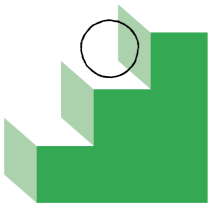
User Needs + Defining Success

Chapter worksheet



Instructions

Block out time to get as many cross-functional leads as possible together in a room to work through these exercises & checklists.

Exercises

1. Evidence of user need [multiple sessions]

Gather existing research and make a case for using AI to solve your user need.

2. Augmentation versus automation [multiple sessions]

Conduct user research to understand attitudes around automation versus augmentation.

3. Design your reward function [~1 hour]

Weigh the trade offs between precision and recall for the user experience.

4. Define success criteria [~1 hour]

Agree on how to measure if your feature is working or not, and consider the second order effects.

1. Evidence of user need

Before diving into whether or not to use AI, your team should gather user research detailing the problem you're trying to solve. The person in charge of user research should aggregate existing evidence for the team to reference in the subsequent exercises.

User research summary

List out the existing evidence you have supporting your user need. Add more rows as needed.

Date	Source	Summary of findings
2024	**Lopez et al.** — *"The Development of a Prediction Model Related to Food Loss and Waste in Consumer Segments of the Agri-food Chain Using Machine Learning Methods."*	Users often waste food due to poor inventory tracking. Machine learning-based forecasting can significantly improve household food management and reduce waste.
2021	**Zhou et al.** — *"Personalized Food Recommendation as Constrained Question Answering over a Large-scale Food Knowledge Graph."*	Recipe recommendations that consider user constraints (time, budget, allergies) lead to higher satisfaction and better personalization than generic popularity-based models.
2024	**Tian et al.** — *"Healthy Personalized Recipe Recommendations for Users."*	Personalized and health-aware recipe recommendation systems increase user engagement and trust, proving the usefulness of AI-driven personalization in food planning.

Make a case for and against your AI feature

Meet as a team, look at the existing user research and evidence you have, and detail the user need you're trying to solve.

Next, write down a clear, focused statement of the user need and read through each of the statements below to identify if your user need is a potential good fit for an AI solution.

At the end of this exercise your team should be aligned on whether AI is a solution worth pursuing and why.

We can help users reduce food waste and decision fatigue by predicting pantry depletion and generating personalized recipes that match their real inventory, time, budget, and dietary constraints.
Can AI solve this problem in a unique way?

Yes — AI uniquely enables:

- **Personalized forecasting** based on past consumption patterns.
- **Constraint-aware recipe generation** that automatically filters by allergens, budget, and available ingredients.
- **Continuous learning** from user feedback to refine predictions and recommendations over time.
  Traditional systems rely on static rules or manual input, while AI can dynamically adapt to real-world changes and user preferences.

AI probably better	AI probably **not** better
☑ The core experience requires recommending different content to different users.	☐ The most valuable part of the core experience is its predictability regardless of context or additional user input.
☑ The core experience requires prediction of future events.	☐ The cost of errors is very high and outweighs the benefits of a small increase in success rate.
☑ Personalization will improve the user experience.	☐ Users, customers, or developers need to understand exactly everything that happens in the code.
☑ User experience requires natural language interactions.	☐ Speed of development and getting to market first is more important than anything else, including the value using AI would provide.
☑ Need to recognize a general class of things that is too large to articulate every case.	☐ People explicitly tell you they don't want a task automated or augmented.
☑ Need to detect low occurrence events that are constantly evolving.	
☑ An agent or bot experience for a particular domain.	

We think AI *can* help solve user need, because:

AI can accurately forecast pantry depletion and personalize recipe suggestions based on real inventory, budget, and allergen constraints.
It reduces user fatigue from manual tracking, learns from individual consumption patterns, and continuously adapts to changing habits.
Unlike traditional rule-based systems, AI can generalize across diverse ingredient types and provide real-time, context-aware assistance.

2. Augmentation versus automation

Conduct research to understand user attitudes

If your team has a hypothesis for why AI is a good fit for your user's need, conduct user research to further validate if AI is a good solution through the lens of automation or augmentation.

If your team is light on field research for the problem space you're working in, contextual inquiries can be a great method to understand opportunities for automation or augmentation.

Below are some example questions you can ask to learn about how your users think about automation and augmentation.

Research protocol questions

- If you were helping to train a new coworker for a similar role, what would be the most important tasks you would teach them first?

  I would teach them how to keep track of pantry items efficiently and plan meals based on what's available, so they can minimize waste and avoid overbuying.

- Tell me more about that action you just took, is that an action you repeat:
  - Daily: Checking what ingredients I have
  - Weekly: Planning what to cook for the week
  - Monthly: Restocking groceries

- If you had a human assistant to work with on this task, what, if any, duties would you give them to carry out? I'd ask them to monitor when ingredients are running low, predict what I'll need next week, and suggest recipes that use what's left in the pantry.

If going to meet your users in context isn't feasible, you can also look into prototyping a selection of automation and augmentation solutions to understand initial user reactions.

The Triptech method is an early concept evaluation method that can be used to outline user requirements based on likes, dislikes, expectations, and concerns.

Research protocol questions

- Describe your first impression of this feature: "It feels very practical. I like that it predicts what I'll run out of and suggests recipes that fit what I already have. It saves both time and food."
- How often do you encounter the following problem: 'Running out of groceries unexpectedly or forgetting what's in the pantry'?
    - Daily
    - Often (a few times a week)
    - Sometimes (a few times a month)
    - Rarely (a few times a year)
    - Never
- How important is it to address this need or problem?
    - Not at all important
    - Somewhat important
    - Moderately important
    - Very important
    - Extremely important

3. Design your reward function

Once your team has had a chance to digest your recent research on user attitudes towards automation and augmentation, meet as a team to design your AI's **reward function**. You'll revisit this exercise as you continue to iterate on your feature and uncover new insights about how your AI performs.

Use the template below to list out instances of each reward function dimension.

	Positive (Outcome is good for user)	**Negative (Outcome is bad for user)**
Positive (Model predicts/acts correctly)	AI correctly predicts low-stock items and recommends recipes using what's available.→ User completes meal successfully and saves cost.→ **Reward: +1 (precision reward)**	AI correctly warns about low stock, but the user finds it unnecessary or too frequent.→ Minor annoyance → **Small penalty: −0.2**
Negative (Model predicts/acts incorrectly)	AI fails to flag an item running out soon but user doesn't notice.→ Stock-out happens later → **Penalty: −0.5 (missed recall)**	AI gives wrong or unsafe recipe (e.g., allergen ingredient included) or incorrect inventory alert.→ User frustration or risk → **Penalty: −1.0 (critical error)**

Take a look at the false positives and false negatives your team has identified.

- If your feature offers the most user benefit for **fewer false positives**, consider optimizing for **precision**.

- If your feature offers the most user benefit for **fewer false negatives**, consider optimizing for **recall**.

Our AI model will be optimized for:
→ **Precision (in forecasting)** and **Recall (in recipe generation)**

because:
→ Users benefit most when pantry alerts are accurate (avoiding false alarms)
and when recipe recommendations include all relevant constraints such as allergens, time, and budget.

We understand that the tradeoff for choosing this method means our model will:
→ Occasionally miss minor variations in consumption patterns to maintain high alert accuracy,
and may generate slightly broader recipe suggestions to ensure full constraint coverage.

4. Define success criteria

Now that you've done the work to understand whether AI is a good fit for your user need and identified the tradeoffs of your AI's reward function, it's time to meet as a team to define success criteria for your feature. Your team may come up with multiple metrics for success by the end of this exercise.

By the end of this exercise, everyone on the team should feel aligned on what success looks like for your feature, and how to alert the team if there is evidence that your feature is failing to meet the success criteria.

Success metrics framework

Start with this template and try a few different versions:

> If **__{ specific success metric }__**
>
> for **__ { your team's specific AI driven feature }__**
>
> **{ drops below/goes above }__ { meaningful threshold }__**
>
> we will **__{ take a specific action }__.**

Version 1

If **stock-out prediction accuracy**
for **the inventory forecasting feature**
drops below 80%,
we will **retrain the model using recent consumption data and perform error analysis to improve prediction stability.**

Version 2

If **recipe constraint satisfaction rate**
for **the AI recipe recommendation system**
falls below 95%,
we will **review failed recipe cases, update allergen/time/budget filters, and refine the validation schema.**

Version 3

If **average user acceptance (thumbs-up) rate**
for **recommended recipes**
drops below 40%,
we will **analyze feedback, adjust prompt templates, and improve personalization weighting in the model.**

Statement iteration

Take each version through this checklist:

- [ ] Is this metric meaningful for all of our users?
    - [ ] How might this metric negatively impact some of our users?
- [ ] Is this what success means for our feature on day 1?
    - [ ] What about day 1,000?

Final version

If overall user satisfaction (thumbs-up rate) for AI-generated recipe and forecasting features drops below 40%,
we will **analyze feedback, retrain the forecasting model if accuracy falls below 80%, and refine recipe-constraint logic if satisfaction or constraint coverage drops below 95%.**

This ensures PantryPilot continues to deliver reliable predictions, personalized recipes, and transparent AI behavior that builds long-term user trust.

Schedule regular reviews

Once you've agreed upon your success metric(s), put time on the calendar to hold your team accountable to regularly evaluate whether your feature is progressing towards and meeting your defined criteria.

Success metric review

Date:

Attendees: