

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра информационных систем

КУРСОВАЯ РАБОТА
по дисциплине «Архитектура информационных систем»
Тема: Проектирование системы дистанционного образования
«АЗБУКА»

Студенты гр. 0361

Преподаватель

Бушуев Д.И.
Геворкян А.А.
Иконников С.Н.
Солонухина А.Л.
Чертков А.Ю.
Водяхо А.И.

Санкт-Петербург
2023

ЗАДАНИЕ

НА КУРСОВУЮ РАБОТУ

Студенты: Бушуев Д.И., Геворкян А.А., Иконников С.Н., Солонухина А.Л., Чертков А.Ю.

Группа 0361.

Тема работы: Проектирование системы дистанционного образования «АЗБУКА».

Исходные данные: спроектировать систему с использованием ПО Enterprise Architect или иных специализированных средств, описать проект, сформировать требования и список тестов, представить набор диаграмм.

Содержание пояснительной записки: введение, требования, архитектурное описание, тесты, заключение, список используемых источников.

Предполагаемый объем пояснительной записки: не менее 42 страниц.

Дата выдачи задания: 16.02.2023.

Дата сдачи отчёта: 13.04.2023.

Студенты гр. 0361

Преподаватель

Бушуев Д.И.
Геворкян А.А.
Иконников С.Н.
Солонухина А.Л.
Чертков А.Ю.

Водяхо А.И.

АННОТАЦИЯ

Содержание курсовой работы заключается в проектировании системы дистанционного образования «Азбука», соответствующей разработанным требованиям (бизнес-требованиям, пользовательским, функциональным, системным и интерфейсным требованиям).

SUMMARY

The content of the course work is to design a distance education system «Azбуka» that meets the developed requirements such as business, user, functional, system and interface requirements.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
1. ТРЕБОВАНИЯ.....	6
1.1. Глоссарий	6
1.2. Бизнес-требования.....	7
1.3. Пользовательские требования.....	7
1.4. Функциональные требования.....	9
1.5. Требования к внешнему интерфейсу.....	11
1.6. Системные требования	12
2. АРХИТЕКТУРНОЕ ОПИСАНИЕ	13
2.1. Архитектурное обоснование	13
2.2. Модели.....	13
2.3. UML описание	17
2.3.1. Use case.....	17
2.3.2. Диаграмма классов.....	28
2.3.3. Диаграмма последовательностей	29
2.3.4. Диаграмма кооперации.....	31
2.3.5. Диаграмма состояний	32
2.3.6. Диаграмма активности/деятельности	34
2.3.7. Диаграмма развёртывания	35
3. ТЕСТЫ.....	37
3.1. Тестирование функциональности.....	37
3.2. Тестирование безопасности.....	38
3.3. Тестирование производительности	39
3.4. Тестирование интерфейса.....	39
ЗАКЛЮЧЕНИЕ	41
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ	42

ВВЕДЕНИЕ

Проект призван решить проблему отсутствия у некоторых учебных заведений собственных систем дистанционного образования, создать независимый от международных ограничений продукт.

Разрабатываемая система позволит как облегчить очный учебный процесс, так и организовать дистанционный.

Целью проекта является разработка доступной отечественной системы дистанционного образования, простой для внедрения и эксплуатации.

Система должна быть ориентирована на русскоговорящего пользователя (иметь русскоязычный интерфейс), основными потребителями будут учебные заведения, включающие в себя как преподавателей, так и учеников/студентов.

1. ТРЕБОВАНИЯ

1.1. Глоссарий

Глоссарий терминов, использованных при составлении требований приведён в таблице 1.

Таблица 1 – Глоссарий

№	Термины	Определения
1	СДО	Система дистанционного обучения
2	Система	Набор, состоящий из баз данных, клиентской и серверной частей и связей между ними
3	Интерфейс	Внешний вид клиентской части системы, включающий необходимые для работы элементы управления
4	Веб-страница	Одна форма отображения элементов системы в окне браузера
5	Учётные данные	Уникальные данные пользователя, используемые для авторизации в системе
6	Роль пользователя «Преподаватель»	Категория пользователей с определенным набором прав, необходимым для доступа учителя к системе и работе в ней
7	Роль пользователя «Студент»	Категория пользователей с определенным набором прав, необходимым для доступа ученика к системе и работе в ней
8	Учебный курс	Объект системы, содержащий в себе учебные материалы, распределённые по темам. Составляется и редактируется пользователями с ролью «Преподаватель». Пользователи с ролью «Студент»

		могут получить доступ для просмотра содержимого курса
9	Учебные материалы	Текст или файлы, располагаемые на страницах курсов

1.2. Бизнес-требования

Бизнес-требования к проектируемой СДО представлены в таблице 2.

Таблица 2 – Бизнес-требования

ID	Бизнес-требование
BRQ-1	Функционал системы должен быть универсальным (общим) для всех ступеней образования (школьное, среднее, высшее, повышение квалификации).
BRQ-2	На каждой веб-странице приложения должны отображаться логотип и название автора системы.
BRQ-3	На случай возникновения технических проблем клиенту должны быть предоставлены контактные данные (электронная почта), куда можно направить жалобу.
BRQ-4	Система должна гарантировать уникальность учетных данных аккаунтов.
BRQ-5	Система должна гарантировать ограничение доступа пользователей к учебным материалам, размещённым в рамках её курса.

1.3. Пользовательские требования

Пользовательские требования к проектируемой СДО представлены в таблице 3.

Таблица 3 – Пользовательские требования

ID	Пользовательское требование
URQ-1	Пользователь должен иметь возможность входить в систему используя свои учетные данные: эл. почту и пароль.
URQ-2	Пользователь должен иметь возможность регистрироваться в системе с выбором роли «Преподаватель» или «Студент».
URQ-3	Пользователь с ролью «Преподаватель» должен иметь возможность создавать курсы.
URQ-4	Пользователь с ролью «Преподаватель» должен иметь возможность редактировать курсы.
URQ-5	Пользователь с ролью «Преподаватель» должен иметь возможность добавлять/удалять темы в его курс.
URQ-6	Пользователь с ролью «Преподаватель» должен иметь возможность разрешать/запрещать отправку работ к темам его курса.
URQ-7	Пользователь с ролью «Преподаватель» должен иметь возможность изменять содержимое тем его курса.
URQ-8	Пользователь с ролью «Преподаватель» должен иметь возможность добавлять файлы к содержимому тем его курса.
URQ-9	Пользователь с ролью «Преподаватель» должен иметь возможность редактировать название его курса.
URQ-10	Пользователь с ролью «Преподаватель» должен иметь возможность выбирать, какой из созданных им курсов открыть.
URQ-11	Пользователь с ролью «Преподаватель» должен иметь возможность оценивать отправленные студентами работы.
URQ-12	Пользователь с ролью «Преподаватель» должен иметь возможность просматривать успеваемость студентов его курса.

URQ-13	Пользователь с ролью «Студент» должен иметь возможность присоединяться к курсу по ссылке и ключу доступа.
URQ-14	Пользователь с ролью «Студент» должен иметь возможность выбирать курс из списка курсов, где он является участником.
URQ-15	Пользователь с ролью «Студент» должен иметь возможность просматривать материалы, размещённые на курсе.
URQ-16	Пользователь с ролью «Студент» должен иметь возможность отправлять работу в предназначенные для этого темы.
URQ-17	Пользователь с ролью «Студент» должен иметь возможность оставлять комментарии к курсу.
URQ-18	Администратор должен иметь возможность редактировать таблицу с пользовательскими аккаунтами.
URQ-19	Администратор должен иметь возможность редактировать списки участников курсов.

1.4. Функциональные требования

Функциональные требования к проектируемой СДО с указанием идентификаторов пользовательских требований, на которые они ссылаются, представлены в таблице 4.

Таблица 4 – Функциональные требования

URQ-ID	ID	Функциональные требования
URQ-1	FRQ-1	Система должна позволять пользователю входить в систему, используя свою эл. почту и пароль.

URQ-2	FRQ-2	Система должна позволять пользователю регистрироваться в системе, при этом выбирая роль «Преподаватель» или «Студент».
URQ-3	FRQ-3	Система должна создавать курс с названием, указанным пользователем, при этом не допускать создание курса с пустым названием.
URQ-4	FRQ-4	Система должна предоставлять расширенный функционал в режиме редактирования курса: для добавления/удаления тем,, разрешения/запрета отправки работ и изменения содержимого тем.
URQ-5	FRQ-5	Система должна предоставлять функционал для добавления и удаления тем в режиме редактирования курса.
URQ-6	FRQ-6	Система должна добавлять форму для отправки файла у обучающихся курса в теме, требующей отправки работы.
URQ-7	FRQ-7	Система должна предоставлять функционал для размещения материалов в темах курса при редактировании его создателем и обновлять содержание тем курса у всех его участников после сохранения изменений.
URQ-8	FRQ-8	Система должна предоставлять функционал для размещения файлов в темах курса, при этом проверять их формат.
URQ-9	FRQ-9	Система должна предоставлять поле для изменения названия курса (в режиме редактирования) и устанавливать его при сохранении преподавателем изменений.
URQ-10	FRQ-10	Система должна предоставлять функционал личного кабинета – выбор преподавателем созданных ранее курсов.
URQ-11	FRQ-11	Система должна предоставлять функционал оценивания работ в ячейках сводной таблицы напротив сданных работ.

URQ-12	FRQ-12	Система должна изменять статус оценённых работ в таблице успеваемости курса на выставленную оценку.
URQ-13	FRQ-13	Система должна запрашивать ключ доступа к курсу у пользователя, не являющегося его участником и обновлять список участников, когда к курсу присоединяется новый пользователь.
URQ-14	FRQ-14	Система должна предоставлять функционал личного кабинета – выбор студентом курса из тех, участником которых он является.
URQ-15	FRQ-15	Система должна предоставлять студенту возможность просматривать материал, размещённый на страницах курсов, участниками которых он является.
URQ-16	FRQ-16	Система должна предоставлять функционал для отправки студентами работ.
URQ-17	FRQ-17	Система должна сохранять комментарии, отправленные пользователем, на странице курса.
URQ-18	FRQ-18	Система должна хранить данные о пользовательских аккаунтах в таблицах, и, при обновлении таблиц, удалять отсутствующих пользователей из списков участников курсов и удалять курсы, созданные отсутствующими пользователями.
URQ-19	FRQ-19	Система должна хранить данные о списках участников курса в таблицах, и, при обновлении таблиц, отображать новые списки участников на страницах курсов.

1.5. Требования к внешнему интерфейсу

Требования к внешнему интерфейсу проектируемой СДО представлены в таблице 5.

Таблица 5 – Требования к внешнему интерфейсу

ID	Требования к внешнему интерфейсу
FERQ-1	Каждая страница системы должна содержать в шапке утвержденный командой проекта логотип.
FERQ-2	Текст всех внешних элементов системы (кнопки, предупреждения, подсказки и т.п.) должен быть написан на русском языке.
FERQ-3	Внешние элементы системы должны быть масштабированы и расположены с расчетом на просмотр в развёрнутом окне браузера на ПК.

1.6. Системные требования

В качестве сервера требуется компьютер с характеристиками не ниже:

1. Процессор – Intel Xeon E5-2650;
2. ОЗУ – 8 Гб;
3. Свободное дисковое пространство – 50 Гб.

Для использования системы требуются:

4. Операционная система – GNU\Linux с версией ядра 4.19 или выше;
5. Пакеты – docker и docker-compose.

2. АРХИТЕКТУРНОЕ ОПИСАНИЕ

2.1. Архитектурное обоснование

СДО написана на NodeJS, с использованием фреймворка Express, для создания пользовательского интерфейса взята библиотека React, в качестве базы данных – MongoDB.

Технологический стек выбран из-за своей простоты, современности и популярности, а также наличия большого количества готовых решений и библиотек, что значительно облегчает разработку системы.

Альтернативные подходы к реализации:

Вместо NodeJS и Express мог быть выбран Python с фреймворками Flask или Django, однако команда разработки имела больший опыт работы именно с выбранным технологическим стеком. Решение остановиться на нем снижает количество потенциальных точек отказа и ускоряет разработку.

По аналогичным причинам был выбран фреймворк Express, вместо которого могли быть выбраны фреймворки Angular или Vue.

На смену MongoDB могла прийти более производительная PostgreSQL, однако работать с ней несколько сложнее, а важной частью разрабатываемой системы является ее простота во внедрении и освоении.

2.2. Модели

Описание классов объектов, их методов и свойств приведено в таблице 6.

Таблица 6 – Описание объектов

Объект	Методы	Свойства
CourseController	createCourse – создание курса	HOST – имя/ip хоста, где запущен сервер
	listCourses – возврат списка курсов	courseModel – объект таблицы MongoDB с курсами
	deleteCourse – удаление курса	userModel – объект таблицы MongoDB с информацией о пользователях
	editCourse – редактирование курса	
	changeTitle – изменение названия курса	
	getCourse – возврат информации о содержимом курса	
	joinCourse – добавление пользователя к курсу	
	deleteCourseMember – удаление пользователя с курса	
	addComment – добавление комментария к курсу	
	deleteComment – удаление комментария к курсу	
	getThemeFile – возврат файла, прикрепленного к курсу	

	uploadFile – загрузка файла на курс	
UserController	registration – регистрация пользователя в системе	tokenService – объект TokenService, использующийся для создания и проверки токена
	login – авторизация пользователя в системе	userModel – объект таблицы MongoDB с информацией о пользователях
TokenService	createToken – создание токена пользователя	JWT_KEY_ACCESS – ключ для подписи токена
	validateAccessToken – проверка подписи и корректности токена	
Server	run – запуск сервера	app – объект Express, реализующий API
	validateAccessToken – проверка подписи и корректности токена	tokenService – объект класса TokenService, использующийся для валидации токена пользователя
		courseController – объект класса CourseController, использующийся для управления курсами
		homeworkController – объект класса HomeworkController, использующийся для управления домашними работами пользователей

		userController – объект класса UserController, использующийся для аутентификации и авторизации пользователей
ValidationController	registerValidation – валидация ввода пользователя при регистрации	validationBody – объект класса express-validation, предоставляющий функционал для валидации
	loginValidation – валидация ввода пользователя при входе в систему	
	fileValidation – валидация запроса пользователя на скачивание файла	
HomeworkController	setMark – выставление оценки	HOST — имя/ip хоста, где запущен сервер
	uploadHomework – загрузка файла домашней работы	courseModel — объект таблицы MongoDB с курсами
	getHomework – возврат файла домашней работы	userModel — объект таблицы MongoDB с информацией о пользователях
	SetCommentToWork – оставление комментария к домашней работе	
	getResultTable – возврат списка с информацией о сданных работах и оценках	

2.3. UML описание

2.3.1. Use case

В СДО есть четыре класса пользователей, которые представлены в таблице 7.

Таблица 7 – Классы пользователей СДО

Класс пользователей	Описание класса
Администратор	Может вручную создавать/удалять пользовательские аккаунты, редактировать списки участников курсов. Занимается организацией и поддержанием работоспособности системы.
Неавторизированный пользователь	Может зарегистрировать новый аккаунт, выбрать роль «Студент» либо «Преподаватель», или войти в уже существующий.
Преподаватель	Может создавать, удалять и редактировать курсы, оценивать работы, просматривать успеваемость и оставлять комментарии для их участников.
Обучающийся	Может присоединяться к курсам, загружать файлы с выполненными работами на специальных разделах, оставлять комментарии к курсам.

Диаграмма вариантов использования (use case) представлена на рисунке 1.

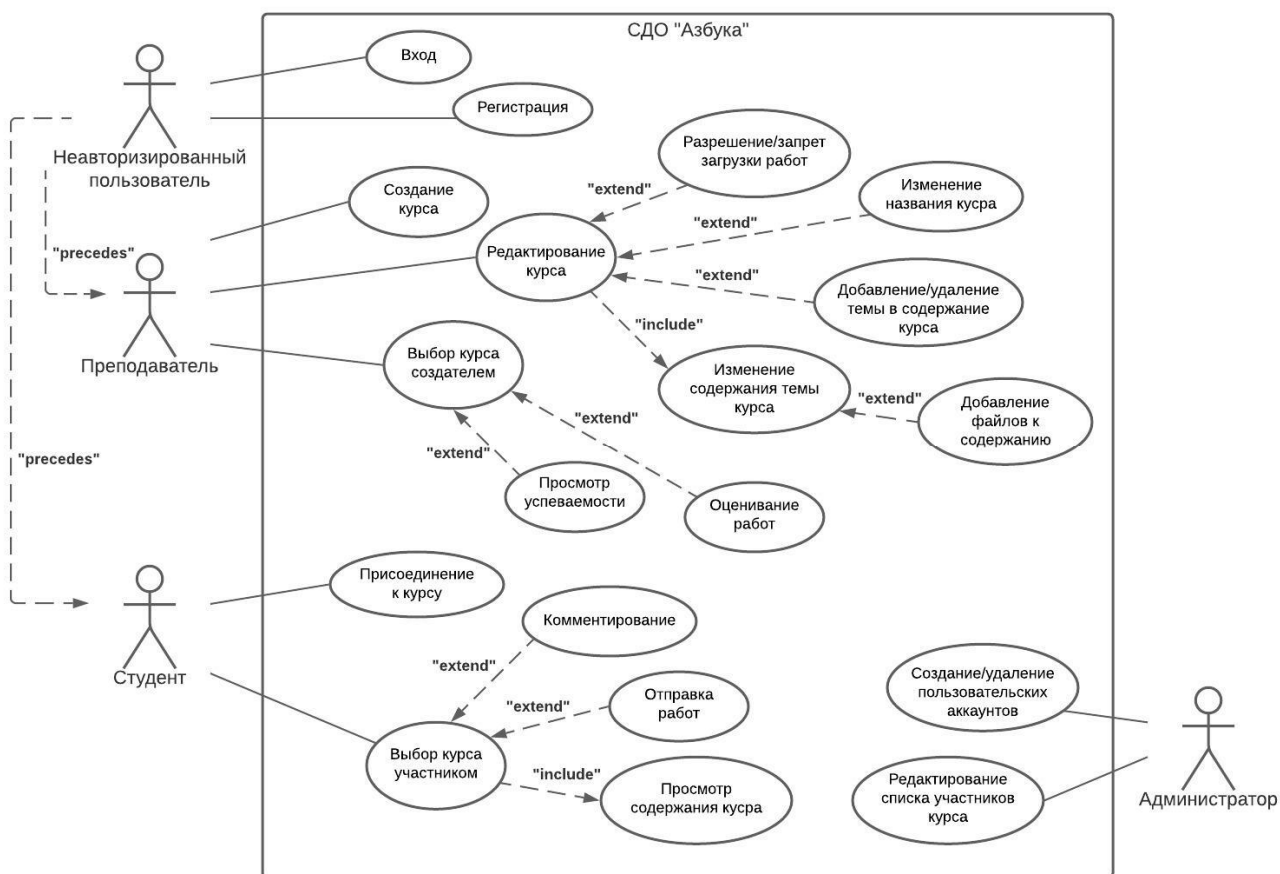


Рисунок 1 – Use case диаграмма

Описание вариантов использования, представленных на use case диаграмме (рисунок 1), содержится в таблице 8.

Таблица 8 – Варианты использования

UC-1	Регистрация
Акторы	Неавторизированный пользователь
Цель	Регистрация нового пользователя в системе для последующей возможности входа
Краткое описание	Пользователь регистрирует аккаунт «Преподаватель» либо «Студент» на незарегистрированную ранее почту, чтобы получить доступ к функционалу системы
Ссылки на другие варианты использования	—

Действия акторов	Отклик системы
1. Пользователь заполняет поля ФИО	2. Система проверяет, заполнены ли поля ФИО
3. Пользователь заполняет поле эл. почты	4. Система проверяет, верно ли заполнено ли поле эл. почты
5. Пользователь заполняет поле пароля	6. Система проверяет, заполнено ли поле пароля минимум четырьмя символами
7. Пользователь выбирает роль «преподаватель» или «студент» и подтверждает регистрацию	8. Система проверяет, не зарегистрирована ли указанная эл. почта в системе, сохраняет информацию о регистрации пользователя и перенаправляет его в созданный для него Личный кабинет
UC-2	Вход
Акторы	Неавторизированный пользователь
Цель	Вход в Личный кабинет пользователя для дальнейшего доступа к функционалу системы
Краткое описание	Ранее зарегистрированный пользователь вводит эл. почту и пароль и получает доступ к своему Личному кабинету
Ссылки на другие варианты использования	—
Действия акторов	Отклик системы
1. Пользователь заполняет поля эл. почты и пароля, подтверждает вход	2. Система проверяет учетные данные и перенаправляет пользователя в его Личный кабинет
UC-3	Создание курса
Акторы	Преподаватель
Цель	Создание курса для дальнейшего размещения на нём учебных материалов и организации оценки успеваемости участников курса
Краткое описание	Пользователь создаёт в системе новый курс с заданным им именем

Ссылки на другие варианты использования	–
Действия акторов	Отклик системы
1. Пользователь заполняет поле с названием курса	2. Система проверяет, заполнено ли поле названия курса
3. Пользователь подтверждает создание курса	4. Система сохраняет данные о созданном курсе и добавляет в Личный кабинет пользователя «плитку» созданного курса
UC-4	Редактирование курса
Акторы	Преподаватель
Цель	Изменение названия, содержания и других характеристик созданного курса
Краткое описание	Пользователь переходит в режим редактирования и, после внесения изменений, сохраняет их
Ссылки на другие варианты использования	<p><i>Включает:</i> [UC-9] Изменение содержания темы курса</p> <p><i>Может включать:</i> [UC-6] Разрешение/запрет загрузки работ, [UC-7] Изменение названия курса, [UC-8] Добавление/удаление темы в содержание курса</p>
Действия акторов	Отклик системы
1. Пользователь нажимает кнопку «Редактировать курс»	2. Система предоставляет пользователю функционал для редактирования курса
3. Пользователь вносит изменения и подтверждает их, нажимая кнопку «Сохранить»	4. Система сохраняет изменения и скрывает функционал для редактирования курса
UC-5	Выбор курса создателем
Акторы	Преподаватель
Цель	Открытие конкретного курса для дальнейшей работы с ним

Краткое описание	Пользователь открывает нужный курс из списка в Личной кабинете
Ссылки на другие варианты использования	<i>Может включать:</i> Просмотр успеваемости, Оценивание работ
Действия акторов	Отклик системы
1. Пользователь переходит в Личный кабинет	2. Система отображает «плитки» с созданными пользователем курсами
3. Пользователь нажимает на название нужного курса, содержащееся в нижней части его плитки	4. Система открывает страницу выбранного курса
UC-6	Разрешение/запрет загрузки работ
Акторы	Преподаватель
Цель	Предоставление/запрет обучающимся возможности загружать свои работы к темам курсов для организации учебного процесса с обратной связью
Краткое описание	В режиме редактирования пользователь устанавливает флажок возможности отправки работ в нужном положении
Ссылки на другие варианты использования	<i>Может быть включено в:</i> [UC-4] Редактирование курса
Действия акторов	Отклик системы
1. Пользователь входит в режим редактирования курса и выбирает тему из содержания, загрузку работ к которой хочет разрешить	2. Система демонстрирует пользователю текущее состояние темы
3. Пользователь устанавливает флажок, предоставляя возможность/запрет загружать работы к теме и сохраняет изменения	4. Система отображает/скрывает поле для загрузки работы на странице курса в содержании темы у участников курса
UC-7	Изменение названия курса
Акторы	Преподаватель
Цель	Изменение названия уже созданного курса

Краткое описание	В режиме редактирования пользователь вводит новое название курса и сохраняет изменения
Ссылки на другие варианты использования	<i>Может быть включено в:</i> [UC-4] Редактирование курса
Действия акторов	Отклик системы
1. Пользователь входит в режим редактирования курса, вводит в поле новое название и сохраняет изменения	2. Система сохраняет новое название вместо предыдущего
UC-8	Добавление/удаление темы в содержание курса
Акторы	Преподаватель
Цель	Добавление/удаление темы в содержании курса для организации структурированного размещения материала на странице курса
Краткое описание	В режиме редактирования пользователь редактирует структуру курса и сохраняет изменения
Ссылки на другие варианты использования	<i>Может быть включено в:</i> [UC-4] Редактирование курса
Действия акторов	Отклик системы
1. Пользователь входит в режим редактирования курса и добавляет/удаляет тему	2. Система добавляет тему в конец списка тем курса/удаляет тему из списка тем курса
3. Пользователь сохраняет изменения	4. Система сохраняет данные о добавленной/удалённой теме
UC-9	Изменение содержания темы курса
Акторы	Преподаватель
Цель	Размещение текстовых учебных материалов в теме курса для реализации учебного процесса
Краткое описание	В режиме редактирования пользователь изменяет содержание темы курса и сохраняет изменения

Ссылки на другие варианты использования	<p><i>Включено в:</i> [UC-4] Редактирование курса</p> <p><i>Может включать:</i> [UC-10] Добавление файлов к содержанию</p>
Действия акторов	Отклик системы
1. Пользователь входит в режим редактирования курса и указывает тему из списка тем курса, содержание которой хочет изменить	2. Система демонстрирует пользователю текущее содержание темы
3. Пользователь изменяет содержание темы в поле редактирования и подтверждает внесённые изменения	4. Система сохраняет изменённое содержание темы и отображает изменённое содержание темы на странице курса
UC-10	Добавление файлов к содержанию
Акторы	Преподаватель
Цель	Размещение файлов учебных материалов в теме курса для реализации учебного процесса
Краткое описание	В режиме редактирования пользователь выбирает и прикрепляет файлы к теме курса и сохраняет изменения
Ссылки на другие варианты использования	<i>Может быть включено в:</i> [UC-9] Изменение содержания темы курса
Действия акторов	Отклик системы
1. Пользователь входит в режим редактирования курса и указывает тему из списка тем курса, содержание которой хочет изменить	2. Система демонстрирует пользователю текущее содержание темы
3. Пользователь загружает/удаляет файлы в содержании темы	4. Система проверяет, поддерживаемого ли формата загружены файлы
5. Пользователь подтверждает внесённые изменения	6. Система сохраняет изменённое содержание темы и отображает изменённое содержание темы на странице курса
UC-11	Оценивание работ

Актеры	Преподаватель
Цель	Оценка загруженных обучающимися работ к темам курсов для организации учебного процесса с обратной связью
Краткое описание	Преподаватель скачивает присланные работы и оценивает их по пятибалльной шкале с возможностью оставления комментариев
Ссылки на другие варианты использования	<i>Может быть включено в:</i> [UC-5] Выбор курса создателем
Действия акторов	Отклик системы
1. Пользователь скачивает работу из таблицы под содержанием темы курса и проверяет работу (за рамками системы «Азбука»), затем выбирает оценку из предложенного списка	2. Система сохраняет оценку к работе, изменяет состояние ячейки (студент, тема) в таблице успеваемости на выставленную оценку, отображает оценку в таблице под содержанием темы курса у преподавателя и отображает оценку в поле под содержанием темы курса у студента, загрузившего работу
3. Пользователь вводит комментарий к работе в поле таблицы под содержанием темы курса и подтверждает отправку	4. Система проверяет, заполнено ли поле комментария, размещает комментарий в таблице под содержанием темы курса у преподавателя и размещает комментарий в поле под содержанием темы курса у студента, загрузившего работу
UC-12	Просмотр успеваемости
Актеры	Преподаватель
Цель	Просмотр сводной таблицы с оценками для контроля успеваемости и сдачи работ
Краткое описание	Преподаватель нажимает кнопку «Успеваемость» и получает доступ к сводной таблице на отдельной странице
Ссылки на другие варианты использования	<i>Может быть включено в:</i> [UC-5] Выбор курса создателем
Действия акторов	Отклик системы

1. Пользователь нажимает на кнопку «Успеваемость» на странице курса	2. Система перенаправляет пользователя на страницу с успеваемостью и показывает ему сводную таблицу (Тема, фамилия, оценка/не сдано)
УС-13	Присоединение к курсу
Актеры	Студент
Цель	Присоединение к конкретному существующему курсу для дальнейшего доступа к его материалам и процессу оценки успеваемости
Краткое описание	Студент переходит по ссылке курса и вводит ключ доступа, после получает доступ к содержанию курса, а «плитка» курса добавляется на его страницу Личного кабинета
Ссылки на другие варианты использования	—
Действия акторов	Отклик системы
1. Пользователь вводит ссылку на существующий в системе курс	2. Система предоставляет поле для ввода ключа доступа
3. Пользователь вводит ключ доступа к курсу	4. Система проверяет, введен ли ключ доступа
5. Пользователь подтверждает ввод ключа доступа	6. Система проверяет ключ доступа, предоставляет пользователю доступ к курсу, добавляет пользователя в список участников курса и добавляет в Личный кабинет пользователя «плитку» курса
УС-14	Выбор курса участником
Актеры	Студент
Цель	Открытие конкретного курса для дальнейшей работы с ним
Краткое описание	Пользователь открывает нужный курс из списка в Личной кабине
Ссылки на другие варианты использования	<i>Включает:</i> Просмотр содержания курса

	<i>Может включать:</i> Комментирование, Отправка работ
Действия акторов	Отклик системы
1. Пользователь переходит в Личный кабинет	2. Система отображает «плитки» с созданными пользователем курсами
3. Пользователь нажимает на название нужного курса, содержащееся в нижней части его плитки	4. Система открывает страницу выбранного курса
UC-15	Отправка работ
Акторы	Студент
Цель	Загрузка обучающимися работ к темам курсов для организации учебного процесса с обратной связью
Краткое описание	Пользователь прикрепляет в соответствующее поле файл и отправляет его на проверку
Ссылки на другие варианты использования	<i>Может быть включено в:</i> Выбор курса участником
Действия акторов	Отклик системы
1. Пользователь загружает работу в поле для загрузки работы на странице курса в содержании темы	2. Система проверяет, поддерживаемого ли формата загружен файл
3. Пользователь подтверждает отправку файла	4. Система сохраняет файл и изменяет состояние ячейки (студент, тема) в таблице успеваемости на «Сдано»
UC-16	Комментирование
Акторы	Студент
Цель	Отправка комментариев к курсу как вид взаимодействия студентов и преподавателей
Краткое описание	Пользователь отправляет комментарии к курсу и видит комментарии других пользователей
Ссылки на другие варианты использования	<i>Может быть включено в:</i> Выбор курса участником

Действия акторов	Отклик системы
1. Пользователь вводит текст комментария в поле под содержанием курса и подтверждает отправку комментария	2. Система проверяет, заполнено ли поле комментария, проверяет количество символов, введенных в поле комментария, сохраняет текст комментария, ФИО отправителя и время отправки и отображает «плитку» комментария под содержанием курса с указанием ФИО отправителя и временем отправки
UC-17	Просмотр содержания курса
Акторы	Студент
Цель	просмотр материалов, размещенных на курсе
Краткое описание	Пользователь отправляет комментарии к курсу и видит комментарии других пользователей
Ссылки на другие варианты использования	<i>Включено в:</i> Выбор курса участником
Действия акторов	Отклик системы
1. Пользователь указывает тему из списка тем курса, содержание которой хочет просмотреть	2. Система показывает пользователю область с развёрнутым содержанием темы под её названием
UC-18	Создание/удаление пользовательских аккаунтов
Акторы	Администратор
Цель	Создание/удаление аккаунтов без использования пользовательского интерфейса системы
Краткое описание	Администратор добавляет/удаляет записи в таблицы с пользовательскими аккаунтами
Ссылки на другие варианты использования	—
Действия акторов	Отклик системы
1. Администратор изменяет таблицу с пользовательскими аккаунтами	2. Система сохраняет данные о новых пользовательских аккаунтах или забывает их

UC-19	Редактирование списка участников курса
Актеры	Администратор
Цель	Добавление/удаление пользователей в список участников конкретного курса без использования пользовательского интерфейса системы
Краткое описание	Администратор добавляет/удаляет записи в таблицы с пользовательскими аккаунтами, являющимися участниками курса
Ссылки на другие варианты использования	—
Действия акторов	Отклик системы
1. Администратор изменяет таблицу с пользовательскими аккаунтами, являющимися участниками курса	2. Система добавляет/удаляет аккаунты в списки участников курса

2.3.2. Диаграмма классов

Диаграмма классов составлена по таблице 6 и представлена на рисунке

2.

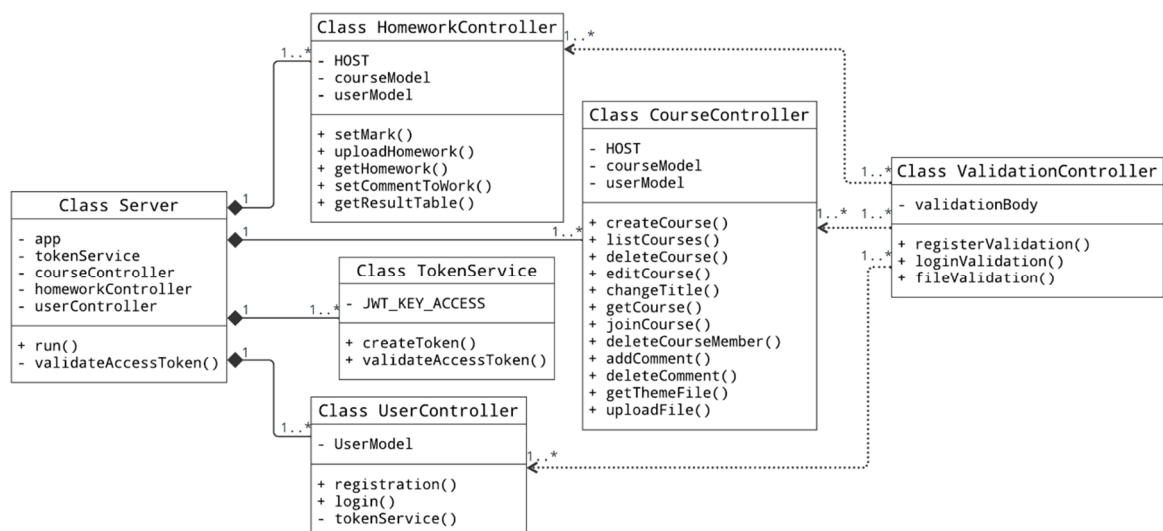


Рисунок 2 – Диаграмма классов

Класс *CourseController* отвечает за работу с курсами. Создает курсы, выдает список курсов, удаляет, изменяет их.

Класс *UserController* отвечает за авторизацию, аутентификацию и регистрацию пользователей.

Класс *TokenService* отвечает за работу с JWT. Проверяет корректность токенов пользователей и создает новые.

Класс *Server* запускает сервер и с помощью фреймворка Express предоставляет API для сайта, соединяет все классы.

Класс *ValidationController* отвечает за валидацию.

Класс *HomeworkController* отвечает за загрузку/скачивание работ, выставление оценок и написание комментариев, а также генерацию таблицы с оценками.

2.3.3. Диаграмма последовательностей

Для демонстрации функционала системы были построены две диаграммы последовательностей:

- 1) для сценариев входа/регистрации (рисунок 3) – здесь последовательности действий разворачиваются в зависимости от того, зарегистрирован ранее пользователь в системе, или нет;
- 2) для основного функционала СДО (рисунок 4) – здесь последовательности действий разворачиваются в зависимости от того, обладает пользователь ролью преподаватель или студент.

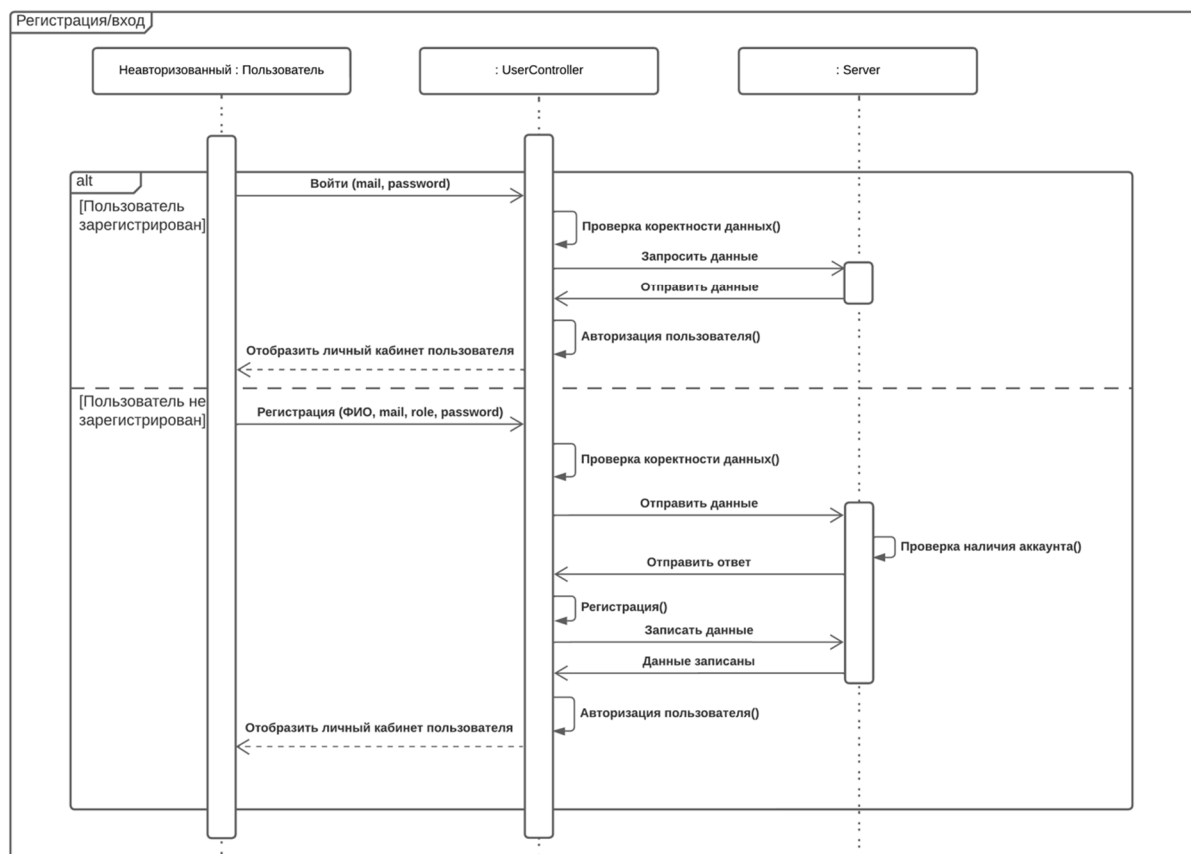


Рисунок 3 – Диаграмма последовательностей для входа/регистрации в СДО

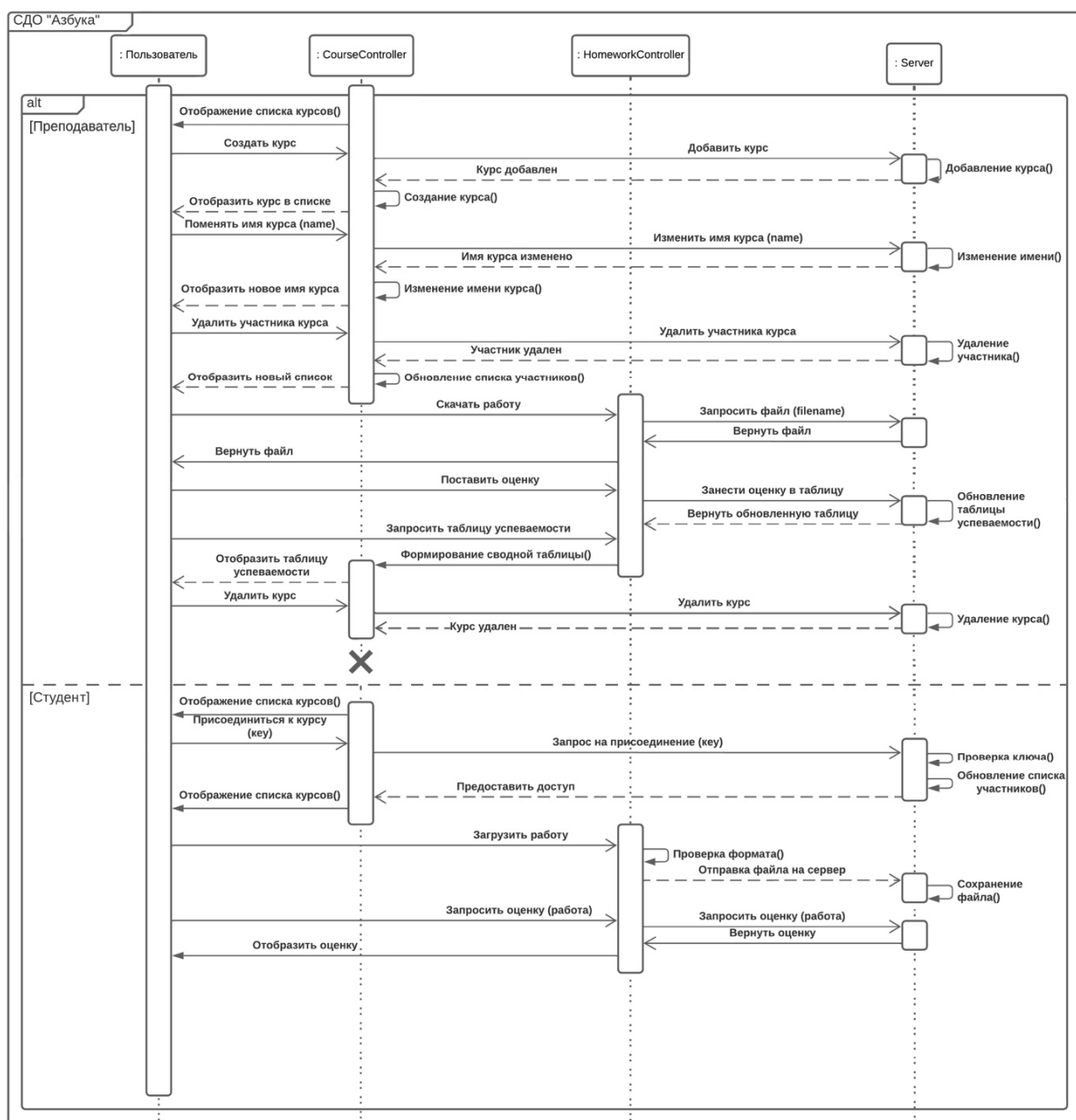


Рисунок 4 – Диаграмма последовательностей для основного функционала СДО

2.3.4. Диаграмма кооперации

Диаграмма кооперации представлена на рисунке 5. В ней задействованы объекты в соответствии с таблицей 6 и пользователи из таблицы 7.

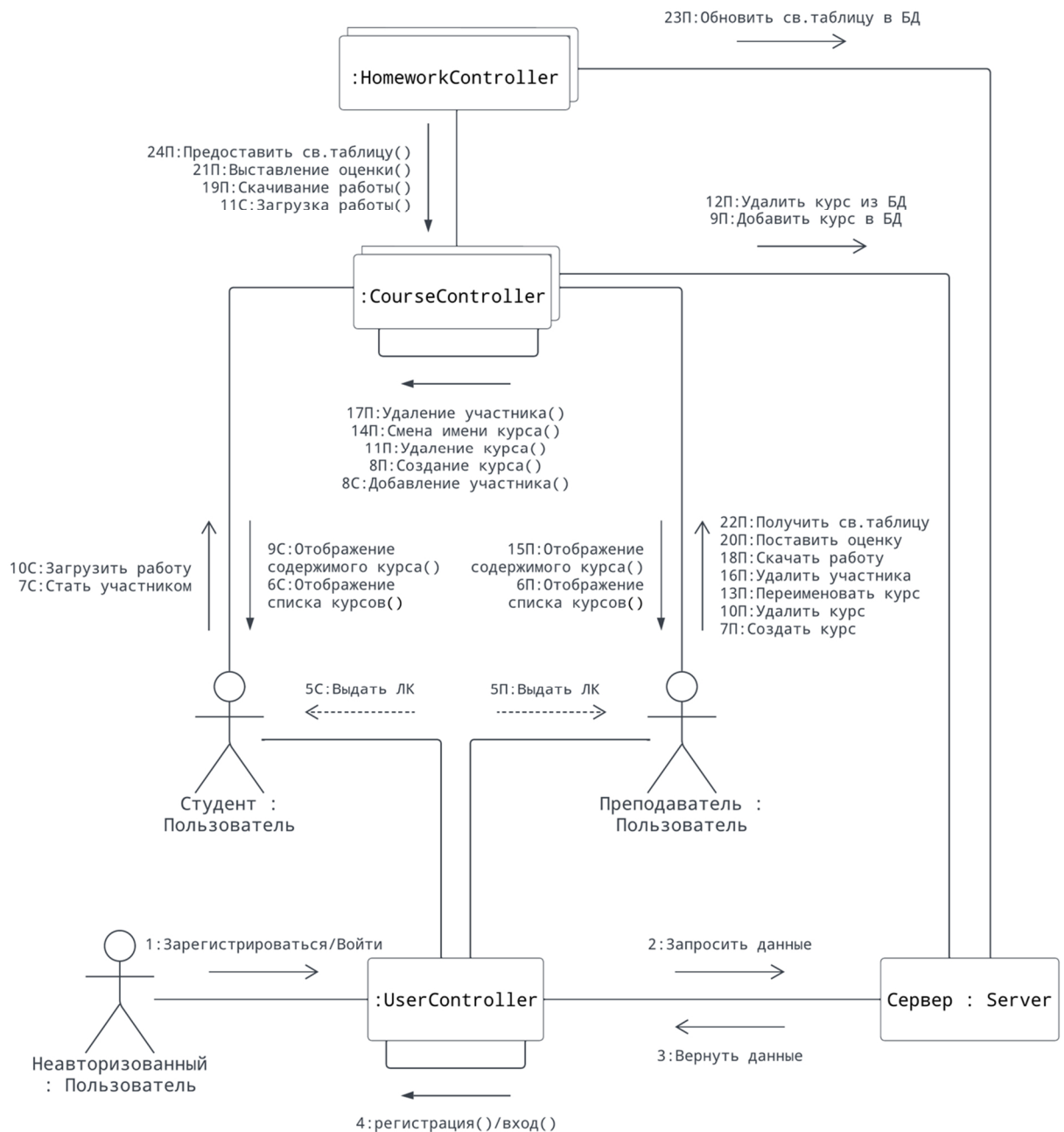


Рисунок 5 – Диаграмма кооперации

2.3.5. Диаграмма состояний

Диаграмма состояний представлена на рисунке 6. Она показывает переход объектов между состояниями.



Рисунок 6 – Диаграмма состояний

2.3.6. Диаграмма активности/деятельности

Для каждого пользователя системы существует определенный набор активностей.

Неавторизированные пользователи могут осуществлять регистрацию и вход. От роли, которая будет в итоге задействована, зависит, какой функционал получит пользователь: преподавателя или студента.

Пользователь с ролью «Преподаватель» может создавать курсы и взаимодействовать с ними: редактировать, просматривать, управлять оценками.

Пользователь с ролью «Студент» может присоединяться к курсам и взаимодействовать с ними: комментировать, просматривать содержание, отправлять работы и просматривать оценки к ним.

Администратор имеет доступ к таблицам с пользовательскими аккаунтами и списками участников курсов. Независимо от взаимодействия пользователей внутри системы, он может редактировать таблицы – удалять, добавлять записи и сохранять изменения.

Диаграмма активности представлена на рисунке 7.

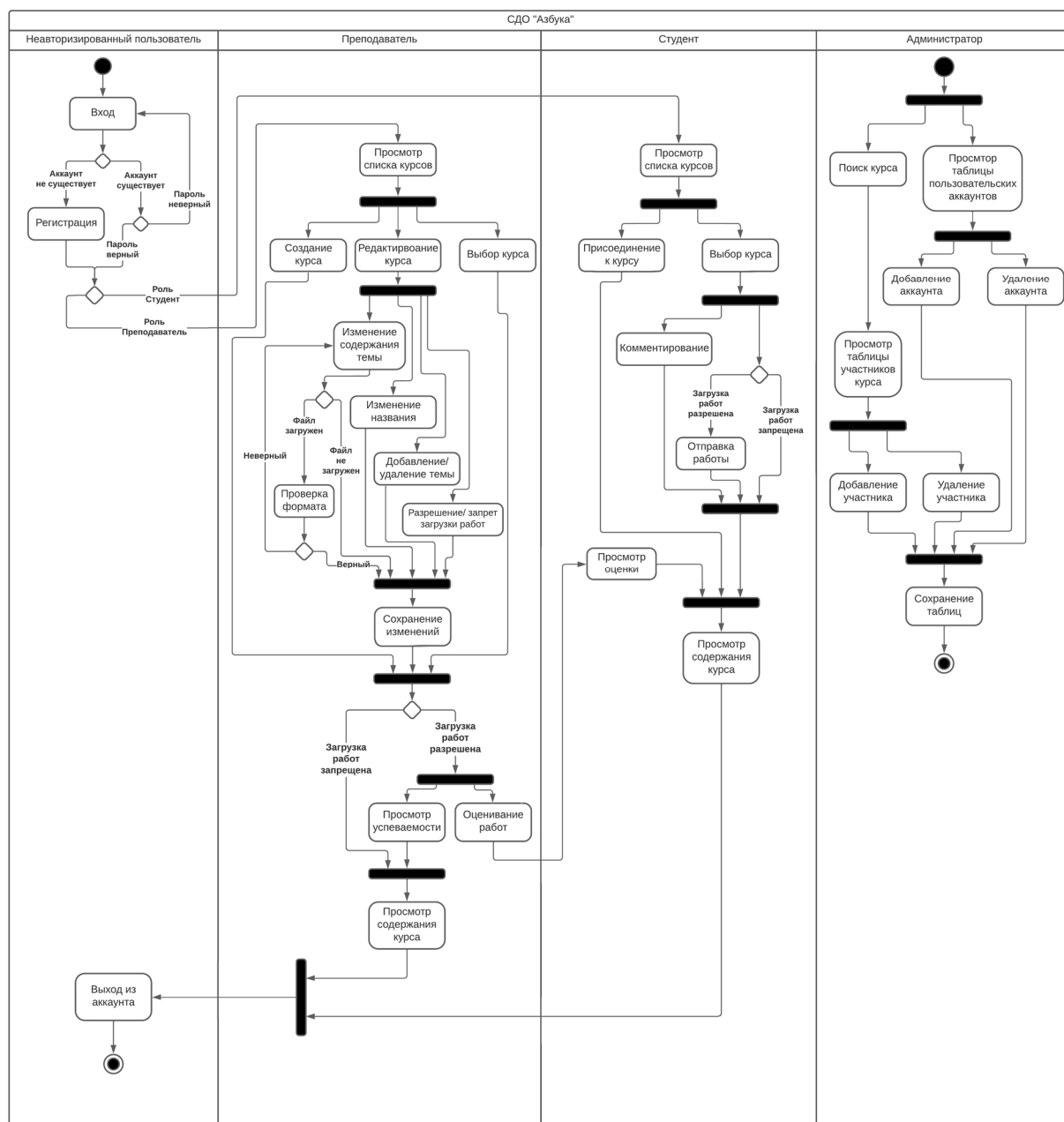


Рисунок 7 – Диаграмма активности

2.3.7. Диаграмма развёртывания

Для развёртывания готового проекта необходимо два сервера. Один сервер с операционной системой GNU/Linux с версией ядра выше 4.19 и установленной средой Node.js, а второй с СУБД MongoDB. Диаграмма развёртывания проекта приведена на рисунке 8.

Администратор сервера должен задать в конфигурационный файл данные о сервере и о СУБД, после чего запустить по руководству веб-

приложение. После успешной развертки станет доступен сайт со всем перечисленным функционалом.

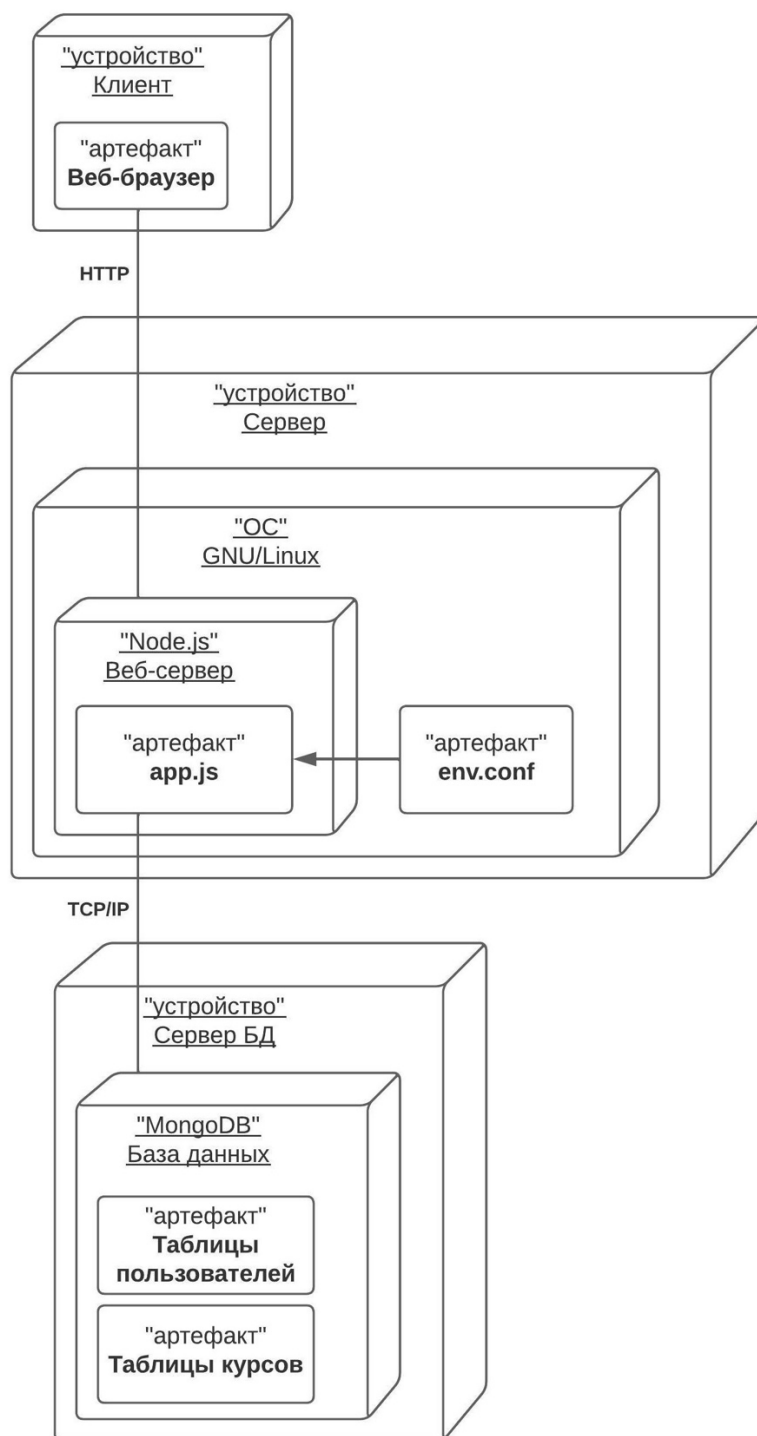


Рисунок 8 – Диаграмма развёртывания

3. ТЕСТЫ

3.1. Тестирование функциональности

Проверки работоспособности системы в соответствии с use cases и заявленным функционалом представлены в таблице 9.

Таблица 9 – Функциональные тесты

ID теста	ID требования	Описание теста
TS-1-1	FRQ-1	Проверка возможности входа в систему.
TS-1-2	FRQ-2	Проверка возможности зарегистрироваться в системе с выбором роли «Преподаватель» или «Студент».
TS-1-3	FRQ-3	Проверка возможности создать курс.
TS-1-4	FRQ-4	Проверка возможности редактировать курс.
TS-1-5	FRQ-5	Проверка возможности создать новые или удалить уже имеющиеся разделы (темы) внутри курса.
TS-1-6	FRQ-6	Проверка возможности отправлять работу в темах курса, отправка работ к которым разрешена преподавателем.
TS-1-7	FRQ-7	Проверка отображения изменений в содержании разделов (тем) курса, в том числе наличие всех добавленных или отсутствие убранных материалов, после сохранения их создателем.
TS-1-8	FRQ-8	Проверка возможности размещения файлов в содержании тем курса, в том числе невозможности размещения файлов неподходящих форматов.
TS-1-9	FRQ-9	Проверка возможности изменить название курса и отображения изменения названия курса у всех его участников.
TS-1-10	FRQ-10	Проверка возможности выбора курса из списка созданных.
TS-1-11	FRQ-11	Проверка возможности оценить или изменить уже выставленную ранее оценку сданных работ участников курса в

		специальной таблице в соответствующем разделе (теме) курса.
TS-1-12	FRQ-12	Проверка отображения выставленной или измененной оценки к сданной работе в сводной таблице.
TS-1-13	FRQ-13	Проверка механизма присоединения к курсу, в том числе ввода ключа доступа к курсу.
TS-1-14	FRQ-14	Проверка возможности выбора курса из списка курсов, в которых пользователь является участником.
TS-1-15	FRQ-15	Проверка отображения материала курса у его участников
TS-1-16	FRQ-16	Проверка возможности отправки работ в специальные поля на странице курса.
TS-1-17	FRQ-17	Проверка отображения оставленного или измененного комментария к сданной работе у участника курса, отправившего её.
TS-1-18	FRQ-18	Проверка возможности редактирования таблиц с пользовательскими аккаунтами, с последующими изменениями в списках участников курсов и удалением курсов, созданных удалёнными пользователями.
TS-1-19	FRQ-19	Проверка возможности редактирования таблиц с участниками курсов, с последующими изменениями в списках участников курсов на страницах курсов.

3.2. Тестирование безопасности

Проверки наличия основных механизмов обеспечения безопасности системы представлены в таблице 10.

Таблица 10 – Тесты безопасности

ID теста	Описание теста
TS-2-1	Проверка отсутствия доступа к системе у пользователя, который не прошел авторизацию.

TS-2-2	Проверка отсутствия возможности повторно присоединиться к курсу пользователю, который ранее был с него исключен.
TS-2-3	Проверка отсутствия возможности подключиться к чужому курсу пользователю с ролью «Преподаватель».
TS-2-4	Проверка способности системы обрабатывать незапланированно высокие нагрузки.
TS-2-5	Проверка невозможности получить доступ к чужим загруженным работам пользователю с ролью «Студент».

3.3. Тестирование производительности

В рамках тестирования производительности следует выполнять стандартные операции в системе, которые описаны в сценариях использования, для выполнения проверок, описанных в таблице 11.

Таблица 11 – Тесты производительности

ID теста	Описание теста
TS-3-1	Проверка основных показателей работоспособности системы при повышении нагрузки со средней до максимальной и определение границ приемлемой производительности.
TS-3-2	Проверка основных показателей работоспособности системы при экстремальной нагрузке.
TS-3-3	Проверка отсутствия отключения и сбоев в работе системы при длительном использовании на средней и высокой нагрузках.

3.4. Тестирование интерфейса

Проверки основных интерфейсных особенностей представлены в таблице 12.

Таблица 12 – Тесты интерфейса

ID теста	Описание теста
TS-4-1	Проверка единства дизайнерских элементов (логотип, утверждённый командой проекта) на каждой странице системы.
TS-4-2	Проверка правильности и языка написания текста на всех элементах интерфейса.
TS-4-3	Проверка отсутствия дефектов интерфейса при масштабировании окна браузера.

ЗАКЛЮЧЕНИЕ

В ходе работы была спроектирована система дистанционного образования, предоставляющая необходимый для организации дистанционного учебного процесса функционал.

К системе были составлены требования, по ним спроектированы следующие UML-диаграммы: диаграмма вариантов использования, диаграмма классов, диаграмма последовательностей, диаграмма кооперации, диаграмма состояний, диаграмма активности, диаграмма развертывания. Данные диаграммы достаточно полно описывают разрабатываемую систему.

Были проведены тестирования требований, а также системы, разработанной по описанным требованиям.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Леоненков А.В. Самоучитель UML 2. – СПб.: БХВ-Петербург, 2007. – 576 с.