Before lecture

1. How are you doing?
2. Let’s take a look at the guide
   a. Diagrams are the must
   b. Everything else is just for your convenience
3. Tomorrow 9-10 you present first page of your draft. And then 12-13.

2024

Solution Design

ML Product Design

Alexander Guschin
Mikhail Rozhkov

OUTLINE

1. Solution and UI/UX
2. Performance Metrics & Validation
3. Project Assignment

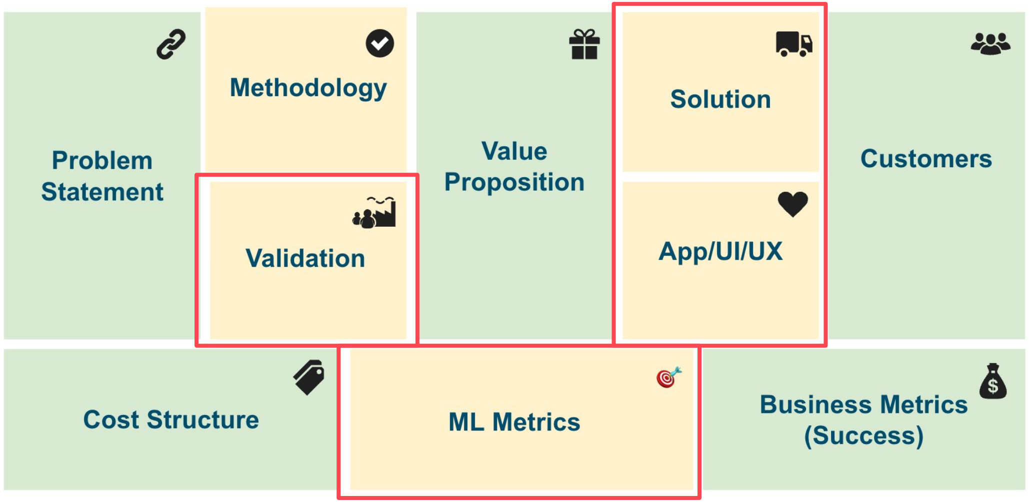
3.x – ML Product Design: Solution

Purpose

- Frame the ML solution

Guiding questions:

- How does the ML coild help?



MODULE DETAILS

Goals

- Recognize the importance of user-centric design in ML solutions
- Learn to translate business requirements into technical specifications
- Understand how to set realistic performance targets for ML systems

Learning Outcomes

- Design user-friendly ML interfaces
- Define key performance indicators
- Develop effective validation strategies
- Draft high-level ML system overviews

HANDS-ON ACTIVITIES

Exercise

- Research possible solutions for the Business Problem, including papers, reports, blog posts, conference talks, etc.
- Goal: Understand requirements for ML Product

Output

1. Drafting the first part of a design document: **Solution Design**
    a. High-Level Solution Overview
    b. User Interface and Experience Design
    c. Performance Metrics
    d. Validation Strategy and Pilot Project Plan
    e. Requirements & Constraints

Solution and UI/UX

ML Product Design

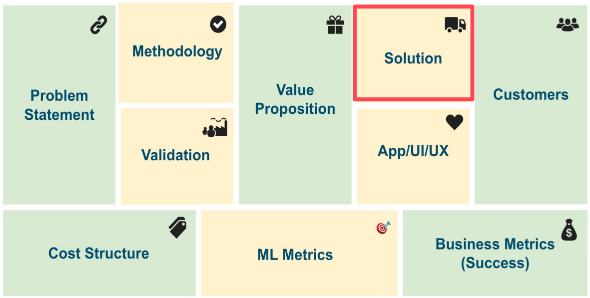
- Guide: 3.1 - High-Level Solution Overview
- Guide: 3.2 - User Interface and Experience Design

3.1 - High-Level Solution Overview

How does the solution look for our customers?

Purpose

- To give all stakeholders a clear understanding of the overall system architecture and components



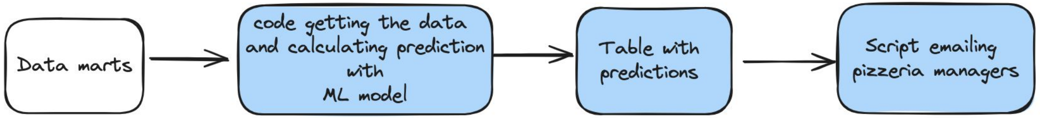
Guiding questions:

- How the predictions should look like for a consumer?
- What are the main components of our ML system?
- How do these components interact with each other?
- How will your system integrate with upstream data (what data we'll pass to the model?) and downstream users (how users will access predictions?)

Case: NewPizza - long waiting time

How does the solution look for our customers?

We're getting the data from data marts, calculating predictions and writing them to a table in the database. Then we send emails to pizzeria managers with our predictions.



Key points:

- **Predicted target:** peak times predicted week ahead on hourly basis
- **Prediction format:** a table with [pizzeria, date, hour, peak_probability] columns
- **Components:** Data marts, ML model.
- **Integration:** We send the table with prediction on pizzeria managers email

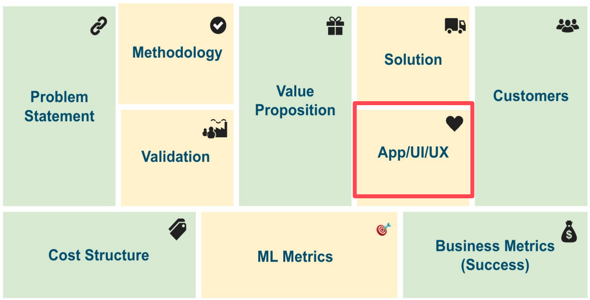
Example

3.2 – User Interface and Experience Design

How will users interact with the ML system?

Purpose

- To ensure the ML system is user-friendly and integrates well with existing workflows.



Guiding questions:

- How will users interact with the ML system?
- What changes to existing workflows are required?
- How can we make the system intuitive and user-friendly?
- How will you incorporate human intervention into your ML system (e.g., product/customer exclusion lists)?

Case: NewPizza - long waiting time

How will users interact with the ML system?

We're getting the data from data marts, calculating predictions and writing them to a table in the database. Then we send emails to pizzeria managers with our predictions.

pizzeria	datetime	peak_probability
bcn-1	2024-08-01 14:00	0.95
val-2	2024-08-01 12:00	0.83

Key points:

- UI: a google sheet
- UX: opening a google sheet, seeing schedule for the next week with highlighted hours that's are predicted to be peak, replanning staff schedule.
- Consider: how to illustrate updated predictions (it was predicted as peak yesterday, but it's not today, or vise versa)

Example

Exercise: Solution and UI/UX

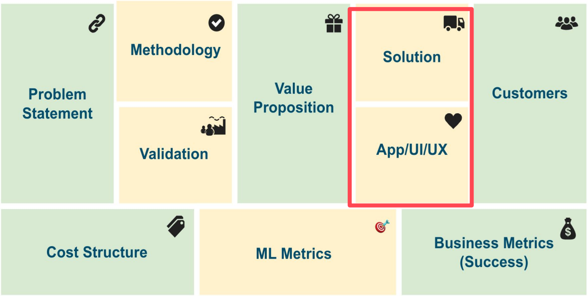
How does the solution look for our customers?

Group task:

- Brainstorm and complete the ML Product Design sections:
  - 3.1 - High-Level Solution Overview
  - 3.2 - User Interface and Experience Design
- 10 min

Key points:

- Format of the predictions
- Key components of the ML system
- Integration with other systems
- User Interface (UI)
- UX considerations



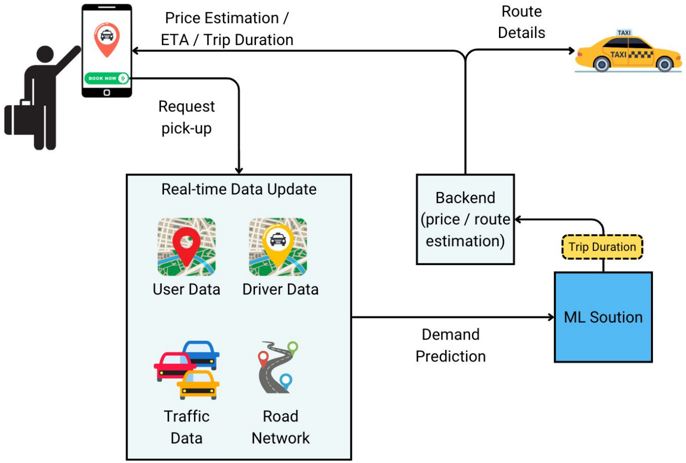
Practice

3.1 - High-Level Solution Overview

How does the solution look for our customers?

Overview:

- The proposed ML system will predict trip durations based on features such as pickup location, planned dropoff location, trip distance, time of day, and day of the week. Predictions will be used to calculate trip prices shown in the app.



Key points:

- Target: trip duration in minutes
- Format: a float number
- Components: Taxi App, Data Ingestion, ML Solution (prediction service), Backend
- Integration: with app backend for real-time predictions.

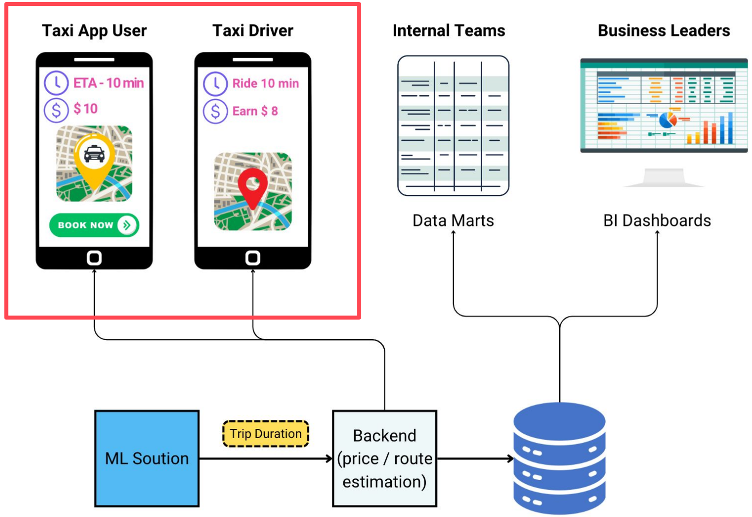
Example

3.2 – User Interface and UX Design

How will users interact with the ML system?

Overview:

- Taxi App Users and Drivers interact with the system through the EasyRide app, which displays the predicted trip duration and cost. The UI design ensures transparency and ease of understanding for users.



Key points:

- UI: ETA, trip duration and cost/earnings.
- UX: Requesting a ride, viewing the predicted cost, booking the ride.

Example

ML Product Design: EasyRide Taxi

Problem Statement

- High MAPE > 30%.
- External provider's prediction service (we can’t improve).
- Competitive market with accurate pricing as a differentiator.
- Critical to EasyRide's strategy of superior customer service.

Methodology

-
-

Validation

-
-

Value Proposition

- Minimize revenue loss
- Improve customer retention
- Improve driver retention

Solution

- Target: trip duration in minutes
- Format: a float number
- Components: Taxi App, Data Ingestion, ML Solution, Backend

App/UI/UX

- UI: ETA, trip duration and cost/earnings.
- UX: Requesting a ride, viewing the predicted cost, booking the ride.

Customers

- Taxi app customers
- Taxi drivers

Cost Structure

- Initial Development: $214,083
- Annual Operations: $386,000
- Annual Benefit: $15,147,500
- ROI: First Year: 2,424% / Subsequent Years: 3,825%

Performance / ML Metrics

-
-

Business Metrics (Success)

- Daily Revenue Increase by $24,000
- Pricing Loss Reduction by $17,000
- Booking Rate Improvement by 6%
- Evaluate metrics daily, with quarterly reviews

Performance Metrics & Validation

ML Product Design

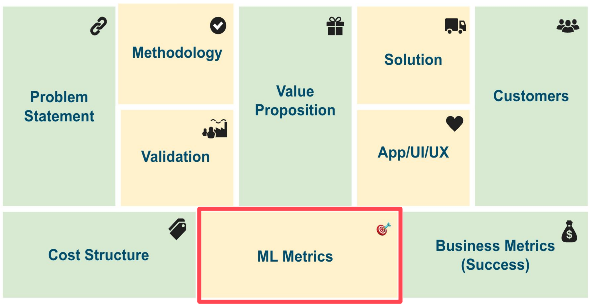
- Guide: 3.3 - Performance Metrics
- Guide: 3.4 - Validation Strategy and Pilot Project
- Guide: 3.5 - Requirements & Constraints

3.3 – Performance Metrics

How do we measure our solution performance?

Purpose

- To establish clear criteria for evaluating the system's technical performance and business impact



Guiding questions:

- How will we know that the solution is successful?

Case: NewPizza - long waiting time

How do we measure our solution performance?

For ML metrics, it make sense to start with Precision/Recall.

For Business metric, we may go for avg waiting times, or % of hours with big delays.

Note that ML metrics improvements may not result in business metric improvement, even if expected. The problem may be the actual application of ML predictions. However good they are, incorrect/missing reaction to them may keep business metrics stale. Because of that, it's important to monitor both.

Key points:

- Precision 90% at Recall 50%
- Business metrics: avg waiting times decrease, % of waiting times > 5min is <10%
- Timeline: week of evaluation

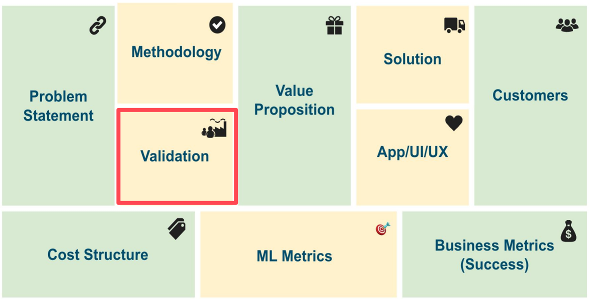
Example

3.4 – Validation Strategy and Pilot Project

How do we validate the solution performance?

Purpose

- To ensure the system performs as expected and delivers value before full-scale deployment.
- Defining a Successful Pilot: The formalized metrics for assessing the success of the pilot will be defined by the Product Owner.



Guiding questions:

- Is the pilot helpful for our problem?
- What does a successful pilot look like? Regardless of ML model used under the hood.
- How will we validate the ML system's performance? How to validate the solution works in production process (system)?
- If you're A/B testing, how will you assign treatment and control (e.g., customer vs. session-based) and what metrics will you measure? What are the success metrics?

Hint:
- Can we design a validation strategy that general enough to test any “black box” solution? (including non-ML)

Case: NewPizza - long waiting time

How do we validate the solution performance?

The validation strategy involves a pilot phase with switchback experiments where we measure impact on chosen business metrics.

See the key points for the rest of the information.

Question: can we get the results faster? How?

Key points:

- Validation Methodology: A/B testing, switchback experiments
- Pilot Scope: 2-4 weeks
- Success Criteria: prediction is adopted and % of waiting times > 5min is <10%

Example

3.5 - Requirements & Constraints (optional)

What are the requirements and constraints should we care about?

Purpose

- To clearly define the system's capabilities and performance standards

Guiding questions:

- What specific functions must the system perform?
- What are the performance, security, regulatory and scalability requirements?
- What's in-scope & out-of-scope? Some problems are too big to solve all at once. Be clear about what's out of scope.
- Corner cases: What’s the worst that can happen if the model is wrong only once but for a very important data point? Are all data points equally important?
- Risks: What are the major risks you are facing? What are you doing to mitigate them? Some examples: are you doing some bleeding-edge research? do you depend on a major infrastructure component that is yet to be built?

Case: NewPizza - long waiting time

What are the requirements and constraints should we care about?

Non-functional Requirements:

- Maintain accuracy across various conditions (weekdays, weekends, state holidays)
- Have probability calibrated (or maybe we should email not with probability, but something else?)
- Emali should be delivered at 7 am daily

Constraints:

- Implementation cost: $50,000
- Operational cost: 10% time of 1 Data Scientist and 1 ML Engineer per month

Example

Exercise: Performance Metrics & Validation

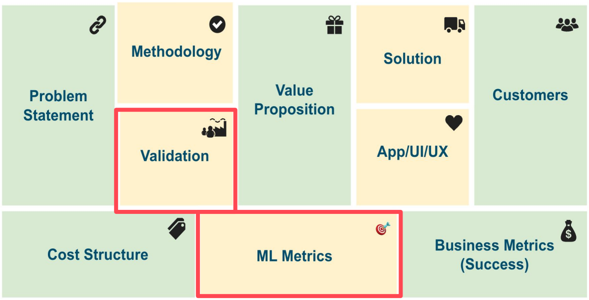
How does the solution look for our customers?

Group task:

- Brainstorm and complete the ML Product Design sections:
  - 3.3 - Performance Metrics
  - 3.4 - Validation Strategy and Pilot Project
- 10 min

Key points:

- Technical metrics (e.g., accuracy, latency)
- Metrics we calculate for model predictions (offline)
- Business metrics (online)
- Validation methodology
- Pilot project scope and timeline
- Success criteria for moving to production



Practice

3.3 – Performance Metrics

How do we measure our solution performance?

Overview:

- Performance will be measured using technical and business metrics, including MAPE, prediction latency, booking rates, and overall revenue impact.

Key points:

- Prediction accuracy (MAPE) < 15%
- Prediction latency < 100ms
- Business Metrics: Booking rate, Daily Revenue Increase
- Timeline: Daily metric evaluation.

Example

3.4 – Validation Strategy and Pilot

How do we validate the solution performance?

Overview:

- The validation strategy involves a pilot phase where the model runs in shadow mode, comparing its predictions with the current provider. Success is measured by improved MAPE and booking rates.
- Pilot Scope:
  - 1-week test period
- Success Criteria:
  - MAPE reduction to 15% or lower
  - Demonstrate potential for $17,500 daily pricing benefit
  - Show capability to increase booking rate to 94%

Key points:

- Validation Methodology: Shadow deployment, A/B testing.
- Pilot Scope: 1 week shadow deployment.
- Success Criteria: Lower MAPE, higher booking rates.

Example

3.5 - Requirements & Constraints

What are the requirements and constraints should we care about?

Overview:

- The system must predict trip durations accurately, integrate with existing infrastructure, and meet performance, scalability, and security requirements.

Non-functional Requirements:

- Handle 10,000 trips per day
- Improve idle time by 10%
- Maintain accuracy across various pricing scenarios (over/underpricing)

Constraints:

- Implementation cost: $100,000
- Operational cost: 10% time of 1 Data Scientist and 1 ML Engineer per month
- Must work within current pricing structure ($1 per minute, $20 average fare)
- Comply with existing driver compensation model

Example

ML Product Design: EasyRide Taxi

Problem Statement

- High MAPE > 30%.
- External provider's prediction service (we can't improve).
- Competitive market with accurate pricing as a differentiator.
- Critical to EasyRide's strategy of superior customer service.

Methodology

-
-

Validation

- Validation Methodology: Shadow deployment, A/B testing.
- Pilot Scope: 1 week
- Success Criteria: Lower MAPE, higher booking rates.

Value Proposition

- Minimize revenue loss
- Improve customer retention
- Improve driver retention

Solution

- Target: trip duration in minutes
- Format: a float number
- Components: Taxi App, Data Ingestion, ML Solution, Backend

App/UI/UX

- UI: ETA, trip duration and cost/earnings.
- UX: Requesting a ride, viewing the predicted cost, booking the ride.

Customers

- Taxi app customers
- Taxi drivers

Cost Structure

- Initial Development: $214,083
- Annual Operations: $386,000
- Annual Benefit: $15,147,500
- ROI: First Year: 2,424% / Subsequent Years: 3,825%

Performance / ML Metrics

- Prediction accuracy (MAPE) < 15%
- Prediction latency < 100ms
- Business Metrics: Booking rate, Revenue Increase
- Timeline: Daily metric evaluation.

Business Metrics (Success)

- Daily Revenue Increase by $24,000
- Pricing Loss Reduction by $17,000
- Booking Rate Improvement by 6%
- Evaluate metrics daily, with quarterly reviews

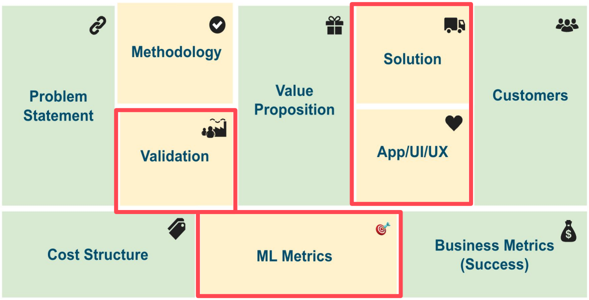
Project Assignment

Start with ML System Design!

Practice: Design a Solution (non-ML)

How could the solution look like for the customer?

- Describe a solution for your project
- Follow the guide to describe each section
- Summarise Solution on the canvas
- Update Business Understanding (Cost Structure) if needed



Materials & Links

- Course Materials: Google Drive
- Practice - EasyRide Taxi - Day 2 - PUBLIC
- Guide - ML System Design - Canvas