Welcome to Apprentice.io!

Welcome to apprentice.io!

We're excited to have you here. This packet contains information that past apprentices have found helpful.

We recommend looking through it now and referring back to it later. Think of this packet as your Hitchhiker's Guide during your apprenticeship. Don't Panic!

This packet is available at https://github.com/thoughtbot/apprentice-onboarding. Feel free to file a pull request to improve any aspect of it, no matter how tiny.

The Plan

Your first two months are spent learning. At the end of your second month, you'll meet with Chad Pytel to discuss whether thoughtbot is interested in hiring you as well as other job prospects at other companies that you're interested in.

Your mentor

Your first mentor's name is: _____. You'll get a new mentor each month.

This person is your guide for the month.

How am I doing?

You should meet with your mentor once a week on Friday to talk through your progress.

We encourage you to create a backpack page where you list your goals for the next week and how you did on your goals for the past week. This is a good document to discuss with your mentor each week. For an example of how previous apprentices have done this, search in Backpack.

When you switch mentors, you will have a Mentor Hand-off. This is a meeting with both mentors to talk through the previous month and the plan for the next month.

Pairing during the week

You will work with them on client projects Monday-Thursday, traveling with them to the client site if necessary/appropriate. Your mentor will pair with you during the week as you work on user stories.

Pairing is the fastest and most effective way to learn, so it should be a priority for you and your mentor. However, do not expect to pair 100% of the time. One particularly effective method of pairing is Ping-Ponging. The idea is for Person 1 to write a test, Person 2 writes the code to make the test pass. Then Person 2 writes the next test and Person 1 writes the code to make the test pass. Rinse, repeat.

Self-guided learning

Whenever appropriate or applicable to whatever you are working on, you should take time away from client work to do some self-guided learning. This can involve reading chapters in books, attending workshops, or learning about and trying out a technique/library.

We recommend starting with Apprenticeship Patterns.

Workshops

We encourage learning. We run workshops and sell webcasts (both on http://learn.thoughtbot.com), which are free for you. You can and should take advantage of these.

Use the coupon code APPRENTICE2012 to get anything on Learn for free.

Training @ Bocoup

We also have a partnership with Bocoup, where you can take any of their workshops (http://training.bocoup.com) for free as well. They offer training in Open Web technologies like HTML5 and JavaScript.

To take a Bocoup course, email Jory (training@bocoup.com). Let her know which course you are interested in and that you are a thoughtbot apprentice.

If there are other apprentices interested, it would be best to coordinate with them first and just send one email.

Getting the Most Out of apprentice.io

Expectations

Your apprenticeship is an opportunity. The more you put into it, the more you will get out of it.

There is some structure to guide you through the program and also people available to help you along the way, but the engine of this experience is your effort.

You should expect this to be a mentally intense period of your life where you are learning a great deal. Hard work pays off.

Your trail map

On your first day, fork the Trail Map (https://github.com/thoughtbot/trail-map) and edit it with your mentor to find trails that match your goals. It can be any of trails, but stick with them and track your progress. Also, submitting pull requests back to the main repo will strengthen the trail map for everyone.

Feedback

It's not easy, but you should get comfortable asking for and receiving feedback. People are nice and none of it is personal. It's a great way to develop professionally.

Tools and Weapons

Laptop setup

Your mentor will help you get your laptop set up. We recommend the laptop script available at https://github.com/thoughtbot/laptop.

Logins

We use many different websites at thoughtbot, and everyone should have accounts for them. Having access will make it much easier for you to participate in thoughtbot's internal conversations.

The list of services is below. By the end of your first week, you should have access to all of them.

Google apps

URL: http://google.com/a/apprentice.io

You should have received an email with your email (NAME@apprentice.io) and email password.

We use the calendar extensively - add someone else's calendar to keep track of what other people are doing.

Dropbox (http://dropbox.com)

You have access to the "Library" Dropbox folder. This folder contains workshops, books, papers, and other resources that we've purchased in the past and now share with everyone at thoughtbot. It's an easy way to learn something for free.

Tumblr (http://tumblr.com)

Our company blog (http://robots.thoughtbot.com) is actually hosted on Tumblr. In order to make a blog post, you need a tumblr account. When you have an account, ask anyone at the company to give you access to the blog. We *strongly* encourage you to make a blog post!

Even things that you think are silly, like a git tip that takes 3 lines and that you're *sure* everyone else knows, is worthy of a blog post. Often our most popular blog posts are the ones that we thought were the silliest. For example, http://j.mp/LP72NP is pretty dumb but got 18 comments.

We also have a semi-secret blog of Campfire quotes at http://tinyrobots.thoughtbot.com.

37signals Suite (Basecamp/Backpack/Campfire)

37signals runs a suite of applications that we use. You should have received a 37Signals invitation in your apprentice.io email. Ensure that you have access to the following:

- Campfire: http://thoughtbot.campfirenow.com/
- Backpack: https://thoughtbot.backpackit.com/
- Basecamp: https://basecamp.com/1719045

If any of those sites give you an error, talk to your mentor. Once you have access to Backpack, check out this onboarding page: https://thoughtbot.backpackit.com/pages/2661936-employee-onboarding. There's probably overlap, and we'd love feedback!

Trajectory

We use Trajectory to manage our projects. Each client has their own account, and your mentor will invite you to whichever project you are assigned to.

If you'd like to get familiar with Trajectory before your apprenticeship, you can create a free account at https://www.apptrajectory.com/plans.

Campfire

We use Campfire *constantly*. Use a desktop Campfire client like Propane (http://propaneapp.com/); it's much better than the web interface. It will ask you to register it when you start it, but you can ignore that and it will work fine.

Campfire has rooms with a specific purpose:

- Everyone: discuss things that interest everyone. You should be in this room whenever you are working.

- Code: discuss your code. When asking a question, pasting your code is encouraged.

- Design: discuss design topics or cool design that you've encountered.

- Watercooler (a/k/a WC): this is where we post animated GIFs and lolcats. Don't post lolcats in other rooms. Sometimes you'll be told "WC!" - that means you should take it to Watercooler.

- Most of the other rooms are specific to projects and are named after the project.

Designers and developers should check out the Code and Design rooms - everyone's welcome in every room.

Basecamp

Basecamp is where we put ideas that we want to discuss, like product ideas or a new Git workflow.

Backpack

We have things like general company policies and non-technical discussion and announcements in backpack. If you have a question about anything HR related, look in backpack first.

You should use your homepage in Backpack (there's a link in the Backpack sidebar) to track your progress.

Dotfiles

Dotfiles are the files in your home directory (referred to as "~") that start with a dot, like "~/.bashrc". They are used to configure various programs, e.g. "~/.vimrc" sets some global options for "vim". Take a look at thoughtbot's dotfiles (https://github.com/thoughtbot/dotfiles), and feel free to fork them. We also encourage you to check out your coworkers' dotfiles for inspiration. For example, Josh Clayton has an extensive configuration for tmux in his dotfiles: https://github.com/joshuaclayton/dotfile

Community

Part of your apprenticeship is learning to be part of a development community. This could be a Ruby Meetup Group, company, or project team.

Meeting people is easy

We are a friendly bunch and we'd like to get to know you better! That being said, it's easy to feel intimidated or uncomfortable in a new environment. Some things you should try:

- Go on coffee walks: we often have informal "coffee walks" at 3-ish, where people go for a walk (coffee optional) and chat about random topics.
- Go get lunch/coffee/breakfast with anyone in the office.
- Suggest a social event for your project team.

Friday Lunches

We order lunch for the whole office each Friday! Please let our office manager Seana (seana@thoughtbot.com) know if you have any allergies or dietary restrictions we should be aware of.

Who?

If you're having trouble matching names to faces, check out http://thoughtbot.com/about, which has pictures and names of everyone at the company (except apprentices). To see pictures of apprentices, go to http://apprentice.io/apprentices.

Development Discussions

We have weekly development discussions most Fridays from 3-4pm in a conference room upstairs. We discuss topics like how to design object oriented systems and the style guide. Development apprentices especially are strongly encouraged to attend.

Design Discussions

We hold these at 3(ish) PM on Fridays when there are enough designers. It's a place to show off what you've been working on as well as a mini-critique. Designer apprentices are strongly encouraged to attend.

Boston.rb

The boston ruby group (bostonrb.org/calendar) meets every Tuesday, alternating Project Nights (where people hack together) and presentations. It's a great place to learn something, as well as to practice your public speaking by submitting a talk. Speaking at boston.rb is great practice since it's a smallish, friendly audience, and it gives your resume a nice bump.

Get a Job (your third month)

One of our goals is to help you land a developer/designer job at the end of your apprenticeship.

Your entire third month is focused on finding a job. Chad Pytel will meet with you at the beginning of your third month to discuss your options and help you find employers. While you will still be learning, the emphasis is more on employment.

We prefer that the first two months, conversely, are focused on learning and not finding employment.

Preparing for interviews

What to expect in a technical job interview

Technical interviews are a different breed of evaluation and should be approached in a specific fashion. You should expect normal interview questions about education, hobbies, interests, and motivation for changes. You should also expect to submit a portfolio of work and do some in-person coding/design.

Preparing Your Portfolio

Every company will expect you to share some code samples as part of their application process. These take a long time to develop so consider building these up over the course of your apprenticeship. Your Breakable Toy (http://redsquirrel.com/dave/work/a2j/patterns/ is an excellent showcase for your talent.

Guidelines for developers: * Code samples should be *entirely* your own work * Some good ideas are at RubyQuiz. This goes without saying, but don't use the published answers (or look at them). * Smaller is easier to digest * Pay extra attention to getting formatting right * Make sure you've got tests for everything * Mock external services like Twitter/GitHub using ShamRack or VCR * Make sure your commit history is clean e.g. no typos, no "wip" etc * Demonstrate use of feature branches and merging, even if it's just you

Questions You Will Get on Your Ruby Code Sample: * What do you like about your code sample? Why? * What would you change and why?

Preparing for Technical Interviews

Technical interviews will run the gamut from very esoteric to very specific. You should be prepared for all of it.

Some general guidelines: * Show up 15 minutes early for the interview. If you show up late without an extraordinary reason, you are wasting their time and should expect to get sent home. * Clean it up. The classic image of the developer has them wearing T-shirts and jeans on a scooter, but if you put on a shirt and maybe a jacket, it shows you care what the interviewee thinks of you. * You should care about specific choices and demonstrate reasoning to support them * Conversely, you should also be flexible and open minded when considering other's opinions, not combatative or dogmatic. * Say so if you don't know the answer, but then try to reason through an answer *out loud* or discuss where you would find an answer. * You should research the company and your interviewer deeply. Know their online presence, read through their public code, and read up on their tweets. Bring any of this up in conversation casually wherever appropriate. Don't force it and be a weirdo. * Prepare some good questions for when they ask you for them. Good questions are not generic or open-ended, demonstrate critical thinking about something they do, understanding of their product/competitors, or are otherwise thoughtful.

Some questions to expect during a developer interview: * What's your favorite gem? * If you could remove one feature from Ruby, what would it be? * Sample SQL questions like "Given Tables XYZ, how would you select A's but not B's?" * What editor do you use? What are it's advantages and disadvantages? * What causes brittle tests? * What is a stub/mock/fake? When would you use each one?

Our internal list of interview questions is here: https://thoughtbot.backpackit.com/pages/2883297-technical-interviews

How to Reach Out to Companies

Wherever possible, you should have someone you know introduce you to to a company. It gets you out of the pile of resumes/submissions and puts a credible face on your candidacy.

Everyone will ask you what apprentice.io is about. Prepare a 2-3 sentence description of what it's like.

A good email should be short and to the point. "Hey, I'm an apprentice at thoughtbot and getting prepared to look for a job as a developer/designer. I'm a big fan of your company because of XYZ concrete reasons. If you have some time to meet or speak on the phone, I'd love to ask you some questions about your company."

Networking

Networking is probably the most important key to success for every single job. If you network consistently and present yourself as a helpful, honest person, jobs and opportunities will come flowing to you.

Basics of Networking: * Do it purposelessly - get to know people when you don't need anything from them, then get in touch with them when you do. * Do it frequently - try to get out and do one thing per week where you meet random people. * Get comfortable telling your story. "So what do you do?" is the #1 question. Respond in a positive, short way. * If you don't have anything to talk about with someone, keep asking the other person questions. * Use "Well I'm going to go get a drink" whenever you don't feel like talking anymore to someone. * Email *everyone* you get a card from to say "Hey" and you're glad you met. * Give your card out liberally. If you don't have one, get some made. * Networking is also about helping people. If you know two people who should meet, introduce them. * Stay in touch. If an article reminds you of someone or you think they would find it interesting, send it to them. * Set up periodic reminders to prompt you to get in touch with people you like and who can be helpful to you. Every 2-3 months is appropriate. Try followup.cc * Everyone will take a coffee meeting with you. Ask them.

If you don't know anyone at a company of interest, you should get to know them before you apply.

Find people in common using LinkedIn or asking your colleagues. thoughtbot has friends all over - it's likely someone at thoughtbot knows someone at the company you're looking at.

Evaluating Potential Employers

Some things to think about: * Product shop vs. Development Consultants - very different jobs. * Recent turnover * Good feeling and positive attitude from interviewers * Refreshed-looking interviewers * Culture of learning/teaching * Test-driven culture. This signals high-quality code

Improve apprentice.io

We encourage you to propose changes to the apprentice process, including hiring, onboarding, and anything else that you think could be better. Everyone at thoughtbot is constantly trying to do a better job and we welcome feedback.

As an apprentice, you have much better insight into what could make the process better for you than does someone who's been working here for five years.

Feedback is always, always appreciated on everything at thoughtbot. If you have feedback on the overall apprentice program, please contact Gabe Berke-Williams, the head of the apprentice.io program.

This page is intentionally left blank. Please write down interesting things you've learned, tips, and areas where apprentice.io could improve. We encourage feedback on apprentice.io and thoughtbot.