

Technology Stack

When developing the “Accessible Kazakhstan” information system, the following tools and technologies were used:

- Symfony framework;
- PostgreSQL DBMS;
- Nuxt.js framework.

One of the popular PHP frameworks, Symfony, was used to develop the backend. This framework is well-suited to complex and loaded web projects. Key factors for choosing Symfony were:

- possibility of further development;
- flexibility and variety of functions;
- constant updates to improve security;
- high performance of the framework.

PHP 7.2.5 and additional extensions such as Ctype, iconv, JSON and Session were installed for the use of the Symfony framework.

It is the Symphony that helps webforms and sending forms via email function, as well as makes the database work simply and quickly. With the help of special features the appearance of a document that is presented to the user is formed.

Thanks to the Symfony framework, the website is written in a modular structure and only the necessary modules are used:

- Cache module - thanks to this module, the website loading process is accelerated. The Symfony cache component comes with a variety of adapters that are specifically designed for high performance.
- The Databases and the Doctrine ORM module is a tool that is used to maintain a database. This tool is supported by relational databases, in particular PostgreSQL, which is used in the project.
- Forms module creates and processes HTML forms, implements validation of submitted data and displaying of those forms in objects, etc.
- Module Creating and Sending Notifications - thanks to this module, users can receive notifications in their personal account. The Symfony Notifier component, in its turn, provides a dynamic way to manage the notification dispatch.
- Translations module - this module is used to translate and convert text on the site.
- Validation module - this module is responsible for validation. The data entered

into the forms is validated (checked) for correctness of the input. After the data is validated, it will be recorded in the database or transferred to the web service.

- Security module maintains system security.

For the development of the database, an object-relational database management system PostgreSQL, which is based on the SQL language, was chosen. The main advantages of this DBMS are:

- support for databases of unlimited size;
- powerful and reliable transaction and replication mechanisms;
- extensible system of embedded programming languages and support for loading c-compatible modules;
- inheritance;
- easy extensibility.

Functions in PostgreSQL are blocks of code that are executed on the server, not on the database client. They can be written in pure SQL; implementation of additional logic, for example, conditional jumps and loops, goes beyond the scope of SQL itself and requires the use of several language extensions. The functions are written using various programming languages. PostgreSQL allows the use of functions that return a set of records, which can then be used in the same way as the result of a regular query. Functions can be executed both with the rights of their creator and with the rights of the current user.

Triggers in PostgreSQL are defined as functions that are triggered by DML operations. For example, an INSERT operation can activate a trigger that checks whether an added record meets certain conditions. Triggers are associated with tables.

The PostgreSQL **rules engine** is a mechanism for creating custom handlers for not only DML operations, but also fetch operations. The main difference from the trigger mechanism is that the rules are triggered at the stage of parsing the request, before choosing the optimal execution plan and the execution process itself. Rules allow one to override the behavior of the system when performing an SQL operation on a table.

Indexes of the following types belong to PostgreSQL **indexes**: B-tree, hash, R-tree, GiST, GIN. New index types can be created when necessary, although it is not a trivial process.

Multiversion is supported in PostgreSQL - it is possible for several users to simultaneously modify the database by using the Multiversion Concurrency Control (MVCC) mechanism. This allows ACID compliance and virtually eliminates the need for read locks.

Extending PostgreSQL for one's own needs is possible in almost any aspect. It is possible to add your own type conversions, data types, domains (user-defined types with initially imposed restrictions), functions (including aggregate ones), indexes, operators (including overriding existing ones) and procedural languages.

Inheritance in PostgreSQL is implemented at the table level. Tables can inherit characteristics and field sets from other tables (parent). In this case, the data added to the generated table will automatically participate (if it is not specified separately) in queries to the parent table.

For development of the frontend, the Nuxt.js framework, which was designed to create universal applications in Vue.js, was used. The advantages of this framework are:

- static rendering;
- seo optimization and indexing in the search engine;
- it includes 50 standard modules for integration with third-party tools;
- it supports automatic code splitting.

Nuxt.js solves the tasks required for global projects. First of all, this is server side rendering. In other words, applications based on Nuxt.js are fully indexed by search engines, since they are rendered on the server side, and ready-made HTML comes to the front. Data received through database queries are also displayed correctly in the page code. All of these queries are done through asyncData and async Fetch.

Nuxt.js works not only in SSR mode, but also as a SPA or a static site generator.

Preconfigured webpack and babel are integrated into Nuxt.js, which at the time of integration transform the site into correct and optimized js and css files.

In the Nuxt.js framework, there are folders for pages, assets, layouts, static, plugins, etc. Nuxt.js represents a generic project architecture to help simplify project development.

Nuxt.js has a built-in Vuex global state library. Through nuxtServerInit, at an early stage of loading the application, a request is made to the server and the response is stored in the store. When this data is output to the front, it will be displayed in the code, which contributes to better indexing by search engines.

Browser and device compatibility

Browsers

Information system “Accessible Kazakhstan” - doskaz.kz - works with the

following browsers:

- Google Chrome (55 and later versions)
- Mozilla Firefox (55 and later versions)
- Microsoft Edge * (15 and later versions)
- Opera (40 and later versions)
- Apple Safari (12 and later versions)

* Most features are not available in legacy versions of Microsoft Edge.

We recommend using the latest version of the browser. Internet Explorer is no longer supported.

Adaptive

Information system “Accessible Kazakhstan” - doskaz.kz - is adapted for extensions below the web version. The system can be used on various tablets and mobile devices.

Технологический стек

При разработке ИС “Доступный Казахстан” были использованы следующие инструменты и технологии:

- Фреймворк Symfony
- СУБД PostgreSQL
- Фреймворк Nuxt.js

Для разработки бэкенда был использован один из популярных PHP-фреймворков - Symfony. Данный фреймворк идеально подходит для сложных и нагруженных веб-проектов. Ключевыми факторами выбора Symfony были:

- возможность дальнейшего развития;
- гибкость и разнообразность функций;
- постоянные обновления для повышения безопасности;
- высокая производительность фреймворка.

Для использования фреймворка Symfony была установлена версия PHP 7.2.5 и дополнительные расширения Ctype, iconv, JSON и Session.

С помощью Symfony функционируют веб-формы, просто и быстро работает база данных, отправка форм с email почтой, с помощью специальных функций формируется внешний вид документа, который представлен пользователю и др.

Благодаря фреймворку Symfony сайт написан по модульной структуре и использованы только нужные модули:

- Модуль Cache - благодаря данному модулю ускорен процесс загрузки сайта. Компонент Symfony cache поставляется с множеством адаптеров, которые специально разработаны для высокой производительности.
- Модуль Databases and the Doctrine ORM представляет инструмент, которые используется для ведения базы данных. Данный инструмент поддерживают реляционные базы данных, в частности PostgreSQL, которая и используется в проекте
- Модуль Forms - создает и обрабатывает HTML формы, реализует валидацию отправленных данных, отображение данных форм в объектах и др.

- Модуль Creating and Sending Notifications - благодаря данному модулю разработано получение уведомлений пользователями в личный кабинет. А компонент Symfony Notifier предоставляет динамический способ управления данной отправкой уведомлений.
- Модуль Translations - с помощью данного модуля переводится и конвертируется текст на сайте.
- Модуль Validation - данный модуль отвечает за валидацию. Данные, вводимые в формы, проходят валидацию (проверку) на корректность ввода. После того, как данные пройдут валидацию, они будут записаны в базу данных или же будут переданы веб-сервису.
- Модуль Security - поддерживает безопасность системы.

Для разработки базы данных была выбрана объектно-реляционная система управления базами данных. PostgreSQL, которая базируется на языке SQL. Основными преимуществами данной СУБД являются:

- поддержка бд неограниченного размера;
- мощные и надёжные механизмы транзакций и репликации;
- расширяемая система встроенных языков программирования и поддержка загрузки с-совместимых модулей;
- наследование;
- легкая расширяемость.

Функции в PostgreSQL являются блоками кода, исполняемыми на сервере, а не на клиенте БД. Они могут писаться на чистом SQL, реализация дополнительной логики, например, условных переходов и циклов, выходит за рамки собственно SQL и требует использования некоторых языковых расширений. Функции написаны с использованием различных языков программирования. PostgreSQL допускает использование функций, возвращающих набор записей, который далее можно использовать так же, как и результат выполнения обычного запроса. Функции могут выполняться как с правами их создателя, так и с правами текущего пользователя.

Триггеры в PostgreSQL определяются как функции, инициируемые DML-операциями. Например, операция INSERT может запускать триггер, проверяющий добавленную запись на соответствия определенным условиям. Триггеры ассоциируются с таблицами.

Механизм правил в PostgreSQL представляет собой механизм создания пользовательских обработчиков не только DML-операций, но и операции выборки. Основное отличие от механизма триггеров заключается в том, что правила срабатывают на этапе разбора запроса, до выбора оптимального плана выполнения и самого процесса выполнения. Правила позволяют переопределять поведение системы при выполнении SQL-операции к таблице.

Индексы в PostgreSQL следующих типов: B-дерево, хэш, R-дерево, GiST, GIN. При необходимости можно создавать новые типы индексов, хотя это далеко не тривиальный процесс.

Многоверсионность поддерживается в PostgreSQL — возможна одновременная модификация БД несколькими пользователями с помощью механизма Multiversion Concurrency Control (MVCC). Благодаря этому соблюдаются требования ACID, и практически отпадает нужда в блокировках чтения.

Расширение PostgreSQL для собственных нужд возможно практически в любом аспекте. Есть возможность добавлять собственные преобразования типов, типы данных, домены (пользовательские типы с изначально наложенными ограничениями), функции (включая агрегатные), индексы, операторы (включая переопределение уже существующих) и процедурные языки.

Наследование в PostgreSQL реализовано на уровне таблиц. Таблицы могут наследовать характеристики и наборы полей от других таблиц (родительских). При этом данные, добавленные в порождённую таблицу, автоматически будут участвовать (если это не указано отдельно) в запросах к родительской таблице.

Для разработки фронтенда был использован фреймворк Nuxt.js, предназначенный для создания универсальных приложений на Vue.js. Плюсами данного фреймворка являются:

- статический рендеринг;
- seo-оптимизация и индексация в поисковой системе;
- в его состав входит 50 стандартных модулей для интеграции со сторонними инструментами;
- поддерживает автоматическое разделение кода.

Nuxt.js решает задачи, которые требуются для глобальных проектов. В первую очередь, это server side rendering. То есть приложения на Nuxt.js полноценно индексируются поисковыми системами, так как они рендерятся на стороне сервера и на фронт уже приходит готовый HTML. Данные, которые получены через запросы к базе данных также корректно отображаются в коде страницы. Все эти запросы выполняются через asyncData и async Fetch.

Nuxt.js работает не только в SSR режиме, но и как SPA или генератор статических сайтов.

В Nuxt.js интегрированы и предварительно настроенные webpack и babel, которые в момент сборки преобразуют сайт в корректные и оптимизированные js и css файлы.

В фреймворке Nuxt.js есть папки pages, assets, layouts, static, plugins и т. д. Nuxt.js представляет типовую архитектуру проектов, что помогает упростить разработку проекта.

В Nuxt.js встроена библиотека глобального состояния Vuex. Через nuxtServerInit на раннем этапе загрузки приложения делается запрос к серверу и ответ сохраняется в сторе. При выводе этих данных на фронт они будут отображаться в коде, что способствует улучшению индексации поисковыми системами.

Совместимость браузеров и устройств

Браузеры

Информационная система “Доступный Казахстан” - doskaz.kz - работает со следующими браузерами:

- Google Chrome (версия 55 и новее)
- Mozilla Firefox (версия 55 и выше)
- Microsoft Edge* (версии 15 и выше)
- Opera (версия 40 и новее)
- Apple Safari (версия 12 и новее)

*Большинство функций недоступно в устаревших версиях Microsoft Edge. Мы рекомендуем использовать новейшую версию вашего браузера. Internet Explorer более не поддерживается.

Адаптив

Информационная система “Доступный Казахстан” - doskaz.kz - адаптирована под расширения ниже веб-версии. Использовать систему можно на различных планшетных и мобильных устройствах.