

UniFind - Lost and Found Management System

UniFind is a dedicated web-based Lost and Found Management System designed specifically for the community of **Northern University of Business and Technology Khulna**. It provides a centralized, efficient, and user-friendly platform for students, faculty, and staff to report and recover lost or found items on campus, fostering a more secure and supportive university environment.

Table of Contents

1. [Problem Statement](#)
2. [Key Features](#)
 - [User Features](#)
 - [Admin Features](#)
3. [Technology Stack](#)
4. [System Architecture](#)
5. [Database Schema](#)
6. [Installation and Setup](#)
7. [Usage](#)
8. [Project Structure](#)

Problem Statement

Losing personal belongings on a large university campus is a common and stressful experience. Without a centralized system, the process of recovering lost items is often disorganized, relying on chance or scattered physical lost-and-found boxes. This leads to a low recovery rate and significant inconvenience for the campus community. UniFind aims to solve this problem by providing a single, reliable online platform to streamline the entire process of reporting, searching for, and returning lost and found items.

Key Features

UniFind is divided into two main components: a public-facing user portal and a comprehensive admin panel.

User Features

- **User Authentication:** Secure user registration and login system (signup.php, login.php).
- **Profile Management:** Users can view and update their personal information and change their password (profile.php).

- **Report Lost Items:** A detailed form allows users to report items they have lost, including title, description, category, location, date of loss, and an optional image upload (report_lost.php).
- **Report Found Items:** A similar form for users to report items they have found, helping to reunite them with their owners (report_found.php).
- **Advanced Search:** A powerful search page with filters for item type (lost/found), category, department, date range, and keywords to easily browse all reported items (search.php).
- **Intelligent Matching System:** When a user reports a lost item, the system automatically suggests potential matches from the found items database based on category, date, and location similarity (report_lost.php, lost_item.php).
- **My Reports Dashboard:** A personal dashboard where users can view the status of all items they have reported (lost, found, and matched) and perform actions like editing or deleting their reports (my_reports.php).
- **Secure Contact:** Logged-in users can view contact information for item reporters to arrange a return. A messaging system allows users to communicate securely about an item (contact_finder.php).

Admin Features

- **Admin Dashboard:** A central dashboard providing a statistical overview of the system, including total users, lost items, found items, and matches (admin/dashboard.php).
- **User Management:** Admins have full CRUD (Create, Read, Update, Delete) capabilities over all user accounts. They can view user details, edit profiles, and grant or revoke admin privileges (admin/manage_users.php).
- **Item Management:** Complete control over all lost and found item reports. Admins can edit details, update the status of an item (e.g., from 'active' to 'returned'), and delete reports (admin/manage_items.php).
- **Match Management:** Admins can review, approve, or reject matches initiated by users, ensuring the integrity of the return process (admin/manage_matches.php, admin/review_match.php).
- **Category & Department Management:** Admins can manage the lists of item categories and university departments available in forms and filters (admin/manage_categories.php, admin/manage_departments.php).

Technology Stack

- **Backend:** PHP
- **Database:** MySQL
- **Frontend:** HTML, CSS, JavaScript, jQuery

- **Framework/Libraries:** **Bootstrap 5** for responsive UI components.

System Architecture

UniFind is built on a classic client-server model with a modular PHP backend.

- **Core Logic (/includes):**
 - `db.php`: Manages the MySQL database connection.
 - `auth.php`: Handles all user authentication logic, including registration, login, sessions, and role-based access control.
 - `functions.php`: A central file containing reusable functions for core application logic like image uploads, data fetching, searching, and automatic matching.
- **User Interface:** The root directory contains the user-facing pages. Each page includes a common header and footer (`includes/header.php`, `includes/footer.php`) for a consistent layout.
- **Admin Panel (/admin):** A separate, secure directory for administrative functions. Access is restricted to users with an 'admin' role via checks in `admin/includes/admin_header.php`.

Database Schema

The MySQL database (`unifind_db.sql`) consists of several related tables to store all application data:

- `users`: Stores user account information, including credentials and role (user/admin).
- `departments`: A list of university departments.
- `categories`: A list of item categories (e.g., Electronics, Books, ID Cards).
- `lost_items`: Contains all details for items reported as lost.
- `found_items`: Contains all details for items reported as found.
- `matches`: Links a `lost_items` record with a `found_items` record, managed by admins.
- `contact_messages`: Stores messages sent between users regarding an item.

Installation and Setup

To set up and run this project on a local server, follow these steps:

1. **Prerequisites:**
 - A local server environment like [XAMPP](#) or WAMP.
 - A web browser.
 - A database management tool like phpMyAdmin (included with XAMPP).
2. **Clone the Repository:**
`git clone https://github.com/your-username/UniFind.git`

Move the cloned project folder into your server's web directory (e.g., C:/xampp/htdocs/).

3. Database Setup:

- Start the Apache and MySQL services from your XAMPP control panel.
- Open phpMyAdmin by navigating to <http://localhost/phpmyadmin>.
- Create a new database and name it `unifind_db`.
- Select the new database and go to the "Import" tab.
- Click "Choose File" and select the `unifind_db.sql` file from the project directory.
- Click "Go" to import the database schema and default data.

4. Database Configuration:

- Open the file `includes/db.php`.
- Verify that the database credentials (`DB_HOST`, `DB_USER`, `DB_PASS`, `DB_NAME`) match your local server setup. By default, they are set for a standard XAMPP installation.

```
define('DB_HOST', 'localhost');  
define('DB_USER', 'root');  
define('DB_PASS', ''); // Default is empty for XAMPP  
define('DB_NAME', 'unifind_db');
```

5. Run the Application:

- Open your web browser and navigate to <http://localhost/UniFind/> (or the name you gave the project folder).

Usage

- **General User:** You can create a new account via the "Sign Up" page or browse publicly listed items.
- **Admin Access:** The database import includes a default admin account.
 - **Username:** admin
 - **Password:** admin123
 - Log in with these credentials to access the Admin Dashboard. **It is highly recommended to change the default password after your first login.**

Project Structure

```
UniFind/  
|  
| — admin/          # Admin Panel Files  
| — assets/
```

```
|
| |— includes/      # Admin-specific header/footer
| |— dashboard.php
| |— manage_users.php # ... and other management pages
|
|— assets/          # Public assets
| |— css/
| |   |— style.css
| |— js/
| |   |— script.js
|
|— includes/        # Core PHP logic
| |— auth.php       # Authentication and session management
| |— db.php         # Database connection
| |— functions.php  # Main application functions
|
|— uploads/         # Directory for user-uploaded images
|
|— index.php        # Homepage
|— login.php        # Login page
|— signup.php       # Registration page
|— search.php       # Search page
|— report_lost.php  # Form to report a lost item
|— report_found.php # Form to report a found item
|— lost_item.php    # Detail page for a single lost item
|— found_item.php   # Detail page for a single found item
|— profile.php      # User profile page
|— my_reports.php   # User's personal reports dashboard
|— unifind_db.sql   # Database schema and initial data
```