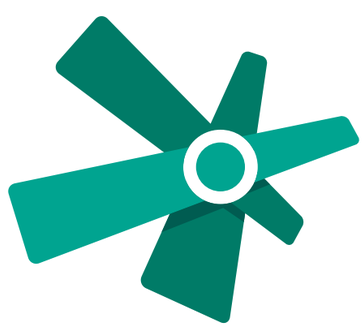
Pivotal



Pivotal **Labs**

Client Designer Playbook

v2.0.0

Contents

Welcome, Product Designer!

Leading product design is a multi-faceted and challenging role. We're here to help you and your team build products that deliver the most value for your customers.

Whether you're coming from a big enterprise company or small start up, this playbook will help you get a sense of the tools and methods that lay the foundation for a healthy product design practice.

By partnering with Pivotal Labs, you are committing to work with collaborators rather than contractors. Together we will produce software based on your continuous input while you lead design decisions and influence outputs.

Together we will be leading all aspects of product design—everything from advocating for the user, to supporting an agile development backlog, to educating a balanced team.

We will help you break down big ideas into manageable pieces and establish a regular design cycle to hypothesize, design, and validate the right features to build a product that is always providing user value.

And we will encourage your entire team to provide regular feedback on process and performance, which will drive continuous improvements in efficiency and happiness.

WHAT TO EXPECT

Learn by doing.
We teach by example. From day one, you will be a practicing Product Designer, guided by us.

Our office is your office.
We expect you to be here full-time to collaborate, disseminate design decisions, and support the team. Co-location in our open and collaborative environment is crucial to faster feedback loops and a healthy project. Also, feel free to help yourself to our awesome array of snacks and daily breakfasts!

Constant collaboration.
We believe that the best products are informed by diverse perspectives. You will be partnering with developers, product managers, and business stakeholders to inform your product decisions. You won't be working alone and you won't see anyone wearing headphones - unless they are pairing with someone remotely.

You lead design.
Your role on the team is to be the design leader; to understand and empathize with the user, design solutions, support the backlog, and run all design meetings. Your design pair is here to help and support you.

Sustainable pace.
Our working hours are 9 am to 6 pm. Healthy product development is a marathon, not a sprint; by maintaining a 40-hour work week we keep a sustainable and predictable pace.

Team Roles

Every team makeup is different but what's important is having the right balance of skills and perspectives. We work in small "Balanced Teams" consisting of Designers, Engineers, and Product Managers—all co-located and fully dedicated to your product.

Your Role as a Product Designer

Your design role is multi-disciplinary, delivering value in the form of design decisions. You are the "Empathizer-in-chief" and your job is to understand users in order to define solutions that are useful, usable, and desirable. You provide design education, direction, and leadership as an expert in design patterns.

Your Pivot Pair

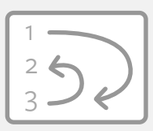
You will be paired with a Pivotal Product Designer who will guide and support you. As part of a Balanced Team, you and your Pivot pair will take a User Centered Design (UCD) approach to problems following the "I do, We do, You do" process. Over the course of the project you will take the lead with your pair playing a supporting role.

OTHER ROLES ON YOUR TEAM

These are the most common roles you will see on a Balanced Team. Every project is unique, however, so your team's roles may differ slightly.

Developer

The Developer delivers user stories and associated tests and assists with story estimation in the Iteration Planning Meeting. You will work with developers to understand the technical consequences of design decisions and guide the implementation. You will help them gain empathy for the user by involving them in user research and collaborating with them in solution generation exercises.



Product Manager

The Product Manager (PM) must have a clear understanding of the users' needs, business objectives, and stakeholders' vision. With this knowledge and additional feedback via the expertise of the team, the PM makes feature decisions—writing stories and prioritizing what to build next. You will work very closely with the PMs throughout the project. They participate in design activities and together you build, maintain, and prioritize the backlog.

Anchor

The Anchor is responsible for the success of the project in terms of technical execution and process. They generally remain on the project from beginning to end. The Anchor acts as a point person for the client and becomes the developer who has experienced the entire history of the project. You will work with them as you would with other developers.

Client Liason

Each team has a Client Liaison (CL). The CL is responsible for the client-Pivotal relationship and the overall success of the project. You will interact with them weekly and they are a good point of contact to raise project concerns with or help unblock the team.

Weekly Project Rhythm

You'll quickly learn that a healthy Agile project has a constant communication rhythm. Most of the time you will be pairing and working collaboratively, but here are some daily and weekly meetings we recommend you have with your team.

Daily Meetings	Meeting Purpose	Your Role
Project Standup	To check in on everyone's progress, plan, and blockers.	• Understand blockers and their implications on the current priorities • Remind everyone of action items and upcoming milestones
Office Standup	To kick off the day with new faces, helps, interestings, and events.	• Ask for help if you need it • Announce new teammates or guests • Share interesting learnings that could help other teams

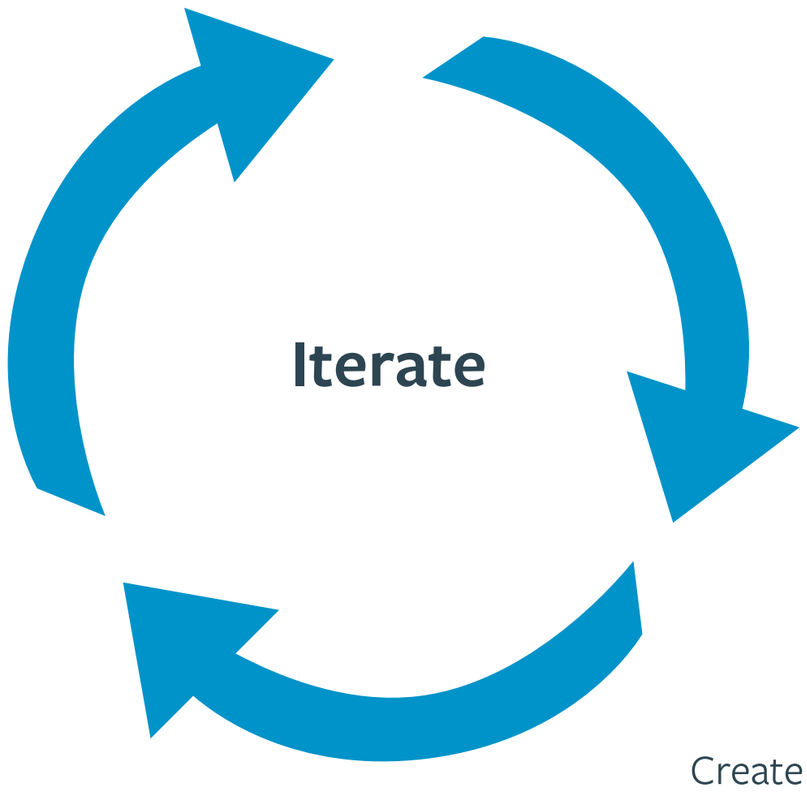
Weekly Meetings	Meeting Purpose	Your Role
Iteration Planning Meeting (IPM)	To estimate complexity of the week's backlog of prioritized user stories that the developers can pick up for implementation. (PM leads.)	• Make sure upcoming stories have been designed for. • Advocate for the user. • Support the PM in facilitation as needed
Retro (short for "Retrospective)	To create a safe space for the team to celebrate the past week's successes, discuss points of confusion, and reflect on challenges	• Openly and honestly share your experiences from the week • Dig deeper on items raised by others to understand root causes • Collect a list of actions in response to the team's reflections
Design Critique (also known as "Crit")	Design feedback comes in many forms. In "Crit", the entire design team gathers to review and give feedback on each others' work, while the participating project's PMs and Developers also attend to witness the feedback. This meeting is done as needed, not necessarily every week.	• If being critiqued: Bring a feature or a flow and identify the kind of feedback you are looking for. • If critiquing: Give observations and feedback based on what is being asked for. Don't offer solutions, and always be kind.

Design Cycle

As Product Designers, we practice User Centered Design and use the "Think, Make, Check" cycle optimized for validation, delivery, and learning (based on Lean Methodologies). We don't treat design as a phase and we keep everyone involved throughout the product lifecycle.

Think

Conduct exploratory and generative research, contextual inquiry, user interviews, and domain modeling. Identify assumptions and define hypotheses. Create personas, scenarios, user flows, and service blueprints.



Make

Create wireframes, hi-fidelity and lo-fidelity mockups, prototypes, and living styleguides.

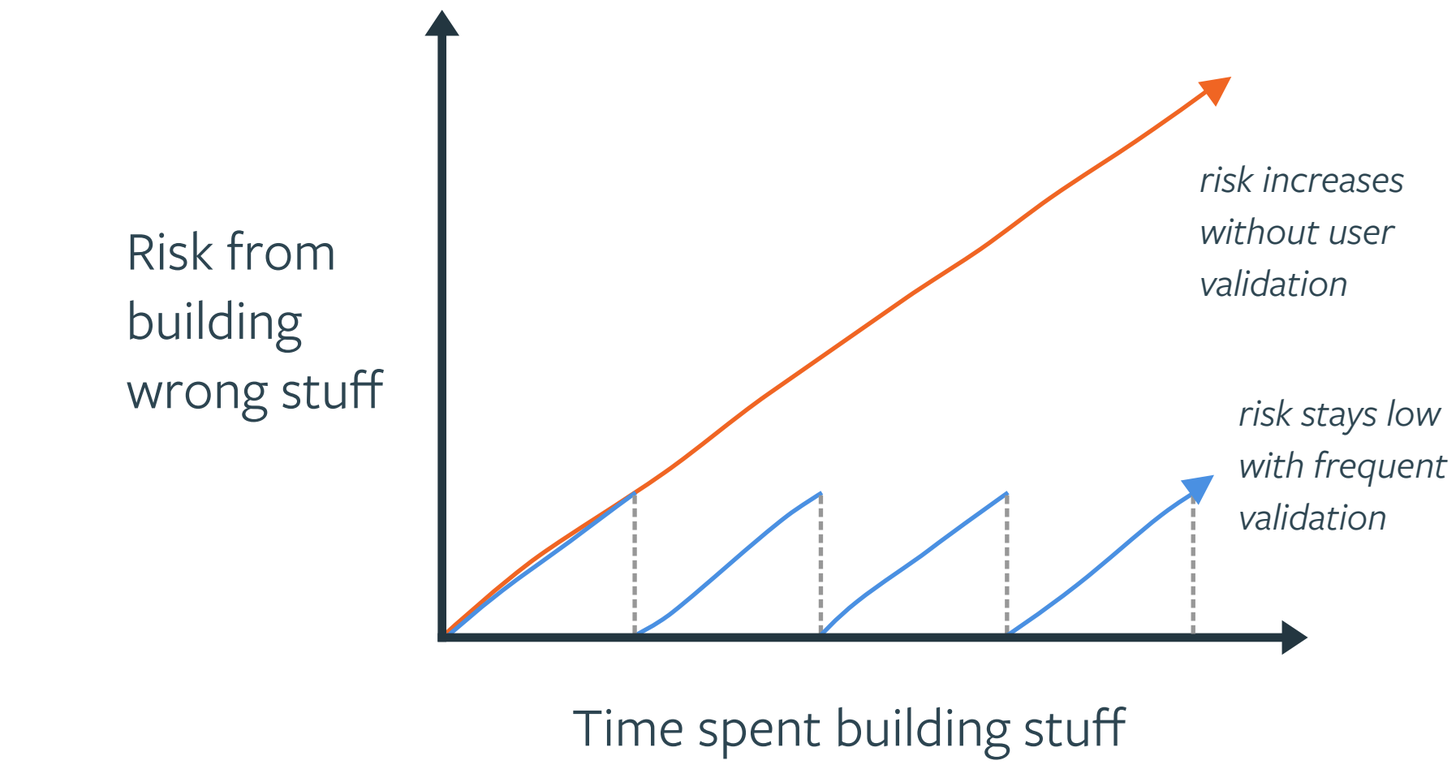
Check

Concept and usability testing, validate assumptions, hypotheses, and priorities, analytics, A/B testing, funnel analysis

Why?

We work within this cycle to reduce risk of spending time and money building products that deliver no meaningful or impactful value to businesses and users.

Product Risk over Time



What Designers Do

In embodying the iterative Design Cycle methodology for a user-centered, lean and agile process, Products Designers follow these design practices and concepts.

Conduct Research

After gathering business problems, the Designer seeks to understand user problems using research methods. The Designer will lead many types of research throughout the product lifecycle with a regular cadence. Early on, they will research potential users to find out what their goals and needs are in order to determine product value. Later on, they will run usability testing to ensure that the design is easy to use and understand.

Collaborate Across Disciplines

Designers are always collaborating informally, but sometimes when there is a challenging design on the horizon, the Designer will lead a more structured 30min - 1 hour Design Studio. This workshop gathers various team members into a room to sketch solutions around a single focused scenario and a prioritized set of features. The goal is to uncover use cases, gather ideas, and provide insight to other team members to make planning smoother.

Deliver Design Decisions

The Designer leads the team through a thoughtful and meaningful process making incremental design decisions along the way. These decisions are delivered in many forms such as a conversation, sketch, whiteboard session, wireframe, detailed mockup, or a prototype.

Seek Feedback

We seek feedback from users and stakeholders constantly, and we like to make time and space for Designers to solicit feedback from each other. When there is a design or flow that you want to get a fresh designer take on, Designers are encouraged to share their work with fellow design peers and gather feedback.

Communicate Early and Often

Being part of a Balanced and Agile team means the designer will communicate with the team as a leader, facilitator, and educator. The team will rely on the Designer to lead how the product will act and look, facilitate conversations to gather feedback and educate the team on research findings.

Conduct Research

Research is an investigative activity to study and learn about users. You will conduct both exploratory and evaluative research to uncover user problems and test solutions. You may also tap into existing research, analytics, and competitive analysis, as well as other forms of qualitative & quantitative research.

Empathy, active listening, and observation are some of the most important skills Designers should utilize in order to best understand their users.

Why?

Research grounds design decisions in reality by ensuring that you're optimizing design for specific people in specific places at specific points in time. Similar to scientific methodology, you will create hypotheses and experiments in order to maximize your chances of success by learning whether your product is succeeding each step of the way. You spend less time building features that provide no impactful value.

How?

There are many ways to conduct research—usability testing, user interviews, contextual inquiry, surveys, card sorting, video diaries, and A/B testing are all methods and tools you can use to gather information and feedback directly from your users.

Example: User Interview

Prepare Goals

- Gather team members to generate questions and prioritize them into top learning objectives.
- Write a plan and script to use during an interview or observation session.

Conduct research

- Recruit (create a screener if necessary) and schedule users.
- Lead user interviews while other team members take notes on user quotes and observations.
- Seek existing and past behaviors by asking non-leading questions and look for stories from the user.

Synthesize into action

- Once a few users have been observed or interviewed, lead a synthesis session with the team.
- Cluster findings into insights and problems. Use findings to build real world personas.
- Prioritize actionable tasks. Reference research to make evidence based decisions.

Deliver Design Decisions

Being co-located with a balanced team and shipping software as a team means that you won't be throwing deliverables over a wall. Design decisions are the real deliverable and you value sharing them through conversation above documentation.

How?

The focus is on creating user value by continuously shipping working software, so your time is spent deciding what the smallest piece of user value is and what the smallest amount of design is to communicate that solution. There are plenty of ways to communicate your design decisions to the broader team without resorting to specs and documentation. Depending on the size and importance, a design decision can be communicated across any or all of the following levels of fidelity.

Conversations. These can be quick, impromptu pow-wows at your team seating area, check-ins with your PM, stand ups, and taps on the shoulder.

Sketches. Sometimes the only thing that you need to communicate an idea is a sketch on a post-it or a few minutes at a whiteboard with a developer. Sketching workshops like Design Studio invite the rest of the team to participate and share their ideas so everyone feels heard.

Wireframes. The natural step after sketching is wireframing. We create and share these to walk through user flows and start thinking about information architecture, navigation, rough layouts, and interactions.

Low to High fidelity Mockups. Mockups can range from digitized wireframes to fully branded and visually designed user interfaces. These can also be turned into clickable prototypes or attached to Tracker stories.

Prototypes. Paper or clickable prototypes help communicate user flow, navigation, and basic interactions. Coded prototypes are an excellent way to rapidly iterate on design and build empathy for developers. Not only are these good for communication with teammates and stakeholders, they are a great way to elicit quick feedback from users.

Living Style Guides. Style guides create consistency. A living styleguide is a coded source of truth the whole team can use and reference. Moreover, its creation and maintenance is a great opportunity for designers and developers to pair together and reach shared understanding.

Communicate Early and Often

An important part of working on an agile and balanced team is having consistent and clear communication at all times. Consistent communication means that teams talk every day. We share the same trajectory, we share an understanding of who our users are, and what their problems are. We prevent the siloing of vital information.

Why?

Traditional design/development firms build their business around a series of handoffs and deliverables. Because our teams integrate product, design, and development, we avoid operating in silos and focus on solving problems together at the same time.

We strive to minimize deliverables and focus on collaboration because consistent communication creates shared understanding which allows us to move at a consistent pace, reducing time wasted in writing emails, creating presentations, and scheduling more and more meetings.

How?

There are several ways you can communicate with the rest of the team.

IPM. This weekly PM-led meeting is where designers, developers, and PMs are able to get on the same page. Together we make sure the upcoming stories are validated, prioritized, and ready for development.

In-flow Conversations. One of the first things people notice when coming to Labs is the audible hum of the office. That's because the teams are communicating all the time. You're encouraged to walk over to your teammates at any point to discuss anything from high-level concepts and questions, down to pixel-sized details. Your teammates will do the same with you.

Pairing. Pairing is the primary way we communicate across the team. As a designer, you will pair with not only other designers but also other disciplines.

- **Design/Dev:** At least once a week designers and developers will pair on design-related tasks and bugs that the designer and the PM have prioritized.

- **Design/PM:** PMs and designers pair often, especially during research, product definition, and problem and feature prioritization.

- **Design/Design:** By working together, you and your pair will share context, design decision-making, and grow each other's skills.

Collaborate Across Disciplines

As a Designer, your job isn't to know everything. There are many opportunities to conduct collaborative brainstorming activities to help you get input and gather ideas from the rest of your team.

Why?

By encouraging cross disciplinary roles to think deeply about a problem simultaneously, we discover a lot more unknowns than if a Designer generated solutions alone.

All disciplines learn how to give and receive constructive design feedback. They also get the opportunity to be informed about product decisions every step of the way increasing shared understanding and product ownership.

Passing sketches around, silently making comments on post-its

Sketches with comments in colored markers

How?

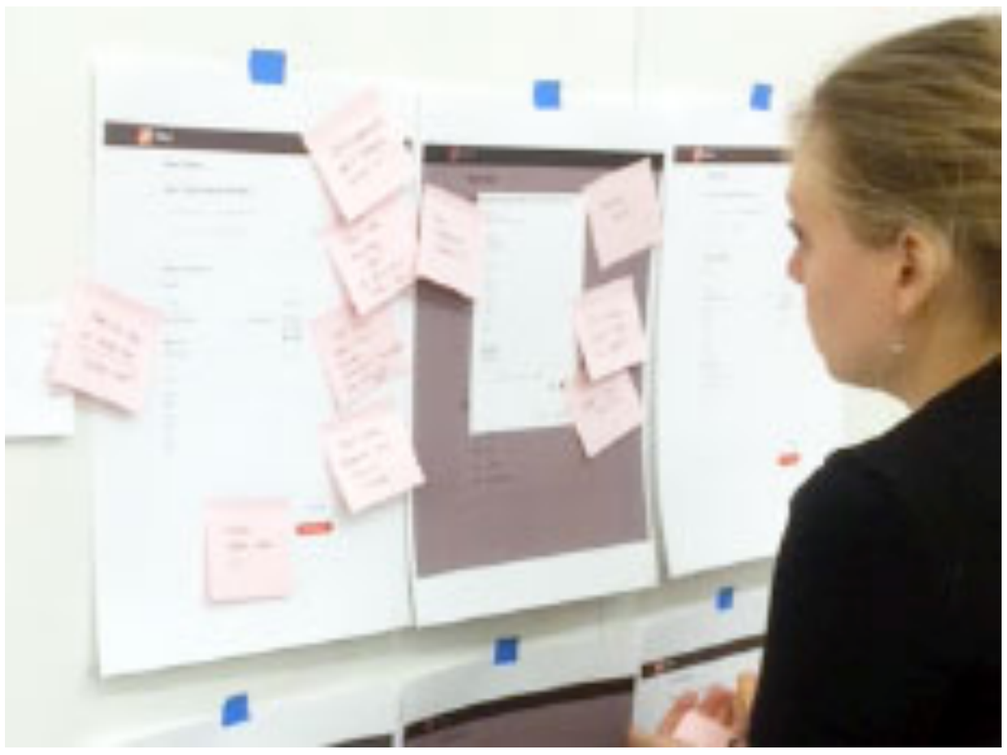
Example: Design Studio

- Focus the team by writing down a specific scenario to sketch a solution for.
- Give team members a limited amount of time to sketch ideas.
- When time is up, take feedback to create one or multiple design solutions from everyone's contribution.

Silently reviewing sketches before discussing

Seek Feedback

You will be continuously be gathering feedback on yourself, on your work, and on the team throughout the design cycle and project engagement.



Why?

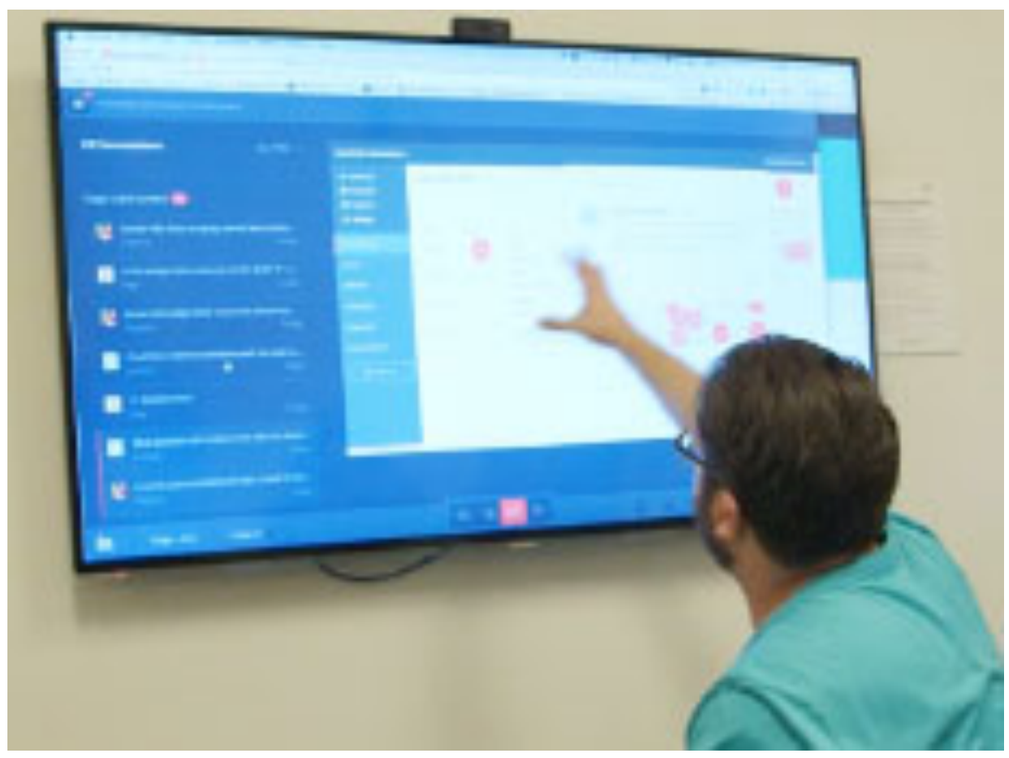
Gathering data and closing feedback loops is how we grow as designers and ensure a great product and a healthy team. In both our work and our team we want to uncover the pain points and areas that don't make sense and discover actionable steps to improve what isn't working and expand on what is.

How?

Weekly retrospectives, one-on-ones, usability testing with users, and Design Critique are just a few ways we get feedback.

Example: Design Critique

There are many formats used in Design Crit, but typically this weekly meeting consists of one designer sharing while fellow designers and teammates give feedback. Sometimes Crit is structured and documented. Other times it can be conversational. Everything from consulting conversations, to high level user flows, to visual design, to research plans is up for feedback.



Discovery and Framing

At the start of a project or a new track of work, we begin with Discovery and Framing (D+F), a collaborative process led by Design (including you) that explores and validates product ideas to gain the definition needed for implementation.

Discovery

WHAT PROBLEM ARE WE SOLVING?

The goal of Discovery is to gain a deep understanding of the business case and users of a proposed product.

ACTIVITIES

Understanding Users

- **User Interviews:** One-on-one interviews with the end users
- **Contextual Inquiries:** Observing users in their daily tasks
- **Generative Interviews:** Collaborative prototyping with the user

Understanding the Business

- **Stakeholder Interviews:** One-on-one interviews with the stakeholders
- **Competitive Analysis:** Analyzing competitor companies and products
- **Business Model Canvas:** Define and gain a collective understanding of the business model

Understanding the Tech Stack

- **Integrations Research:** Developers analyze existing code bases and necessary integrations

Framing

HOW ARE WE SOLVING THE PROBLEM?

The goal of Framing is to gain the ability to make prioritization decisions about features of the product.

ACTIVITIES

Understanding Users

- **Personas:** An overview of a target user outlining their process or workflow, needs, and goals
- **Design Insights:** Things to consider when designing for a particular persona
- **Scenarios:** Writing product use cases
- **Low Fidelity Wireframing:** A rough outline of a screen, outlining key activities
- **Usability Testing:** One-on-one interviews with end users testing product ideas through wireframes and scenarios

Understanding the Business

- **Define and Prioritize Feature Set:** Generate a high-level set of features to describe a minimum viable product

Understanding the Tech Stack

- **Basic App Creation:** Developers scaffold out the basic structure of the app

Outcomes

WHAT TO EXPECT AT THE END OF A D+F

- A shared understanding of the business
- Reduced product risk by outlining a validated product strategy of prioritized features based on evidence from target users
- Definition of the project MVP
- Validated target users, along with a deep understanding and strong empathy for them
- Two weeks of task for developers to start building
- A low fidelity UX framework of wireframes, and core use scenarios

Glossary

A

Agile
A group of software development methods based on iterative and incremental development, in which requirements and solutions evolve through collaboration between self-organizing, cross-functional teams. It is a conceptual framework that promotes well planned, small iterations throughout the development cycle.

Anchor Role
An experienced developer who, in addition to coding full-time, leads the technical aspects of the project from from start to end. The anchor acts as a resource for the rest of the development team for technical and non-technical issues.

B

Backlog
The backlog is a list of prioritized stories that make up the planned work for the current iteration. Stories can be added and removed from the backlog and re-prioritized as necessary during an iteration.

Balanced Team
An autonomous team that has people with a variety of skills and perspectives that support each other towards a shared goal. The team values cross-disciplinary collaboration and iterative delivery.

Blocker
A situation or issue, brought up at project standup, that is delaying or preventing project progress. The team self-organizes to resolve the blocker and when it cannot be unblocked by the team, it can be escalated to the Client Liason and/or stakeholders.

C

CI
CI stands for “Continuous Integration”, it is a server which is a dedicated machine that runs a project-specific defined set of tests periodically.

D

Design Crit
A feedback session Designers run to get feedback from fellow Designers and Product Managers.

Design Review
A weekly team meeting to get an update on the progress of design and research. The goal of a design review is for a designer to leave the meeting with a clear list of updates to the designs and workflows they’ve presented based on feedback from developers, PMs, and client stakeholders on feasibility, business value, brand, priority, and scope.

Design Studio
A solution brainstorming activity a Designer leads to gather ideas in the form of sketches.

E

Empathy
Empathy is an ability to understand what other people are thinking and feeling. We hire for empathy and compassion because it enables us to be kind and effective collaborators. Good collaborators build happier teams.

Epic	An epic is a collection of stories that make up a larger product release, feature set, or development focus. Epics are useful for prioritizing groups of stories against other groups of stories.

I

Iteration	A 1-week planning cycle. Planning and development is iterative. Because we are constantly coding and testing, the products we build are always ready to go live. This iterative process allows for changes as business requirements evolve. Daily and weekly software builds provide constant validation that the software meets the business requirements. You always have complete control of the product and the timeline.
IPM	IPM stands for Iteration Planning Meeting. A weekly meeting at the start of an iteration, during which the team reviews the upcoming stories in the backlog, ensures the backlog is full for the next two or three iterations, confirms the prioritization of stories, and estimates any unestimated stories using a point system.

L

Lean	Lean is the practice of build products that deliver the most value for your customers and minimizing waste by systematically identifying assumptions and validating with actual users.

M

MVP	A Minimum Viable Product is the cheapest, fastest, simplest thing that can help validate or invalidate hypothesis about customer behavior.

P

Pair Programming	Pair Programming is the practice of having two developers work together at the same computer to complete each task. At Pivotal Labs we pair all of the time. This practice of focusing two minds on the same challenge leads to better decisions the first time around, fewer knowledge gaps, and continual implicit training and knowledge transfer. Pairing results in fewer defects, better code, and ultimately much more sustainably efficient development. As pairs rotate, knowledge is spread rapidly through the team, avoiding silos of knowledge and allowing for team growth.
Persona	A model of a user of your product/service based on similar needs, goals, context, and tasks of many actual users. Developed through user research. Used to gain and retain empathy for the users we are solving for.
Pointing	In order to measure a story's complexity, Pivotal uses a point system of 0, 1, 2, 3, 5, and 8 points. During an IPM, developers will discuss and agree on a point estimate that reflects the story's complexity to the best of their understanding. Points do not reflect a measure of how much time a story will take to complete since time is especially difficult to estimate and can vary based on external factors. Measuring relative complexity is easier, and allows for much more consistent measures of team progress.

R

Retro	This is a meeting that provides a team to give positive and negative feedback to each other in a structured way that yields action items. Team members will each identify aspects of the previous week that went well, and aspects that went poorly. We typically reflect on issues ranging from technical choices, to inter-role communication, to working environment. These thoughts are then grouped into themes and the team brainstorms and assigns action items to remedy any issues. Sometimes each Retro can begin with a review of last week's action items to improve accountability.

Risk (Product)	Probability of wasting time and resources on features that don't provide business or user value.

S

Service Blueprint	A diagram of a user's path through an experience, including physical, in person, and screen touchpoints. The diagrams are key in communicating a complex flow, especially if there are multiple interactions spread over several days, weeks, or even months.
Standup	A daily short meeting (usually first thing in the morning) to discuss what was accomplished the previous day, share any info that is valuable to the entire team, ask for help, and determine pairs for the day. The meeting is meant to be as short as possible and any discussions that only involve a subset of the team are moved into separate meetings.
Style guide	A document that serves as a guide for visual design elements. More specifically, a living style guide is a living document of code, which details all the various elements and coded modules of the application.

T

Test-Driven Development	Test-Driven Development (TDD) is a software development process that relies on the repetition of a very short development cycle: developer writes an (initially failing) automated test case that defines a desired improvement or new function, produces the minimum amount of code to pass that test and finally refactors the new code to acceptable standards.

Reading List

Check out these books, videos, and articles to dive deeper into concepts and tools.

Design

- The Design of Everyday Things
  by Don Norman
- Universal Methods of Design
  by Bruce Hanington and Bella Martin
- Just Enough Research
  by Erika Hall
- Observing the User Experience
  by Elizabeth Goodman, Mike Kuniavsky, Andrea Moed
- Practical Empathy
  by Indi Young
- You're my Favorite Client
  by Mike Monteiro
- Talking to Humans
  by Giff Constable
- Balanced Team
  by Janice Fraser

Lean and Agile

- Extreme Programming Explained
  (*Especially Chapters 1-7*)
  by Kent Beck with Cynthia Andres
- Lean UX
  by Jeff Gothelf and Josh Seiden
- Running Lean
  by Ash Maurya
- Agile Product Ownership in a Nutshell
  by Henrik Kniberg

Pivotal Tracker

- Getting Started with Pivotal Tracker
  by Pivotal

Thanks for reading! If you have any questions or feedback about what you've read, please share with us at: **client-designer-playbook@pivotal.io**.