**ADREVU High Level Design Document**

Contents:

1. System Overview

2. System architecture

3. Page-wise High Level Design

3.1 Login page

3.2 Sign-up page

3.3 Reset password page

3.4 Home page

3.5 Post Ad page

3.6 Edit Ad page

3.7 Customer dashboard

3.8 Seller dashboard

3.9 Admin dashboard

3.10 Ad details page

3.11 Seller profile (public view)

3.12 Seller profile (admin view)

3.13 About us page

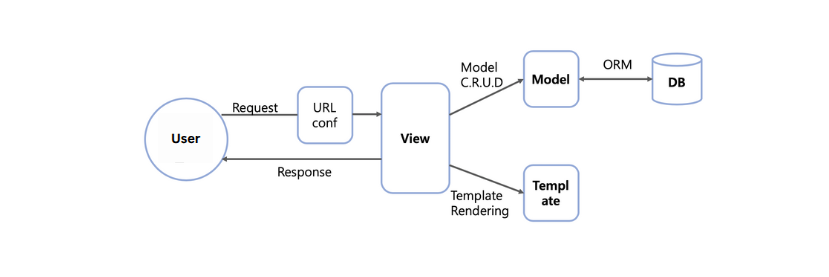
3.14 Contact us page

3.15 404 - Not Found Page

**1. System Overview:**

This document presents the high-level design for an online classified ads platform built with Python Django. The platform enables user registration, ad posting and management, admin moderation, search/filtering, and location-based listings, mirroring the structure of leading classified portals.

**2. System architecture:**



**3. Page-wise high level design:**

**3.1 Login page:**

Flow:

User enters credentials

⇩

Django Auth checks

⇩

Redirect based on role

Web Service:

URL: /login/

Method: POST

Request Body:

json

{

"email": "hello@adrevu.com",

"password": "encrypted-password"

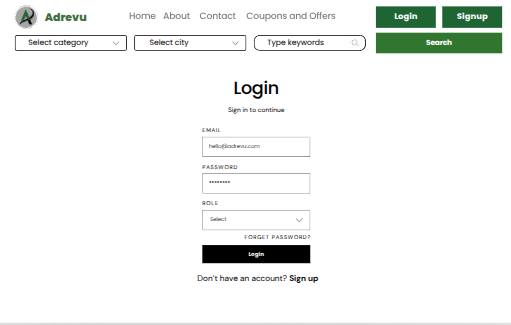
}

Redirects:

Admin → /admin/dashboard/

User → /dashboard/

Page layout:



**3.2 Sign-up page:**

Flow:

User enters registration details

⇩

Django validates and creates account

⇩

Redirect to login or dashboard

Web Service:

URL: /signup/

Method: POST

Request Body:

json

{

"full\_name": "Mahesh",

"email": "hello@adrevu.com",

"password": "password",

"confirm\_password": "password"

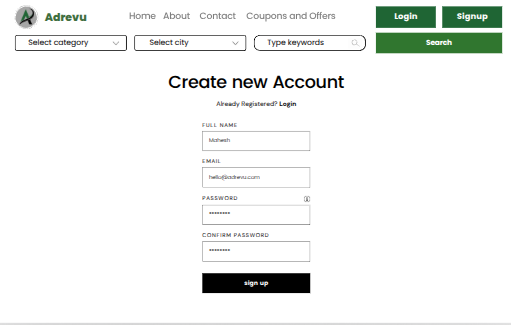
}

Redirects:

On success → /login/

On failure → Stay on sign up page with error messages

page-layout:



**3.3 Reset password page:**

Flow:

User enters registered email

⇩

Django sends password reset link to email

Web Service:

URL: /password\_reset/

Method: POST

Request Body:

json

{

"email": "hello@adrevu.com"

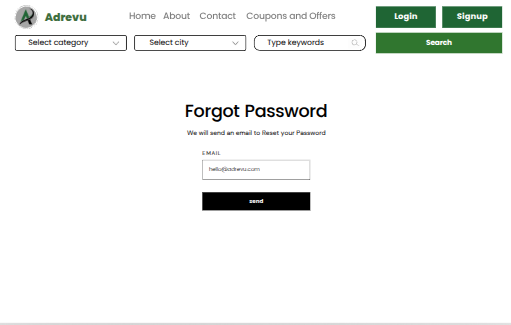
}

Redirects:

On success → Show confirmation message (e.g., "We will send an email to Reset your Password")

On failure → Stay on page with error message

page-layout:



**3.4 Home page:**

Flow:

Display featured/latest ads

⇩

User can search/filter by keyword, city, or category

⇩

View list of ads

⇩

Navigate to ad details

Web Services:

URL: /

Method: GET

Loads homepage layout and static content

URL: /ads/

Method: GET

Fetches latest ads, optionally filtered by query parameters (e.g., keyword, city, category)

URL: /categories/

Method: GET

Fetches all categories for the category dropdown

URL: /cities/

Method: GET

Fetches all cities for the city dropdown

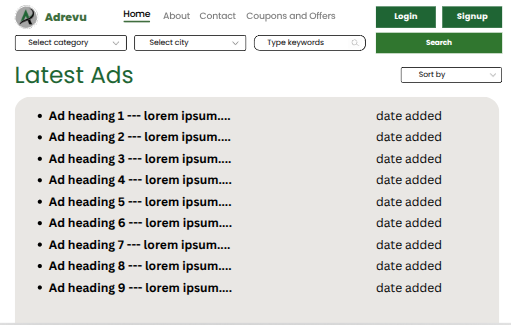
Redirects:

Clicking an ad heading → /ads/<ad\_id>/ (ad details page)

Clicking "Login" → /login/

Clicking "Signup" → /signup/

page-layout:



**3.5 Post Ad page:**

Flow:

User enters Ad details

⇩

Django stores the ad details in the database

⇩

On success redirects to the edit page

Web Service:

URL: /postad/

Method: POST

Post Body:

json

{

"title": "Wireless Bluetooth Headphones",

"description": "High-quality over-ear headphones with noise cancellation and long battery life. Compatible with all Bluetooth-enabled devices.",

"price": 2499,

"image": "https://example.com/images/headphones.jpg",

"category": "Electronics",

"city": "Hyderabad"

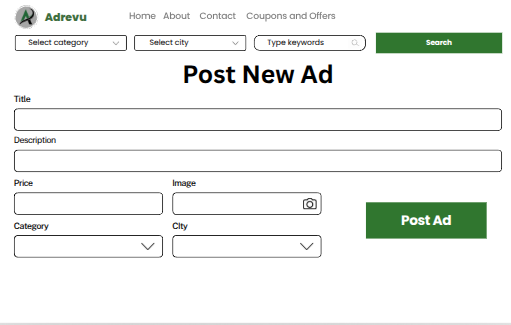
}

Redirects:

On success → /editad/

On failure → Stay on sign up page with error messages

page-layout:



**3.5 Edit Ad page:**

Flow:

Ad available update by the seller

⇩

Django verify the changes if any visibles the update button

⇩

On success redirects to the same edit page

Web Service:

URL: /editad/

Method: POST

Post Body:

json

{

"title": "Wireless Bluetooth Headphones",

"description": "High-quality over-ear headphones with noise cancellation and long battery life. Compatible with all Bluetooth-enabled devices.",

"price": 2499,

"image": "https://example.com/images/headphones.jpg",

"category": "Electronics",

"city": "Hyderabad"

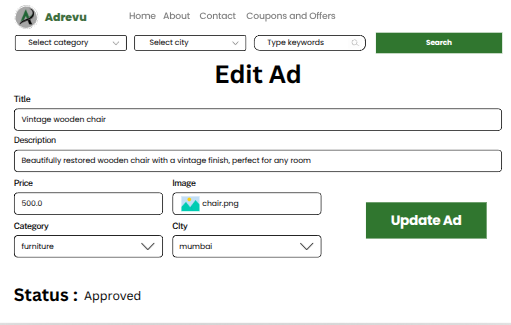
}

Redirects:

On success → /editad/

On failure → Stay on sign up page with error messages

page-layout:



**3.7 Customer dashboard**

Page Flow:

On page load, GET /mywishlists/ is called to display all wishlist items.

Clicking "Remove" on an item sends DELETE /mywishlists/{item\_id}/ and updates the list.

Web Service Endpoints:

GET /mywishlists/

Retrieves the list of items in the customer’s wishlist.

DELETE /mywishlist/{item\_id}/

Removes a specific item from the wishlist when the "Remove" button is clicked.

Redirects:

Clicking Dashboard: /mydashboard/

Clicking Notifications: /mynotifications/

Clicking My Wishlist: /mywishlists/

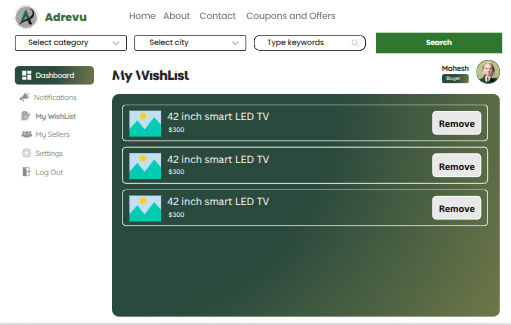
Clicking My Sellers: /mysellers/

Clicking Settings: /settings/

Clicking Log Out: Redirects to the login page

Sidebar links allow navigation to other dashboard sections via the corresponding redirects.

page-layout:



**3.8 Seller dashboard**

Page Flow:

On page load, the app calls GET /myads/ to display the user's ads.

Clicking "Edit" on any ad opens the edit form (could be a modal or a new page) and uses PUT /myads/{ad\_id}/ to save changes.

Clicking "Delete" sends a DELETE /myads/{ad\_id}/ request and removes the ad from the list.

Web Service Endpoints:

GET /myads/

Fetches the list of ads created by the logged-in user.

PUT /myads/{ad\_id}/

Updates the details of a specific ad when "Edit" is clicked.

DELETE /myads/{ad\_id}/

Deletes a specific ad when "Delete" is clicked.

Redirects:

Clicking Dashboard in the sidebar: redirects to /mydashboard/

Clicking Notifications: redirects to /mynotifications/

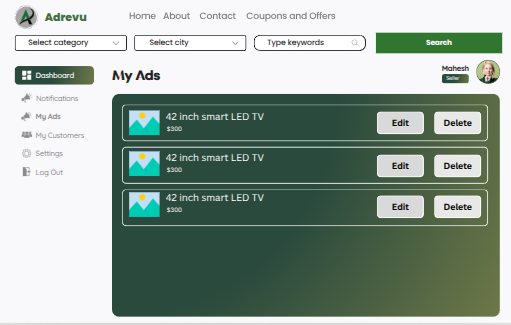
Clicking My Ads: stays on /myads/

Clicking My Customers: redirects to /mycustomers/

Clicking Settings: redirects to /settings/

Clicking Log Out: logs out the user and redirects to the login page

page-layout:



**3.9 Admin dashboard**

Page Flow:

On page load, GET /managesellers/ displays all sellers.

Entering a search term and clicking "Search Users" calls GET /managesellers/?search={query} to filter the seller list.

Web Service Endpoints:

GET /managesellers/

Fetches the list of all sellers.

GET /managesellers/?search={query}

Searches sellers by name or email.

Redirects:

Clicking Dashboard: /mydashboard/

Clicking Notifications: /mynotifications/

Clicking Manage Sellers: /managesellers/

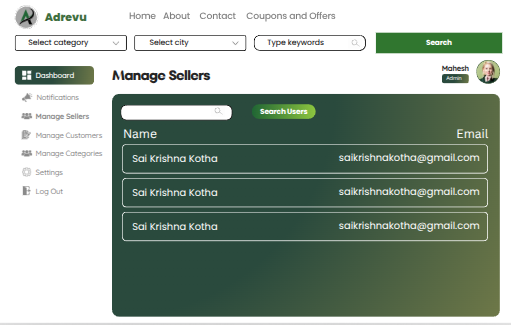
Clicking Manage Customers: /managecustomers/

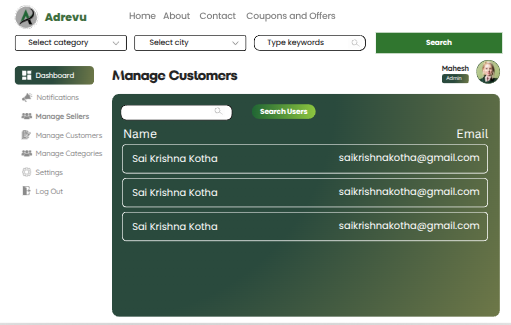
Clicking Manage Categories: /managecategories/

Clicking Settings: /settings/

Clicking Log Out: Redirects to login page

Page-layout :





**3.10 Ad details page**

Page Flow:

On page load, GET /ads/{ad\_id}/ is called to display all ad information (title, price, description, seller details, item details).

If the user navigates using the top menu or search, the page redirects to the respective section.

Web Service Endpoints:

GET /ads/{ad\_id}/

Fetches the details of a specific ad, including title, price, description, seller info, and item details.

Redirects:

Clicking Home: /

Clicking About: /about/

Clicking Contact: /contact/

Clicking Coupons and Offers: /offers/

Clicking Login: /login/

Clicking Signup: /signup/

Using the search bar and clicking Search: /search/?category={category}&city={city}&keywords={keywords}

Request Body:

json

{

"title": "Wireless Bluetooth Headphones",

"description": "High-quality over-ear headphones with noise cancellation and long battery life. Compatible with all Bluetooth-enabled devices.",

"price": 2499,

"image": "https://example.com/images/headphones.jpg",

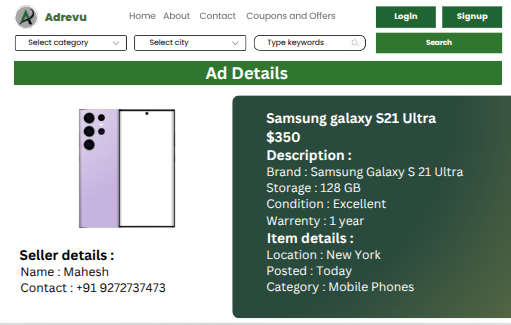
"category": "Electronics",

"city": "Hyderabad"

}

Page-layout :

\



**3.11 Seller profile (public view)**

Page Flow:

On page load, GET /sellers/{seller\_id}/ displays the seller's public details (name, description, contact, email).

Default tab "Seller Products" calls GET /sellers/{seller\_id}/products/ and displays the seller’s products.

Clicking "All reviews" tab calls GET /sellers/{seller\_id}/reviews/ and shows all reviews for the seller.

Clicking "Feedback" tab calls GET /sellers/{seller\_id}/feedback/ and shows customer feedback.

Search bar within the page allows filtering the seller’s products. Entering a query and searching updates the product list using GET /sellers/{seller\_id}/products/?search={query}

Web Service Endpoints:

GET /sellers/{seller\_id}/

Fetches public profile details of the seller, including name, description, contact number, and email.

GET /sellers/{seller\_id}/products/

Retrieves the list of products posted by the seller.

GET /sellers/{seller\_id}/reviews/

Retrieves all reviews for the seller.

GET /sellers/{seller\_id}/feedback/

Retrieves feedback for the seller.

Redirects:

Clicking Home: /

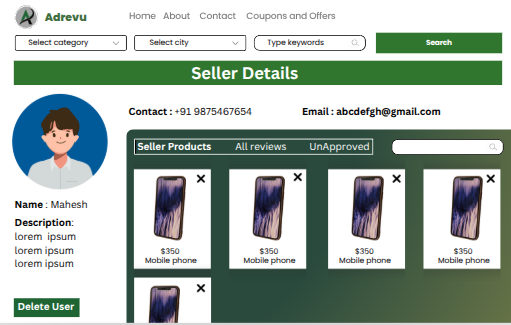
Clicking About: /about/

Clicking Contact: /contact/

Clicking Coupons and Offers: /offers/

Using the search bar and clicking Search: /search/?category={category}&city={city}&keywords={keywords}

Page-layout:



**3.12 Seller profile (Admin view)**

Page Flow:

On page load, GET /admin/sellers/{seller\_id}/ displays the seller’s profile details (name, description, contact, email)

Default tab "Seller Products" calls GET /admin/sellers/{seller\_id}/products/ to display all products posted by the seller.

Clicking "All reviews" tab calls GET /admin/sellers/{seller\_id}/reviews/ to display all reviews.

Clicking "UnApproved" tab calls GET /admin/sellers/{seller\_id}/products/?status=unapproved to display unapproved products.

Clicking "Delete User" sends DELETE /admin/sellers/{seller\_id}/ to remove the seller from the platform.

The search bar above the product list allows admins to filter the seller’s products; entering a query updates the product list using GET /admin/sellers/{seller\_id}/products/?search={query}

Web Service Endpoints:

GET /admin/sellers/{seller\_id}/

Retrieves full seller profile details for admin, including name, description, contact, and email.

GET /admin/sellers/{seller\_id}/products/

Retrieves all products listed by the seller, including approved and unapproved items.

GET /admin/sellers/{seller\_id}/reviews/

Retrieves all reviews for the seller.

GET /admin/sellers/{seller\_id}/feedback/

Retrieves all feedback for the seller.

DELETE /admin/sellers/{seller\_id}/

Deletes the seller account when the "Delete User" button is clicked.

GET /admin/sellers/{seller\_id}/products/?status=unapproved

Retrieves all unapproved products for the seller.

Redirects:

Clicking Home: /

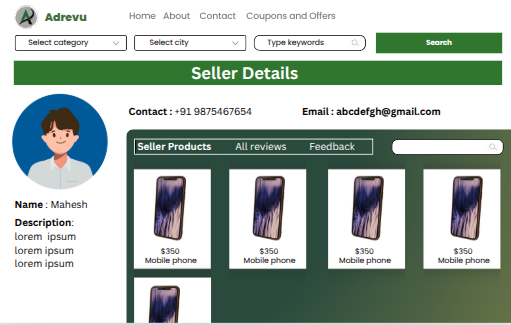
Clicking About: /about/

Clicking Contact: /contact/

Clicking Coupons and Offers: /offers/

Using the search bar and clicking Search: /search/?category={category}&city={city}&keywords={keywords}

Page-layout :

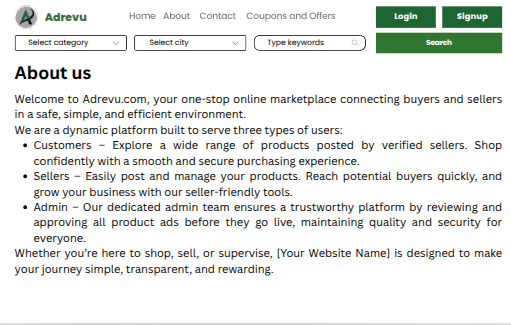


**3.13 About us page**

Flow:

User ables to view the purpose of this website

page-layout:

****

**3.14 contact us page**

Flow:

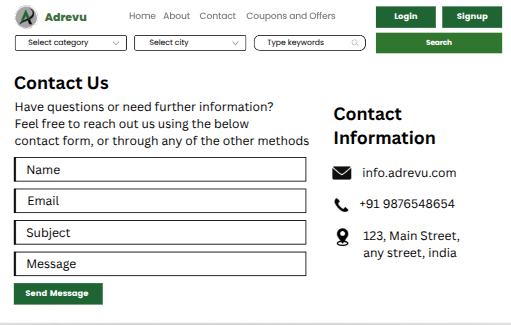
Takes user query from fields

Web Service Endpoints:

URL : /aboutpage/

Method : POST

Page-layout :

****

**3.15 404 not found page**

Web Service Endpoints:

URL : /404-not-found/

