**Отчет по лабораторной работе № 5**

**студента гр. И-2-19**

**Ситжалилов Ахтем Леманович**

По дисциплине: Проектирование информационных систем

Тема занятия: Разработка - sprint 3

Цель занятия: научиться разрабатывать ИС с использованием Scrum методологии.

Sprint по реализации бизнес-логики ИС (MVP)

**1.Задание**

1) Бизнес-логика (базовые сущности и их взаимодействия) информационной системы;

2) Валидация данных, в т.ч. обязательных полей, в бизнес-сервере;

3) Обработка ошибок. Если пользователь допускает ошибку, или же программа

выполняет недопустимое действие и происходит незапланированная ошибка, то

поведение программы не должно вызвать у пользователя непонимание. Программа не

должна аварийно прекращать свою работу.

4) Блокировка данных. В многопользовательской ИС очень важно следить за

конфликтами доступа к данным различных пользователей. Если пользователь начал

редактировать объект, необходимо не допустить к этому объекту других пользователей.

**2. Результат выполнения лабораторной работы**

****1) Бизнес-логика (базовые сущности и их взаимодействия) информационной системы;

Рисунок 1 – Система разделения базы

Информационная система должна быть разделена на три подсистемы: Клиентское приложение, серверное приложение и сервер базы данных рисунок 1. Такое разделение позволяет ограничить доступ к базе данных, тем самым защитить личные данные пользователей.

2) Валидация данных, в т.ч. обязательных полей, в бизнес-сервере;

Валидацию вводимых данных можно выполнять как на стороне сервера, так и на стороне клиентского приложения. Наиболее верным решением является определения логики валидации данных на стороне сервера. Это обеспечит мгновенное изменение валидации данных, в случае изменения бизнес-логики процесса.

3) Обработка ошибок. Если пользователь допускает ошибку, или же программа

выполняет недопустимое действие и происходит незапланированная ошибка, то

поведение программы не должно вызвать у пользователя непонимание. Программа не

должна аварийно прекращать свою работу.

При разработке информационной системы следует учитывать различные варианты и случаи ошибок, которые могут возникнуть лишь в определенных случаях. Такие проблемы должны быть выявлены в результате тестирования и исправлены разработчиками, также возможны варианты, в которых исправление проблемы невозможно по причине отсутствия необходимого отображения на экране. В таких случаях обычно применяют заглушки, которые способны остановить выполнение этого случая.

4) Блокировка данных. В многопользовательской ИС очень важно следить за

конфликтами доступа к данным различных пользователей. Если пользователь начал

редактировать объект, необходимо не допустить к этому объекту других пользователей.

Блокировка данных и запрет на свободный доступ к данным является крайне необходимым условием в работе информационной системы. Это может быть связано не только с конфиденциальностью данных, а также и с враждебными намерениями конкурентов владельца информационной системы.

Ссылка на git репозиторий: https://github.com/Ahtem707/AutoOffice