|  |  |
| --- | --- |
| **Изображение выглядит как ночное небо  Автоматически созданное описание** | **автономная некоммерческая образовательная организация**  **высшего образования Центросоюза Российской Федерации**  **«Сибирский университет потребительской кооперации»** |

Кафедра информатики

**ОТЧЕТ ОБ ОЗНАКОМИТЕЛЬНОЙ** **ПРАКТИКЕ**

Место прохождения практики

АНОО ВО Центросоюза РФ СибУПК

Обучающегося(ейся) 2 курса

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

*(Фамилия И.О.)*

группы ПИБ-01

Руководитель практики

доцент кафедры информатики,

канд. пед. наук

***Колдунова Ирина Дмитриевна***

Оценка после защиты \_\_\_\_\_\_\_\_\_\_\_

Дата защиты 30.12.2021

Новосибирск

2021

|  |  |
| --- | --- |
| **Изображение выглядит как ночное небо  Автоматически созданное описание** | **автономная некоммерческая образовательная организация**  **высшего образования Центросоюза Российской Федерации**  **«Сибирский университет потребительской кооперации»** |

Кафедра информатики

ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

Обучающемуся(ейся) 2 курса очной формы обучения

направления подготовки 09.03.03 Прикладная информатика

направленность (профиль) Прикладная информатика в информационной сфере

ФИО обучающегося(ейся)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Шифр \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Вид практики: учебная

Тип практики: ознакомительная

Приказ о направлении на практику: № Сз-668-Э от 13 декабря 2021 г.

Срок прохождения практики с «17» декабря 2021г. по «30» декабря 2021г.

Место прохождения практики АНОО ВО Центросоюза РФ СибУПК

Срок сдачи студентом отчета на кафедру30.12.21

1. Перечень подлежащих разработке вопросов и общее направление работы:
2. Постановка задачи, краткое описание основных средств языка, которые предполагается использовать для решения поставленной задачи.
3. Разработка алгоритма решения поставленной задачи
4. Создание интерфейса приложения.
5. Тестирование приложения.

II. Объем отчета по практике \_\_\_\_\_\_\_\_\_\_ страниц (+приложение)

Задание выдано

Руководитель практики от университета Колдунова Ирина Дмитриевна, канд. пед. наук, доцент кафедры информатики СибУПК \_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

Задание получено

Обучающийся(аяся) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

**Содержание**

[**Введение** 5](#_Toc90811128)

[**1.** **Теоретические основы разработки оконных приложений** 6](#_Toc90811129)

[**1.1.** **…** 6](#_Toc90811130)

[**1.2.** **…** 6](#_Toc90811131)

[**2.** **Разработка приложения** 7](#_Toc90811132)

[**2.1.** **Разработка сценария и алгоритма решения поставленной задачи** 7](#_Toc90811133)

[**2.2.** **Создание интерфейса приложения** 7](#_Toc90811134)

[**2.3.** **Тестирование приложения** 7](#_Toc90811135)

[**Заключение** 8](#_Toc90811136)

[**Библиографический список** 9](#_Toc90811137)

[**Приложения** 10](#_Toc90811138)

# Введение

Учебно-ознакомительная практика обеспечивает последовательность и непрерывность в формировании у студентов универсальных компетенций и навыков.

Целью данной практики является приобретение первоначального практического опыта, связанного с освоением специальности 09.03.03 Прикладная информатика, на основе разработки собственного приложения.

Учебно-ознакомительная практика ставит следующие задачи:

- обобщение, систематизация, конкретизация и закрепление теоретических знаний при разработке собственной программы, а также при помощи выполнения индивидуального задания;

- выполнение технических требований индивидуального знания;

- составление технической документации создания приложения информатизация прикладных процессов;

- овладение необходимым набором универсальных компетенций;

- получение практического опыта создания и тестирования собственного приложения;

- формирование практических умений: программирование на языке C# в соответствии с профилем подготовки, сбор необходимой информации для предметной области и т.д.

# Теоретические основы разработки оконных приложений

Язык C# был создан в 1998-2001 годах группой инженеров под руководством Андерса Хейлсберга в компании Microsoft как язык разработки приложений для платформы Microsoft .NET Framework. C# относится к семье языков с C-подобным синтаксисом. Его синтаксис наиболее близок к C++ и Java. Учитывая опыт применения языков C++ и Java, разработчики C# исключили из него некоторые модели, вызывавшие при разработке приложений проблемные ситуации. Это в некоторой степени упростило язык и сделало его более надежным.

Приведем некоторые особенности C#:

• C# создавался параллельно с .Net Framework, поэтому в полной мере учитывает все возможности FCL и CLR;

• C# является полностью объектно-ориентированным языком, и даже типы, встроенные в язык, представлены классами;

• C# является наследником языков C/C++ и сохраняет лучшие черты этих популярных языков программирования. В то же время язык стал проще и надежнее;

• благодаря каркасу .Net Framework, ставшему надстройкой над операционной системой, программисты C# получают те же преимущества работы, что и программисты Java. Эффективность кода даже повышается, поскольку исполнительная среда CLR представляет собой компилятор промежуточного языка, в то время как виртуальная Java-машина является интерпретатором байт-кода;

• мощная библиотека каркаса поддерживает удобство построения различных типов приложений на C#, позволяя достаточно просто сохранять и получать информацию из базы данных и других хранилищ данных.

# Платформа Microsoft .NET

Платформа Microsoft .NET представляет собой единую среду исполнения программ и поддержки их разработки.

Ключевыми задачами при построении платформы .NET являлись:

1. Поддержка разработки распределенных корпоративных приложений, включая серверные и мобильные компоненты, на базе Web-сервисов и XML.

2. Унификации библиотек функций для всех приложений, независимо от используемого языка программирования.

Платформа .NET основана на единой объектно-ориентированной модели; все сервисы, предоставляемые программисту платформой, оформлены в виде единой иерархии классов. Модель платформы .NET существенно упрощает разработку приложений по сравнению с программированием для Windows-платформ, где практически вся функциональность предоставлялась разработчику как неструктурированный набор функций в Windows API.

Проблема создания объектно-ориентированной надстройки над функциями Windows API решалась независимо в различных языках программирования. С появлением платформы .NET впервые в истории программирования применяется единая модель, позволяющая на равных пользоваться различными языками для создания приложений. Поскольку базовые классы .NET стали общие для всех систем программирования, то это означает, что изменилось соотношение язык - базовые функции. Если раньше каждый язык содержал свою библиотеку классов, представляющих надстройку над функциями Windows API, например MFC в VC++, то теперь язык программирования адаптируется к платформе. Библиотека классов MFC - это набор статических объектных модулей . Они подключаются к приложению на этапе компоновки исполняемого модуля программы и становятся при этом его неотъемлемой частью. В то же время .NET Class Library - это динамические библиотеки классов, которые являются составной частью операционной среды (специальным видом исполняемых модулей) и используются приложением только в момент его выполнения. Разработчики .NET лишены возможности выбора между статическими объектными библиотеками (LIB) и библиотеками классов (DLL) (за исключением тех, кто пишет на C/C++, которые занимают особое положение в средствах разработки .NET).

3. Повышения управляемости приложений с точки зрения эффективного использования ресурсов и безопасности.

# 1.2