

УТВЕРЖДЕНО

RU.96636777.62.01.29-02 100 01-1 ЛУ

**Программный комплекс «АльтерОфис Специальный»/«AlterOffice»
Средства применяемых для разработки
RU.96636777.62.01.29-02 100 01-1**

Листов 8

СОДЕРЖАНИЕ

Введение	3
1 Описание средств разработки, применяемых для создания средства	4
2 Описание процесса сборки ПК AlterOffice.....	8

Введение

Настоящий документ распространяется на Программный комплекс «AlterOffice» (далее — ПК AlterOffice, Система, Изделие, ОО). ПК AlterOffice представляет собой программное средство, реализующее функции защиты от несанкционированного доступа к информации, обрабатываемой на средствах вычислительной техники, находящихся под управлением данного программного средства.

Разработчик: ООО «Алми Партнер» (юридический адрес: 117186, г.Москва, вн.тер.г. Муниципальный округ Котловка, ул. Нагорная, д. 24 к. 9, помещ. 1/1, лицензия ФСТЭК России на деятельность по разработке и производству средств защиты конфиденциальной информации № Л050-00107-00/00584159 от 08 августа 2019 г.).

Производитель: ООО «Алми Партнер» (юридический адрес: 117186, г.Москва, вн.тер.г. Муниципальный округ Котловка, ул. Нагорная, д. 24 к. 9, помещ. 1/1).

Настоящий документ содержит описание средств, применяемых для разработки средства и описание использованных опций средств, применяемых для разработки средства.

1 Описание средств разработки, применяемых для создания средства

1.1 Инструменты, используемые для разработки ПК AlterOffice

1.1.1 Visual Studio Code

В качестве среды разработки для ПК AlterOffice используется Visual Studio Code.

Visual Studio Code поддерживает большое количество языков программирования, включая популярные, такие как JavaScript, Python, Perl, C++, а также менее распространенные, например, Rust, Go, Swift. Кроме того, с помощью плагинов можно добавить поддержку для других языков.

Описание данной среды разработки доступно по ссылке: <https://code.visualstudio.com/>.

1.1.2 JetBrains PyCharm

В качестве среды разработки для ПК AlterOffice используется JetBrains PyCharm.

JetBrains PyCharm – это кроссплатформенная интегрированная среда разработки для языка программирования Python.

Описание данной среды разработки доступно по ссылке: <https://www.jetbrains.com/pycharm/>.

1.1.3 Qt Creator

В качестве среды разработки для ПК AlterOffice используется Qt Creator.

Qt Creator – это кроссплатформенная интегрированная среда разработки для языков C/C++.

Описание данной среды разработки доступно по ссылке: <https://www.qt.io/product/development-tools>

1.2 Компиляторы и инструменты сборки

1.2.1 GCC

В качестве компиляторов для ПК AlterOffice используются gcc и g++, входящие в состав GCC.

GCC (the GNU Compiler Collection) – набор компиляторов для различных языков программирования, разработанный в рамках проекта GNU.

Описание данного компилятора доступно по ссылке: <https://gcc.gnu.org/>.

GCC включает утилиту для исследования покрытия кода gcov, позволяющую генерировать точное количество исполнений для каждого оператора в программе и позволяет добавлять аннотации к исходному коду.

Описание используемых опций компилятора GCC указывается в make-файлах. Во время сборки вызывается утилита automake, которая запускает Makefile, который в свою очередь инициирует запуск компилятора GCC с указанными опциями.

Используемые версии в средах сборки:

- CentOS 7: GCC 10.2.1
- Astra Linux 1.7: GCC 8.3.0

1.2.2 Python

Python используется в ПК AlterOffice для выполнения скриптов автоматизации и работы некоторых компонентов системы.

Используемые версии в средах сборки:

- CentOS 7: Python 2.7.5
- Astra Linux 1.7: Python 2.7.16

1.2.3 GNU Make

Утилита для автоматизации процесса сборки программного обеспечения.

Используемые версии в средах сборки:

- CentOS 7: GNU Make 4.2.1
- Astra Linux 1.7: GNU Make 4.2.1

1.2.4 Perl

Perl применяется в ПК AlterOffice для обработки текстовых данных и выполнения скриптов автоматизации.

Используемые версии в средах сборки:

- CentOS 7: v5.16.3
- Astra Linux 1.7: v5.28.1

1.2.5 pkg-config

Программа для получения информации о установленных библиотеках при сборке ПО.

Используемые версии в средах сборки:

- CentOS 7: 0.27.1
- Astra Linux 1.7: 0.29

1.2.6 Autoconf

Инструмент для создания автоматизированных скриптов конфигурации сборки.

Используемые версии в средах сборки:

- CentOS 7: GNU Autoconf 2.69
- Astra Linux 1.7: GNU Autoconf 2.69

1.2.7 Фреймворк Qt5

Кроссплатформенный фреймворк для разработки графических интерфейсов.

Используемые версии в средах сборки:

- CentOS 7: Qt 5.15.2
- Astra Linux 1.7: Qt 5.15.2

1.3 Языки программирования

1.3.1 C

При разработке ПО ПК AlterOffice используется компилируемый язык программирования C, описание которого доступно по ссылке: https://www.iso-9899.info/wiki/The_Standard.

1.3.2 C++

При разработке ПО ПК AlterOffice используется компилируемый язык программирования C++, описание которого доступно по ссылке: <https://isocpp.org/std/the-standard>.

1.3.3 Perl

При разработке ПО ПК AlterOffice используется интерпретируемый язык программирования Perl, описание которого доступно по ссылке <https://www.perl.org/>.

1.3.4 Python

При разработке ПО ПК AlterOffice используется интерпретируемый язык программирования Python, описание которого доступно по ссылке: <https://www.python.org/about/>

1.4 Инструменты, используемые для управления исходным кодом

1.4.1 Git

Для хранения истории изменений исходного кода ПО ПК AlterOffice используется распределенная система управления версиями Git, описание которой доступно по ссылке: <https://git-scm.com/doc>.

1.4.2 Gitlab

Для интеграции с Git-репозиторием в целях организации хранения и версионирования исходного кода ПО ПК AlterOffice используется веб-сервис Gitlab, описание которого доступно по ссылке: https://docs.gitlab.com/ee/tutorials/gitlab_navigation.html.

1.5 Инструменты, используемые для организации разработки ПО ПК AlterOffice

1.5.1 Gitlab CI/CD

В качестве системы автоматизации сборки ПО ПК AlterOffice и для обеспечения процесса непрерывной интеграции используется программная система Gitlab CI/CD, описание которой доступно по ссылке: <https://docs.gitlab.com/ee/ci/>.

1.5.2 Gitlab Wiki

Хранение документации на ПК AlterOffice выполняется в системе Gitlab Wiki. Gitlab Wiki – это вики-система для внутреннего использования организациями с целью создания единой базы знаний. Описание данного средства доступно по ссылке: <https://docs.gitlab.com/ee/user/project/wiki/>.

1.5.3 Jira

В качестве платформы управления проектами используется Atlassian Jira. Jira – это система отслеживания ошибок, предназначенная для организации взаимодействия разработчиков, и для управления проектами. Описание данного средства доступно по ссылке: <https://confluence.atlassian.com/alldoc/atlassian-documentation-32243719.html>.

1.5.4 Nginx

Для доступа к хранящимся дистрибутивам ПК AlterOffice используется программное средство Nginx, описание которого доступно по ссылке: <https://nginx.org/ru/>.

1.5.5 Help&Manual

Формирование инструкции по установке ПК AlterOffice осуществляется с использованием HelpAndManual на основании комментариев, которые указаны в скриптах сборки. Описание данного средства доступно по ссылке: https://www.helpandmanual.com/help/hm_intro_about.html.

2 Описание процесса сборки ПК AlterOffice

Генерация ПК AlterOffice и дистрибутива из программного кода осуществляется автоматизированными средствами. Сборка ПК AlterOffice осуществляется на серверах с сертифицированными операционными система «Astra Linux Special Edition» и «AlterOS», для сборки deb или rpm соответственно. Система автоматизации сборки не осуществляет преобразования (инструментирование, модификации произвольного вида) байт-кода в процессе сборки ПК AlterOffice.

Шаги процесса сборки:

1. Команда продукта публикует исходный код проекта в Gitlab
2. Команда продукта запускает производственные задания для всех типов пакетов (deb, rpm)
3. Gitlab CI/CD загружает исходный код для сборки пакетов на соответствующую заданию сборочную машину
4. В зависимости от задания (сборка deb или rpm), Gitlab CI/CD проводит предсборочную конфигурацию исходного кода (вызов automake) на соответствующей заданию сборочной машине
5. Gitlab CI/CD, на соответствующей заданию сборочной машине, запускает сборку всего проекта, результатом которого будут все необходимые для работы пакеты (deb или rpm)
7. Команда продукта выгружает все собранные пакеты во внутреннее хранилище для последующего тестирования
8. При успешном прохождении тестирования, собранные ранее пакеты публикуются в основном (публичном) репозитории