

довании, которое там установлено. А автоматическое создание документации сократит время на составление отчетов, актов, паспортов объектов и плановых графиков обслуживания, так как не нужно будет составлять их вручную, а просто выбирая необходимые данные.

Литература

1 Стефанов, С. React.js. Быстрый старт / С. Стефанов. – СПб. : Питер, 2017. – 304 с.

К. В. Андрусенко, Е. А. Ружицкая
(ГГУ имени Ф. Скорины, Гомель)

UI/UX И ЛОГИКА ВЗАИМОДЕЙСТВИЯ В WEB-ПРИЛОЖЕНИИ «БАНКОВСКИЕ ТРАНЗАКЦИИ»

Web-приложение разработано с целью создания удобного и безопасного инструмента для управления банковскими картами и проведения финансовых транзакций. Основной задачей являлось обеспечение простоты использования, высокой производительности и безопасности пользовательских данных. Web-приложение предназначено для выполнения широкого спектра операций, таких как авторизация, регистрация, добавление новых карт, управление балансом, перевод средств между картами, а также просмотр истории транзакций.

При разработке использовались HTML5, CSS3 и JavaScript.

В разделе авторизации предусмотрены поля для ввода имени пользователя и PIN-кода, а также кнопки для входа в систему и регистрации нового пользователя. После успешной авторизации пользователь попадает на главную страницу приложения, где отображаются его банковские карты. Реализована карусель карт, позволяющая переключаться между разными счетами. Карты отображают текущий баланс и последние транзакции, чтобы пользователь мог легко отслеживать свои финансовые операции. Одной из ключевых функций является добавление новых карт. Пользователь может ввести номер карты, дату окончания срока действия и код безопасности. Эти данные проверяются с помощью JavaScript на корректность ввода, чтобы предотвратить возможные ошибки. Приложение также поддерживает перевод средств между картами. Пользователь выбирает кар-

ту-отправителя и карту-получателя, вводит сумму перевода и подтверждает операцию вводом PIN-кода. Валидация данных предотвращает отрицательный баланс и ошибочный ввод суммы.

История транзакций отображается в отдельном разделе. Здесь пользователь может просмотреть дату, сумму и тип каждой операции.

JavaScript-код структурирован в виде модулей, что улучшает читаемость и поддержку проекта. Разделение логики на отдельные модули для авторизации, управления картами, перевода средств и отображения истории транзакций делает код более организованным и легким для расширения в будущем.

Проект также учитывает требования к производительности. Все стили и скрипты минимизированы для ускорения загрузки страницы.

Е. Д. Антонов

(ГрГУ имени Янки Купалы, Гродно)

РАЗРАБОТКА ROGUELIKE-ИГРЫ „LABYRINTH OF FATE“

Игры в жанре «Roguelike» [1] представляют собой разновидность ролевых игр, отличающуюся процедурной генерацией уровней, высокой сложностью и механикой необратимой смерти персонажей. К известным представителям жанра можно отнести:

- Hades – экшен-игра с элементами roguelike, известная своим стремительным игровым процессом и насыщенной сюжетной линией;
- Slay the Spire – карточная стратегическая игра, в которой пользователи строят тактики по мере прохождения подземелий;
- Dead Cells – сочетание жанров metroidvania и roguelike, позволяющее игрокам исследовать постоянно изменяющиеся уровни и сражаться с различными противниками.

Эти примеры демонстрируют потенциал жанра в создании увлекательных игровых механик. Разработка нового проекта в данном направлении соответствует актуальным трендам игровой индустрии.

„Labyrinth Of Fate“ будет включать ключевые особенности roguelike-игр, среди которых:

- генерация уровней в процедурном формате;
- система развития персонажей с возможностью прокачки навыков и управления инвентарем;