Doordash Robonav

PRD

PM: Siddhant Tembhurnikar
UX: Pat Pixels
EM: Casey Code
DS: Noel Numbers

STATUS: DRAFT

Background

Consumer behaviour has shifted towards a demand for faster deliveries. As a result a lot of them order from neighbouring restaurants. However in spite of the increase in the demand it has

a negative effect on our bottom line. Last mile delivery across the industry constitutes a huge cost of the supply chain while the demand keeps growing.

Problem

Last mile delivery constitutes 40% of distribution costs. Net profit is impacted by 26% because of last mile delivery logistics. Currently on average last mile deliveries cost us about $10. However we subsidise this cost because consumers will be unwilling to pay high delivery prices especially when they originate from nearby restaurants. We currently charge between $2-6 which means we lost about $4-8 of revenue on each last mile delivery. Our Consumers are demanding faster deliveries and Studies show that they'd be willing to pay more as well. 55% of users may switch to a competitor for a faster service. Currently our competitors are working on a competing automation solution to solve this problem. If we are able to reduce our costs to $2 we are looking at a TAM of around $480 million.

Goals

- Build an automated delivery system using self-driving robots to deliver food
- Robots should eventually replace most of our last mile delivery logistics
- Robots need to be small, safe and secure
- Build the system to be scalable over time
- Reduce last mile delivery costs down to <$2 per order

Success Metrics

- Reduce delivery time by 30%
- Increase total % of autonomous driving to 90%
- Increase total number of deliveries in the area by 20%
- Minimize number of accidents

Key Features & Scope

Priority	Feature	Description
P2	Map View	Overall view of all the robots assigned to the operator. The operator can see the robots on a map to know where exactly where they are
P0	List View	List View displays a list of all robots assigned with an overview of high level

		statistics of the robot. The user can click on an item to go into navigation tool
P0	Statistics	All diagnostic statistics of sensor data from the robot useful to diagnose the issue with the robot
P0	Navigation Tool	Tool that displays all the camera views from the robot and provides controls to the remote operator to remotely move the robot. The robot is controlled by the operator using pair of joysticks, one to move the camera and the other to move the robot
P2	On screen controls	On screen controls to move the robot
P2	Communication controls	Video and Audio Controls to communicate with the customer or another human in front of the robot
P0	Sensor Reset/Reboot	Remotely reset or reboot the robot sensors to fix any minor issues
P0	Emergency Alarm	Issue an emergency alarm if someone tries to open the robot container or steal the robot. The alarm would sound an alert on the robot send the location of the robot to the closest mobile operator and make a 911 call
P2	Smart Quick Service Menu	A Quick service menu on the navigation tool page that will automatically change according to the frequency of occurrence of an issue. For example if the ‘Opening the container” is the most frequently used feature, it will

		float in the quick service menu on the navigation tool page for quick access
P2	Call Restaurant	Place a call at the restaurant and connect the customer through
P2	Resend SMS	Resend SMS with the link to the customer to open and access the container
P1	Open Container	Remotely open the container for a dasher or customer who doesn't have access to their primary phone
P1	Remote Troubleshoot	Remote Troubleshooting feature to diagnose issues with the robot
P0	Generate Log Report	Generate a historical log of the historical status of the robot and its sensors
P0	Statistics Report	Generate a report of current status of the robot and its sensors
P2	Service menu map	Map that displays all the dashers and remote ops in the locality near the chosen robot
P0	Assign Ops	Assign the issue to a remote ops team to come and service a faulty robot
P1	Assign Dasher	Assign the order to a Dasher to come and relay the order to the customer from a faulty robot
P2	Call Dasher/Ops	Call a dasher or ops team member to check on their ETA before assigning them the job

Target Market

Since it is an internal tool, the users are the members of the operations team who will be responsible for monitoring the status of the Self Driving Delivery Robots as they deliver orders from restaurants to the customers. In the end our customers are people who order food through online food delivery platforms.

Core UX Flow

[Mocks and Prototype]

TAM

- ARPU = 200$ approx[2]
- takeout price for 1 person 15$ approx[1]
- Household size in US = 2.52
- Hence takeout price = 15*2.52 = 37.8
- delivery plus service = 10$ approx[1]
- Commission from restaurants = 20% * 37.8 = $7.56
- 200/17.56= 11-12 number of orders per year per customer
- 60 million people order food from platforms[2]
- 1/3rd is doordash cust 20 mill
- Assuming all of them live in urban areas and 1/5th of them order from neighbouring restaurants: 20/4 = 5 million • For 5 million users We can make additional 10-4 + 4-2 = $8 per order
- That is 5*12*8 = **$480 mill in additional revenue**

Sources:

1. https://www.nytimes.com/2020/02/26/technology/personaltech/ubereats-doordash-postmates-grubhub-review.html
2. https://www.statista.com/outlook/376/109/platform-to-consumer-delivery/united-states

Competitors

Postmates:

- Currently have 8% of the Total Platform to consumer food delivery market[1]
- Platform to consumer food delivery market is worth US$ 10,896m[2]
- Thus Total Revenue earned by Postmates is around US$ 871.68m
- Number of users = Total number of platform to delivery users * market share = 58.9[2] * 8% = 4.712 million users
- Currently also have a Self Driving Robot project called postmates serve
- Currently in pillot stage

- Build their own robot
- Travels much longer than competing vendor robots • More capacity than competing vendor robots
- Friendly and socially aware to pedestrians

UberEats:

- Currently have a market share of 24%[1]
- Platform to consumer food delivery market is worth US$ 10,896m[2]
- Number of users = Total number of platform to delivery users * market share = 58.9[2] * 24% = 14.136 million users
- Thus total revenue earned by Ubereats in the last year is around US$ 2615.04m
- Currently have a Drone Delivery project
- Can potentially expand faster into automated delivery because of wide user base of other business verticals such as Uber
- Drone delivery can be faster than robots on the ground
- Currently only in a pilot program stage
- Can face more regulatory hurdles and safety concerns with drone delivery

Sources:

1. https://secondmeasure.com/datapoints/food-delivery-services-grubhub-uber-eats-doordash-postmates/#:~:text=Postmates%20earned%208%20percent%20of,1%20percent%20of%20national%20sales.
2. https://www.statista.com/outlook/376/109/platform-to-consumer-delivery/united-states

Channel Acquisition Strategy

Since the product is an internal tool there is no need to market the product on any public facing channel. Instead the product would be championed by:

Product Team: This includes the product team responsible for development of the product. They will setup a Demo at the end of the sprint which will showcase the benefits of using the product to the operations team.

Product Manager: Responsible for the overall ideation of the product working alongside the operations team to understand their requirements. After the development of the product the PM would work to evangelize the product to the operations team.

Operations Team Head: The initiator of the project who requested the product to be developed. They will be looking over the adoption of the product by the team responsible for monitoring the Robots

Marketing Guide

https://docs.google.com/document/d/1Zg25BWPgaPU7te2oqEIHaFCGL8O-uTExfI418njUxvU/edit?usp=sharing

Pricing Strategy

- Currently last mile costs the company $10 on average
- Consumers would be charged between $2-6 approx $4
- Automation robot vendors charge $2 per delivery
- Total revenue gained compared to previous = (4-2) + (10-4) = $8 per delivery
- Lets calculate for one neighbourhood of 2km radius the return and investment • 20k house holds - 60 robots
- Robots cost as much as a high end laptop i.e. around $2k = $120k for buying the robots
- Average number of people in a household in US = 2.52
- Assume 1/3rd of the people are Food delivery users
- From TAM calculations we know that one person orders 11 times in a year
- With each delivery doordash earns 17.56$ from delivery charges as well as commission.
- Total deliveries in a year = 20k x 2.52/3 = 50/3. 50*11/3 = 550k deliveries
- Total revenue = 550k *17.56/3 = $3,219,000. This is our Revenue goal for the neighbourhood
- This goal allows us to realistically look the gains we make from each neighborhood launch and takes a bottom up approach to calculate overall revenue from the entire project launch across the country over a period of time
- The project would achieve this in the form of savings over the previous delivery system, prices charged to the customers remain the same. Total revenue gained per delivery now would be $8 per delivery

Pre Launch Checklist

- Internal IT team: Acts as the customer service team for the internal tool users to address any issues related to the product
    - Discuss training required by the team to support the launch
- Technical Writers
    - Discuss the documentation requirements for the final product
- Product Specialist
    - Work alongside them to deploy the product and provide all necessary training guidance through them to the operations team

Anticipate Risks

- Is the product thoroughly tested before launch for all possible test cases? A failed product might make it difficult for the operations team to service a Robot or send a remote team to service it
    - Mitigation: Make sure there is a roll back plan to the previous stable version of the product so that in case of a failure the user can be moved to the previous

version of the product and there is not breakdown in the remote monitoring process

- Is there enough support available for the product
    - Make sure there is enough flexibility in the IT team capacity assigned to support the operations team wrt the product such that any surge in queries are quickly handled and there is no delay in getting the order delivered to the customers
- Are there any legal implications to recording video footage through the Robot camera in certain neighbourhoods
    - Work alongside the legal team to make sure the Self Driving Robot system is compliant with all local laws when it launches in a particular area since laws can differ from place to place. Make sure to have this meeting on a regular basis if in case there are changes in law.

Training Guide for Customer Support Team

https://docs.google.com/document/d/1Hjwo41DNjmdrKOyDeDzBySXJm6YBqLnTvtCavZ_fklo/edit?usp=sharing

User Guide

https://docs.google.com/document/d/1jW-t_4lmUZx2J8Ei_OjNoCIMMuCt0klvi6DKs_ScFrM/edit?usp=sharing

Post Launch Feedback Issues

Post the launch we've received feedback that around 25% of all our deliveries there are certain issues. As of now 25% amounts to around 150 deliveries in a week, and as we scale further it may increase the number of issues further. We believe this could be possibly because of certain malfunctions being undetected by the operations team. From our analysis of this issue we find that most of these 150 orders suffered issues from a LIDAR sensor that wasn't functioning properly. Our Map View does not provide the LIDAR status on the Robot card and the priority in the list is decided only according to idle time. This leads the Robot to move in the wrong direction on its own until the LIDAR resets on its own, thus increasing the delivery times in spite of low traffic. And since the robot is still moving the issue remains undetected to the operator. To fix this we need to display the LIDAR status on each of the Robot cards on the Map View itself so that the operator can monitor the status right away and reset the sensor in case of an error.

However in order to determine this fixes the problem we need to setup an A/B test:

A/B Test

- For the users in the control group: we will do nothing (group A)

- For the users in the variant group: we will add the LIDAR status feature to the Map View (Group B)
- Our hypothesis is that adding the LIDAR status to the Map View will make it easier for operators to detect errors related to LIDAR status failing quickly and the number of order issues would be lower in Group B
- Our metric to determine success would be if the number of issues in group B have gone down over a period of time compared to Group A and is also lesser than the current error rate that is 25%

Launch Email

https://docs.google.com/document/d/1l_knil_jlQDYHTKjxRzTFT1pCGcrJohp3GC6LvvSWUU/edit?usp=sharing