

# MELEC PROJECT

DUFRENOT Dwyane

LACAMPAGNE Arthur

DU COUEDIC DE KERGOALER Alban

Link Github : <https://github.com/arthurlcp/Aldwar>

# PROJECT OVERVIEW AND MOTIVATION

A software solution to help electricians record materials used on job sites.

Currently, electricians rely on paper or Excel sheets. Our goal is to modernize and simplify material management.

Better traceability, centralized data, and less paperwork.

# CHOSEN SDLC METHODOLOGY

We chose the **Incremental SDLC model** for the MELEC Software project because it allows us to build the software in small, manageable parts. This approach lets us:

- Develop the project in stages, where each increment builds on the previous one.
- Prioritize the most critical features first (like material logging), so we always have a functional base.
- Adapt to new ideas and changes more easily, based on feedback or new requirements.

# CHOSEN SDLC METHODOLOGY

## Why this model fits our project:

- **Flexibility:** As we discussed with the electrician, some requirements evolved during the project. The incremental model allows us to integrate those changes smoothly.
- **Risk Management:** Each increment is tested and validated, reducing the risk of errors.
- **Faster Delivery:** We can deliver a working prototype sooner, making it easier to get user feedback and improve.

# REQUIREMENTS GATHERING & ANALYSIS

## How:

- Direct interviews with a company owner to understand needs
- Analysis of existing software to identify gaps

## Key Requirements:

- Easy recording of materials per job site
- Ability to edit and save circuit diagrams
- Database with material references

# **SOFTWARE REQUIREMENTS SPECIFICATION**

## **Main Features:**

- Add, update, and delete materials
- Visually build electrical circuits
- Save and export plans

# **SOFTWARE REQUIREMENTS SPECIFICATION**

## **Constraints:**

- **Private software (not a public website):**  
The MELEC Software is designed for private use by electricians, so it does not require a web-based deployment or complex hosting solutions.
- **Intuitive interface, aesthetics not the main priority:**  
The focus is on usability and practical functionality, ensuring that electricians can quickly understand and use the software, even if it is not visually fancy.

# SYSTEM AND SOFTWARE DESIGN

## Technical Stack:

- **Python:** Entirely used for coding (logic, GUI, data management).
- **Tkinter:** GUI library for a simple, user-friendly interface.
- **SQL Database:** Secure and scalable material tracking.
- **File Storage:** PDF and editable format for plans.



# **SYSTEM AND SOFTWARE DESIGN**

## **Key Strengths:**

- Fully Python-based for consistency and easier maintenance.
- Fast updates and simple debugging with one language.
- Clear separation of interface, logic, and database.
- Adaptable design for future improvements.

# IMPLEMENTATION PHASE

## **Technologies and Tools:**

**Python:** Entirely used for coding (logic, interface, database handling).

**Tkinter:** GUI framework for a clear and intuitive interface.

**Database Management:** Handled within Python using SQL libraries.

# IMPLEMENTATION PHASE

## **Features Developed:**

- Interface to add, update, and manage materials by site.
- Visual circuit builder for electricians.

## **Visual circuit builder for electricians:**

- - Enables users to create simple circuit diagrams within the application,**
- **making it easier to plan and visualize electrical installations.**

# IMPLEMENTATION PHASE

## Challenges Faced:

- Designing a user-friendly GUI that meets electricians' needs.
- Implementing plan storage that is both editable and shareable.
- Ensuring reliable data handling entirely in Python.

# IMPLEMENTATION PHASE

```
def ajouter_materiau():
    nom = input("Nom du matériau: ")
    description = input("Description: ")
    conn = connect_db()
    c = conn.cursor()
    c.execute("INSERT INTO matériaux (nom, description) VALUES (?, ?)", (nom, description))
    conn.commit()
    conn.close()
    print("Matériau ajouté.")

def associer_materiaux():
    chantier_id = input("ID du chantier: ")
    lister_materiaux()
    materiau_id = input("ID du matériau à associer: ")
    conn = connect_db()
    c = conn.cursor()
    c.execute("INSERT INTO chantier_materiaux (chantier_id, materiau_id) VALUES (?, ?)", (chantier_id, materiau_id))
    conn.commit()
    conn.close()
    print("Matériau associé au chantier.")
```

# TESTING STRATEGY

## Types of Tests:

**Unit Testing:** Verified core features (adding materials, exporting plans).

**Integration Testing:** Checked interactions between interface, database, and file handling.

**User Testing:** Real electricians tested usability and practicality.

# TESTING STRATEGY

## Tools:

- Python's `unittest` for automated checks.
- Manual tests for real-world scenarios.

## Goals:

- Ensure software reliability and data consistency.
- Validate user-friendly design and fix early bugs.

# TEST RESULTS AND VALIDATION

## **Expected Results:**

- Functional software for recording materials
- PDF export working correctly
- Editable plan storage validated

## **Validation:**

- Aligned with the needs gathered from the electrician
- Real-user tests to confirm usability



# PROCESS CHALLENGES AND LESSONS LEARNED

## Challenges:

- Developing both the Python logic and a user-friendly GUI (Tkinter).
- Managing editable plan storage for easy updates.
- Completing all tasks within a 30-hour project timeframe.

# PROCESS CHALLENGES AND LESSONS LEARNED

## **Lessons Learned:**

- The importance of clearly understanding user needs from the beginning.
- Flexibility is key to adapt at each stage of the project.
- Prioritizing essential features ensures the project stays focused and achievable.

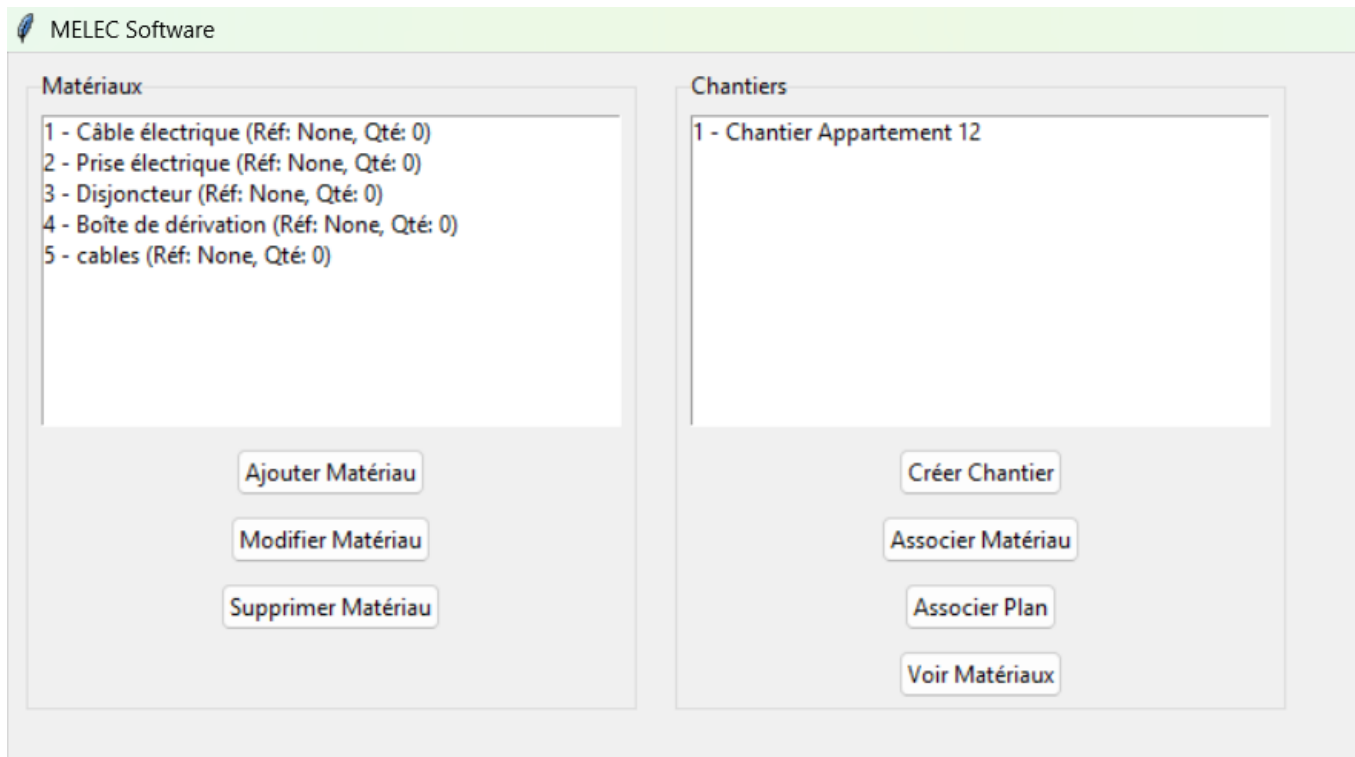
# DEMO OVERVIEW

We will showcase:

- **Adding and managing materials** for each job site.
- **Building simple electrical circuits** and linking them to materials.
- **Saving and exporting plans** as editable files and PDFs.
- **User-friendly interface** with real-time database updates.

This demo will highlight how MELEC Software makes daily work easier for electricians.

# DEMO OVERVIEW



# CONCLUSION AND FUTURE WORK

## **Achievements:**

- Centralized software for material management
- Reduced reliance on paper

## **Possible Future Improvements:**

- Automatic stock management
- Integration with supplier catalogs
- Expansion to other trades (plumbers, masons...)

# Q&A / FEEDBACK

We're now ready to answer any questions you may have about MELEC Software.

We'd love to hear your feedback on:

- Features you find most useful
- Any challenges you think we should address next
- Ideas for future improvements or expansions