* Вывод: Современной мультимединое веб-приложение может служить альтернативой настольным графическим приложениям, таким как игры.
* Вывод: Веб-приложение состоит из клиентской и серверной частей между которыми разделена логика приложения. Существует множество технологий для реализации клиентской и серверной части.
* Вывод: Существует достаточно обширный выбор графических технологий для веб-приложений.
* Вывод: Программный интерфейс 2d-context предоставляет минимально необходимые методы для создания двумерного статического изображения. Методов направленных на реализацию анимации интерфейс 2d-context не содержит.
* Можно сделать вывод, что использование 2d-context влечет за собой следующие трудности:
  + Нет возможности работать с частями изображения как с самостоятельными объектами.
  + Для создания анимации требуются перерисовка экрана для изменения изображения.
  + Описание несложных сцен требуют достаточно большое количество кода.
  + Неготовый стандарт. Стандарт в стадии тестирования – W3C объявил о планах, согласно которым окончательная версия стандарта HTML5 будет утверждена лишь к 2014 году[6].
  + Отсутствие визуальных сред, вроде Flash Professional CS6.
  + Слабое развитие специализированных каркасов, вызванные, скорее всего, незавершенностью стандарта.
* Вывод: Современный программный каркас для создания двумерной графики с помощью 2d-context должен реализовывать следующую функциональность:
  + графический цикл;
  + слои;
  + объектное представление примитивов;
  + call-back функции для событий объектов;
  + работа с мышю и клавиатурой;
  + классы-обертки для примитивов;
  + загрузка мультимедиа-зависимостей объектов;
  + приоритеты очереди прорисовки;
  + модульность;
  + вызовы методов каркаса цепочкой.
* Вывод: Принятие стандарта кодирования позволило увеличить читаемость, однородность внешнего вида кода. Продуманная структура файлов проекта упростила сборку каркаса из отдельных компонентов, а также потенциально сделала проще процесс дальнейшего развития каркаса.
* Вывод: Осознанный выбор способа интеграции позволил реализовывать каркас с помощью ряда итерации, в ходе которых постепенно наращивалась функциональность каркаса, что давало возможность проверить удачность реализованной за итерацию части.
* Вывод: Использование утилиты для анализа кода помогло упростить процесс кодирования, а написание своей библиотеки общего назначения позволило повысить количество повторного использования уже реализованных решений и накопить набор потенциально полезных в других проектах, функций.
* Вывод: Использование интегрированной среды разработки дало возможность сосредоточиться на процессе кодирования, что вероятно уменьшило количество возможных ошибок и улучшило качество кода.
* Вывод: выделение трех целевых браузеров позволило оптимизировать каркаса только к наиболее востребованным браузерам.
* Вывод: использование отладчика и профайлера позволило выявлять ошибки и оценивать производительность отдельных участков кода, что позволило повысить производительность каркаса
* Вывод: Использование системы контроля версий уменьшило риск потери исходного кода проекта и упростило процесс кодирования.