

Общество с ограниченной ответственностью «Си Икс Лаб»

**РЕШЕНИЕ ПО СОЗДАНИЮ АВТОМАТИЗИРОВАННЫХ РАБОЧИХ МЕСТ**

**СХВОХ V6.0**

**РЕГЛАМЕНТ ЭКСПЛУАТАЦИИ**

Листов 6

Москва

2024

В настоящем документе приведен регламент эксплуатации научно-технического продукта – программы для ЭВМ «Решение по созданию автоматизированных рабочих мест (СХВОХ) v 6.0» (далее – Система).

## СОДЕРЖАНИЕ

|  |   |
|--|---|
| 1. Общие сведения .....  | 4 |
| 1.1. Наименование и условное обозначение системы .....         | 4 |
| 1.2. Пользователи, на которых распространяется регламент ..... | 4 |
| 1.3. Общее описание системы .....                              | 4 |
| 1.4. Исходные коды системы .....                               | 5 |
| 2. Участие в развитии .....                                    | 6 |
| 2.1. Создание вопросов (issues) .....                          | 6 |
| 2.2. Пул-реквесты.....   | 6 |
| 2.3. Участие в инспекциях.....                                 | 6 |
| 2.4. Организация веток .....                                   | 6 |

## **1. Общие сведения**

### **1.1. Наименование и условное обозначение системы**

Полное наименование: программа для ЭВМ «Решение по созданию автоматизированных рабочих мест (СХВОХ) v 6.0».

Краткое наименование: СХВОХ v6.0.

### **1.2. Пользователи, на которых распространяется регламент**

Настоящий регламент распространяется на участников сообщества развития решения СХВОХ v6.0.

### **1.3. Общее описание системы**

Решение СХВОХ v6.0 представляет из себя систему для декларативного создания интерфейсов, автоматизации рабочих мест и интеграции в ландшафт заказчика с использованием готовых функциональных блоков. Под “декларативным” созданием интерфейсов понимается подход, при котором элементы интерфейса автоматизированного рабочего места описываются в соответствии с предусмотренным системой форматом, а система отвечает за интерпретацию полученного описания, отрисовку интерфейса автоматизированного рабочего места и реализацию логики работы элементов интерфейса.

Основные возможности решения:

- обеспечение существенного ускорения автоматизации рабочих мест предприятий за счет предоставления готового набора интерфейсных элементов и возможность конструирования из них интерфейсов любой сложности;
- использование простого декларативного языка описания интерфейса и отсутствие необходимости описывать бизнес-логику стандартных элементов интерфейсов;
- возможность интеграции Системы в оперативное окружение систем клиентов;
- системные дополнения, реализующие служебные, технические и интеграционные решения: «S3 File Storage», «Интеграционный модуль», «Базовая авторизация», «Аутентификация», «Поддержка различных СУБД», «Поддержка микросервисов»;
- дополнения разработчика, интегрируемые со средой разработки IntelliJ IDEA и предназначенные для ускорения и упрощения процесса разработки: «Шаблонизатор кода», «Дополнение быстрой навигации», «Автодополнение кода», «Поиск и исправление ошибок», «Инструменты мониторинга», «Интерактивная документация», «Навигация к документации»;

- бизнес-дополнения, интегрируемые с ядром бэкенда такие как: «Интеграция с email», «Push-уведомления», «Предварительный просмотр документов»;
- административное дополнение, реализующие задачи администратора приложения: «Планировщик заданий».

Алгоритм функционирования системы приведен в документе «Решение по созданию автоматизированных рабочих мест СХВОХ v6.0. Описание функциональных характеристик».

## **1.4. Исходные коды системы**

Для хранения исходных кодов используется распределенная система управления версиями Git. Она используется для отслеживания изменений в исходном коде программного обеспечения и управления разработкой проектов.

Git предоставляет разработчикам возможность сохранять историю изменений, создавать ветки для параллельной разработки, сливать изменения из разных веток и откатывать изменения к предыдущим версиям. Это позволяет упростить совместную работу в команде и управление разработкой проектов.

Одной из главных особенностей Git является его распределенная природа. Каждый разработчик имеет полную копию репозитория Git на своем локальном компьютере, что позволяет им работать независимо от подключения к сети или удаленному серверу. Когда разработчик выполняет коммит изменений, они сохраняются локально, а затем могут быть отправлены на удаленный Git-сервер (Remote Git) для совместного доступа и резервного копирования.

Remote Git - это термин, используемый для обозначения удаленного репозитория Git. Remote Git-сервер обычно предоставляет доступ к репозиторию через сетевое соединение, такое как HTTP, SSH или другие протоколы. Публикация исходного кода на удаленном Git-сервере позволяет другим разработчикам получить доступ к проекту, вносить изменения и синхронизироваться с общей историей изменений.

Исходные коды для загрузки доступны со следующих Git-remote серверов:

1. Внутренние Remote репозитории компании.
2. <https://gitflic.ru/project/cxbox/cxbox-demo> (хранение кода, процесс разработки см.п.1 и п.3)
3. <https://github.com/CX-Box/cxbox-demo> (выполняется ниже описанный регламент работы с системой)

## **2. Участие в развитии**

Решение и организация его инфраструктуры допускают возможность создания сообщества разработчиков, обогащающих систему новыми дополнениями, которые становятся доступными для других участников сообщества. Команда разработки системы приветствует участие сторонних разработчиков в развитии.

### **2.1. Создание вопросов (issues)**

При создании запроса, во избежание дублирования, следует убедиться, что похожих запросов еще не создавалось. Также рекомендуется предварительно ознакомиться с документацией.

### **2.2. Пул-реквесты**

Существенные улучшения, которые могут изменять концепцию решения, необходимо предварительно обсуждать с командой разработки решения, для того чтобы согласовать общий взгляд на проблему и ее решение.

### **2.3. Участие в инспекциях**

При инспекции пул-реквеста разработчику следует воздерживаться от подтверждения или отклонения, если он не является членом команды разработки системы.

### **2.4. Организация веток**

Мажорные версии выпускаются с мастер-веток, отмеченных соответствующим тегом.

Минорные версии сначала вливаются в ветку «develop».

Улучшения и исправления ошибок должны именоваться в формате «улучшение/номер вопроса».