**2 СИСТЕМНОЕ ПРОЕКТИРОВАНИЕ**

В результате изучения требований к разрабатываемой системе перейдем к разбиению системы на функциональные блоки (модули). Данный подход позволяет создавать более гибкую архитектуру приложения, что позволяет модернизировать существующие блоки и создавать новые без внесения значительных изменений в общую схему работы всей системы целиком.

В разрабатываемом настольном приложении выделяются следующие общие блоки:

* блок авторизации пользователей;
* блок реляционной базы данных.

Далее, в зависимости от роли, полученной после взаимодействием с блоком авторизации, выделяют разные блоки. При получении роли «преподаватель» выделяют следующие блоки для взаимодействия с ними:

* блок меню для взаимодействия с пользователем;
* блок выбора учащегося для выставления отметки;
* блок выставления отметки учащемуся;
* блок выбора объема отображаемой информации об учащихся;
* блок выбора конкретного учащегося и отображения информации о нем для роли «преподаватель»;
* блок выбора группы учащихся и отображения информации них.

При получении роли «родитель», в дополнение к общим блокам, выделяют следующий блок для взаимодействия с ними:

* блок отображения информации о конкретном учащемся для роли «родитель».

При получении роли «учащийся», в дополнение к общим блокам, выделяют следующий блок для взаимодействия с ними:

* блок отображения информации о конкретном учащемся для роли «учащийся».

Структурная схема, иллюстрирующая перечисленные блоки и связи между ними приведена на чертеже ГУИР.400201.XXX C1.

Каждый из представленных выше модулей необходим для выполнения определенной заранее задачи, однако ничего не мешает масштабировать их в дальнейшем. Чтобы система могла функционировать полноценно, каждый модуль взаимодействует друг с другом передавая идентификационные данные пользователя (имя пользователя).

Рассмотрим детальнее функциональные блоки приложения.

*Блок реляционной базы данных* включает данные, используемые настольным приложением во время его использования. Для реализации данного блока использовалась база данных PostgreSQL. Она, как и любая другая база данных, позволяет хранить и обрабатывать информацию в структурированном виде. Структуризация происходит благодаря таблицам и связям между ними. В данном проекте используются два вида связей:

* многие ко многим. Пример из проекта – «У учащегося может быть несколько преподавателей. У преподавателя может быть несколько учащихся». Для реализации данной связи используется «посредник» (дополнительная таблица) между двумя рассматриваемыми таблицами. Он хранит два внешних ключа, первый из которых ссылается на первую таблицу, а второй - на вторую;
* один к одному. Пример из проекта – «Один учащийся может принадлежать лишь одной учебной группе. Однако одна учебная группа может включать в себя несколько учеников». При реализации данного вида связи также необходим «посредник». Он хранит лишь один ключ, который ссылается на таблицу со множественной связью.

Также можно отметить факт того, что PostgreSQL управляет доступом при помощи так ролей. Роли могут быть членами других ролей, что позволяет им наследовать параметры привилегий, определённых ранее ролей. Что будет использовано при разработке приложения и описано позже.

*Блок авторизации* пользователей является модулем, необходимым для обеспечения безопасности при включении приложения, разграничения прав пользователей, защиты от несанкционированного доступа в личным данным. Главная возможность данного блока – авторизация пользователя, без передачи личных данных, кроме индивидуального номера в приложении. Этот блок взаимодействует со следующими блоками:

* реляционной базы данных;
* взаимодействия с пользователем;
* отображения информации о конкретном учащемся для роли «учащийся»;
* отображения информации о конкретном учащемся для роли «родитель».

В дальнейшем, при масштабировании, планируется добавление варианта регистрации новых пользователей в приложении, используя описанный ранее блок.

*Блок меню для взаимодействия с пользователем* доступен пользователям, авторизовавшимся с ролью «преподаватель». Основная его задача – предоставить интуитивно понятный выбор дальнейшего действия посредством интерфейса (выставление, просмотр отметок; просмотр контактной информации). Взаимодействует данный блок со следующими блоками:

* блок авторизации;
* блок выбора объема отображаемой информации об учащемся;
* блок выбора учащегося для выставления отметки;
* блок выставления отметки учащемуся.

*Блок выбора учащегося для выставления отметки* также доступен пользователям с ролью «преподаватель». Он необходим для конкретизации учащегося, которому в дальнейшем будет выставлена отметка и, по желанию, комментарий к ней. Данный блок напрямую взаимодействует с блоками реляционной базы данных, для понимания существует студент, которому намереваются выставить отметку, и блоком авторизации, для понимания кто именно хочет выставить отметку. Взаимодействует описанный выше блок со следующими блоками:

* реляционной базы данных;
* выставления отметки учащемуся;
* меню взаимодействия с пользователем.

*Блок выставления отметки учащемуся* является логическим продолжением блока выбора учащегося для выставления отметки. Он позволяет завершить процесс выставления отметки учащемуся путем ввода значения в соответствующее поле. В данном блоке также производится проверка: имеет ли возможность ранее авторизованный «преподаватель» выставлять отметки указанному в прошлом блоке учащемуся. Производится это в данном блоке сугубо из соображений безопасности и производительности: при выставлении отметок придется делать обращение к этим переменным и будет не целесообразно получать их новыми запросами к базе данных или передавать как параметры одного блока другому. Этот блок взаимодействует с такими блоками как:

* реляционной базы данных;
* выбора учащегося для выставления отметки;
* меню для взаимодействия с пользователем.

*Блок выбора объема отображаемой информации об учащихся* доступен сугубо пользователям с ролью «преподаватель». Используется в целях упрощения интерфейса для конечного пользователя, с целью понимания последним какую информацию необходимо получить: о конкретном студенте или их группе. Взаимодействие данного блока ограничивается связью со следующими блоками:

* реляционной базы данных;
* меню взаимодействия с пользователем;
* выбора группы учащихся и отображения информации о них
* выбора конкретного учащегося и отображения информации о нем для роли «преподаватель».

*Блок выбора конкретного учащегося и отображения информации о нем для роли «преподаватель* является продолжением блока выбора объема отображаемой информации и также доступен лишь «преподавателям». Он предназначен для выбора конкретного обучающегося и дальнейшей визуализации основной информации о нем:

* отметок;
* комментариев;
* контактных данных «родителей» выбранного учащегося;

Этот блок связан с такими блоками как:

* реляционной базы данных;
* блок выбора объема отображаемой информации об учащихся.

*Блок выбора группы учащихся и отображения информации них* также рассматривается как продолжение блока выбора объема отображаемой информации и доступен для пользователей с ролью «преподаватель». Он используется для:

* отображение краткой информации об обучающихся во всех группах (при условии, что конкретная группа учащихся (класс) не указана);
* определенной группы (класса), при конкретном вводе.

Вводимая информация проверяется на наличие запрашиваемых данных в базе данных. Взаимодействие этого блока происходит со следующими блоками:

* реляционной базы данных;
* выбора объема отображаемой информации об учащихся;
* меню взаимодействия с пользователем.

*Блок отображения информации о конкретном учащемся для роли «родитель»* является единственным «дополнительным» блоком для пользователей с ролью «родитель». Он отвечает сугубо за корректность отображения информации о их «ребенке»:

* классе, в котором обучается ребенок;
* комментариях к учащемуся от преподавателей;
* контактной информации об учреждении.

*Блок отображения информации о конкретном учащемся для роли «учащийся»* точно как и предыдущий является уникальным: доступен пользователям лишь с ролью «учащийся». Его функционал ограничен отображением сугубо отметок об авторизованном «ученике».

Последние два блока взаимодействуют лишь с блоком реляционной базы данных.