ОТЧЕТ

о проектной работе

по теме: «MemDev. Развитие памяти»

по дисциплине: Практикум программной инженерии

Ханты-Мансийск

2024

Содержание

[ВВЕДЕНИЕ 3](#_Toc169739182)

[КОМАНДА 5](#_Toc169739183)

[ЦЕЛЕВАЯ АУДИТОРИЯ 6](#_Toc169739184)

[ОПРЕДЕЛЕНИЕ ПРОБЛЕМЫ 7](#_Toc169739185)

[ПОДХОДЫ К РЕШЕНИЮ ПРОБЛЕМЫ 8](#_Toc169739186)

[ТРЕБОВАНИЯ К ПРОДУКТУ 9](#_Toc169739187)

[СТЕК ТЕХНОЛОГИЙ 10](#_Toc169739188)

[ПРОТОТИПИРОВАНИЕ 11](#_Toc169739189)

[РАЗРАБОТКА СИСТЕМЫ 12](#_Toc169739190)

[ЗАКЛЮЧЕНИЕ 15](#_Toc169739191)

# ВВЕДЕНИЕ

Человеческая память очень ограничена. А в условиях современного мира, где мы сталкиваемся с обилием информации практически каждый день, наша память вынуждена запоминать огромный пласт совершенно ненужной информации, которая перекрывает собой все нужные факты. Это зачастую мешает сконцентрироваться на работе или учёбе. Но память можно развивать, что приведёт к удержанию нужной информации в разуме человека на более долгий период. И делать это лучше с раннего возраста. Развитие памяти может происходить по-разному. Можно проводить различные мнемонические упражнения в кругу семьи, можно заниматься в специальных секциях.

Одним из перспективных методов развития памяти считается использование электронных приложений, представляющих из себя игровые сборники. В таких сборниках можно собирать разные в своей основе мнемонические игры, что позволяет эффективно и разносторонне развивать память.

К сожалению, существующие приложения не способны обеспечить полноценного развития памяти у детей младшего школьного возраста в силу того, что для серьезного охвата аудитории необходимо применять многочисленные техники удержания внимания. Если этого не учитывать, то, несмотря на всю свою эффективность, приложение окажется невостребованным. Поэтому мы решили разработать собственное приложение, учитывающее этот недостаток.

Цель работы – проектирование и разработка приложения, развивающего мнемонические навыки и эффективно удерживающего внимание аудитории.

Задачи проекта:

1) Анализ современных технологий, используемых для разработки игр

2) Формирование технического задания на разработку приложения

3) Проектирование приложения с учетом, разработанного тех. задания

4) Реализация приложения

# КОМАНДА

Половодов Семён Александрович, 1511б – Программист, Тимлид

Черноскутов Артем Леонидович, 1511б – Программист, дизайнер

# [ЦЕЛЕВАЯ АУДИТОРИЯ](#ЦА)

В качестве продукта мы предлагаем нашей целевой аудитории различные мнемонические упражнения, собранные в виде компьютерной игры.

Это приложение предназначено для детей в возрасте от 7 до 14 лет, поскольку, с одной стороны, именно в этот момент идет активное формирование когнитивных и мнемонических навыков, а с другой - дети подобного возраста легко увлекаются различными игровыми приложениями.

Поскольку сейчас среди выбранной возрастной аудитории очень популярен мобильный гейминг, платформа, на которой будет выпускаться наш продукт, была определена как OC Android. Это наиболее распространенная среди всех мобильных устройств операционная система, и игра, выпущенная на ней, охватит большую целевую аудиторию.

В перспективе наша игра будет выпущена на таких платформах, как Google Play и GetApps, которые являются основными дистрибьютерами игровых приложений для Android. Самым же реалистичным для первого периода жизни продукта является распространение приложения через передачу данных между мобильными устройствами посредством Телеграмм бота.

# ОПРЕДЕЛЕНИЕ ПРОБЛЕМЫ

В нынешних условиях «цифрового века» детям младшего школьного возраста может быть трудно усваивать учебные материалы в связи с захламлением их краткосрочной памяти контентом из интернета. Своим проектом мы стремимся помочь детям сохранять внимание на уроках в школе и на других учебных занятиях, повышая уровня их концентрации путём ежедневного выполнения упражнений в нашем приложении.

Проведя анализ отзывов об аналогичных программах, а так же в результате личных встреч с людьми, попадающими в целевую аудиторию, был выявлен основной недостаток уже существующих игр для развития памяти: несмотря на то, что они интересны при первом знакомстве, через некоторое весьма короткое время они надоедают.

Кроме того, не все существующие логические и мнемонические игры подходят для выделенной нами аудитории: они либо слишком простые, рассчитанные на детей дошкольного возраста, либо слишком сложные, рассчитанные на подростков от 12 до 18, глубоко увлеченных развитием собственной памяти.

Также, лишь малая часть опрошенных вероятных пользователей считает развитие памяти серьезным делом, которым нужно регулярно заниматься. Около тридцати процентов опрошенных вообще не подозревали о том, что существуют игры для развития памяти.

# ПОДХОДЫ К РЕШЕНИЮ ПРОБЛЕМЫ

Для развития памяти важны несколько моментов. Один из них это комплексность выполняемых упражнений. Упражнения должны быть разнообразны, а выполняться они должны систематически. Для соответствия этим требованиям этого мы решили создать множество непохожих друг на друга мини-игр, развивающих разные виды памяти. Системность посещения будет поддерживаться системой опыта, наград и рейтинга, при использовании которой пользователь поощряется за провождение времени в игре.

Вторым важным пунктом является динамическая сложность игры. Чтобы игрок не привыкал (и, как следствие, не уставал от процесса) и прогрессировал в развитии, приложение должно иметь автоматически изменяемую сложность, которая будет поддерживать интерес при прохождении мини-игр и динамику роста игрока.

Поскольку развитие памяти – не самое распространенное среди выбранной целевой аудитории времяпрепровождение, необходимо привлекать новых пользователей не столько содержанием приложения, сколько его формой. Следовательно, необходим яркий и запоминающийся дизайн и простой интерфейс. Развитая система поощрения за посещаемость также увеличит процент потребителей, которые не удаляют приложение через день после установки, а регулярно заходят в него хотя бы на 10-15 минут в день.

Выбранная нами возрастная группа, увлекается играми - «тайм киллерами». Это простые аркады, игровая сессия в которых может занимать при этом очень большой временной промежуток. Они занимают свободное время, не заставляя игрока напрягаться во время игрового процесса. Наше приложение можно замаскировать под один из таких «тайм-киллеров», что повысит его привлекательность в глазах целевой аудитории.

# ТРЕБОВАНИЯ К ПРОДУКТУ

Исходя из анализа целевой аудитории и аналогов, мы составили реестр требований к нашему приложению и его MVP.

Главная цель системы – развитие мнемонических навыков пользователя. Платформа, на которую рассчитано приложение – OC Android.  
Приложение должно включать в себя расширяемый набор мини-игр на развитие памяти, расположенных в одном каталоге. Доступ к каждой из игр должен осуществляться не более, чем за три клика после запуска приложения.  
Посещаемость игр должна отслеживаться, и на ее основе должна формироваться система поощрений в виде подсказок.

В приложении должна быть система рейтинга, построенная на количестве и результативности игр. В ней должна учитываться системность запусков приложения.

Сложность игры должна изменяться динамически, в зависимости от успехов игрока. Должна присутствовать возможность ручной регулировки сложности.

Каждое меню должно иметь не более пяти кнопок. Все надписи на кнопках должны быть выполнены одним и тем же шрифтом. Все кнопки должны быть выполнены в одной стилистике.

Добавление новой мини-игры не должно приводить к изменению кода остальных систем.  
Приложение должно состоять из одного файла. Это необходимо для простой передачи через Телеграмм бота.

# СТЕК ТЕХНОЛОГИЙ

С# – это объектно-ориентированный язык программирования. Он был создан в период с 1998 по 2002 год командой инженеров Microsoft под руководством Андерса Хейлсберга и Скотта Вильтаумота. Язык входит в семью С-подобных языков. Синтаксис приближен к Java и C++

C# без преувеличения крайне популярен среди создателей видеоигр. Язык используется для разработки игр под Windows, macOS, Android и iOS. Все дело в Unity – платформе для работы с 3D-графикой. С# лучше остальных языков адаптирован под работу с этим движком.

Этот движок на данный момент является крайне популярным при разработке мобильного ПО, поскольку понятный редактор и инструментарий позволяет быстро начать разработку даже тем, кто ранее с ним не работал. Большое коммьюнити обеспечивает ответы на практически все возникающие вопросы. Unity обладает полноценным графическим редактором, позволяющим быстро видоизменять программу, не используя сторонние инструменты.

Из популярных проектов стоит выделить такие хиты игровой индустрии, как Bastion, Wasteland, знаменитый Doom 3 и Hearthstone.

Мы решили использовать связку Unity-C#. Такой выбор обусловлен, отчасти, тем, что на изучение возможностей движка и языка программирования должно потребоваться минимальное количество времени, а отчасти – особенностями самих средств разработки, обеспечивающими соблюдение технического задания.

Для разработки дизайна интерфейса был выбран графический редактор Adobe Photoshop. Его функционал достаточно широк и перекрывает все наши потребности, начиная с простого рисования и заканчивая сложной обработкой фотографий.

# ПРОТОТИПИРОВАНИЕ

В ходе работы над созданием интерфейса были созданы различные прототипы, позволяющие оценить простоту и удобность интерфейсов.  
В итоге, мы остановились на интерфейсе с минимальным набором кнопок, который позволяет запустить любую из имеющихся игр за три клика.

При выборе дизайна мы остановились на спокойных тонах, поскольку такая палитра не отвлекает пользователя от решения самих задач. Кроме того, опытным путем мы выяснили, что количество цветов, одновременно используемых на экране, не должно превышать четырех-пяти. Это позволяет сделать приложение, от которого не рябит в глазах, но при том достаточно красочное, чтобы заинтересовать целевую аудиторию.

Занявшись таким вопросом, как расположение кнопок на экране, мы выявили следующую закономерность: то, что расположено в центре экрана, привлекает больше всего внимания. Кнопки, расположенные по краям экрана, а также нетипично расположенные, плохо воспринимаются пользователем.

При работе над прототипами будущего программного обеспечения мы обратили внимание на легкость обновления: все игры вынесены в отдельное меню, к которому легко прибавить новый элемент. При том этот элемент не будет выбиваться из уже существующего интерфейса.

# РАЗРАБОТКА СИСТЕМЫ

При разработке системы мы выделили следующие программные блоки, требующие реализации:

1. Меню

А) Главное меню

Б) Меню выбора игры

В) Меню настроек

2) Мини-игры (модули, необходимые для каждой из них)

А) Основная логика игры

Б) Алгоритм, изменяющий сложность игры

В) Система, награждающая за посещаемость

3) Модуль дизайна

А) Музыкальное сопровождение

Б) Иконка игры

В) Общий стиль

Далее приведена основная логика реализованных мини-игр.

Игра 1. «Повтори путь»

Имеется поле NxN блоков. При начале игры поочередно загораются и исчезают X блоков, после чего необходимо нажать на блоки в той же очередности, в которой они загорались.

При успешном соблюдении последовательности игрок видит большую зеленую галочку, после чего начинается следующая партия, в которой загорается уже X+1 блоков. При ошибке следующая партия снова начинается с X блоков.

В статистику записывается лучший результат - максимальное количество правильно нажатых блоков умноженное на 10, а также последний результат - количество правильно нажатых блоков в последней игровой итерации умноженное на 10.

Игра 2. «Запомни слова»

В начале игры запускается первый таймер на N секунд, и на экране появляется X полей, в каждом из которых написано слово. Пока работает первый таймер, игроку необходимо запомнить максимально возможное количество слов, и соответствующие им поля. По истечении таймера слова исчезают, а поля становятся пустыми. Запускается второй таймер, во время работы которого игроку необходимо заполнить поля соответствующими словами. По истечении второго таймера, поля, куда игрок вписал правильный ответ, загораются зеленым цветом и надпись в них, меняется на "Правильно!", а поля, где ответ оказался неверным, загораются красным цветом, и надпись в них меняется на "Ошибка!".

В статистику записывается лучший результат - количество правильно введенных слов в той партии, в которой их было наибольшее количество, умноженное на 10, и последний результат - количество правильно введенных слов в последней игровой партии, умноженное на 10.

Игра 3. «Найди пару»

Перед игроком имеется набор карточек, которые расположены рубашкой вверх. В начале партии карты переворачиваются на несколько секунд, игроку необходимо запомнить их взаимное расположение. После этого карты переворачиваются обратно, и игроку необходимо нажимать на карточки.

Карты открываются парами. При нажатии на первую карту, она переворачивается, и игроку необходимо перевернуть вторую, на которой находился тот же рисунок, что и на первой. В случае правильного выбора второй карты, обе карты остаются перевернутыми вверх рисунком и становятся неактивными. В случае неправильного выбора второй карты, через короткий промежуток времени, обе карты переворачиваются рубашкой вверх.

Новая партия начнется тогда, когда все карты останутся перевернутыми рисунком вверх.

Игра 4. «Минное поле»  
  
Перед игроком появляется квадратное игровое поле, состоящее из клеток. На каждой клетке может быть расположена мина. Также на поле располагаются две клетки-цели, подсвеченные особым образом. Через несколько секунд после начала игровой сессии мины и особые клетки прячутся, и игроку необходимо построить маршрут от одной клетки к другой, минуя мины. Если это удается, в следующей партии на поле добавляется больше мин, само поле становится больше.  
В статистику записывается список самое высокое соотношение количества пройденных уровней к длине маршрутов, а также номер последнего уровня, деленный на длину последнего пути.

# ЗАКЛЮЧЕНИЕ

Был составлен список требований к проекту. В этот список вошли требования к визуальному оформлению, функциональные требования и требования к итоговому виду проекта. Полный список требований предоставлен в реестре.

Для разработки нашего продукта мы выбрали C# в качестве основного языка программирования и Unity в качестве игрового движка. Это обусловлено, с одной стороны, простотой их освоения, а с другой – наличием всего необходимого функционала.

Во время проектирования мы выбрали наиболее удачный дизайн приложения и рассмотрели различные системы расположения кнопок, что позволило более органично вписать их в интерфейс.

В процессе реализации игры мы разработали несколько мини-игр с различной механикой. Нами был создана внутренняя логика приложения, состоящая из различных модулей, которые можно совершенствовать, не затрагивая остальные. Нам удалось создать систему настройки сложности. Были реализованы четыре мини-игры, а также статистика для каждой из них.  
Полученная программа состоит из одного apk файла, предназначенного для установки на устройства с ОС Android.

Таким образом, результатом нашей работы является полностью функциональное мобильное приложение для развития памяти. Благодаря внутренней логике, в перспективе это приложение может включать в себя куда большее количество мини-игр, что увеличит интерес игрового процесса.