Dostt App

Strategy & Products Assignment

Rikita Jatav

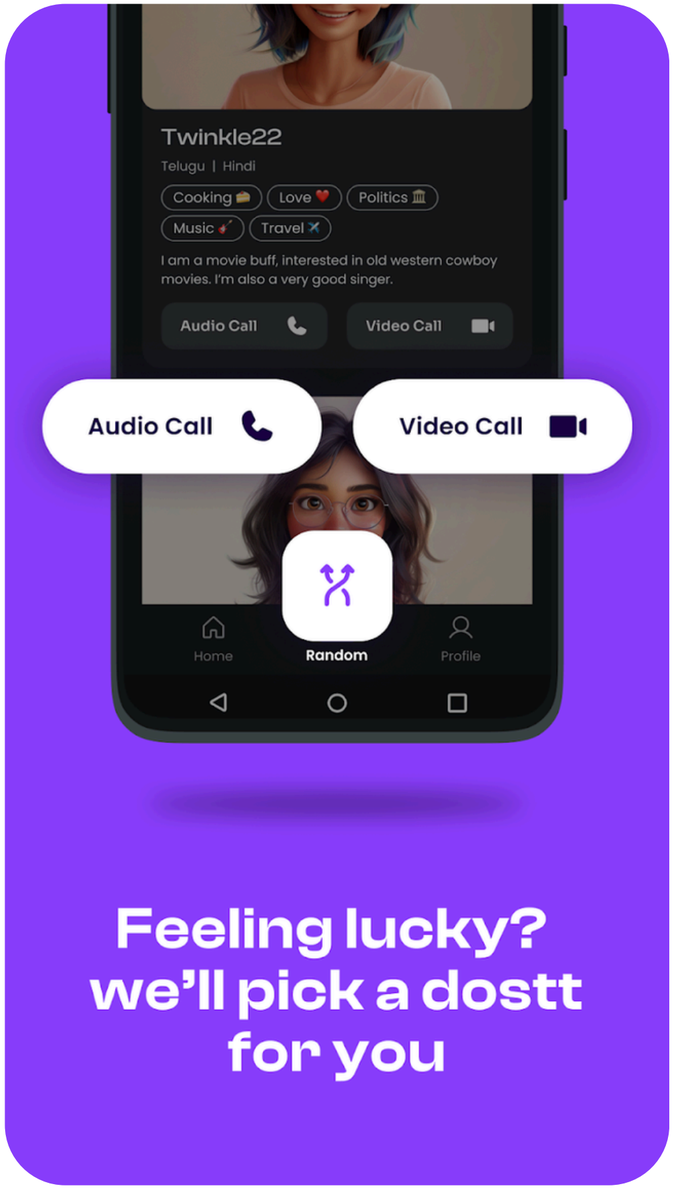
July 2024

Dostt, where connections begin

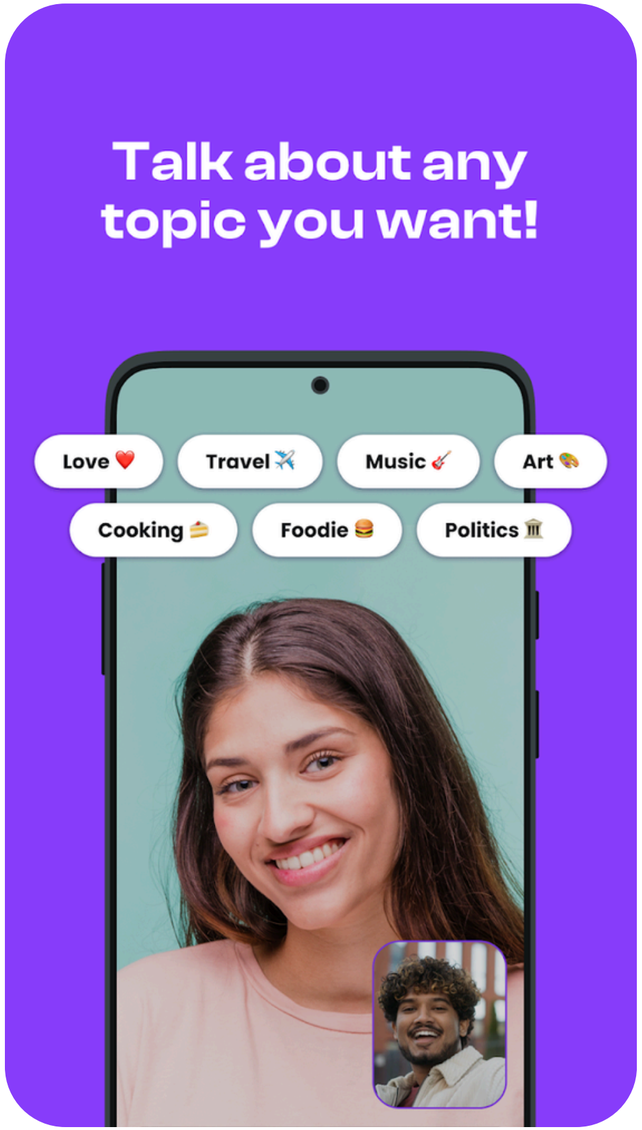
About Dostt

Dostt is a rapidly growing startup revolutionizing the way people connect and form genuine friendships. Designed as a social networking platform, Dostt provides a secure space for meaningful conversations through audio and video calls, ensuring user privacy and safety.

- Genuine Connections: Engage in authentic interactions with verified profiles. Discuss topics like relationships, career, and more with like-minded individuals.
- Superdosts: Dedicated 'super friends' available for calls, offering companionship and support. Users can discuss their life, issues, and have casual conversations via audio and video calls.
- Privacy First: Enjoy private chats without revealing personal information. Dostt ensures a safe and non-judgmental environment.
- Diverse Community: Connect with people across India, with support for multiple languages including Hindi and Telugu.
- Safety Assured: Verified profiles and stringent content moderation for a secure chatting experience.



Problem Statement



Assignment Overview

Problem Statement to be solved:

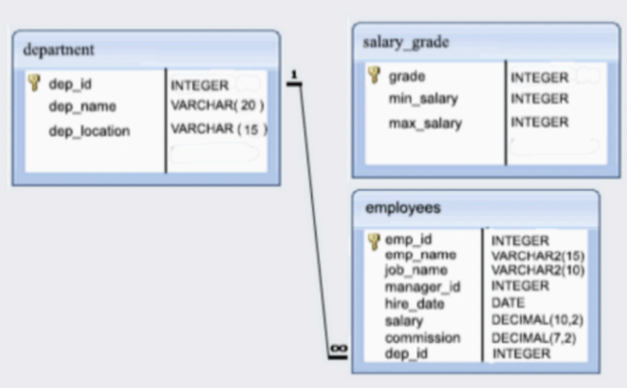
1. **Supply-Demand Equation:**

Our task is to develop a supply-demand equation tailored for Dostt's two-sided marketplace. This equation will help determine the optimal number of Superdosts required to meet the needs of our users. By assuming relevant metrics and using mock data, we aim to ensure a balanced and efficient interaction between users and Superdosts.

2. **Supply-Side Strategy:**

We need to devise a comprehensive strategy to build and manage the lifecycle of Superdosts. This includes identifying sourcing channels, setting selection criteria, creating an onboarding and training process, establishing a pricing and payout structure, assessing performance, and developing retention strategies. Additionally, we will list potential threats to each method in supply acquisition and propose an action plan to mitigate them. A 6-month plan to scale the supply of Superdosts will also be outlined.

3. **SQL Assignment:**



Solving the following SQL Questions based on the given database schema:

1. Write a query in SQL to list the department where there are no employees.
2. Write a query in SQL to find out the employees whose salaries are greater than the salaries of their managers.
3. Write a query in SQL to list the employees whose designation is the same as either the designation of ADLYNE or the salary is more than the salary of WADE.
4. Write a query in SQL to list the ID, name, location, salary, and department of all the employees who have been hired in the months of June & October.

Solutioning

(Solving 1st & 3rd)

Solutioning

Problem Statement 1: Supply-Demand Equation for Superdosts

Objective: To determine the number of Superdosts required for a given number of users on the Dostt platform

Approach:

Understand the Variables:

- **U**: Number of users
- **C**: Average number of calls per user per day
- **D**: Average duration of each call (in minutes)
- **H**: Average hours a Superdost works per day
- **B**: Buffer factor for peak times & service quality
- **S**: Number of Superdosts required

Formulating the Equation:

- Total call hours required per day = U x C x D
- Total Hours a Superdost can handle per day = H
- Including buffer factor: $S = (U \times C \times D/H) \times (1 + B)$

Why add Buffer Factor?

The buffer factor accounts for unexpected spikes in demand, Superdost unavailability, and maintains service quality. It ensures the platform can handle variability and provide a consistently positive user experience.

Assumptions and Reasoning:

a) **C (Average calls per user per day) = 2**
Reasoning: Assuming users make an average of two calls per day, balancing between active and less active users.

b) **D (Average call duration) =** 15 minutes (0.25 hours)
Reasoning: Short enough for casual conversations, long enough for meaningful interactions.

c) **H (Average working hours per Superdost per day) =** 6 hours
Reasoning: Allowing for part-time work and breaks, considering the potentially emotionally demanding nature of the job.

d) **B (Buffer factor) =** 20% (0.2)
Reasoning: Provides a 20% cushion to handle peak times and maintain service quality.

Now, Plugging these assumptions into our equation....

Applying the Equation:

• Let’s apply this with our assumed/mock data

U = 1000 users
C = 2 calls per user per day
D = 0.25 hours (15 minutes) per call
H = 6 working hours per Superdost per day
B = 20% (0.2) buffer factor

Now, Total call hours required per day = 1000 x 2 x 0.25 = 500 hours

Number of Superdosts required, including the buffer factor: $S = (U \times C \times D/H) \times (1 + B)$

S = (500/6) x (1.2) = 100 Superdosts! “For every 10 users, we need 1 Superdost”

So, For a platform with 1000 users, each making an average of 2 calls per day, each lasting 15 minutes, and with each Superdost working 6 hours per day, the platform would need 100 Superdosts to meet the demand, accounting for a 20% buffer for peak times and maintaining service quality.

Further Enhancements and In-depth Analysis:

In the previous section, we established a basic supply-demand equation to determine the optimal number of Superdosts needed for a given number of users on the Dostt platform. By considering key variables and assumptions, we can maintain an efficient balance between user demand and service quality. These Further considerations will help in understanding the Superdost’s availability on the platform in a comprehensive manner.

Now, let's dive deeper into refining our analysis and addressing its limitations. We will explore:

- Adjustments based on varying user behaviors and time-of-day effects.
- Refinements using real app data to enhance accuracy.
- Identifying and improving the limitations of our initial model.

Changes Based on User Behavior and Time of Day:

The equation can be adjusted to account for varying user behaviors and time-of-day effects:

a) **Peak hours**: Increase the buffer factor (B) during known busy periods.
Example: S_peak = ((U * C * D) / H) * (1 + 0.4) // Assuming 40% buffer for peak hours

b) **Night shifts**: Reduce available Superdosts (H) during off-peak hours.
Example: S_night = ((U * C * D) / 4) * (1 + B) // Assuming 4-hour night shifts

c) **Weekends vs. Weekdays:** Adjust C (calls per user) based on observed patterns.
Example: S_weekend = ((U * 3 * D) / H) * (1 + B) // Assuming more calls on weekends

Refining the Equation with Real Data:

To refine this model using real app data:

a) Calculate actual average values for C and D from user activity logs.
b) Analyze Superdost performance data to determine optimal H.
c) Use historical data to identify peak times and adjust B accordingly.
d) Implement time-based variables: Ct (calls at time t) and Dt (duration at time t).

Refined equation: St = ((U * Ct * Dt) / H) * (1 + Bt)

This allows for ‘dynamic adjustment of Superdost numbers throughout the day!’

Limitations and Improvements:

• **User Segmentation:** The current model assumes all users behave similarly.
Improvement: Segment users based on activity levels and create separate equations for each segment.

S = S_high_activity + Smedium_activity + Slow_activity

• **Superdost Efficiency:** The model assumes all Superdosts are equally efficient.
Improvement: Introduce an efficiency factor (E) for each Superdost.

S = ((U * C * D) / (H * E)) * (1 + B)

• **Predictive Modeling:** The current model is reactive.
Improvement: Implement machine learning algorithms to predict demand and adjust supply proactively.

Conclusion:

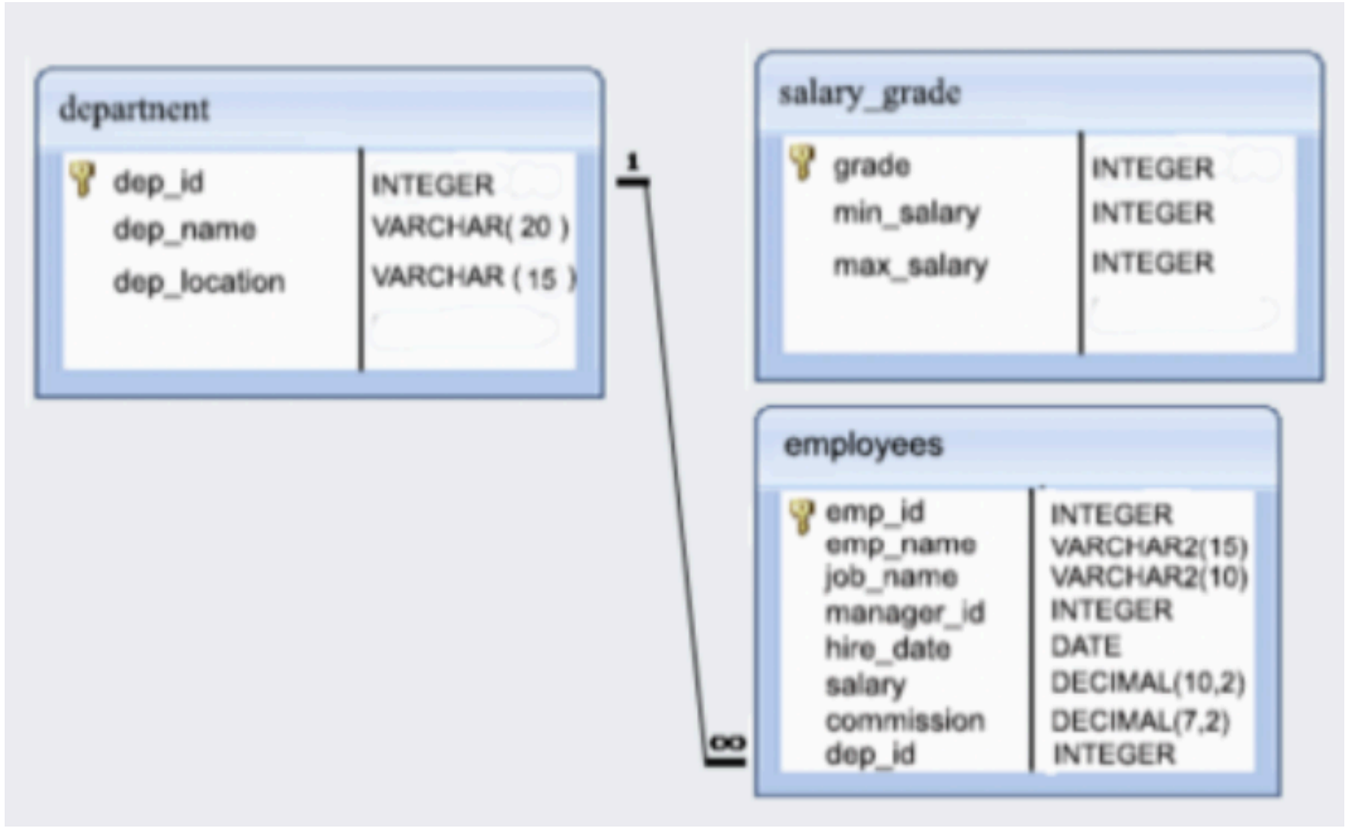
This comprehensive supply-demand model enables Dostt to optimize user satisfaction and resource efficiency while supporting scalable growth. By continuously refining the model with real-world data, Dostt can make informed decisions, adapt to changing market dynamics, and maintain a competitive edge. Ultimately, this approach positions Dostt to meet current needs effectively and shape the future of digital friendship platforms, ensuring long-term success and user value creation.

Solutioning

Problem Statement 3 - SQL Assignment

Solving the given SQL Assignment

Provided Database Schema:



Solutioning

Solving the given SQL Questions:

1. Write a query in SQL to list the department where there are no employees.

solution:

```
SELECT d.dep_name
FROM department d
LEFT JOIN employees e ON d.dep_id = e.dep_id
 WHERE e.emp_id IS NULL;
```

Solutioning

Solving the given SQL Questions:

2. Write a query in SQL to find out the employees whose salaries are greater than the salaries of their managers.

solution:

```
SELECT e1.emp_name AS employee_name, e1.salary AS employee_salary, e2.emp_name AS manager_name, e2.salary AS manager_salary
FROM employees e1
JOIN employees e2 ON e1.manager_id = e2.emp_id
WHERE e1.salary > e2.salary;
```

Solving the given SQL Questions:

3. Write a query in SQL to list the employees whose designation is the same as either the designation of ADLYNE or the salary is more than the salary of WADE.

solution:

```
SELECT e.emp_name, e.job_name, e.salary
FROM employees e
WHERE e.job_name = (SELECT job_name FROM employees WHERE emp_name = 'ADLYNE')
  OR e.salary > (SELECT salary FROM employees WHERE emp_name = 'WADE');
```

Solving the given SQL Questions:

4. Write a query in SQL to list the ID, name, location, salary, and department of all the employees who have been hired in the months of June & October

solution:

```
SELECT e.emp_id, e.emp_name, d.dep_location, e.salary, d.dep_name
FROM employees e
 INNER JOIN department d ON e.dep_id = d.dep_id
WHERE MONTH(e.hire_date) IN (6, 10);
```

Thank You!