

времени и средств. Высококачественная автоматизация технологических процессов значительно облегчает работу в целом.

В представленной дипломной работе осуществляется разработка программного обеспечения для автоматизации табельного учета рабочего времени для ОАО «Рогачевский завод «Диaproектор». Данное программное обеспечение предназначено для работников подразделений (цехов) завода по ведению табельного учета использования рабочего времени.

Основной задачей является формирование табеля учета рабочего времени и вывод в Excel файл на основании введенных данных. Программное обеспечение позволит производить различные действия над введенными данными: редактировать уже имеющиеся записи, добавлять новые, сортировать по различным условиям, производить поиск по заданному критерию, удалять ненужные, выводить данные в Excel файл, а также вести учет отгулов, времени отработанного сверхурочно и в выходной день, печатать данные.

Пользовательский интерфейс разработан в стандартном виде (экранные формы, меню, выходные машинограммы), так, что не представляет трудностей для работы сотрудника любого уровня знаний, умений и навыков, у неподготовленного пользователя не возникнет проблем с взаимодействием с программой.

Основные операции ПО: возможность ввода, редактирования, удаления данных; просмотр данных посредством экранных форм; формирование отчетности в соответствии с требованиями и выводом их в Excel файл.

**Е.В. Леванцов** (УО «ГГУ имени Ф. Скорины», Гомель)  
Научн. рук. **В.Н. Леванцов**, старший преподаватель

## **ВНЕСЕНИЕ ИЗМЕНЕНИЙ В РАБОТУ ПРИЛОЖЕНИЯ ВО ВРЕМЯ ИСПОЛНЕНИЯ**

Все чаще при разработке в современных приложений встает вопрос о внесении изменений в уже скомпилированную и запущенную программу на сервере. Это связано с необходимостью незначительно дополнить функциональность какой-либо части приложения. Так же существует потребность в современном бизнесе гибко менять уже работающую программу, не прибегая к услугам программистов и не перезапуская сервер. Для данной проблемы в современной разработке программированного обеспечения имеются готовые решения.

В frontend разработке в первую очередь такими решениями являются динамически генерируемые web страницы. Ярчайшие примеры – php страницы, asp страницы, jsp и scala-html страницы. Основной плюс применения динамической генерации в том, что при помощи специального синтаксиса, а зачастую и прямых инъекций кода в страницу, можно менять логику статического html. Так же, несмотря на возможность инъекций кода, пересборки и перезапуска всего приложения не требуется, чаще всего достаточно обновить ресурсы web server, например, данная возможность присутствует в Apache Tomcat.

В рамках backend разработке существуют несколько подходов к решению данного вопроса. Наиболее популярным является использование скриптовых языков программирования. Они в большинстве своём просты в освоении при наличии документации и простейшие скрипты могут писаться даже непрограммистами. Многие современные платформы имеют собственные расширения, которые позволяют обеспечить понятный UI, что позволяет писать подобные инъекции функционала даже не зная скриптовых языков, а просто имея документации о интерфейсе. Второй подход для данного решения заключается всё также в создании пользовательского интерфейса, который будет создавать шаблон, который будет компилироваться в код прямо во время исполнения. Это поможет существенно расширить функционал не загромождая UI. Минус же данного подхода в необходимости компиляции, ограниченности функционала в ущерб очевидности, отсутствия возможности реализации многопоточного программирования вследствие существования определённого порядка исполнения кода, нарушения которого может привести к неправильному поведению.

**Е.В. Леванцов** (УО «ГГУ имени Ф. Скорины», Гомель)

Научн. рук. **В.Д. Левчук**, канд. техн. наук, доцент

## **ПРОБЛЕМЫ МНОГОПОТОЧНОГО ПРОГРАММИРОВАНИЯ В ЯЗЫКЕ JAVA**

В современных серверах в нагруженных системах часто установлено огромное количество многоядерных процессоров, чтобы справляться с той нагрузкой, которую пользователи оказывают на приложение. При работе в одном потоке выполнения огромная часть этой вычислительной мощности попросту не используется. В современных web серверах с помощью настроек можно сделать так, что один и тот же код может выполняться одновременно благодаря созданию «пула» экземпляров класса, обрабатывая запросы множества пользователей, колоссально