Meet Fresh Design Alternatives: Brainstorming Methodologies and Prototyping Planning

Motivation

Following the product design lifecycle, we use discoveries from previous need-finding exercises, and move on to the brainstorming and prototyping phase. In order for us to create a truly impactful product, a well-planned and executed brainstorming and prototyping methodology is essential. The goal for this product design stage is to generate and refine ideas for our product, and to create multi-level prototypes for testing and evaluating its design and functionality.

This report provides a methodology outline for the process of brainstorming and prototyping, highlighting key considerations and techniques during this process for ultimately optimizing our product design.

Brainstorming Phase

I. Design Space

Guided with our problem space defined in the need-finding phase, we grow our design space by incorporating a wide range of possible designs that address the challenges of our users, as summarized in our need-finding results. At the same time, we plan on thinking outside of our primary scope and consider users, scenarios and problems that are not included in our initial problem space.

Our initial target users are **existing shallow customers with a low share of wallet**, whom we would like to convert into committed and loyal customers at Meet Fresh. Our need-finding exercises revealed that our problem space is in fact much more complex and multi-dimensional and can be summarized in two main ways:

1. **Language and cultural barriers** for non-Chinese customers may require our product design to go back to the basics and provide even more context in navigation and guidance.
2. **Location-based divergences** (i.e. suburban vs. urban, driving distance vs. walking distance) heavily impact customer behaviors, expectations and needs for desserts, thus requiring different solution design approaches.

It is important to consider our initial problem space and understand the needs, preferences, and limitations of our target users, while considering users and needs

beyond our problem space during brainstorming. We would like to use this understanding to grow our design space that is ideal for idea exploration and iteration.

II. Individual Brainstorming

<Individual Brainstorming plan and key considerations>
What is an Individual Brainstorm?
What's the purpose of BSTM?
When do we use it?
What does a good BSTM process look like? What does the result look like? What should be avoided/taken care of, during/before BSTM?
Summary & plan
(Keyword: individual)

A. Method Intro:

Brainstorming is a process of coming up with a large amount of ideas given a problem or a task. While Individual Brainstorming (IB) follows a similar definition, it's done by one person solely. The individual keeps him or herself in an isolated space to avoid outside interruptions and focus on generating ideas according to a given task. During an individual brainstorming session, individuals think freely and note down all the thoughts generated, without considering the feasibility, viability or any other constraints of them, leading to a wider range of ideas.

The root purpose of individual brainstorming is to find a way to solve the given task, by encouraging creative thinking, protecting personal perspective and avoiding peer-interactions.

People conduct individual brainstorming after deciding the problems, or requirement space, and when in need of solutions to the problems. After previous needfinding practices, our team has extracted some spiky problems and listed them in *Chapter I. Design Space* and now it's time to think of solutions. Individual brainstorming should also be conducted before Group Brainstorming, which includes interactions with peers while practicing. We understand the individual brainstorming process emphasizes individual creativity, and we tried not to share with other teammates till the end of the individual brainstorming process.

B. Dos and Don'ts

One individual brainstorming process may contain multiple individual brainstorming sessions, and each of them gives us 5-10 ideas, ideally. You can initialize the sessions iteratively. To conduct a good individual brainstorming session, you will need:

1. A clearly defined problem
2. A notebook, or any other media to note down your thoughts
3. A safe, quiet and comfortable environment

4. A time limit
5. A target amount of ideas (aim for quantity instead of quality)
6. A water-break
7. No judgment upon ideas
8. No sweeping out ideas
9. No sharing ideas during the session
10. Asking for feedback after the session if needed
11. Think of new ideas based on previous ones

C. Our Plan

To summarize, individual brainstorming is a way to gather large amounts of ideas given a pre-defined problem. It emphasizes personal creativity and avoids peer-interactions. Here's our plan to proceed the individual brainstorm exercise:

1. Get prepared:
   a. Set aside 30 minutes and make sure you'll not be disturbed during that time.
   b. Set up a quiet and comfortable environment with a desk and a chair.
   c. Bring a notebook and a pen.
   d. Write down your problem at the top of the notebook.
   e. Set an alarm for 30 minutes
2. Start thinking.
   a. Note down your ideas following tips in Section B.
   b. Keep thinking until the alarm rings
   c. Feel free to take a break
3. Ending:
   a. You can stop after the alarm rings. If you feel you still have many ideas and want to continue, just start another round of brainstorming sessions.
   b. Collect all your thoughts from all papers.
   c. Clean up the environment and leave.

III. Group Brainstorming

For our Meet Fresh project, we are planning a structured group brainstorming approach, inspired by the methodologies shared in the video we watched in class. This approach will be comprehensive, engaging all team members in generating and discussing diverse ideas that align with our project's goals and the unique needs of Meet Fresh's clientele. For our Meet Fresh project, our group brainstorming approach will include:

- **Preparation**: Team members will understand diverse customer perspectives, and review individual brainstorming results for an idea-rich discussion.
- **Session Structure**: Our brainstorming session will reflect the multi-layered nature of our problem space.
    - *Idea Sharing*: Discuss ideas from individual brainstorming, focusing on their potential to tackle challenges of different customer segments.
    - *Idea Generation*: Create new ideas collectively, encouraging the blending of thoughts for innovation that caters to Meet Fresh customers' diverse needs.
    - *Idea Evaluation*: Collaboratively select promising ideas that can enhance customer engagement and loyalty across our varied customer base.
- **Cultural Sensitivity**: We'll ensure a respectful environment that values cultural nuances, allowing us to generate ideas attuned to the unique needs of our diverse customer base.
- **Facilitation**: A facilitator will guide the discourse, maintain focus on our project objectives, balance the contribution of ideas, and ensure cultural sensitivity.
- **Idea Capture**: Visual aids like whiteboards or sticky notes will be used to capture ideas, track the diversity of thoughts, identify potential synergies, and facilitate the idea evaluation and selection process.
- **Follow-up**: We'll consolidate the outcomes, including next steps, to share with the entire team. These steps may involve refining chosen ideas, developing culturally attuned prototypes, or conducting further research.

Challenges and Mitigation Strategies

- **Dominance of Certain Voices**: Some team members might dominate the conversation, which may limit the input from quieter members.
    - **How we can mitigate it**: Establish ground rules for equal participation. Use techniques like "round-robin" to ensure everyone has a chance to speak.
- **Homogeneity Bias**: As all team members are Chinese, there may be a tendency to favor ideas that resonate with Chinese cultural norms and personal experiences, which could limit the diversity of ideas.
    - **How we can mitigate it**: Encourage the team to think from the perspective of various customer segments, not just those they personally identify with.
- **Assumption Bias**: Team members might make assumptions about customer behaviors or preferences based on their own cultural understanding, which might not be universally true.

- - ○ **How we can mitigate it**: Validate assumptions with actual customer data or through user research. Differentiate between personal preferences and broader customer needs.
  - **Groupthink**: The desire for harmony or conformity might result in an incorrect or poor decision.
    - ○ **How we can mitigate it**: Promote an environment where dissenting ideas or viewpoints are encouraged and discussed.
  - **Off-topic Discussions**: Conversations may deviate from the main topic, wasting valuable time.
    - ○ **How we can mitigate it**: Assign a facilitator to guide the discussion and keep the focus on the topic. Use a "parking lot" to save off-topic ideas for later discussion.

IV. Criteria & Ideas Prioritizing

We foresee that a lot of ideas will be generated. Before we are able to move on to the prototype development phase, we need to narrow down ideas from the brainstorming phase by adopting approaches that help us evaluate and identify promising ideas for developing prototypes.

In general, we would like to prioritize our ideas based on whether they address our user requirements from our need-finding phase. In addition, we plan to evaluate ideas using the following questions:

1. *Which ideas align best with our product and business goal?*
2. *Which ideas generate the highest level of impact?*
3. *Which ideas are most technically feasible given complexity level and time/budget constraints?*

Following the Impact-Complexity quadrant analysis for mapping our user requirements (**Fig. A.1**), we are able to develop a better understanding of these requirements to help with prioritization given their technical complexity and business impact into the following:

1. **Quick Wins**: low-hanging fruits that can be achieved quickly for immediate results and impact
2. **Big Bets**: high-risk and high-reward initiatives that require extensive resources but critical to business success
3. **Money Pit**: resource-draining initiatives with low ROI
4. **Fill-Ins**: low-effort and low-impact initiatives that are not top priorities but still essential for maintenance
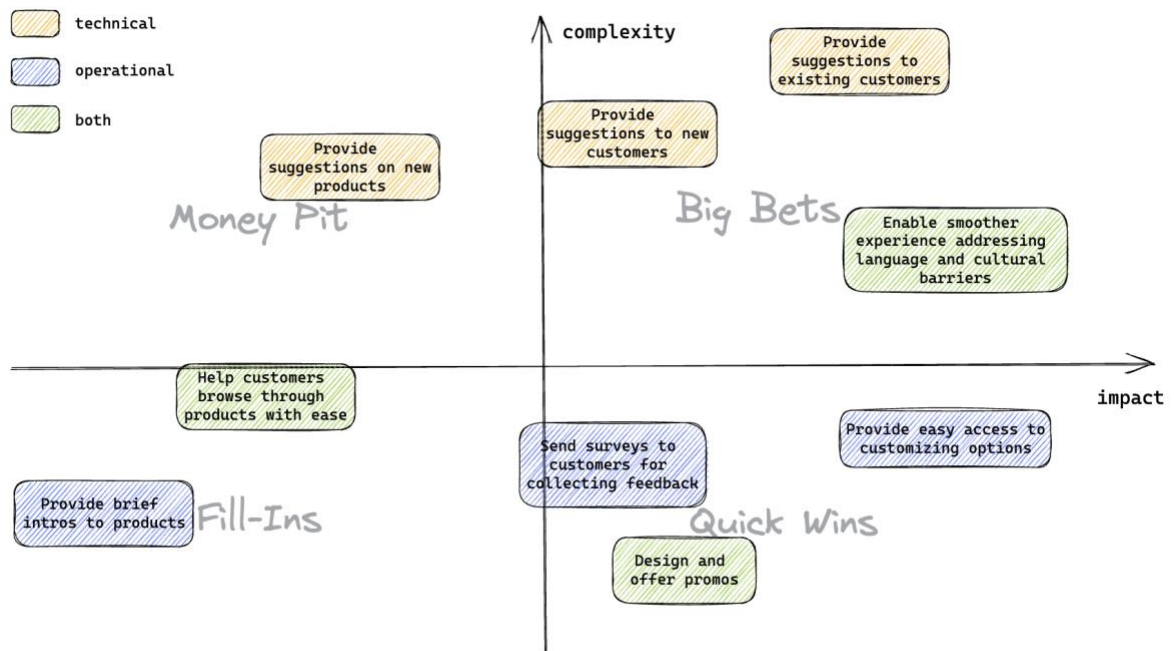
(Fig. A.1)

This quadrant mapping exercise serves as a basis for how we would like to prioritize our long list of ideas.
We also would like to conduct the following idea mapping exercise ((**Fig. A.2**) to support our idea prioritization and ensure that our selected ideas align with our user requirements and business priorities before we move to the prototype development phase.

	Impact	Complexity	Category	Requirement Alignment	Technical Resource Required	Cost	Data Availability
Idea 1	High	High	Big Bets	Provide suggestions to existing customers given order history	DS, DE, MLE, Front-End		Historical transaction data available
Idea 2							
Idea 3							
Idea 4							
Idea 5							
Idea 6							
Idea 7							

(Fig. A.2)

The table in Fig. A.2 is where we would conduct the idea mapping exercise. We list all the ideas collected during group brainstorming as the rows, and highlight their features

as the columns, for example the Impact, Complexity, Category, Requirement, Cost, etc. After completion, this table will serve as a guide to prioritize the ideas.

Prototyping Phase

I. Low-Fidelity Prototype Development

Verbal prototyping

Verbal prototyping involves describing ideas in words. It is the **simplest and quickest** way to create a prototype, and it can be a **great way to get feedback on your ideas early on.** To create a verbal prototype, simply start by describing ideas to someone in person, over the phone, or even through email. The description should be as specific as possible, such as different features of the product or service, and how users will interact with it. Once the description is completed, ask the audience for feedback. Ask them what they think of the ideas, and what they think could be improved. This feedback will help you to **refine the ideas** and make them better. Verbal prototyping is also considered as a **low-cost** way to get feedback and identify potential problems.

1. Define the scope and purpose of the prototype: Determine what aspect of your product you want to test with your verbal prototype, and what are the features of your product or service?
2. Be specific when describing your idea: for example, when describing a web service, do not say "It's a website." Instead, say "It's a website that allows users to book appointments with doctors."
3. Use visuals to help explain your idea. If you can, create sketches or wireframes of your product or service. This will help people to understand your idea better.
4. Get feedback from others. Once you have described your idea, ask for feedback from others. This feedback will help you to refine your idea and make it better.
5. Iterate on your prototype. Once you have received feedback, iterate on your prototype. Make changes based on the feedback that you have received.

To better execute verbal prototyping successfully, here are some additional tips:

- *Practice* describing your idea before you share it with others. This will help you to feel more confident and prepared.

- _Be open_ to feedback. Remember that the goal of verbal prototyping is to get feedback on your ideas. So be open to hearing what others have to say, even if it's not what you want to hear.
- _Don't be afraid to experiment._ Verbal prototyping is a low-cost way to experiment with different ideas. So don't be afraid to try new things and see what works.

Paper prototyping

Paper prototyping allows us to prototype a **digital product interface** without using digital software. The technique is based on creating hand drawings of different screens that represent user interfaces of a product. While this is a relatively simple technique, it can be useful when a product team needs to explore different ideas and refine designs quickly. This is especially true in the early stages of design when the team is trying different approaches. Here's how we should conduct a paper prototype.

1. Define the scope and purpose of the prototype: Determine what aspect of your product you want to test with your paper prototype, and what specific tasks or interactions you want to simulate.
2. Sketch the screens: Using pen and paper or a whiteboard, draw out the screens or user interface elements that will be included in the prototype. Focus on basic layout, navigation, and content placement.
3. Cut and assemble the screens: Cut out the individual screens or UI elements and arrange them in a logical sequence. You can use tape or magnets to attach the pieces to a wall or board.
4. Add interactivity: To simulate user interactions, use additional pieces of paper or sticky notes to represent buttons, menus, and other interactive elements. You can also use a pen or marker to draw arrows or other visual cues to show the flow of the interface.
5. Test the prototype: Invite users or stakeholders to interact with the prototype and complete the tasks or scenarios you have designed. Observe their behavior and take note of any pain points or usability issues.
6. Iterate and refine: Based on the feedback you receive, make changes to the prototype and repeat the testing process until you have a design that is functional and user-friendly.

Wizard of Oz prototypes are prototypes with **fake functions -** for instance, where you get a team-mate to mimic complex interactions rather than code a piece of software for it. Like the Wizard of Oz in the story (who generates an ominous and deceptive appearance from behind a curtain), you are mimicking some aspects of your product. They're a kind of low-to-medium-fidelity prototype, where the key functions are not functional at all while other aspects such as visuals are fully designed.

The idea of Wizard of Oz prototypes is to simulate a system and get users to believe that the system is fully functional, so you can test it and collect feedback from users while saving time and resources. For example, you can create a Wizard of Oz prototype for a smart assistant, where your team-mate types out responses to trick the user into thinking that the smart assistant is fully functional.

1. Define the scope and purpose of the prototype: Determine what aspect of your product you want to test with your WoZ prototype, and what specific tasks or interactions you want to simulate.
2. Design the interface: Create a basic interface design that will be used by the user to interact with the system. This can be as simple as a series of buttons or a voice recognition interface.
3. Recruit a wizard: Find someone who can act as the wizard behind the scenes. The wizard should be familiar with the tasks and interactions you want to simulate, and should be able to accurately mimic the behavior of the system.
4. Conduct the test: Invite users to interact with the interface and complete the tasks or scenarios you have designed. The wizard should be listening or observing the user's actions and manually responding to their inputs in a way that simulates the behavior of the system.
5. Observe and take notes: Observe the user's behavior and take note of any pain points or usability issues. Also, observe the wizard's behavior to ensure that they are accurately simulating the system's behavior.
6. Iterate and refine: Based on the feedback you receive, make changes to the prototype and repeat the testing process until you have a design that is functional and user-friendly.

Evaluation Phase

I. Prototype Evaluation

1. Three Types of Evaluation
   a. Qualitative Evaluation – Get qualitative feedback from users
      i. Assess and understand the quality, nature, and characteristics of a particular phenomenon or subject.
      ii. Gathering non-numerical data such as words, descriptions, opinions, and observations, rather than numerical data.
      iii. Provide insights and interpretations of the underlying factors, contexts, and meanings associated with the subject under study.
      iv. We can implement techniques used in the needfinding process.
   b. Empirical Evaluation – Control experiment, quantitative result
      i. Systematic and objective assessment of theories, hypotheses, or research questions through empirical data
      ii. Conducting experiments, observations, surveys, or other methods of data collection
   c. Predictive evaluation – Evaluate without users
      i. Although not my favorite type of evaluation, real users are expensive and slow.
      ii. Simple evaluation on day to day basis
2. Evaluation Metrics
   a. Reliability: Whether a measure consistently returns the same results for the same phenomenon
   b. Validity: Whether a measure's results actually reflect the underlying phenomenon
   c. Generalizability : to what extent we can apply the result to other people
   d. Precision: The level of detail a measure supplies
   e. Efficiency: how long it takes user to accomplish user to complete task
   f. Accuracy: how many errors does the user commit while committing task
   g. Learnability: how long does user to hit expertise
   h. Memorability: user's ability to remember how to use an interface over time
   i. Satisfaction: How satisfied the user is
3. Evaluation Timeline
   a. Early: Formative
      i. Primary purpose: redesign and improve interface
      ii. Tend to be more qualitative (interpretative, informal)
      iii. Improve or understand the task
      iv. Use qualitative data
      v. Lab setting:

1. Explain what we are going for
2. Get their feedback
   b. Late: Summative
      i. Conclusively saying what the difference is
      ii. Tend to be more empirical, controlled, formal
      iii. Demonstrate change
      iv. Use quantitative data
      v. Field testing
         1. Already have a working prototype
         2. Reflect on it after the task that the interface has helped

II. Iterative Refinement

<Highlight importance of iterative prototype refinement and go back to product design lifecycle>

A. What is an Iterative Refinement?

Iterative refinement is the process of gradually improving a solution through multiple cycles of evaluation, feedback, and adjustment, resulting in continuous enhancements and iterative improvements.

B. What's the purpose of IR?

Iterative refinement aims to improve solutions through multiple cycles of evaluation, feedback, and adjustment. It enhances solution quality, incorporates feedback, adapts to changing requirements, mitigates risks, fosters collaboration, and manages complexity. By gradually refining the solution, it aligns with user needs, addresses deficiencies, and achieves incremental improvements. The iterative approach allows for flexibility, course correction, and continuous evolution, resulting in more robust and effective solutions. Ultimately, iterative refinement fosters a mindset of continuous improvement and drives the development of high-quality, user-centric solutions.

C. When do we use it?

   a. Iterative refinement is used when developing complex solutions, solving intricate problems, or designing products with evolving requirements. It is valuable when the initial solution is not fully defined or when user feedback and collaboration are critical. It is employed in software development, engineering, design, and various problem-solving domains. Iterative refinement allows for adaptability, mitigating risks, incorporating feedback, and achieving incremental improvements throughout the development or problem-solving process.

D. What are the methods of IR and how are they different?

a. Agile Development: Agile methodologies, such as Scrum or Kanban, emphasize iterative and incremental development. They involve breaking down the solution into small, manageable tasks or user stories that are completed within short iterations called sprints. Feedback from stakeholders is actively sought and incorporated in subsequent iterations.

b. Spiral Model: The spiral model combines iterative development with risk management. It consists of multiple cycles or spirals, each encompassing planning, risk analysis, prototyping, development, and evaluation. The process loops back through these stages, with each iteration building upon the lessons learned and addressing identified risks.

E. What does a good IR process look like? What does the result look like?

a. A good iterative refinement process involves clear goals, prototyping, evaluation and feedback loops, collaboration, adaptability, and incremental improvements. The result is a refined solution that meets objectives, addresses user needs, and exhibits improved quality and functionality through iterative enhancements.

Methodology Summary

To summarize, we plan to execute our brainstorming and prototyping as follows.
Firstly, everyone in the group will conduct an individual brainstorming process, to gather a large amount of ideas.
Secondly, all the members gather together and conduct a group brainstorming process by sharing their individual brainstorming results on the same paper.
Next, we'll analyze and prioritize all the ideas we have collected, quantitatively from different perspectives.
After finding the top prioritized ideas, we'll design and implement 3 kinds of low-fidelity prototypes gradually: Verbal, Paper and Wizard of Oz. Then we utilize the prototypes and interact with customers to collect feedback and suggestions. We evaluate the prototypes by checking if it satisfies customers' needs, if not, we'll improve them according to feedback collected.
We keep doing the above exercise iteratively until prototypes meet customers' needs.