1. Разработка системы совместного редактирования текстов
2. Предметом выпускной квалифакционной работы является программное обеспечение совместной работы для редактирования текста. Целью является реализация системы совместного редактирования текстов в режиме реального времени
3. Задачи:
   1. изучить существующие подходы к организации совместной работы;
   2. спроектировать программное обеспечение;
   3. реализовать программный продукт;
   4. описать его.
4. Требования которые были выдвинуты перед начало разработки:
   1. Реализация в виде расширения редактора Sublime Text (то есть плагина). Выбор обусловлен популярностью редактора и личным предпочтением автора.
   2. Автоматический или полуавтоматический режим синхронизации текста. Другими словами требуется минимизация ручной синхронизации текста между пользователями (например, при конфликтах двойной записи).
   3. Поддержка модульности исходного кода с целью его повторного использования в качестве расширения к другим текстовым редакторам. Планируется расширить количество поддерживаемых редакторов, чтобы разные пользователи могли работать за разными редакторами.
   4. Возможность поиска ожидающего компьютера в локальной сети. Когда два пользователя в локальной сети хотят начать сессию редактирования, один из них становится сервером и становится видимым для остальных желающих компьютеров в локальной сети.
5. [видео]
6. [алгоритм]
7. Архитектура клиент-серверная. В сети есть сервер-координатор и клиенты-писатели. Алгоритм состоит из двух частей. На данном слайде можно увидеть, что писатель обладает двумя версиями одного текста. Тот который он сейчас видит и редактирует, и тот который является одинаковым у данного писателя и координатора.
8. На первом шаге берутся две версии одного документа…
9. … И рассчитывается дельта между ними
10. Дельта это изменения, которые необходимо внести, чтобы один текст стал другим. В дальнейшем будет показано почему здесь фигурируют две дельты.
11. .
12. Дельта отправляется координатору и писатель обновляет свою последнюю синх. версию присваивая ей значение Цэ т
13. .
14. .
15. Координатор берет дельту и свою синх. версию и…
16. Получает новую синх. версию которая содержит модификации писателя
17. [На слайде]
18. Поскольку нет блокировок на редактирование между пользователями, координатор мог изменить свою редактируемую версию. К тому же он должен получить модификации писателя Пэ. Поэтому берется дельта и редактируемый текст.
19. И применяется дельта с учетом ошибок. Данная работа выполняется алгоритмом Fuzzy Patch
20. [на слайде]
21. Таким образом после первой и второй частей алгоритма гарантированно модификации от писателя получит координатор, послед. синхронизированная версия останется равной у обоих, изменения которые конфликтуют не будут потеряны, а будет произведена попытка применить их с учетом нарушения контекста. Чтобы писатель получил изменения координатора, необходимо запустить весь процесс в обратную сторону и тогда общий вид последовательности работы будет такой.
22. С случае когда писателей больше чем один, координатор делит между пользователями лишь редактируемый текст, таким образом изменения от писателя 1 попадают к писателю 2
23. .
24. Тест производительности показал, что текст объемом 44 КБ обрабатывается без проблем даже на нетбуках пятилетней давности. Невысокие показатели могут быть обяснены тем, что не выполнялась никакая оптимизация
25. Результаты работ:
    1. на основании изучения различных существующих подходов разработан собственный алгоритм
    2. написана реализация плагина для редактора Sublime Text
    3. сделано описание программного продукта
26. Дальнейшее развитие проекта: предоставить возможность отправлять непрерывный поток обновлений без ожидания свой очереди, так как при наличии большого числа писателей, каждый обязан будет ждать остальных
27. Еще одно из направлений развития: добавление возможности отслеживать, какой пользователь сделал какие изменения. Это позволит отменять ошибочные модификации какого-то определенного человека.