

# **ПРОЕКТИРОВАНИЕ АРХИТЕКТУРЫ ИНФОРМАЦИОННОЙ СИСТЕМЫ ПЛАТФОРМЫ ДЛЯ РАЗРАБОТКИ И ПУБЛИКАЦИИ ОБРАЗОВАТЕЛЬНЫХ КУРСОВ**

**Иманов В. А.**, студент 4 курса факультета физики и математики по направлению «Прикладная информатика в информационной сфере»  
ФГБОУ ВО «Уфимский университет науки и технологий», г. Бирск

**Аннотация:** В данной статье рассматривается проектирование информационной системы для платформы, предназначенной для разработки и публикации образовательных курсов. Будут освещены ключевые аспекты проектирования, включая архитектуру системы и даталогическую модель данных.

**Ключевые слова:** проектирование, информационная система, платформа, образовательные курсы, архитектура системы, чистая архитектура, даталогическая модель данных, диаграмма отношений.

## **Введение**

В современном мире, где информационные технологии развиваются в очень ускоренном темпе, образование становится все более и более доступным, благодаря платформам, предоставляющим возможности обучения в любом месте и в любое время. Соответственно, данные платформы становятся все более известными широкой аудитории. Вместе с тем, проектирование информационной системы для данных платформ требует все более глубокого анализа и учета различных аспектов системы.

## **Проектирование архитектуры системы**

Одним из важных этапов при проектировании системы является описание ее архитектуры, так как это во многом определяет дальнейшую программную реализацию данной информационной системы.

Важно определить модули системы, их взаимосвязи и функциональность.

Типичная архитектура такой платформы может включать следующие компоненты:

1. Серверная часть: включает в себя обработку запросов от клиентской части, предоставляет внутреннюю бизнес-логику системы, включая управление курсами, их содержанием, а также пользователями, их ролями и т. д.
2. Клиентская часть: отвечает за пользовательский интерфейс, предоставляет взаимодействие с пользователями, а также отправляет запросы на серверную часть и обрабатывает полученные ответы;
3. Хранилище данных: используется для хранения информации о курсах, пользователях, их ролях, а также прогрессе обучения.

Далее будет рассмотрена более детальная архитектура информационной системы.

При проектировании информационной системы для платформы разработки и публикации образовательных курсов использовалась концепция чистой архитектуры (Clean Architecture), которая обеспечивает высокую гибкость, масштабируемость и поддерживаемость системы [4]. В данной структуре выделяются следующие компоненты:

- 1) Компонент Core (ядро системы): представляет собой основные сущности предметной области, такие как пользователи, роли, курсы, модули, подписки и прочие. Здесь определяются ключевые абстракции, не зависящие от конкретных технических деталей.
- 2) Компонент Application (инфраструктура системы): представляет собой взаимодействие с сущностями и описывает основную часть бизнес-логики информационной системы, такие как: создание курсов и их редактирование, заполнение курсов модулями, добавление новых пользователей и т.д. Отвечает за координацию работы различных компонентов системы, а также обеспечивает выполнение бизнес-правил.

- 3) Компонент Adapter(слой данных системы): представляет собой компонент, на уровне которого происходит работа с данными, в которую включается соединение и взаимодействие с базой данных, преобразование данных, полученных из СУБД, в данные, которые используются в приложении;
- 4) Компонент Shared (слой общих данных): представляет собой компонент, в котором описываются общие данные для взаимодействия клиентской и серверной части. Этот слой содержит абстракции, которые могут быть использованы как на сервере, так и на клиенте. Такими абстракциями служат, так называемые DTO (DataTransferObjects), которые представляют собой простые объекты, которые легко можно преобразовать в формат данных для общения сервера с клиентом.
- 5) Компонент Api: приложение, представленное контроллерами для обработки запросов и описанием конечных точек (Endpoints). Этот компонент обеспечивает взаимодействие клиентского приложения с сервером [2].
- 6) Компонент клиентской части: клиентское приложение, в котором реализован пользовательский интерфейс. Отвечает за представление данных и взаимодействие с пользователем.

Далее представлена UML-диаграмма компонентов[1], которая отражает зависимости вышеописанных компонентов, а также их взаимодействие.

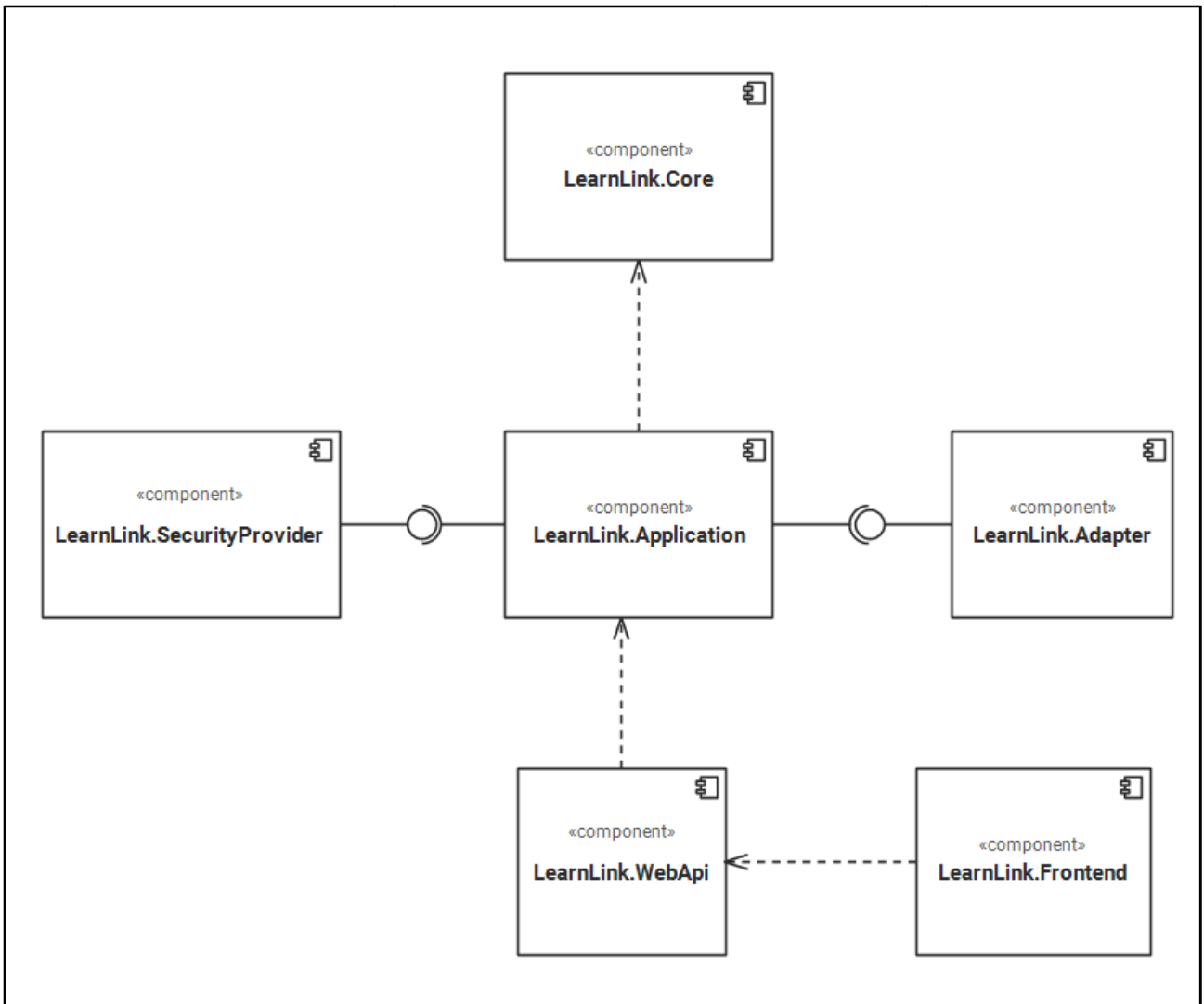


Рис 1. UML диаграмма компонентов информационной системы

Таким образом, в данной архитектуре каждый слой имеет четко определенную ответственность и зависит только от абстракций более высокого уровня, что обеспечивает легкость замены или модификации отдельных компонентов без влияния на остальные. Такой подход позволяет создавать масштабируемые и гибкие системы, способные эффективно адаптироваться к изменяющимся требованиям и условиям.

### **Проектирование даталогической модели данных**

Даталогическая модель предоставляет высокоуровневое представление структуры данных и их отношений в системе. На рисунке 2 отображена даталогическая модель данных в виде диаграммы отношений [3](EntityRelationshipsDiagram).

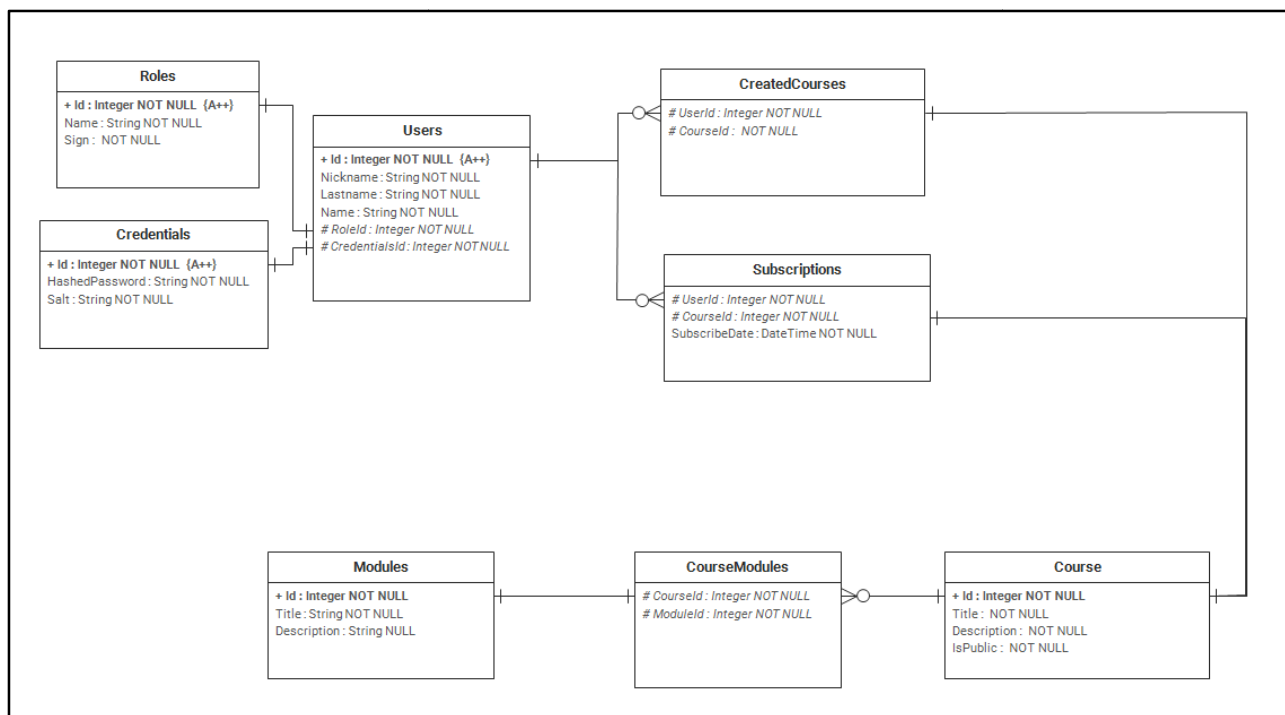


Рис 2. Дatalogическая модель данных в виде диаграммы отношений (ERD)

Таким образом, было определено 8 сущностей: «Пользователь», «Роль», «Учетные данные», «Созданный курс», «Подписка», «Курс», «Модуль курса», «Модуль». Каждый из них в системе имеет свою собственную функциональную роль и представляет собой ключевой элемент для организации и управления данными.

## Вывод

Этап проектирования является одним из важных этапов при разработке информационной системы. На данном этапе может быть описана архитектура системы, могут быть построены различные модели и диаграммы, что позволит учесть многие аспекты для более эффективной разработки программной реализации информационной системы.

## Литература

1. Аникеев, Д. В. Архитектура информационных систем : учебное пособие / Д. В. Аникеев. — Рязань : РГРТУ, 2022. — 72 с. — Текст :

- электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/380360>(дата обращения: 25.03.2024). — Режим доступа: для авториз. пользователей.
2. Коптева, А.В. АНАЛИЗ ЧИСТОЙ АРХИТЕКТУРЫ GOLANG REST API С ВНЕДРЕНИЕМ ЗАВИСИМОСТЕЙ, СЛЕДУЯ ПРИНЦИПАМ SOLID / А. В. Коптева, И. В. Князев // Проблемы современной науки и образования. — 2021. — № 9. — С. 23-30. — ISSN 2304-2338. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/journal/issue/333386>(дата обращения: 25.03.2024). — Режим доступа: для авториз. пользователей.
  3. Махмутова, М. В. Теория и практика разработки баз данных : учебное пособие / М. В. Махмутова. — 2-е изд., стер. — Москва : ФЛИНТА, 2023. — 185 с. — ISBN 978-5-9765-3695-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/348275>(дата обращения: 25.03.2024). — Режим доступа: для авториз. пользователей.
  4. Чистая архитектура для веб-приложений // Habr[Электронный ресурс]. URL: <https://habr.com/ru/post/534092/>(дата обращения 25.03.2024). — Режим доступа: общедоступный.