Home Worker Finder

* Table of Contents
* Introduction
* System requirements
* Installation and Setup
* Code Structure
* User Guide
* Future Improvements
* Contributors
* License

Introduction

This document provides an overview of Home Worker Finder, a web-based application that is designed to help home owners find workers for various tasks such as cleaning, babysitting, gardener, plumber, carpenter. It includes feature like user registration, user authentication, search by service and map integration.

System Requirements

* Python 3.11+
* Virtual environment

Installation and Setup

* Clone the repository
* Create and activate virtual activate

Window(powershell)

python -m venv venv

venv\Scripts\Activate

* Install required dependencies
* pip install django django-allauth django-crispy-forms crispy-tailwind django-chapa pillow

Verify with

* pip list

You should see

* Django
* django-allauth
* django-crispy-forms
* crispy-tailwind
* django-chapa
* Pillow
* Apply database migration

python manage.py migrate

* Create superuser

python manage.py createsuperuser

* Run the development server

Python manage.py runserver

Code structure

* manage.py: Entry point for running the Django project.
* settings.py: Contains configuration settings for the project.
* urls.py: Defines URL routing for the application.
* models.py: Contains database models for workers
* views.py: Handles the logic for displaying and processing requests.
* templates/: Directory containing HTML templates for frontend views.
* static/: Directory containing CSS, JavaScript, and image files.
* admin.py: Configures how models appear in the Django admin panel.
* forms.py: Defines Django forms used in the application.

User Guide

* Admin user

1. Demand tracking by search
2. Revenue analytics
3. View both home owners and workers information

* Worker

1. Register an account and login
2. Create and manage profiles
3. Create and manage their profiles.
4. List their services and skills.

* Home owners

1. Search for registered workers
2. View workers profiles
3. Find workers by their location
4. Pay for the service paypal, venmo, debitcard

Future improvement

* Implement real-time chat
* Improve search filters by skills
* Home owners can review workers
* Owners can pay using chapa and telebirr
* Workers can pay for premium visibility
* Mobile freindly design
* Calendar integration for reminder and scheduling

Contributors

1.Ragad Abdelwahab

2.Yididya Tesfaye

3.Mahlet Habtom

4.Biruktawit Kassahun

5.Ephrata Walelign