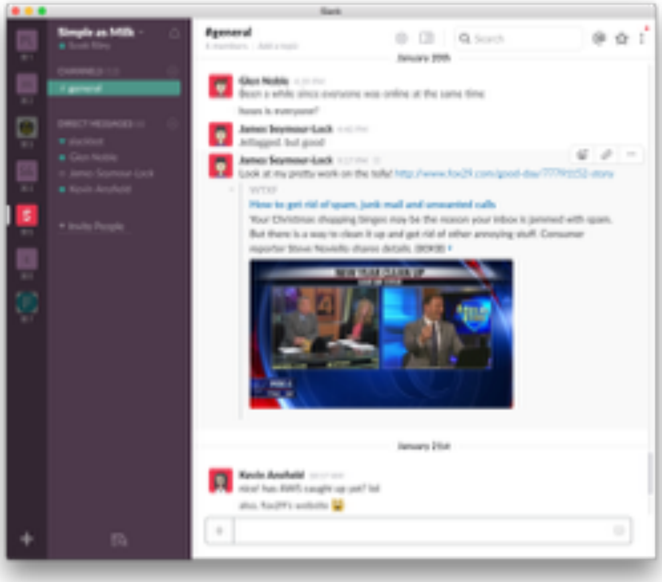
Communication & Feedback

As a remote project, we'll have a reliance on mostly asynchronous communication. This means that discussion, feedback and collaboration can happen without trying to arrange times where everyone involved in the project is available, helps us stay productive, and avoids an overload of meetings or big reveals.

Slack – day-to-day communication

Most of the general discussion around our project will take place in Slack. If you haven't used it before, Slack lets us set up what's essentially a big ole chat room for project discussion. We'll use it for daily updates, general questions, discussing inspiration and sharing rough ideas. Slack has 'channels' too, which let us create individual places to talk about specific things, like sites we like, development progress, design ideas, etc.



January 20th

Glen Noble
Been a while since everyone was online at the same time
hows is everyone?

James Seymour-Lock
Jetlagged, but good

James Seymour-Lock
Look at my pretty work on the tofu! http://www.fox29.com/good-day/[illegible]-story

WTXF
How to get rid of spam, junk mail and unwanted calls
Your Christmas shopping binges may be the reason your inbox is jammed with spam. But there is a way to clean it up and get rid of other annoying stuff. Consumer reporter Steve Novello shares details. [illegible]

January 21st

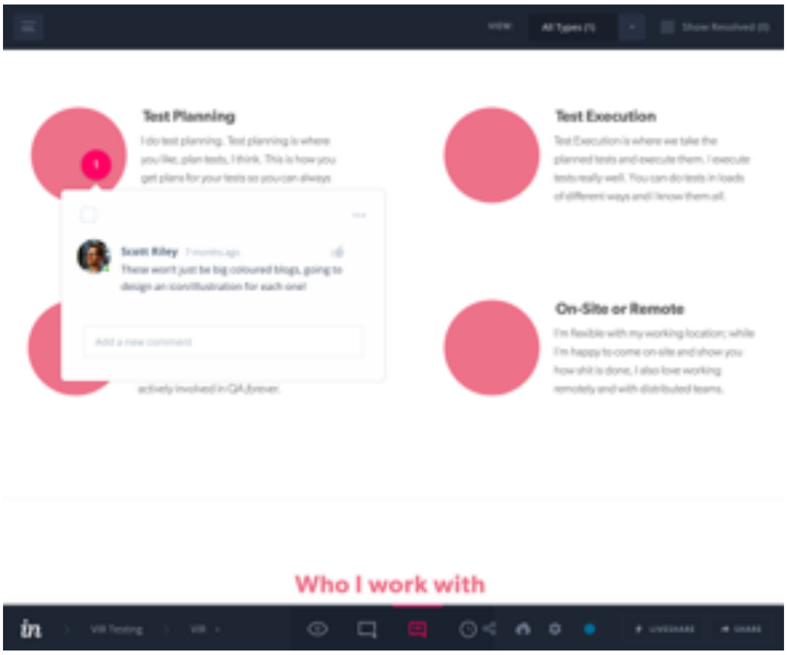
Kevin Anofield
nice! has [illegible] caught up yet? lol
also, fox29's website 😱

By using Slack as the basis for our communication, we make sure that every question, answer, suggestion, etc. is in one central place, written down and searchable. This means we don't have to worry about some of the common problems with silo'd, face-to-face meetings and anyone involved with the project can catch up in their own time.

Slack is also the perfect replacement for 90% of email communication, so no more long reply-all email chains and out-of-sync responses.

InVision – prototypes and feedback management

InVision is a lovely tool that lets you upload static designs and link them up as rough prototypes. It also has a super great commenting system that makes asynchronous feedback an absolute breeze.



Test Planning

I do test planning. Test planning is where you like, plan tests, I think. This is how you get plans for your tests so you can always

Scott Riley 7 months ago

These won't just be big coloured blogs, going to design an icon/illustration for each one!

Add a new comment

actively involved in QA forever.

Test Execution

Test Execution is where we take the planned tests and execute them. I execute tests really well. You can do tests in loads of different ways and I know them all.

On-Site or Remote

I'm flexible with my working location; while I'm happy to come on-site and show you how shit is done, I also love working remotely and with distributed teams.

Who I work with

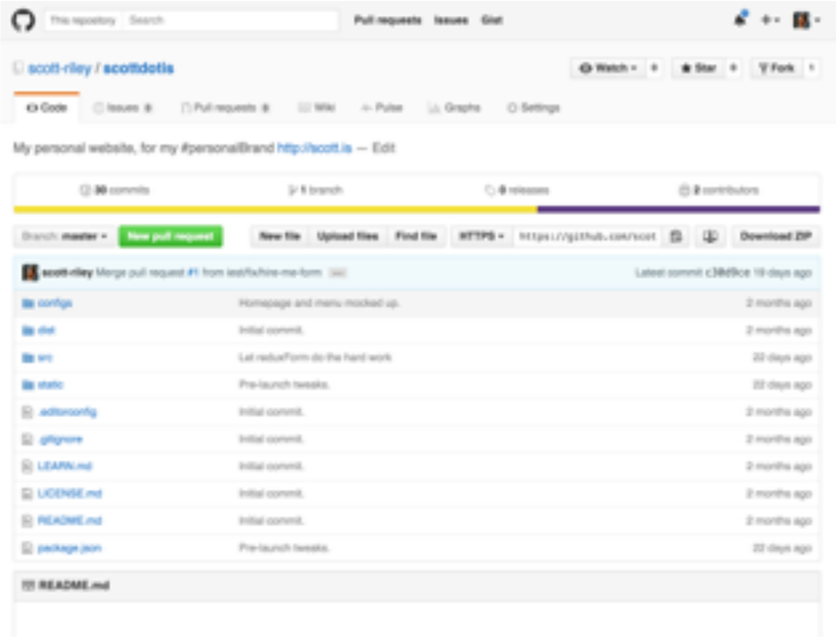
For this project, InVision will be used to get rough ideas up early and start getting feedback as quickly as possible. There won't be any agency-style 'big reveals' where you hear nothing for 2 months and suddenly get shown a finished product. InVision allows for super-specific feedback and discussion, letting you drop a comment onto any part of a design to start discussing or providing feedback.

As well as allowing for extremely efficient feedback, InVision also allows us to track all our design changes, so anyone involved with the project can look at the InVision project's history and see how we got from A–B–C–etc. – this is extremely useful as eliminates 'opaque' feedback that you'd get from a face-to-face chat or email chain, allowing us to look through every iteration and see what feedback led to a specific change.

InVision lets us have all our feedback in one place, specifically attached to the areas we're discussing. No more digging through emails, chats or meeting notes to find a specific piece of feedback. Leaving feedback in this way encourages conversation and ensures your feedback rarely gets ignored or lost.

GitHub – version control and code collaboration

All code for our project will be in a Git repository, hosted on GitHub. By using version control from the beginning, we can ensure that our code is always revertible, as well as allowing for collaboration from the get-go.



scott-riley / scottdotis

My personal website, for my #personalBrand http://scott.is — Edit

scott-riley Merge pull request #1 from lest/fix/hire-me-form … Latest commit c38d9ce 19 days ago

configs	Homepage and menu mocked up.	2 months ago
dist	Initial commit.	2 months ago
src	Let reduxForm do the hard work	22 days ago
static	Pre-launch tweaks.	22 days ago
.editorconfig	Initial commit.	2 months ago
.gitignore	Initial commit.	2 months ago
LEARN.md	Initial commit.	2 months ago
LICENSE.md	Initial commit.	2 months ago
README.md	Initial commit.	2 months ago
package.json	Pre-launch tweaks.	22 days ago

Hosting our project’s code in a GitHub repository means that code changes, iterations, big fixes and issues are all self-contained and searchable.

As well as allowing for streamlined developer collaboration, there are tonnes of hosting services out there, such as Heroku, that let us automatically deploy our code to a staging server/environment, meaning we can have a staging server set up early into the project that anyone involved in the project can view.

Summary

Asynchronous communication frees us up to do our jobs without constant interruptions and meetings.

Slack lets us have a ‘virtual office’ where we can discuss anything without arranging chats or meetings.

InVision lets us create prototypes and discuss wireframe/design ideas directly inside the prototype, removing the need for long, design-by-committee-style feedback meetings and ‘big reveals’, focussing instead on early, rough ideas and super-specific feedback.

GitHub lets us host our code in repositories, making it really easy to collaborate on code, search through iterations and deploy to staging environments.

Face-to-face meetings/calls

While the majority of our communication will be asynchronous as discussed above, there will absolutely still be times where real-time meetings are necessary. The point of this process is not to completely eliminate these kind of meetings, but often they'll be a 'last resort' or a scheduled catch-up/stand-up meeting.

Face-to-face meetings or calls to gather feedback, or to provide small status updates, are no longer necessary with the tools we have now. With the above tools and process, we make sure that when we do have a face-to-face meeting, Skype chat, or call, that we're doing it because it's the most efficient method of discussing a specific topic, rather than defaulting to these kind of meetings for everything.