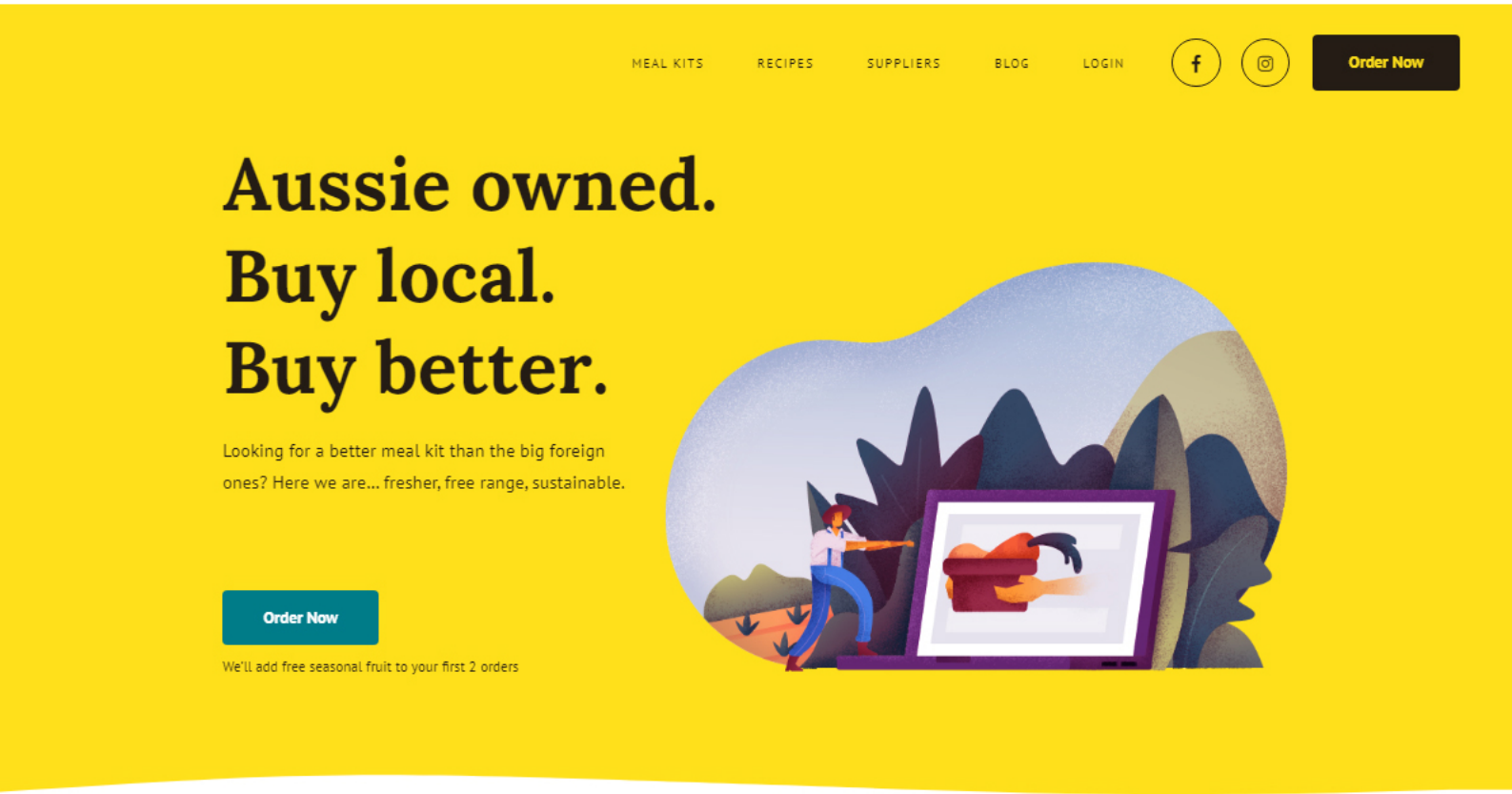


EXECUTIVE SUMMARY

Food Delivery Customer is a family-run, local Meal Kit Delivery company based in Melbourne. Running since 2015, they pride themselves on working with Australian producers and delivering food quickly from farm to table. Founded by Simon Kahil, this Melbourne based Meal Kit service is founded on the belief that fresh produce is one of the most important things for a great meal. When you order from Food Delivery Customer, you really do feel like it's arriving from a family farm with the whole experience being just a little bit rough around the edges.

Food Delivery Customer have a really simple website, where you can learn more about their suppliers and the Meal Kit service they provide. Unique to Food Delivery Customer, you are able to order 1 person Meal Kits as well as the standard 2 and 4 person options.

THE CHALLENGE

This is was the ‘Challenge accepted’ project for us. They have a legacy PHP (Symfony) project and they wanted to implement CI & CD on AWS.

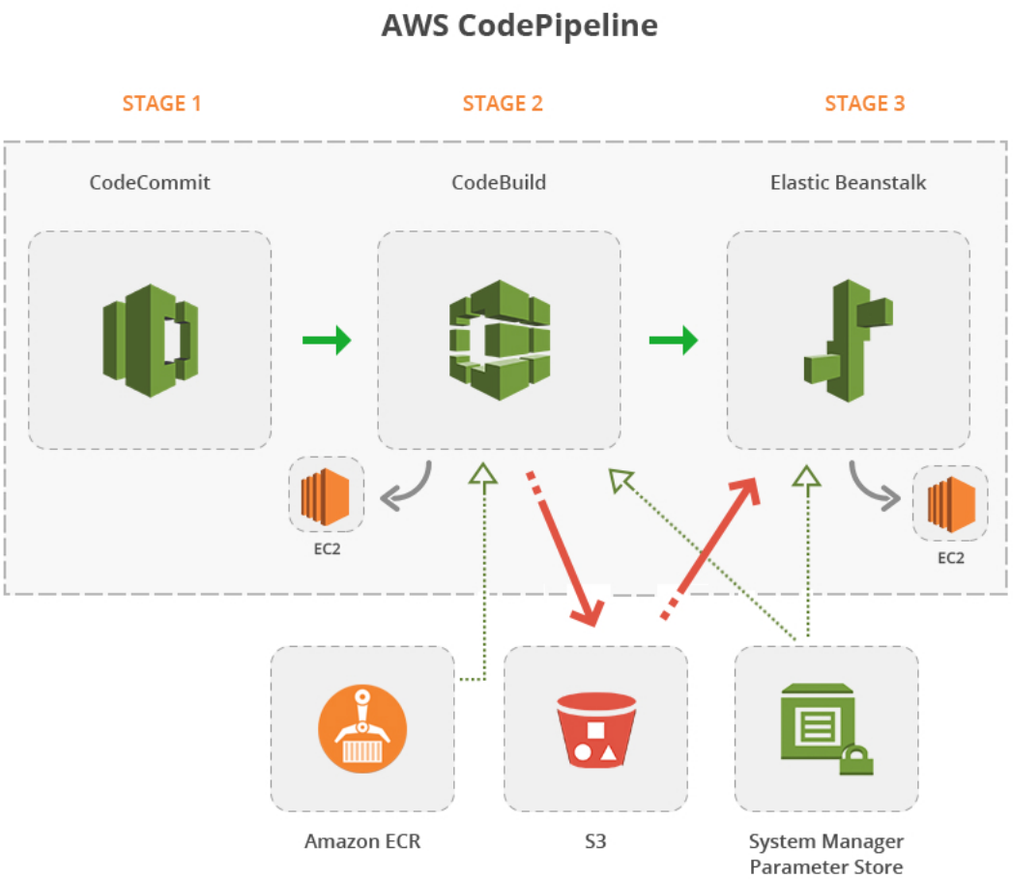
The major challenges were

- It was a legacy PHP project
- It Couldn’t be updated to the latest version of PHP
- We cannot update the Symfony Framework version to the latest.

OUR SOLUTION

We started our working with a couple of options and based on our experience and also due to a request from a client, we first tried with Bitbucket as a version control tool, BuildKite as a build tool and AWS S3 to store .zip bundle created by BuildKite.

However, to increase efficiency and keep all tools under the same umbrella, we went to our best friend AWS and we could successfully achieve our desired goals.



Continuous Integration is becoming a de-facto standard in today's software industry. Many organizations have used CI/CD successfully to continuously integrate code from various developers and deploy it on dev/integration/production environments. There are several benefits to this approach, like,

We used a few AWS services to achieve this

- **CodeCommit** – Repository for codebase. **(Version Control)**
- **CodePipeline** – To automate pipeline. (**Build**)
- **CodeBuild** – To install all dependencies and generate a .zip bundle that is ready for deployment. **(Integration and Unit Test)**
- **Elastic Beanstalk** – To deploy the application on Dev/UAT server. **(Deploy)**
- **Amazon Elastic Container Registry (Storage for Docker image)**
- **AWS Systems Manager** Parameter Store **(Storage for Configuration Data)**

THE BENEFITS

Now that the entire development cycle is automated on the AWS environment, they have benefited from a highly available service and very fast deployments. The major benefits that we have achieved are

- Happy customers and development team.
- Fast and quick update release.
- The huge cost and time saving on the development cycle.
- Developers can quickly find out integration errors and feedback about their work.
- They save time they would otherwise spend on manual integration and uploads.
- Dev/UAT environment is always up to date for developers, QA, and reviewers.