

## **Лабораторная работа №4**

# **РЕАЛИЗАЦИЯ АРХИТЕКТУРЫ ИНФОРМАЦИОННОЙ СИСТЕМЫ**

**Цель:** описать функциональную структуру программной реализации, модули, сценарии и использованные библиотеки.

### **Ход работы**

Для создания исполняемых модулей сайта использованы средства языка программирования PHP. PHP (рекурсивный акроним словосочетания PHP: Hypertext Preprocessor) - это распространенный язык программирования общего назначения с открытым исходным кодом. PHP специально сконструирован для веб-разработок и его код может внедряться непосредственно в HTML.]

Сайт содержит несколько программных модулей.

#### **Форма регистрации и Форма авторизации**

Модуль предназначен для регистрации новых пользователей и последующей авторизации для входа на закрытую часть сайта. По уникальному логину и паролю происходит идентификация пользователя.

В исходном файле form\_register.php содержатся блок вывода ошибок и форма регистрации, состоящая из текстовых полей: name, email, password и кнопки btn\_submit\_register. Обработка данных из формы происходит в файле register.php. В этом файле происходит запуск сессии с помощью функции session\_start(), подключение к файлу с вызовом базы данных с помощью require\_once(). Далее объявляются переменные для хранения сообщений об ошибках и успехах операции. После проверки нажатия кнопки отправки происходит обработка данных полученных с формы: удаляются лишние пробелы, специальные символы, шифруется пароль. Если какое-либо из полей не было заполнено, то выводится сообщение об ошибке и открывается форма регистрации. Происходит проверка на наличие пользователя с таким email в

базе данных. Новый пользователь добавляется в базу данных и происходит перенаправление на страницу авторизации.

В исходном файле `form_auth.php` содержит окно для вывода ошибок и форма авторизации, состоящая из текстовых строк `email` и `password` и кнопки `btn_submit_auth`. Данные с формы отправляются на обработку в файл `auth.php`. В этом файле происходит запуск сессии с помощью функции `session_start()`, подключение к файлу с вызовом базы данных с помощью `require_once()`. Далее объявляются переменные для хранения сообщений об ошибках и успехах операции. После проверки нажатия кнопки отправки происходит обработка данных полученных с формы: удаляются лишние пробелы, специальные символы, шифруется пароль. Если какое-либо из полей не было заполнено, то выводится сообщение об ошибке и открывается форма авторизации. Данные с формы проходят проверку на валидность, в случае неправильно заполненных или незаполненных строк на странице авторизации появляется сообщение об ошибке. Выполняется запрос к таблице пользователей в базе данных, если результат запроса содержит одну запись, т.е. пользователь с такими данными существует, то в массив сессий сохраняется логин и пароль. Далее проверяется роль пользователя. Если это администратор, то происходит переход на панель администрации сайта, если обычный пользователь, то на главную страницу. Во время этой проверки в массив сессий указываются роль пользователя и идентификатор пользователя в базе данных.

На Рисунки 4.1 и 4.2 представлены алгоритмы работы модуля авторизации и регистрации

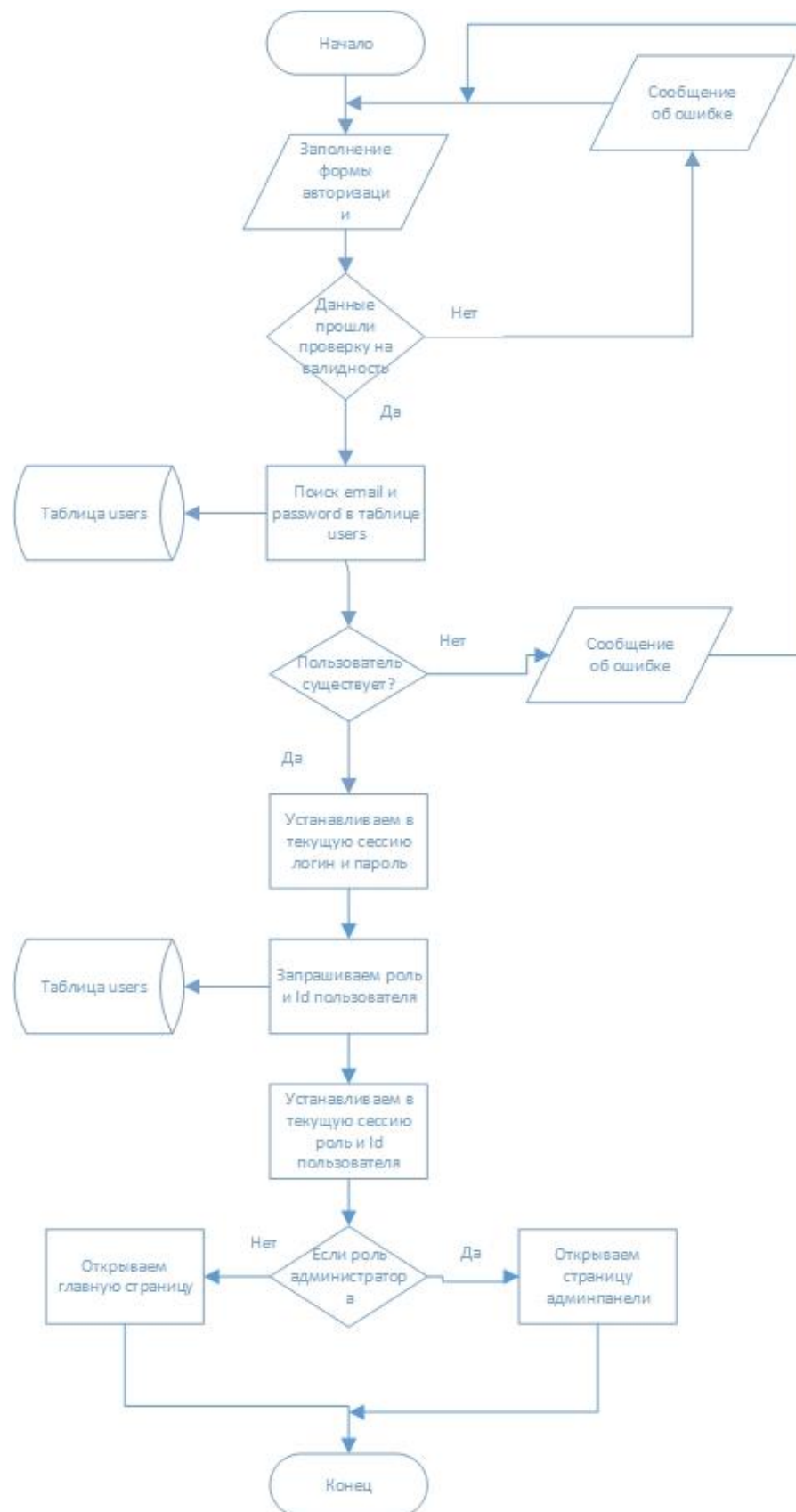


Рисунок 4.1 – Алгоритм авторизации

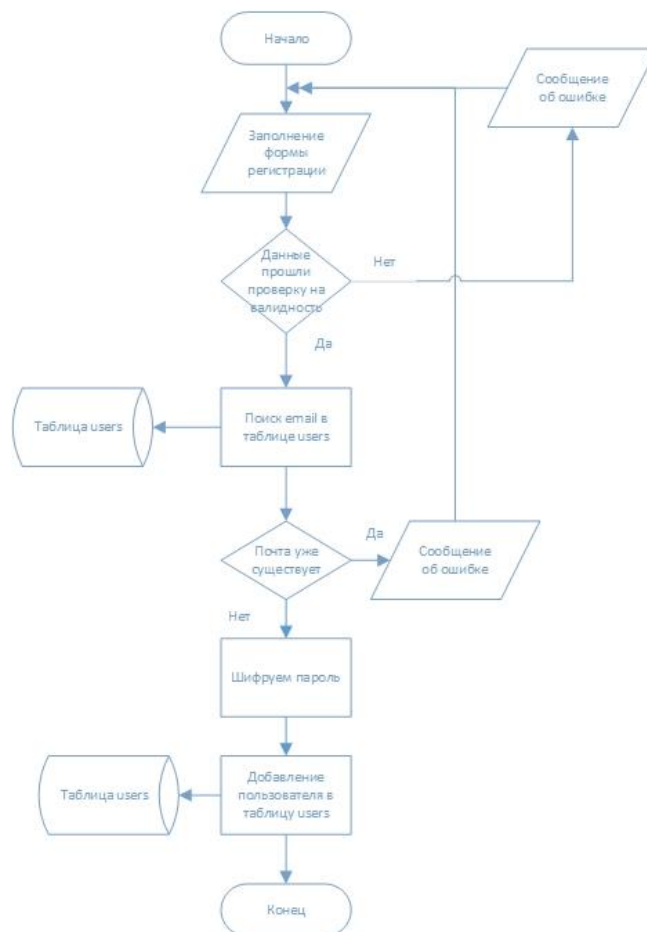


Рисунок 4.2 – Алгоритм регистрации

#### Форма подачи обращения

Форма предназначена для подачи обращений авторизованных пользователей сайта в администрацию города. Пользователь может указать подразделение, в которое должно быть направлено обращение. Данные с формы обрабатываются в appointment.php. В нём происходит проверка на заполнение всех полей формы, обработка данных, создание запроса на вставку в таблицу переменных, содержащих данные с формы. На Рисунок 4.3 представлен алгоритм работы модуля обращений.

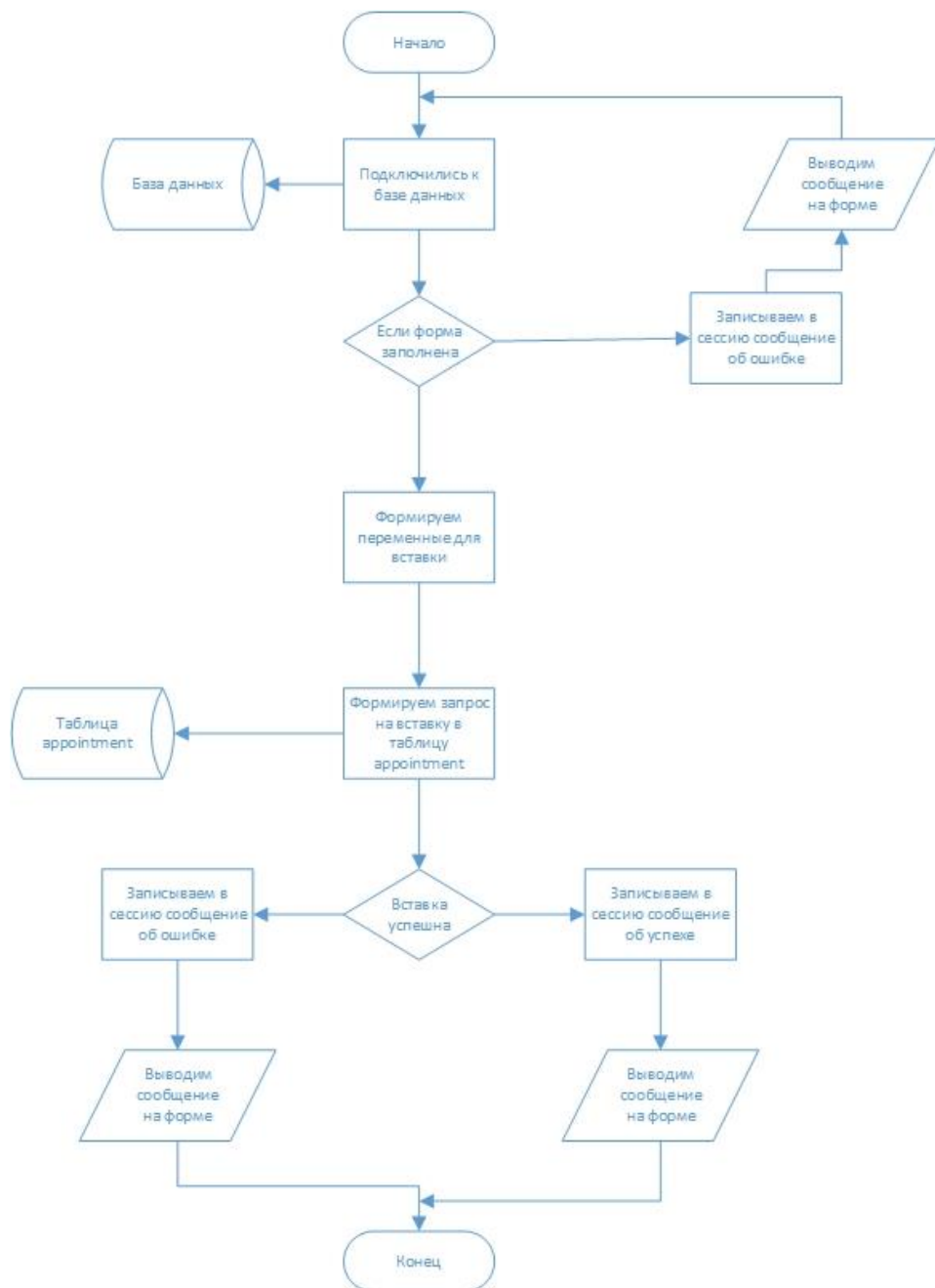


Рисунок 4.3 – Алгоритм подачи обращений

## Расписание

Модуль формирует и выводит на страницу сайта расписание отдела.

В исходном файле `timetable.php` подключается база данных. В зависимости от значения переменной `act`, устанавливаемой методом `GET`, определяется содержимое страницы. Если `act` установлено в `home`, то на странице выводится список подразделений города. В названия отделов и структур включены ссылки, открывающие расписание работы, а значение `act` устанавливается в `view`. Расписание выводится в виде таблицы. Также

выводится ссылка возврата на страницу со списком расписаний, которая возвращает act значение home. На Рисунок 4.4 представлен алгоритм работы модуля расписания.

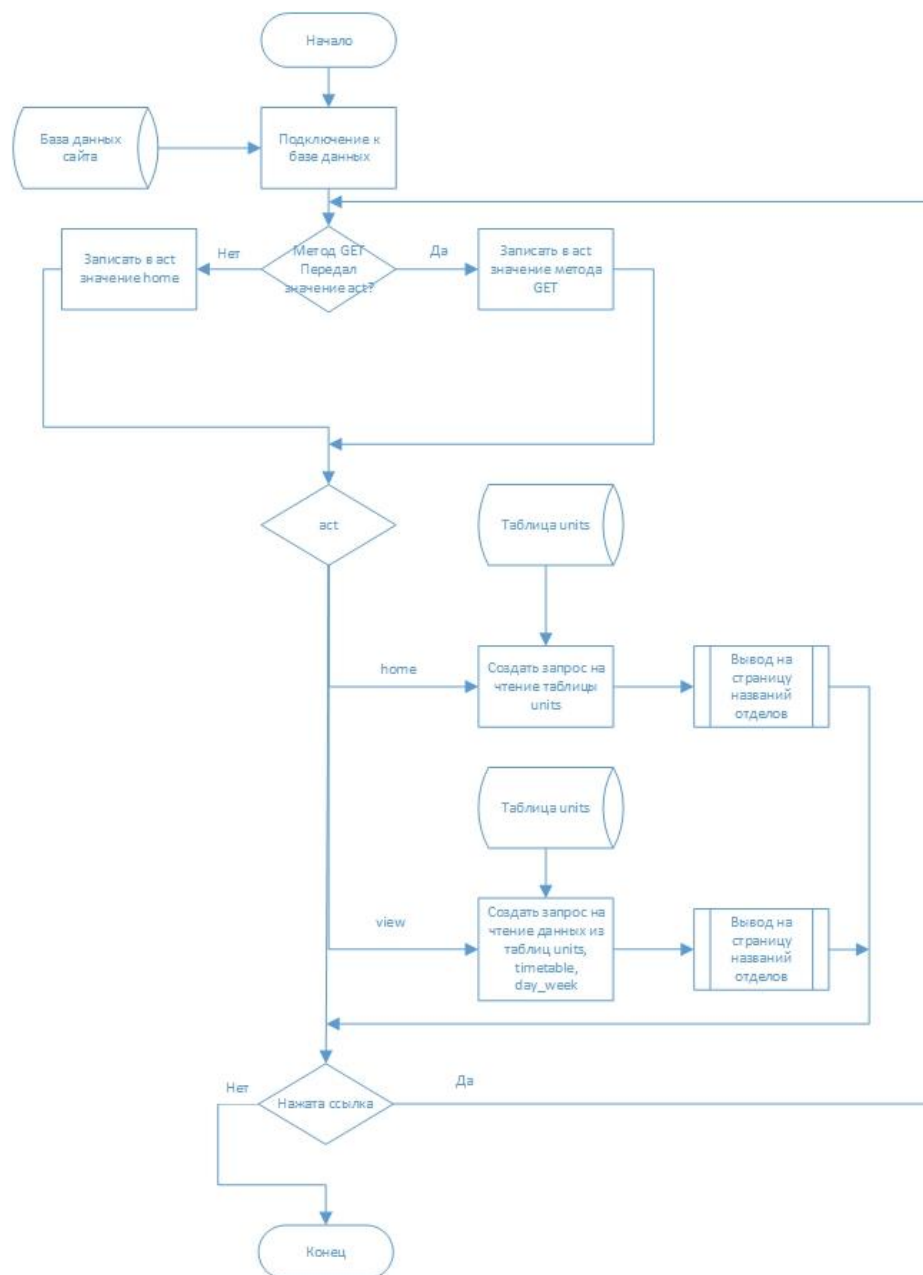


Рисунок 4.4 – Алгоритм расписания

#### Новостной блок на главной странице

Выводит анонсы новостей на главную страницу. В файле index.php происходит подключение к базе данных с помощью require\_once(), затем выполняется запрос на получение двух новостей из базы данных, которые и выводятся в новостном блоке.

#### Блок объявлений на главной странице

Публикация объявлений о событиях и плановых ремонтных работах в городе. Выполняется запрос к таблице объявлений в базе данных на вывод пяти последних объявлений.

Блок подразделений на главной странице

Вывод названий подразделений города с возможностью перехода на страницу подразделений. Выполняется запрос к таблице подразделений на вывод названий отделов из базы данных.

Новостной раздел

Модуль предназначен для публикации новостей на сайте. Позволяет разделить новости по направлениям. Новость можно просмотреть в виде анонса, а можно перейти по ссылке в заголовке новости, прочитать новость целиком.

В исходном файле `news.php` происходит подключение к базе данных. После этого проверяется значение `act`, устанавливаемого с помощью метода GET, если установлено значение `home`, то на странице выводятся новости и панель переключения страниц. Для этого происходит запрос в базу данных и результаты выводятся на экран. При нажатии на название новости переменной `act` присваивается значение `read_news` и на экран выводится подробная информация о новости, а также ссылка «Вернуться». При нажатии на нее переменной `act` передается значение `home`, и происходит возврат на страницу со всеми новостями. Номера страниц выводятся с помощью функции `link_bar`. На Рисунок 4.5 представлен алгоритм работы модуля новостей.

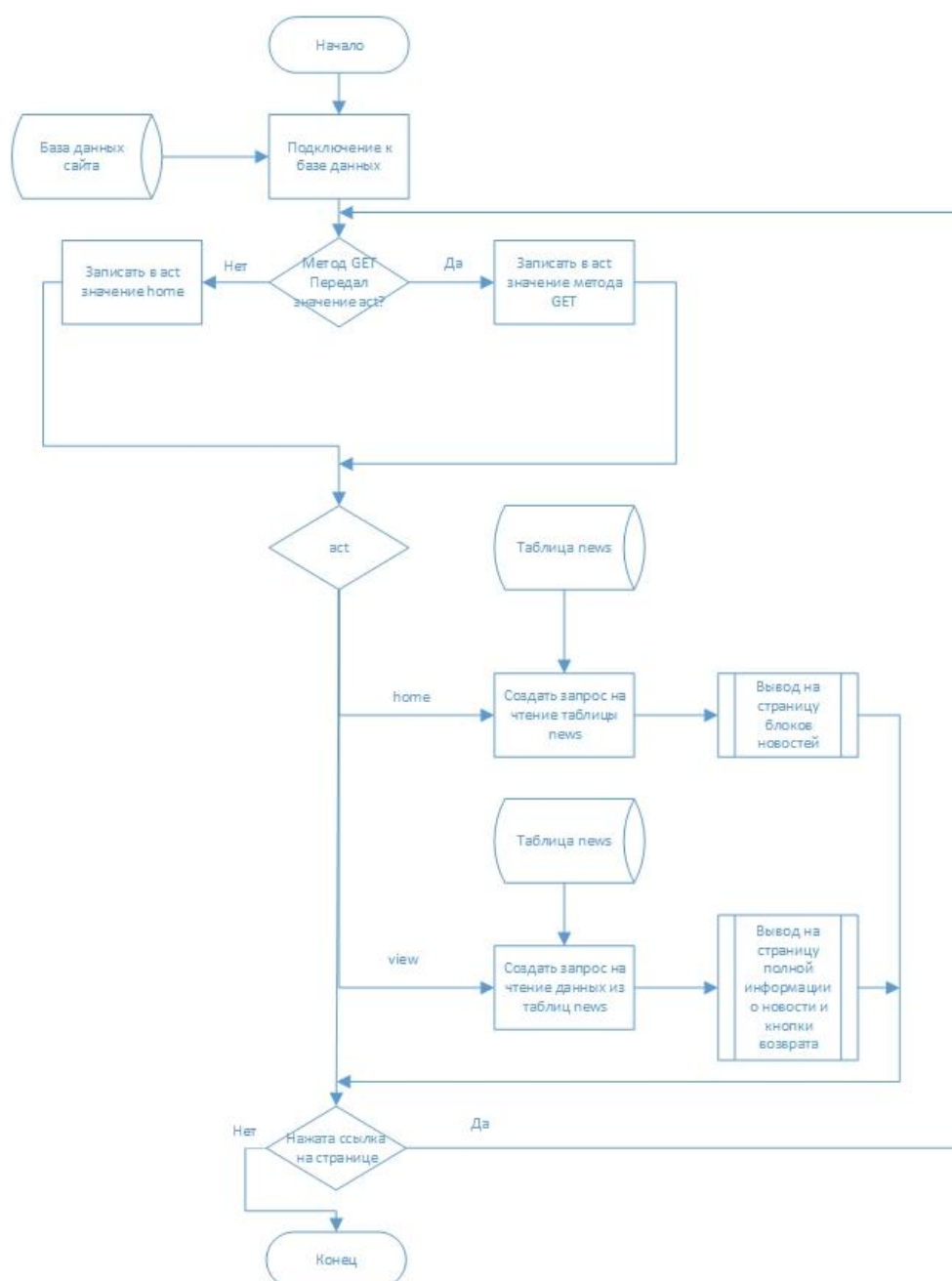


Рисунок 4.5 – Алгоритм работы новостного раздела

### Раздел документов

Модуль предназначен для публикации документов. Позволяет разделить документы по направлениям. Возможен как краткий просмотр документа, так его полный текст при переходе по ссылке в заголовке документа.

В исходном файле documents.php происходит подключение к базе данных. После этого проверяется значение act, устанавливаемого с помощью метода GET, если установлено значение home, то на странице выводятся документы и панель переключения страниц. Для этого происходит запрос в



базу данных и результаты выводятся на экран. При нажатии на название документа переменной `act` присваивается значение `read_documents` и на экран выводится подробная информация о документе, а также ссылка «Вернуться». При нажатии на нее переменной `act` передается значение `home`, и происходит возврат на страницу со всеми документами. Номера страниц выводятся с помощью функции `link_bar`. На Рисунок 4.6 представлен алгоритм работы модуля документов.

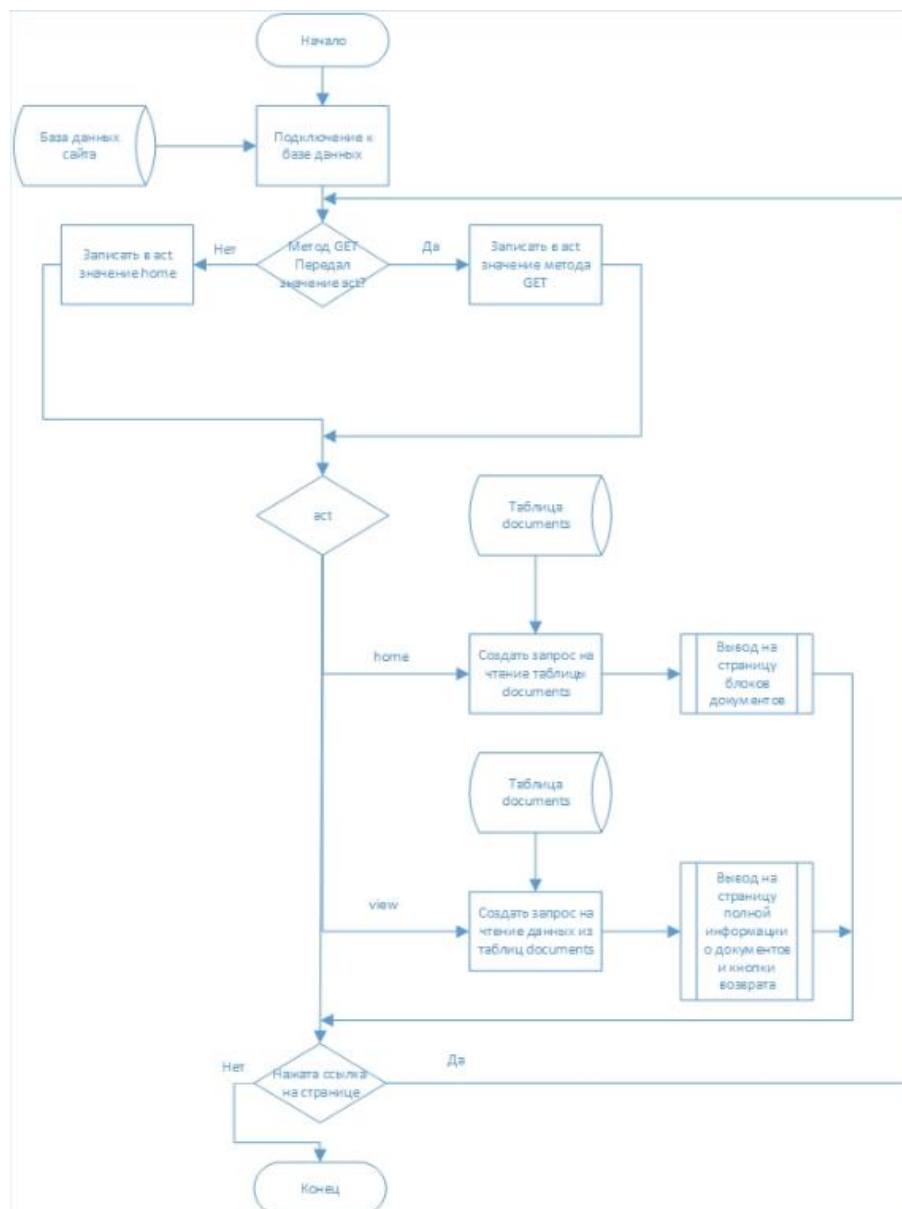


Рисунок 4.6 – Алгоритм работы раздела документов

Панель администрирования сайта

Предназначена для управления содержимым сайта. Позволяет управлять данными в таблицах новостей, документов, обращений, объявлений, пользователей.

В начале исполняемого файла `adminpanel.php` происходит проверка роли пользователя (если не администратор, то перенаправляется на страницу авторизации). Происходит подключение к базе данных. Далее устанавливается значение переменной `act` с помощью метода `GET`. Эта переменная определяет содержимое страницы.

Если значение `act` установлено в `home`, то странице отображаются пять таблиц из базы данных и две строки с количеством документов и новостей в базе данных. Таблицы с данными формируются следующим образом: делается запрос на выбор всех данных из одной таблицы базы данных, если запрос успешный, результат формируется в массив, который в цикле выводится на экран. В том же цикле к полю названия добавляется ссылка на переход в окно редактирования (значение `act` устанавливается в `edit_@название_таблицы`), ссылка на удаление записи из таблицы (значение `act` устанавливается в `delete_@название_таблицы`), внизу таблицы выводится ссылка на добавление новой записи (значение `act` устанавливается в `add_@название_таблицы`).

Если значение `act` установлено в `edit_@название_таблицы`, то в переменную `id` с помощью метода `GET` записывается `id` обрабатываемой записи. Далее выполняется запрос на чтение данных из таблицы, если он успешен, то результат записывается в массив и данные выводятся на форму. Если данные на форме изменились, то формируется запрос на обновление данных, если он успешен, то в формы записываются новые значения полей. Также на экран выводится ссылка на возврат на главную страницу админпанели, по ней в `act` устанавливается значение `home`.

Если значение `act` установлено в `add_@название_таблицы`, то на экран выводится форма отправки данных. С помощью метода данные отправляются из формы и записываются в соответствующие переменные. Формируется запрос на вставку данных в таблицу, если он успешен, то выводится

сообщение об этом. Также на экран выводится ссылка на возврат на главную страницу админпанели, по ней в аст устанавливается значение home.

Если значение аст установлено в delete\_@название\_таблицы, то в переменную id с помощью метода GET записывается id обрабатываемой записи. Далее формируется запрос на удаление записи с соответствующим id из таблицы базы данных, в случае успеха выводится сообщение об этом. Также на экран выводится ссылка на возврат на главную страницу админпанели, по ней в аст устанавливается значение home. На Рисунок 4.7 представлен алгоритм работы панели администрации.

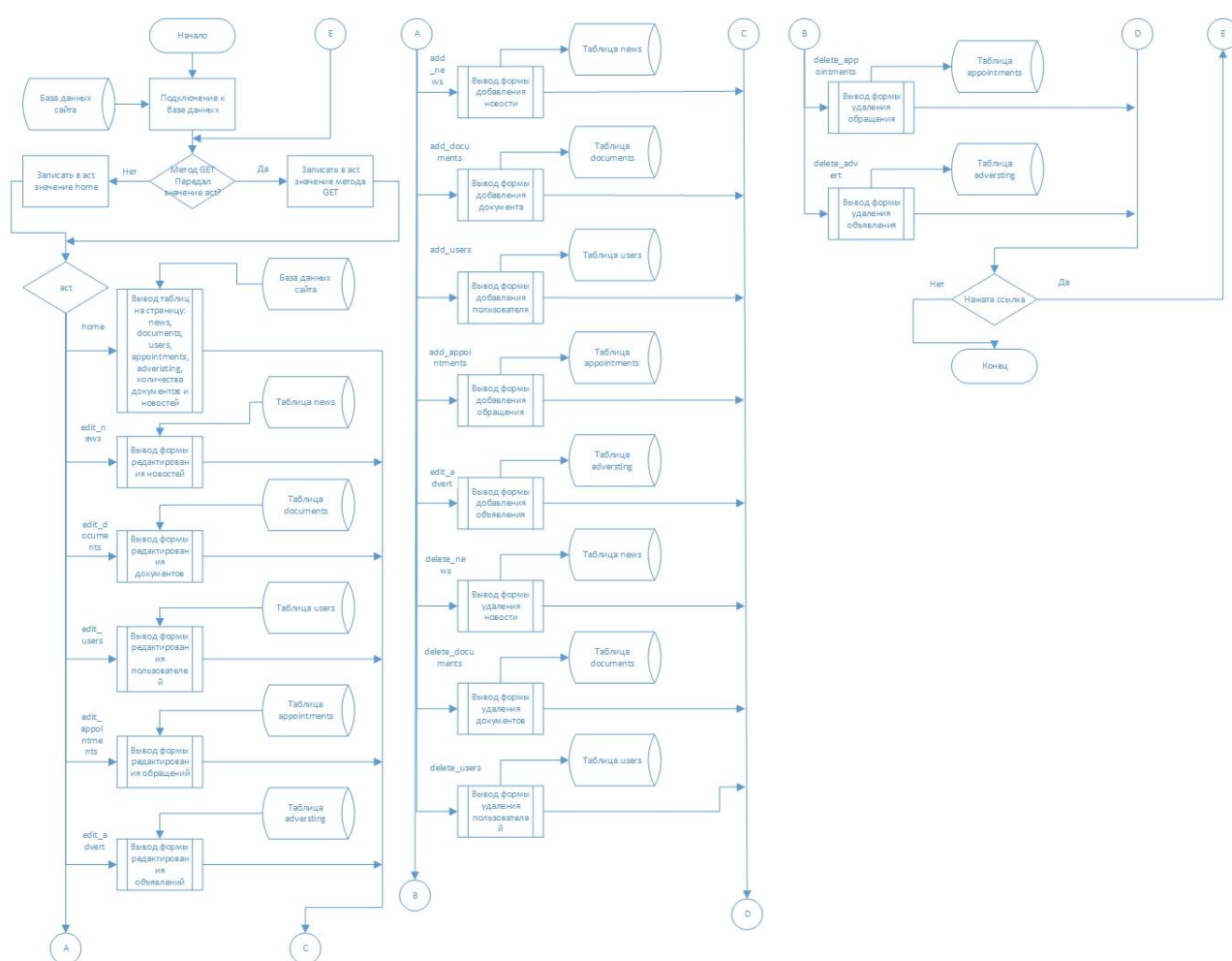


Рисунок 4.7 – Алгоритм работы панели администрации

### Карта сайта

Без вмешательства администратора сайта, формирует информационное дерево сайта, и выводит его в нужном месте в виде раскрытого вплоть до последнего уровня дерева ссылок. Карта сайта предназначена для упрощения

навигации по сайту и для наиболее полного индексирования сайта роботами поисковых машин.

### Форма поиска

Предназначена для быстрого поиска текстовых материалов, касаемых новостей и документов. Поиск осуществляется по документам, новостям. Также доступен поиск по названию, теме, направлению, дате и содержанию.

В исходном файле `search.php` подключается база данных, формируется форма поиска, и исполняемый блок. В `php`-блоке если поисковый запрос был выполнен из строки поиска в навигационном меню, то формируется запрос на поиск по двум таблицам. Если запрос не дал результатов выводится соответствующее сообщение. Если запрос был отправлен из формы поиска на странице, то в соответствии с отправленными данными формируется запрос и его результаты выводятся на экран. В противном случае, выводится сообщение об отсутствии данных. Алгоритм модуля на Рисунок 4.8.

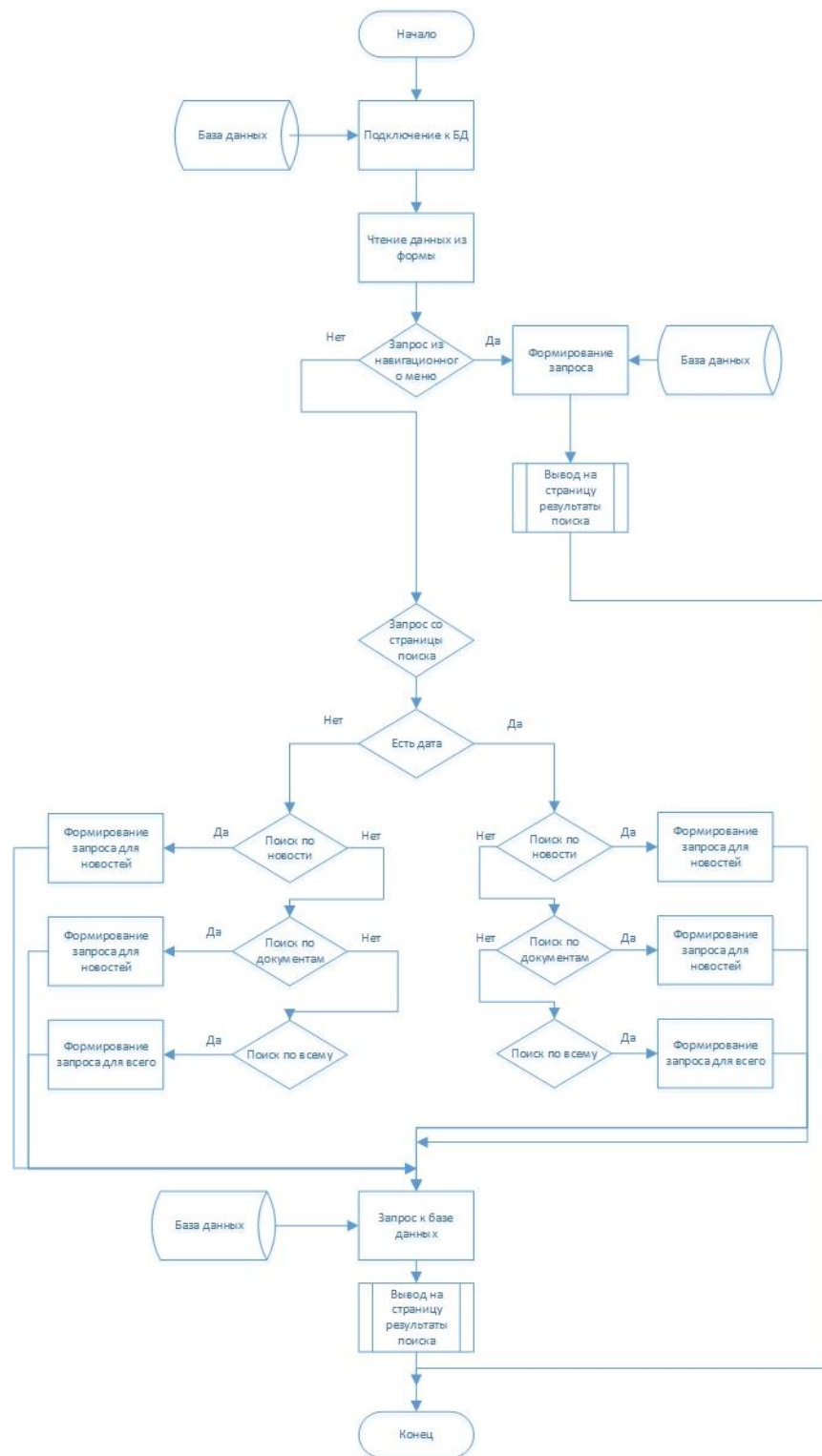


Рисунок 4.8 Алгоритм поиска

Валидация данных.

Модуль проверяет правильность заполнения форм на некоторых страницах сайта.

На странице подачи обращений `form_appointment.php` при незаполненных полях Фамилия, Имя, Отчество, Телефон и e-mail средствами

HTML появляется сообщение о неверном формате введенных данных. При отсутствии введенных данных о Подразделении, способе отправки и подтверждении на отправку данных с помощью `java-script` функции `validate_form()` выводятся сообщения с предупреждениями, а значению события `onsubmit` присваивается `false`. Если поля формы заполнены верно, то данные отправляются на обработку в файл `appointment.php`.

На странице регистрации и авторизации (`form_register.php` и `form_auth.php`) валидация выполняется с помощью кода `java-script` в заголовке страницы. Для проверки электронной почты составлено регулярное выражение. Если введенный в поле формы адрес соответствует регулярному выражению, то кнопка «Отправить» становится рабочей, в противном случае блокируется. Поле пароля проверяется на длину, если меньше 6, то кнопка «Отправить» блокируется и то выводится сообщение об ошибке, если введена пустая строка, то выводится сообщение «Введите пароль».

## 4.2 Описание пользовательского интерфейса

Навигационное меню сайта должно обязательно содержать следующие отдельные пункты:

- главная (домашняя) страница;
- администрация города;
- новости; 4 – приемная;
- документы;
- поиск;
- глава администрации;
- структурные подразделения администрации;
- социальная сфера;
- городская сфера;
- расписание приема;
- подать обращение;

- устав города;
- регламент администрации;
- правила землепользования и застройки города;
- поиск по документам;
- поиск по новостям;
- карта сайта;
- авторизация;
- регистрация;
- страница администрирования сайта.

Если пользователь прошел авторизацию с ролью «Пользователь», то он получает доступ к пункту меню «Подать обращение». У администратора открыт доступ ко всем страницам сайта.

Навигационная схема представлена на Рисунке 4.9

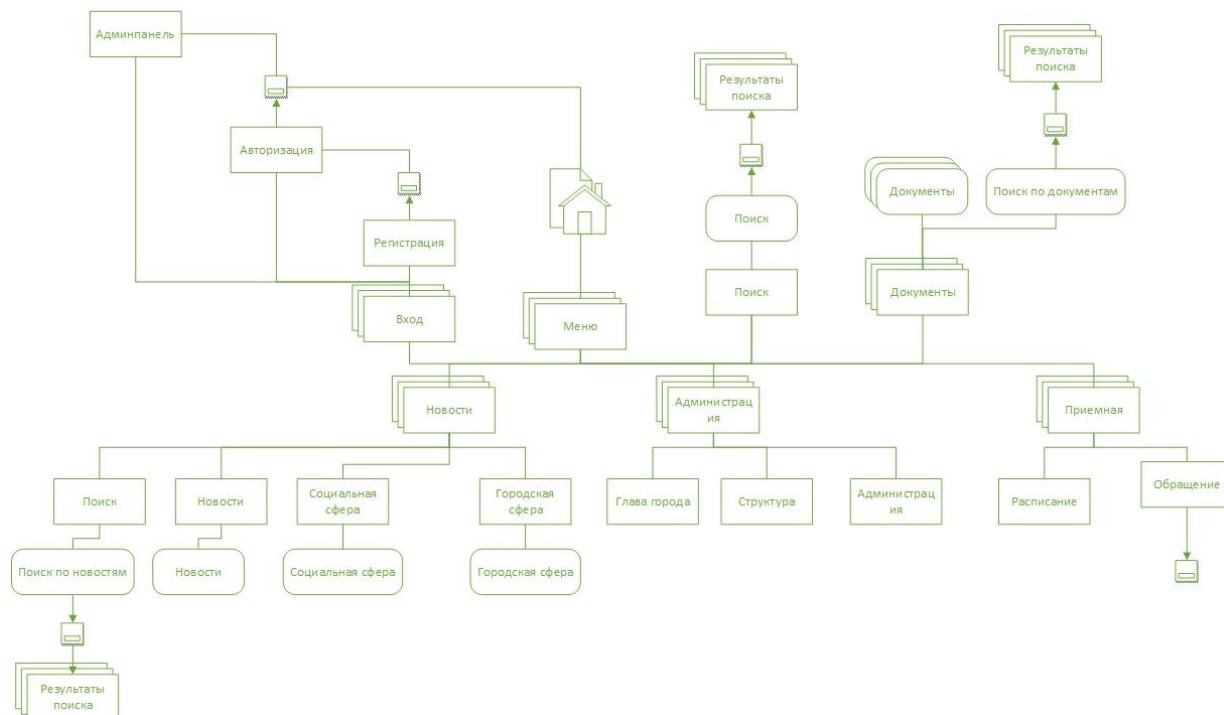


Рисунок 4.9 - Навигационная схема

## Вывод

Описали функциональную структуру программной реализации, модули, сценарии и использованные библиотеки.