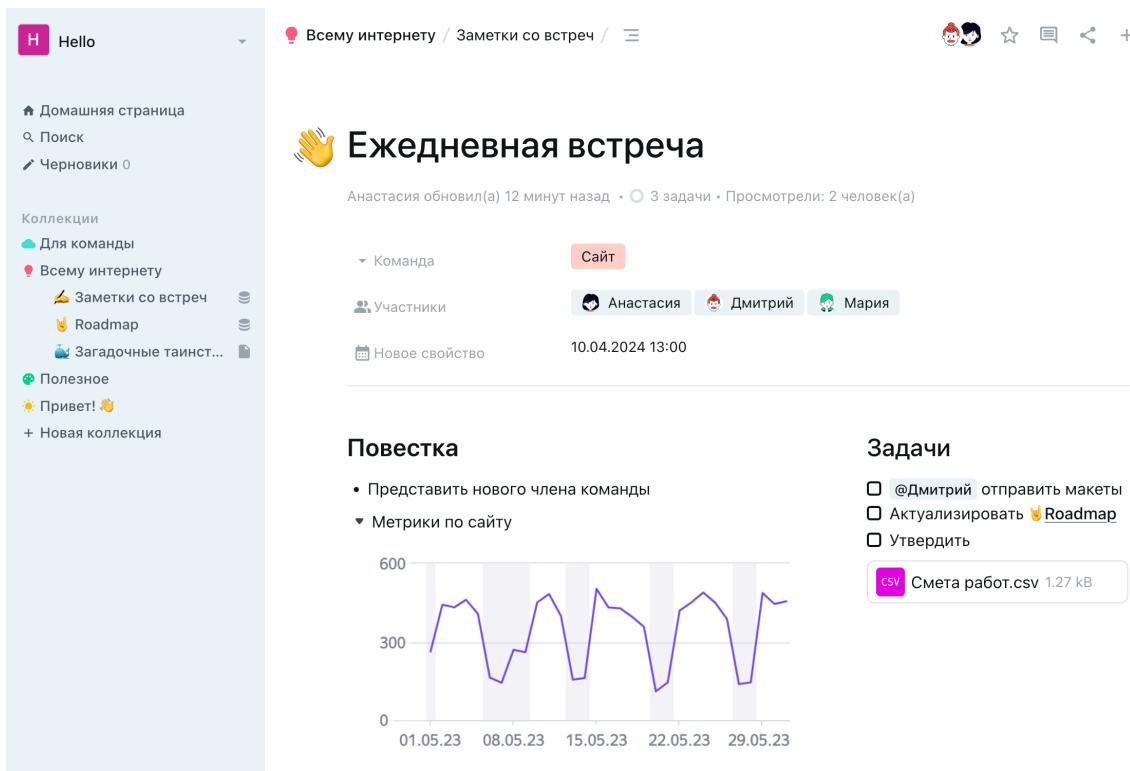


Рисунок 3: Интерфейс платформы YoNote

Релевантные функции для предметной области:

- 1. Создание и ведение заметок:** поддержка форматирования, структурирование конспектов и фиксация ключевых сведений по дисциплинам.
- 2. Организация материалов:** группировка по предметам/темам, использование разделов/категорий для навигации по учебному контенту.
- 3. Поиск:** поиск по заметкам и содержимому для быстрого восстановления нужных фрагментов и тем.
- 4. Работа с вложениями и ссылками:** добавление файлов и внешних ресурсов как дополнения к конспектам.
- 5. Доступ с разных устройств:** использование веб-доступа/синхронизации для работы с материалами в любом месте.
- 6. Обмен материалами:** возможность делиться заметками (по ссылке или через доступ) — как базовый механизм обмена.

Сильные стороны:

- 1. Фокус на учебных сценариях:** сервис подходит для регулярного ведения конспектов и подготовки к занятиям, помогает поддерживать порядок в материалах.
- 2. Быстрый доступ к информации:** структурирование и поиск позволяют оперативно находить нужные темы и фрагменты.

3.Удобство накопления знаний: заметки, вложения и ссылки формируют единый контекст по дисциплине и облегчают повторение.

Ограничения как платформы материалов для студентов:

1.Ориентация на личные заметки, а не на каталог: даже при наличии обмена сервис чаще выступает как персональная “тетрадь”, а не как публичная витрина материалов.

2.Нет строгой доменной модели каталога: обычно отсутствует предопределённая структура вида “вуз → дисциплина → материал” с едиными правилами классификации.

3.Ограниченные сценарии массового поиска: поиск эффективен внутри личного пространства/группы, но сложнее масштабируется на большую аудиторию и множество дисциплин.

4.Не всегда есть стандартизация карточки материала: публикация материалов может быть разноформатной, что осложняет сравнение, оценку качества и повторное использование.

5.Механизмы избранного/коллекций могут быть ограничены: если есть закладки или списки, они не всегда реализованы как отдельная функция “избранного материалов” в каталоге.

Выводы для требований к системе:

1.Стоит перенять: удобное ведение заметок, структурирование по темам, быстрый поиск, работу с вложениями/ссылками и доступ с разных устройств.

2.Важно реализовать отличия: каталогизацию по вузам/дисциплинам и типам материалов, “карточку материала” с метаданными, полноценное “избранное” и масштабируемый поиск по большому массиву публикаций, при необходимости — модерацию/правила публикации.

3) StuDocu — онлайн-платформа для обмена учебными материалами, где студенты загружают и находят конспекты, краткие конспекты, шпаргалки и материалы для подготовки к экзаменам. Доступ к материалам обычно организован через выбор учебного заведения и курса/дисциплины, чтобы находить документы, релевантные конкретной программе обучения. Типичный сценарий использования: студент выбирает свой вуз и курс, ищет нужную дисциплину, просматривает доступные документы и при необходимости загружает собственные материалы. На платформе также

используется модель “вознаграждения” за публикацию (дни Premium-доступа) и есть инструменты подготовки на базе загруженных материалов (например, краткие выжимки и тренировочные вопросы).

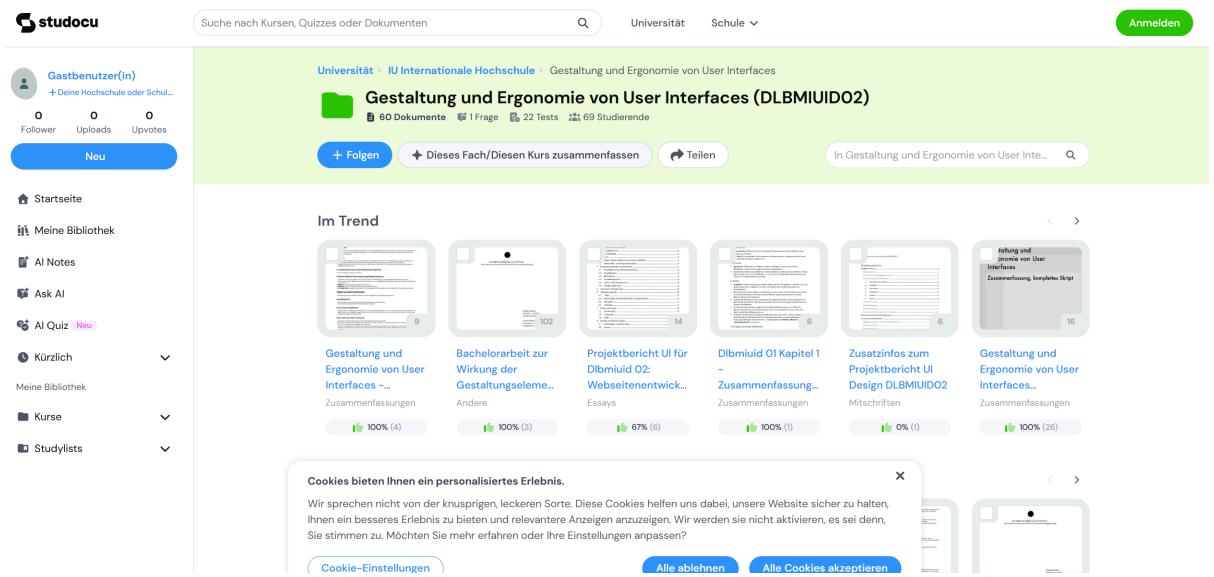


Рисунок 4: Интерфейс платформы StuDocu

Релевантные функции для предметной области:

- 1. Каталог и поиск по учебному контексту:** навигация и поиск материалов через учебное заведение и курс/дисциплину.
- 2. Публикация материалов:** загрузка учебных документов в поддерживаемых форматах и добавление базовых сведений о документе.
- 3. Просмотр и скачивание:** предпросмотр материалов и скачивание (часть функций/контента может быть ограничена).
- 4. Механизм мотивации публикации:** получение Premium-доступа за загрузку и принятие документов.
- 5. Инструменты подготовки:** преобразование загруженных материалов в учебные форматы (например, краткие конспекты, тестовые вопросы).

Сильные стороны:

- 1. Каталог “от студентов для студентов”:** большой объём пользовательских материалов по конкретным дисциплинам и курсам.
- 2. Ориентация на учебный поиск:** удобно искать материалы под конкретный курс/программу, а не “абстрактно по теме”.
- 3. Низкий порог участия:** пользователь может быстро пополнять библиотеку и получать выгоды за публикацию материалов.

4. Усиление подготовки к экзаменам: наличие инструментов, которые помогают “переварить” загруженные документы в более удобный формат.

Ограничения как платформы материалов для студентов:

1. Неоднородное качество контента: материалы пользовательские, поэтому полнота и корректность зависят от автора и источника.

2. Риски авторских прав: возможны загрузки материалов, на которые у пользователей нет прав; требуется модерация и процедура удаления по жалобам.

3. Ограничения доступа: часть популярного контента и/или расширенные функции могут быть доступны только через Premium.

4. Слабая стандартизация “карточки материала”: метаданные часто минимальны, из-за чего сложнее сравнивать материалы и контролировать качество.

Выводы для требований к системе:

1. Стоит перенять: идею каталога по учебному контексту (вуз/дисциплина), сильный поиск и фильтрацию, удобный сценарий публикации и мотивацию участников.

2. Важно реализовать отличия: строгую доменную модель “вуз → дисциплина → материал” с обязательными метаданными, прозрачные правила публикации и лицензирования, модерацию/жалобы, а также “избранное” как отдельный механизм внутри каталога.

- 4) StudFiles — файловый архив учебных материалов, в котором контент публикуется пользователями и группируется по учебному контексту: вузам и дисциплинам. Платформа ориентирована на хранение и распространение документов (конспекты, методички, лабораторные, курсовые и т.п.) с возможностью быстрого просмотра и скачивания. Типичный сценарий использования: студент выбирает свой вуз и предмет, просматривает список доступных файлов, ищет нужный документ по названию и при необходимости загружает собственные материалы в соответствующий раздел. Для ускорения публикации предусмотрена пакетная загрузка (например, архивом с последующей распаковкой).

Фильтр

В каталоге: 1316 вузов, 5364 предметов, 8 864 702 файлов на 6937.55 ГБ

Gold Coast, Australia	Костанай	Петрозаводск
Bond Uni Bond University 1 136	КГУ Костанайский Государственный университет им. Ахмета Байтурсынова 444	КГПА Карельская государственная педагогическая академия 590
Nitra	Кострома	ПГК Петрозаводская государственная консерватория им. А.К. Глазунова 90
UKF Univerzita Konstantina Filozofa 111	КГУ Костромской государственный университет 4 354	ПетрГУ Петрозаводский Государственный Университет 5 110
Riga	Краматорск	Петропавловск
NMC Novikontas Maritime College 4 283	ДГМА Донбасская государственная машиностроительная академия 2 604	СКГУ Северо-Казахстанский государственный университет им. М. Козыбекова 936
Termiz	ДонНАСА Донбасская национальная академия строительства и архитектуры 2 426	Петропавловск-Камчатский
TISU Termiz Igisodiyot ya Servis universiteti 13	Красноармейск	КамчатГТУ Камчатский государственный технический университет 124
Абакан	КИИ Красноармейский индустриальный институт ДонНТУ 207	Печора
XГУ Хакасский государственный университет им. Н.Ф.Катанова 2 775	ДонНТУ	ППЗТ Печорский промышленно-экономический техникум 194
XТИ Хакасский технический институт (филиал СФУ) 324	Краснодар	Пинск
Актау	КГУКИ Краснодарский государственный университет культуры и искусств 699	ПГПТКМ Пинский государственный профессионально-технический колледж машиностроения 98
KGTU Каспийский государственный университет технологий и инноваций им. Есенева 882	ККБМК Краснодарский Краевой Базовый Медицинский Колледж 3	ПолесГУ Полесский государственный университет 2 540
Актобе	КубГАУ Кубанский государственный аграрный университет 2 299	Полтава
ARGU Актюбинский региональный государственный университет им. К. Жубанова 968	КубГМУ Кубанский государственный медицинский университет 4 597	ПГАА Полтавская государственная аграрная академия 1 232
ЭКГМУ Западно-Казахстанский государственный медицинский университет им. М. Осанова 631	КубГТУ Кубанский государственный технологический университет 10 789	ПНПУ Полтавский национальный педагогический университет им. В. Г. Короленко 2 577
Алматы	КубГУ Кубанский Государственный Университет 10 355	ПолтНТУ Полтавский национальный технический университет им. Ю. Федьковича 3 269
ALMU MAB Almaty Management University 912	КГУФКСТ Кубанский государственный университет физической культуры, спорта и туризма 429	ПУЭТ Полтавский университет экономики и торговли 1 550
АГКЭУ Алматинский государственный колледж энергетики и электронных технологий 3 053	КСЭИ Кубанский социально-экономический институт 85	УМСА Украинская медицинская стоматологическая академия 1 510

Рисунок 5: Интерфейс платформы StudFiles

Релевантные функции для предметной области:

- 1.Каталог по вузам и дисциплинам:** навигация по структуре «вуз → предмет → файлы», что упрощает поиск материалов для конкретной учебной программы.
- 2.Публикация материалов:** добавление файлов в выбранный предмет, включая пакетную загрузку (нескольких файлов за один раз).
- 3.Поиск и фильтрация:** поиск по названию файла и базовые фильтры в каталоге для ускорения подбора материалов.
- 4.Просмотр и скачивание:** просмотр страницы материала и скачивание файла для дальнейшего использования.

Сильные стороны:

- 1.Ориентация на учебный контекст:** структура каталога через вузы и предметы близка к сценариям студентов и их учебным планам.
- 2.Большой объём пользовательских материалов:** наличие множества файлов по разным дисциплинам повышает шанс быстро найти нужный документ.
- 3.Простой вход в публикацию:** загрузка материалов не требует сложного оформления и позволяет быстро пополнять архив.

Ограничения как платформы материалов для студентов:

- 1.Неоднородное качество и актуальность:** контент пользовательский, поэтому возможны ошибки, дубли и устаревшие версии.

2. Слабая стандартизация описания материалов: метаданные часто ограничены, из-за чего сложнее оценивать релевантность до скачивания и сравнивать материалы между собой.

3. Полнотекстовый поиск может быть ограничен: быстрый поиск часто сводится к названию файла, а поиск по содержимому зависит от индексации.

4. Нет явной доменной модели “карточки материала”: отсутствует единый формат публикации (тип, авторство, проверка, источники), что усложняет контроль качества.

Выводы для требований к системе:

1. Стоит перенять: каталогизацию по учебному контексту (вуз/дисциплина), простой сценарий публикации, предпросмотр и доступность материалов.

2. Важно реализовать отличия: “карточку материала” с обязательными метаданными (вуз/дисциплина/тип/автор/год/описание), встроенный поиск и фильтры по метаданным, “избранное” как отдельный механизм, а также правила публикации и модерацию/жалобы при необходимости.

Таблица 1 – Сравнение аналогов

Критерий	Notion	Yonote	StuDocu	StudFiles
Тип решения	Конструктор workspace: страницы + базы данных (БЗ/документация).	Командная база знаний: документы + базы данных/ представления.	Публичный каталог учебных материалов (UGC) + инструменты подготовки.	Файловый архив студентов (UGC) с каталогом по вузам/ предметам.
Каталогизация (вуз/ дисциплина)	Нет «вуз— дисциплина» из коробки: задаётся вручную (страницы/ поля/ БД).	Нет «вуз— дисциплина» из коробки: задаётся вручную (документы/БД).	Каталог/ поиск по университету и курсу/ дисциплине.	Каталог «вузы → предметы → файлы».
Публикация материалов	Создание страниц/ БД внутри workspace; публикация вовне — через доступ/ссылку.	Создание документов/ БД внутри пространства; обмен — через доступ/ссылку.	Загрузка документов пользователями в каталог; доступ может быть частично ограничен.	Загрузка файлов в выбранный предмет; возможна пакетная загрузка (архивом).
Поиск и фильтры	Глобальный поиск по workspace + фильтры/ представления в БД.	Поиск по базе знаний + представления БД (таблица/ борд/календарь).	Поиск по каталогу и дисциплинам; ориентация на нахождение готовых документов.	Поиск/ навигация в каталоге; фильтрация и выбор по вузу/ предмету.
Публичность	По умолчанию закрыто: доступ по приглашению/ ссылке и правам.	По умолчанию закрыто: доступ по правам/ссылке (командное пространство).	Публичная библиотека материалов (контент ищется и просматривается на платформе).	Публичный архив: материалы доступны через каталог сайта.
Совместная работа	Совместное редактирование, комментарии, права доступа (в рамках workspace).	Совместное редактирование и комментарии (в рамках пространства).	Коллективность через публикацию/ потребление, а не через редактирование одного документа.	Коллективность через публикацию/ потребление; совместного редактирования нет.

Продолжение Таблицы 1

Критерий	Notion	Yonote	StuDocu	StudFiles
Избранное	Закладки/ избранные страницы (не «избранное материалов» как каталог).	Закладки/ быстрый доступ к документам/ страницам (логика work- space).	Сохранение/ закладки материалов как часть работы с библиотекой.	Закладки/ избранное на уровне материалов (если используется), без “кatalogной” логики.
Контроль качества / модерация	Контроль через процессы и права команды; нет модерации публичного UGC-каталога.	Контроль через права и регламенты команды; нет витрины UGC для всех.	Проверка/ принятие материалов и правила публикации; качество зависит от автора.	Качество зависит от пользователей; возможны дубли и устаревшие версии.
Мотивация публикации	Нет встроенной “награды”; мотивация — удобство команды/личная организация.	Нет встроенной “награды”; мотивация — работа внутри команды.	Награды (Pre- mium-дни) за принятые загрузки.	Обычно без явной награды; мотивация — обмен и доступ к архиву.

