CS 301: Guest Lecture Session

Adib Roumani
Director, Software Engineering @ Uber

A little about me

Adib Roumani
Director of Engineer, Mobility
Uber Technologies

- **Hometown**:
  - Current: Seattle, Washington
  - Past: Toronto (North York), Canada
- **Work**:
  - Currently: Director @ Uber
  - Past: Amazon, BlackBerry/RIM, A few Startups
- **What I do when not working**:
  - All things Sports!
    - Playing, Watching, etc
    - Huge hockey fan (Go Leafs Go) but also cheer for Seattle now too!
  - Watching stand-up comedy

My Journey into Tech

- **Childhood**
  - Always enjoyed sports, fast-paced shows and was always bored!
  - Imagined I would want to eventually have a job that would always change / never a dull day.
  - (never for once thought science / engineering would fulfill that!)

- **Early Influence**
  - Father (Hamzeh Roumani) is a Professor at York University for Computer Science
  - Learned to code at in middle school
  - Quickly found gratification in being able to think of something and then bring it to life!
    - Made little games / etc

My Journey into Tech

- **High School**
  - Given I was strong at coding I quickly took a lot of classes I could in high school and continued to learn and enjoy
- **University**
  - I was lucky in that I always knew I wanted to go into CS
  - Attended the University of Toronto, St George Campus
- **PEY (Professional Experience Year)**
  - Took a job at Research In Motion (Now BlackBerry)
  - This ***was the first time I really got industry experience***

Computer Science Industry vs School

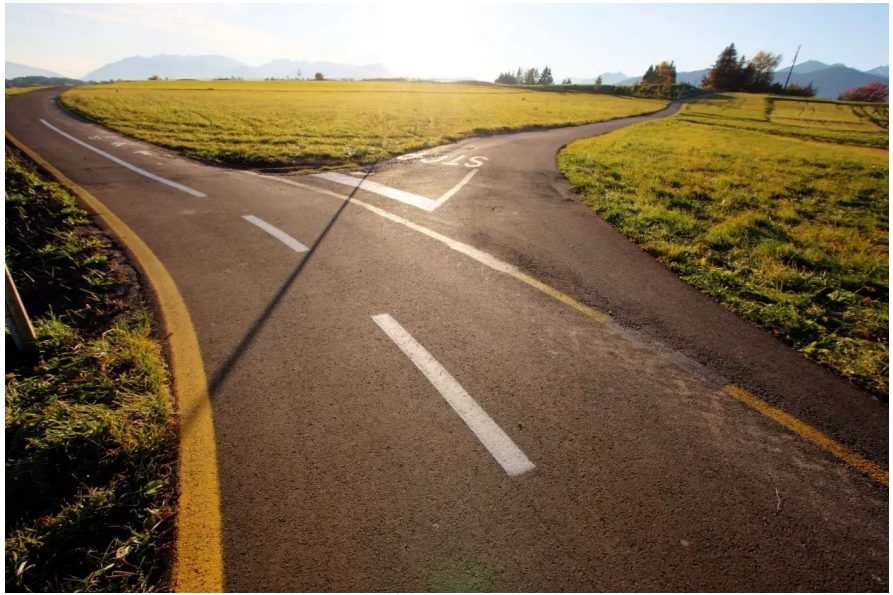
- **What you learn in school**
  - CS gives you all the fundamentals you need to then apply skills in the workforce.
    - Logic, Programing Acumen, Basic Coding Skills, Design Patterns, etc

- **What's different in the industry**
  - New Programing languages are ever emerging
  - New Concepts & Technologies come out every year

Think of your degree as a toolkit that gives you the skills not to do the job, but rather the tools to learn new skills to do the job.

- Most of what you use in school you will not use directly in the real-world
- Indirectly you will use all of the logic & problem solving skills you've garnered over the years of academics

My Journey into Tech



- **An Unexpected turn of events**
  - During my fourth year of school, I was living with my parents in North York
  - They lived in a condo with a gatehouse in the front and traditionally parking was complex as people would abuse the parking (since it was close to the subway)
  - The condo board created a rule that guests could only visit a unit 3 times in a rolling 30 day period
  - In reality this meant every car that would come, the security guard would have to check a log-book of the trailing 30 days to see how many times they would come

My Journey into Tech

- **Manual Checks meant congestion**
  - This congestion would spill onto a busy neighboring street causing even further backups

- **I overheard this issue and immediately thought there has got to be a “tech” solution!**

So… I started a company

compuscope CS consulting

Innovative Management Solution for your Condo

The Condominium Management Suite (CMS) is a fully automated database system designed to track and manage all levels of condominium operations in a secure environment. Not only will CMS increase communication between management teams, board members and residents, it will also improve your work efficiency and lower your administrative costs. CMS lets you avoid tedious and repetitive tasks by letting our software do the work for you.

Here is what CMS can offer your community:

Your Property Management Team Can:

- Store Property Information
- Search and Access Resident Information Quickly
- Create Custom Reports
- Receive Live Updates of Reports, Permits and Events

Your Concierge Staff Can:

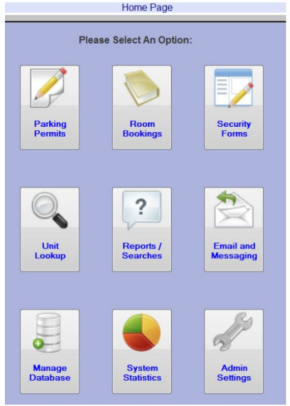
- Issue and Track Parking Permits
- Book Amenities and Rooms
- Create Incident Reports, Shift Reports, and Maintenance Logs
- Call Units Directly from the System

Your Residents Can:

- Read Announcements, News and Services
- Access Building Documents, Event Calendars, and Helpful Links
- Participate in Community Discussions and Forums
- Organize and View Special Events

Innovative Management Solution for your Condo

The Condominium Management Suite (CMS) is a user-friendly program that is easy to operate. As you can see from the screen shots below, any administrator or security personal can quickly book rooms, issue parking permits and access information in just a few clicks.



Fully automated reporting from Security and Maintenance personnel is available through shift reports, incident reports and maintenance logs. All information is stored in the system and is easily retrieved without fumbling through piles of paper. CMS will also automatically email these reports to any email account you desire.

Additionally, CMS stores numerous statistics on how the system is being used. Find interesting trends in permits, room bookings, etc. This information can be valuable in staffing and managing your property.

CMS can be customized to fit your specific needs!
Please contact us for a demo.

What I Considered

- **Clear User Problem**
  - Congestion
  - Manual Task
  - Phone number lookup
  - Manually filling out permit form

- **What / How Could I help**
  - Automation to solve all of the above
  - Database
  - Auto-Rule Checks on Licence Plates
  - Printing of permits

Problems I Hadn't Considered

- **No idea what tech to use**
  - Databases?
  - UX / UI Front Ends?

My PEY experience was invaluable to know what was being used in the real world

- **How do you print permits**
  - Smaller paper sizes?

- **How do you buy receipt paper at bulk?**
- **How much do I charge?**
- **How do I host the tech?**
  ...
  ...
- **But also.. How do you even start a company!?....**

While I had pretty much 0 answers, my schooling and industry experience combined gave my just enough experience to unblock myself, fail a bit and ultimately build and deliver the product.

But there's more...

If this worked so well for 1 condo...

Why couldn't I sell this to other condos in Toronto?

But there's more...

For the next 4 months I spent my days going to every single condo in the city.

And during the nights, I ended up coding more of the platform to make it work more generally.

And waited...

After 4 months...

And still no sales...

Every other condo agreed it was a major issue. But a few major gotchas:

1. A Kid trying to sell software door-to-door is tough
2. Decision makers at condos were “board” members that tended to be older residents less excited about change

A Chance Encounter

- **Met with a security vendor**
  - They were looking to bundle a solution like this with their offering

- **This ultimately solved all major problems**
  - A “Kid” selling tech was no longer the sale - it was security: something all condos already paid for

This partnership had my company go from 1 condo where we started to over 20 in 12 months.

Until recently: On any given day in Toronto, every 2.8 mins a parking permit is generated from this solution!

What's the lesson here?

1. Tech Solutions and being able to code is only one small part of the problem.

2. Understanding the customer and what they want is just as important.

3. Being able to learn & speak to customers & users of your products is critical to success

As you work through new ideas / projects keep in mind - it's the end customers that needs to "buy-in" and ultimately make your idea successful

My Journey into Tech

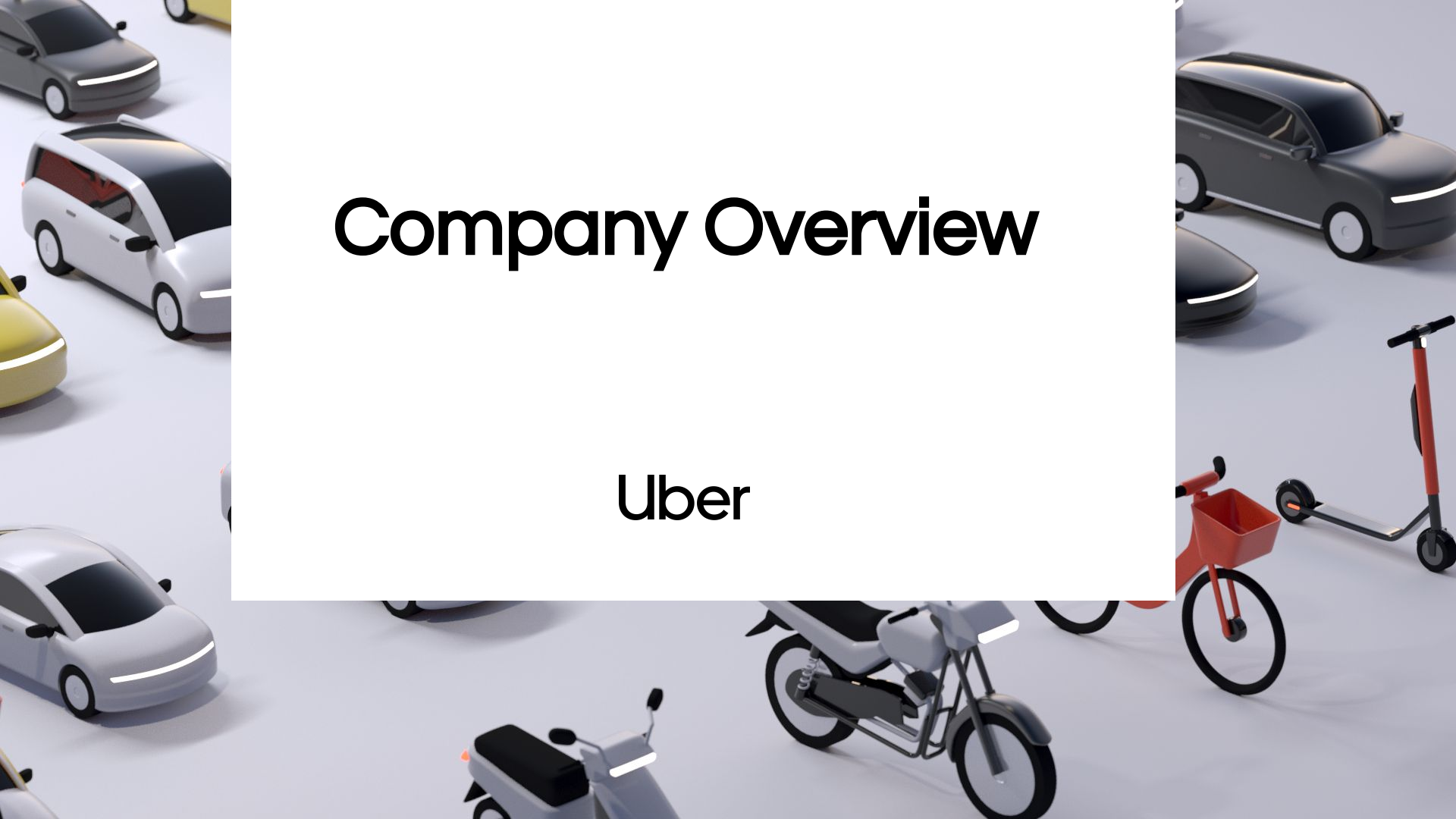
- **BlackBerry**
  - Spent 1 year as a Software Engineer
  - Spent 3 years as Tech Lead / Manager
- **Amazon (Seattle)**
  - Moved to Seattle and led teams in management roles for 4.5 years
- **Uber (Seattle)**
  - Director of Engineering for Mobility

Why did I go into Management?

Scale & Impact

a. Only so many hours in a day
b. Either through influence indirectly or directly

There are many job / career path for CS students in the industry, very common to assume all CS students code.

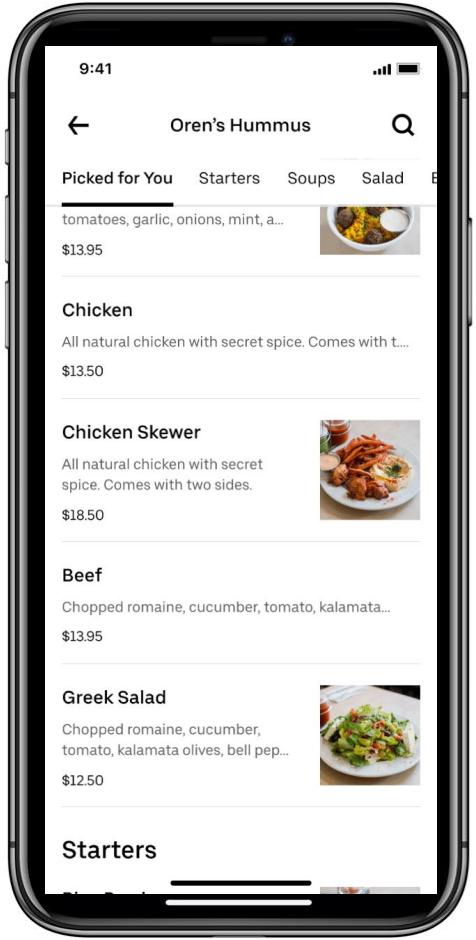


Company Overview

Uber


Uber

And our story is just beginning...

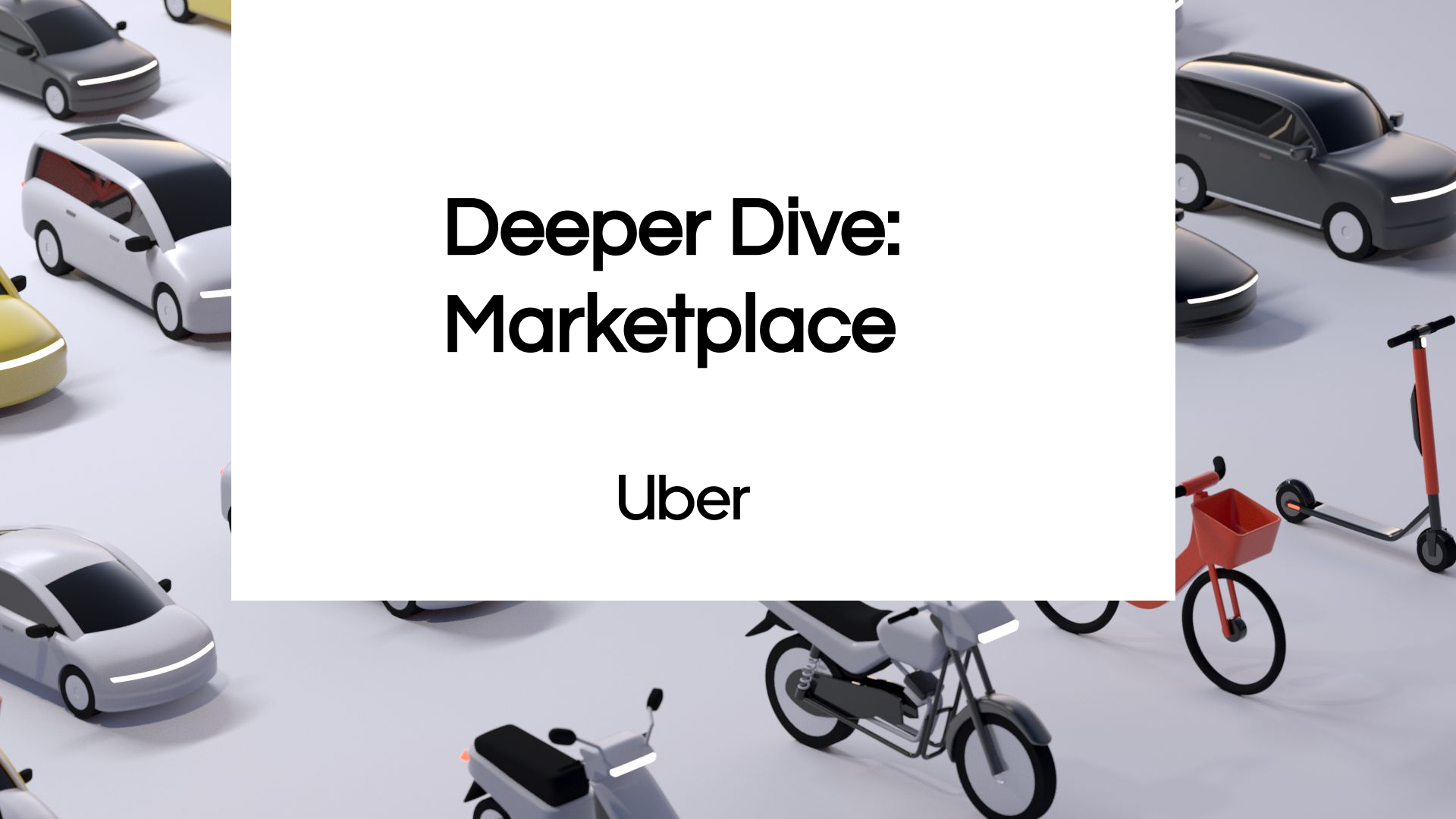

9:41
Oren's Hummus
Picked for You
Starters
Soups
Salad
tomatoes, garlic, onions, mint, a...
$13.95
Chicken
All natural chicken with secret spice. Comes with t....
$13.50
Chicken Skewer
All natural chicken with secret spice. Comes with two sides.
$18.50
Beef
Chopped romaine, cucumber, tomato, kalamata...
$13.95
Greek Salad
Chopped romaine, cucumber, tomato, kalamata olives, bell pep...
$12.50
Starters

By growing verticals like Rides/Mobility & Eats/Delivery we scale better!

Cross Dispatch

Hardware Scale / Cost

Tech & Innovation



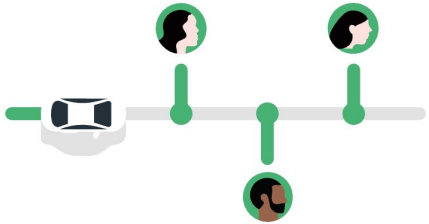
Deeper Dive: Marketplace

Uber

Create the most efficient marketplace to sustainably make Uber the platform of choice for riders and drivers.

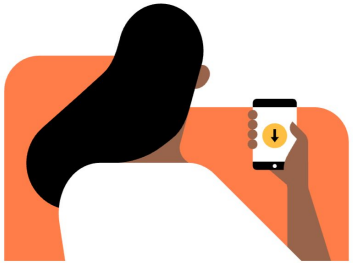
Marketplace Mission

Marketplace Efficiency Tradeoff



Hypothetical 1: 2X drivers overnight to the Uber Platform

- Improved ETAs for Riders, less surge (increased pricing when we don't have enough drivers to fulfill requests)
- Lower driver earnings / hour, risk over time that drivers will leave the platform



Hypothetical 2: 2X riders overnight to the Uber Platform

- More earnings for Drivers / hour
- Longer ETAs for Riders, risk they will use a competitor's product, more surge

Marketplace efficiency is the achieved by putting the right rider and driver incentives and features in place in real-time

Upfront Pricing

There are numerous factors and considerations that need to be applied when computing the price a rider sees:

The Route (Uber Mapping teams + tech to generate a route with Time & Distance estimates)

Using this time + distance we can come up with a base fare (Time * Time Rate + Distance * Distance Rate)

But wait, there's more...

- **Uber Booking Fees**
- **Tolls (bridges, highways, etc)**
- **Taxes**
- **Surcharges (airports, city rules: congestion fee)**
- **Promotions & Discounts**
- **And more...**

Fares needs to ensure that all of these are calculated into the price you see, and be able to compute these instantly when customers request a ride

Multi-Modal Trips:

The intersection of Matching, Pricing & Fares

New York City → JFK can take upwards of 2 hours at busy times

We had an idea: lets fly! (in <30 min)

- We launched Uber Copter as a premium offering to get from Downtown to JFK for ~$200 (context an Uber Black can cost ~120 at these times)

- Three legs in these trips that all require Fare calculations and payments to various Drivers / Fleets involved:

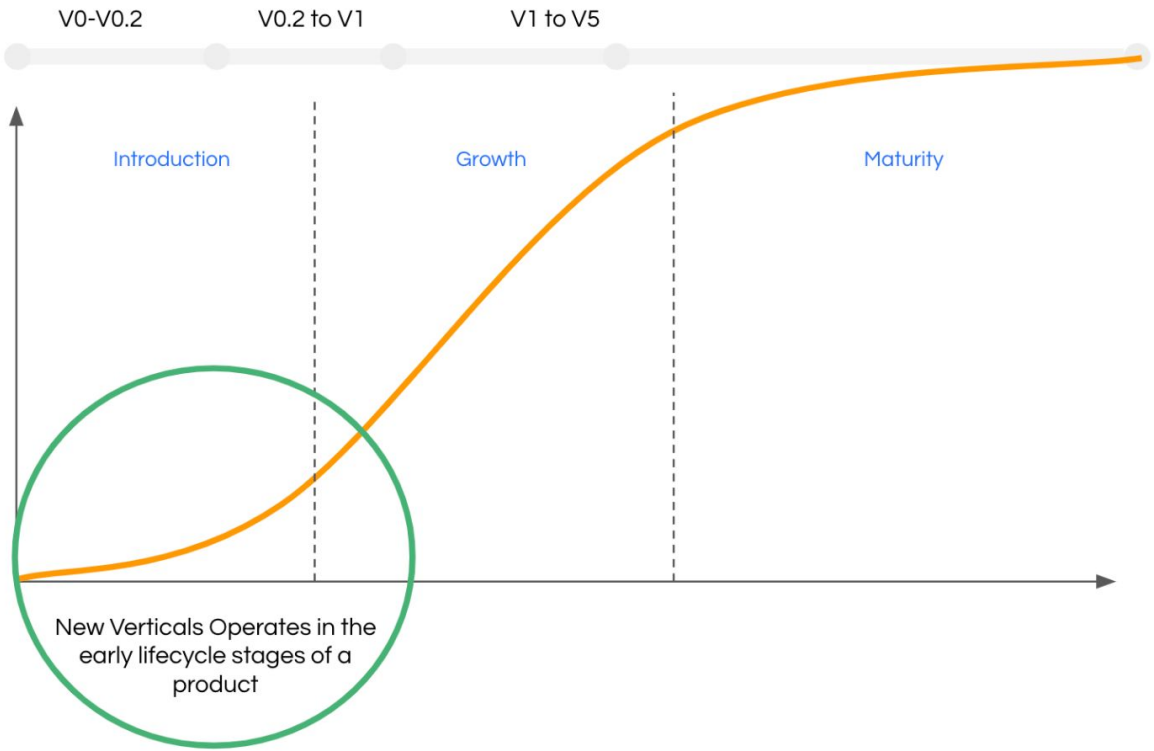
  - [Uber Black] A → B
  - [Uber Air] B → C
  - [Uber Black] C → D

- Other complexities:
  - ***Scheduling Rides in advance***: so the drivers are ready and in place on all legs of the trips (How to price these trips that will happen in the future)
  - All of our **systems need to be flexible & generic** such that we can price out products that are not assumed to be rides

How to innovate? Choose a great project? And build fast?
Pilots & MVPs is how companies prove solutions quickly at scale.

New Verticals Product Life Cycle

Test, Learn & Iterate



Pilot = Idea that needs validation

What are the objectives of a Pilot?

- Be at frontier of finding signals to grow Uber's trajectory in 3-10 years
- Validate Pilots & de-risk bets
- Test, Learn and Iterate

Why is testing Pilots important?

It's about determining WHICH problems to solve and invest

- *Uber is about investing a variable amount of time to get a fixed outcome.*
- *Pilots is about investing a fixed amount of time with a variable outcome.*
- *Speed is key to unlock & determine long run bets.*
- *Example: Intercity is not the right Pilot to test as the PMF is known. The outcome of a sample Pilot would be unknown.*

Objectives of Pilots

Be at frontier of finding signals to grow Uber's trajectory in 3-10 years

Broaden core services and ways to earn.

Validate Pilots & de-risk bets

Validate before stand up a dedicated tech team

Test, Learn and Iterate

Build for learning, not for tech scale

*As a **consumer**, if you had to spend $500 on Uber a month, what would you want?*
*As an **earner**, how can I find more ways to earn? Or taking this even further, start a business on Uber?*

Team Operating Advantages

Cheaper Tech

Can test ideas out faster by for eg focusing on web

Funding

Ops driven to check business opportunities that need validation on product-market fit

Culture / Speed

Cranking out verticals within a quarter is a muscle that we need to build and nail. Once nailed, share learnings to help operate faster.

As you think through your projects ask yourself some key questions:

1. What is the total addressable market (TAM) your solution is going after?

2. How much of that opportunity will your project capture?

3. Are there elegant ways to reduce your large deliverable into smaller ones that incrementally show value?

4. [In Industry] How do you track your work? Ensure you are hitting timelines? What about when you are working with others and there are dependencies?
    a. In many cases by cleverly breaking down problems into sub-parts, especially when those parts are disparate it's cleaner to work truly in parallel.

Why Communication is so KEY?

- Overtime, especially as you grow in a company the Software Engineering skills do not separate the good from the great.
- It's the ability to work with others, influence others and drive impact beyond just yourself
- In larger companies you oftentimes need to work with more teams, many of which are not no your "team", across time-zones, and even across many job functions.
  - Concise.
  - Crisp.
  - Confident
- The more sr you are, the less you will be coding, but the more you will be designing and influencing. Communication is also written communication.

Common Misconception
about CS jobs

Jobs in our field are not always just about coding...

Product Management

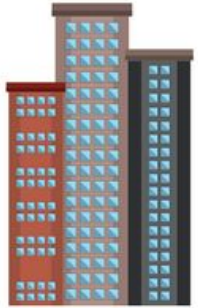
Technical Program Management

Engineering Manager

Startup

VS

Big Company



What is better for my career?

What do I look for in a Software Engineer?

- ➔ Ambition, bias-for-action, eager to learn (I can't teach this)
- ➔ Confident to ask & say when unsure
- ➔ Logic, logic, logic… & problem solving
- ➔ Ability to clearly articulate reasoning
- ➔ Collaboration with others
- ➔ Business / Product intuition
- ➔ Ability to sway / influence others for the better

What do folks in higher levels do?

➔ Engineers execute & architect

➔ Managers think about career + team while ensuring eng quality is high (1-2 year horizon)

➔ Directors think about groups of teams & longer term strategy (3+ year horizon)

Every level looks to hold the one under accountable and effective.

Within engineering there is a scale too, whereby the more sr the less coding and the more influence & design

What types of projects should I take on? Breadth vs Depth*

**intern programs may force into depth*

There are some jobs that require deep domain knowledge in a particular domain taking years to master: ML Expertise, Database Building, many platforms /etc. (Think Google Mapping)

There are other jobs that focus more on using the collection of many platforms to build game changing products (Think CLEAR)

Closing Thoughts

You've already done the hard part in choosing CS as a career path! The industry despite current headwinds will power the next generation of continued innovation.

Tech continues to be greatest disruptive innovation function of all industries around the world.

Absorb & learn as much as possible - and keep doing so - the exact programing language / solution you've learned the past 4 years more than likely will be replaced in the next 5 years with something newer, continue to learn and re-learn throughout your career

While it's critical you master Software Engineering - overtime it's your ability to drive changes larger than just you can pull off - learn how to effectively communicate, convince, and influence others.