##### С.В. Минаш (ГГТУ имени П.О. Сухого, Гомель)

Науч. рук. **Е.Г. Стародубцев,** доцент

**СРЕДСТВА ДЛЯ СОЗДАНИЯ ПРИЛОЖЕНИЯ «Автоматизированная информационная система сервисного центра по ремонту электроинструмента»**

Целью создания автоматизированных информационных систем управления является повышение эффективности производственно-хозяйственной деятельности предприятия за счет улучшения использования имеющихся ресурсов.

Повышение качества решения планово-экономических задач обусловлено следующими факторами:

– рационализация или оптимизация производственно-хозяйственных планов предприятия;

– оптимизация уровня запасов материальных ресурсов;

– оптимизация календарного планирования;

– ускорение процессов обработки данных.

Создаваемая автоматизированная информационная система должна, в первую очередь, решать традиционную задачу учёта оказываемых услуг сервисного центра, его клиентов и мастеров. При этом требуется производить учёт трудовой деятельности каждого сотрудника и получать данные о суммарной трудовой деятельности за месяц. Данная задача усложняется тем, что на данном предприятии каждый день сотрудник может выполнять разную работу в разном количестве, от чего будет зависеть его заработная плата.

Раньше для введения учёта деятельности организации использовались журналы и создавались огромные архивы для хранения информации. Для составления отчётов за месяц или другой промежуток времени требовалось нанимать соответствующих специалистов и для анализа данных требовались часы работы, а иногда даже дни, в зависимости от величины предприятия. Прогноз и анализ данных на крупных предприятиях был практически неосуществим или требовал огромного количества различных ресурсов.

С появлением вычислительных машин и с увеличением вычислительных возможностей, решение данных задач становится осуществимым и требует на порядок меньше ресурсов для их решения, как временных, так и человеческих.

Для реализации указанной функциональности бизнес-мониторинга используются различные информационные технологии и программные средства.

Для реализации *HTTP*-сервера был использован язык программирования *Kotlin* и *web*-фреймворк *Ktor*. В качестве системы управления базами данных (СУБД) использовалась *MySQL*. Для удобного взаимодействия с базой использовался паттерн *Exposable.*

Для разработки серверной части приложения был выбран язык программирования *Kotlin* так как он имеет ряд преимуществ перед другими языками программирования, а именно:

– *Kotlin* – это статически типизированный язык, который очень легко читать и писать;

– программы *Kotlin* не требуют точки с запятой в своей программе. Это делает код простым и более читабельным;

– этот язык позволяет осуществлять обмен и использование информации из *Java* различными способами. Более того, код *Java* и *Kotlin* могут сосуществовать в одном проекте;

– система типов *Kotlin* нацелена на исключение *NullPointerException* из кода;

– написание нового кода на *Kotlin* обычно занимает меньше времени. Еще проще развернуть код *kotlin* и поддерживать его в масштабе.

Так же у данного языка программирования есть несколько особенностей, которые делают его ещё более привлекательным для разработчиков:

– *Kotlin* – это язык программирования на основе объектно-ориентрированного программирования (ООП), где строка кода может быть урезана до 40%, что делает *Kotlin* идеальным выбором для разработки программного обеспечения или веб-разработки;

– *Kotlin* использует *Java Virtual Machine (JVM)* и сочетает в себе функции ООП и функционально-ориентированного программирования;

– код легко компилируется при работе с *Kotlin*, что приводит к повышению производительности для разработки под *Android*, а также объясняет, какой тип функции данных можно использовать во всем коде.

*Ktor* ­– это асинхронный веб-фреймворк, написанный и разработанный для *Kotlin* для создания упрощенных и легких веб-сервисов. Он полностью асинхронен благодаря использованию *coroutines* и как таковой должен очень хорошо масштабироваться с нагрузкой. Он все еще находится в активной разработке, поэтому может иметь некоторые шероховатости, но в целом он надёжен.

СУБД *MySQL* была выбрана исходя из её преимуществ по сравнению с остальными СУБД:

– многопоточность, поддержка нескольких одновременных запросов;

– оптимизация связей с присоединением многих данных за один проход;

– записи фиксированной и переменной длины;

– *ODBC* драйвер;

– гибкая система привилегий и паролей;

– гибкая поддержка форматов чисел, строк переменной длины и меток времени;

– скорость. Высокая производительность системы обеспечивается за счет упрощения некоторых используемых в ней стандартов;

– интерфейс с языками *C* и *Perl*, *PHP*;

– масштабируемость;

– совместимость с *ANSI SQL*;

– безопасность. Система изначально создана таким образом, что множество встроенных функций безопасности в ней работают по умолчанию;

– бесплатна в большинстве случаев;

– хорошая поддержка со стороны провайдеров услуг хостинга;

– быстрая поддержка транзакций через механизм *InnoDB*.

Клиентская часть приложения реализована с помощью языка программирования *JavaScript*, языка разметки *HTML* и *CSS*, которые позволяют создавать удобный и понятный интерфейс и используются для упрощения разработки клиентских приложений.

Для разработки клиентской части выбран язык программирования *JavaScript* так как он имеет ряд достоинств:

– незаменимость для веб-разработки. *JavaScript* поддерживается на всех операционных системах, во всех видах браузеров, и на настольных компьютерах, и на мобильных устройствах, полностью интегрирован в вёрстку страниц и серверной части;

– скорость работы и производительность. *JavaScript* позволяет частично обрабатывать веб-страницы на компьютерах пользователя без запросов к серверу. Это экономит время и трафик, снижает нагрузку на сервер;

– мощная инфраструктура (экосистема);

– простота и рациональность применения;

– удобство пользовательских интерфейсов. Заполнение форм, выбор действий, активация кнопок, проверки ввода, реагирование на наведение/клики мыши и так далее;

– гибкая работа с функциями. В *JavaScript* функции можно не только выполнять, но еще и возвращать функции из функций, передавать функции в качестве параметров другим функциями и присваивать функции в качестве значения переменных;

– взаимодействие с приложением может осуществляется даже через текстовые редакторы – *Microsoft Office* и *Open Office;*

– лёгкость освоения. Сначала код может показаться сложным, но его синтаксис и логика более понятные, чем у других языков программирования. Отображение действий также играет немалую роль в простоте освоения языка.