

Project Track 1 - Report

Team 047 - Join_The_Party

Project Title: HandyIllinois - Connecting Homeowners with Reliable Handyperson Services

1. Overview and Objectives

HandyIllinois was conceptualized to provide a reliable and user-friendly platform connecting homeowners in Illinois with professional handyperson services. It simplifies the process of finding, booking, and managing service requests while offering agencies and service providers advanced tools for performance tracking.

The final implementation includes:

1. Full CRUD functionality for users, bookings, and reviews.
2. Keyword search for handyperson skills, ratings, and locations.
3. Advanced database features like stored procedures, triggers, and transactions.

Backend and Database are entirely hosted on **Google Cloud Platform (GCP)**, the project demonstrates scalability and real-world relevance.

2. Changes from Stage 1 Proposal

Initial Proposal

The initial design featured materialized views for real-time reporting, interactive dashboards, and datasets transformed from NYC to Illinois.

Final Adjustments

Advanced Queries: Focused on stored procedures and triggers, replacing materialized views.

Scope Added: Added visualization dashboards to prioritize core functionality.

Dataset Augmentation: Successfully localized NYC and Urban Company datasets for Illinois.

3. Achievements and Usefulness

Achievements

- **CRUD Operations:** Enabled seamless user registration, profile updates, bookings, cancellations, and reviews.
- **Keyword Search:** Fully functional search for handyperson services by skill, location, or price range.
- **Analytics Tools:** Implemented procedures for tracking earnings, top-performing handypersons, and customer satisfaction.

Shortcomings

- Lack of visualization tools due to time constraints.
- Full-text search optimization was limited.

4. Schema and Data Source Changes

Schema Updates -

- **BaseCharge:** Added to the **Handyperson** table for dynamic pricing.
- **Triggers:** Automated **Handyperson** rating updates post-review submission.

ER Diagram Adjustments -

- Strengthened relationships:
 - **Agency** to **Manager** (1:N).
 - **ServiceRequest** to **Review** (1:1).

Rationale: Improved scalability and better alignment with application requirements.

5. Functionalities Added/Removed

Added

1. **Triggers:** Automated data integrity (e.g., rating updates).
2. **Transactions:** Ensured service request consistency.
3. **Advanced Stored Procedures:** Optimized manager and handyperson analytics.

Removed

1. Real-time dashboards due to scope adjustments.
2. Materialized views replaced with dynamic queries.
3. Profile image fetch ran into difficulties.

Reasons -

Focused resources on core functionalities that enhance user experience.

6. Advanced Database Programs and Complementarity

Stored Procedures

1. **Top Handypersons by Manager:** Lists top performers based on ratings and reviews.
2. **Manager's Earnings Analysis:** Aggregates earnings by service type and manager.

Triggers

1. **Rating Update:** Automatically calculates average ratings for handypersons upon new reviews.
2. **Review Logging:** Tracks review submissions for audit purposes.

Transactions

Handled service booking requests, ensuring isolation levels to maintain data consistency during simultaneous operations.

Constraints

Primary key and foreign key constraints have been implemented for all tables. Unique constraint has been implemented for the email attribute of the Customer table because email is used for authentication purposes.

Relevance: These features complement the application by automating critical tasks and optimizing query performance.

7 . Technical Challenges Encountered and Solutions

1. **Data Adaptation:** Transforming NYC datasets for Illinois with Python scripting.
2. **Trigger Debugging:** Iteratively adjusted **AFTER INSERT** logic for ratings.
3. **Concurrency Issues:** Optimized GCP connection pooling for high-demand operations.
4. **Indexing Strategy:** Balanced between query complexity and indexing overhead.

8. Final Comparison

Changes Compared to Stage 1

- Focused on CRUD operations and analytics over real-time features.
- Simplified dataset usage while ensuring functionality.

Future Improvements

1. **Enhanced Search:** Full-text and semantic capabilities.
2. **Visualization Dashboards:** Predictive analytics for user engagement.
3. **Mobile Responsiveness:** Expanded accessibility.

9. Creative Components

GCP Hosting

The application and database are hosted on **Google Cloud Platform**, ensuring high availability and scalability.

Keyword Search

Implemented an intuitive search interface for users to find services based on:

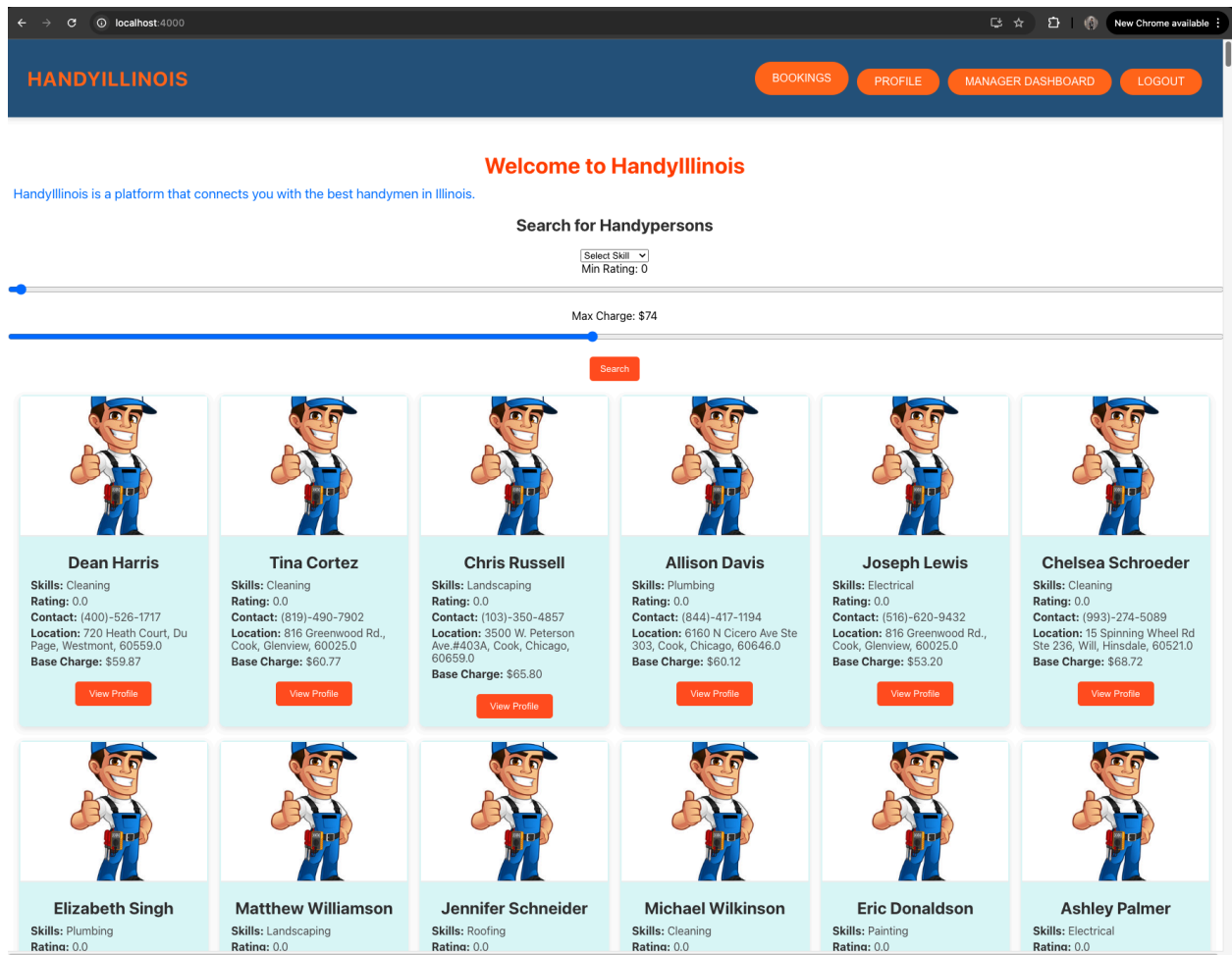
- Location
- Skillset

- Ratings and charges.

Creative Idea: Indexing on frequently queried columns significantly reduced query costs.

UI Design:

Some UI design renders:



Bookings

Ongoing

Past

Fix leaky faucet. Come after 9am and before 4pm

Service Type: Cleaning

Charges: \$60.77

Status: Cancelled

Leave a Review

Review Title

Comment

Rating (1-5)

Submit Review

Manager Dashboard

Manager ID:

Service Requests

Landscaping ▾

ServiceRequestId: 752
Description: apt # 2-l bath: check water pressure in bathroom. restore to proper working condition.
Date: 2024-07-03T00:00:00.000Z
Time: 01:28:42
Status: Completed
Type: Landscaping
Charges: \$24.00
Customer: Shirley Espinoza
HandyMan: Ellen Sharp

Handyperson Performance

HandyId: 821
Handyperson Name: Ellen Sharp
Skills: Landscaping
Average Rating: 1.00000 stars
Total Reviews: 1

Customer Satisfaction

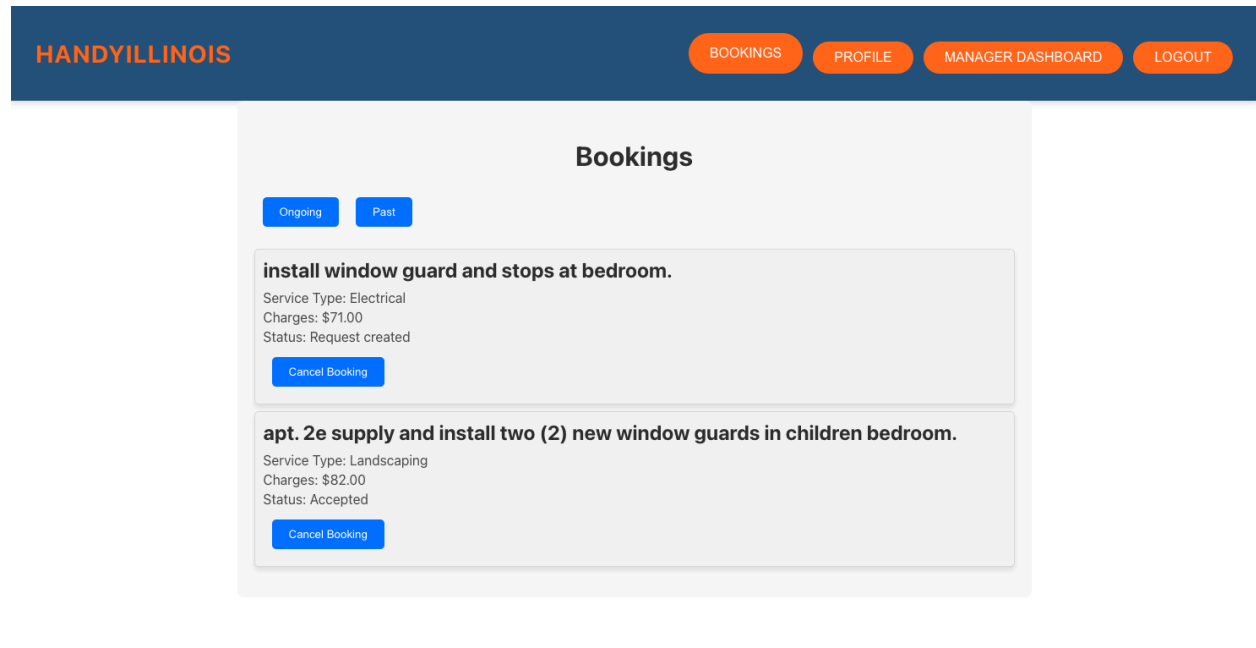
Service Type: Landscaping
Average Customer Satisfaction: 1.00000

Service Earnings

Service Type: Plumbing
Total Requests: 2
Total Earnings: \$275.00

Service Type: Electrical
Total Requests: 1
Total Earnings: \$183.00

Service Type: Cleaning



10. Teamwork and Division of Labor

Backend Development:

- API Design: Trusha Talati, Akshat Bhat, Prajwal Wable, Venkatesh Kamat
- API Node.js Implementation: Trusha Talati, Akshat Bhat
- Database Setup: Trusha Talati, Akshat Bhat, Prajwal Wable, Venkatesh Kamat

Frontend Development:

- React UI Implementation: Prajwal Wable, Venkatesh Kamat

Advanced Features: Collaborative effort on triggers and stored procedures.

Teamwork Assessment: Regular meetings ensured cohesion and efficient task allocation.