# Портал для любителей «Что? Где? Когда?»

# Общее определение

# Централизованное постоянно обновляющееся хранилище информации по теме ЧГК; интерактивное соревнование между людьми, увлекающимися игрой.

# Круг пользователей

# От простых физических до юридических лиц и организаций: школьники, студенты, простые любители "Что? Где? Когда?", ученые, научные организации и сообщества.

# Общие возможности

# Ознакомление с игрой "Что? Где? Когда?"; просмотр новостей в мире "Что? Где? Когда?"; чтение статей, на различные темы; создание вопросов и статей; участие в соревнованиях; создание личного кабинета; Приложение предоставляет определенной группе пользователей полный контроль над содержимым с точки зрения информации.

# Спецификация вариантов использования системы

Визуальное моделирование в UML можно представить, как некоторый процесс поуровневого спуска от наиболее обшей и абстрактной концептуальной модели исходной системы к логической, а затем и к физической модели соответствующей программной системы. Для достижения этих целей вначале строится модель в форме так называемой диаграммы вариантов использования (use case diagram), которая описывает функциональное назначение системы или, другими словами, то, что система будет делать в процессе своего функционирования. Диаграмма вариантов использования является исходным концептуальным представлением или концептуальной моделью системы в процессе ее проектирования и разработки. На рисунке 3.1 представлена диаграмма вариантов использования разрабатываемой системы.



Рисунок 3.1 – Диаграмма вариантов использования

Рассматриваемая система имеет трех актеров: администратора, пользователя, гостя.

Гость – это незарегистрированный пользователь, который имеет ограниченный набор функций. Гостю доступна возможность просмотра статей, новостей и информации о проекте. Также гость всегда имеет возможность зарегистрироваться в системе и стать пользователем.

Пользователь же имеет доступ к базе вопросов и может отвечать на вопросы, создавать свои, подавать апелляции в случае необходимости, а также соревноваться с другими пользователями. У каждого пользователя есть личный кабинет, в котором отображаются его данные и рейтинг.

В системе предусмотрена роль администратора, который вправе утверждать вопросы, присланные пользователями, разрешать апелляции, начислять баллы, а также банить пользователей, нарушающих правила пользования порталом.

# Информационная модель системы и её описание

База данных интернет-портала строится на основе данных, добавляемых пользователями и администраторами системы, ответственность за достоверность информации, содержащейся в публикациях, несут лица, их публикующие.

С помощью интернет-портала проводится сбор и обработка информации, имеющей отношение к игре «Что? Где? Когда?». Некоторые данные, например, вопросы для игры могут являться результатами интеллектуальной деятельности пользователей портала. Для поддержания работоспособности системы потребуется постоянное обновление и добавление данных.

В результате обработки исходных данных пользователю предоставляется актуальная информация по выбранной тематике. Использование предлагаемого интернет-портала предоставит пользователям возможность интерактивной работы и удаленного доступа к необходимой информации, а также позволит получать информацию в мультимедийном виде.

Так, как данный портал предназначен для работы с большими объемами данных, поэтому необходимо создать наиболее оптимальную информационную модель системы. В связи с этим необходимо провести нормализацию базы данных. Целью нормализации является исключение избыточного дублирования данных, которое является причиной аномалий, возникающих при добавлении, редактировании и удалении кортежей (строк таблицы).

Ниже описаны 3 нормальные формы (далее НФ) базы данных.

Отношение находится в 1НФ, если все его атрибуты являются простыми, все используемые домены должны содержать только скалярные значения. Не должно быть повторений строк в таблице.

Отношение находится во 2НФ, если оно находится в 1НФ и каждый не ключевой атрибут неприводимо зависит от Первичного Ключа (ПК).

Неприводимость означает, что в составе потенциального ключа отсутствует меньшее подмножество атрибутов, от которого можно также вывести данную функциональную зависимость.

Отношение находится в 3НФ, когда находится во 2НФ и каждый не ключевой атрибут нетранзитивно зависит от первичного ключа. Проще говоря, второе правило требует выносить все не ключевые поля, содержимое которых может относиться к нескольким записям таблицы в отдельные таблицы.

В соответствии с вышеизложенными правилами была спроектирована информационная модель системы. На рисунке 4.1 изображена схема информационной модели системы, легко убедиться, что она соответствует трем нормальным формам.

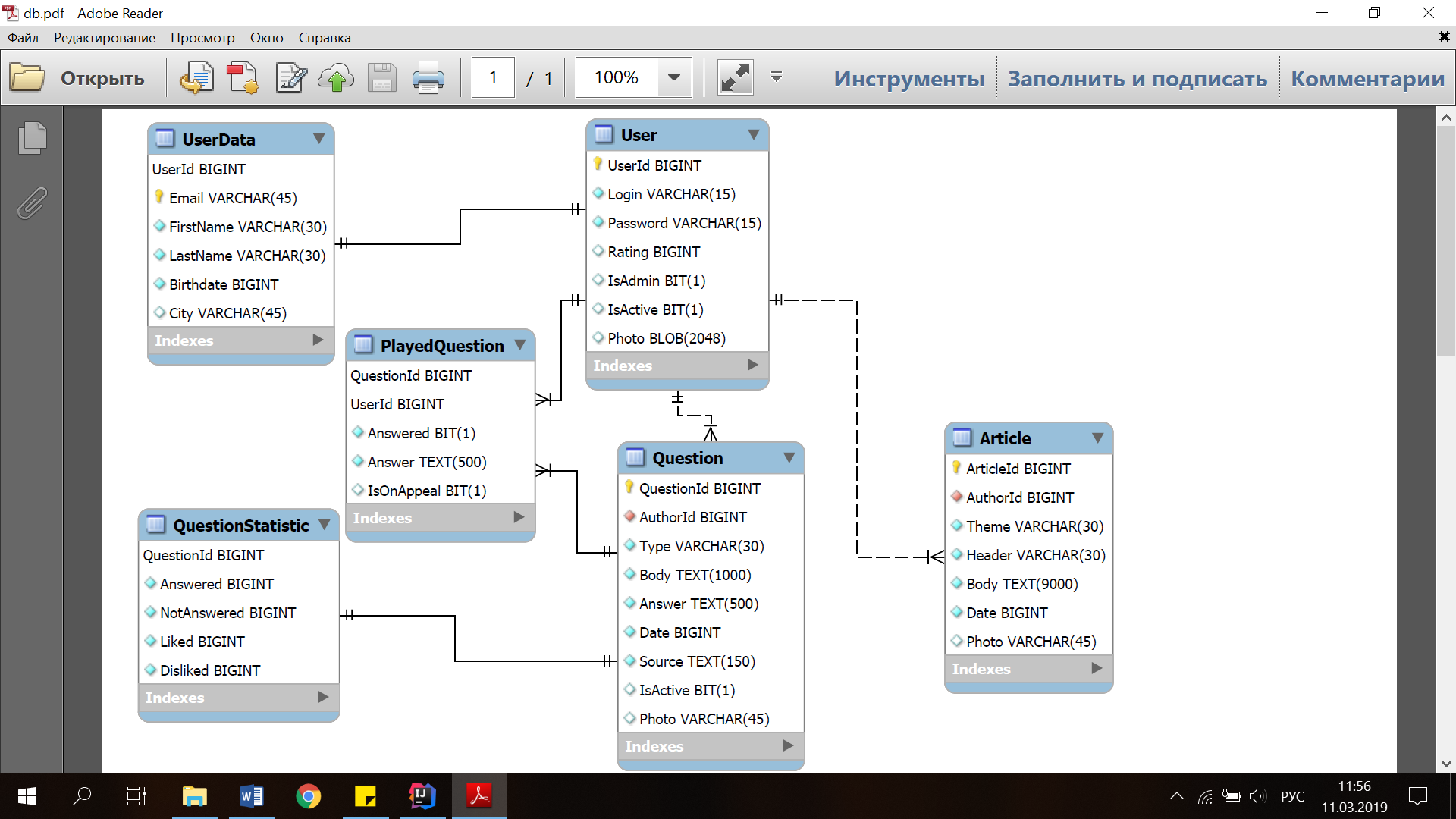


Рисунок 4.1 – Информационная модель системы

Информационная модель представлена 6 таблицами, в которых хранятся все необходимые данные информационного портала.

# Модели представления системы и их описание

Создаваемое программное обеспечение (ПО) постоянно усложняется. При работе над любыми информационными (распределенными) системами в первую очередь возникает проблема взаимопонимания программиста и заказчика уже на стадии обсуждения структуры системы. До начала кодирования программы предлагаемая концепция предусматривает решение двух предварительных задач: проанализировать поставленную перед разработчиком задачу и разработать проект будущей системы. Программа разбивается на отдельные модули, взаимодействующие между собой с помощью механизмов передачи параметров.

Унифицированный язык моделирования (Unified Modeling Language) − графический язык визуализации, специфицирования, конструирования и документирования программного обеспечения. С помощью UML можно разработать детальный план создаваемой системы, отображающий системные функции и бизнес-процессы, а также конкретные особенности реализации.

На рисунке 6.1 изображена диаграмма развертывания, которая показывает топологию системы.

# 

Рисунок 6.1 – Диаграмма развертывания системы

На рисунке 6.2 изображена диаграмма состояний, показывающая переходы между состояниями при создании вопроса пользователем системы.

# 

Рисунок 6.2 – Диаграмма состояний системы

На рисунке 6.3 представлена диаграмма компонентов, она показывает структуру системы.

# 

Рисунок 6.3 – Диаграмма компонентов

Взаимодействие объектов системы представлено с помощью диаграммы последовательностей, изображенной на рисунке 6.4.

# 

Рисунок 6.4 – Диаграмма последовательностей

Далее на рисунке 6.5 приложения представлена диаграмма классов системы. Использованы следующие паттерны: поведенческий паттерн Command, порождающий паттерн Singleton, MVC(Model/View/Controller) и DAO(Data Access Object).



Рисунок 6.5 – Диаграмма классов