MVP & Functionality

Presented by Nursultan Askarbekuly

Course flow

- Explore domain and meet customers
- Formulate the requirements - **we are here!**
- Prototype and get feedback
- Explore architecture and notation
- Deliver an MVP
- Iterate to improve the product

Dealing with the customers

- Establish regular meetings
- Show what you have to get feedback
- Concentrate on what and why, not how.
- Be tactful and professional
- The expected workload is 10 hours per team member
- Each customer will pick their favorite team, and we’ll reward the team extra 10 points.

What features will your product/service have?

A minimum viable product

is a version of a product

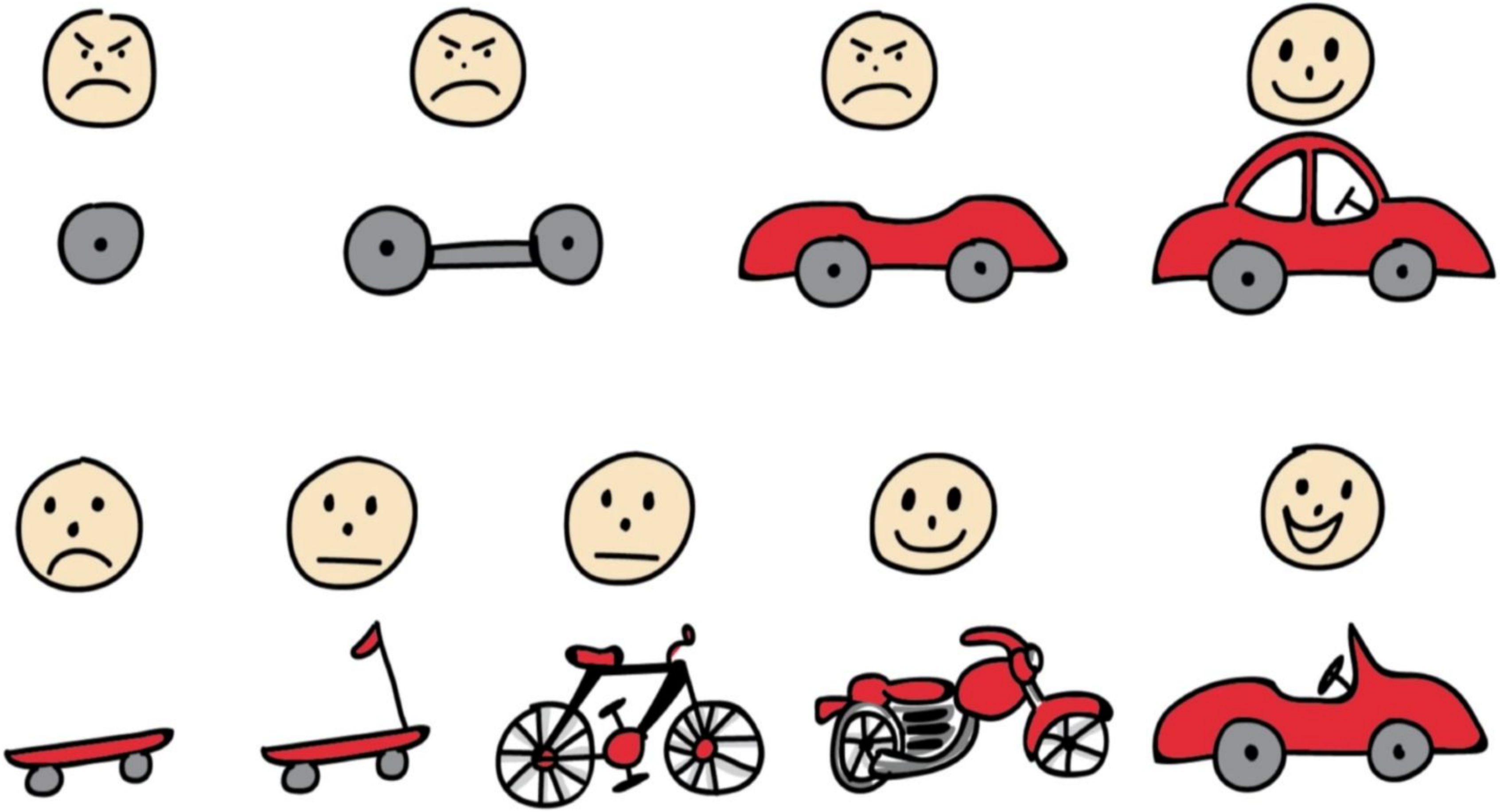
with **just enough** features **to be usable**

by the customer who can then provide feedback for future development.

MVP: https://en.wikipedia.org/wiki/Minimum_viable_product

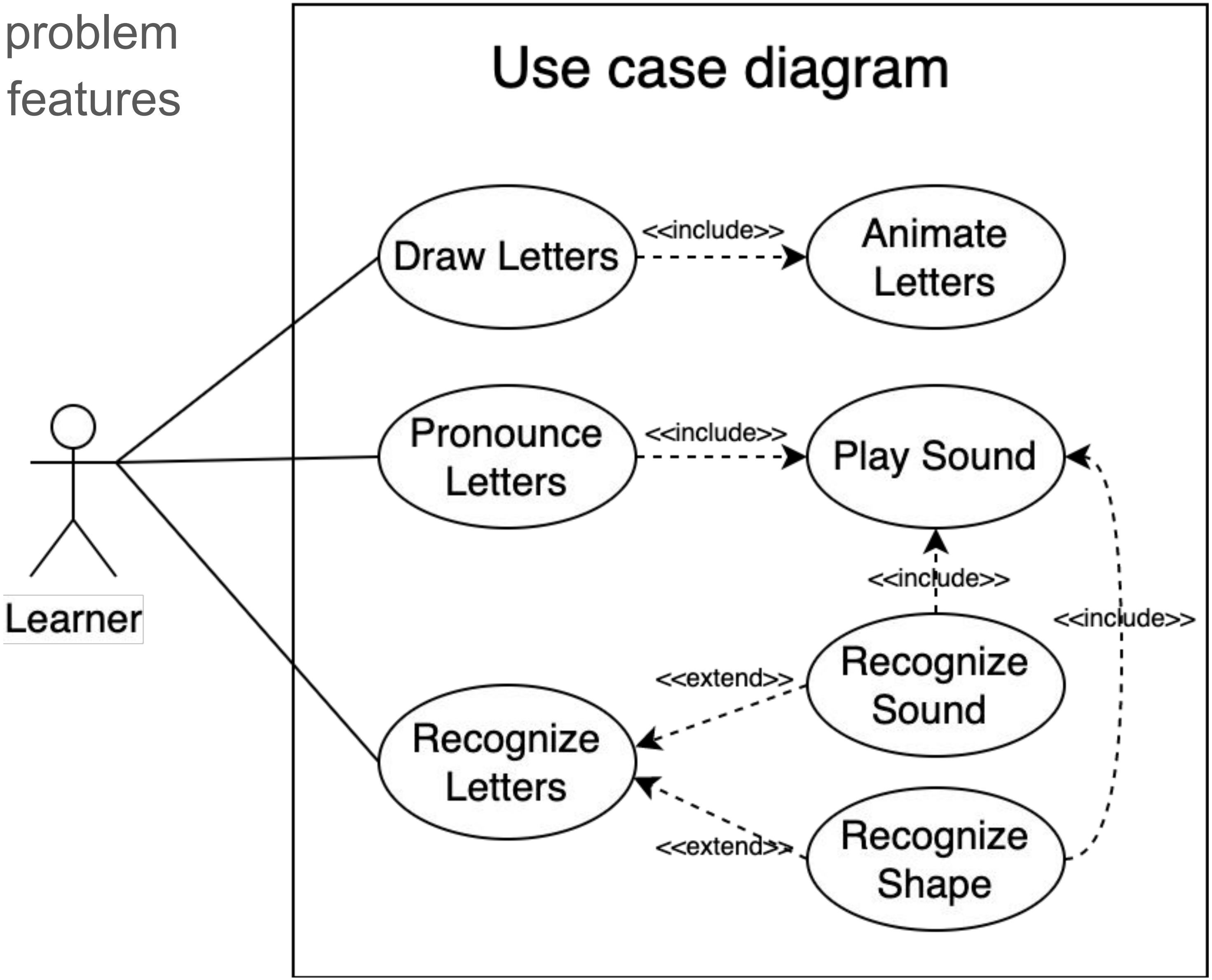
What is the simplest possible version of my product that will have value for my user?

Problem: Help users get from point A to point B.



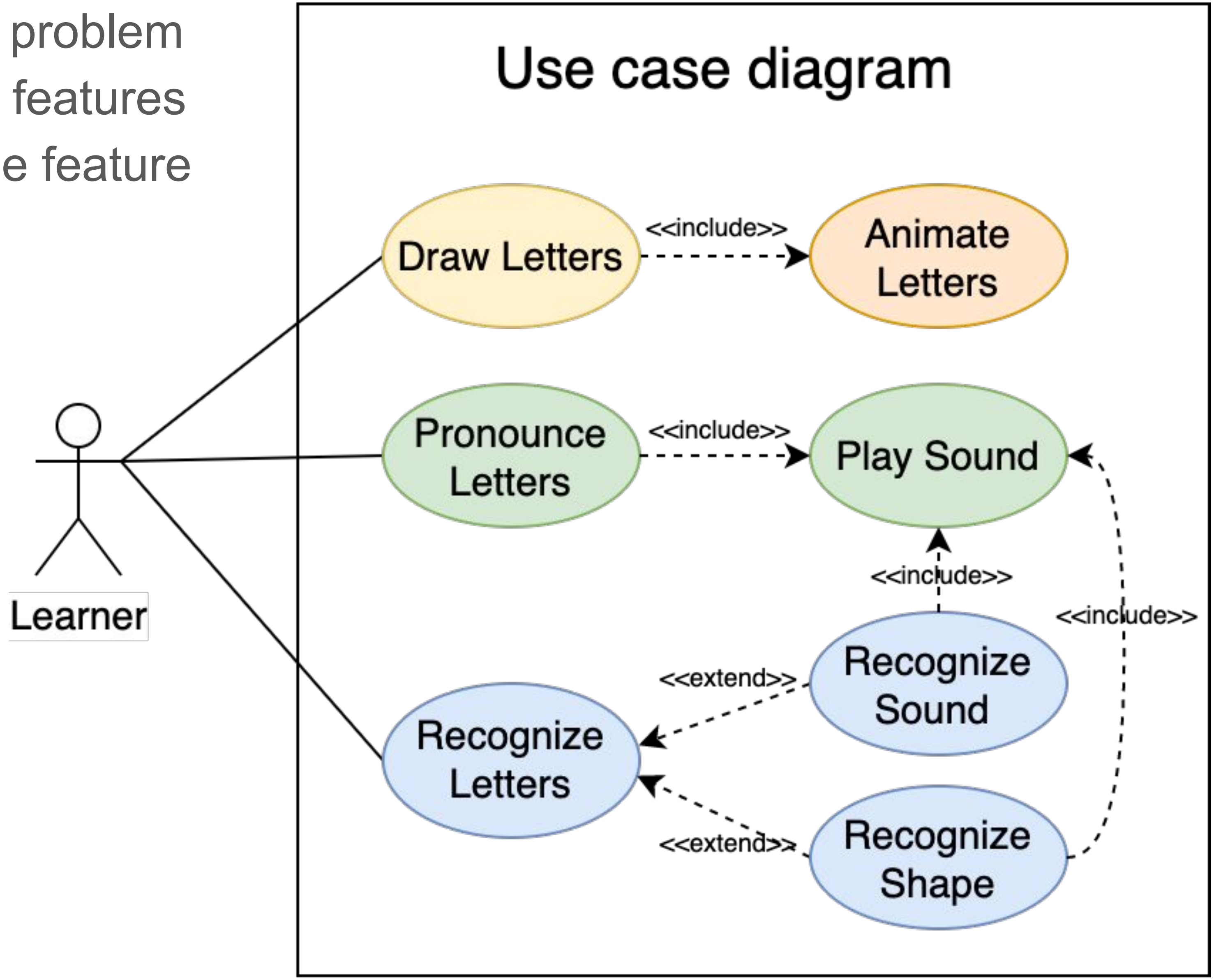
Breaking down the functionality into phases

- Narrow the problem
- Specify the features



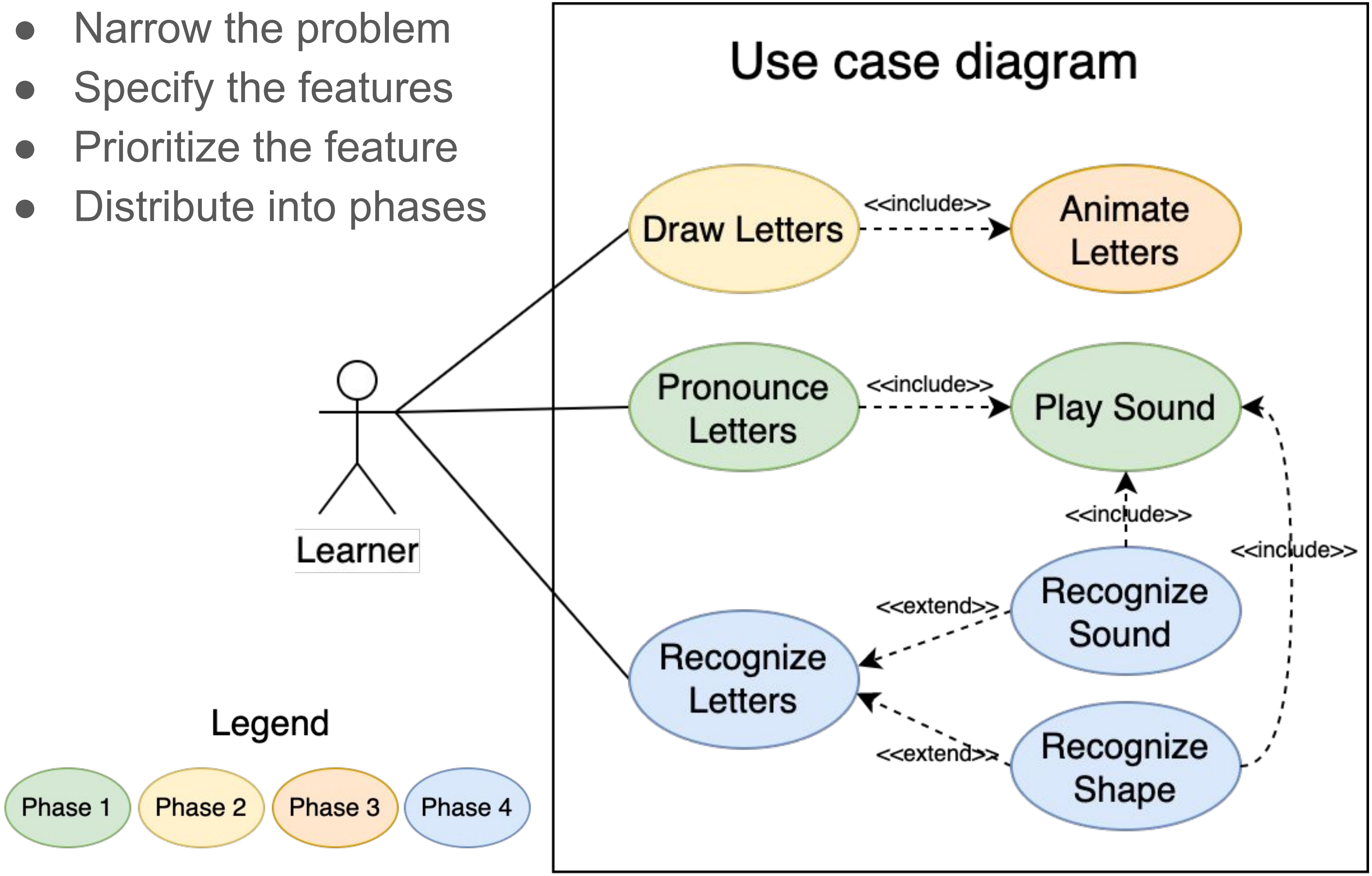
Breaking down the functionality into phases

- Narrow the problem
- Specify the features
- Prioritize the feature



Breaking down the functionality into phases

- Narrow the problem
- Specify the features
- Prioritize the feature
- Distribute into phases



What is the simplest version of my product that I can give to the customer to try?

Specify the features your app will have

- Create a use case diagram
- Prioritize and distribute into phases
- Arrive at a clear MVP feature set

MVP: https://en.wikipedia.org/wiki/Minimum_viable_product

Resources

Minimum Viable Product:
https://en.wikipedia.org/wiki/Minimum_viable_product

Use case diagram:
https://www.lucidchart.com/pages/uml-use-case-diagram