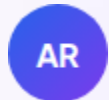


AI-Based Lost & Found System

Revolutionizing campus recovery with computer vision and intelligent matching.



Anushka Rathour

12200884 | B. Tech. CS & AI



Lost & Found

The Campus Problem

"Have you seen my keys?"

In a bustling university environment, traditional lost and found methods are broken. They rely on:

01

Physical Notice Boards

Limited reach, cluttered, and often ignored.

02

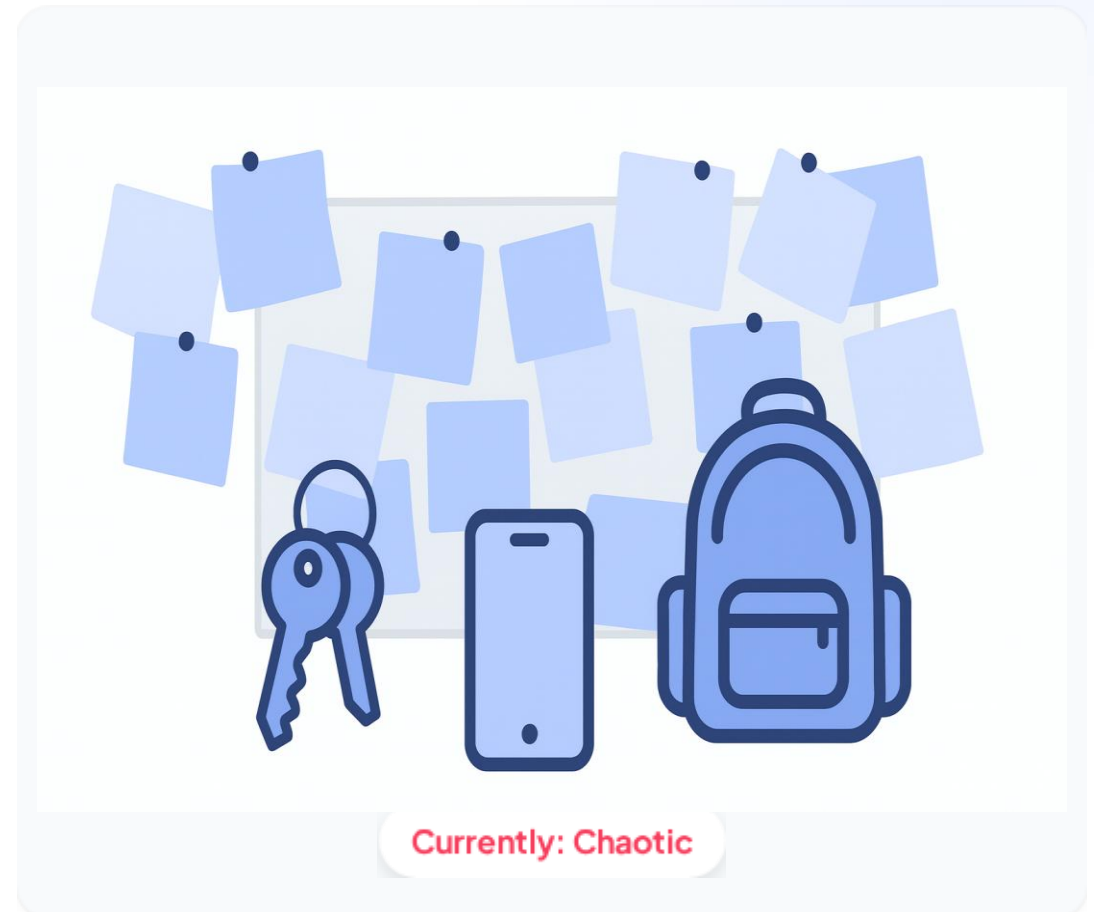
Unverified WhatsApp Groups

Spam-filled and lacks privacy or security.

03

Manual Verification

No way to prove ownership effectively.



System Objectives



Centralization

Replacing scattered manual logs with a single, unified cloud database accessible 24/7.



Visual AI

Integrating **CLIP ViT-L/14** to allow users to search by image similarity, not just text.



Security

Ensuring data integrity through student ID verification and restricted admin controls.

Technology Stack

Backend Engine

Python FastAPI

High-speed async API handling.

Sentence Transformers

Running the CLIP Model logic.

SQLite

Reliable, serverless storage.

Frontend & UI

HTML5 / CSS3

Semantic structure & styling.

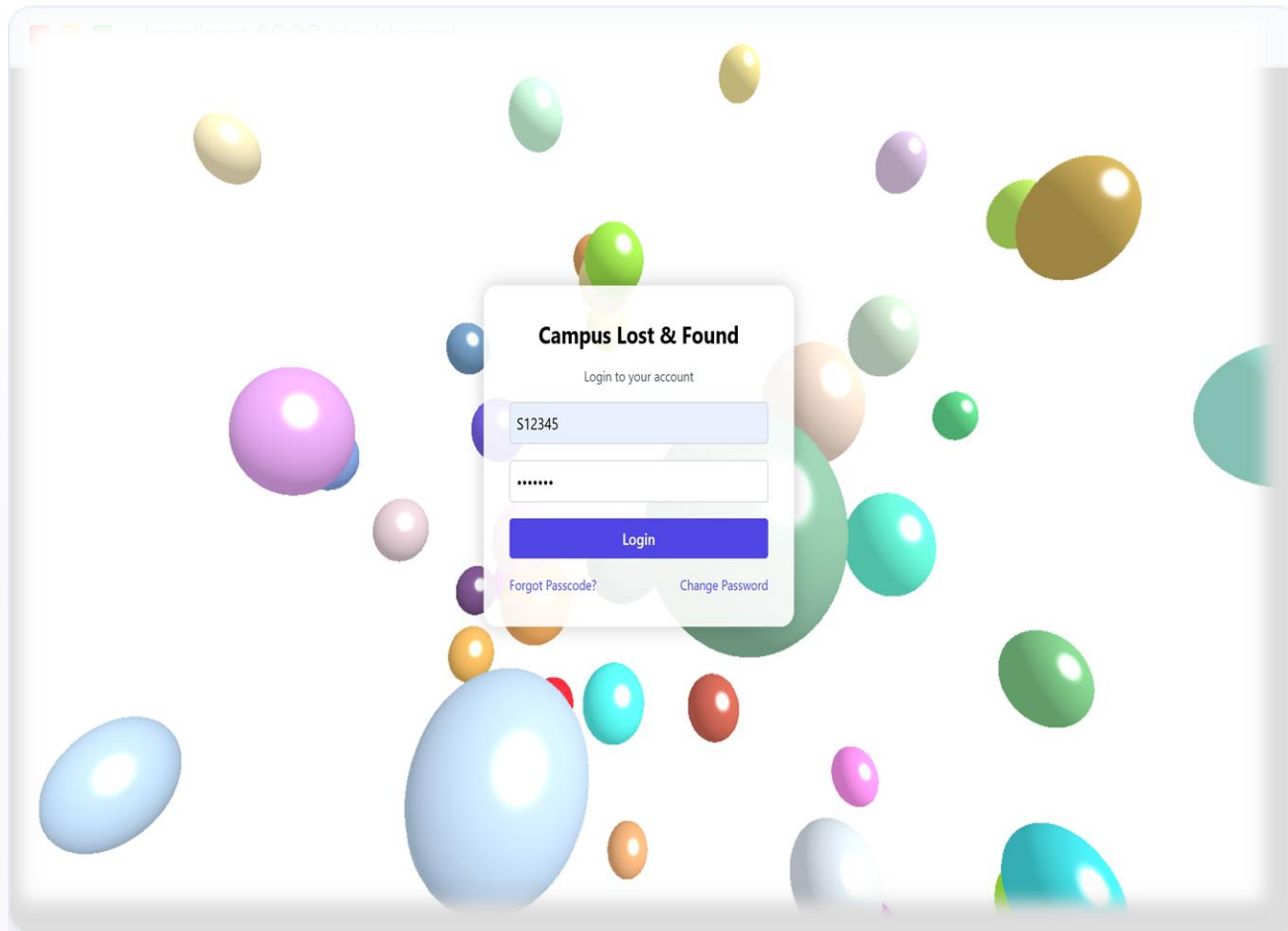
Vanilla JavaScript

Dynamic interactions & AJAX.

Jinja2

Server-side template rendering.

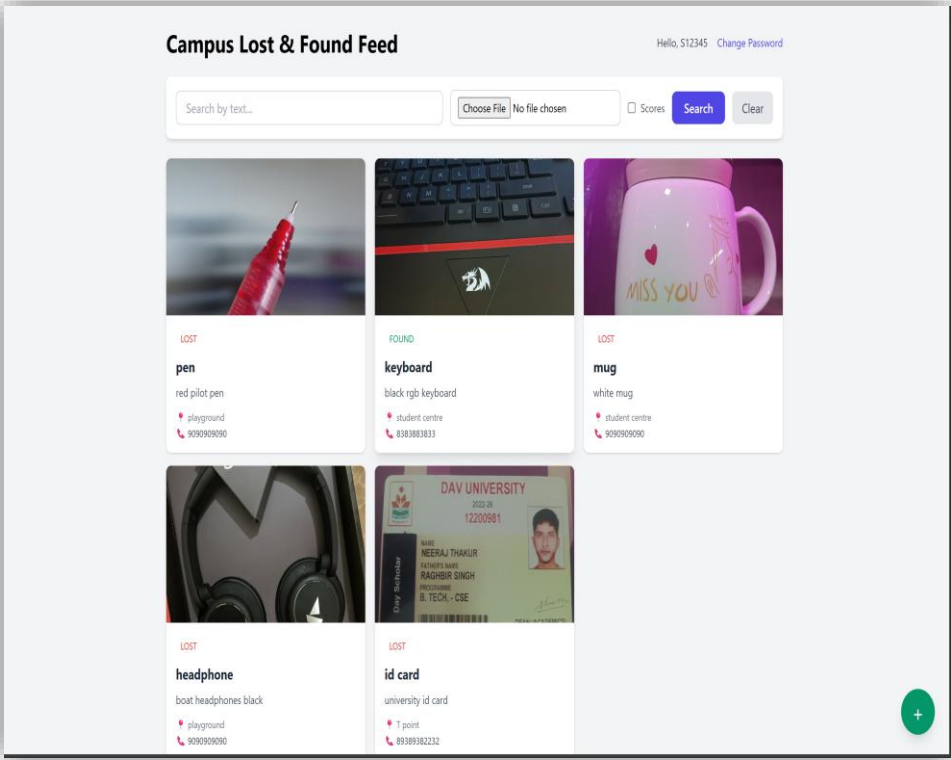
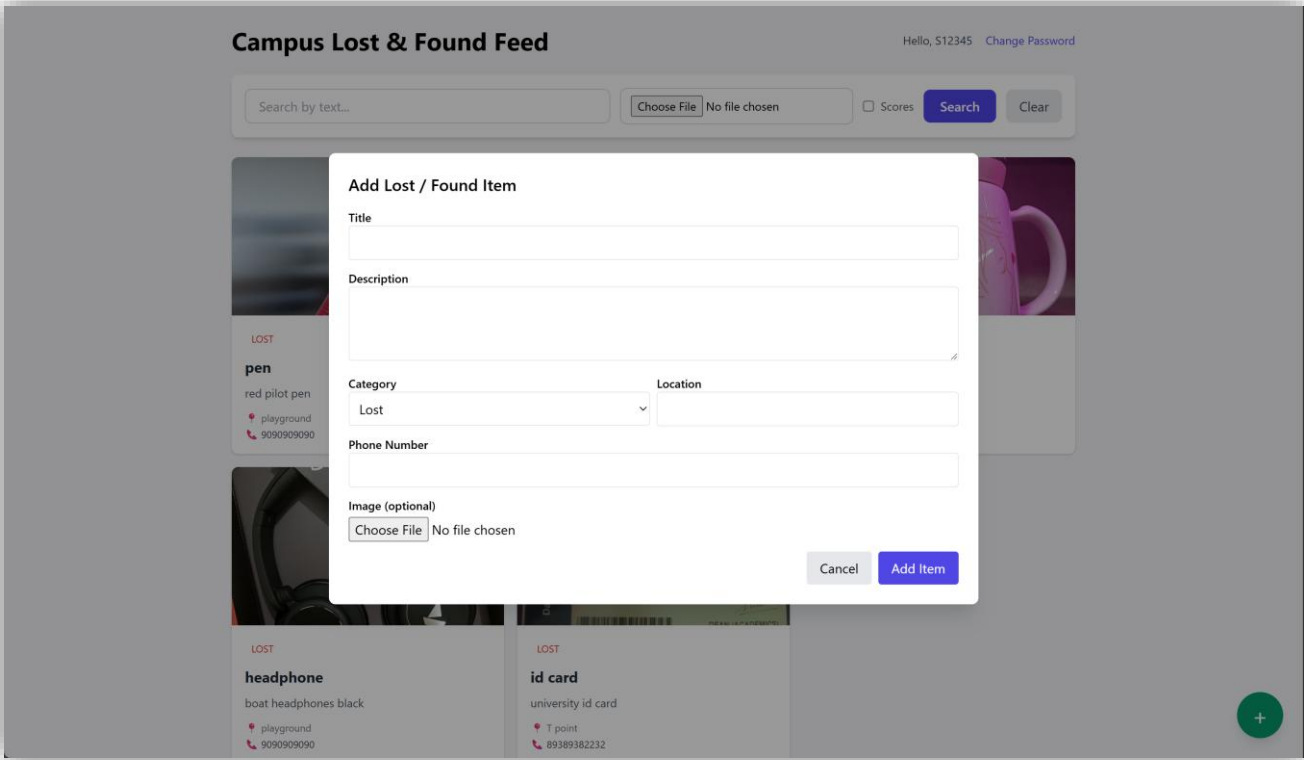
System Implementation



```
inal Help ← → lost and found

main.py M X
54
55 # Serve login page at root
56 @app.get("/", response_class=HTMLResponse)
57 async def root(request: Request):
58     return templates.TemplateResponse("login.html", {"request": request})
59
60 # Handle login and redirect based on role
61 @app.post("/login", response_class=HTMLResponse)
62 async def login(
63     request: Request,
64     student_id: str = Form(...),
65     passcode: str = Form(...),
66     conn: sqlite3.Connection = Depends(get_db),
67 ):
68     role = authenticate_user(student_id, passcode)
69     if not role:
70         return templates.TemplateResponse(
71             "login.html",
72             {"request": request, "error": "Invalid credentials"},
73         )
74     if role == "admin":
75         return RedirectResponse(url="/admin-dashboard", status_code=303)
76     # Student view: load items for index.html
77     items = conn.execute(
78         "SELECT id, title, description, category, location, phone, image_path FROM items ORDER BY id DESC"
79     ).fetchall()
80     return templates.TemplateResponse(
81         "index.html",
82         {"request": request, "student_id": student_id, "items": items},
83     )
84
85 # Student report page (optional direct access)
86 @app.get("/report", response_class=HTMLResponse)
87 async def report_page(request: Request, conn: sqlite3.Connection = Depends(get_db)):
88     items = conn.execute(
89         "SELECT id, title, description, category, location, phone, image_path FROM items ORDER BY id DESC"
90     ).fetchall()
91     return templates.TemplateResponse(
92         "index.html",
93         {"request": request, "student_id": "student", "items": items},
94     )
95
```

System Implementation



The Brain: CLIP ViT-L/14

Zero-Shot Learning

We utilize OpenAI's **CLIP (Contrastive Language-Image Pre-training)**. It maps images and text to the same vector space.

Why it matters: If you lose a "Blue Hydroflask" but search for "Water Bottle", the AI understands they are visually related, even without exact keywords.



How It Works

01



Upload

User uploads photo & details.

02



Process

FastAPI validates input.

03



Vectorize

CLIP creates embeddings.

04



Match

Cosine similarity search.

Key Distinctions

Real-World AI

Moving beyond theory. This project bridges the gap between academic Deep Learning concepts and practical, everyday utility tools.

Optimized Architecture

By using **Transfer Learning** (pre-trained weights), we achieve high accuracy without the need for massive computational power or expensive GPUs.

Future Scope



Mobile App

Developing a React Native application for camera-first reporting.



Auto Alerts

SMS/Email notifications when a potential match is found.



Multi-Campus

Scaling the database to PostgreSQL for larger deployments.

Q&A

Thank you for your
attention.



**Anushka
Rathour**

| B Tech CS & AI
12200884