

**МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ЛЬВІВСЬКА ПОЛІТЕХНІКА**

**Кафедра ЕОМ**



**АВТОМАТИЗОВАНЕ ПРОЕКТУВАННЯ  
КОМП'ЮТЕРНИХ СИСТЕМ**

**Task 4. Create doxygen documentation**

*Виконав:*

*Ст. гр. КІ-402*

*Павлосюк В.П.*

*Прийняв:*

*Федак П.Р.*

Львів 2024

**Мета:** Створити doxygen документацію для свого проекту.

### Завдання

1. Add doxygen comments for all public functions, classes, properties, fields...
2. Generate documentation based on doxygen comments
3. Required steps

### Варіант:

19	tik-tac-toe 3x3	XML
----	-----------------	-----

### Теоретичні відомості

Doxygen — це потужний інструмент для автоматичного створення документації з вихідного коду програмного забезпечення. Він використовується для генерації читабельної та структурованої документації на основі спеціальних коментарів, які розробники додають у свій код. Основною перевагою Doxygen є те, що він дозволяє підтримувати документацію актуальною, оскільки вона генерується безпосередньо з вихідного коду. Інструмент підтримує велику кількість мов програмування, таких як C, C++, Java, Python, PHP, і навіть спеціальні формати, як-от Qt, що робить його універсальним для багатьох проєктів.

Коментарі, що використовуються для створення документації, пишуться у спеціальному форматі, який розпізнається Doxygen. Наприклад, коментарі містять ключові слова або теги, такі як `@brief`, `@param`, `@return`, які пояснюють призначення функцій, параметрів чи змінних у коді. Це дозволяє структуровано описувати класи, методи, константи, змінні та сигнали. Doxygen може створювати документацію у різних форматах, включаючи HTML, LaTeX, CHM і PDF, що зручно для різних видів поширення — як для внутрішньої розробки, так і для публічного використання.

Однією з ключових функцій Doxygen є можливість візуалізації зв'язків між класами та файлами у вигляді графів і діаграм, що особливо корисно для великих і складних проєктів. Окрім цього, Doxygen інтегрується з багатьма іншими інструментами, наприклад, Graphviz, для створення ще більш наочного графічного представлення структури проєкту. У Qt-проєктах Doxygen часто використовується для документування класів, методів і сигналів/слотів, що забезпечує зрозумілу структуру і полегшує підтримку коду.

Doxygen не лише сприяє підтримці високого рівня якості документації, а й спрощує процес ознайомлення нових членів команди з кодом. Оскільки

документація автоматично синхронізується з кодом, це зменшує ризик розбіжностей між описом і реальною реалізацією. Таким чином, використання Doxygen стає важливим елементом сучасної розробки програмного забезпечення, особливо у великих командах або в проєктах з відкритим кодом, де документація є критичною для забезпечення прозорості та зрозумілості коду.

## Деталі реалізації

Додав doxygen коментарі до проєкту:

```
*/
class SerialConnection : public QObject {
    Q_OBJECT

public:
    /**
     * @brief Constructs a SerialConnection object.
     * @brief Initializes the serial connection and sets up the serial port for communication.
     * @param parent The parent QObject for this instance.
     */
    explicit SerialConnection(QObject *parent = nullptr);

    /**
     * @brief Sends a command to the serial device.
     * @brief This function sends a command string to the connected serial device (e.g., Arduino).
     * @param command The command to send to the serial device.
     */
    void sendCommand(const QString &command);

signals:
    /**
     * @brief Emitted when the board state is received from the serial device.
     * @brief This signal is emitted when a valid response (e.g., board state) is received from the serial device.
     * @param state The board state as a string.
     */
    void boardStateReceived(const QString &state);

    /**
     * @brief Emitted when an error occurs with the serial connection.
     */
}
```

Рис. 1. Фрагмент коду клієнтської частини з doxygen коментарями

```
C:\University\CSAD\SW_ClientGame>doxygen -g Doxyfile

Configuration file 'Doxyfile' created.

Now edit the configuration file and enter

    doxygen

to generate the documentation for your project

C:\University\CSAD\SW_ClientGame>doxygen Doxyfile
Doxygen version used: 1.12.0 (c73f5d30f9e8b1df5ba15a1d064ff2067cbb8267)
Searching for include files...
Searching for example files...
Searching for images...
Searching for dot files...
Searching for msc files...
```

Рис. 2. Створення документації на основі вихідних файлів і коментарів Doxygen через команди в терміналі.

Ім'я	Дата змінення	Тип	Розмір
class_serial_connection-members.html	25.11.2024 17:12	Chrome HTML Do...	6 КБ
classes.html	25.11.2024 17:12	Chrome HTML Do...	5 КБ
clipboard.js	25.11.2024 17:12	JSFile	3 КБ
closed.png	25.11.2024 17:12	Файл PNG	1 КБ
cookie.js	25.11.2024 17:12	JSFile	2 КБ
doc.svg	25.11.2024 17:12	Chrome HTML Do...	2 КБ
docd.svg	25.11.2024 17:12	Chrome HTML Do...	2 КБ
doxygen.css	25.11.2024 17:12	CSSfile	49 КБ
doxygen.svg	25.11.2024 17:12	Chrome HTML Do...	16 КБ
doxygen_crawl.html	25.11.2024 17:12	Chrome HTML Do...	3 КБ
dynsections.js	25.11.2024 17:12	JSFile	8 КБ
files.html	25.11.2024 17:12	Chrome HTML Do...	5 КБ
folderclosed.svg	25.11.2024 17:12	Chrome HTML Do...	2 КБ
folderclosedd.svg	25.11.2024 17:12	Chrome HTML Do...	2 КБ
folderopen.svg	25.11.2024 17:12	Chrome HTML Do...	4 КБ
folderopend.svg	25.11.2024 17:12	Chrome HTML Do...	4 КБ
functions.html	25.11.2024 17:12	Chrome HTML Do...	6 КБ
functions_func.html	25.11.2024 17:12	Chrome HTML Do...	6 КБ
gamelogic_8h_source.html	25.11.2024 17:12	Chrome HTML Do...	16 КБ
hierarchy.html	25.11.2024 17:12	Chrome HTML Do...	6 КБ
index.html	25.11.2024 17:12	Chrome HTML Do...	4 КБ
jquery.js	25.11.2024 17:12	JSFile	173 КБ
mainwindow_8h_source.html	25.11.2024 17:12	Chrome HTML Do...	15 КБ

Рис. 3. Згенерована документація в html

TicTacToeClient

Main Page

Classes

Files

Search

SerialConnection Class Reference

This class handles serial communication with an external device (e.g., Arduino). It is responsible for setting up the serial port, sending commands, handling responses, and managing serial port errors. [More...](#)

```
#include <serialconnection.h>
```

Inheritance diagram for SerialConnection:

```

graph BT
    SerialConnection --|> QObject
  
```

Signals

void boardStateReceived (const QString &state)

Emitted when the board state is received from the serial device. This signal is emitted when a valid response (e.g., board state) is received from the serial device.

void errorOccurred (const QString &error)

Emitted when an error occurs with the serial connection. This signal is emitted when an error occurs, such as failure to open the serial port or data transmission issues.

Public Member Functions

SerialConnection (QObject \*parent=nullptr)

Constructs a **SerialConnection** object. Initializes the serial connection and sets up the serial port for communication.

void sendCommand (const QString &command)

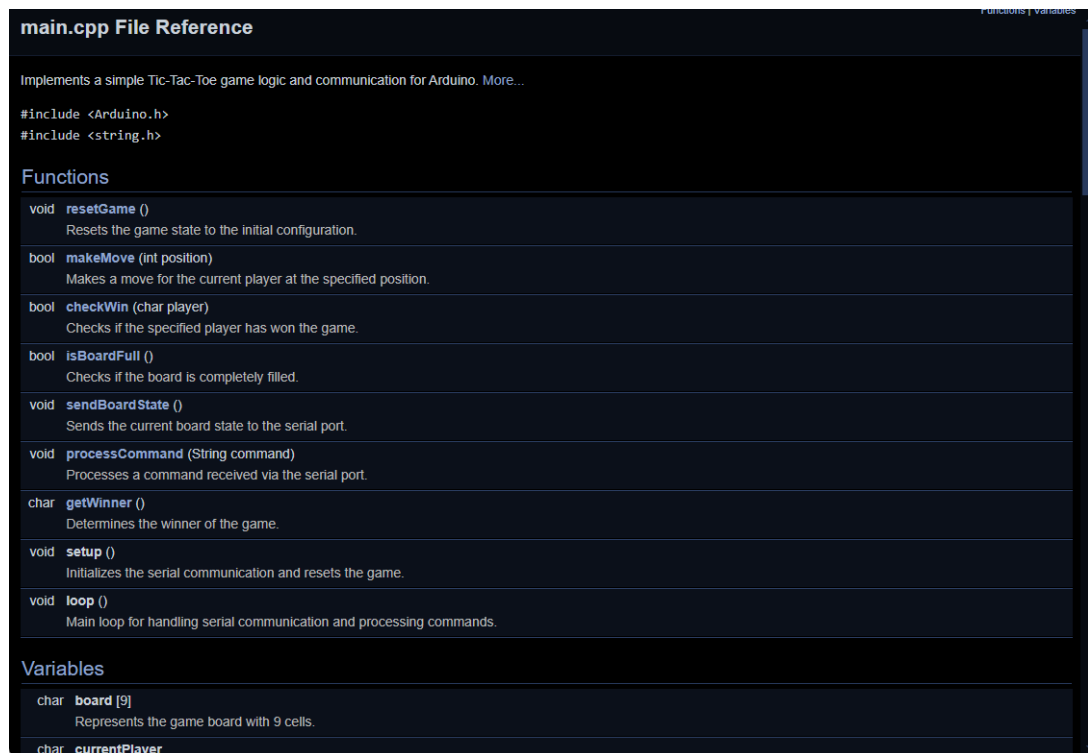
Sends a command to the serial device. This function sends a command string to the connected serial device (e.g., Arduino).

Detailed Description

This class handles serial communication with an external device (e.g., Arduino). It is responsible for setting up the serial port, sending commands, handling responses, and managing serial port errors.

Handles communication with a serial device (e.g., Arduino) via a serial port. This class manages the serial port

Рис. 4. Фрагмент документації класу *SerialConnection* клієнта



*Рис. 5. Фрагмент документації серверної частини*

## Висновок:

У ході виконання лабораторної роботи були успішно реалізовані всі поставлені завдання. Додано Doxygen-коментарі до всіх публічних функцій, класів, властивостей і полів. Це дозволило чітко та детально задокументувати кожен елемент коду, що сприяє кращому розумінню його структури і функціональності. Згенеровано документацію на основі створених коментарів за допомогою Doxygen. Документація охоплює як технічні аспекти реалізації, так і опис доступного функціоналу.

## Список використаної літератури:

1. GitHub Docs, <https://docs.github.com/>
2. CSAD instructions for practical tasks and coursework