МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет физико-технический**

**Кафедра теоретической физики и компьютерных технологий**

**КУРСОВОЙ ПРОЕКТ**

**Оформление текста модулей на 1С:EDT**

Работу выполнил \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Андреев Иван Денисович

Курс 2

Направление подготовки 09.03.02 Информационные системы и технологии

Научный руководитель

канд. физ.-мат. наук, доцент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д. С. Шевалдин

Нормоконтролёр, преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ .Н. А. Мицук

Краснодар

2023

Реферат

Курсовой проект 19 с., 7 рис., 7 ист., 2 прил.

1С:EDT, СТРУКТУРА, разработка модулей, пользовательский интерфейс, Плагин, Конфигурация, верстка, форматирование

Объектом разработки данного курсового проекта является модуль, способный предупреждать пользователя о нарушении конструкции объекта в среде разработки 1C:EDT.

Целью работы является разработка плагина, способного отображать пользователю ошибку при назначении свойств объекту в 1C:EDT, а также пояснений и инструкций к программе.

В результате работы над курсовым проектом разработан плагин, подключаемый к любой конфигурации с возможностью просмотра конструкции и исправления ошибок, влияющих на читаемость кода и количество ошибок в коде. Также были разработаны инструкции по эксплуатации данного модуля, благодаря которым любой пользователь сможет разобраться в плагине и модернизировать его под свои нужды.

**Содержание**

Перечень обозначений и сокращений 4

Введение 5

1 Обзор существующих подходов к оформлению документов для 1С:EDT 7

1.1 Обзор платформ 7

1.2 Методология плагинов 8

1.3 Анализ существующего оформления документов в 1С:EDT 9

2 Проектирование текстовых модулей под конкретные задачи с использованием Eclipse 12

2.1 Разработка текста модулей на 1С:EDT 12

2.2 Описание программного средства 13

2.3 Рекомендации и инструкции к программе 15

Заключение 17

Список использованных источников 19

Приложение А Код демонстративного программного средства на языке программирования Java 20

Приложение В Код готового программного средства на языке программирования Java 21

**Обозначения и сокращения**

|  |  |
| --- | --- |
| *JDT* | Набор плагинов для Java |
| *IDE* | Интегрированная среда разработки |
| *PDE* | Инструмент для разработки, отладки и установки плагинов |
| *SDK* | Инструменты для разработки программного обеспечения |
| *Ключ* | Значение, хранящиеся в системе |
| *Конструктор* | Метод, отвечающий за создание нового объекта |

Введение

Программисты могут столкнуться с различными проблемами при написании кода, такими как повторяющиеся задачи кодирования и отсутствие знаний или опыта в определённых областях. Также сильно загруженный код мешают нормальному пониманию текста и его редактированию. Для решения этих проблем создаются специальные шаблоны, именуемые текстовыми модулями.

Подобные конструкции представляют собой небольшую часть программы, которая добавляет новые функции к существующему приложению. Также они могут быть легко вставлены в работу программиста, что экономит время и уменьшает количество ошибок. Кроме того, текстовые модули могут помочь восполнить пробелы в знаниях, предоставляя примеры кода.

Целью данной работы является разработка такого модуля, написанного в Eclipse для 1С:EDT, со способностью интегрироваться на многие площадки, а также создание инструкция для пояснения работы написанного кода.

Для достижения этой цели необходимо решить следующие задачи:

− найти неоптимизированные участки кода;

− выбрать необходимые конструкции к плагину;

− создать на требованиях плагин;

− проверить работоспособность кода;

− создать инструкции по эксплуатации программы.

Плагин может быть использован в любой программе, что даёт возможность экономии времени при написании кода. На данный момент плагины позволяют разработчикам расширять функционал программы и создавать собственные решения под нужды своих клиентов. В условиях постоянных изменений требований и технологических достижений возможность создавать и модифицировать плагины имеет решающее значение для разработчиков программного обеспечения, чтобы не отставать от меняющихся требований своих клиентов. Из-за этого растет число неопытных сотрудников и, соответственно, число логических и синтаксических ошибок, которые необходимо быстро устранять.

Кроме того, наличие плагинов от сторонних разработчиков позволяет использовать еще более широкий спектр решений и вариантов настройки различных приложений. Таким образом, модули остаются жизненно важным аспектом в разных средах разработки, и они до сих пор сохраняют свою актуальность в индустрии разработки программного обеспечения.

**1 Обзор существующих подходов к оформлению документов в 1С:EDT**

**1.1 Обзор платформ**

1С:EDT − это IDE для разработки бизнес-приложений на платформе «1С:Предприятие». Платформа широко используется в различных сферах бизнеса. Но сама EDT – это target – платформа, то есть конфигурация аппаратного и программного обеспечения, для которой оптимизирована программа (c.541 из [1]).

При выборе платформы для создания плагинов нужно учитывать многие факторы. Самое главное – это совместимость с 1C, чтобы быстро и эффективно разработать плагин, подходящий под основные требования. Также выбранная платформа должна предлагать нам необходимый функционал для создания плагинов.

Поэтому выбор пал на среду разработки Eclipse, так как она создана как система на основе плагинов, предоставляя богатую платформу разработки плагинов, которая упрощает создание, тестирование и развёртывание плагинов.

Eclipse − это IDE, используемая разработчиками программного обеспечения для создания своих программных проектов и управления ими. Его преимущества заключаются в отрытом исходном коде, широком спектре доступных плагинов и расширений, а также поддержке нескольких языков программирования (c. 44 из [2]).

Можно сказать, Eclipse разработан для того, чтобы помочь разработчикам более эффективно писать, тестировать и выполнять отладку кода. Он предоставляет ряд инструментов и функций, которые помогают разработчикам оптимизировать свой рабочий процесс и повысить производительность.

Теперь, когда мы определились с выбором платформы, мы можем рассмотреть принципы написания плагинов в Eclipse.

**1.2 Методология плагинов**

Для создания плагинов в Eclipse необходимо добавить target – платформу, содержащую в себе множество библиотек и компонент. Найти нужную платформу можно в профиле компании «1С» на GitHub, где есть репозиторий «dt-example-plugins» с примером плагина для EDT (см. информацию из [6]**)**.

Нужные нам объекты представляют собой ссылку, которая будет рассматриваться в примерах модулей и использоваться в создании. Она находится в свойстве «repository» файла «default.target». Нужная ссылка на репозиторий текущей target – платформы изображена на Рисунке 1.

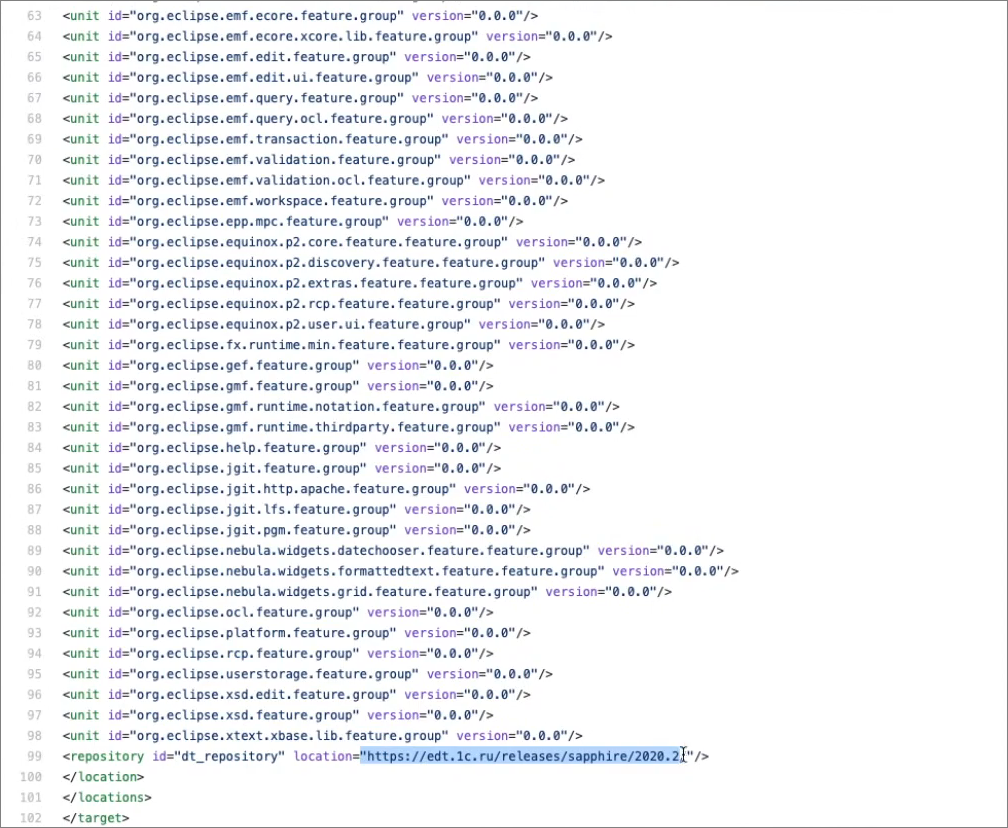


Рисунок 1 – местоположение target – платформы для работы в Eclipse

Для создания целевой платформы необходимо выбрать группу «Plug-in Development», а далее «Target Definition». В запустившемся редакторе указать, что target – платформу нужно взять с определённого сайта и указать ссылку (c. 163 из [3]).

Созданную платформу можно использовать как в качестве основы при демонстрации создания плагинов, так и при их полной разработке.

**1.3 Анализ существующего оформления документов в 1С:EDT**

Основу модулей закладывают конструкторы, которые позволят создавать самые различные плагины, начиная с самых обычных и заканчивая разработкой мобильных приложений (c. 2174 из [4]).

При создании проекта плагина есть возможность использовать стандартные шаблоны Eclipse. Для примера было использовано стандартное создание плагина.

В новых проектах можно выбрать шаблоны готовых плагинов, чтобы продемонстрировать их возможности или взять их в качестве основы. Для демонстрации был взят пример «Hello, World Command». Кроме того, на Рисунке 2 показано описание того, что выбранный шаблон делает и как это реализовано.

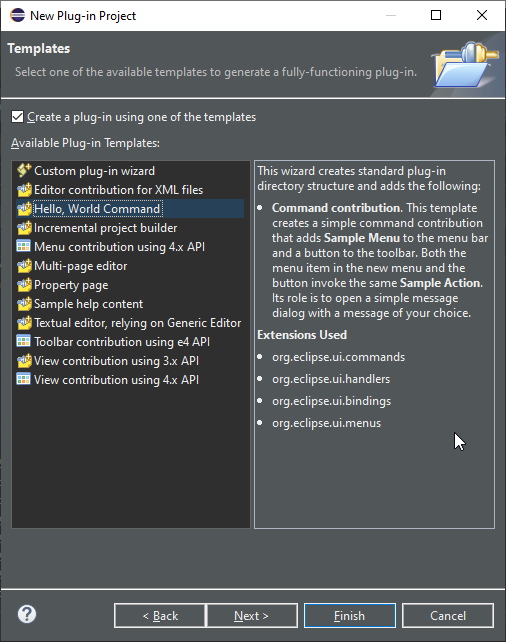


Рисунок 2 – Выбор шаблонов модулей и описания плагина «Hello, World Command»

После выбора шаблона автоматически появляется плагин. При запуске которого открывается 1C:EDT, демонстрирующее пример работы выбранного кода. Уже в нём выбирается пункт меню «Sample Menu» и открывается команда «Sample Command», где получается диалоговое окно с надписью «Hello, Eclipse world», представленное на Рисунке 3.

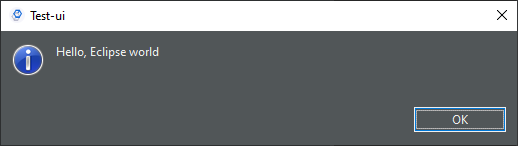


Рисунок 3 – результат работы программы

После выбора модуля появился файл SampleHandler с кодом выполнения внутри (рассмотреть его можно в Приложении А).

Для примера использовался простой метод вывода символов на диалоговое окно. В обработчике команды вызывается метод «MessageDialog.openInformation()» – это диалог, предназначенный для показа сообщений пользователю. В нём создается определённый класс UI-диалога, а также выводятся заголовок окна и само сообщение «Hello, Eclipse World».

**2 Проектирование текстовых модулей под конкретные задачи с использованием Eclipse**

**2.1 Разработка текста модулей для 1С:EDT**

В качестве основной задачи плагина был взят стандарт, который гласит, что в конструкторе структуры неправильно использовать больше 3-х ключей, потому что это сильно усложняет понимание кода (см. информацию из [7]).

Часто бывает так, что разработчик сначала написал в конструкторе структуры три ключа, потом кто-то добавил в него еще один ключ и т.д.

На Рисунке 4 видно, что в конструкторе находиться пять свойств. Если продолжать расширять и добавлять значения и переменные дальше, то сильно пострадает его понятность и велика вероятность ошибки при его использовании.

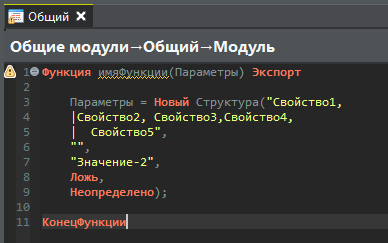


Рисунок 4 – пример нарушения присваивания большого количества свойств в конструкторе

Соответственно, все такие случаи хотелось бы находить, и для решения этой задачи нужно построить плагин.

**2.2** **Описание программного средства**

В начале создаётся пустой проект, в котором будет располагаться модуль. Создание происходит точно так же, как и при рассмотрении шаблонов плагинов. Сначала задаётся имя проекта, затем номер версии и указание версии платформы средs разработки.

Далее нужно добавить точку расширения, позволяющую работать со встроенным языком (c. 174 из [5]), расположенную во вкладке «Extensions». После этого идёт создание класса с указанием его имени, типа и настроек конструктора.

Созданный файл с именем класса автоматически отобразится в папке «src», что и изображено на Рисунке 5.

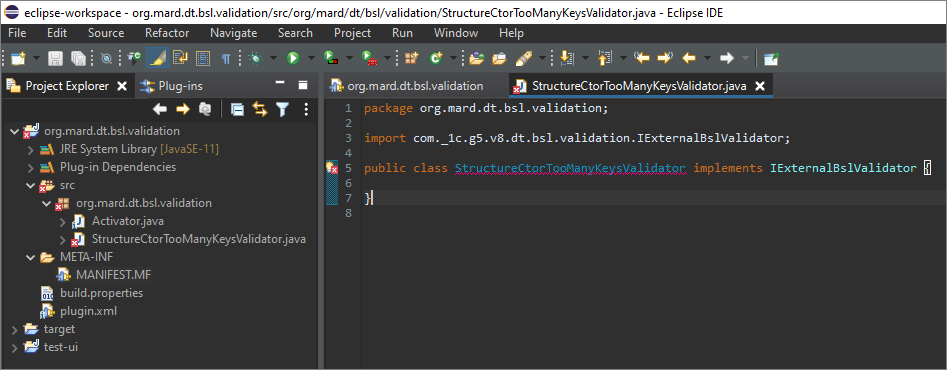


Рисунок 5 – файл класса, где будет происходить вся работа

По ходу написания программы добавляются методы, которые реализуют определённый интерфейс. В нём записываются переменные, условия и команды для работы плагина.

Интерфейс Eclipse позволяет быстро импортировать библиотеки, определять интерфейсы и создавать новые классы и подключать их к основному файлу.

Для примера можно рассмотреть класс «IEObjectTypeNames», содержащий в себе огромное количество типов. Он находится в одной из библиотек, скачанной ранее, как видно на Рисунке 6.

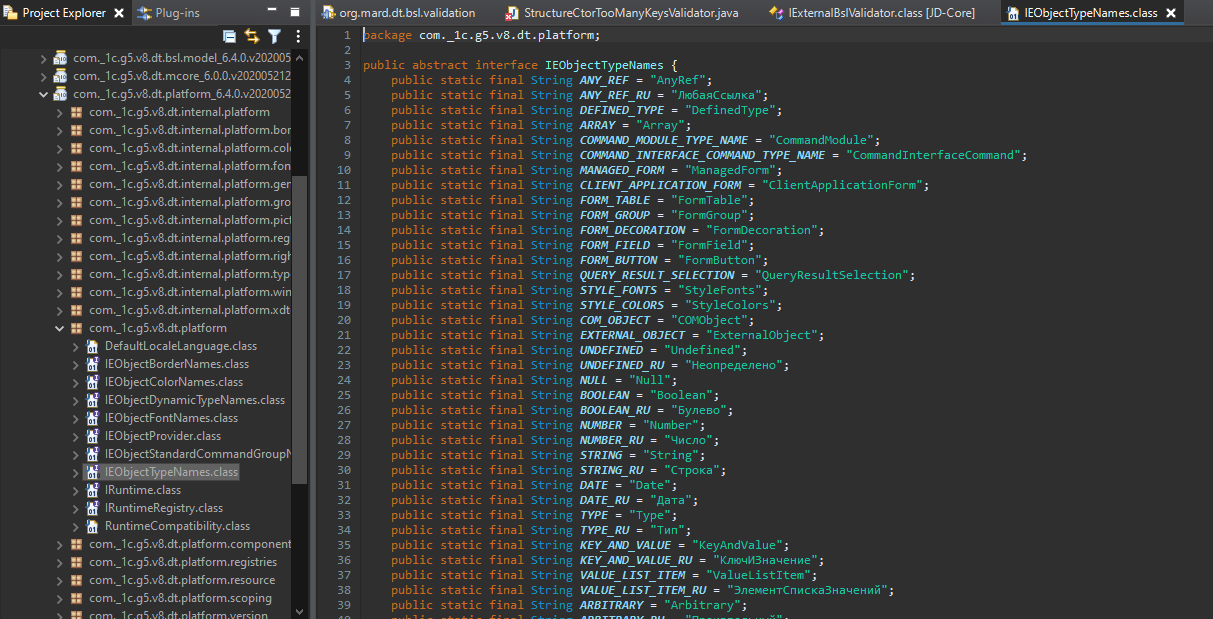


Рисунок 6 – изображение класса «IEObjectTypeNames»

Тип является двуязычным, поэтому текст может быть написан на каком-то из вариантов скрипта или уже с объектом, который имеет сразу встроенную двуязычность.

Из класса «IEObjectTypeNames» нам нужно значение константы «STRUCTURE», которое соответствует представлению типа структуры на английском. Это значение сравнивается с результатом метода «getName()», который возвращает имя полученного из параметра типа на английском (для русских имен есть метод «getNameRu()»).

Теперь необходимо определять по литералам количество свойств в структуре и задать константу «MAX\_STRUCTURE\_KEYS», которая будет максимально допустимым значением количества ключей.

В итоге был получен код, который делает проверку кода на наличие нарушений синтаксиса и, если таковы были найдены, показывает пользователю предупреждение, что структура конструктора превышает допустимое значение (всё это вместе с комментариями можно рассмотреть в Приложении Б).

**2.3 Рекомендации и инструкции к программе**

Программный код сопровождён комментариями, а во избежание проблем с пользовательским интерфейсом программного средства была создана инструкция-иллюстрация, размещённая на Рисунке 7.

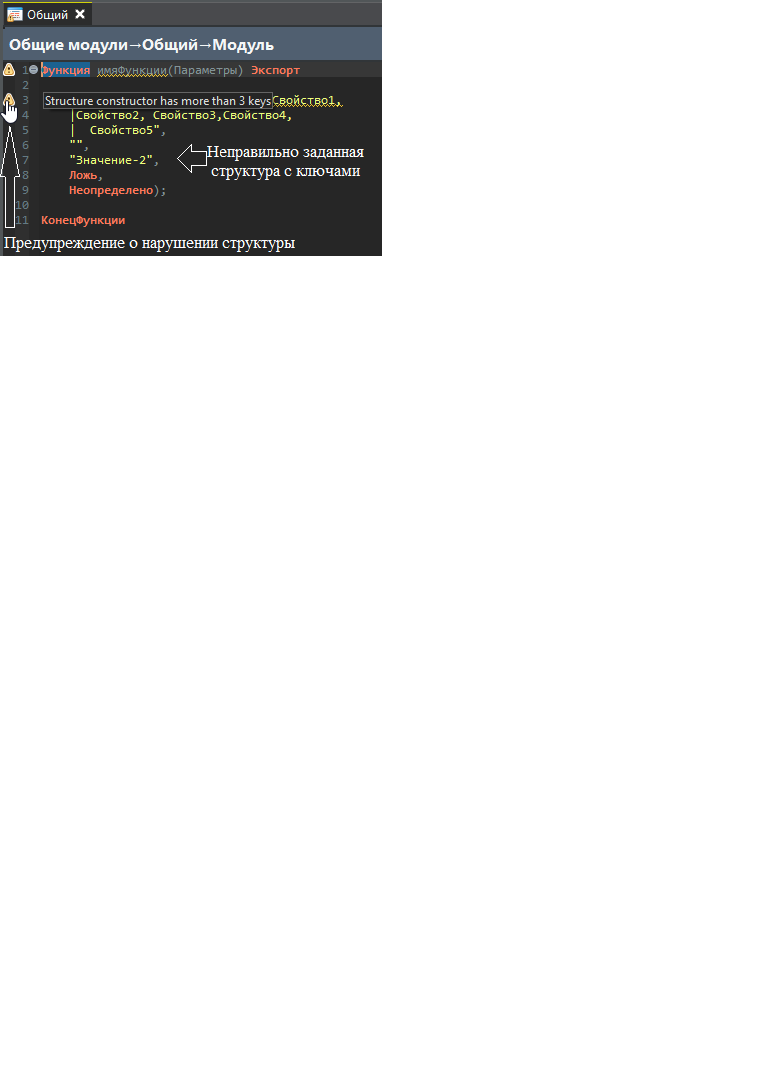


Рисунок 7 – инструкция-иллюстрация для работы с пользовательской частью программы

Для улучшения дизайна текстовых модулей для 1С:EDT можно предложить следующие рекомендации:

– Использовать одинаковый размер и стиль шрифта во всех текстовых модулях, чтобы сделать его легко читаемым.

– Расставлять интервалы между строками, абзацами и разделами текста, что облегчит пользователям просмотр содержимого и поиск необходимой информации.

– Добавлять маркеры или специальные знаки, чтобы выделить ключевые моменты и представить информацию в ясной и краткой форме.

– Предоставить пользователям отдельное окно с интуитивно понятным интерфейсом для упрощения работы с программой.

– Включить в проверку больше стандартов для улучшения возможностей плагина.

Следуя этим рекомендациям, можно создать такой модуль, который будут визуально привлекательным, легко читаемыми и эффективным для работы пользователям.

**ЗАКЛЮЧЕНИЕ**

Основные результаты курсового проекта состоят в следующем:

– Проведён анализ на неоптимизированные участки кода. Подобраны варианты оптимизации и улучшения различных частей кода. Разобраны способы изменения и редактирования кода в реальном времени..

– Выбраны необходимые конструкции к плагину. Изучены проблемы при написании кода.

– Создано программное средство, выполняющее проверку конструкции на наличие ошибок и выдающее предупреждение пользователю (в том числе благодаря нескольким более мелким элементам системы).

– Внутри программного средства создан режим стресс-теста, при включении которого происходит симуляция запуска программы, позволяющее оценить эффективность кода в разных условиях.

– Созданы сопроводительные инструкции к программному средству в виде подробных комментариев в коде и инструкции-иллюстрации, поясняющей все нюансы работы с пользовательской частью программы.

Таким образом была создана абсолютно новое программное средство, проводящее проверку синтаксиса и предупреждающее пользователя об ошибках в построении конструкции, которое можно использовать в различных платформах. В дальнейшем планируется развивать программу, специализируя её под определённые условия. Это сделает получившийся продукт намного более полезным и позволит расширять спектр возможностей.

В ходе работы были выработаны следующие компетенции:

ПК-7. Способность создания технической документации информационно-методического и маркетингового назначения в сфере информационных технологий и систем

ПК-10. Способность выполнения менеджмента проектов в области информационных технологий (планирование, организация исполнения, контроль и анализ отклонений) для эффективного достижения целей проекта.

ПК-11. Способность выполнять концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности.

ПК-12. Способность выполнять проектирование и графический дизайн интерактивных пользовательских интерфейсов.

**Список использованных источников**

*Издание с указанием редактора, составителя*

1. 1C:Предприятие 8.3. Практическое пособие разработчика. Примеры и типовые приемы. Издание 2, стереотипное. / под общей редакцией М. Г. Радченко – Москва, Издательство: ООО «1C-Паблишинг» 2022. – 965 с.

2. Eclipse: разработка RCP-, Web-, Ajax- и Android-приложений на Java / под общей редакцией Т. С. Машнина – БХВ-Петербург, 2013 год, 384 стр.

3. Настольная книга 1C: по технологическим вопросам. Издание 2 / под общей редакцией Е. В. Филиппова. – Москва, Издательство: ООО «1C-Паблишинг» 2015. – 247 с.

4. Профессиональная разработка в системе «1C:Предприятие 8». Издание 2 (+диск): в 2-х т. / под общей редакцией Ажеронока В. А.– Москва, Издательство: ООО «1C-Паблишинг» 2012. – 2800 с.

5. 1C:Enterprise Development Tools Руководство разработчика / под общей редакцией Радченко М. Г. – Москва, Издательство: ООО «1С-Софт», 2022. – 270 с.

*Электронные ресурсы (статья, журнал, электронные издания,*

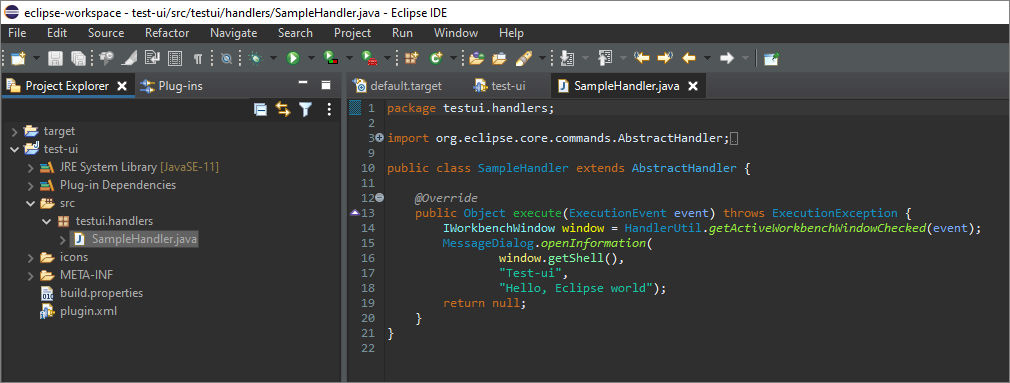
*данные с сайтов в сети Интернет)*

6. Target – платформа с необходимыми библиотеками и объектами − 2023. − URL: https://docs.python.org/3/library/tkinter.html (дата обращения: 05.02.2023).

7. Стандарт об использовании объектов типа Структура в 1С − 2023. − URL: https://its.1c.ru/DB/V8STD#CONTENT:693:HDOC (дата обращения: 19.03.2023).

**Приложение А**

Код демонстративного программного средства  
на языке программирования Java



**Приложение Б**

Код готового программного средства на языке программирования Java

