**Project Name:** DJ CRM

**Purpose:** This project is a web-based application designed to manage and track leads, agents, and categories. It provides a comprehensive system for lead tracking, agent management, and category classification.

**Technologies/Tools Used:** The project is built using Python’s Django framework. It uses Django’s built-in views and URL routing. It also uses Django’s authentication views for login, logout, and password reset functionalities.

**Main Features:**

1. **Lead Management:** The application allows for the creation, retrieval, update, and deletion of leads. Leads can be assigned to agents and categorized. Follow-ups can also be created, updated, and deleted for each lead.
2. **Agent Management:** Agents can be created, viewed in detail, updated, and deleted.
3. **Category Management:** Categories can be created, viewed in detail, updated, and deleted.
4. **Authentication:** The application includes signup, login, logout functionalities along with password reset options.
5. **Dashboard:** A dashboard view is available for authenticated users.

**Project Name:** Restaurant Management System (RMS)

**Purpose:** This project is a Python-based application designed to manage and track users, stocks, products, and financial reports in a restaurant setting. It provides a comprehensive system for user tracking, stock management, product management, and report generation.

**Technologies/Tools Used:** The project is built using Python. It uses Python’s built-in functionalities and object-oriented programming concepts. It also uses SQL for data storage and retrieval. The user interface is built using the Tkinter library in Python. **Main Features:**

1. **User Management:** The application allows for the retrieval of users by their username. It also provides a login system for users.
2. **Stock Management:** The application allows for the creation, retrieval, update, and deletion of stocks. Stocks can be retrieved by their name, new stocks can be inserted, existing stocks can be updated, and stocks can be deleted by their name.
3. **Product Management:** The application allows for the creation, retrieval, update, and deletion of products. Products can be retrieved by their category or id, new products can be inserted, existing products can be updated by their id, and products can be deleted by their id.
4. **Report Generation:** The application allows for the generation of various reports based on sold products, sold tables, sold users, and also provides daily, weekly, and monthly graph reports.
5. **Dashboard:** The application provides different dashboards for admin and user roles.
6. **Error Handling:** The application includes error handling functionalities to show errors and information to the users.
7. **Login System:** The application includes a login system with clear functionality.

Project Name: Kaya Camp   
  
Purpose: This project is a web-based application designed to manage and track campgrounds, users, and reviews. It provides a comprehensive system for campground tracking, user management, and review classification.

Technologies/Tools Used: The project is built using Node.js with Express.js framework. It uses Express.js’s built-in views and URL routing. It also uses Passport.js for authentication functionalities. Main Features:

1. Campground Management: The application allows for the creation, retrieval, update, and deletion of campgrounds. Campgrounds can be assigned to users and categorized. Reviews can also be created, updated, and deleted for each campground.
2. User Management: Users can be created, viewed in detail, updated, and deleted. Authentication functionalities such as login and logout are also provided.
3. Review Management: Reviews can be created, viewed in detail, updated, and deleted.
4. Authentication: The application includes signup, login, logout functionalities.

Dashboard: A dashboard view is available for authenticated users to manage their campgrounds and reviews.  
  
**Project Name:** SmartEdu

**Purpose:** SmartEdu is a web-based learning platform designed to provide a seamless learning experience for both students and teachers. It offers comprehensive functionalities for course management, user authentication, and teacher profiles.

**Technologies/Tools Used:** The project is built using Python’s Django framework. It uses Django’s URL routing for navigation and Django’s built-in User model for authentication. It also uses Django’s generic class-based views for displaying pages.

**Main Features:**

1. **Course Management:** The application allows for the creation, retrieval, update, and deletion of courses. Courses can be searched and filtered by categories or tags. Students can enroll in or unenroll from courses.
2. **User Management:** Users can register, login, logout, and view their dashboard. The dashboard displays the courses a user is enrolled in.
3. **Teacher Profiles:** Teacher profiles can be viewed in detail, displaying the courses they offer.
4. **Authentication:** The application includes signup, login, logout functionalities.
5. **Static Pages:** The application includes static pages like the home page, about page, and contact page.

**Project Name:** Personal Portfolio Website

**Purpose:** This project is a web-based personal portfolio designed to showcase skills, education, and professional experiences. It provides a comprehensive platform for potential employers or clients to understand the user’s capabilities and achievements.

**Technologies/Tools Used:** The project is built using HTML, CSS, and JavaScript. It uses Bootstrap for responsive design and various JavaScript libraries such as AOS, Swiper, Isotope Layout for dynamic features. Django is used for backend development.

**Main Features:**

1. **Home Section:** The home section displays the user’s name and a brief introduction. It also includes links to the user’s GitHub, LinkedIn, and Instagram profiles.
2. **About Section:** This section provides more detailed information about the user, including their skills and competencies. It also includes a button to download the user’s resume.
3. **Education Section:** This section lists the user’s educational qualifications along with the institutions they attended.
4. **Professional Section:** This section showcases the user’s professional experiences and achievements.
5. **Portfolio Section:** This section displays the projects that the user has worked on. It includes filters to view projects based on different categories.
6. **Contact Section:** This section includes a form that visitors can fill out to get in touch with the user.
7. **Footer:** The footer includes copyright information and credits for the website design.

The website is designed to be responsive and works seamlessly across various device sizes. It also includes a back-to-top button for easy navigation. The website uses AJAX for form submissions and includes preloader functionality for better user experience.