PROJECT ABSTRACT

Daily Wage Job Finder Android App

An Android daily-wage job finder connecting workers with nearby contractor-posted work: hyperlocal listings, upfront wages, one-tap apply and application tracking — delivered built-to-order with full source, API design, report, PPT and viva kit.

01. Title

Daily Wage Job Finder Android App.

02. Introduction / Background

Millions of daily-wage workers find work through labor chowks and word of mouth — showing up at dawn hoping a contractor needs hands that day, with no record of wages promised versus paid. General job portals ignore this segment: they assume resumes, email IDs and English fluency. This project builds a job finder designed for how daily-wage work actually happens: hyperlocal listings within a few km, daily wage stated upfront on every card, timing and distance visible before tapping, one-tap apply with no resume, and application tracking from applied to accepted. Contractor verification badges and ratings add trust to an informal market, and Hindi/Marathi cues with large touch targets serve first-time smartphone users.

03. Problem Statement

Daily-wage work discovery is informal, uncertain and intermediated: workers waste mornings seeking work and have no recourse on wage disputes. A hyperlocal mobile marketplace with upfront wages and application tracking formalizes discovery without demanding resumes or English.

04. Objectives

- List nearby work with wage, timing and distance on every card, sorted by distance.
- Filter by six work categories via one-tap chips.
- Show job details: requirements, verified contractor, rating, wage, timing, distance.
- Enable one-tap apply with confirmation and next-steps messaging.
- Track applications (applied / accepted / pending) with plain-language status.

05. Proposed Methodology

The app follows MVVM with a listings data layer. The feed renders cards from the dataset sorted by distance; filter chips re-query by category. Detail screens assemble requirements, contractor trust signals and logistics. Apply writes an application record with pending status; the confirmation screen sets expectations (contractor response within hours via SMS/app notification). The demo runs on six realistic listings; the REST API design covers listings, applications and contractor verification for production.

06. System Architecture / Working

Screens: feed (chips + cards), job detail (requirements + contractor + apply bar), confirmation (next steps), applications (status list). Data: listings with wage/timing/distance/category/contractor; applications with status. Location services power distance sorting; SMS fallback covers feature-phone status updates.

07. Hardware / Software Requirements

- Android 8.0 (API 26)+ device; GPS for distance sorting.
- Android Studio + Kotlin for building; demo needs no server.
- Production: REST backend per API design, SMS gateway for status fallback.

08. Expected Outcomes

- A worker-completable journey (browse → filter → apply → track) demonstrable without training.
- Application lifecycle visibly moving applied → accepted.
- Complete Kotlin source, API design, report, PPT and viva Q&A.;

09. Applications

- Urban daily-wage labor markets (construction, loading, painting).
- Rural seasonal farm-labour matching.
- NGO livelihood programs digitizing job linkage.

10. Future Scope

- Contractor KYC verification pipeline.
- In-app worker–contractor chat.
- Wage escrow and digital payment on completion.