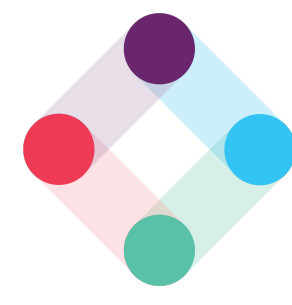


Iterable is a cross channel customer engagement platform



ITERABLE

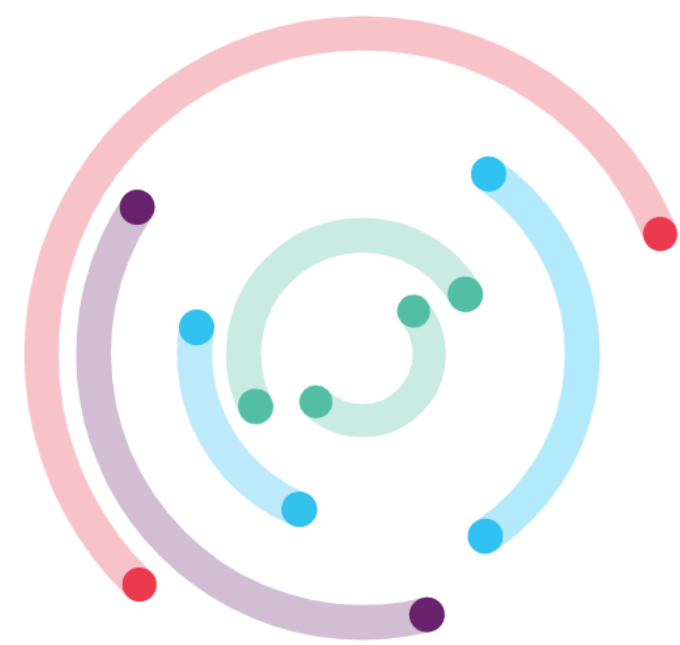
The Iterable marketing department needed to increase the speed at which they develop and iterate on their marketing website in order to achieve key company growth metrics. They also need the flexibility to change directions quickly and increase or decrease team output as needed. Hiring in house is a lot of investment, slow and difficult to quickly scale up and down. Therefore they engaged with BAC to bring needed expertise and resources quickly.

APPROACH 1

Enhance and Augment Team NOT Over the Fence Development

At BAC we try to avoid the traditional “over the fence” agency development model. In an ideal engagement we quickly learn what internal resources a team has and what it’s upcoming objectives are and then augment the team with additional experts: devs, designers, pms as appropriate. From there, the team (both BAC and client) work together to quickly and iteratively achieve objectives.

Iterable started with an internal PM and a developer. Initially we augmented their team with 1 fulltime developer and a part time architect to streamline workflow chose appropriate technologies and increase dev speed. As the teams needs grew we quickly added a second fulltime developer and designer.



RESULTS

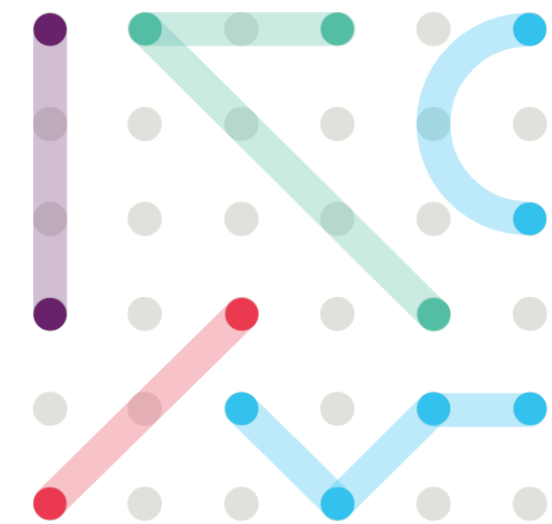
- faster iteration and flexibility
- better communication
- more learning

APPROACH 2

Best Practices SDLC and CI/CD

In order to work together effectively, especially when team members are in different time zones, we need a set of lightweight team processes. Each team is different so we start each engagement by tailoring a team process selecting from our experience of best practices.

Iterable’s internal marketing dev team was new and small and therefore had very little process in place so within the first week of the engagement we worked together to design and implement a SDLC and CI/CD pipeline. Here’s what it looked like:



SCRUM

- 1 Project Tracking - Asana
- 2 Standup - 15 mins daily to update, callout issues and setup needed convos after
- 3 Biweekly Planning + Retro - Address team issues and improvements and plan for the next 2 weeks

BRANCHES & ENVIRONMENTS

We setup buddy.works pipelines and environments so that the following branches automatically deployed to the corresponding environment: Work is done on a feature branch off of master.

Git Branching

- Dev Branch → Dev Env
- Master Branch → Staging Env
- Prod Branch → Prod Env
- Feature Branch → Per Dev Env

Developers can push whatever they like (cherry picked code/branches) they want to their dev branch which is automatically deployed to their dev environment so they can get quick feedback. Once complete and tested the developer makes a Pull Request to master. After code review and acceptance, the PR it is merged to master. Master is automatically deployed to staging. When the marketing team decides they’d like to release we merge the master branch to the prod branch and it is automatically deployed.

RESULTS

- quick feedback and faster release cycle
- safer releases and less bugs
- more flexibility to change work
- predictable work and streamlined communication
- learning from feedback and code reviews
- easy to add/remove additional resources to team