# Herramientas usadas

A continuación, se presentan todas las herramientas necesarias durante el desarrollo de la plataforma Sigma.

* **GitHub**: Repositorios de código
* **Waffle.io**: Administración de proyecto mediante *Issues*
* **Slack**: Comunicación entre el cliente (Stratos Agein) y el desarrollador.
* **GitBook**: Para la documentación formalizada.
* **Microsoft Azure**:
  + **Máquinas virtuales**: La cuales contienen Sigma.Data y Sigma.API
  + **Azure Website**: Contiene Sigma.Website
* **Codeship**: Continous Delivery para Sigma.API
* **Appveyor**: Continous Deliver para Sigma.Website

## GitHub

Cada usuario colaborador debe tener un usuario el cual lo podrá crear en <https://github.com/join>, es importante poner una imagen de perfil donde se vea el rostro del colaborador para que el desarrollador quien decidirá los permisos pueda reconocer a dicho colaborador.

La organización en GitHub está creada, sin embargo, si desean dejar los repositorios privados al público es necesario actualizar la licencia que se tiene; este proceso se hace en <https://github.com/organizations/StratosAgein/billing/plans> (solo es visible para miembros de la organización) en el que para este proyecto solo se recomienda actualizar a Bronze.

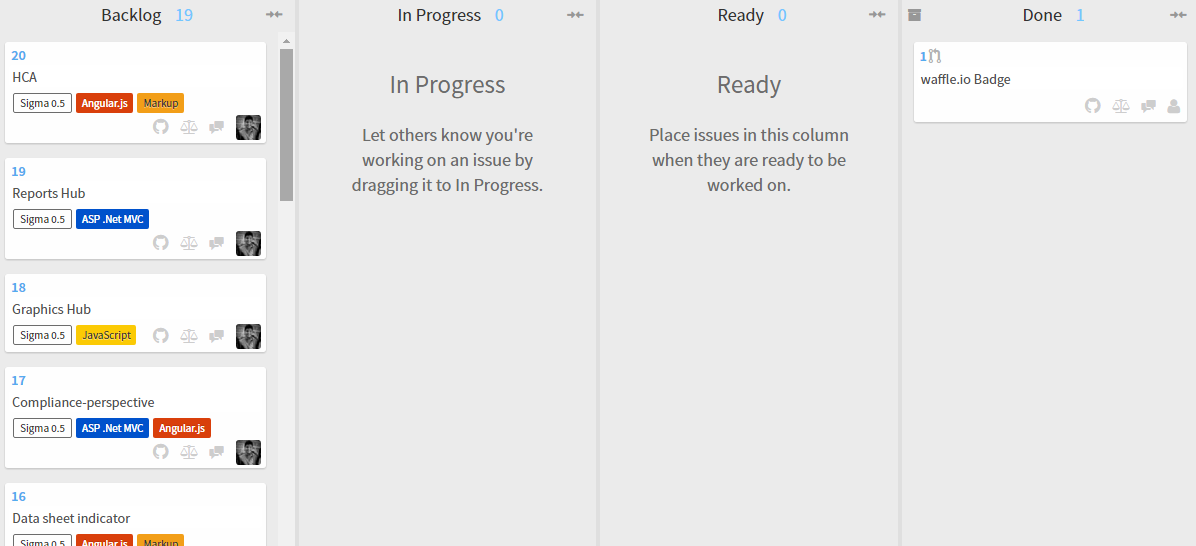
Se requiere el logo de la empresa para distinguirlo a nivel global, esto generará posicionamiento en el “gremio” de desarrolladores de software; empresas como Google, Facebook, Microsoft, entre otros gigantes están en esta plataforma.

<https://github.com/StratosAgein>

## Waffle.io

Es un tablero kanban donde se lleva un control detallado de todos los *issue* (requerimientos) contenidos en GitHub, el cual irán reportándose el cambio de estado en plataformas de comunicación como Slack que será explicado más adelante en el presente documento. También ayuda a planificar prioridades y ver el estado actual del proyecto.

Estado actual de Sigma.Website:

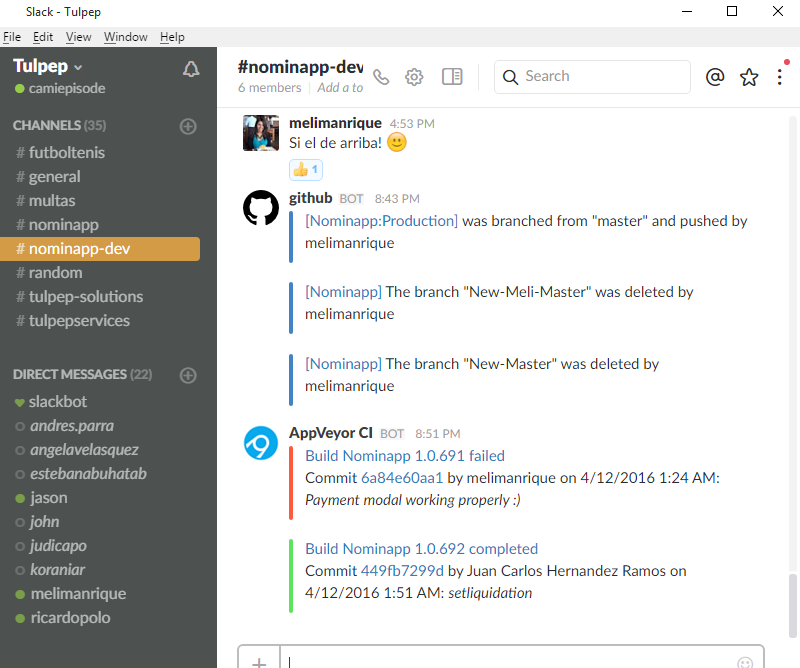


## Slack

Plataforma de comunicación por excelencia entre empresas de tecnología, se basa en comunicaciones asíncronas como Whatsapp y permite la integración con herramientas de desarrollo como GitHub, Waffle.io, entre otras.

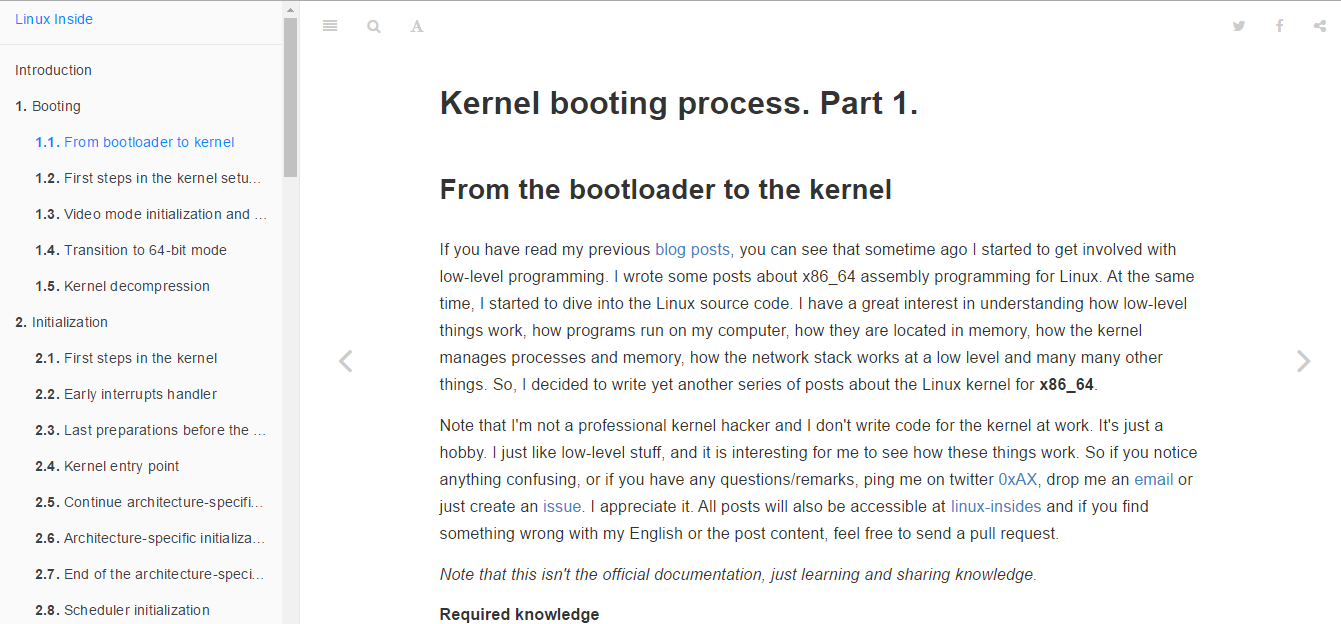
Es importante mantener este canal de comunicación aislado para temas que tienen que ver con el proyecto y la empresa para no utilizar canales menos profesionales, como Facebook Messenger, Whatsapp y otros.

Ejemplo de la herramienta en Tulpep S.A.S.:



## GitBook

La documentación en un proyecto de software es vital, tanto para continuar el proceso de desarrollo como para crear distintos productos. Además, es de las mejores prácticas de desarrollo de software para que este no se vuelva una caja oscura donde solo el desarrollador de este proyecto conozca qué es lo que contiene. Por lo tanto, una de las mejores herramientas llevar a cabo dicha tarea es **GitBook,** un ejemplo de Linux Insider:



## Microsoft Azure

Esta plataforma creada por Microsoft es su propuesta a la nube y en la actualidad una de las más fuertes según varios consultores en tecnología como The Verge o el cuadrante de Gartner (<https://azure.microsoft.com/en-us/blog/microsoft-the-only-vendor-named-a-leader-in-gartner-magic-quadrants-for-iaas-application-paas-cloud-storage-and-hybrid/>), dada esta información se lleva a cabo el desarrollo y el despliegue de todas las aplicaciones en esta nube. Se crearán varias máquinas virtuales para dicho fin y un servicio llamado Azure Websites.

## Codeship & Appveyor

Estos dos proveedores de Continous Delivery proporcionan un medio automatizado de enviar a un ambiente de producción la versión más estable de desarrollo, ambas cumplen casi el mismo fin pero con distintos propósitos; mientras Appveyor se especializa en productos Microsoft, Codeship tiene un portafolio más amplio y el fin con el cual se usa es el despliegue por FTP en la máquina virtual de Sigma.API.