**LAPORAN**

**TUGAS AHIR**

Mata Kuliah Pemrograman Berorientasi Objek



Disusun Oleh:

1.      Syailendra Julian W

2.      M. Hafid

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS NUSANTARA PGRI KEDIRI

TAHUN 2023

**Daftar Isi**

[Flowchart System 3](#_Toc155179247)

[Class Diagram 3](#_Toc155179248)

[Modul-modul 4](#_Toc155179249)

[Fitur 5](#_Toc155179250)

[ Mysql.connect 5](#_Toc155179251)

[ Change background color 6](#_Toc155179252)

[ Save doc & pdf 6](#_Toc155179253)

[ Font 7](#_Toc155179254)

[ Cut,copy,paste, select all 7](#_Toc155179255)

[ Insert image 8](#_Toc155179256)

[ Spacing 8](#_Toc155179257)

[ Symbol & emoji 9](#_Toc155179258)

[ Count (word, space) 9](#_Toc155179259)

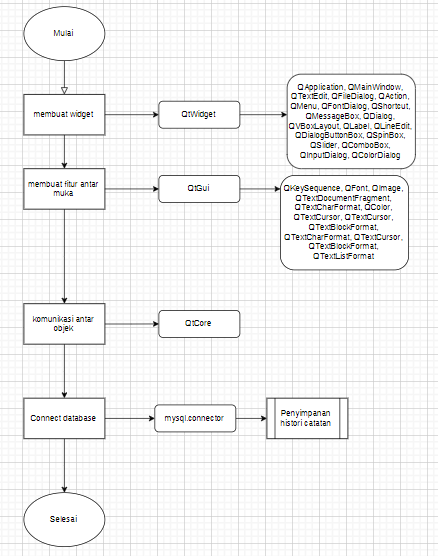
[ Bullets & Numbering 10](#_Toc155179260)

[ Alignment 10](#_Toc155179261)

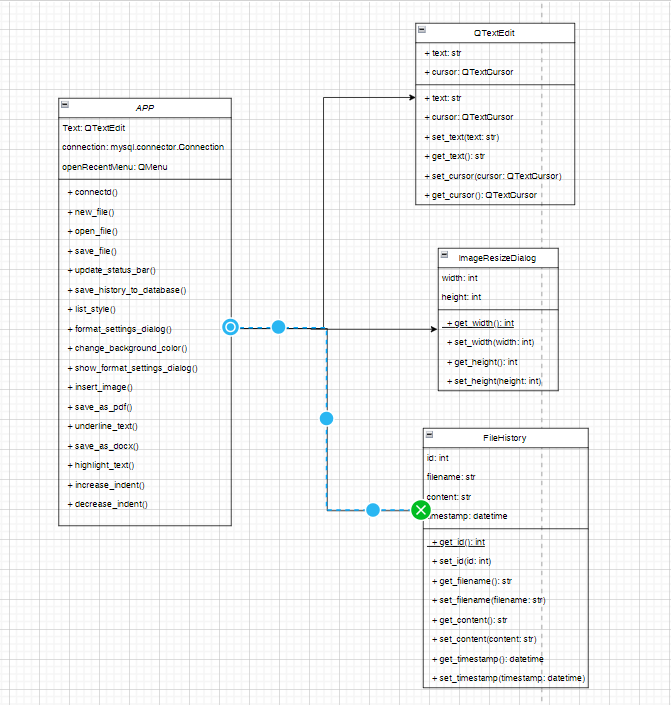
[ Increase & decrease indent 10](#_Toc155179262)

[Daftar Pustaka 11](#_Toc155179263)

# Flowchart System



# Class Diagram



# Modul-modul

import sys

import mysql.connector

from PyQt5.QtWidgets import QApplication, QMainWindow, QTextEdit, QFileDialog, QAction, QMenu, QFontDialog, QShortcut, QMessageBox, QDialog, QVBoxLayout, QLabel, QLineEdit, QDialogButtonBox, QSpinBox, QSlider, QComboBox, QInputDialog, QColorDialog

from PyQt5.QtGui import QKeySequence, QFont, QImage, QTextDocumentFragment, QTextCharFormat, QColor, QTextCursor, QTextCursor, QTextBlockFormat, QTextCharFormat, QTextCursor, QTextBlockFormat, QTextListFormat

from PyQt5.QtCore import Qt

from reportlab.pdfgen import canvas

import docx as Document

* Sys

Modul ini sudah ada pada saat python diinstall dan fungsinya untuk akses file local

* Mysql.connector

Bagian ini digunakan untuk menghubungkan source code dengan basis data mysql

* Pyqt5
* QtWidgets

Digunakan untuk membuat jendela pada aplikasi GUI jadi semua turunan modulnya berfungsi untuk tampilan luar aplikasi

* QtGui

Dimodul ini disimpan komponen-komponen interface dengan pengguna

* QtCore

Inti dari non-GUI, digunakan oleh modul lain

* Reportlab

Bagian ini untuk mencetak file bertipe pdf maupun grafis

* Python-docx

Digunakan untuk membaca, membuat dan mengupdate file ms.word

Fitur

## Mysql.connect

Aplikasi/GUI ini menggunakan mysql untuk penyimpanan, walau sebenarnya lebih baik menggunakan path yang nantinya disimpan langsung di local

def connectd(self):

        self.connection = mysql.connector.connect(

            host="localhost",

            user="root",

            password="",

            database="history"

        )

        self.cursor = self.connection.cursor()

        create\_table\_query = """

        CREATE TABLE IF NOT EXISTS file\_history (

            id INT AUTO\_INCREMENT PRIMARY KEY,

            filename VARCHAR(255),

            content TEXT,

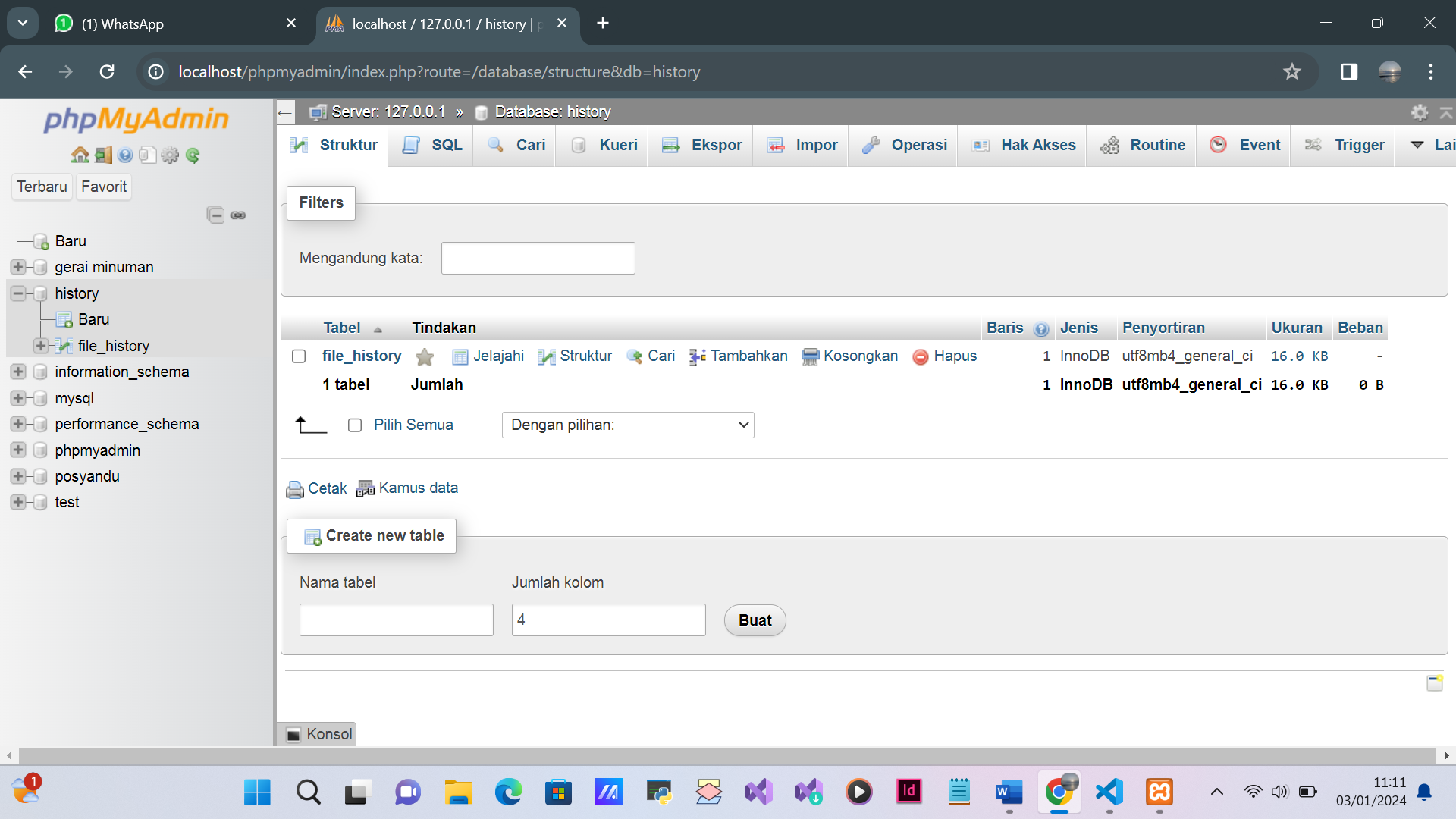
            timestamp TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

        )

        """

        self.cursor.execute(create\_table\_query)

        self.connection.commit()



## Change background color

Notepad ini memiliki fitur mengganti warna background sesuai yang diinginkan pengguna

def change\_background\_color(self):

     color = QColorDialog.getColor(self.palette().color(self.backgroundRole()), self, "Select Background Color")

     if color.isValid():

        self.set\_background\_color(color)

    def set\_background\_color(self, color):

     palette = self.palette()

     palette.setColor(self.backgroundRole(), color)

     self.setPalette(palette)

## Save doc & pdf

Fitur ini digunakan untuk merubah jenis file .text ke pdf/docx

def save\_as\_pdf(self):

        filename, \_ = QFileDialog.getSaveFileName(self, "Save as PDF", "", "PDF Files (\*.pdf);;All Files (\*)")

        if filename:

            try:

                pdf\_canvas = canvas.Canvas(filename)

                pdf\_canvas.setFont("Helvetica", 12)  # You can change the font and size

                pdf\_canvas.drawString(72, 800, self.text.toPlainText())

                pdf\_canvas.save()

            except Exception as e:

                QMessageBox.critical(self, "Error", f"An error occurred while saving as PDF:\n{str(e)}")

    def save\_as\_docx(self):

        filename, \_ = QFileDialog.getSaveFileName(self, "Save as DOCX", "", "Word Documents (\*.docx);;All Files (\*)")

        if filename:

            try:

                if not filename.lower().endswith(".docx"):

                    filename += ".docx"

                document = Document()

                document.add\_paragraph(self.text.toPlainText())

                document.save(filename)

            except Exception as e:

                QMessageBox.critical(self, "Error", f"An error occurred while saving as DOCX:\n{str(e)}")

## Font

Terinspirasi dari ms.word, fitur ini digunakan untuk mengganti font penulisan dalam notepad

def add\_font\_actions\_to\_menu(self):

        font\_list = ["Arial", "Times New Roman", "Courier New", "Verdana", "Serif", "Sans Serif", "Monospace", "Cursive", "Fantasy"]

        for font\_name in font\_list:

            action = QAction(font\_name, self)

            action.triggered.connect(lambda \_, font\_name=font\_name: self.change\_font\_for\_selected\_word(font\_name))

            self.context\_menu.addAction(action)

    def change\_font\_for\_selected\_word(self, font\_name):

        cursor = self.text.textCursor()

        selected\_text = cursor.selectedText()

        if selected\_text:

            char\_format = QTextCharFormat()

            char\_format.setFont(QFont(font\_name, 12))

            cursor.mergeCharFormat(char\_format)

## Cut,copy,paste, select all

Untuk yang satu ini adalah fitur dasar yang Sebagian besar aplikasi untuk menulis memilikinya

def cut(self):

        self.text.cut()

    def copy(self):

        self.text.copy()

    def paste(self):

        clipboard = QApplication.clipboard()

        mime\_data = clipboard.mimeData()

        if mime\_data.hasText():

            text\_fragment = QTextDocumentFragment.fromHtml(mime\_data.text())

            cursor = self.text.textCursor()

            cursor.insertFragment(text\_fragment)

## Insert image

Notepad ini juga bisa digunakan untuk menginsertkan gambar

def insert\_image(self):

        options = QFileDialog.Options()

        options |= QFileDialog.ReadOnly

        file\_name, \_ = QFileDialog.getOpenFileName(self, "Open Image File", "", "Images (\*.png \*.jpg \*.bmp \*.gif);;All Files (\*)", options=options)

        if file\_name:

            try:

                dialog = ImageResizeDialog(self)

                if dialog.exec\_() == QDialog.Accepted:

                    width, height = dialog.get\_size()

                    image = QImage(file\_name)

                    if width > 0 and height > 0:

                        image = image.scaled(width, height, Qt.KeepAspectRatio)

                    cursor = self.text.textCursor()

                    cursor.insertImage(image)

            except Exception as e:

                QMessageBox.critical(self, "Error", f"An error occurred while inserting the image:\n{str(e)}")

## Spacing

Fitur ini berfungsi untuk mengatur jarak antar baris dalam paragraf

def show\_format\_settings\_dialog(self):

     dialog = FormatSettingsDialog(self)

     result = dialog.exec\_()

     if result == QDialog.Accepted:

        line\_spacing, paragraph\_spacing, background\_color = dialog.get\_format\_settings()

        self.set\_text\_formatting(line\_spacing, paragraph\_spacing)

        self.set\_background\_color(background\_color)

def set\_text\_formatting(self, line\_spacing, paragraph\_spacing):

        cursor = self.text.textCursor()

        block\_format = cursor.blockFormat()

        block\_format.setLineHeight(line\_spacing, QTextBlockFormat.ProportionalHeight)

        block\_format.setBottomMargin(paragraph\_spacing)

        block\_format.setTopMargin(paragraph\_spacing)

        cursor.setBlockFormat(block\_format)

        self.text.setTextCursor(cursor)

        char\_format = cursor.charFormat()

        char\_format.setFontPointSize('font\_size')

        cursor.setCharFormat(char\_format)

        self.text.setTextCursor(cursor)

## Symbol & emoji

Fitur tambahan untuk mempercantik karya tulis

def show\_symbol\_dialog(self):

        symbol\_list = ["©", "®", "™", "€", "£", "¥", "§"]

        symbol, ok = QInputDialog.getItem(self, "Insert Symbol", "Select a symbol:", symbol\_list, 0, False)

        if ok and symbol:

            cursor = self.text.textCursor()

            cursor.insertText(symbol)

    def show\_emoji\_dialog(self):

        emoji\_list = ["😊", "😍", "👍", "🎉", "❤️", "🚀"]

        emoji, ok = QInputDialog.getItem(self, "Insert Emoji", "Select an emoji:", emoji\_list, 0, False)

        if ok and emoji:

            cursor = self.text.textCursor()

            cursor.insertText(emoji)

## Count (word, space)

Fitur ini digunakan untuk menghitung jumlah kata dan spasi

def update\_status\_bar(self):

        text = self.text.toPlainText()

        word\_count = len(text.split())

        space\_count = text.count(' ')

        status\_text = f"Words: {word\_count}, Spaces: {space\_count}"

        self.statusBar.showMessage(status\_text)

## Bullets & Numbering

Biasanya digunakan untuk daftar nama benda atau lainnya

def set\_list\_style(self, list\_style):

     cursor = self.text.textCursor()

     block\_format = cursor.blockFormat()

     list\_format = QTextListFormat()

     list\_format.setStyle(list\_style)

     cursor.mergeBlockFormat(block\_format)

     cursor.createList(list\_format)

## Alignment

Digunakan untuk mengatur letak paragraf

def set\_alignment(self, alignment):

        cursor = self.text.textCursor()

        block\_format = cursor.blockFormat()

        block\_format.setAlignment(alignment)

        cursor.setBlockFormat(block\_format)

## Increase & decrease indent

Fungsinya seperti tombol tab pada keyboard

def increase\_indent(self):

        cursor = self.text.textCursor()

        cursor.insertText("\t")

    def decrease\_indent(self):

        cursor = self.text.textCursor()

        cursor.movePosition(QTextCursor.StartOfBlock)

        cursor.movePosition(QTextCursor.NextCharacter, QTextCursor.KeepAnchor, 1)

        selected\_text = cursor.selectedText()

        if selected\_text == "\t":

            cursor.removeSelectedText()

# Daftar Pustaka

doc.qt.io. (n.d.). QWidget Class.

docs.reportlab.com. (n.d.). QtGui.

tinkq.blogspot.com. (n.d.). Fungsi Decrease indent dan Increase indent pada Microsoft Word.