**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МОЭВМ**

отчет

**по учебной практике**

Тема: Спутники



|  |  |  |
| --- | --- | --- |
| Студент гр. 3343 |  | Коршков А.А. |
| Руководитель |  | Бахарев Д.С. |

Санкт-Петербург

2025

**ЗАДАНИЕ**

**на УЧЕБНУЮ практику**

|  |  |  |
| --- | --- | --- |
| Студент Коршков А. А. | | |
| Группа 3343 | | |
| Тема практики: Спутники | | |
| Задание на практику:  Разработать приложение с GUI для отображения информации о спутниках на основе данных TLE.  Программа должна отобразить следующую статистику:   * Общее количество спутников, данные по которым загружены; * Дата самых старых данных из загруженных; * Разбитое по годам количество запущенных спутников; * Разбитое по градусам (с точностью до одного градуса) количество спутников с разным наклонением орбиты. | | |
| Сроки прохождения практики: 25.06.2025 – 08.07.2025 | | |
| Дата сдачи отчета: 08.07.2025 | | |
| Дата защиты отчета: 08.07.2025 | | |
|  | | |
| Студент |  | Коршков А.А. |
| Руководитель |  | Бахарев Д.С. |

**Аннотация**

Целью практики является изучение основ методологии DevOps иприменение инструментов для автоматизации процессов разработки. В ходе практики осваиваются технологии Gitflow, Docker и GitLab CI, которые позволяют управлять версиями кода, развертывать приложения в контейнерах и

настраивать непрерывную интеграцию. В рамках практики выполняются лабораторные работы, включающие работу с Linux, создание Docker-контейнеров и управление ими, а также настройку CI/CD-пайплайнов в GitLab CI. Полученные навыки помогают понять принципы DevOps и их применение в реальных проектах.

Кратко (в 8-10 строк) указать цель и основное содержание практики.

Целью практики является создание программы для отображения информации о спутниках на основе данных TLE. В ходе практики осваиваются технологии фреймворка QT, принципы построения GUI интерфейсов, умение работать с форматом TLE, извлекать из него необходимую информацию и язык программирования C++. В рамках практики создаётся интерфейс для окна с информацией о спутниках,. Полученные навыки помогают понять принципы построения понятного интерфейса, .

**содержание**

|  |
| --- |
|  |

[ВВЕДЕНИЕ 5](#_Toc202432541)

[1. ВЫПОЛНЕНИЕ РАБОТЫ 6](#_Toc202432542)

[1.1. Описание использованных технологий 6](#_Toc202432543)

[1.2. Полученные результаты 7](#_Toc202432544)

[ЗАКЛЮЧЕНИЕ 8](#_Toc202432545)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 9](#_Toc202432546)

[ПРИЛОЖЕНИЕ А. СНИМКИ ЭКРАНА ПРОГРАММЫ 10](#_Toc202432547)

[ПРИЛОЖЕНИЕ Б. ОТЗЫВ О ПРОХОЖДЕНИИ ПРАКТИКИ 11](#_Toc202432548)

1. Описание постановки задачи (1-2 страницы А4)

2. Список и краткое описание новых языков программирования и (или) технологий, которые нужно было изучить для решения задачи (1-2 страницы А4)

3. Краткое описание полученных результатов (1-2 страницы А4)

# **ВВЕДЕНИЕ**

Цель практики – разработать на C++ с использованием Qt Widgets и CMake приложение для загрузки, парсинга и визуализации статистики по TLE‑данным, закрепив навыки работы с GUI, сетевыми запросами и обработкой данных.

Задачи:

1. Написать синтаксический анализатор (парсер) для файлов TLE формата
   1. Создать структуру для хранения записи об одном космическом объекте.
   2. Создать структуру, которая будет хранить необходимую обработанную информацию, в том числе и набор полученных объектов.
   3. Создать функции, реализующие загрузку и обработку информации из файлов. Применить язык регулярных выражений для проверки соответствия данных с шаблоном TLE формата.
   4. Реализовать загрузку и обработку информации из сетевого хранилища, c помощью специальных методов из QtNetwork.
   5. Реализовать функцию проверки контрольной суммы для определения целостности данных.
   6. Реализовать сигналы, которые будут помогать GUI приложению выполнять определённую логику
2. Создать GUI для главного окна и окна с информацией о спутниках.
   1. Реализовать в главном окне необходимый функционал для загрузки из файлов или из сети.
   2. Информационное окно должно в себе содержать в понятном виде данные согласно заданию,
   3. Пользователь должен иметь возможность создать ещё окно с уже другими данными, при этом не теряя информацию с предыдущего
   4. Реализовать
3. Подключить парсер к логике работы GUI приложения
   1. В главном окне реализовать слоты, которые будут обрабатывать сигналы нажатия с кнопок и выполнять необходимую логику
   2. В окне с информацией о спутниках

# **1. ВЫПОЛНЕНИЕ РАБОТЫ**

## 1.1. Описание использованных технологий

В ходе выполнения учебной практики были использованы следующие технологии:

1. ОС Windows 11 и Fedora Linux
   1. В ходе практики использовалась операционная система Windows 11 как основная ОС для создания и использования данной программы.
   2. Также была использована операционная система Fedora Workstation 42 для создания и тестирования программы под семейство операционных систем на базе ядра Linux.
2. Фреймворк QT5/QT6
   1. Фреймворк для разработки кроссплатформенного программного обеспечения на языке программирования C++.
   2. Для разработки программы был использован традиционный подход на основе Qt Widgets, который использует систему виджетов. Данный подход является очень проверенным для создания качественных настольных приложений, где в первую очередь важна фунциональность.
   3. Для создания интерфейсов создавались .ui файлы в формате XML, которые компилировались в C++ и с помощью специальных классов можно настраивать поведение каждого объекта в окне.
   4. Основным для разработки использовалась версия QT 6.8.3 LTS, для создания современного дизайна и использования более современного стека технологий, но также осуществлена поддержка на QT 5.15.17 LTS для совместимости с более старыми версиями операционных систем.
3. Язык программирования C++
   1. Для разработки приложения с концепцией Qt Widgets необходимо использование языка C++, который позволяет точно работать с объектами, реализовывать логику окон и их внутренних объектов и форм.
   2. В концепции QT Widgets используется связка «Сигнал-Слот», где сигналом является какое-то действие в программе, а слот является перехватчиком данного сигнала и выполнение необходимой логики.
4. Git
   1. Git – распределенная система контроля версий, необходимая для совместной работы и управления исходным кодом. Использовалась для сохранения изменений и улучшений в коде программы для того, чтобы можно было быстро перенести программу на другую операционную систему и откатить изменения в случае регрессии.

## 1.2. Полученные результаты

В результате прохождения учебной практики были получены следующие

результаты:

Получен опыт в создании адаптивного интерфейса под разный размер экрана

Github репозиторий с исходниками представлен по данной ссылке

<https://github.com/KorzikAlex/Satellites>

Внешний вид программы представлен в приложении А.

Отзыв о прохождении учебной практики в приложении Б.

# **ЗАКЛЮЧЕНИЕ**

В ходе прохождения практики были освоены основные принципы создания и построения графических приложений на базе фреймворка QT. Было написано приложение, которое позволяет получить TLE данные из файла или из сетевого хранилища и вывести необходимую информацию, с возможностью её сохранения в файл. Все поставленные учебные задачи выполнены в полном объеме. Полученные знания и навыки соответствуют современным требованиям к построению специализированных графических приложений и могут быть применены в дальнейшей профессиональной деятельности.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. TLE — двухстрочный набор данных. Формат // Space-π URL: https://spacepi.space/wiki/article/tle-dvustrochnyj-nabor-dannyh-format/ (дата обращения: 25.06.2025).

2. TLE // Википедия URL: https://ru.wikipedia.org/wiki/TLE (дата обращения: 25.06.2025).

3. TLE // Wikipedia (English Version) URL: https://en.wikipedia.org/wiki/TLE (дата обращения: 25.06.2025).

4. Qt Documentation // Qt Group URL: https://doc.qt.io/ (дата обращения: 25.06.2025).

5. Main // Qt Wiki URL: https://wiki.qt.io/ (дата обращения: 25.06.2025).

6. Руководство по языку программирования C++ // Metanit URL: https://metanit.com/cpp/tutorial/ (дата обращения: 26.06.2025).

7. Руководство по программированию с помощью фреймворка Qt и языка С++ // Metanit URL: https://metanit.com/cpp/qt/ (дата обращения: 26.06.2025).

8. Celestrak: Current GP Element Sets // Celestrak URL: https://celestrak.org/NORAD/elements/ (дата обращения: 30.06.2025).

9. Regex101: build, test and debug // Regular expressions 101 URL: https://regex101.com/ (дата обращения: 27.06.2025).

10. QML vs Qt Widgets – detailed comparison // Scythe Studio URL: https://scythe-studio.com/en/blog/qml-vs-qt-widgets-detailed-comparison (дата обращения: 25.06.2025).

# **ПРИЛОЖЕНИЕ А. СНИМКИ ЭКРАНА ПРОГРАММЫ**

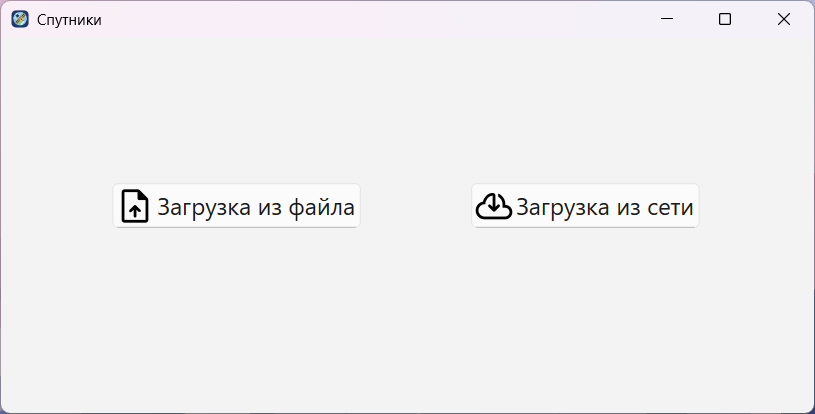


Рисунок 1 - Главное окно программы

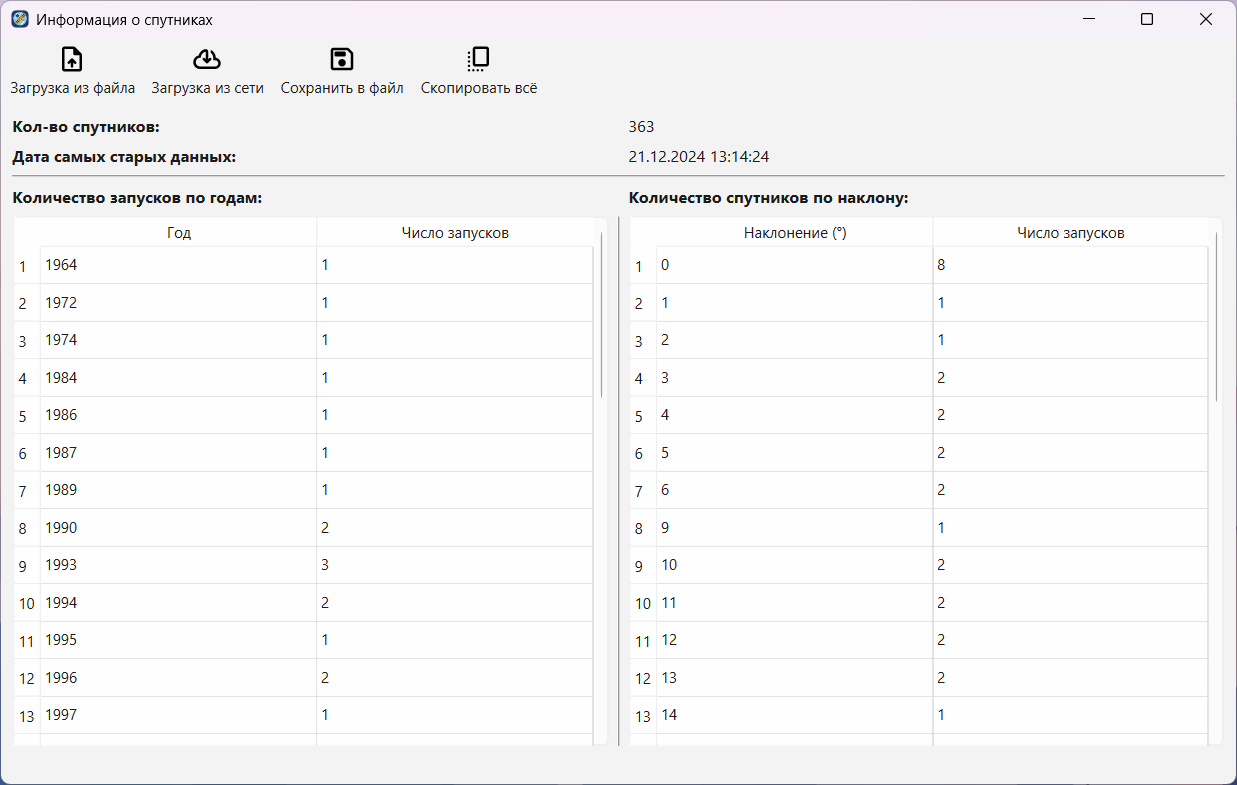


Рисунок 2 - Окно с информацией о спутниках

# **ПРИЛОЖЕНИЕ Б. ОТЗЫВ О ПРОХОЖДЕНИИ ПРАКТИКИ**