

Chemical and Inventory Management System
SoFab Inks
Team 3

Hilton Benson, Blake Hourigan, CJ Johnstone, Maggie Jackey

November 29, 2024

Contents

1	Introduction/Executive Summary	4
1.1	Introducing SoFab Inks	4
1.2	SoFab Inks Inventory Management Problem	4
2	System Description	5
2.1	Needs Assessment/System Requirements	5
2.2	Initial System Specification	5
2.3	Final Specifications	5
2.4	System Diagrams	5
2.5	Hardware Overview Diagram	5
2.6	Software Overview Diagram	5
2.7	Economical, Technical, and Time Constraints	6
3	Detailed Implementation	6
3.1	Hardware Detailed Implementation	6
3.2	Software Detailed Implementation	6
3.2.1	Docker-Compose	6
3.2.2	PostgreSQL	6
3.2.3	PgAdmin4	6
3.2.4	Budibase	6
3.2.5	Python Database Insertion/Label Generation	9
4	Test/Evaluation Experimental Procedure and Analysis of Results	10
5	Societal Impact of Project/Legal and Ethical Considerations	10

6 Contribution of Project to Society/Expected Effects	10
7 Engineering Standards, Constraints, and Security	11
8 Conclusions	11
9 Recommendations for Future Work	11
References	11
Appendices	12
A Customer Contact Information	12
B Data Sheets	12
C Additional Drawings and Diagrams	12
D Source Code	12
E Experimental and/or Simulation Test Results	12
F Software Installation Instructions	12
G User Manual	12
H Quotes, Including Ordering Information	12
I White Papers	12

[relist sections, if formatted properly can auto-generate and update it by clicking on the table above and asking it to update]

1 Introduction/Executive Summary

1.1 Introducing SoFab Inks

SoFab Inks is a chemical manufacturing startup that was spun-out from the University of Louisville, Conn Center for Renewable Energy Research with support from the US DoE. SoFab inks focuses on accelerating the commercialization of Perovskite Solar Cells through the development and manufacturing of functionalized inks that improve cell efficiency, reduce module cost, and enable scalable manufacturing. [1]

1.2 SoFab Inks Inventory Management Problem

SoFab Inks is doing important work in the field of solar cell technology, helping to drive humanity towards a cleaner, more energy abundant future. The problem, however, is that the team currently faces issues with managing inventory. These challenges prevent SoFab's talented team from working on the most important aspects of their work. These challenges include expending valuable energy on menial tasks like locating inventory, managing a growing number of shipments manually, scouring inventory entries found across several Google Sheets or handwritten labels to pinpoint important product information, and more.

TEAM REPORT (minimum of 40 pages) Each team needs to prepare a team report that will be clearly identified and attached to the individual portfolio that includes the following:

CAC 3. Communicate effectively in a variety of professional contexts

2 System Description

2.1 Needs Assessment/System Requirements

2.2 Initial System Specification

(External design document) EAC 1. Identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics and CAC 1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions)

2.3 Final Specifications

(finalized internal design document) EAC 1. Identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics and CAC 1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions and CAC 6. Apply computer science theory and software development fundamentals to produce computing-based solutions)

2.4 System Diagrams

Detail all interfaces between the environment and the components EAC 2. Apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors)

2.5 Hardware Overview Diagram

CAC 2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline)

2.6 Software Overview Diagram

CAC 2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline)

2.7 Economical, Technical, and Time Constraints

3 Detailed Implementation

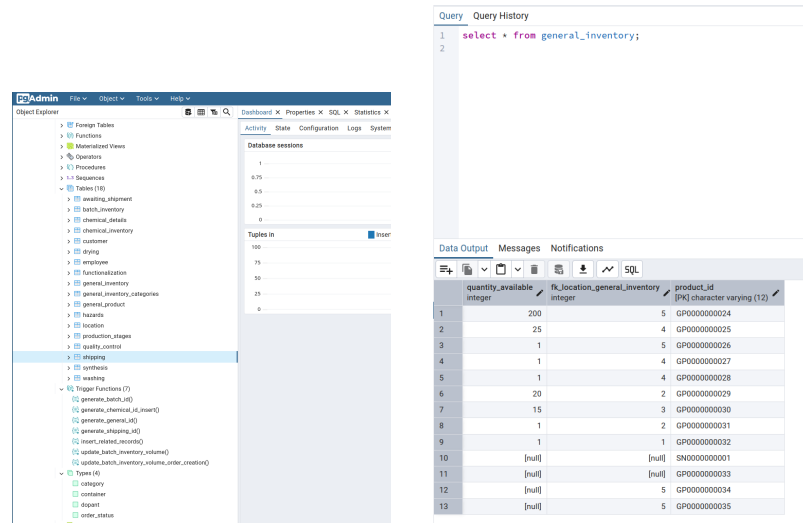
3.1 Hardware Detailed Implementation

3.2 Software Detailed Implementation

3.2.1 Docker-Compose

3.2.2 PostgreSQL

3.2.3 PgAdmin4



(a) Viewing existing tables in PgAdmin4 interface. (b) Running PostgreSQL queries inside of PgAdmin4.

Figure 1

3.2.4 Budibase

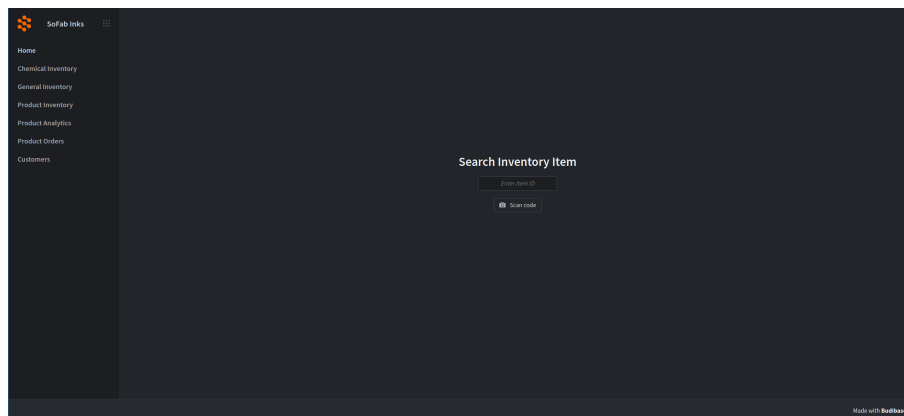


Figure 2: Budibase home page. Global search that searches all types of inventory via QR code or direct ID entry.

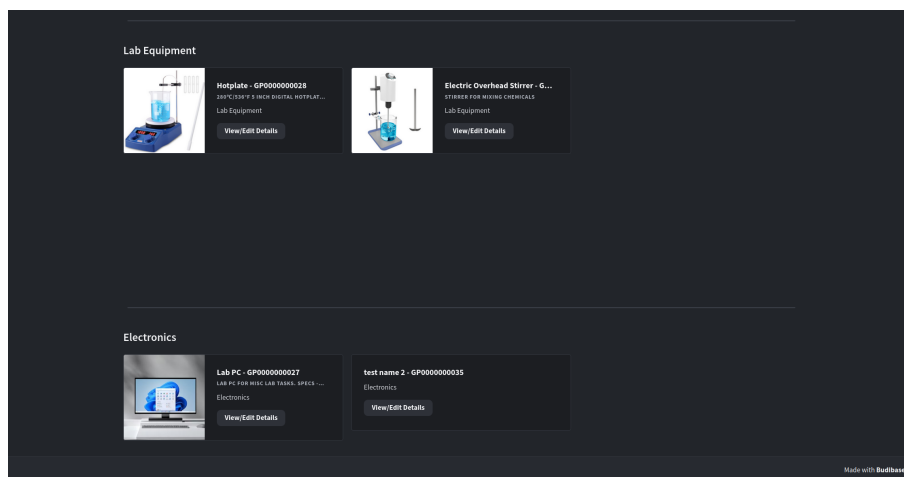


Figure 3: Budibase general inventory page. Keep track of general lab items.

3.2.5 Python Database Insertion/Label Generation

```
1 class App(Tk):
2     def __init__(self, controller):
3         # initializing the Tk class instance
4         super().__init__()
5
6         self.title("SoFab_Inventory_Managment_System")
7
8         self.style = Style(theme="darkly")
9
10        self.controller = controller
11        self.controller.set_view(self)
12
13        self.notebook = ttk.Notebook()
14        self.notebook.pack(fill="both", expand=True)
15        self.notebook.bind("<<NotebookTabChanged>>", self.
            on_tab_selection)
16
17        # filling the notebook (top tabs) with frames
18
19        item_type_tables = self.controller.
            get_item_type_tables()
```

```
1 class HazardPrecautionFrame(tk.Frame):
2     def __init__(self, parent, controller, warning_dict,
3         images=False):
4         super().__init__(parent)
```

```
1 def generate_checkboxes(self, images=False):
2     self.checkboxes_frame = tk.Frame(self)
3     self.checkboxes_frame.grid(row=0, column=0)
4
5     for item in self.warning_items:
6         if images:
7             image = Image.open(item[1])
8             resized_image = image.resize((100, 100), Image.
                LANCZOS)
9             image = ImageTk.PhotoImage(resized_image)
10            item = item[0]
11        else:
12            image = None
13        var = tk.BooleanVar()
14        checkbox = ttk.Checkbutton(
15            self.checkboxes_frame,
16            text=item,
17            image=image,
18            compound="left",
19            variable=var,
```

```

20         command=lambda var=var: self.parent.
           update_text_box(),
21     )
22     checkbox.image = image
23
24     checkbox.pack(anchor="w", fill="x")

```

EAC 1. Identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics and EAC 2. Apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors, and CAC 6. Apply computer science theory and software development fundamentals to produce computing-based solutions)

4 Test/Evaluation Experimental Procedure and Analysis of Results

EAC 6. Develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions)

5 Societal Impact of Project/Legal and Ethical Considerations

include legal and ethical considerations

CAC 4. Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles)

6 Contribution of Project to Society/Expected Effects

CAC 4. Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles)

7 Engineering Standards, Constraints, and Security

EAC 1. Apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors)

8 Conclusions

9 Recommendations for Future Work

References

- [1] SofaBinks. (n.d.). About SofaBinks. Retrieved November 29, 2024, from <https://www.sofabinks.com/about>

Appendices

A Customer Contact Information

B Data Sheets

C Additional Drawings and Diagrams

D Source Code

E Experimental and/or Simulation Test Results

F Software Installation Instructions

G User Manual

H Quotes, Including Ordering Information

I White Papers