CONFIDENTIAL

Geo-Aware Food Sourcing v 1.1

Document Status	DRAFT
Authors	Daniel Anthony, Chief Product Officer Natasha Jarson, CIO
Technical feedback	Jack Harkness, Operations Lead
Project sponsor:	Ash Callum, CEO
Expected funding requirement for Phase 1	Budget for two quarters
Current secured budget:	First quarter
Associated architectural repository	See here

Objectives of This Document

- Provide context on the project's drivers.
- Define KPIs and success criteria for the initiative.
- Define commercially sensitive timelines.
- Clarify high-level objectives which satisfy Foosus' product goals.
- Ensure alignment between technical and product teams.

Background Context and Drivers

Foosus' legacy platform has reached a point where it can no longer support the company's plans for growth and expansion. After several years of development, our complex technical solution is not scaling with the business and possibly impeding our growth. Market and commercial analysis show that our customers want to buy local and support local growers.

Our competitors have not targeted this niche. We want to build on what we've learned over the past three years and create a platform which will connect consumers with local producers and artisans for all of their needs. Key business objectives are as follows.

- Leverage geolocation to link suppliers to consumers and propose available products which are close to where consumers live. A distance calculator should be included for consumers to find the nearest suppliers.
- The architecture must be scalable so that our services can be deployed to various geographies across selected cities and countries.
- Enhancements and other changes to the production systems should minimize or eliminate need for downtime for deployment.
- Our solution must be available by our suppliers and consumers wherever they are located. It must embrace mobile and desktop devices. It must account for bandwidth constraints for cellular networks as well as high speed internet connections.
- It must support different users (e.g. suppliers, back office, consumers) with the features and services unique to those user categories.
- Deliverables need to be provided at regular intervals in order to get the new system up and running quickly and add new features over time.

Initial research on architecture best practices show several best practices with minimal technical risk. These include potential microservices, standards for supporting web and mobile solutions, standard databases, and similar approaches.

Project Objectives

The existing platform can't be decommissioned while we drop tools to build again. Further, to continue to accept sign-ups for new suppliers and consumers, we need to unshackle new delivery from the existing architecture and infrastructure to minimize outages.

Our goal is to unleash the creativity and experience of our technical teams; allowing it to shine through in a new platform designed to take our customer base into the next million registered users and beyond. We want to drive Foosus marketing campaigns in several major cities, with the confidence that our platform will remain usable and responsive while delivering first-class customer experience.

We have identified several high-level objectives which need to be satisfied with any new technical direction taken towards improving our business capability.

The new platform should allow our product teams to rapidly innovate by repurposing existing solutions, experimenting with change, and providing ease of integration with internal and external partners.

Scaling with Our Customer Base

The technology stack must be designed to scale naturally with our customer base. We've witnessed outages caused by our software system not being able to cope with spikes in customer usage or marketing drives.

The performance constraints of the existing system could not cope with the expected engagement and growth following our planned marketing drives.

If the system gets overloaded, logged in users should still have access to all services. This has not been the case to date!

A Secure, Usable, and Responsive Platform

We plan to drive Foosus campaigns in several geographic regions and do not just want the platform to cope, but naturally scale to handle increased loads. Further, we need the platform to be easy to localize and meet the usability requirements which best engage our customers.

In the past, we've focused on usability at the cost of security. This has nearly impacted our reputation many times. We want to avoid the risk of brand damage and require an approach that ensures security with every increment to our platform.

Transparent Technology

It is no longer acceptable to bring the platform down for a new release or a change to the database schema. In a market where we're targeting cities around the globe, we no longer have a quiet hour in the middle of the night. We are *always on!*

Every release needs to be small, low-risk, and transparent to our users as well as being accessible at any time and place. We succeed when our users can easily access our services and love our product.

Users in different regions should expect similar performance. We want to target consumers in specific geographies over low bandwidth connections (e.g. cellular) as well as high speed networks. Any solutions to be responsive to this.

Scalability for Future Growth

In the past year, 12 of our outages were due to one or more teams releasing large changes that didn't behave as expected.

We have also had issues with integrating work done by different teams working on unrelated changes to our platform.

We should not have these issues as a small company. The problem lies with the time it takes before any new software version is seen by other teams, or tested in our live environments. We need to close the gap between when a line of code is written and when it gets proven in an integrated environment. This can also help us learn about our customers' reactions to new features as we develop them.

Experimentation

Our product teams would like to be able to perform multivariate or A-B to try and compare different solutions with users.

To achieve this, we need visibility into how our software is used and the ability to reverse architectural decisions when it is still cheap to do so. Or iterate towards a platform which meets our ability to experiment with new products in line with our overarching business goals.

Success Metrics

| Metric	Desired Change in Metric
User sign-ups per day	A 10% increase
Food producer sign-ups	Growing from 1.4/month to 4/month
Average feature* cycle times	Reduced from 3.5 weeks to under a week
Rate of P1 production incidents	Initially: Reduced from 25+/month to under 1/month.

Initial Wires for Geo-Aware Food Sourcing

Food Supplier Search

The most criticized feature of the existing platform is our food supplier search feature. Current learning indicates that 48% of journeys start with a search but drop off before a customer has even looked at a Food Offer from one of our suppliers.

The UX and CX teams have tested a new design which relies heavily on prioritizing local producers and artisans based on geo-location.

The intended outcome for Q1 is to improve our search capability using the target state technology stack.

Initial testing of wireframes has produced the following two structures for desktop and mobile breakpoints, both of which give prominence to a supplier's proximity to the user.
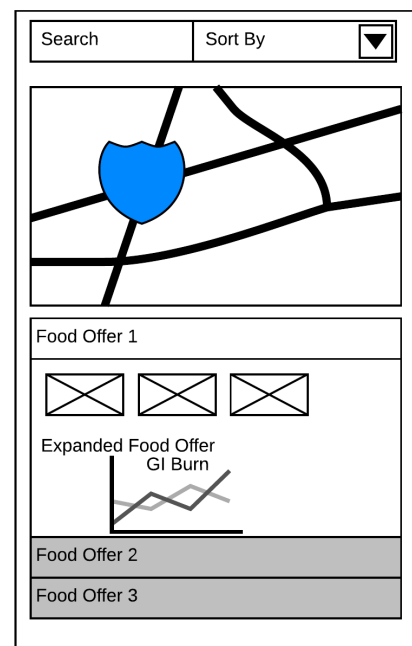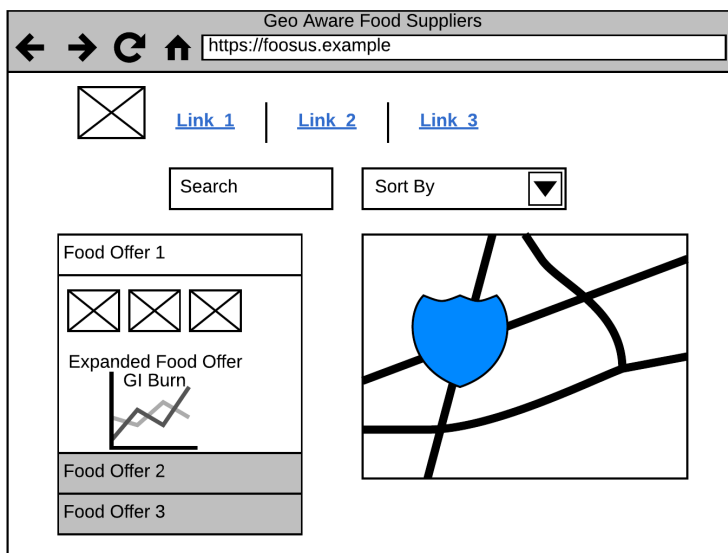
When creating backlogs, product teams can elaborate on specific behavior; however, any new design should account for:
- Location of food offers provided by suppliers.
- Proximity to the currently searching user.
- Visualization of secondary, industry-based, statistics relating to the food item; for instance, details about its Glycemic Index.

Food Offer Ordering

While our current order process is under review, it has generally received positive feedback, and we plan to base any new designs on the existing process, which involves the following steps:
- Search and identify foods required.
- Add food offers to basket.
- Seek agreement to pay on delivery.
- Email delivery instructions and a commission invoice to food supplier.

The long-term product vision is to modify those last two steps so we can:

- Integrate with third-party payment providers.
- Handle all communication with Food suppliers in a custom UI.

Any platform should support our eventual move to such solutions; however, these have been deprioritized for the time being.

Assumptions

- Rather than investing more into the existing platform, we will preserve it in maintenance mode; no new features will be developed.
- The new architecture should be built with current technologies in mind with the ability to adapt to new technologies as they become available.

- As teams are attached to the existing platform, leaders need to avoid taking false-shortcuts by implementing new behavior into the existing system.
- The initial offering will involve co-existing two platforms and empirically ramping up the volume of users who migrate to the new platform as the product evolves. This will be incremental as functionality evolves.
  - For instance, early users might opt-in to utilize the new search features integrated with the existing checkout process.
- GEO-targeting, if modeled early in the new platform, will allow further innovations based on a user or food supplier's location.
- Tailoring a lean architecture approach can help accomplish this roadmap, so it doesn't disempower the teams or threaten rapid release cycles.

Open Issues

- How can we maintain team participation and ownership while still benefiting from more explicit governance?
- Metrics to continuously measure our platform's architectural health, so we can avoid the previous issues.
- Cloud vs on-premise hosting has been a hot topic but requires guidance to ensure that any decision is data-driven.
- Until a new guardian for our enterprise architecture is brought in, we don't know if our initial architectural vision is achievable. Ideally, we will find a new architect who can create an appropriate Architectural Requirements Specification.