СОДЕРЖАНИЕ

Задание 2

Аннотация 4

Annotation 6

[Введение 8](#_Toc138151954)

[1 Определение объекта исследования и постановка задачи 10](#_Toc138151955)

[1.1 Определение методики RSVP 10](#_Toc138151956)

[1.2 Преимущества и недостатки методики RSVP 12](#_Toc138151957)

[1.3 Обзор существующих решений 13](#_Toc138151958)

[1.4 Постановка задач квалификационной работы 20](#_Toc138151959)

[1.5 Обоснование актуальности работы 21](#_Toc138151960)

[2 Разработка концепции и методики достижения цели и решения задач дипломного проектирования. 22](#_Toc138151961)

[2.1 Память и типы восприятия 22](#_Toc138151962)

[2.2 Определение цветовосприятия 23](#_Toc138151963)

[2.3 Идеи, формирующие основу исследования и разработки 30](#_Toc138151964)

[2.4 Выбор средств реализации 30](#_Toc138151965)

[3 Детальная проработка конкретного элемента, входящего в общую концепцию решения. 32](#_Toc138151966)

[3.1 Описание функционала приложения с помощью use-case диаграмм 32](#_Toc138151967)

[3.2 Планирование архитектуры приложения 37](#_Toc138151968)

[3.3 Проектирование на уровне компонентов приложения 39](#_Toc138151969)

[3.4 Проектирование интерфейса приложения 42](#_Toc138151970)

[3.5 Описание процесса разработки 46](#_Toc138151971)

[3.6 Описание классов и объектов 46](#_Toc138151972)

[3.7 Масштабирование и расширение функционала 49](#_Toc138151973)

[4 Практическая реализация приложения 50](#_Toc138151974)

[4.1 Демонстрация работы приложения 50](#_Toc138151975)

[Заключение 54](#_Toc138151976)

[Списо литературы 55](#_Toc138151977)

Приложение А. Листинг программы 56

# ВВЕДЕНИЕ

Темой выпускной квалификационной работы является: «Разработка веб-приложения на основе технологии RSVP (Rapid Speed Visual Presentation) и психологии цветовосприятия».

RSVP — это технический термин, который используется для обозначения метода, при котором большой объем информации или контента разбивается на отдельные фрагменты и показывается пользователю по мере его запроса или действия. В контексте методов скорочтения, RSVP Reader представляет собой приложение, разделяющее тест на фрагменты (слова и словосочетания), достаточно маленькие для их восприятия сразу целиком.

Психология цветовосприятия —область психологии, которая изучает влияние цвета на человеческое поведение и настроение. Цвета могут вызывать различные эмоции и оказывать влияние на мыслительный процесс, а также на выбор и оценку предметов и событий. В работе планируется исследовать возможные способы использования знаний в этой области для усиления восприятия текстовой информации, за счет более прямого воздействия цвета на эмоции.

Цель работы – произвести углубленный анализ предметной области, погрузившись в такие темы как методология RSVP и основы психологии цветовосприятия, ознакомиться и выявить сильные и слабые стороны существующих решений, реализующих методологию чтения Rapid speed visual presentation, разработать способы их возможного усовершенствования, а также исследовать возможности дополнительного воздействия на восприятие информации с помощью использования цветовосприятия и подсознательного внимания пользователя. Разработать приложение, демонстрирующее результаты исследования, и способствующее сбору информации для его продолжения.

Актуальность исследования обоснована в первую очередь характерным ростом количества информации и информационного шума, что требует больших усилий для обработки поступающего контента. Метод скорочтения RSVP применим именно в таких условиях – так как дает преимущество в скорости восприятия информации, в то время как один из главных недостатков – ухудшение понимания материала, в данном случае не критичен.

Вторым важным аспектом является необходимость уделения особого внимания форме представления информации для современного пользователя, так как неторопливое и регрессионное чтение вытесняется медиа контентом и способом визуального оформления «бесконечная лента».

Задачами выполнения выпускной квалификационной работы являются:

- Закрепление полученных в ходе обучения навыков работы с литературой;

- Демонстрация полученных в ходе обучения навыков проектирования и разработки информационных систем;

- Проверка степени подготовленности к профессиональной трудовой деятельности по выбранной специальности;

# Определение объекта исследования и постановка задачи

## Определение методики RSVP

Быстрое последовательное визуальное предъявление (Rapid serial visual presentation), далее RSVP – это метод представления визуальной информации, при котором она разделяется на минимальные дискретные порции и демонстрируется последовательно на достаточной для восприятия одной порции скорости. Идея такого метода заключается в концентрации пользователя на одной порции, и таким образом сведении к минимуму времени и усилий на движение и фокусировку зрачка.

На рисунке 1 проиллюстрированы примерные границы фокусирования, необходимого для чтения.

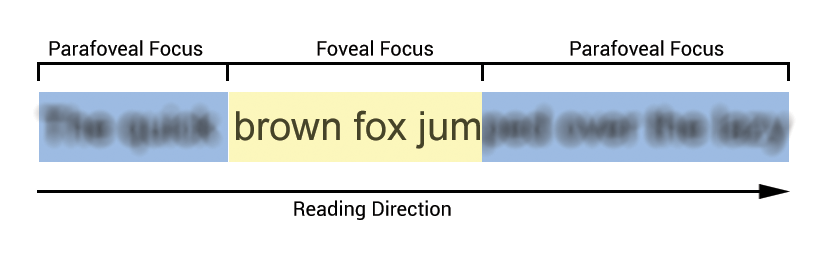


Рисунок 1.1 – Демонстрация восприятия информации при чтении

Фовеальная и парафовеальная области — это две области нашего зрительного поля, которые отличаются по своим особенностям.

Фовеальная область - центральная область зрительного поля, состоящая из около 5 градусов визуального поля. Она содержит высокую концентрацию фоторецепторов - клеток в сетчатке глаза, которые реагируют на свет. Фоторецепторы в фовеальной области называются конусами и обеспечивают острое зрение, цветовое видение и способность различать мелкие детали. Благодаря тому, что фовеальная область содержит множество конусов, люди могут читать, писать, смотреть телевизор и выполнять другие задачи, требующие точности и детализации.

Парафовеальная область - область зрительного поля, которая находится вне фовеальной области, но все еще имеет достаточно высокую плотность фоторецепторов для обеспечения хорошей видимости. Она расположена вокруг фовеальной области и занимает около 10-15 градусов визуального поля. Фоторецепторы в парафовеальной области называются палочками и менее чувствительны к свету, чем конусы. Поэтому они не предназначены для работы при ярких условиях освещения или для обеспечения остроты зрения, но они обладают более высокой чувствительностью в темноте и позволяют видеть объекты на периферии зрительного поля.

Согласно исследованиям, в том числе с применением методики движущегося окна, было установлено, что именно фовеальная область фокуса несет для пользователя основную часть информации, тогда как текст слева и справа от центра фиксации будет восприниматься скорее подсознательно.

Методика движущегося окна (Moving Window) - метод, используемый в психологии и когнитивной науке для изучения чтения и восприятия текста. Она состоит в том, чтобы предъявлять испытуемому текстовые фрагменты, при этом окно с постоянным размером перемещается по тексту.

Перемещение окна происходит со скоростью, которую контролирует экспериментатор. При этом читаемый текст проходит через окно, оставляя за ним только текущий фрагмент текста. Эта методика позволяет рассчитать время, затраченное на чтение каждого слова или группы слов.

Один из способов использования методики движущегося окна состоит в том, чтобы изменять размер окна, чтобы проверить, какой размер фрагмента текста оптимально подходит для обработки мозгом. Используя этот метод, исследователи могут изучать, как люди воспринимают текст на разных уровнях, таких как буквы, слова, предложения и абзацы.

Также методика движущегося окна может использоваться для измерения внимания и когнитивной загрузки. Например, если окно передвигается слишком быстро, человек может пропустить несколько слов или даже фраз, что свидетельствует о низкой концентрации внимания. И наоборот, если окно передвигается слишком медленно, это может вызвать скученность и отвлечь внимание от чтения текста.

Методика движущегося окна является полезным инструментом для изучения когнитивных процессов, связанных с чтением и пониманием текста, а также может быть применена для повышения эффективности обучения и улучшения качества чтения.

Таким образом, исходя из подобных исследований, было сделано предположение, что чтение может быть быстрее и эффективнее, если учитывать перечисленные выше особенности и представлять информацию небольшими фрагментами, подстраивая их под границы допустимой области.

## 1.2 Преимущества и недостатки методики RSVP

К преимуществам использования методики RSVP для чтения можно отнести:

* Увеличение скорости восприятия информации (при выборе комфортных для конкретного пользователя параметров)
* Так как текст показывается порционно – для комфортного чтения требуется относительно небольшой участок экрана, что актуально для использования методики на мобильных устройствах или смарт часах.
* В основе метода лежат исследования движений глаз и стремление свести их к минимуму, что делает методику особенно полезной для людей с нарушениями зрения и глазодвигательной активности

К недостаткам на данный момент относится

* Небольшое ухудшение понимания материала, в основном из-за сложности регрессии – возврата к предыдущему фрагменту, которое сказывается на сложности изучения с помощью методологии научной литературы (так как незнакомые термины скорее всего потребуют больше времени для восприятия, что сложно предсказать с помощью алгоритма)
* Затруднительность поиска по тексту, например, из-за отсутствия ассоциаций с положением материала в определенном месте страницы.
* Для большого числа пользователей методики – сложности с концентрацией, вниманием и координацией (пользователи жалуются на то, что забывают моргать, либо испытывают сложности с удерживанием взгляда в центре экрана, легко отвлекаются)

## 1.3 Обзор существующих решений

Существует ряд готовых реализаций чтения на основе технологии RSVP, все они включают в себя основной функционал

* Показ фрагментов текста;
* Возможность постановки показа на паузу;
* Один, либо несколько способов загрузки текстового контента;
* Регулирование скорости показа фрагментов.

Далее представлены самые популярные приложения и их отличительные особенности.

**Spritz**

Spritz часто путают с RSVP: и называют отдельной технологией. Таким образом, ряд приложений на рынке идентифицируют себя как «чтение методом spritz», что на самом деле просто клонирует оригинальный продукт.

На рисунке 1.2 представлено сравнение графиков популярности поиска по запросам «speed reading app» и «spritz reading».



Рисунок 1.2 – Анализ популярности запросов «speed reading app» и «spritz reading»

По графикам видно значительный всплеск в 2014 году, причем интерес к способам скорочтения в целом совпадает с интересом к конкретному приложению – Spritz. Это можно также увидеть на графиках популярности по регионам и похожим запросам – рисунок 1.3.

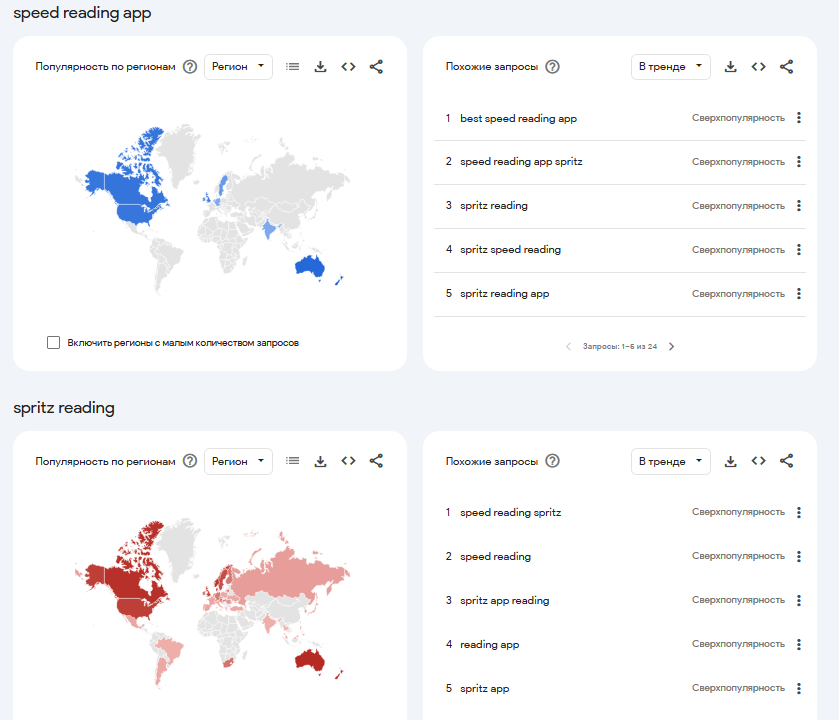


Рисунок 1.3 – Анализ популярности по регионам и частым запросы по темам «speed reading app» и «spritz reading»

Официальная версия spritz – мобильное приложение, призванное облегчить чтение с небольших экранов, при этом значительно повысив скорость чтения. На данный момент также существует в виде расширения chrome.

Интерфейс приложения представлен на рисунке 1.4.



Рисунок 1.4 – Интерфейс приложения Spritz

**Reedy**

Reedy – личная разработка своей версии приложения для скорочтения, фактически представляет из себя расширение к браузеру Chrome, соответственно требует только наличие самого браузера и установку расширения, вызывается из области с остальными расширениями на веб-странице с целевым контентом. На данный момент имеет версию в форме мобильного приложения.

Особенностью этого расширения можно назвать показ контекста – экспериментальная функция автора, позволяющая видеть текст до, после, либо вокруг текущего фрагмента.

Интерфейс приложения представлен на рисунке 1.5.

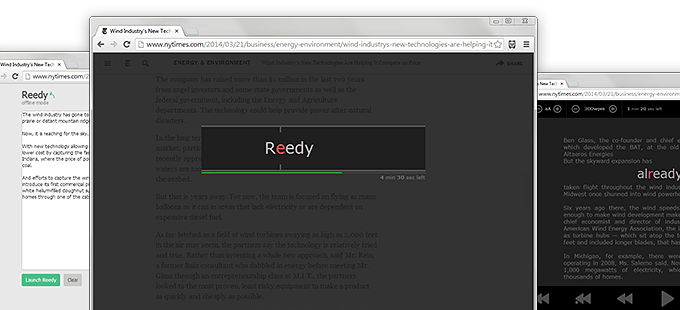


Рисунок 1.5 – Интерфейс приложения Reedy

[**Spreeder**](http://www.spreeder.com)

Spreder – один из проектов компании eReflect*,* которая специализируется на разработке ПО для улучшения навыков письма, быстрого чтения, набора текста. По веб-странице компании (рисунок 1.6) можно сказать, что Spreeder, как и остальные проекты создавался и развивался компетентными специалистами, причем не только в сфере информационных технологий и разработки ПО, но и в сферах лингвистики, психологии и непосредственно области скорочтения.

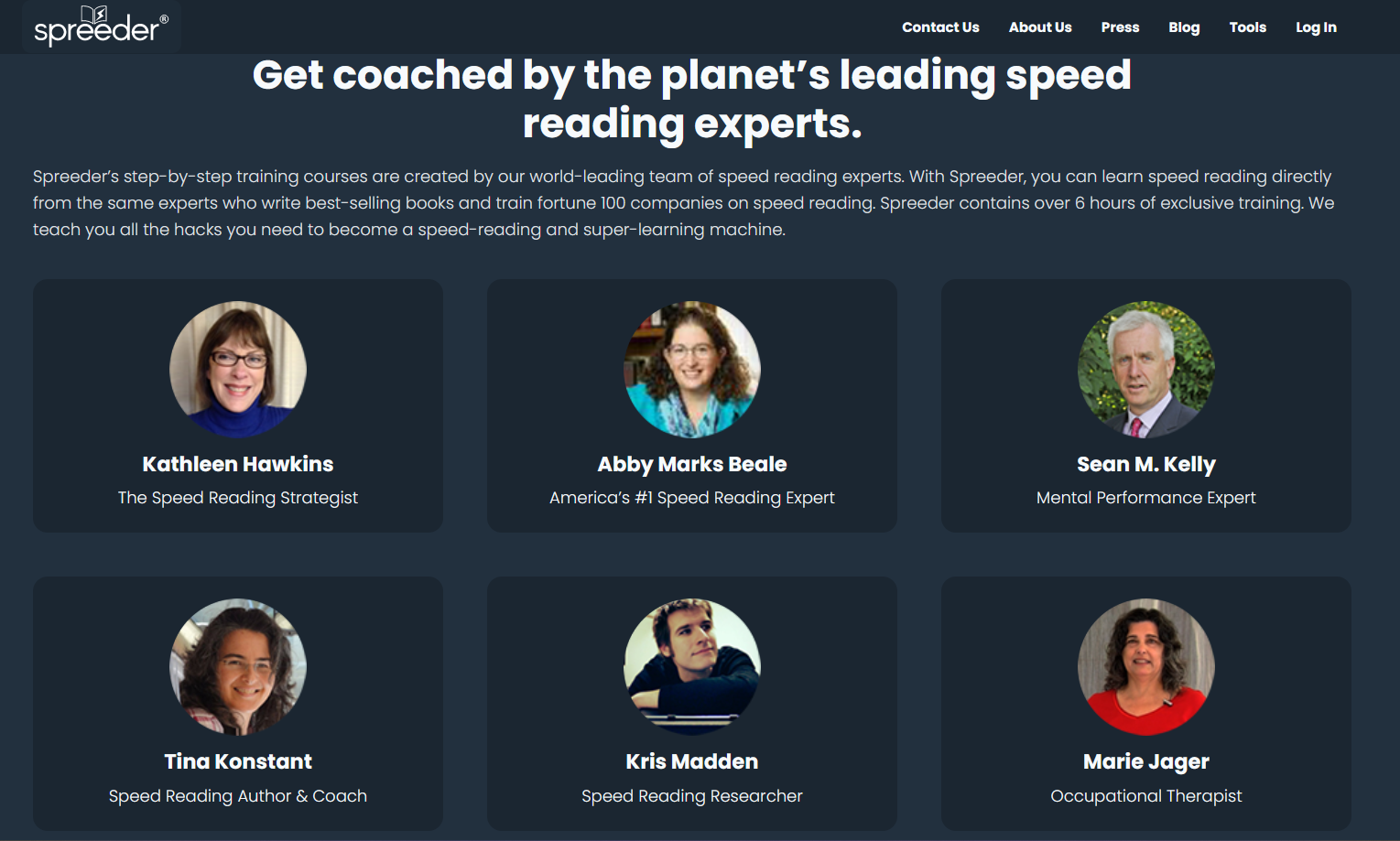
**

Рисунок 1.6 – Эксперты компании eReflect, участвовавшие в создании Spreeder.

Spreeder возможно является одной из первых имплементаций технологии скорочтения по методу RSVP – было найдено упоминание и инструкция использования spreeder от 2009 года.

Интерфейс приложения представлен на рисунке 1.7.

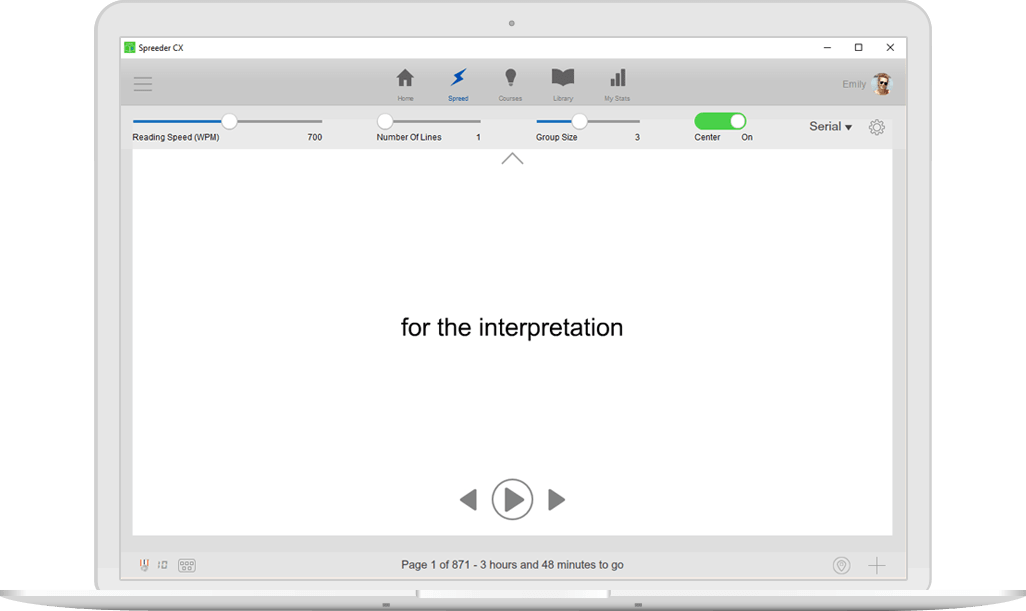


Рисунок 1.7 - Интерфейс приложения Spreeder

Тем не менее, опираясь на анализ доступных интернет-ресурсов, и зафиксированные упоминания приложений для скорочтения, интерес к данной области был вызван именно разработкой spritz reading, что видно по графикам на рисунке 1.8. Для получения такой зависимости были добавлены графики приложений по отдельности – «Spritz reading» и «Spreeder», и общая формулировка «Speed reading app». Так как повышение популярности запроса «Speed reading app» совпадает с годом старта и всплеском по количеству запросов «Spritz reading», можно сделать вывод, что одно спровоцировало другое.

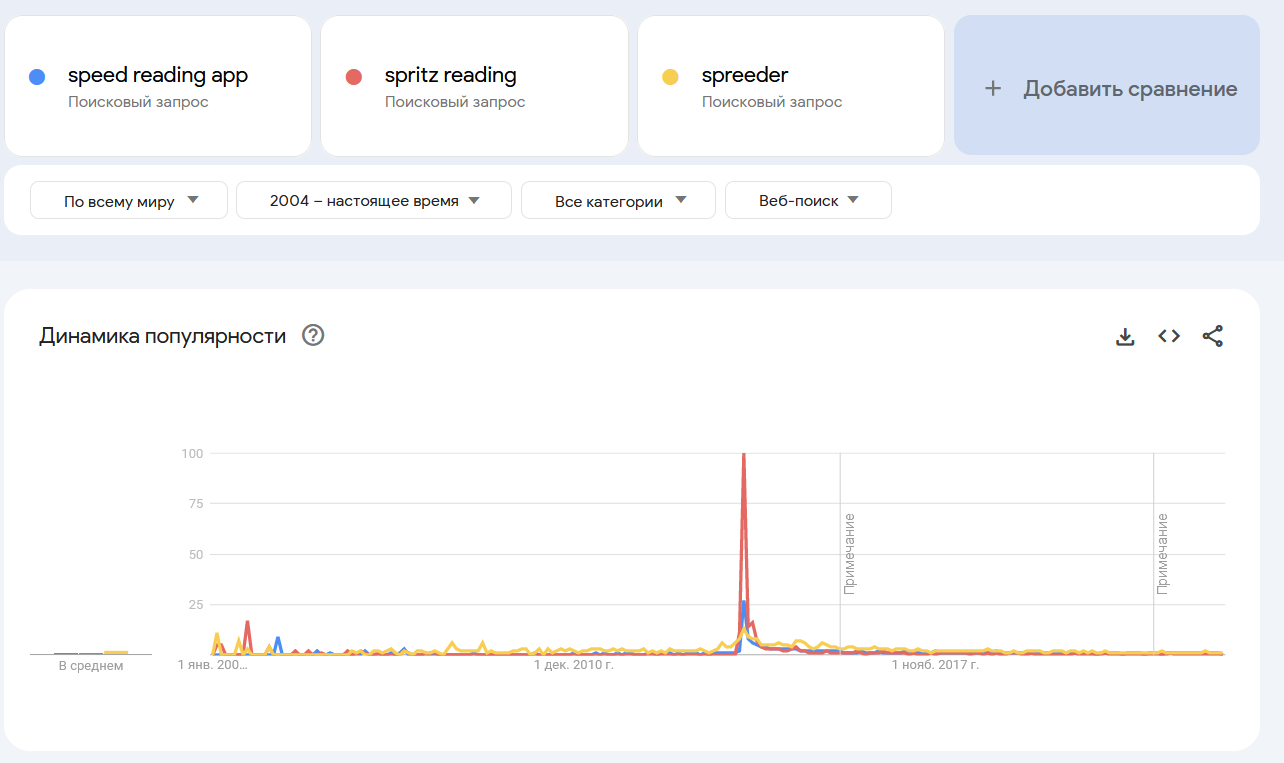


Рисунок 1.8 – Сравнение популярности запросов «speed reading app», «spritz reading» и «spreeder»

Так как значительных отличий в работе приложений не было найдено, можно сделать вывод, что популярность Spritz обусловлена грамотной маркетинговой кампанией.

## 1.4 Постановка задач квалификационной работы

Решения, основанные на RSVP постоянно совершенствуются для упрощения фиксации зрения на фрагменте текста и увеличения скорости восприятия информации, и в зависимости от приложения можно увидеть использование следующих приемов и алгоритмов:

* Комфортное разбиение текста на фрагменты – обработка предлогов, инициалов, сокращений и устойчивых выражений, обработка пунктуации;
* Комфортное для восприятия позиционирование фрагмента, которое не всегда совпадает с привычным центрированием (рисунок);
* Выделение центра фрагмента контрастным цветом;
* Вычисление времени на демонстрацию фрагмента, исходя из количества символов в нем;

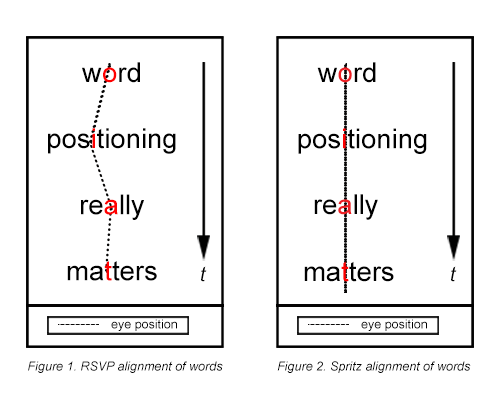


Рисунок 1.9– демонстрация пользы дополнительных алгоритмов центрирования фрагментов

Задачей выпускной квалификационной работы является анализ существующих решений, разработка способов их возможного усовершенствования, а также исследование возможности дополнительного воздействия на восприятие информации с помощью использования цветовосприятия и подсознательного внимания пользователя.

## 1.5 Обоснование актуальности работы

Актуальность исследования обоснована в первую очередь характерным ростом количества информации и информационного шума, что требует больших усилий для обработки поступающего контента. Метод скорочтения RSVP применим именно в таких условиях – так как дает преимущество в скорости восприятия информации, в то время как один из главных недостатков – ухудшение понимания материала, в данном случае не критичен.

Вторым важным аспектом является необходимость уделения особого внимания форме представления информации для современного пользователя, так как неторопливое и регрессионное чтение вытесняется медиа контентом и способом визуального оформления «бесконечная лента».

# 2 Разработка концепции и методики достижения цели и решения задач дипломного проектирования.

## 2.1 Память и типы восприятия

Существует три основных типа восприятия: визуальный (визуальное восприятие), аудиальный (аудиальное восприятие) и кинестетический (кинестетическое восприятие).

Визуальное восприятие связано с предпочтением визуальной информации, такой как картинки, диаграммы, графики и другие визуальные элементы.

Аудиальное восприятие связано с предпочтением звуковой информации, такой как разговоры, лекции, музыка и другие звуковые элементы.

Кинестетическое восприятие связано с предпочтением ощущений и физических воздействий, таких как прикосновения, движения, напряжение мышц и другие подобные ощущения.

Каждый человек имеет индивидуальный набор восприятий и часто использует несколько типов восприятия одновременно. Знание своего типа восприятия может помочь улучшить эффективность обучения и работы, так как позволяет использовать наиболее эффективные методы обработки информации.

Визуальное восприятие является доминирующим в большинстве случаев потому, что большая часть информации, поступающей к нам из окружающего мира, передается через зрение. Глаза способны воспринимать различные цвета, формы, текстуры и другие характеристики объектов, которые мы видим. Визуальные данные обрабатываются в разных участках нашего мозга, что позволяет нам быстро и точно распознавать и интерпретировать окружающую среду.

Кроме того, наше зрительное восприятие также играет важную роль в коммуникации и социальном взаимодействии. Мы можем определять эмоции и настроение других людей по их выражению лица, жестам и другим невербальным сигналам, которые мы видим.

Визуальное восприятие также имеет прямое отношение к нашему поведению, так как оно помогает нам ориентироваться в окружающей среде, находить предметы и избегать опасности. Кроме того, зрительное восприятие играет важную роль в нашем обучении и понимании новой информации, так как мы часто используем изображения для того, чтобы запоминать и классифицировать новые концепции.

## 2.2 Определение цветовосприятия

Восприятие цвета может оказывать сильное влияние на мнение человека, так как цвета могут вызывать различные ассоциации и эмоции. Данное исследование строится из предположения, что большинство объектов, признаков и понятий (а следовательно слов) можно ассоциировать с определенным цветом, рисунок 2.1.



Рисунок 2.1 – Таблица цветовых ассоциаций

Примеры влияния цвета на восприятие информации могут быть различными, их можно увидеть во многих сферах жизни. Вот некоторые из них:

Реклама: использование ярких и контрастных цветов может привлечь внимание к рекламе и вызвать положительные эмоции у зрителей.

Дизайн интерьера: цвета стен, мебели и декора могут создавать определенную атмосферу и настроение в помещении. Например, использование теплых цветов может создать ощущение уюта и комфорта, в то время как холодные цвета могут добавлять прохладу и спокойствие.

Спортивная форма: цвета формы могут создавать ощущение единства и командного духа. Например, команды могут использовать цвета своей формы для привлечения болельщиков и поддержки.

Упаковка продукта: выбор цвета упаковки может помочь выделить продукт на полке в магазине и привлечь внимание покупателей.

Образование: использование цвета в диаграммах и схемах может сделать информацию более понятной и легкой для запоминания.

Интернет-магазины: использование определенных цветов на сайте может помочь создать у покупателей ощущение доверия и профессионализма, что может повысить вероятность того, что они сделают заказ.

Транспорт: использование ярких цветов на автомобилях и общественном транспорте может помочь повысить видимость и безопасность на дороге.

Медицина: использование определенных цветов на стенах больницы и в комнатах пациентов может помочь создать более спокойную атмосферу и уменьшить стресс у пациентов.

Презентации: использование цвета в графиках и диаграммах на презентациях может помочь выделить ключевую информацию и сделать ее более запоминающейся.

Кино и телевидение: использование определенных цветовых фильтров и эффектов может помочь передать определенное настроение или эмоцию в фильме или телешоу.

Фотография: использование определенных цветовых схем и фонов может помочь создать определенную атмосферу и настроение в фотографии.

В целом, выбор и использование определенных цветов может быть очень мощным инструментом для передачи информации, создания настроения и эмоций.

Стоит так же выделить различные цветовые группы, например, теплые и холодные цвета, рисунок 2.2.



Рисунок 2.2 – Теплые и холодные цвета

Теплые цвета (красный, оранжевый, желтый, розовый) ассоциируются с теплом, энергией, весельем и горячим климатом. Они могут вызывать чувства страсти, радости и удовольствия. Холодные цвета (синий, зеленый, фиолетовый), скорее ассоциируются с прохладой, спокойствием, миром и холодным климатом. Они могут вызывать чувства прохлады, умиротворения и покоя.

Кроме того, теплые цвета обычно выглядят ближе к наблюдателю, чем холодные цвета. Это связано с тем, что длина волны теплых цветов более короткая, что приводит к более интенсивной отражаемости света. В то время как длина волны холодных цветов длиннее, что приводит к менее яркому отражению света.

В цветовой теории, использование теплых и холодных цветов может быть использовано для создания определенных настроений и эмоций в дизайне и искусстве.

Также в дизайне используют термины "тяжелые" и "легкие" цвета для определения визуального веса цветов и их воздействия на общее восприятие.

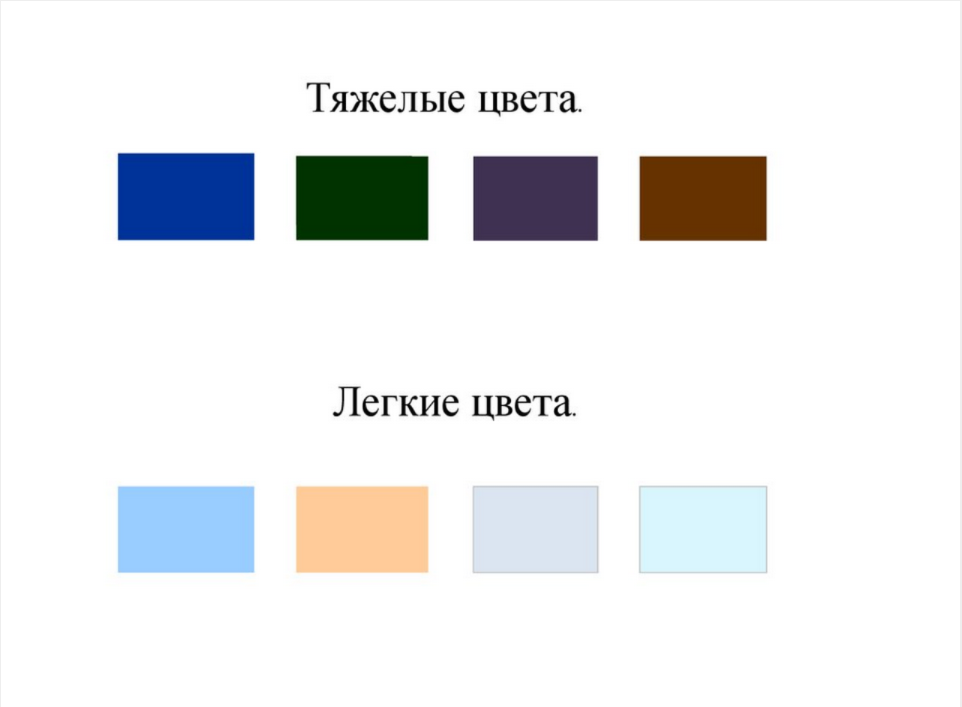


Рисунок 2.3 – Тяжелые и легкие цвета

Тяжелые цвета – это темные, глубокие цвета, которые могут создавать ощущение веса и стабильности. Они часто используются для придания предметам визуальной тяжести или солидности. К таким цветам относятся черный, темно-серый, коричневый, бордовый и глубокий зеленый.

Легкие цвета – это светлые, яркие цвета, которые создают ощущение легкости и нежности. Они могут использоваться для создания воздушности и элегантности. К таким цветам относятся белый, светло-серый, светло-зеленый, голубой, розовый и светло-желтый.

Определить, какие цвета можно отнести к тяжелым или легким, можно путем сравнения их насыщенности и яркости. Чем менее насыщен и ярок цвет, тем больше его визуальный вес. Например, темно-серый цвет может выглядеть тяжелым, если он очень насыщенный и яркий, в то время как светло-серый цвет может быть легким, если он менее насыщенный и более бледный. Однако, визуальный вес цвета также зависит от контекста, в котором он используется, и его сочетания с другими цветами.

Существует ряд цветовых сочетаний, которые особенно часто используют в рекламе, рисунок 2.4

Например, красный цвет на желтом фоне привлекает внимание благодаря своей контрастности и интенсивности. Красный является самым ярким из всех основных цветов, и он легко заметен даже на расстоянии. Желтый цвет также является ярким и заметным, что создает сильный контраст с красным.

Кроме того, эти цвета имеют разную символику, которая может привлекать внимание людей. Красный цвет ассоциируется с энергией, страстью, силой и возбуждением, в то время как желтый цвет символизирует радость, оптимизм, уверенность, тепло и дружелюбие. Комбинация этих двух цветов вызывает эмоции и чувства, которые могут привлечь внимание и запомниться людям.



Рисунок 2.4 – Цветовые сочетания, которые часто используют в области маркетинга

Такие цвета определенно привлекают к себе внимание за счет большого контраста, однако в данной работе такие сочетания использоваться не будут, исходя из предположения, что так как их восприятие достаточно резко провоцируется (что объясняет их частое применение в рекламе), но долго удерживать на них внимание будет некомфортно и быстро вызовет усталость.

## 2.3 Идеи, формирующие основу исследования и разработки

Исследование строится на основе следующих идей и предположений:

* Современные способы представления информации требуют больше работы, а также более индивидуального подхода.
* Текстовая информация может быть обработана с помощью информационных систем и алгоритмов, для придания ей большей эмоциональной окраски, используя визуальные приемы – работу с цветом, шрифтом, фоном.
* Обработка текстовой информации с целью деления предложения на логические части и выделение «якорных» точек может помочь упростить и тем самым ускорить восприятие.
* Представление информации сразу в нескольких её типах может положительно повлиять на комфорт и скорость её восприятия.

## 2.4 Выбор средств реализации

Для написания веб-приложения будет использован React и среда разработки Visual Studio Code. Для контроля версий планируется использовать GIT.

React - библиотека для создания пользовательских интерфейсов, которая позволяет разработчикам строить масштабируемые и быстрые веб-приложения с использованием модульной архитектуры. Одним из главных преимуществ React является его декларативный подход к описанию пользовательского интерфейса, который позволяет легко изменять состояние приложения и перерисовывать компоненты только при необходимости. Это помогает улучшить производительность приложения и уменьшить общее количество кода. Кроме того, React также предлагает широкий спектр инструментов для отладки, тестирования и развертывания приложений, что делает его очень удобным для работы над любыми проектами.

Visual Studio Code (VS Code) - это популярный бесплатный редактор кода, который предоставляет широкий спектр инструментов разработки для создания веб-приложений. Одним из главных преимуществ VS Code является его расширяемость и наличие большого числа плагинов, которые могут помочь улучшить производительность и обеспечить более комфортную работу над проектом. VS Code предоставляет удобный интерфейс пользователя, который позволяет легко открывать и управлять несколькими файлами одновременно, быстро переключаться между различными проектами и осуществлять навигацию внутри кода. Также VS Code имеет интеграцию с различными технологиями и сервисами, что делает его очень удобным для работы над любыми проектами. И в целом, VS Code является популярным инструментом среди разработчиков благодаря своей производительности, функциональности и богатому экосистеме расширений.

Git - распределенная система контроля версий, которая позволяет отслеживать изменения в коде и совместно работать над проектом. Она является одним из наиболее популярных инструментов разработчиков благодаря своей простоте, мощности и гибкости. Одним из главных преимуществ Git является его возможность сохранять все версии приложения в репозитории, что делает управление изменениями более простым и эффективным. Это также позволяет быстро и легко переключаться между различными версиями приложения и отменять изменения в случае ошибок. Кроме того, Git обеспечивает эффективную командную работу над проектом, позволяя разработчикам одновременно работать над одним и тем же кодом, не задевая другие части кода. Это улучшает производительность и повышает качество кода. Git также предоставляет различные инструменты для слияния и синхронизации кода, обеспечивая более гладкое сотрудничество между разработчиками. Кроме того, Git имеет широкий выбор хостинг-платформ, таких как GitHub или GitLab, которые предоставляют инструменты для управления проектами и совместной работы над кодом.

# 3 Детальная проработка конкретного элемента, входящего в общую концепцию решения.

## 3.1 Описание функционала приложения с помощью use-case диаграмм

Для формализации требований к функционалу приложения была составлена use-case диаграмма. Верхний уровень диаграммы представлен на рисунке 3.1.

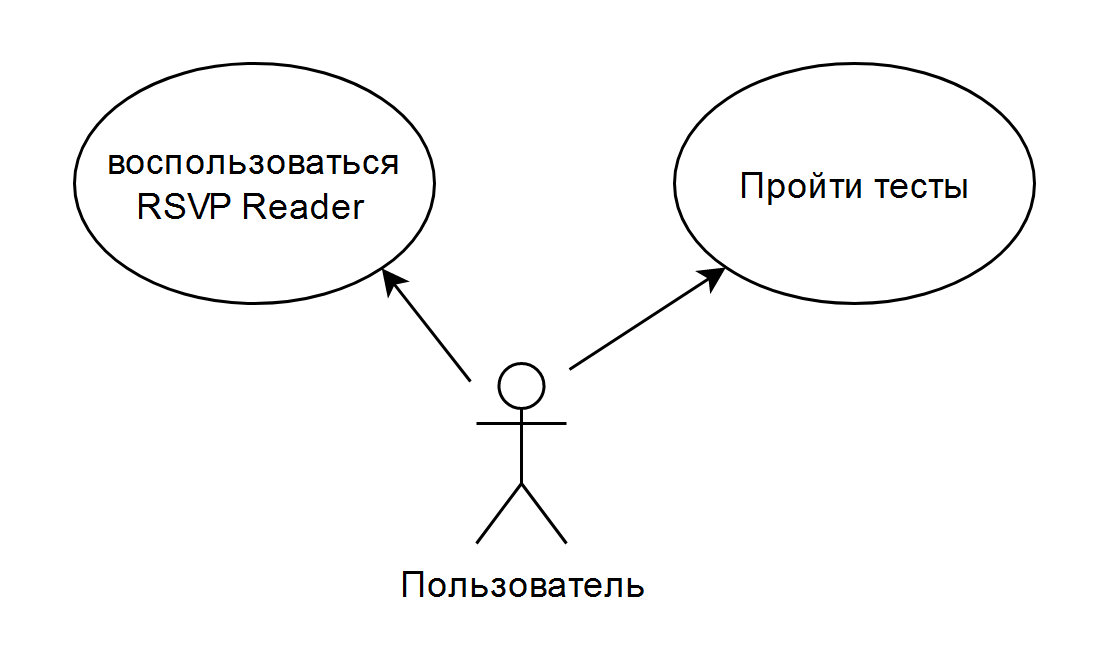


Рисунок 3.1 – Верхний уровеньUse-Case диаграммы

Пользователю приложения доступны 2 основных функционала – использовать приложение и принять участие в исследовании – пройти тестирование.

Декомпозиция по ветке «воспользоваться RSVP Reader» представлена на рисунке 3.2.

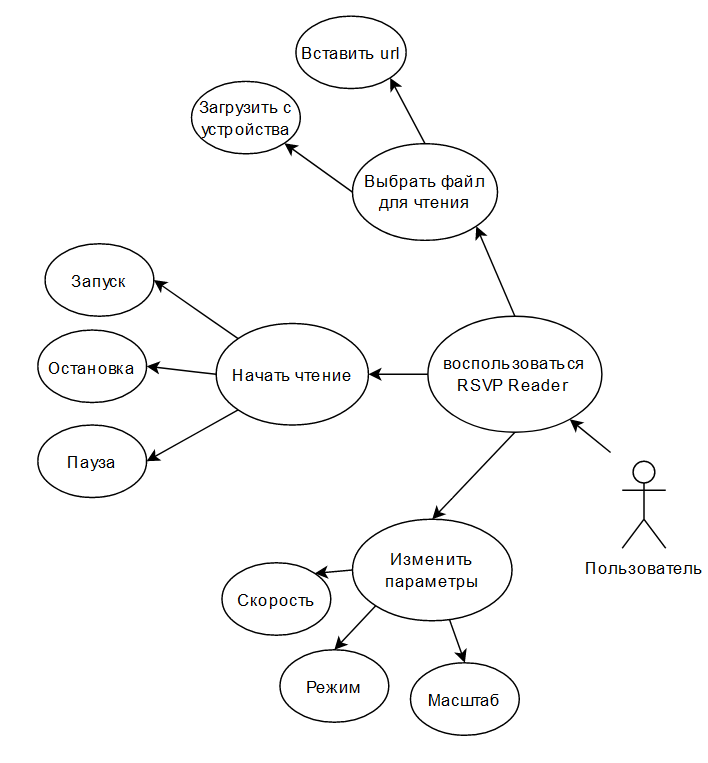


Рисунок 3.2 – Декомпозиция use-case диаграммы, первый уровень, функционал «Воспользоваться RSVP Reader»

Данная декомпозиция отображает минимально необходимый функционал модуля чтения.

В первую очередь пользователю должна быть доступно пространство для чтения – область, где будут непосредственно отображаться фрагменты текста. Демонстрацию фрагментов можно начать, остановить, либо поставить на паузу. Возможен дополнительный функционал, связанный с особыми режимами работы модуля.

Также, к основному функционалу можно отнести функционал, связанный с управлением текстовым контентом – возможность загрузки файла из локальной памяти устройства, либо возможность вставки url, содержащего текстовый контент - файл, либо html страницу с текстом.

Для повышения гибкости и удобства использования приложения, основные параметры должны быть вынесены в пользовательский интерфейс, предоставляя возможность изменения:

* Скорости чтения – регулируется среднее количество слов в минуту, которое будет показано;
* Масштаба – регулируется размер основных и управляющих компонентов;
* Режима – возможность переключения между, например, обычным чтением и экспериментальным, включающим в себя разработанные в ходе работы дополнения текстового контента.

Модуль тестирования включает в себя геймифицированый сбор информации и систему тестирования для определения доминирующего типа восприятия информации, а также исследования факторов, влияющих на комфортность восприятия, помимо биологически доминирующего типа.

Декомпозиция use-case диаграммы для функционала «Пройти тесты» представлена на рисунке 3.3.

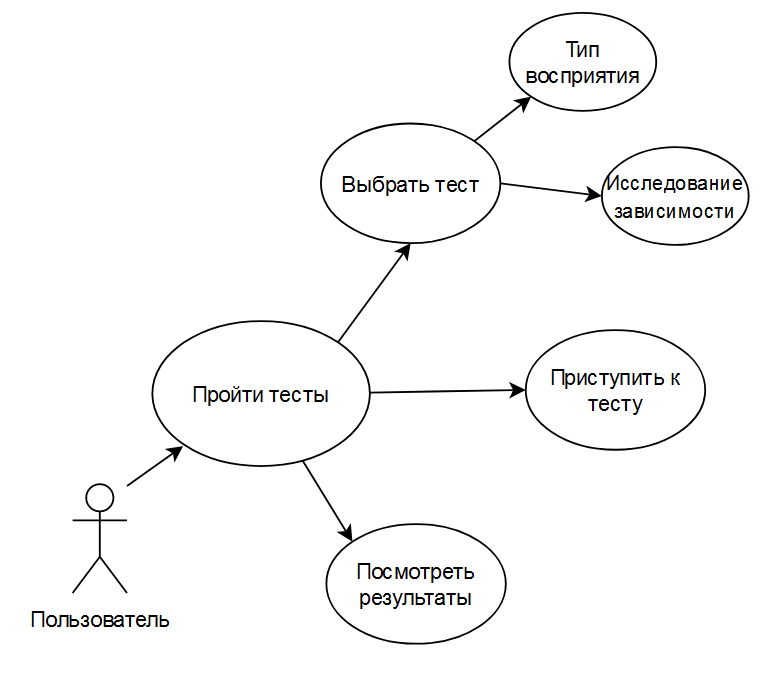


Рисунок 3.3 – Декомпозиция use-case диаграммы, первый уровень, функционал «Пройти тесты»

Тест представляет собой игру на память – пользователю показываются объекты на определенное время, после чего необходимо восстановить их наибольшее количество. Для того, чтобы тест нес исследовательскую ценность, понадобится проходить тест в несколько итераций.

В таблице ниже (Таблица 3.1) представлены предлагаемые режимы тестирования, объект – то, что предстоит запоминать, цветовая гамма – описание красочности объекта, количества используемых цветов.

Таблица 3.1 – Вариации тестовых режимов

|  |  |
| --- | --- |
| Объект | Цветовая гамма |
| Текст | нейтральная |
| Текст | Подходящая по смыслу текста (окрашивается сам текст) |
| Текст | Подходящая по смыслу текста (окрашивается фон, текст имеет нейтральный цвет, читабельный на подобранном фоне) |
| Текст  + изображение, подходящее по смыслу текста | Текст – нейтральная  Изображение - исходная |
| Изображение | Исходная цветовая гамма |
| Изображение | Эскизный стиль – только черный контур, цвета нейтральные |
| Изображение + аудио | Изображение |
| Аудио | Нейтральный фон |
| Текст + аудио | Нейтральный фон |
| Текст + аудио+изображение | Изображение - исходный |

Данное тестирование можно будет использовать для

* исследования общих тенденций и склонностей к разным типам восприятия,
* определения собственного доминирующего типа восприятия,
* повышения эффективности обучения, например, иностранному языку.

## 3.2 Планирование архитектуры приложения

Так как предполагается, что приложение не будет работать с большими массивами данных и хранить их – отсутствует необходимость в проектировании базы данных. Кроме того, реализация основного алгоритма – представления текста фрагментами требует высокой скорости, поэтому не имеет смысл выносить этот функционал в backend часть приложения.

Однако, база данных может понадобится при масштабировании исследования – для хранения его результатов по каждому пользователю и добавления возможности просмотра общей статистики результатов прохождения тестов.

На рисунке ниже представлена общая схема взаимодействия частей приложения, функционал «чтение» (Рисунок 3.4).

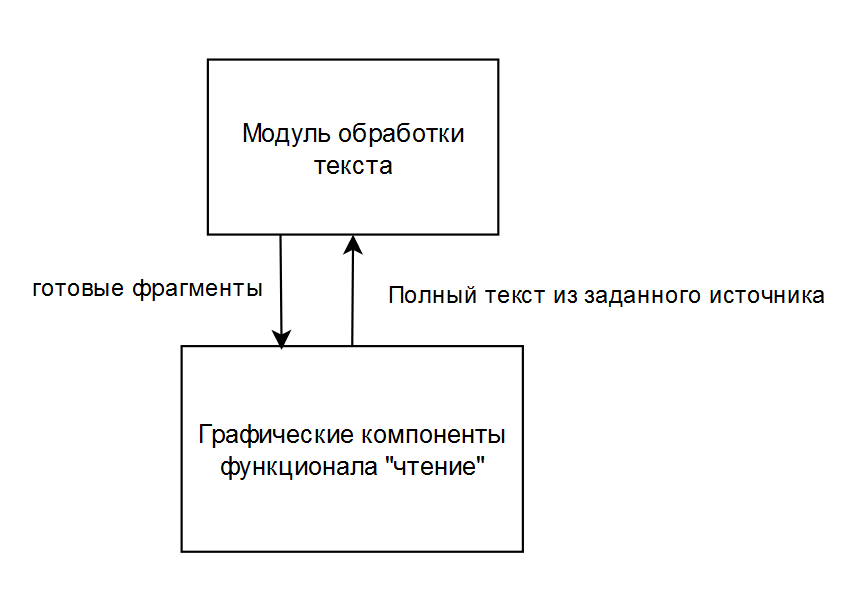


Рисунок 3.4 – Общая схема взаимодействия компонентов приложения для функционала «чтение»

Основной функционал входит в модуль обработки текста, целью графических компонентов является визуализация и предоставление компонентов для получения исходного текста, который затем передается в обработчик.

**Модуль обработки текста**

В модуле обработки текста должна быть реализована функциональность разделения текста на фрагменты, согласно методике, описанной в главе 1, а именно распознание слов, распознание знаков препинания и их грамотное представление. В данной работе алгоритм фрагментации не был написан с нуля, так как уже имеет ряд имплементаций с открытым исходным кодом.

На рисунке ниже представлена общая схема взаимодействия частей приложения, функционал «тесты» (Рисунок 3.5).



Рисунок 3.4 – Общая схема взаимодействия компонентов приложения для функционала «тесты»

Основной функционал входит в модули:

* Модуль обработки результатов теста
* Модуль подготовки тестирования

**Модуль подготовки тестирования**

В данном модуле формируется тестовый набор, исходя из заданных пользователем параметров – режима, и, возможно, результатов прошлого тестирования.

**Модуль обработки результатов тестирования**

Модуль принимает сырые данные для их последующего анализа – в него поступают численные результаты каждой итерации теста, после анализа которых модуль возвращает агрегированную статистику и выводы по тестированию.

## 3.3 Проектирование на уровне компонентов приложения

Ниже представлены схемы реализации функционала на уровне компонентов – рисунок 3.3.

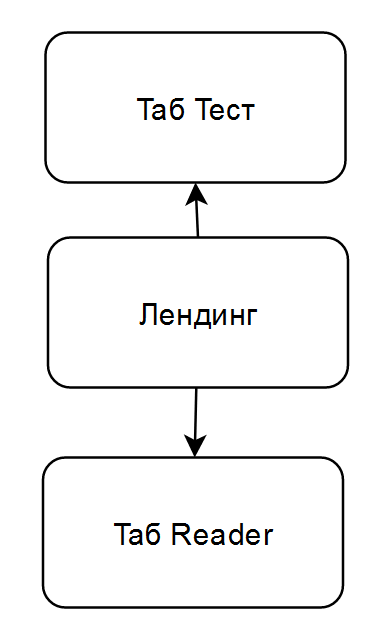


Рисунок 3.3 – Высокоуровневые компоненты веб-приложения

Лендинг – основная страница, содержащая навигацию по веб-приложению и его описание. Соответственно, от лендинга доступен переход к табам с функционалом по модулям – таб «Reader» и таб «Тест».

Компоненты, реализующие функционал модуля чтения (таб «Reader») представлены на рисунке 3.4.

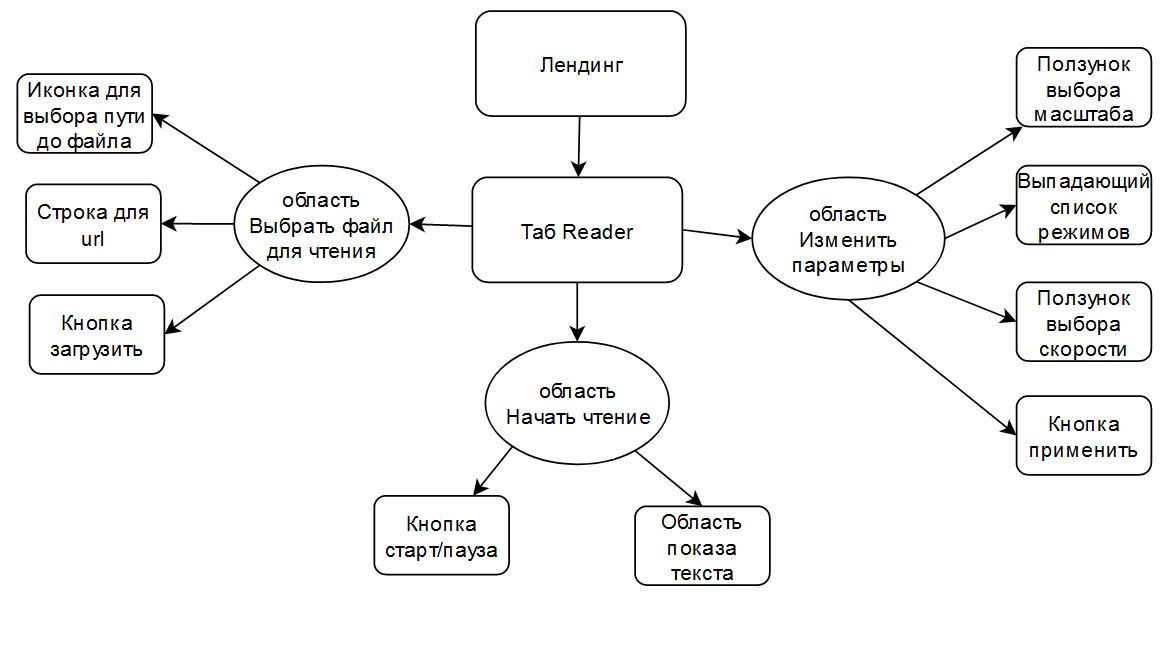


Рисунок 3.4 – Компоненты, реализующие функционал «Воспользоваться RSVP Reader»

Должны быть элементы, реализующие сам функционал –

* Область показа текста
* Кнопка старт/пауза

Помимо этого, понадобятся компоненты для загрузки самого текста и выбора параметров его показа.

На рисунке 3.5 представлены компоненты для функционала «Пройти тестирование».

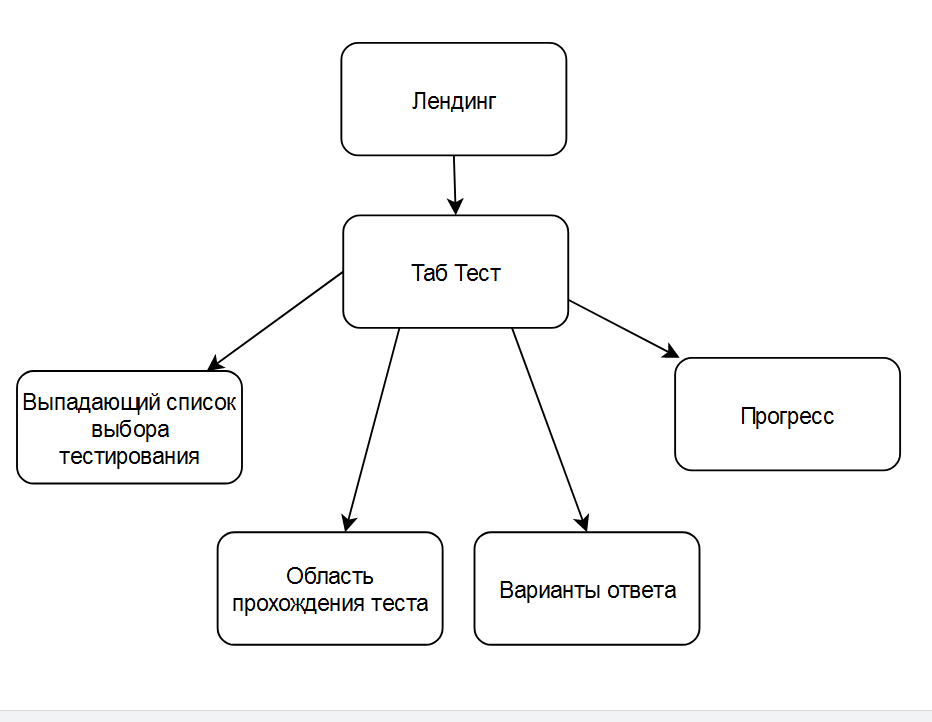


Рисунок 3.5 – Компоненты, реализующие функционал «Пройти тестирование»

Выпадающий список выбора тестирования – предоставляет возможность выбора одного из типов тестирования.

Прогресс – визуализация процесса прохождения теста вида «stepper».

Под областью прохождения теста в данном случае подразумевается фиксированная область, в которой пользователю будут покатываться объекты для запоминания.

Варианты ответа – компоненты, предоставляющие возможность выбрать очередность показанных объектов.

## 3.4 Проектирование интерфейса приложения

Интерфейс приложения был спроектирован в Figma, использовались компоненты из UI Kit “Desktop components” и плагины Color wheel, color palette.

Общая структура страницы лендинга представлена на рисунке 3.6.

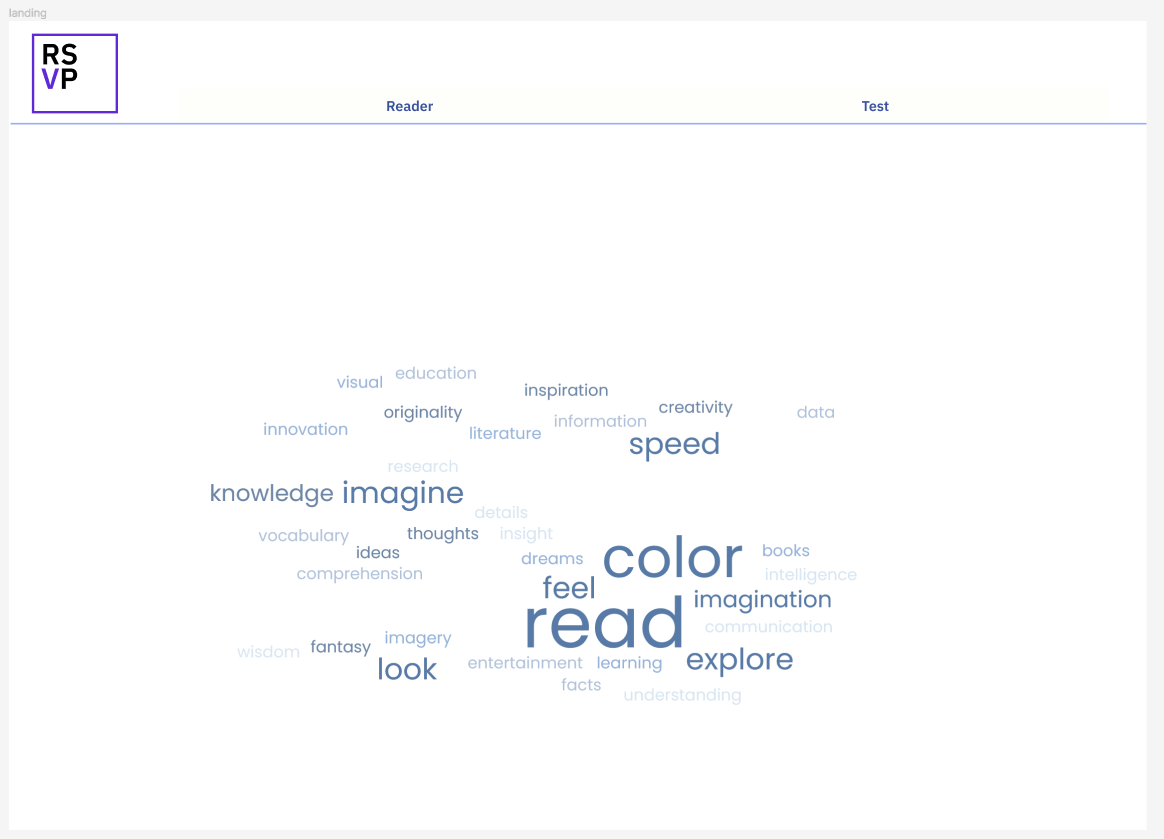


Рисунок 3.6-Лендинг

Здесь пользователю при необходимости показана общая информация о приложении и информация о функционале. Также доступна возможность перехода к вкладкам с основным функционалом.

При переводе таба в положение “Reader”, пользователь переходит в режим чтения рисунок 3.7.

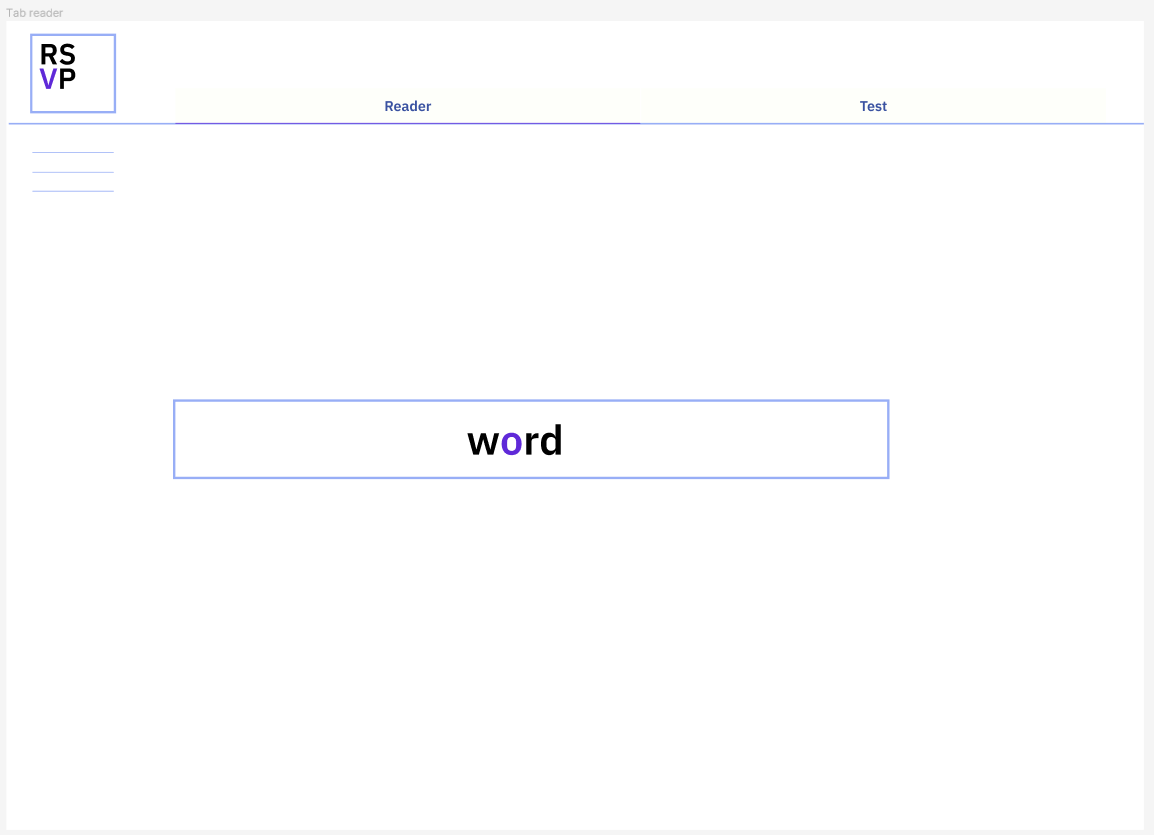


Рисунок 3.7 - Таб в положении Reader

Здесь пользователю доступно меню с основными параметрами и область чтения.

На рисунке 3.8 представлена функциональность меню.

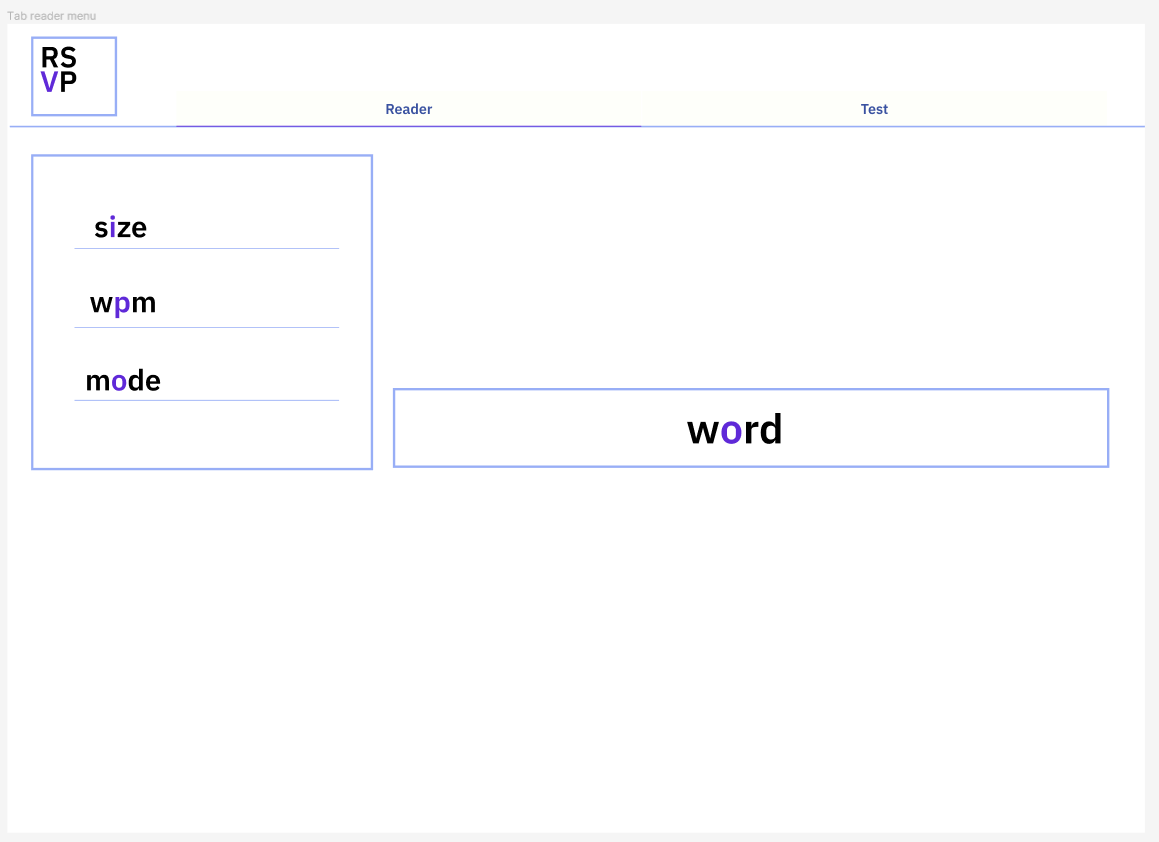


Рисунок 3.8 – Меню настройки Reader

Здесь пользователь может изменить настройки масштаба, режима и количестве показываемых слов в минуту.

Меню не должно занимать много места, раскрывается только на время регулирования параметров. Параметры должны применятся без дополнрительных действий, чтобы можно было сразу подобрать необходимые скорость и масштаб для комфортного чтения.

При переводе таба в положение “Test”, пользователь переходит в режим тестирования рисунок 3.9.

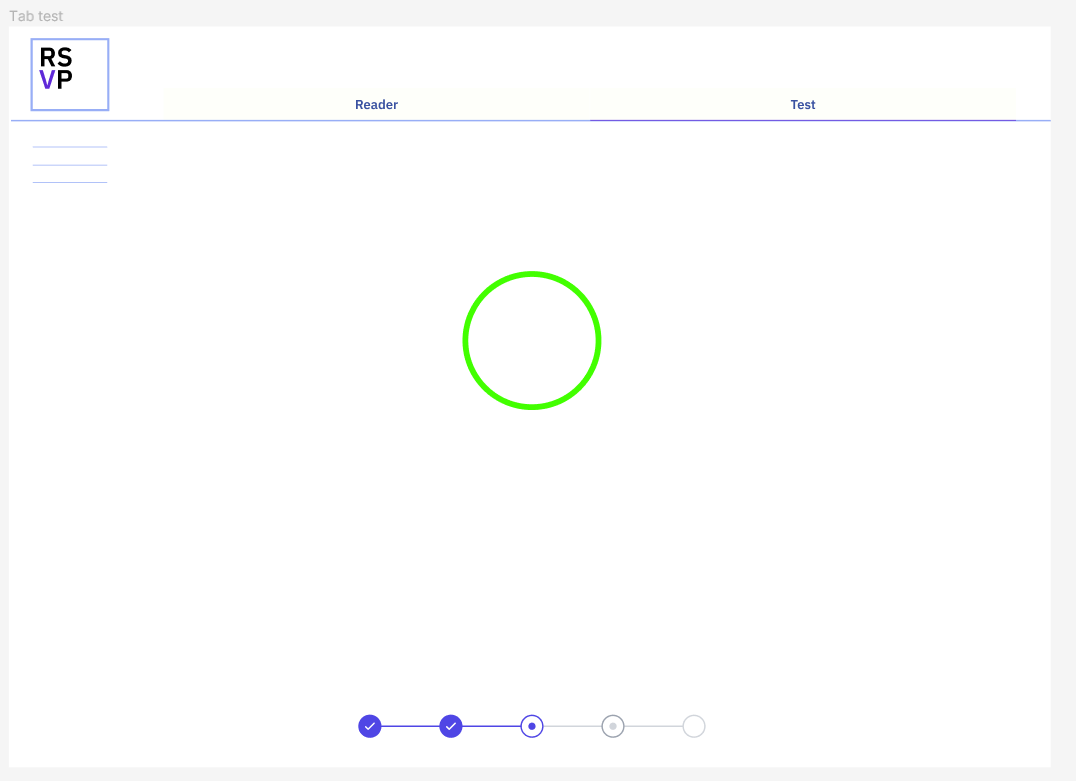


Рисунок 3.9 - Таб в положении Test

В центре, по аналогии с режимом чтения, пользователю показываются объекты, порядок которых ему потом предстоит воспроизвести по памяти.

Меню активно только на момент выбора режима тестирования, затекм сворачивается, чтобы не акцентировать на себе лишнее внимание.

Степпер в нижней части экрана позволит ориентироваться по количеству пройденных и предстоящих итераций.

## 3.5 Описание процесса разработки

Основа представления текста по технологии RSVP будет взята из открытых open-source проектов, по причине того, что тема достаточно хорошо исследована, имеет несколько способов реализации и данная работа не преследует цели создать алгоритм представления текстовой информации.

Будет написана frontend часть приложения, для возможности регулировать способы представления.

Помимо этого, потребуется доработка backend части – необходимо спроектировать алгоритмы, позволяющие ассоциировать текстовые значения с графическим представлением – цветом/изображением, а также для возможности представления необходимой аудиоинформации.

## 3.6 Описание классов и объектов

Для того, чтобы дополнить функционал чтения по методологии RSVP необходимыми экспериментальными функциями, недостаточно будет представления слова в качестве текстового объекта, потребуется расширить его структуру до объекта («WordItem»), дополнив полями.

Поля представляют из себя расширенный вариант представления слова, дополнение его признаками, с целью упростить восприятие каждого отдельно взятого слова.

Предлагается следующая структура объекта «word», рисунок 3.10

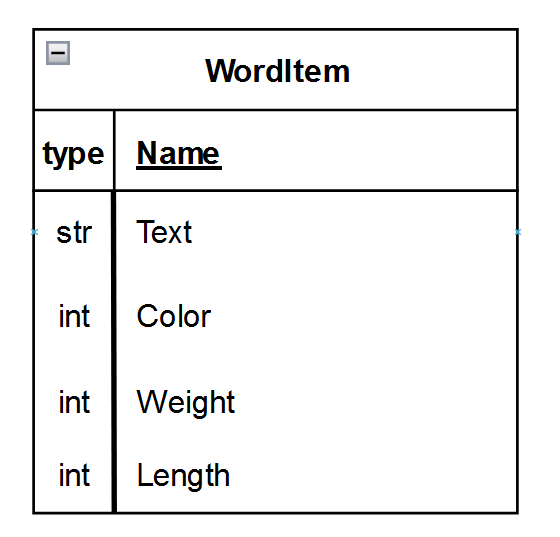


Рисунок 3.10 – структура объекта класса «WordItem»

**Text**

Исходное текстовое представление слова, служит источником данных для остальных признаков.

**Color**

Цветовая ассоциация слова, определяемая с помощью нейросети. Не все слова имеют общепринятую цветовую ассоциацию, однако исследование строится из предположения, что большинство слов можно ассоциировать с определенным цветом. Понадобится для режима чтения, когда слова будут иметь цвет, либо цветной фон.

**Weight**

Мера для обозначения веса, насколько важно слово относительно других в текущем контексте, ставится ли на него логическое ударение. Понадобится для режима, когда текст отличается шрифтом и размером, для расстановки дополнительных акцентов.

**Length**

Длина слова, понадобится для правильного представления слова, в зависимости от его размера – короткие слова воспринимаются быстрее, следовательно время на чтение такого слова можно уменьшить (относительно заданного пользователем). Также может возникнуть необходимость значительного понижения скорости на время прочтения новой сложной терминологии, а так же многосоставных слов.

Для проведения тестирования также потребуется особая структура объектов, рисунок 3.11.

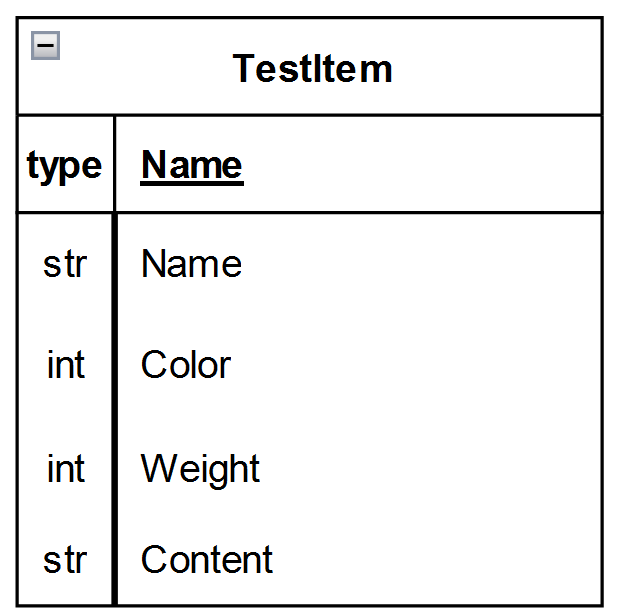


Рисунок 3.11 – структура объекта класса «TestItem»

**Name -** Название фигуры/картинки, для представления в статистике.

**Color -** Ассоциируемый, либо закрепленный цвет. В разных режимах может соответствовать как ассоциативному цвету, как в WordItem, так и навязанной цветовой ассоциации с целью расширения количества признаков конкретного объекта, для его лучшего запоминания.

**Weight -** Относительный акцент на текущем объекте, объекты с большим весом показываются в большем масштабе относительно других. Добавляется для исследования зависимости восприятия от размера текста и шрифта.

**Content** **-** Сама картинка или фигура, которая будет показана на экране.

## 3.7 Масштабирование и расширение функционала

Возможны следующие варианты расширения функционала:

* Возможность загружать контент в аудио форматах
* Возможность загружать контент в формате ссылки на видео/аудио запись
* Возможность загружать контент в формате ссылки на html страницу
* Обработка html страницы – дифференциация медиаконтента, распознавание заголовков, удобное отображение списков и таблиц, подписей к картинкам.

# 4 Практическая реализация приложения

## 4.1 Демонстрация работы приложения

В процессе разработки было принято решение, что темная цветовая гамма будет меньше отвлекать пользователя от просмотра контента, поэтому итоговый вариант интерфейса выглядит темнее изначальных макетов.

Первое, что видит пользователь – лендинг с описанием работы.



Рисунок 4.1- Страница лендинга

Здесь указана основная информация о проекте, а так же возможно размещение SEO-тегов, для облегчения обнаружения приложения поисковыми сервисами.

Далее пользователь может перейти в режим чтения, рисунок 4.2.

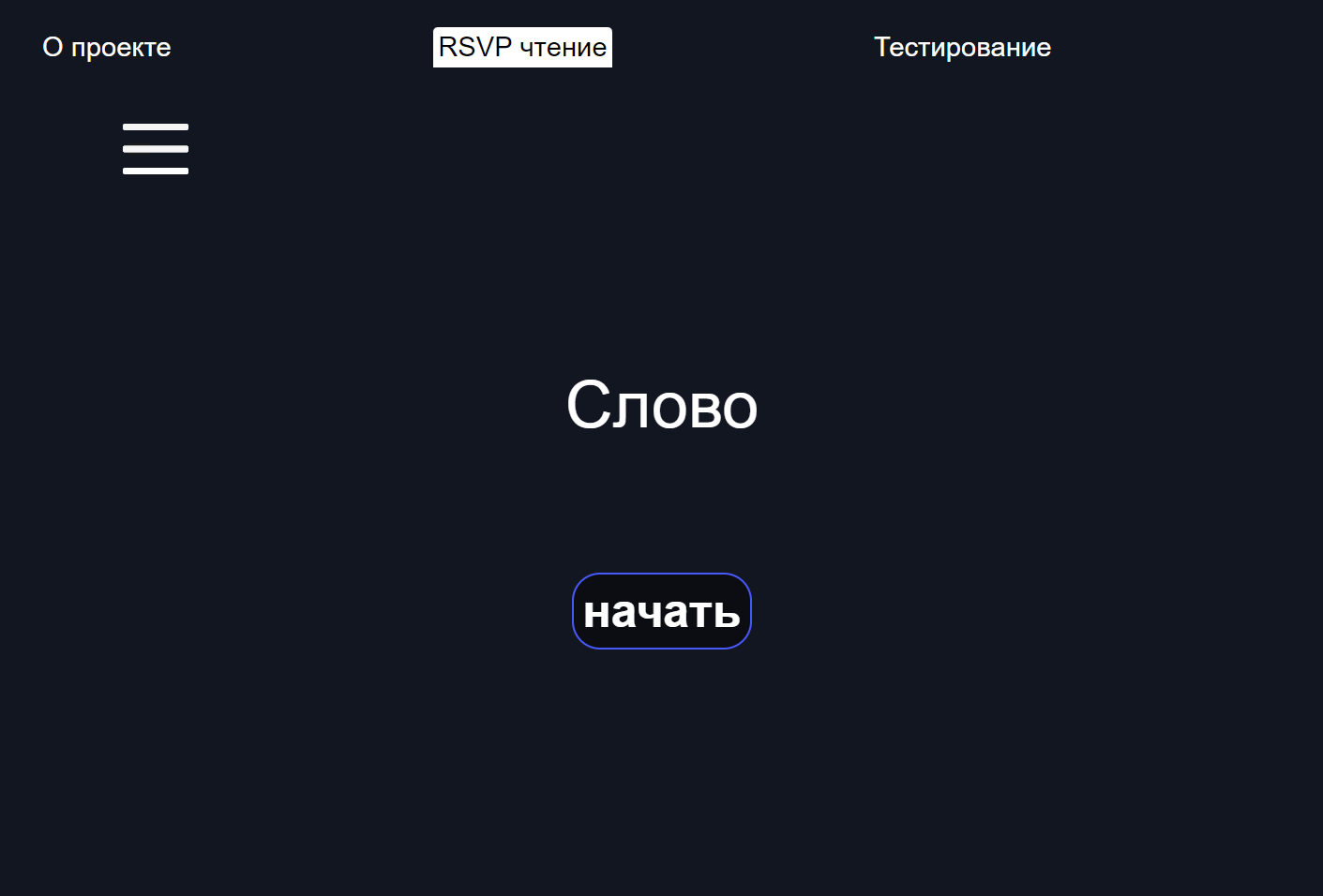


Рисунок 4.2- Режим чтения

Здесь можно увидеть основные элементы - кнопка перехода к боковому меню с параметрами, кнопка начала показа, сама область демонстрации текста.

Предполагается, что во время чтения элементы становятся менее насыщенными, чтобы не привлекать на себя лишнее внимание пользователя.

При нажатии на конопку “начать”, начинается воспроизведение текста с заданной скоростью, рисунок 4.3.

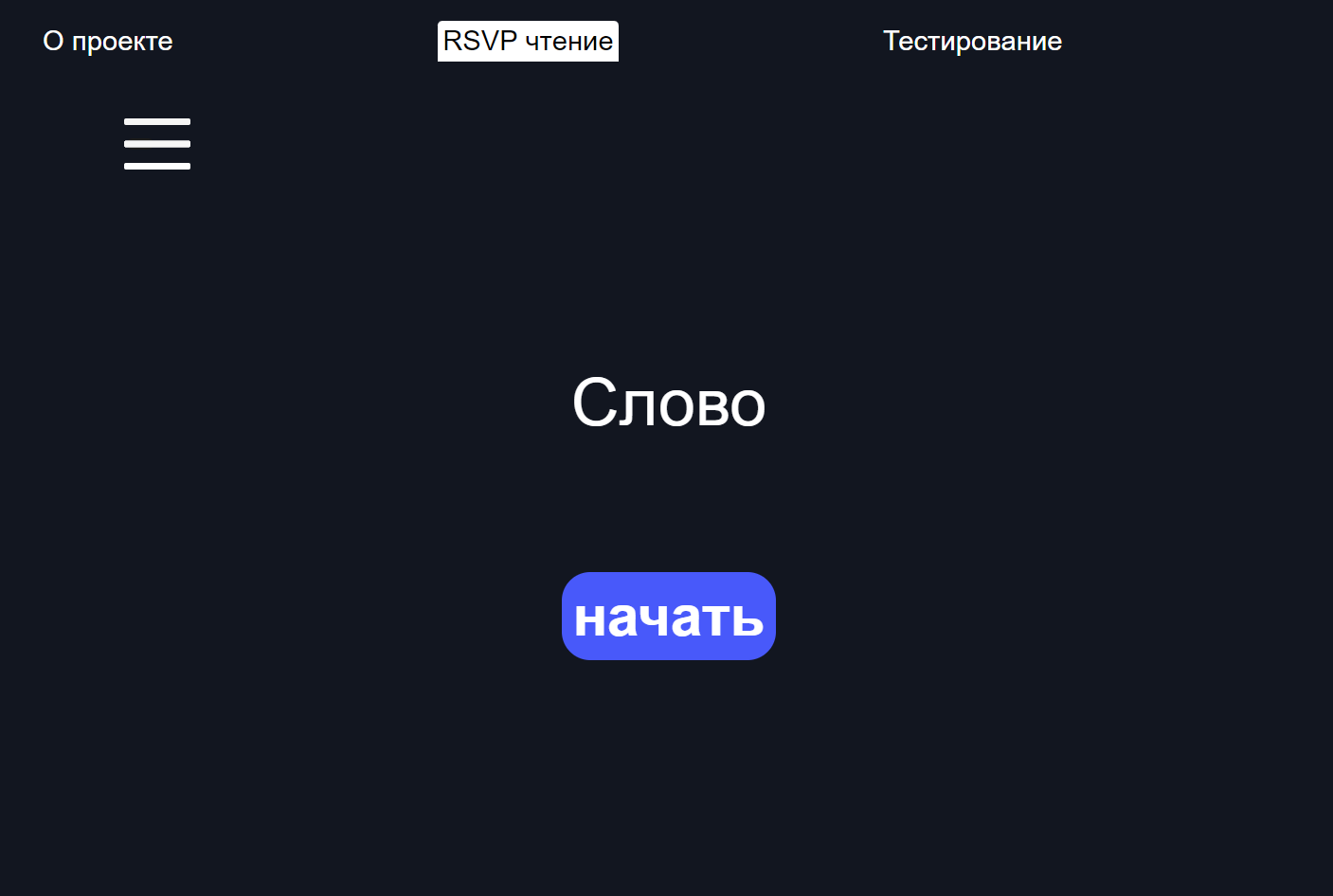


Рисунок 4.3- Активная кнопка “начать”

В боковом меню есть возможность регулировать необходимые параметры показа, рисунок 4.4.

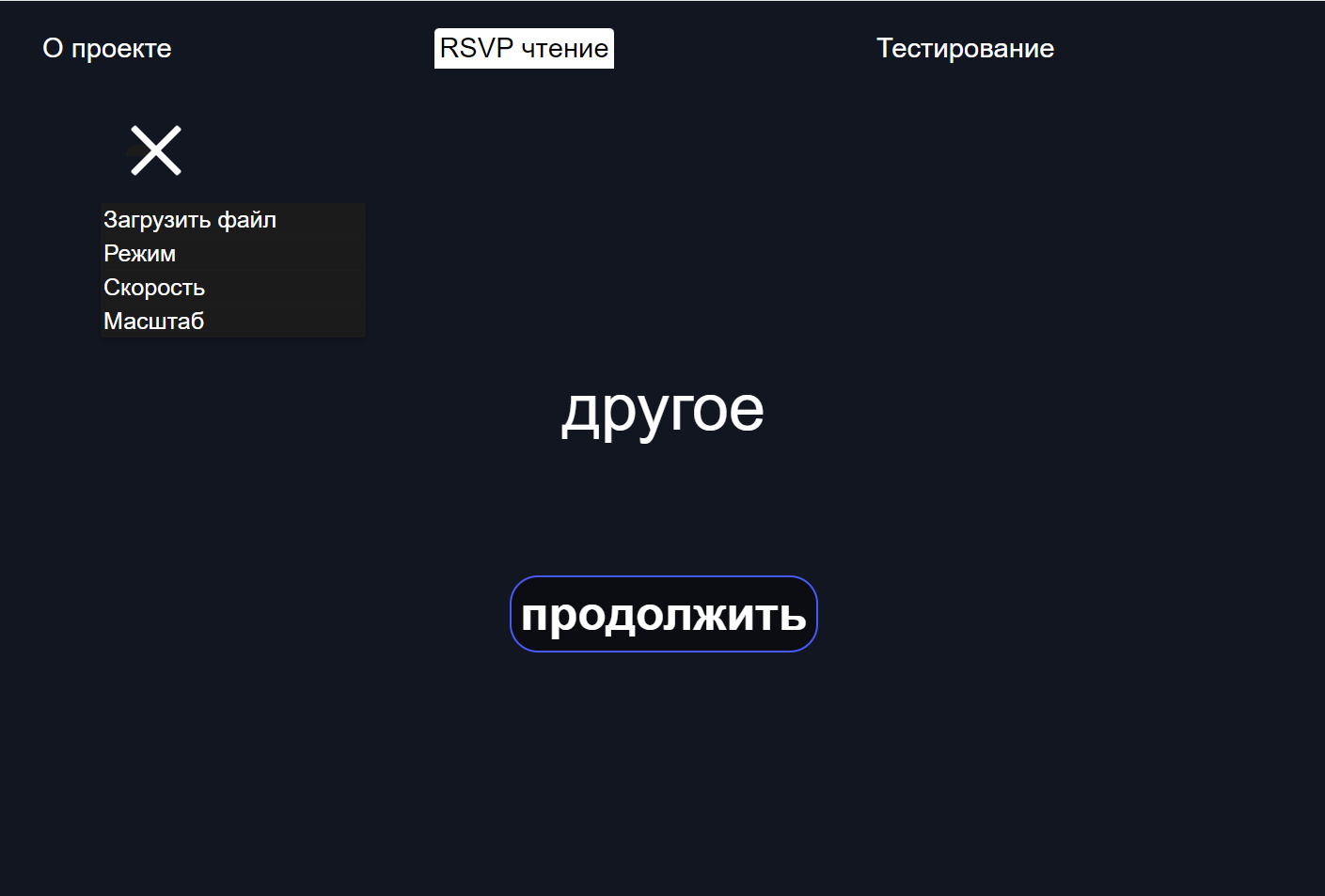


Рисунок 4.4 - Боковое меню

В боковом меню можно загрузить файл, настроить масштаб, режим и скорость чтения.

Так же через переключение вкладки можно перейти к тестам, рисунок 4.5.

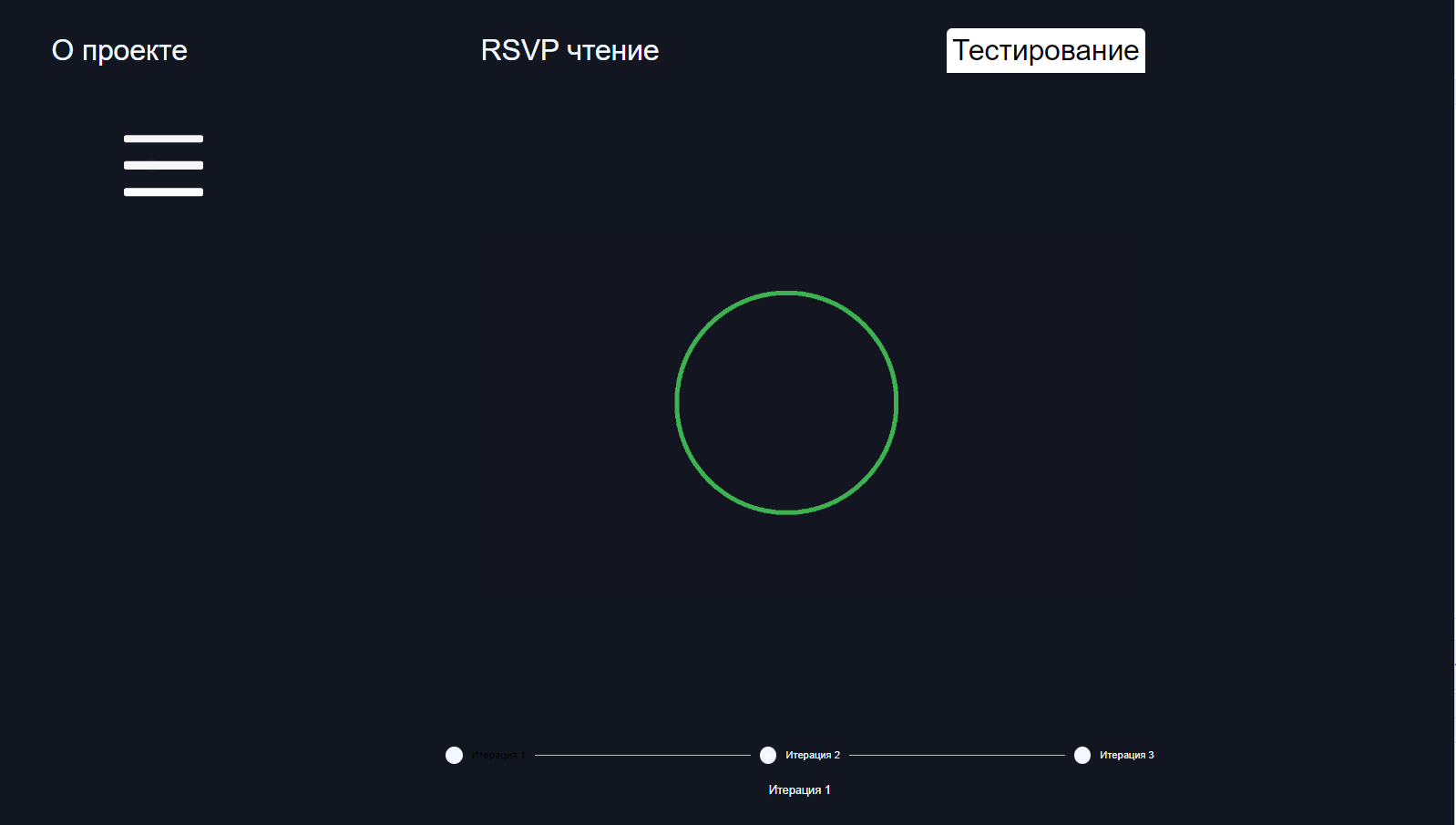


Рисунок 4.5 - Режим тестирования

Здесь пользователю демонстрируются объекты, после чего предлагается по памяти восстановить их последовательность.

# 

# ЗАКЛЮЧЕНИЕ

Таким образом, в ходе выполнения работы были выполнены поставленные задачи:

* Проведен углубленный анализ предметной области;
* Раскрыты такие темы, как методология RSVP и основы психологии цветовосприятия;
* Проанализированы сильные и слабые стороны существующих решений, реализующих методологию чтения Rapid speed visual presentation;
* Разработаны способы их возможного усовершенствования;
* Исследованы возможности дополнительного воздействия на восприятие информации с помощью использования цветовосприятия и подсознательного внимания пользователя;
* Разработано приложение, демонстрирующее результаты исследования, и способствующее сбору информации для его продолжения.

# СПИСОК ЛИТЕРАТУРЫ

1. Черниговская Татьяна Владимировна. Регистрация движений глаз в психолингвистических исследованиях, 2019. — 229 с. — ISBN 978-5-288-05882-0 –
2. Мардан Азат. React быстро. Веб-приложения на React, JSX, Redux и GraphQL. — СПб.: «Питер», 2019. — С. 560. — ISBN 978-5-4461-0952-4.
3. Бэнкс Алекс, Порселло Ева. GraphQL: язык запросов для современных веб-приложений. — СПб.: «Питер», 2019. — С. 240. — ISBN 978-5-4461-1143-5.
4. Бэнкс Алекс, Порселло Ева. React и Redux: функциональная веб-разработка. — СПб.: «Питер», 2018. — С. 336. — ISBN 978-5-4461-0668-4.
5. Томас Марк Тиленс. React в действии. — СПб.: «Питер», 2019. — С. 368. — ISBN 978-5-4461-0999-9.
6. Кирупа Чиннатамби. Изучаем React. — СПб.: «Питер», 2019. — С. 368. — ISBN 978-5-04-098028-4.
7. Василевич А.П. Синий, синий, голубой... Или всегда ли слово было изгоем? [Электронный ресурс] / Режим доступа: http://www.ruscenter.ru/626.html
8. Василевич А.П., Кузнецова С.Н., Мищенко С.С. Каталог названий цвета в русском языке. – М., 2002.
9. Садыкова И.В. Обозначение красного цвета в русском языке в историкоэтимологическом аспекте. Автореферат дисс. ... к.филол.н. – Томск, 2006. 4. Bryson, Bill. Shakespeare. The Illustrated Edition. – Harper Press, 2009.
10. . ГОСТ 9241-151-2014 «Эргономика взаимодействия человек — система. Часть 151. Руководство по проектированию пользовательских интерфейсов сети Интернет».
11. ГОСТ 9241-210-2016 «Эргономика взаимодействия человек — система. Часть 210. Человеко-ориентированное проектирование интерактивных систем».
12. Долгих М. Н. Дизайн и виртуальная среда: дигитальные ландшафты в аспекте гибридных медиа // Вестник Томск. гос. ун-та. 2014. № 379. С. 86–91.
13. Ходин А.В., Шуклин Д.А., Шалобаев Е.В. Особенности UX-дизайна интернет-проектов // Альманах научных работ молодых ученых университета ИТМО: Статья в сборнике трудов конференции. — 2016.
14. Cassandra R., 38. Usability Testing — 2016. Technical Communications.
15. Davis F. D., Bagozzi R. P., Warshaw P. R., User acceptance of computer technology: A comparison of two theoretical models, Management Science, vol. 35, no. 8, pp. 982-1003. — 1989.
16. Davis F. D., Perceived usefulness, perceived ease of use, and user acceptance of information technology, MIS Quarterly, vol. 13, no. 3, pp. 319-340. — 1989.
17. Е. Н. Косова. Гуманистические ценности и этическая направленность дизайна в культуре модернизма. Феномен инклюзивного дизайна [Электронный ресурс] // Сборник трудов международной научнопрактической конференции «Инновации и дизайн». 2020.
18. Патнюк И.В., Универсальный дизайн — инновационная стратегия безопасности и сохранения окружающей среды [Электронный ресурс] //Актуальные проблемы дизайна и дизайн-образования: материалы III Междунар. науч.-практ. конф., Минск. 2019. URL:https://elib.bsu.by/handle/123456789/239136 (Дата обращения: 03.06.2021).
19. Ландэ Д. В., [Снарский А. А.](https://ru.wikipedia.org/wiki/%D0%A1%D0%BD%D0%B0%D1%80%D1%81%D0%BA%D0%B8%D0%B9,_%D0%90%D0%BD%D0%B4%D1%80%D0%B5%D0%B9_%D0%90%D0%BB%D0%B5%D0%BA%D1%81%D0%B0%D0%BD%D0%B4%D1%80%D0%BE%D0%B2%D0%B8%D1%87), Безсуднов И. В. [Интернетика: Навигация в сложных сетях: модели и алгоритмы](http://webground.su/services.php?param=book&part=internetica_content.htm) — M.: Либроком (Editorial URSS), 2009. — 264 с. – Текст: непосредственный.
20. Ландэ Д. В., Снарский А. А., Безсуднов И. В. Интернетика: Навигация в сложных сетях: модели и алгоритмы. — M.: Либроком (Editorial URSS), 2009. — 264 с. — ISBN 978-5-397-00497-8. – Текст: непосредственный.
21. Одинцов И. О. Профессиональное программирование. Системный подход. — 2-е изд.. — СПб.: БХВ-Петербург, 2004. — 624 с. – Текст: непосредственный.
22. Куликов Г.Г. Набатов А.Н. Речкалов А.В. Автоматизированное проектирование информационно-управляющих систем. Системное моделирование предметной области. — Уфа: Уфимский государственный авиационный технический университет, 1998. — 104 с. – Текст: непосредственный.
23. Крэг Ларман. Применение UML 2.0 и шаблонов проектирования — 3-е изд. — М.: Вильямс, 2006. — 736 с. – Текст: непосредственный.
24. Джозеф Шмуллер. Освой самостоятельно UML 2 за 24 часа. Практическое руководство— М.: Вильямс, 2005. — 416 с. – Текст: непосредственный.
25. Василевич А.П. Синий, синий, голубой... Или всегда ли слово было изгоем? [Электронный ресурс] / Режим доступа: http://www.ruscenter.ru/626.html
26. Василевич А.П., Кузнецова С.Н., Мищенко С.С. Каталог названий цвета в русском языке. – М., 2002.
27. Садыкова И.В. Обозначение красного цвета в русском языке в историкоэтимологическом аспекте. Автореферат дисс. ... к.филол.н. – Томск, 2006. 4. Bryson, Bill. Shakespeare. The Illustrated Edition. – Harper Press, 2009.

ПРИЛОЖЕНИЕ А

Листинг программы

**index.js**

import React from 'react';

import ReactDOM from 'react-dom/client';

import './index.css';

import App from './App/App';

import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(

<React.StrictMode>

<App />

</React.StrictMode>

);

reportWebVitals();

**index.css**

body {

margin: 0;

font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',

'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',

sans-serif;

-webkit-font-smoothing: antialiased;

-moz-osx-font-smoothing: grayscale;

}

code {

font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',

monospace;

}

\* {

box-sizing: border-box;

}

body {

margin: 0;

font-family: Inter, sans-serif;

height: 100vh;

}

nav {

display: flex;

align-items: center;

height: 90px;

padding: 30px 25px;

}

**App.js**

import './App.css';

import ContentTabs from '../Tabs/ContentTabs';

function App() {

return (

<div className="App">

<header className="App-header">

<div><ContentTabs/></div>

</header>

</div>

);

}

export default App;

**App.css**

.App {

text-align: center;

}

.App-logo {

height: 40vmin;

pointer-events: none;

}

@media (prefers-reduced-motion: no-preference) {

.App-logo {

animation: App-logo-spin infinite 20s linear;

}

}

.App-header {

background-color: #111620;

min-height: 100vh;

display: flex;

flex-direction: column;

font-size: calc(10px + 2vmin);

color: white;

}

.App-link {

color: #61dafb;

}

@keyframes App-logo-spin {

from {

transform: rotate(0deg);

}

to {

transform: rotate(360deg);

}

}

**Landing.js**

import './Landing.scss'

import $ from 'jquery';

import logo from "./Logo.png"

function Landing() {

console.log("starting");

console.log($(".e-button").hasClass("open"))

return (

<div className="Landing">

<h2>О проекте</h2>

<img src={logo} />

</div>

)

}

export default Landing

**Landing.css**

.Landing{

margin-top:10%;

}

img{

width: 100%;

height:100%

}

**ContentTabs.js**

import { Tab, Tabs, TabList, TabPanel } from 'react-tabs';

import 'react-tabs/style/react-tabs.css';

import './Tabs.css'

import Reader from '../Reader/Reader';

import Landing from "../Landing/Landing"

import Testing from "../Reader/Testing/Testing"

function ContentTabs(){

return(

<Tabs>

<TabList className="tabs">

<Tab className="tab">О проекте</Tab>

<Tab className="tab">RSVP чтение</Tab>

<Tab className="tab">Тестирование</Tab>

</TabList>

<TabPanel>

<Landing/>

</TabPanel>

<TabPanel>

<Reader/>

</TabPanel>

<TabPanel>

<Testing/>

</TabPanel>

</Tabs>

)

} export default ContentTabs;

**Tabs.css**

li {

list-style-type: none;

}

.tabs{

display: flex;

}

.tab{

margin-right: 20%;

background-color: none;

outline: none;

padding: 0.5rem;

min-width: 100px;

}

h2{

color: rgb(255, 255, 255);

}

**Reader.js**

import ViewArea from "./ViewArea/ViewArea";

import Dropdown from "./Dropdown/Dropdown";

import Settings from "./Settings/Settings";

import './Reader.css'

export default () => (

<div className="Reader">

<div className="Settings">

<div className="Upload">

<Dropdown/>

</div>

</div>

<div className="Read">

<ViewArea/>

</div>

</div>

);

**Reader.css**

body {

}

.Reader {

display: flex;

}

.Read {

width: 100%;

margin: 1%;

}

.Settings {

position: absolute;

top: 10%;

margin: 0.5%;

min-width: 10px;

width: 20%;

}

**ViewArea.js**

import './ViewArea.css'

import React, { useState, useEffect } from 'react';

function ViewArea() {

//var fr = new FileReader();

var text = "другое Слово This is sample text. And i'll try to read it!"

//var words=["word1","word2","word3"];

var words = text.split(" ");

var wordNumber=0;

var word = words[wordNumber];

var textSize =words.length

const[count,setCount] =useState(0);

const incrementCount = () => {

setCount(count+1);

if (count == textSize) setCount(0);

}

/\*useEffect(()=>{

document.title = `count is ${count}`

});\*/

word =words[count];

const start =()=>{

while(count<5){

console.log("next");

incrementCount();

//setTimeout(() => { incrementCount(); }, 1000).then(start());

/\*const interval = setInterval(() => {

}, 1000)

incrementCount();\*/

}

}

return(

<div className="ViewArea">

{word}

<br></br>

<button onClick={start}>продолжить</button>

</div>

)

}

export default ViewArea

function sleep(milliseconds) {

const date = Date.now();

let currentDate = null;

do {

currentDate = Date.now();

} while (currentDate - date < milliseconds);

}

**ViewArea.css**

.ViewArea{

width: 100%;

margin-top: 20%;

font-size: 7rem;

}

button {

margin-top: 10%;

padding: 10px;

text-align: center;

font-size: 5rem;

color: white;

background: rgba(0, 0, 0, 0.4);

width: auto;

height: auto;

border: 2px solid #315cfd;

border-radius: 30px;

transition: all 0.3s;

cursor: pointer;

font-weight: 550;

font-family: 'Montserrat', sans-serif;

}

button:hover {

background: #315cfd;

color: white;

font-size: 6rem;

}

**Dropdown.js**

import './Dropdown.scss'

import $ from 'jquery';

function Dropdown() {

console.log("starting");

console.log($(".e-button").hasClass("open"))

return (

<div className="m-dropdown">

<div onClick={slide} className="e-button open">

<div className="e-burger">

<span></span>

<span></span>

<span></span>

<span></span>

</div>

</div>

<ul className="e-list">

<li><a href="#">Загрузить файл</a></li>

<li><a href="#">Режим</a></li>

<li><a href="#">Скорость</a></li>

<li><a href="#">Масштаб</a></li>

</ul>

</div>

)

}

export default Dropdown

function slide(){

console.log("sliding")

console.log($(".e-button").hasClass("open"))

if ($(".e-button").hasClass("open")) {

$(".e-list").slideUp(function() {

$(".e-button").removeClass("open");

console.log("open removed");

});

} else {

$(".e-button").addClass("open");

console.log("open added");

setTimeout(function() {

$(".e-list").stop().slideDown();

}, 200);

}

}

**Dropdown.scss**

$f-title:'Roboto', sans-serif;

$f-body:'Open Sans', sans-serif;

@mixin centre($axis: "both") {

position: absolute;

@if $axis == "y" {

top: 50%;

-webkit-transform: translateY(-50%);

-moz-transform: translateY(-50%);

-ms-transform: translateY(-50%);

-o-transform: translateY(-50%);

transform: translateY(-50%);

}

@if $axis == "x" {

left: 50%;

-webkit-transform: translateX(-50%);

-moz-transform: translateX(-50%);

-ms-transform: translateX(-50%);

-o-transform: translateX(-50%);

transform: translateX(-50%);

}

@if $axis == "both" {

top: 50%;

left: 50%;

-webkit-transform: translate(-51%, -50%);

-moz-transform: translate(-51%, -50%);

-ms-transform: translate(-51%, -50%);

-o-transform: translate(-51%, -50%);

transform: translate(-51%, -50%);

}

}

@mixin transition($property) {

-webkit-transition: $property;

-moz-transition: $property;

-ms-transition: $property;

-o-transition: $property;

transition: $property;

}

@mixin border-radius($radius) {

border-radius:$radius;

-moz-border-radius:$radius;

-webkit-border-radius:$radius;

-ms-border-radius: $radius;

}

html {

font-size: 62.5%;

}

body{

padding:0;

font-family:$f-body;

background:#ffffff;

}

.m-dropdown{

margin:50px auto;

width:100%;

display:block;

.e-download-button{

@include transition(all .3s ease-in-out);

@include border-radius(20px);

padding: 0.2rem;

font-size: 1rem;

text-align: center;

z-index: 900;

color: #fff;;

font-weight: 100;

position: relative;

border: 2px solid transparent;

margin-top: 2px;

font-family: $f-body;

z-index: 5;

display:block;

width:100%;

margin:0px auto;

background:#1b1b1b;

&:hover{

cursor:pointer;

box-shadow:0 3px 6px rgba(0,0,0,0.2);

}

}

.e-button{

@include transition(all .3s ease-in-out);

@include border-radius(20px);

font-size: 1rem;

padding: 0p8x 40px 8px 15px;

padding: 0.5rem;

z-index: 900;

color: #fff;;

font-weight: 100;

position: relative;

border: 1px solid transparent;

margin-top: 1px;

font-family: $f-body;

z-index: 5;

display:block;

width:10%;

margin:0 auto;

background:#1b1b1b;

&:hover{

cursor:pointer;

box-shadow:0 3px 6px rgba(0,0,0,0.2);

}

&:after,

&:before {

display: block;

content: " ";

position: absolute;

top: 2px;

height: 100%;

width:100%;

z-index: 0;

}

&:after {

left: 100%;

margin-left: 2px;

border-bottom-left-radius: 10px;

}

&:before {

right: 100%;

margin-right: 2px;

}

.e-burger {

@include transition(all .3s ease-in-out);

@include centre(y);

right: 15px;

width: 15px;

height: 14px;

transform-origin: center;

z-index: -1;

span {

@include transition(all .2s ease-in-out);

@include centre(y);

@include border-radius(2px);

width: 7rem;

height: 7px;

margin-bottom: 100px;

background: #f6f6f6;

display: block;

left: 0;

transform-origin: center;

&:first-child {

background: #f10000;

bottom: 100px;

transform: none;

}

&:nth-child(1) {

background: #f5f5f4;

top: -20px;

}

&:last-child {

background: #feffff;

top: auto;

bottom: -120px;

transform: none;

}

}

&:after {

@include transition(height .3s ease-in-out);

content: " ";

display: block;

position: absolute;

top: 12px;

right: -37px;

height: 0;

width:100%;

left: auto;

z-index: -2;

opacity: 0;

}

}

&.open {

border-color: transparent;

border-bottom-left-radius: 0px;

border-bottom-right-radius: 0px;

color:#fdfdfd;

.e-burger {

span {

background:#ffffff;

&:first-child {

opacity: 0;

}

&:nth-child(2) {

transform: translateY(-50%) rotate(-45deg);

}

&:nth-child(3) {

transform: translateY(-50%) rotate(45deg);

}

&:last-child {

opacity: 0;

}

}

&:after {

height: 1rem;

opacity: 1;

}

}

}

}

.e-list{

@include border-radius(3px);

box-shadow:0 3px 6px rgba(0,0,0,0.2);

position:relative;

width:100%;

margin-top: 5rem;

margin-bottom:5rem;

margin-left: 10rem;

padding:0;

list-style:none;

background:#1b1b1b;

float:left;

overflow:hidden;

z-index:999;

align-items: left;

li{

display:block;

width:100%;

float:left;

border-bottom:1px solid rgba(255,255,255,0.01);

box-sizing:border-box;

&:last-child{

border:none;

}

a{

@include transition(all .3s ease-in-out);

font-size: 2.5rem;

text-align: left;

padding:0.3rem;

float:left;

width:100%;

text-decoration:none;

color:#fff;

font-weight:10;

box-sizing:border-box;

&:hover{

background:#fff;

color:#111;

}

}

}

}}