Министерство науки и высшего образования Российской Федерации

Новгородский государственный университет имени Ярослава Мудрого

Кафедра ИТИС

План разработки

VSTi Синтезатор

Версия 0.0.1

Выполнили:

ст. гр. 8091

Васильев И.В.

Проверил:

Преподаватель Макаров В. А.

Дата

Великий Новгород

2021

# **Введение**

Техническое задание разработано по личному проекту. Данный документ содержит набор требований, на основе которых будет разработан программный продукт “VSTi Синтезатор”.

# **Описание проекта**

**Назначение и область применения:** Продукт предназначен для генерации звуковых сигналов в области программных средств звукового редактора. Аудитория пользователей состоит из заинтересованных лиц в написание музыкальных партий и исследованию принципа работы синтеза звука.

**Цель разработки:** Получение практических и теоретических навыков в области разработки музыкального программного продукта, результатом которого является встраиваемый модуль для цифровых программных средств обработки звука.

## **Обеспечение проекта**

* + 1. **Команда проекта**
* Васильев Иван Владимирович
  + 1. **Контакты**

Телефон: +789116294490

Электронная почта: [vanyavasil9@gmail.com](mailto:vanyavasil9@gmail.com)

ВК: <https://vk.com/golovonogg>

* + 1. **Форма представления результатов**
* Файл synthesizer.dll содержащий последнюю версию разрабатываемой программы.
* Исходный код программы на языке С++.
* Техническое задание в формате .docx.
* План разработки в формате .docx.
* Функциональная спецификация в формате .docx.
* Руководство пользователя в формате .docx.

Все результаты проекта доступны в открытом доступе для скачивания в хранилище git, указанном в пункте 4.

1. **Организация проекта.**
   1. Модель процесса разработки – **каскадная**.
   2. **Организационная структура**
      1. Роли на проекте

* Васильев Иван Владимирович – ведущий разработчик.

1. **Хранилище материалов**

Все материалы проекта должны храниться в публичном github репозитории по следующему адресу: https://github.com/VasilevIvanVladimirovich/VST-Synthesizer

Материалы включают в себя программный код, документацию и файлы для выпуска программы.

Иерархия папок должна быть реализована в следующем виде:

* **Synthesizer** (Все файлы, связанные с программированием синтезатора)
* **Documentation**(Все файлы, связанные с документацией)
* **Release** (Файлы, необходимые для выпуска программы)

1. **Управление проекта**
   1. Календарный план расположен в приложении 1 в формате PDF
      1. **Вехи проекта:**
         1. Завершение разработки проектной документации, 08.05.2021, на основании которой будет разрабатываться проект.
         2. Получено необходимое понимание предметной области и сформирован набор инструментов, необходимых для разработки программного обеспечения, 22.05.2021.
         3. Разработан прототип генератора звуковых волн, 03.05.2021
         4. Разработан модуль звуковой обработки, 22.05.2021
         5. Разработан модуль эффектов, 2.06.2021
         6. Разработан прототип с графическим интерфейсом, 10.06.2021
         7. Завершён этап тестирования, 16.05.2021
         8. Выпуск программы и подготовка с эксплуатации пользователем, 23.06.2021

1. **Техника проекта**
   1. **Программное обеспечение проекта**
      1. Microsoft Visual Studio c Visual C++ версии 2005 и выше.
      2. Приложение Jucer для редактирования графических интерфейсов.
      3. Библиотека классов Juce.
   2. **Аппаратное обеспечение проекта**
      1. Персональный компьютер

* Видеокарта AMD Radeon R7 Graphics
* 12 Гб ОЗУ DDR4 2400 Mhz
* Процессор AMD A12-9720P RADEON R7
* Материнская плата asus prime A320M-E socket AM4

1. **Процедура принятия кода**

Принятие кода происходит в соответствии с разработанными тест-кейсами. Процедура принятия кода состоит из частей:

1. Ручное тестирование приложения.
2. Опытная эксплуатация.
   1. **Ручное тестирование графического интерфейса.**
      1. Тест кейс №VSTSYN-1М

**Название тест-кейса:** Проверка работоспособности генератора звуковых волн

**Предусловия тест-кейса:** N/A

**Шаги тест-кейса:**

1. Запустить плагин из звукового редактора.
2. Поочерёдно выбирать тип звуковой волны.
3. Проверить звуковую волну на осциллографе.

**Ожидаемый результат:**

Все типы звуковых волн должны визуально совпадать с теоретической осциллограммой.

* + 1. Тест кейс №VSTSYN-2М

**Название тест-кейса:** Проверка огибающей ADSR

**Предусловия тест-кейса:** N/A

**Шаги тест-кейса:**

1. Запустить плагин из звукового редактора.
2. Поочерёдно изменять параметры огибающей.

**Ожидаемый результат:**

Все параметры огибающей ADSR должны изменять звуковую волну в соответствии с теоретическими значениями.

* + 1. Тест кейс №VSTSYN-3М

**Название тест-кейса:** Проверка слияния звукового сигнала из двух генераторов.

**Предусловия тест-кейса:** N/A

**Шаги тест-кейса:**

1. Запустить плагин из звукового редактора.
2. Сгенерировать звуковую волну на первом осцилляторе.
3. Сгенерировать звуковую волну на втором осцилляторе.
4. Использовать функция изменения громкости и слияния двух сигналов.

**Ожидаемый результат:**

Результатом выполнения последнего пункта является синтезируемая звуковая волна с помощью двух осцилляторов.

* + 1. Тест кейс №VSTSYN-4М

**Название тест-кейса:** Проверка частотного фильтра.

**Предусловия тест-кейса:** N/A

**Шаги тест-кейса:**

1. Запустить плагин из звукового редактора.
2. Сгенерировать звуковую волну.
3. Включить модуль частотного фильтра.
4. Изменить частоту на определённый параметр.
5. Проверить изменения АЧХ на графическом анализаторе.

**Ожидаемый результат:**

Частотный фильтр должен изменять АЧХ звуковой волны, в соответствии с введёнными параметрами.

* + 1. Тест кейс №VSTSYN-5М

**Название тест-кейса:** Проверка частотного фильтра.

**Предусловия тест-кейса:** N/A

**Шаги тест-кейса:**

1. Запустить плагин из звукового редактора.
2. Сгенерировать звуковую волну.
3. Включить модуль частотного фильтра.
4. Изменить частоту на определённый параметр.
5. Проверить изменения АЧХ на графическом анализаторе.

**Ожидаемый результат:**

Частотный фильтр должен изменять АЧХ звуковой волны, в соответствии с введёнными параметрами.

* + 1. Тест кейс №VSTSYN-6М

**Название тест-кейса:** Проверка модуля эффекта дисторшн.

**Предусловия тест-кейса:** N/A

**Шаги тест-кейса:**

1. Запустить плагин из звукового редактора.
2. Сгенерировать звуковую волну.
3. Включить модуль эффекта дисторшн.
4. Изменить параметр искажения на определённую величину.
5. Проверить осциллографе звуковую волну.

**Ожидаемый результат:**

Дисторш должен искажать звуковой сигнал, ограничивая его по амплитуде.

* + 1. Тест кейс №VSTSYN-7М

**Название тест-кейса:** Проверка модуля эффекта дилей.

**Предусловия тест-кейса:** N/A

**Шаги тест-кейса:**

1. Запустить плагин из звукового редактора.
2. Сгенерировать звуковую волну.
3. Включить модуль эффекта дилей.
4. Изменить параметр задержки повтором на некоторый максимальный параметр.

**Ожидаемый результат:**

Дилей должен имитировать затухающее повторы(эхо) исходного сигнала.

* 1. **Опытная эксплуатация**

Опытная эксплуатация проводится после ручного тестирования. Данный вид тестирование проходит в условии работы над музыкальным проектом. Тестирование проводят пользователи, занимающие созданием музыкальных партий. Все найденные недостатки записываются в раздел «Отчёт».

Заказчик : Исполнитель:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

м.п. м.п.