

Национальный исследовательский университет  
«Высшая школа экономики»

Лицей

Индивидуальная выпускная работа

Отчёт о проекте

**Приложение для составления армий в настольной игре Warhammer  
40K**

*Выполнил(а) Шевляков Фёдор Сергеевич*

Научный консультант:  
Купцов Александр Александрович

Москва 2022

IT-проект, лично для меня – самая интересная область ИВР из существующих, поскольку именно в ней можно проявить себя, создав нечто оригинальное и одновременно полезное, решающее какую-либо поставленную проблему. К тому же, мне очень нравится программировать, особенно заниматься разработкой таких проектов, где в итоге получается готовый продукт, который можно не только показать публике, но и воспользоваться самому.

Мой проект решает проблему неясности и неудобства в составлении игровых армий в настольной игре Warhammer 40.000. Составление армии в этой настольной игре является ключевым аспектом игрового процесса, и новичку достаточно легко запутаться в цифрах статистики, особенно когда статистика одной игровой модели (юнита) состоит из 10 уникальных значений, которые стоит учитывать при создании армии. Поэтому новичкам приходится либо составлять армии на бумаге, либо составлять их в Excel или Word, самостоятельно внося статистики каждого юнита вручную. Оба этих способа, очевидно, очень неудобны, если ещё учитывать тот факт, что к составленной армии придётся достаточно часто обращаться в течение игрового процесса. Существуют похожие приложения, которые позволяют составлять армии на мобильном устройстве, но они все либо устарели и больше не поддерживаются на последних версиях iOS, либо версия редакции правил устарела (последняя редакция – 9, 2020 год; в похожих приложениях 6 редакция, 2016 год), либо попросту возможности составления армий нет.

Заказчика как такового у меня нет, моей целевой аудиторией являются игроки в настольную игру Warhammer 40.000. Сообщество игроков в РФ и по всему миру в эту настольную игру является достаточно большим, и по ней регулярно проводятся международные турниры.

Мой проект изначально задумывался в виде мобильного приложения под iOS. По моему мнению, решение поставленной проблемы мобильным приложением является оптимальным. В приложении планировались

возможности создания армии (в том числе подбор типа подразделения и возможность выбора цели по очкам(стоимости) юнитов), её модификации; добавления дополнительного снаряжения к юнитам (и возможность добавления собственного доп. снаряжения), а также возможность сохранения миниатюр в коллекцию для более удобного доступа к ним; возможность просмотра статистики всех юнитов армии сразу, на одном экране, чтобы было удобно обращаться к ней во время игры; возможность сохранения данных об армиях и коллекциях в облачное хранилище данных, для возможности просмотра на других устройствах.

В итоге получилось реализовать весь проект. Присутствуют все вышеперечисленные возможности, присутствует система авторизации и регистрации, а также приложение готово к работе не только на смартфонах, но и на планшетах с iOS. Также планировался выход приложения в App Store, но в связи с отсутствием официальных способов приобрести подписку Apple Developer в РФ, к сожалению, это не получилось.

Этапы разработки состояли из разработки и планирования пользовательских сценариев; проектирования интерфейса и навигации приложения; непосредственно разработки приложения и бэкенда, а также подключения их; тестирования приложения средствами языка программирования Swift и среды разработки Xcode, финальной отладки.

Для разработки самого приложения мною был выбран язык программирования Swift, фреймворк SwiftUI и Xcode как IDE, поскольку нативная разработка имеет значительные преимущества, по сравнению с кроссплатформенной по типу более широких возможностей для создания интерфейсов, особенно на мобильных устройствах. В приложении у меня также используется Swift-библиотека Alamofire, для более удобного сохранения Cookies в функции авторизации. Для тестирования использовался фреймворк XCTest, который позволяет тестировать проекты Xcode. В качестве бэкенда у меня связка сервера на Flask (на языке

программирования Python и PyCharm в качестве IDE), с использованием библиотеки Flask-Login для авторизации, и база данных MariaDB (Open-Source форк MySQL), для соединения бэкенда и базы данных я использовал библиотеку SQLAlchemy. Эту связку сервера и базы данных я использовал из-за того, что я уже имел ранее опыт разработки с ней, а также простоты настройки её на машине, на которой будет запущен сервер. Этой машиной является мой старый домашний компьютер, которому получилось придать вторую жизнь, установив на него Linux и запустив сервер. Для парсинга Книг Правил я пользовался онлайн OCR-сервисами, а также Numbers, редакторами JSON-файлов и Python для корректировки пропаршенных данных.

Во время процесса разработки был один главный риск – проблемы с парсингом статистики, но, как было сказано выше, эту проблему удалось решить с помощью онлайн OCR-сервисов и последующей корректировки данных. Во время процесса разработки я сталкивался с немалым количеством проблем связанных с различными особенностями языка Swift и среды разработки Xcode. Также, после выхода бета-версии iOS 16.0 примерно в начале лета этого года, мною сразу было замечено большое количество багов интерфейса из-за нововведений в новой версии Xcode и новой версии фреймворка SwiftUI. Все проблемы, связанные со средой разработки и языком программирования, в итоге удалось решить внимательным чтением официальной документации определенных функций, что, определенно, прибавило знаний об iOS разработке. В дальнейшем, мой проект можно развить, добавив ещё больше информации по каждому юниту, например, фотографию миниатюры (чтобы новичкам было легче идентифицировать свои фигурки) или короткий отрывок из литературной части вселенной Warhammer 40K о данном юните. Помимо этого, для новичков можно добавить функцию автоматического создания армий на основе миниатюр в коллекции и цели по очкам. Также, возможно расширить приложение для настольной игры Warhammer Fantasy,

структура правил которой несколько отличается от 40K, но составление армий происходит примерно в таком же порядке.