

UX IMMERSION – TASK 1.8

Thinking Like a UXer: Lean UX vs. Agile UX

- By Marlon McDonald

► **Teams Involved**

- UX Design, UI Design, Web/App Developers, Product Management, Customer Service, Finance/Accountant(?), Marketing, Industry Professionals, Beta Testers

► **Possible Features to be Tested**

- Chat system – video, voice, chat feature
- Tip feature – easy way to tip Experts after advice stage
- System for pairing the right expert to user – 3/4 multiple choice questions that ascertain what you need and filter you to the right expert for your needs (should still give you a choice of experts, so you can peruse their cv)
- Screen after pairing system that details the expert's credentials – think of it like Uber or My Taxi's driver rating system

INITIAL IDEAS & NOTES

- ▶ **Expert Pairing System** (EPS): User's don't want to spend excessive amounts of time hunting down the appropriate expert for their needs, so there should be a system put in place that automates the pairing, to an extent.
- ▶ **Tip System:** Artist's all share one common knowledge: struggling to make ends meet whilst chasing "the dream." So, we need to create a culture of paying forward. Yes, our experts get paid for their time and advice, but tips help to go a long way.

HYPOTHESES

- ▶ **Build:** With an idea of how the EPS will function, first we'll need to decide whether we want to build the plugin ourselves or buy the code and augment it to fit our product. After choosing to build it in-house (with an eye to leasing the plugin for extra capital) we head into our first Sprint Goal.
- ▶ **Sprint Goal/SCRUM 1:** Firstly, one team will create a list of necessary questions to figure out how to connect users to experts. Another team will work on developing the code needed to facilitate the pairing. And the last team will figure out how to make something that sounds so boring visually gratifying within the webapp. SCRUM-ing now should lead to fresher minds working passionately on the product.
- ▶ **Measure:** After creating the plugin, we will test it on industry professionals and beta testers and scrape quantitative data to learn how well the plugin functions and whether there is scope for it. Should the user simply hunt down their own expert? Should the app just connect you to someone in the applicable field?
- ▶ **Learn:** This is the moment where we learn whether the hypothesis was successful. If so, we'll go back to the drawing board and trim the fat, as it were. Maybe add more questions for deeper and more succinct selection of experts? Or, if it wasn't successful, we'll isolate why and start from the top again.
- ▶ **Sprint Goal/SCRUM 2:** This is where we'll rework the hypothesis, either streamline the EPS, or amend it majorly based on the data gained from the **learn** phase.

FEATURE 1: EXPERT PAIRING SYSTEM

- ▶ **Build:** Now that we have a subscription model in place and an idea of how and when our experts will be paid, a big part of what we're trying to build is a pay-it-forward ideal that promotes tipping experts, because it's tough to make a living in the arts. If a person has assisted you majorly, why shouldn't they earn a little extra on top of the warm fuzz of helping someone in need? We can also use the system to tip ourselves with survey data, by prompting users to answer quick questions about their experience using the product.
- ▶ **Sprint Goal/SCRUM 1:** First we'll get our teams discussing what it means to pay it forward and how we can implement that into the app. Then, one team designs the flow and look of the system, one programs it and our fledgling finance team will discuss how to keep tips bitesized enough to not feel like an extra financial burden. We'll SCRUM after the build phase because ideas are fresh in the mind and, hopefully, the meetings will have pumped everyone up enough to create excited work.
- ▶ **Measure:** Once again, we'll get the feature out to beta testers and industry professionals and see, on average, how much users tip, if they even tip at all. Does the system make sense? Should we keep it? Is the survey prompt annoying?
- ▶ **Learn:** Let's look over the results and feedback now. How well did the tip system function? If well, we'll take a look at potentially cleaning it up a little. And if it didn't work, it's back to formula to figure out a different way of creating a culture of paying it forward.
- ▶ **Sprint Goal/SCRUM 2:** Amend, trim or rethink altogether? Let's have a look at our initial prototypes and feedback and see if there's anything we can add or eject. We'll SCRUM after the **learn** phase, because we've figured out what needs to be done to move along into the next phase.

FEATURE 2: TIP SYSTEM

- ▶ The part of the day that should be treasured! All of our teams, or representatives of, come together for hot or cold drinks, snacks and the beauty of team work. This is where we'll outline our **sprint goals** and dispatch them to teams, all interview and survey data will be cross-examined and discussed, and we'll also discuss whether the project is still sparking joy for everyone involved.
- ▶ We'll look over user stories, discuss our personas and whether they're the right fit for the project, and generally air all grievances in a safe space. Short presentations will be held to see where each team is at with their part of the puzzle, to which we'll question and applaud everyone for good work done and eat many Danishes.

DAILY STANDUP MEETINGS