

обработкой результатов тестирования, помогают решить автоматизированные системы проверки знаний. Они значительно ускоряют получение результатов тестирования и предоставляют студенту широкие возможности для самостоятельного контроля знаний.

Разработанное приложение содержит тесты, охватывающие все темы, пройденные во время курса «Серверный язык программирования: PHP»: синтаксис языка PHP, типы данных, операторы, работа с массивами, функции для обработки строк, циклы и условные выражения и т. д. При каждом прохождении теста студенту генерируется список из 30 вопросов из разных тем, тест считается успешно пройденным, если пользователь ответил более чем на 75% вопросов в каждой теме. После прохождения теста результаты сохраняются в базе данных, поэтому при повторном тестировании пользователь может видеть свой прогресс.

Преподаватель, войдя в свою учетную запись, может редактировать тест, добавляя и изменяя вопросы и группируя их по темам. Также он имеет доступ к списку всех учащихся, зарегистрировавшихся в системе, и может просматривать их результаты тестирования, осуществлять фильтрацию по учебной группе и поиск по фамилии.

Для хранения вопросов использована документно-ориентированная система управления базами данных MongoDB, серверная часть приложения реализована с использованием программной платформы Node.js и фреймворка Express. Разработано REST API для выполнения всех необходимых операций с данными (добавление, удаление, редактирование и получение).

Клиентская сторона приложения разработана с помощью фреймворков Angular и AngularJS Material, TypeScript и CSS. Для облегчения работы пользователя реализован простой и интуитивно понятный интерфейс, который одинаково корректно отображается как на десктопных, так и на мобильных устройствах.

*Д. С. Гурский*

*Науч. рук. Н. Б. Осипенко,*

*канд. физ.-мат. наук, доцент*

## **РАЗРАБОТКА БИБЛИОТЕК ДЛЯ ИГРЫ DANGER SPACE**

В процессе разработки компьютерной игры Danger Space были созданы две библиотеки, предназначенные для распределения ответственностей по управлению физическими объектами и процессами PHYS\_OBJECT, а также формированию и управлению блоком меню DS\_MENU. Библиотеки разработаны на объектно-ориентированном языке программирования C++, обеспечивающем требуемое для создаваемой игры быстродействие. Разработка собственных библиотек вместо использования готовых решений позволяет полностью определить механизмы взаимодействия между элементами системы и достичь достаточной для симуляции космических процессов производительности.

Библиотека PHYS\_OBJECT позволяет создавать, обновлять и удалять физические объекты игры (звезды, планеты, астероиды и космические аппараты), а также объединять их в коллекции для упрощения использования. Данная библиотека предоставляет методы для взаимодействий типа: объект – объект, объект – коллекция, коллекция – коллекция. Взаимодействие между конкретными экземплярами объектов происходит по упрощенной модели закона всемирного тяготения. В библиотеке реализованы методы простых столкновений объектов: упругое столкновение, столкновение с удалением одного из двух или обоих объектов, столкновение с присоединением первого объекта ко второму или их слияния и появления нового объекта.

Библиотека DS\_MENU создана для отделения блока меню и упрощения его дальнейшего обновления. Она содержит в себе полный функционал меню и для работы требует следующие действия: вызов инициализации меню перед началом его использования; вызов функции итерации меню с передачей положения курсора и состояния некоторых кнопок

клавиатуры и мыши с целью отрисовки кадра меню в текущий контекст окна windows и последующего изменения буфера кадра; вызов функции освобождения ресурсов после завершения использования меню.

Библиотека DS\_MENU содержит структуру настроек и методы для работы с ними. Помимо вышеперечисленного эта библиотека использует и экспортирует классы для работы с шейдерными программами и шрифтами, перечисления событий меню для их интерпретации в главной функции приложения.

Бета-тестирование компьютерной игры Danger Space показало её хорошие функциональные качества и вызвало удовлетворенность тестировщиков, знакомых с играми подобного типа.

**П. Д. Загорюев**

*Науч. рук. Н. Б. Осипенко,*

*канд. физ.-мат. наук, доцент*

## **РЕКОМЕНДАТЕЛЬНАЯ СИСТЕМА ДЛЯ МЕДИЦИНСКОГО ЦЕНТРА**

Хорошо зарекомендовали себя рекомендательные системы, учитывающие большое количество данных непосредственно перед принятием решения. Области их применения являются самыми разнообразными, но наиболее эффективны они в областях прогнозирования, планирования, обучения, диагностики и управления.

В процессе разработки рекомендательной системы для медицинского центра были созданы соответствующие сервисы, интерфейсы, фасады, утилиты, модели данных и другие объектные сущности для достижения поставленной задачи. Бета-тестирование выявило возможности улучшения некоторых компонент, и они находятся в стадии доработки и разработки. Все это предназначено для реализации возможности предлагать клиенту медицинского центра наиболее подходящие для него медицинские услуги в соответствии с его электронной медицинской картой. Логика рекомендательной системы разрабатывается на объектно-ориентированном языке программирования Java, обеспечивающем требуемое быстродействие, модульность, безопасность и легковесность.

Решение по созданию данной рекомендательной системы было принято на основании исследования выгоды по ее применению во многих других проектах на рынке. Проценты увеличения доходности могут отличаться в зависимости от предоставляемых ресурсов, но самое главное, что прирост наблюдается во всех сферах.

На данный момент для предоставления рекомендаций используются данные клиента, такие как: его история обращений в центр, виды проведенных процедур, посещенные специалисты центра, а также среднестатистические направления докторов на конкретные процедуры после обследования пациента.

В дальнейшем планируется использовать полезные данные других клиентов медицинского центра для увеличения точности рекомендаций. Это могут быть их рекомендации и успешность пройденных процедур после какого-либо направления. Т.е. если проведенная процедура помогла отчасти решить проблему пациента, либо направила на правильный путь лечения, то ее следует рассматривать как рекомендацию для клиентов с похожими проблемами.

Основной визуальный эффект будет наблюдаться в процессе регистрации клиента. На стадии выбора процедуры будут выданы рекомендации, которые клиент может использовать для записи.

После завершения написания основной логики рекомендательной системы будет проводиться тестирование и подсчет пользы от ее применения.