DoorDash: Autonomous Delivery

Design Sprint

Product Manager: Christopher O'Hara

Set the stage

Set the stage for the Design Sprint by framing the problem

Initial PRD

Link to PRDv1

Understand

Create a shared understanding of the space, problem, and goals

How Might We: Christopher

How might we enable robots to send confirmations to users?

How might we allow users to track robots/orders?

How might we optimize path planning?

How might we ensure robots adhere to safety policies?

How might we construct robot greetings for interacting with users?

How might we control multi-agent systems effectively?

How might we prevent degradation of robots due to high usage?

How might we prevent mold from growing inside the robot?

How might we handle charging of robot fuel cells?

How might we prevent robots from harming pedestrians?

How might we raise user trust in the domain of robotics?

How might we prevent robots from being stolen or abused?

How might we make the API/platform open source?

How might we implement affective computing in robots?

How might we confirm robots deliver to the proper client?

How might we protect the privacy of users when interacting?

How might we evaluate customer satisfaction upon delivery?

How might we collaborate with other delivery services?

How do we prevent hacking for our robots?

How might we autonomously check for degradation in components?

How Might We: Shin

How might we keep vermin away from the robots?

How might we keep robots odor free, even when carrying smelly food?

How might we make routes more efficient?

How might we use robots to make people excited about our brand?

How might we have robots entertain customers at delivery?

How might we have robots signal distress when something goes wrong?

How might we teach robots to avoid trouble?

How might we make robots not scary for dogs?

How might we share robot progress with consumers?

How might we teach robots to avoid obstacles?

How might we move robots to a safe place before stopping?

How might we make our robots tamperproof?

How might we change robot's appearance?

How might we teach users to interact with humans?

How might we see real-time traffic on the route?

How might we communicate with humans around the robot?

How might we monitor robot progress?

How might we confirm that the robot is at the right address?

How might we anticipate mechanical failures?

How might we give robots a personality?

How Might We: Tiffany

How might we deal with accidents that might occur?

How might we control robots?

How might we alert consumers if their delivery is delayed?

How might we determine when to recharge robot batteries?

How might we establish preferred routes?

How might we determine the best area for launching this program?

How might we allow users to help us with tracking and feedback?

How might we teach robots manners?

How might we mitigate accidents between robots and pedestrians?

How might we train our operations team on monitoring and controlling a robot?

How might we use existing technologies?

How might we track each robot?

How might we collect data about where the robot got stuck?

How might we handle edge case issues that may arise?

How might we get food to people quickly when the robot fails?

How might we use use greener energy to power our robots?

How might we leverage existing tools to better track our robots?

How might we create ML/AI models to help robots learn to get better overtime?

How might we build redundancy into our system?

How might we ensure food gets delivered without incident?

How Might We: Alex

How might we streamline communications between operators and robots?

How might we address a sudden power outage?

How might we program robots to address delays in deliveries?

How might we enable robots to interpret and speak different languages?

How might we make interacting with robots more fun?

How might we Increase robot speed?

How might we overcome technical glitches during a delivery?

How might we allow robots to detect real-time traffic patterns?

How might we prepare robot to handle deliveries to persons with disabilities?

How might we accept tips that some customers may want to give a robot?

How might we enable robots to detect missing items in the order during pickup?

How might we make our robots act like people?

How might we enable “emotion” modes in robots?

How might we detect when a robot needs help?

How might we alert operators of need for robot intervention conveniently?

How might we program robots to address order cancellations?

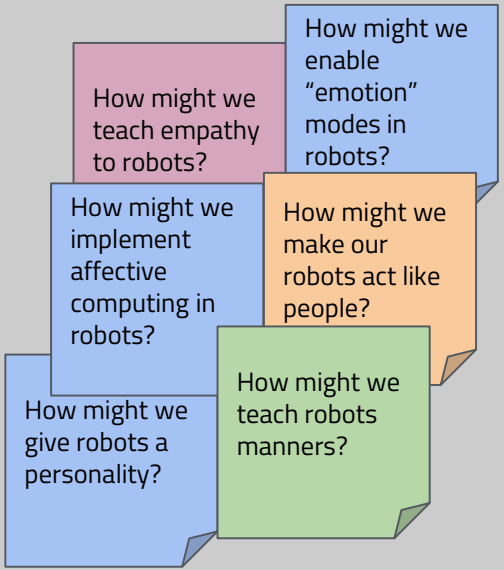
How might we teach empathy to robots?

How might we enable robots enter a crowded restaurant to pickup food?

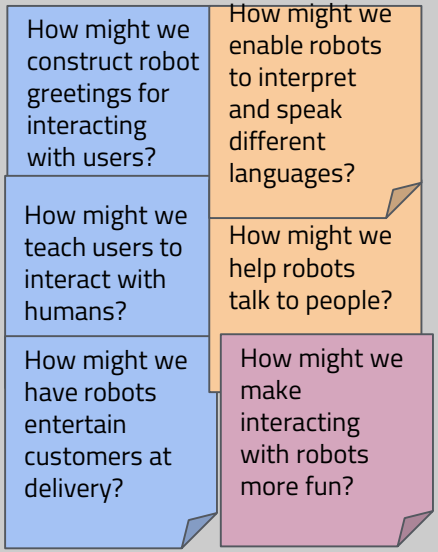
How might we help robots talk to people?

How might we program robots to address customer returns?

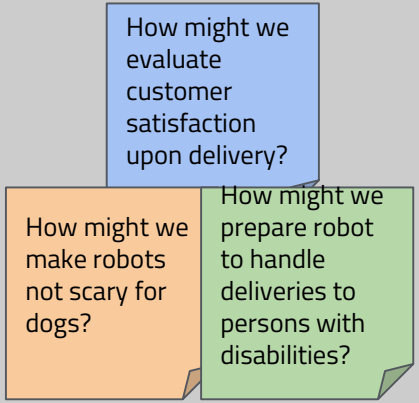
[Affective/Emotional Intelligence]



[Affective Computing]

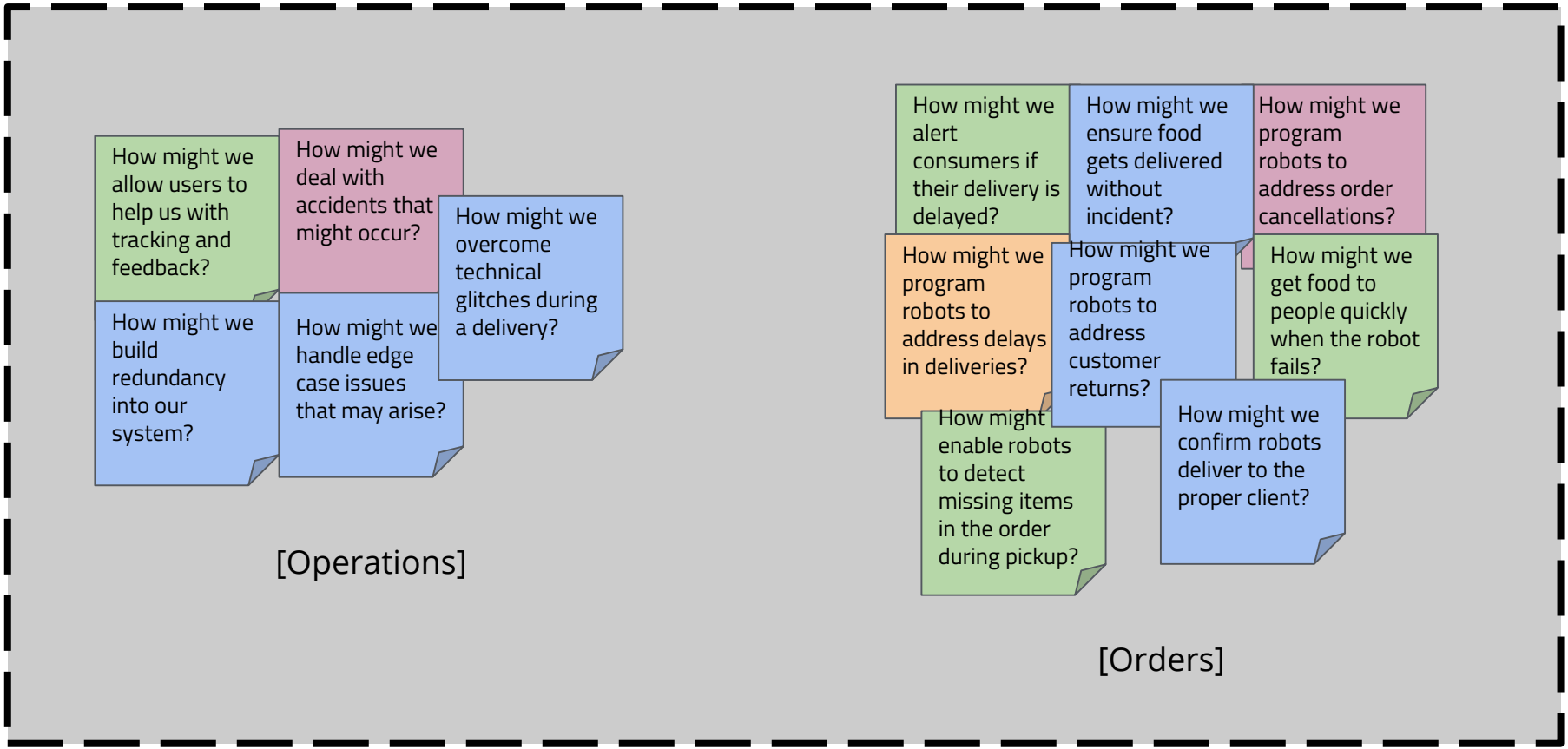


[Human-Robot Interaction]

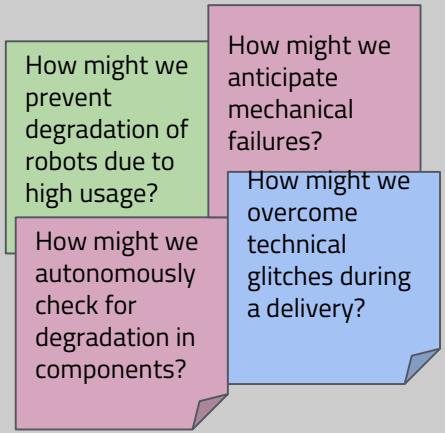


[Edge Cases]

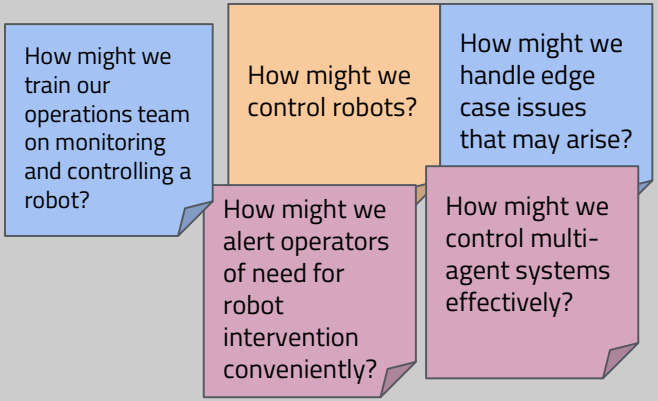
[Quality Assurance]



[Maintenance & Control]

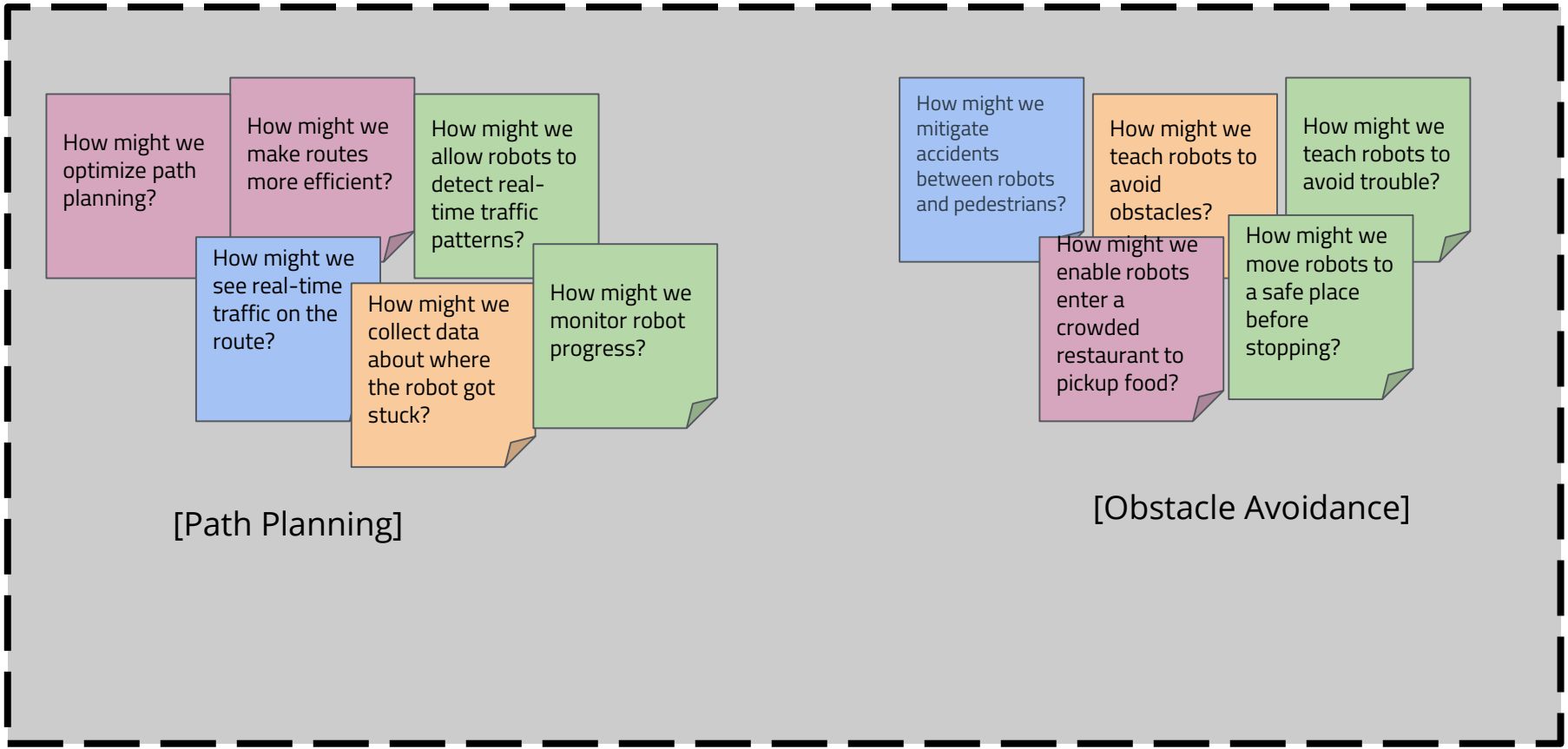


[Maintenance]

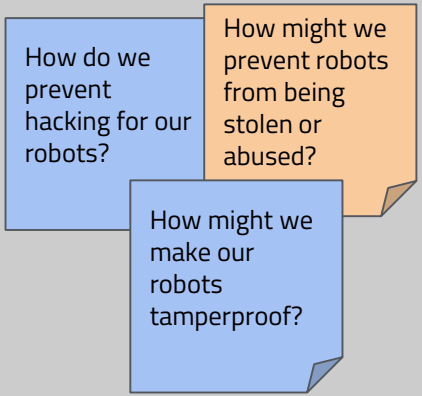


[Control]

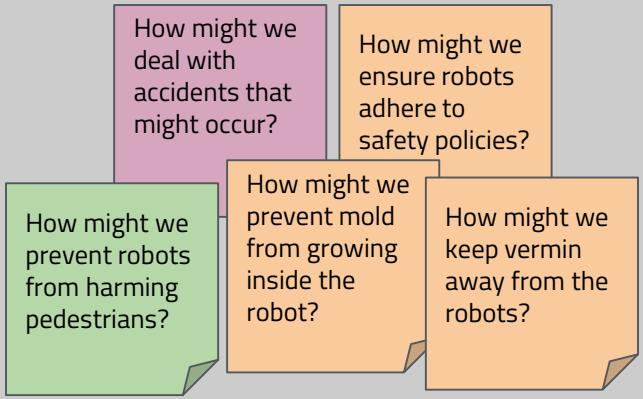
[Path Planning & Obstacle Avoidance]



[Security & Safety]

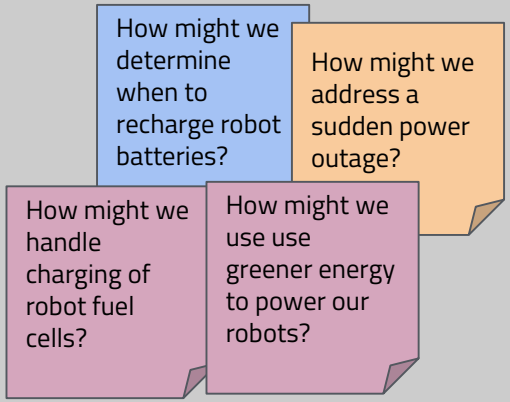


[Security]

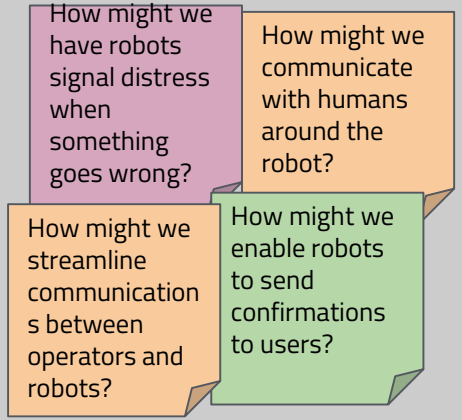


[Safety]

[Performance]

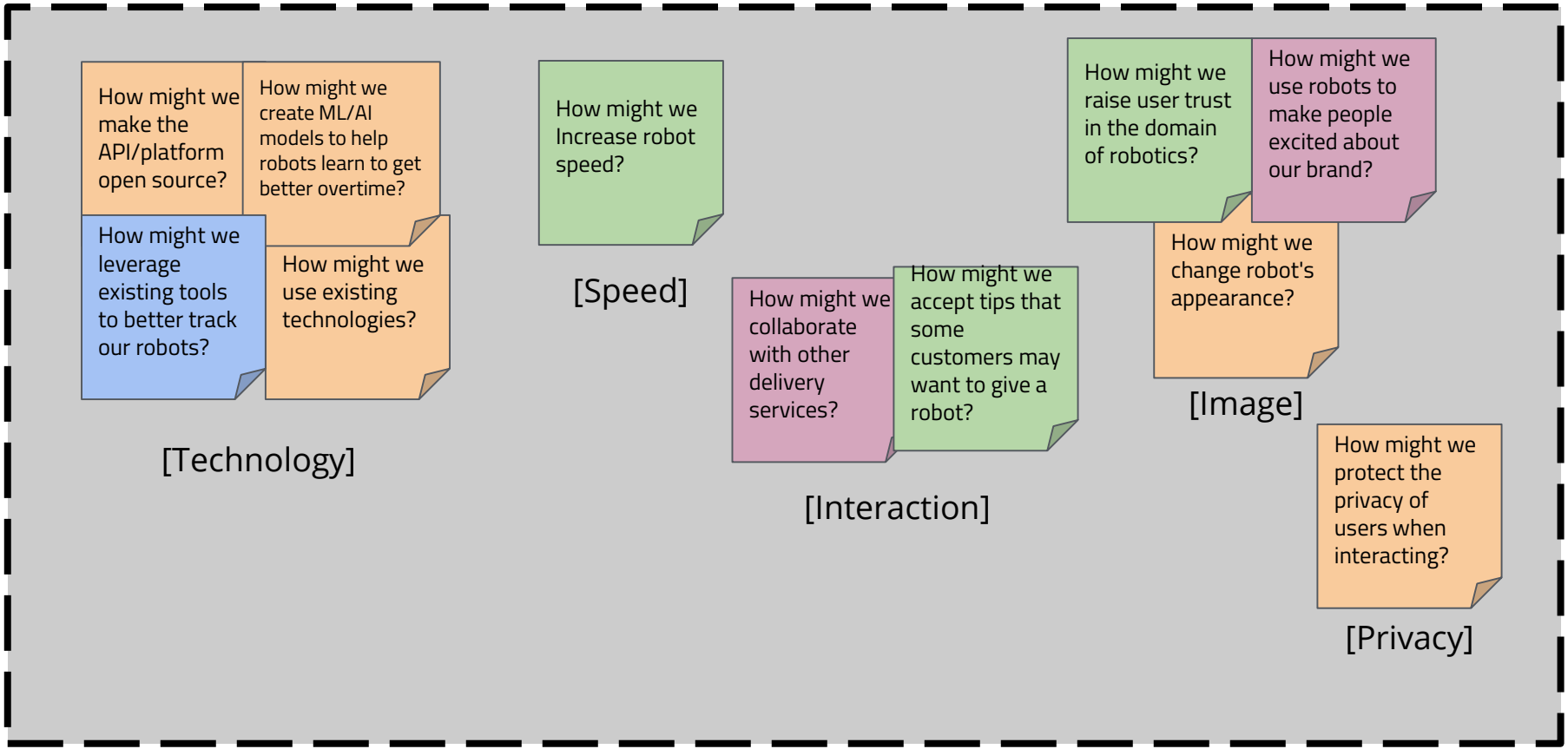


[Energy]



[Communication]

Others


How might we make the API/platform open source?
How might we create ML/AI models to help robots learn to get better overtime?
How might we leverage existing tools to better track our robots?
How might we use existing technologies?
[Technology]
How might we Increase robot speed?
[Speed]
How might we collaborate with other delivery services?
How might we accept tips that some customers may want to give a robot?
[Interaction]
How might we raise user trust in the domain of robotics?
How might we use robots to make people excited about our brand?
How might we change robot's appearance?
[Image]
How might we protect the privacy of users when interacting?
[Privacy]

Sprint Focus

Focus	Quality Assurance: Orders
Slide #	List slide #10
Theme Selection Decision:	This theme appears to have a largest number of opportunities that are technologically feasible to solve. Furthermore, QA allows for user-centered design and engagement for user interfaces. The user interface is ultimately where users will spend the most time and should be intuitive and optimal in design to ensure proper user experience.

Define

With an understanding of the problem space, create focus and align on specific outcomes for the Design Sprint

Success Metrics

	Goals	Signals	Metrics
Happiness	- High level of user satisfaction when using app - DoorDash review ratings improve	- Users provide positive reviews for DoorDash online	- Number of high-level ratings on appropriate websites (quantitative) - Positive feedback and content within review/rating (qualitative)
Engagement	- Users directly communicate with the robot regarding order status - Users look forward to interacting with the "delivery robot"	- Changes in the amount of time spent in app	- Percent change of time spent in app to complete a task (quantitative, lower % is better) - Percent change of amount of app uses (higher % is better)
Adoption	- Users recommend service to others (self-advertisement) - Potential users have a peaked interest	- Increased number of new user acquisitions - Expansion of desired coverage area	- Quantitative increase in unique (new) users - Maximum range of delivery location value
Retention	- Users continue using the service/app - Users use the service more often	- Previous users utilizing app for additional orders	- Increased number of returning users - Increased frequency of user orders
Task Success	- Users can easily order/adjust using the app - Users can easily interact with the delivery robot	- Proportion of customer support requests are decreased for hotline/chat and increased for the robot/app	- Percent decrease in human representative conversations - Percent increase in automated tickets to the robot

“DoorDash: Robots Get it Right” by MIT Media Lab

Ordering food has become ubiquitous in modern society with over 971 million users generating over $107,438,000,000 in 2019 [Statista 2019]. However, it is no secret that ordering food online comes with its own share of problems including delayed orders, user cancellations, and missing items (like your dipping sauce for your wing). Calling into the restaurant or delivery service for inquiries can be troublesome and it is often not clear for users what they should. This leads to customer dissatisfaction and a desire to use a different service (or no service).

However, all of that is about to change with DoorDash’s new robot delivery service. Cute, autonomous robots will be able to handle food delivery orders by communicating directly with the user via an application. From the application, users will be able to track, cancel, and modify orders by interacting with the delivery robot. The app will allow for the robot to “talk” to the user in natural language or via text. This will lessen the annoyance and burden of the customer that does not understand why their Pizza is missing and who they should talk to. A quick google search for “[x] pizza company reviews” will demonstrate what an “unhappy” customer looks like. This technology will result in a decrease in the need for contacting the ordering service directly.

According to the Project Manager, Christopher O’Hara, the goal is to “provide users with a comfortable interface that reduces the shortcomings of food delivery services while providing users with a direct line of communication to a friendly, cute, autonomous agent.” Future plans are to incorporate affective computing via sentiment analysis, as well as users being able to “virtually personalize” the robot from their app (customizable colors, voice, and personality). Depending on the impact and adoption rate, it is anticipated that all orders will be handled autonomously in the future. To handle delivery issues, DoorDash will have an operations team that can remotely control the robot. Happy eating!

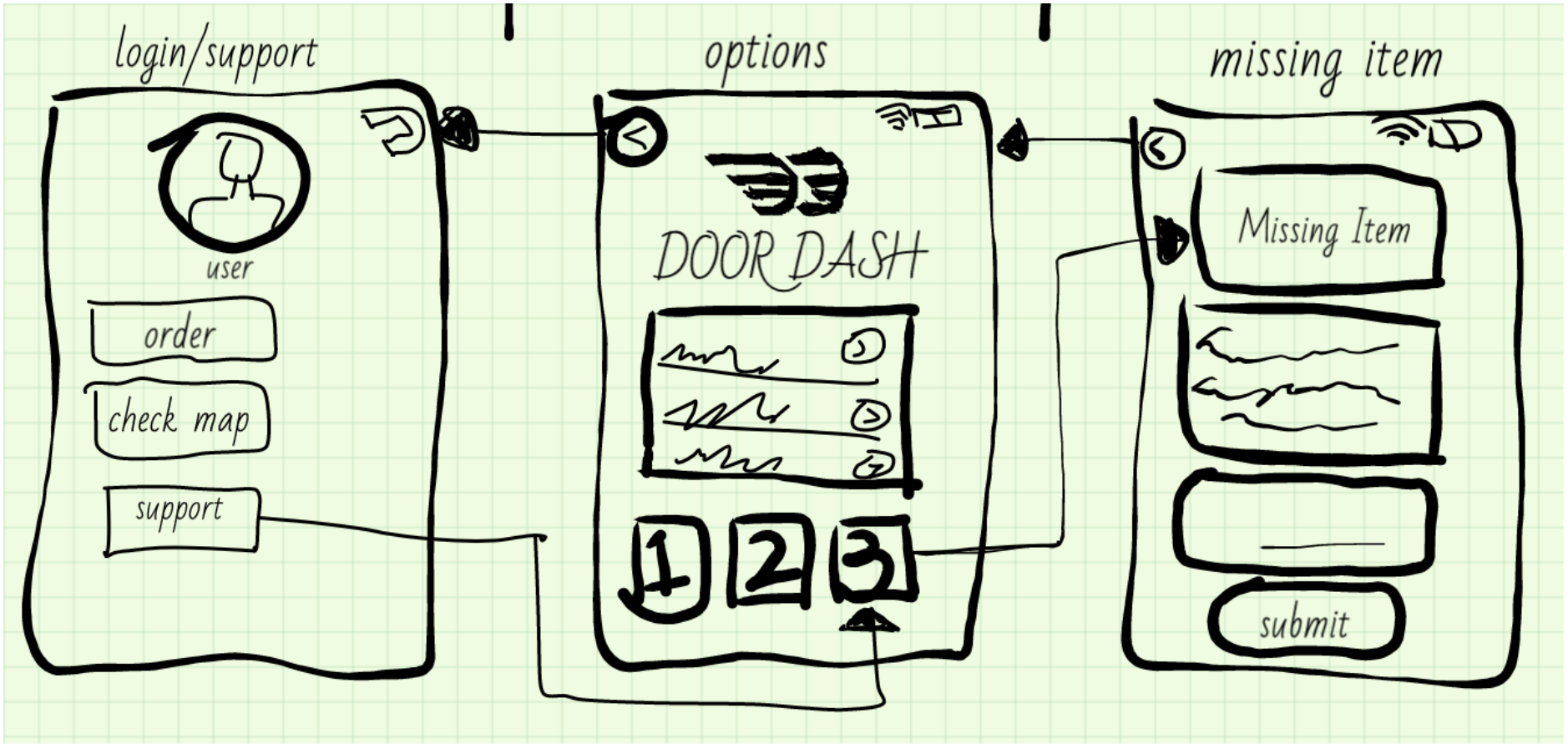
Sketch

Generate tons of ideas, then narrow them down to two in depth solution sketches

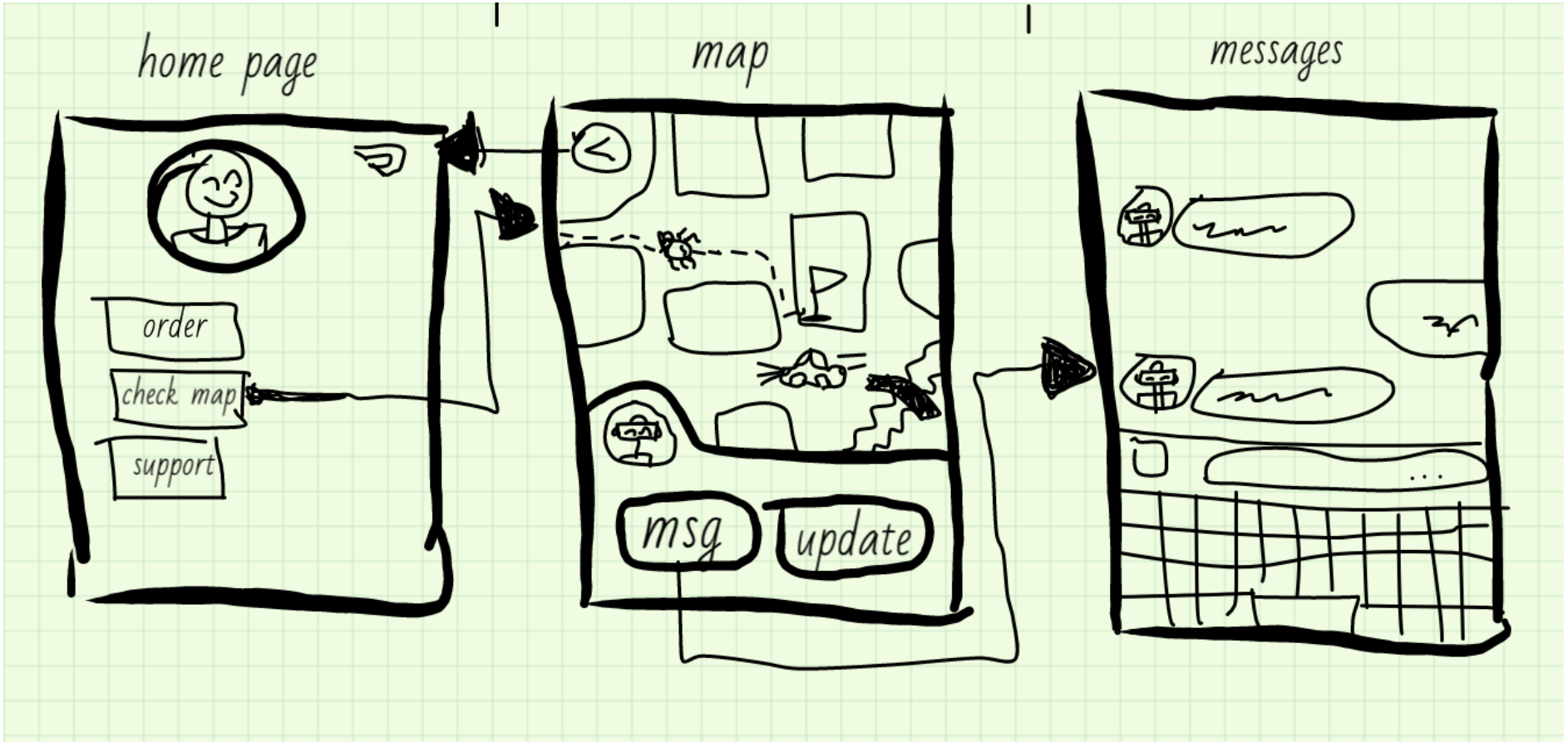
8 Sketches


Login
MAP
CHAT
MENU
Interact
robot
Notifx
18:37
Here
feedback
Thx!
Game

Login/Missing Item Sketch



Map/Messages Sketch



Decide

Pick the final concept that you develop into a prototype

Decision

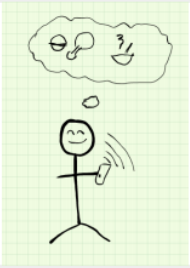
Decision	Combination: Login/Missing Item and Map/Messages Sketches
Rationale	As each UI path can be accessed from the “Home” screen, it should require little additional effort to explore both sketches simultaneously. Furthermore, it is not expected to require additional narration to complete a task that has been designed within three layers (standard UI/UX rules).

Prototype

Turn your concept into a realistic, interactive prototype that you will use to validate your assumptions and ideas

Storyboard

Link to plot



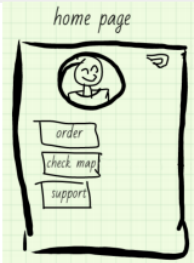
1

SCRIPT

A hungry customer would like to order food from their favorite restaurant. They use the DoorDash app to order food.

ACTION

[order food]



2

SCRIPT

The order has been placed and the user would like to check the status of the robot. They select the [check map] button.

ACTION

click [check map]

3

SCRIPT

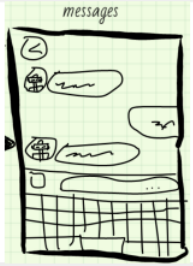
The map is displayed with the current route of the robot. The user wants to message the robot directly to find the estimate time of arrival.

ACTION

click [msg]

Storyboard

Link to plot



4

SCRIPT

The robot responds with the current ETA. The user receives their food shortly after.

ACTION

[no operation] or [click back]

5

SCRIPT

However, one of the items was missing from the user's order. They would like a replacement item.

ACTION

click [missing item]

6

SCRIPT

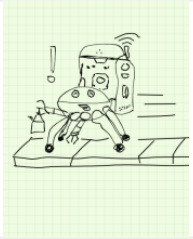
The [missing item] screen appears and the user is able to communicate with the robot for a replacement order. The customer places the order directly with the robot and waits for their food.

ACTION

enter information and click [submit]

Storyboard

Link to plot



7

SCRIPT

The robot acquires the missing item from the restaurant and returns to the user. The customer is satisfied with the service, the restaurant has little overhead, and the autonomous agent is not frustrated. Everybody is happy.

ACTION

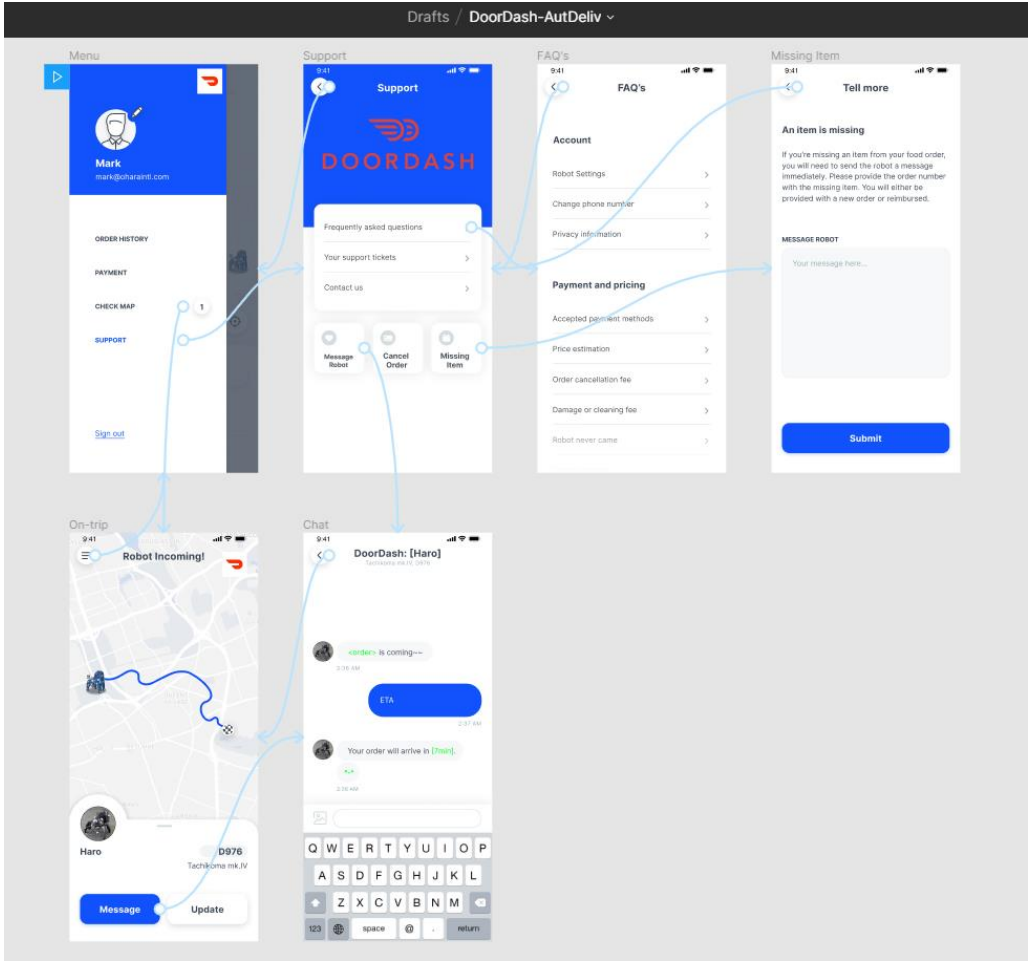
[no operation]

Prototype

Description	The prototype is designed to take the user through two possible uses of the application. The user will check the location of the robot on the map and send them a “dummy message.” The user is also able to report a missing item.
Assumptions	Assumptions made regarding the prototype: • Users will not engage in a detailed walkthrough of the prototype • Users will have already signed-up/signed-in • Users can easily understand symbols
Tasks	Flows/tasks that a user can complete in the prototype: • Users can check the map/robot status • Users can send the robot a message • Users can report a missing item

Link to Prototype

Prototype


Drafts / DoorDash-AutDeliv
Menu
Mark
ORDER HISTORY
PAYMENT
CHECK MAP
SUPPORT
Sign out
Support
DOORDASH
Frequently asked questions
Your support tickets
Contact us
Message Robot
Cancel Order
Missing Item
FAQ's
Account
Robot Settings
Change phone number
Privacy information
Payment and pricing
Accepted payment methods
Price estimation
Order cancellation fee
Damage or cleaning fee
Robot never came
Missing Item
Tell more
An item is missing
MESSAGE ROBOT
Submit
On-trip
Robot Incoming!
Haro
D976
Message
Update
Chat
DoorDash: [Haro]
ETA

Validate

Users will go through your prototype and provide feedback on your concept. This is also an opportunity to have an engineering feasibility discussion

Plan and recruit for research

Link to research plan

User Testing

Link to notes

Link to audio recording

Key Findings from Participant 1

What worked well	General layout, task flow for using map and messaging the robot.
Where participants got stuck	Difficulty in translating the “support” button as the method for reporting an item was missing. Appropriate clicking of buttons on the home page.
Other observations	The application seemed intuitive and easy to use. Reminded the user of a taxi application.

User Testing

Link to notes

Link to audio recording

Key Findings from Participant 2

What worked well	General layout, task flow for using map and messaging the robot are good.
Where participants got stuck	Difficulty in translating the “support” button as the method for reporting an item was missing. Appropriate clicking of buttons on the home page.
Other observations	The application seemed intuitive and easy to use. Reminded the user of the Uber application. “Support” seems more for technical problems like reporting email issues. Enthusiastic about autonomous robotic delivery.

Improvements

Improvement #1	Make larger buttons on Home screen.
Rationale	User disliked the “list” method for buttons on the top page. The buttons will be enlarged to make them more easily clickable.
Improvement #2	Improve Home screen layout.
Rationale	User disliked the layout and button style on the Home screen. The layout will be updated to move the DoorDash logo and the buttons will be placed in a more intuitive and easier to access manner.

Feasibility

	Your Assumptions	Specific feasibility questions
Drawing the UI	- Robot telemetry information needed in real-time - Restaurant order information needed in real-time - Text from messages between user and robot should be recorded	- Which GPS service would be the most reliable? - What metadata related to the user will be stored? - Should data be updated to the Cloud in real-time or based on a schedule?
User generated data	- User data will be stored on a Cloud server - User data will be used to improve application performance and delivery service	- What is the minimal amount of data needed for future datasets? - How much overhead costs will be incurred for using Cloud services?
Latency	- Screens should load within 200ms - Telemetry data might have delays in the returning the accurate position of the robot	- Can the application throughput be optimized by continuously synchronizing the application? - Will this synchronization require additional energy consumption of the device? - How will restaurant, user, and robot data be synchronized?

Iterate

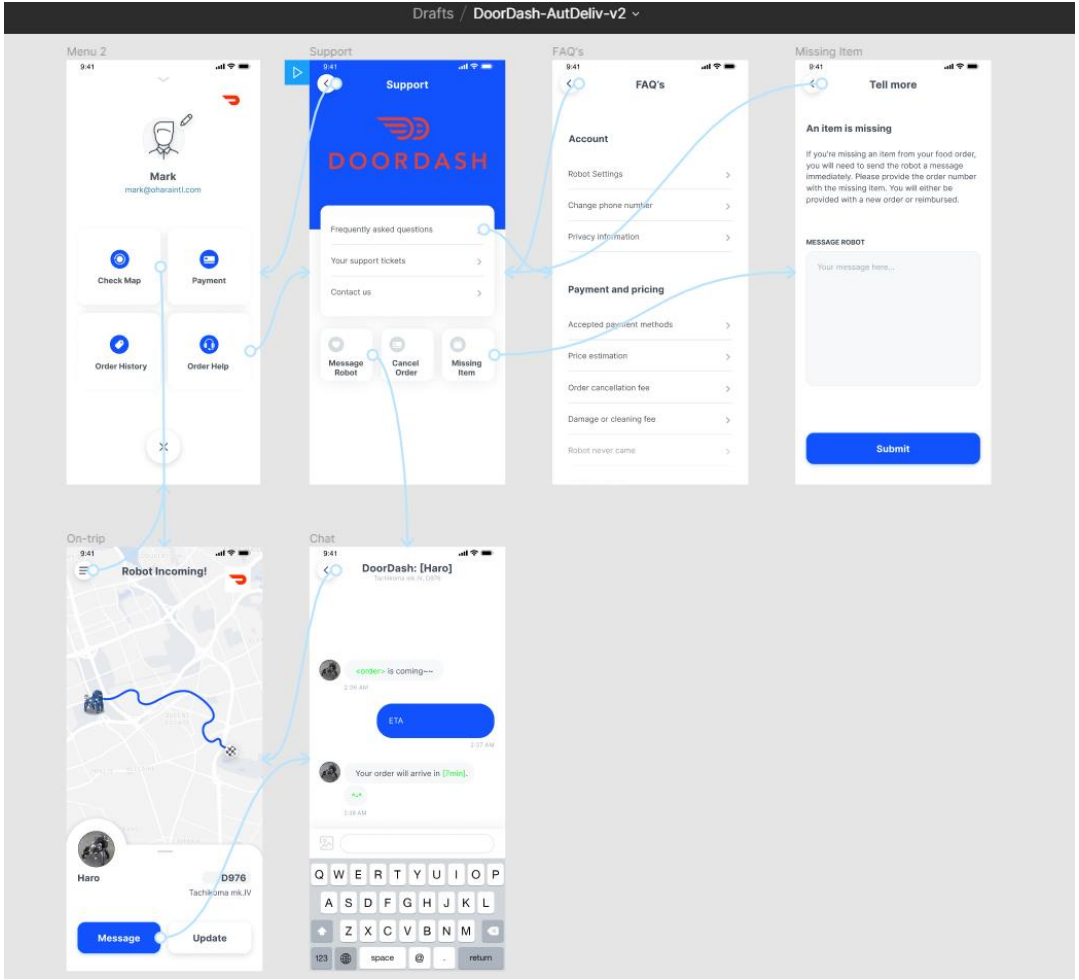
Leverage learnings from your first two user interviews to make changes to your prototype. Then run another round of user interviews.

Prototype v2

Description	The prototype is designed to take the user through two possible uses of the application. The user will check the location of the robot on the map and send them a “dummy message.” The user is also able to report a missing item.
Assumptions	Assumptions made regarding the prototype: • Users will not engage in a detailed walkthrough of the prototype • Users will have already signed-up/signed-in • Users can easily understand symbols
Tasks	Flows/tasks that a user can complete in the prototype: • Users can check the map/robot status • Users can send the robot a message • Users can report a missing item

Link to Prototype

Prototype v2



User Testing Round 2

Link to notes

Link to audio recording

Key Findings from Participant 3

What worked well	The application is easy to navigate, familiar, and contains all of the features needed for users.
Where participants got stuck	The user got stuck with a non-interactive “Update” button. Otherwise, there were no issues.
Other observations	The user compared the app to Postmates.

Handoff

Updated PRD

Link to PRD v2