1. **Постановка задачи**
   1. **Организационно-экономическая сущность задачи**

Наименование задачи:Онлайн-Курс для кондитеров

Цель разработки:Создание сайта для покупки курсов для кондитеров

Назначение: ПП разрабатывается с целью создания удобной и надежной онлайн платформы, которая предоставляет кондитерам возможность повышать свои навыки благодаря приобретенным курсам

Периодичность использования: Пользователи могут посещать сайт и осуществлять торговые операции в любое удобное для них время в зависимости от своих потребностей. Сайт доступен для использования круглосуточно, 7 дней в неделю. Периодичность использования сайта зависит от активности пользователей и может варьироваться от ежедневного посещения до реже в зависимости от потребностей и предпочтений каждого отдельного пользователя.

Источники и способы получения данны**х:** собственные разработки

* 1. **Функциональные требования**

Описание перечня функций и задач, которые должен выполнять будущий ПП:

Гость:

* просмотр страниц сайта;
* просмотр отзывов;
* ознакомление со всеми видами курсов;
* регистрация;

Пользователь:

* aвторизация;
* приобретение курса;
* написание отзывов;
* взаимосвязь с менеджером сайта (чат);

Менеджер сайта:

* управление заказами;
* обновление информации;
* отслеживание отзывов;
* обратная связь с пользователями;
  1. **Описание процессов с входной, выходной и условно-постоянной информацией**

Входная информация :

* фото;
* описание курса;
* данные пользователя;

Выходная информация :

* видео;
* теория ( в зависимости от вида курса );

Условно-постоянная информация :

* курсы;
* Отзывы;
* контактная информация;
  1. **Эксплуатационные требования**

требования к применению:помогает быстро приобрести курс и повысить свои навыки благодаря им

требования к реализации:Для реализации статических страниц и шаблонов должны использоваться языки HTML и CSS. Для реализации интерактивных элементов клиентской части должны использоваться языки JavaScript. Для реализации динамических страниц должен использоваться язык PHP.

требования к надежности:Система может быть недоступна не более чем 24 часа в год. У администратора приложения должна быть возможность выгрузить и загрузить копию **.**

требования к интерфейсу:При разработке сайта должен быть лаконичный

и подробный дизайн, который будет привлекательным для пользователей, с удобной навигацией преимущественного нежно розового/белового оттенка. Основные разделы сайты должны быть доступны с первой страницы. Удобная навигация играет ключевую роль в опыте пользователя, поэтому основные разделы сайта должны быть легко доступны с первой страницы. Это поможет пользователям быстро ориентироваться на сайте и находить необходимую информацию без лишних усилий. Грамотный пользовательский интерфейс включает в себя интуитивно понятные элементы управления, понятную структуру страниц и легкость взаимодействия с контентом. Важно также обеспечить адаптивность сайта под различные устройства, такие как компьютеры, телефоны и планшеты, чтобы пользователи могли комфортно пользоваться сайтом независимо от устройства, которое они используют. В личном кабинете пользователя следует предусмотреть набор видео для обучения и возможность общения с менеджером через чат. Это поможет пользователям получать необходимую информацию и поддержку, делая пользовательский опыт более полноценным и удовлетворительным.

1. **Проектирование**
   1. **Выбор стратегии разработки и модели жизненного цикла**

Таблица 1 **–** Выбор модели жизненного цикла на основе характеристик требований

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерий требования | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Являются ли требования к проекту легко определимыми и реализуемыми? | Да | Да | Да | Нет | Нет | Нет |
| 2. | Могут ли требования быть сформулированы в начале ЖЦ? | Да | Да | Да | Да | Нет | Нет |
| 3. | Часто ли будут изменяться требования на протяжении ЖЦ? | Нет | Нет | Нет | Нет | Да | Да |
| 4. | Нужно ли демонстрировать требования с целью их определения? | Нет | Нет | Да | Нет | Да | Да |
| 5. | Требуется ли проверка концепции программного средства или системы? | Нет | Нет | Да | Нет | Да | Да |
| 6. | Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 7. | Нужно ли реализовать основные требования на ранних этапах разработки? | Нет | Нет | Да | Да | Да | Да |

Продолжение таблицы 1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 8. | Итого: | 3 | 3 | 4 | 4 | 4 | 4 |

Итог: на основе результатов заполнения табл. 3 подходящей является RAD, инкрементная, быстрого прототипирования и эволюционная модель.

Таблица 2 **–** Выбор модели жизненного цикла на основе характеристик команды разработчиков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории команды разработчиковпроекта | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Являются ли проблемы предметной области проекта новыми для большинства разработчиков? | Нет | Нет | Нет | Нет | Да | Да |
| 2. | Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков? | Да | Да | Нет | Нет | Нет | Да |
| 3. | Изменяются ли роли участников проекта на протяжении ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 4. | Является ли структура процесса разработки более значимой для разработчиков, чем гибкость? | Да | Да | Нет | Да | Нет | Нет |
| 5. | Важна ли легкость распределения человеческих ресурсов проекта? | Да | Да | Да | Да | Нет | Нет |
| 6. | Приемлет ли команда разработчиков оценки, проверки, стадии разработки? | Да | Да | Нет | Да | Да | Да |

Продолжение таблицы 2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 7. | Итого: | 4 | 4 | 3 | 4 | 3 | 2 |

Итог: На основе результатов заполнения табл. 4 подходящими являются каскадная, v-образная, и инкрементная модель.

Таблица 3 **–** Выбор модели жизненного цикла на основе характеристик коллектива пользователей

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории коллектива пользователей | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Будет ли присутствие пользователей ограничено в ЖЦ разработки? | Да | Да | Нет | Да | Нет | Да |
| 2. | Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки? | Нет | Нет | Нет | Да | Да | Да |
| 3. | Будут ли пользователи вовлечены во все фазы ЖЦ разработки? | Нет | Нет | Да | Нет | Да | Нет |
| 4. | Будет ли заказчик отслеживать ход выполнения проекта? | Нет | Нет | Нет | Нет | Да | Да |
| 5. | Итого: | 1 | 1 | 2 | 2 | 3 | 1 |

Итог: На основе результатов заполнения табл. 5 подходящей является модель быстрого прототипирования

Таблица 4 **–** Выбор модели жизненного цикла на основе характеристик типа проектов и рисков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории типов проекта и рисков | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Разрабатывается ли в проекте продукт нового для организации направления? | Нет | Нет | Нет | Да | Да | Да |
| 2. | Будет ли проект являться расширением существующей системы? | Да | Да | Да | Да | Нет | Нет |
| 3. | Будет ли проект крупно- или среднемасштабным? | Нет | Нет | Нет | Да | Да | Да |
| 4. | Ожидается ли длительная эксплуатация продукта? | Да | Да | Нет | Да | Нет | Да |
| 5. | Необходим ли высокий уровень надежности продукта проекта? | Нет | Да | Нет | Да | Нет | Да |
| 6. | Предполагается ли эволюция продукта проекта в течение ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 7. | Велика ли вероятность изменения системы (продукта) на этапе сопровождения? | Нет | Нет | Нет | Да | Да | Да |

Итог: На основе результатов заполнения табл. 6 подходящими являются v-образная, инкрементная и эволюционная модель.

Общий итог: в итоге заполнения табл. 1 – 4 наиболее подходящей является инкрементная.

13 каскадная модель

14 V-образная модель

13 RAD модель

16 инкрементная модель

14 быстрого прототипирования модель

13 эволюционная модель.

**Приложение А**

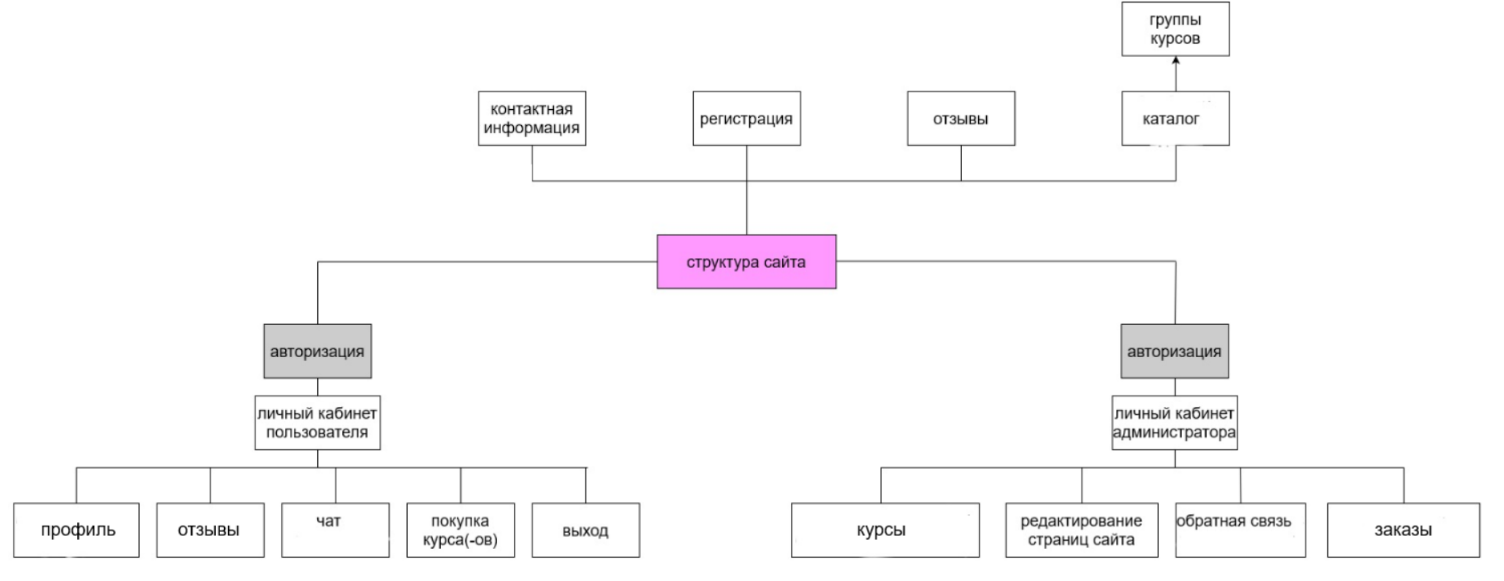


Рисунок 1 - Структурное проектирование ПО



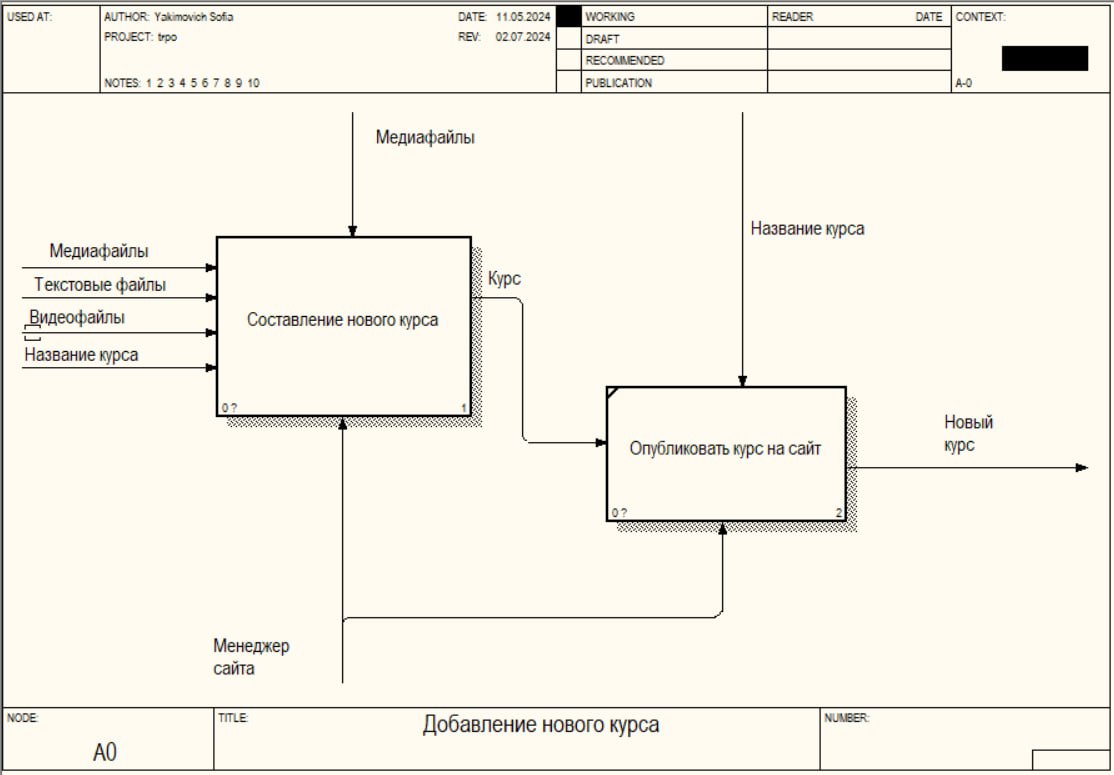


Рисунок 3 - Функциональная модель

Рисунок 2 - Функциональная модель

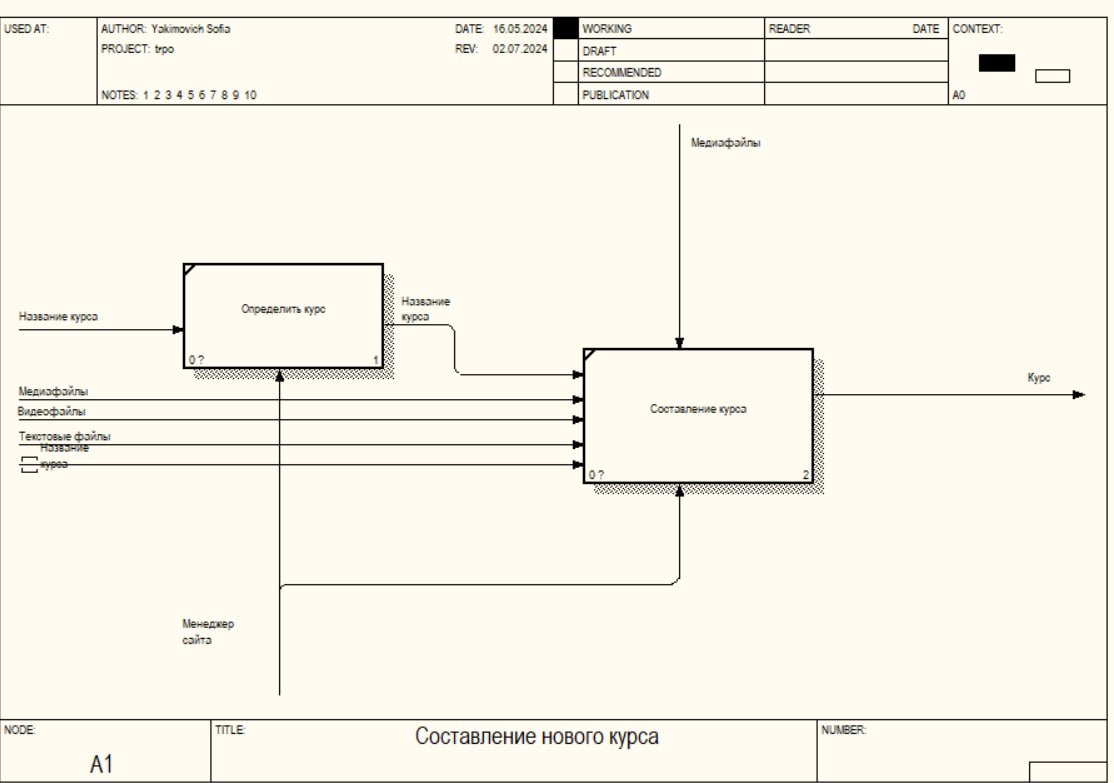


Рисунок 3 - Функциональная модель

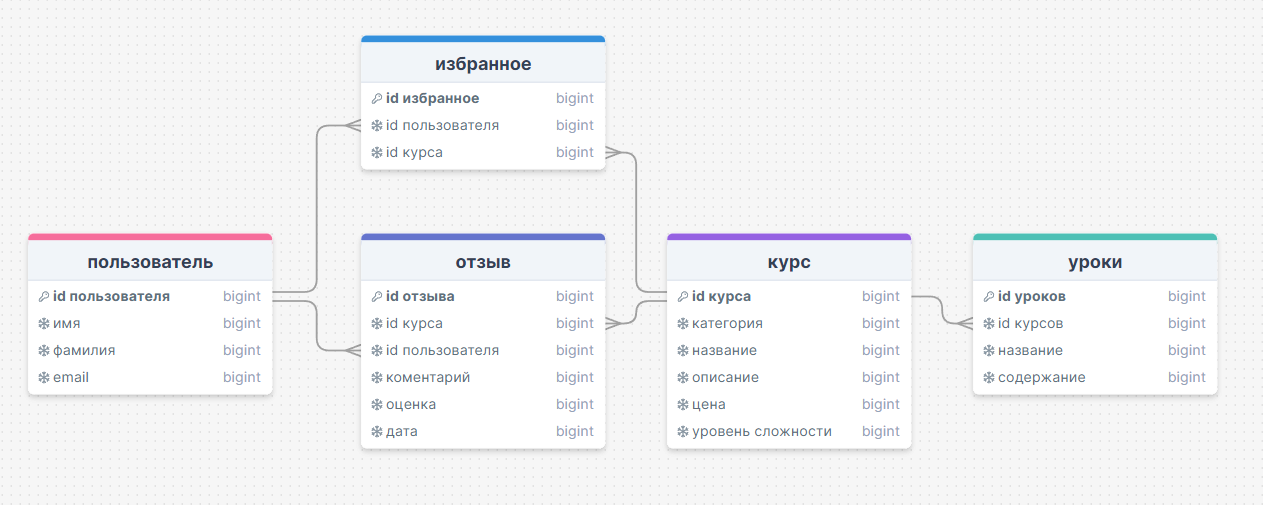


Рисунок 4 – База данных

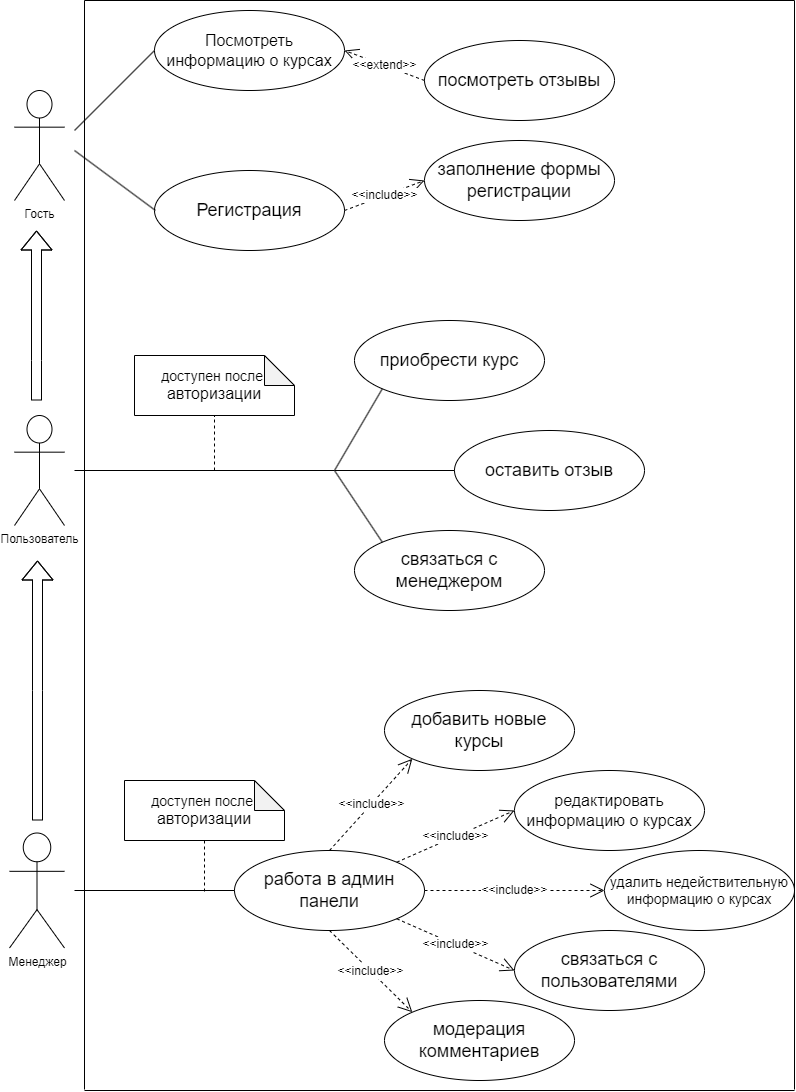


Рисунок 5 – Диаграмма вариантов использования

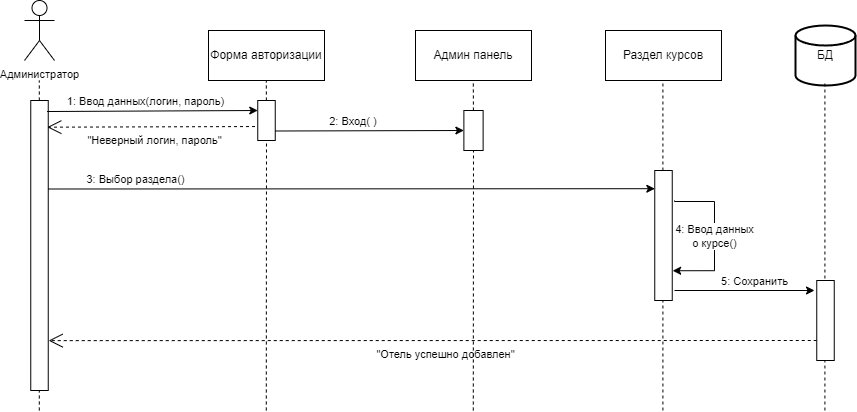


Рисунок 6 – Диаграмма последовательности

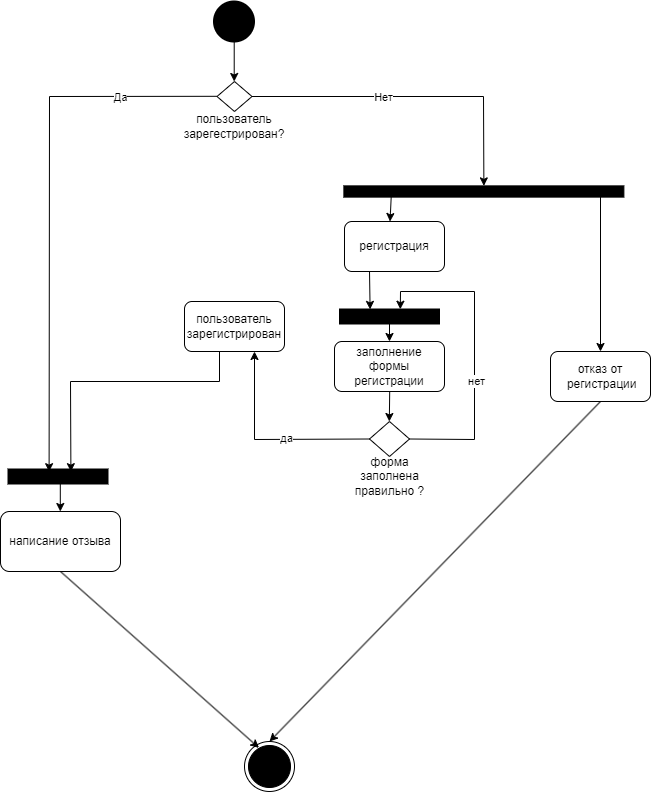


Рисунок 7 – Диаграмма деятельности

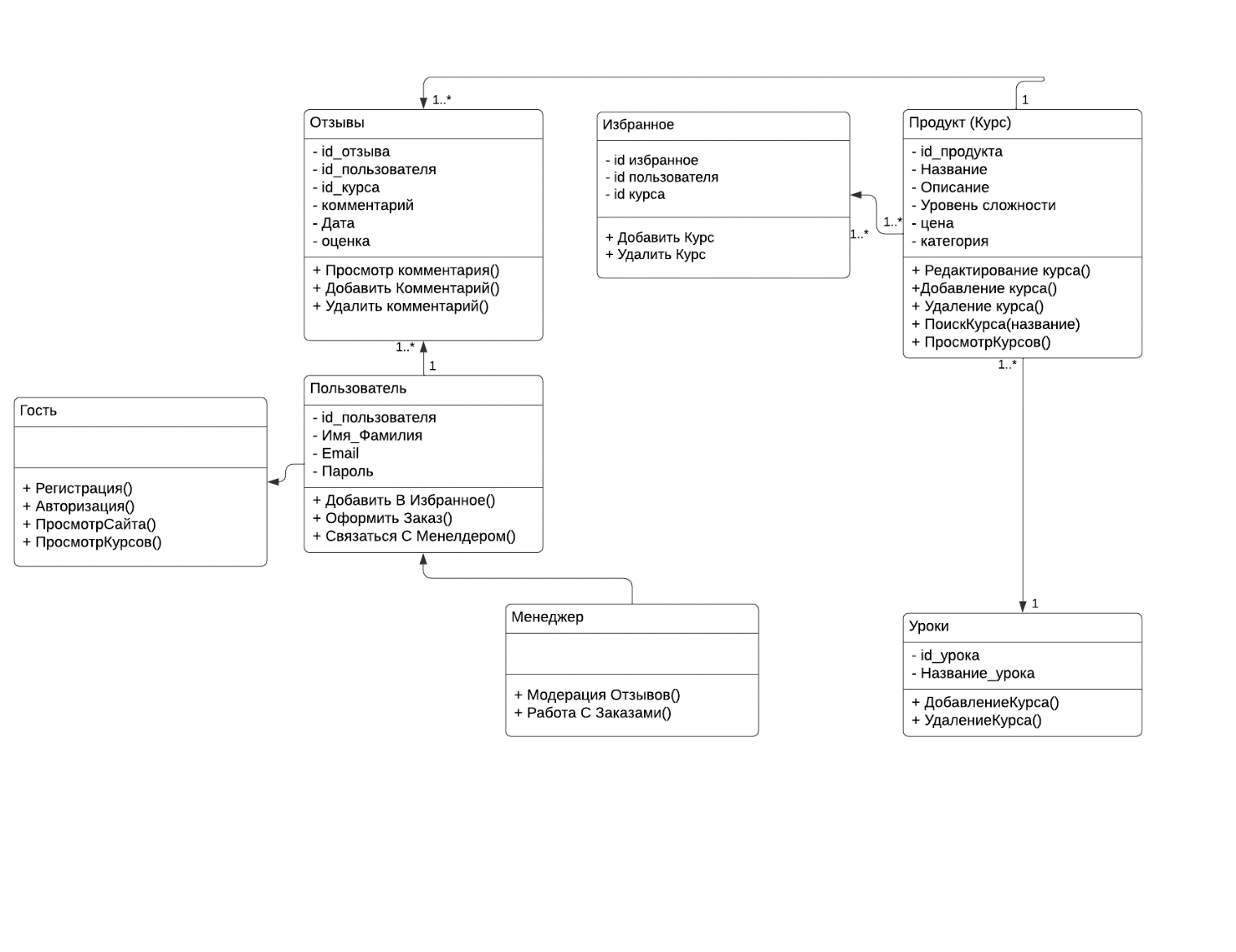


Рисунок 8 – Диаграмма классов

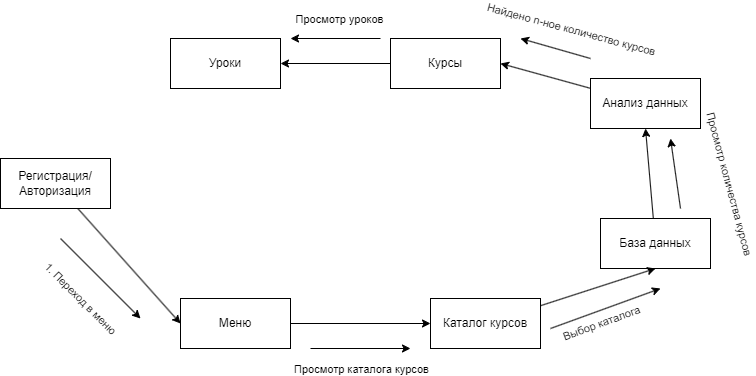


Рисунок 9 – Диаграмма объектов