

“CADHU”
Versión 1.0

Documento: Handbook de Arquitectura

Autores: Alejandro Salmón Félix Díaz
Filiberto Francisco Vázquez Rodríguez
Santiago Valencia Corona
Luis Rodriguez Santos
Marco Antonio Luna Calvillo
Mauricio Hernandez Siles

30 de abril de 2018

Bitácora

| No. de Versión | Cambio | Autor | Aprobado | Fecha de cambio |
|----------------|------------------------|---------------------------------------|----------|-----------------|
| 1.0 | Creación del documento | Luis Rodriguez Alejandro Salmón | | 02/05/2018 |

Contenido

| | |
|-----------------------------------------|---|
| Introducción | 4 |
| Descripción general | 4 |
| Arquitectura General | |
| Flujo Lógico | 4 |
| Diagrama despliegue | 4 |
| Diagrama de vistas y componentes | 4 |
| Diseño base de datos | 4 |
| Hardware/Software | 4 |
| Dependencias del sistema | 4 |
| Especificaciones técnicas | 4 |
| Pruebas de unidad e integración | 4 |
| Configuración servidor | 4 |
| Actualización servidor | 4 |
| Configuración dominio | 4 |
| Habilitar HTTPS: Certificado SSL | 4 |
| Glosario | 4 |

1. Introducción

El objetivo de este manual es brindar un documento de consulta el cual facilite el desarrollo y mantenimiento del proyecto CADHU. En este documento se encontrarán todos los artefactos utilizados en la fase de diseño y desarrollo, para que cualquier persona pueda agregar funcionalidad o darle mantenimiento al sistema.

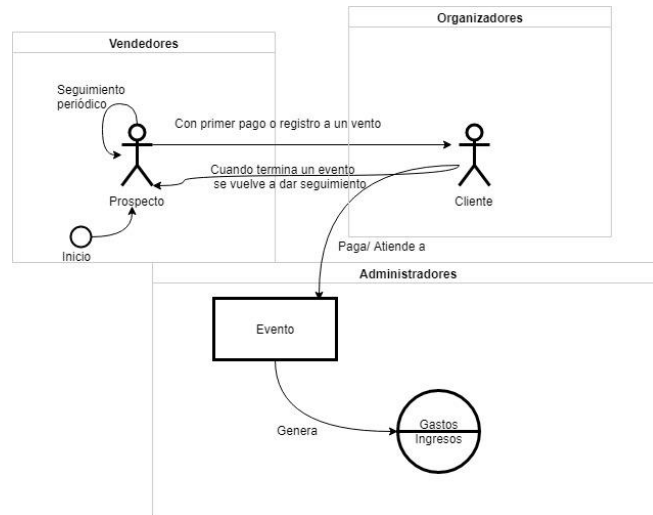
2. Descripción general

La problemática presentada por *CADHU* consiste principalmente en el uso de distintas herramientas. Las bases de datos utilizadas para poder realizar un proceso administrativo no están concentradas en un solo sistema, sino que los ejecutivos tienen que acudir a distintas aplicaciones web para poder acceder a ciertas bases de datos. Las áreas con deficiencia incluyen clientes, eventos y contabilidad.

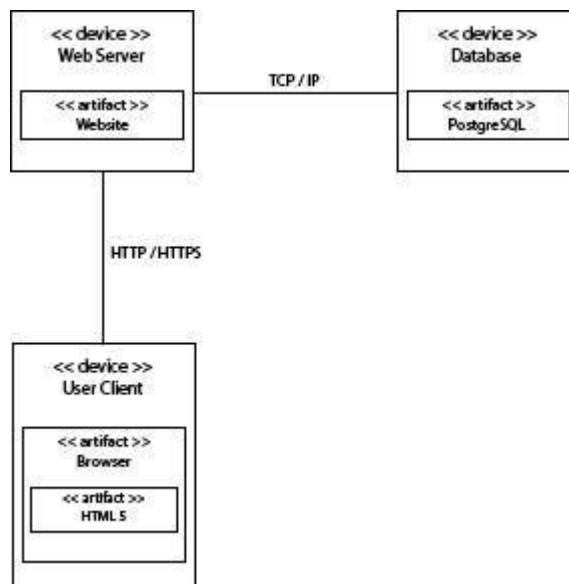
La área de mayor oportunidad del proyecto se encuentra en la fase de captación de prospectos como clientes donde la desorganización produce una caída de interés. La causa consiste en que al contactar a los prospectos, se hace varias veces por error y se crea un conflicto en el registro informal de la información al no tener una manera normalizada para almacenar dicha información. Por otro lado las bases de datos de los eventos están fragmentadas lo que causa retrabajo el momento de registrar sus clientes y tener seguimientos de asistencia y pagos. Como consecuencia, esto constituye a un problema de logística e ineficiencia. Debido a la falta de unicidad y normalización de los datos, la información no puede ser manipulada de manera que se puedan generar reportes para poder tomar decisiones empresariales y generar un valor agregado a la información. Finalmente, se atribuye a fallas dentro del sistema de contabilidad donde se pierde un control preciso de los ingresos y gastos que se generan.

3. Arquitectura General

3.1. Flujo Lógico

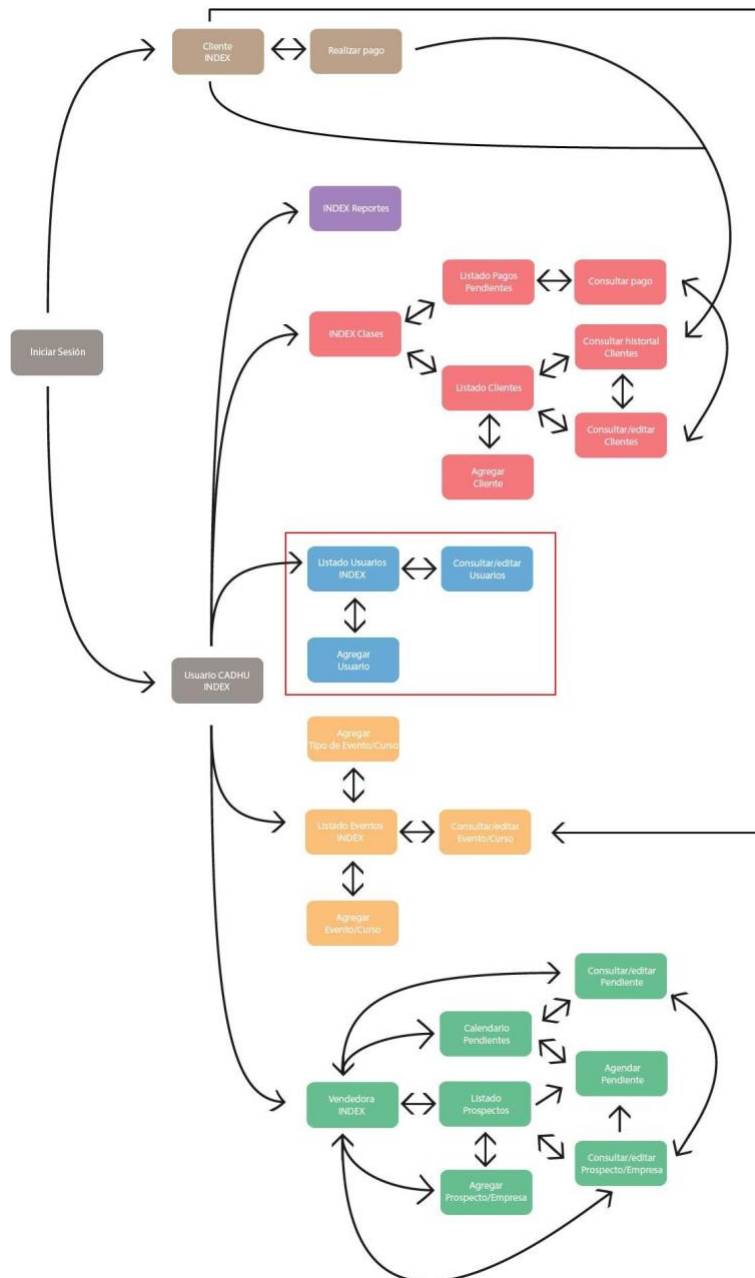


3.2. Diagrama despliegue



3.3. Diagrama de vistas y componentes

3.3.1. Diagrama de vistas



3.3.2. Guía de estilos ([link](#))

3.4. Diseño base de datos

3.4.1. MER([link](#))

4. Hardware/Software

Servidor: Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-119-generic x86_64)

5. Dependencias del sistema

[Describir todos los módulos y componentes externos al sistema con su versión]

6. Especificaciones técnicas

[Describir los servicios de hosting y dominio con el plan utilizado y las especificaciones de los mismos]

7. Pruebas de unidad e integración

Se utilizó la suite de testing de Django de Unit Testing. Cada prueba se diseñó a partir de los casos de aceptación definidos previamente. Para un mayor entendimiento de cómo construir y ejecutar pruebas en Django consultar el siguiente link:

<https://docs.djangoproject.com/en/2.0/topics/testing/overview/>

8. Configuración servidor

La configuración del servidor se hizo utilizando Nginx, Ubuntu 16.04 y PostgreSQL. La guía paso a paso para poder configurar el servidor se encuentra en el siguiente link:

<https://www.digitalocean.com/community/tutorials/how-to-set-up-django-with-postgres-nginx-and-gunicorn-on-ubuntu-16-04>

9. Actualización servidor

Después de iniciar sesión se insertan los siguientes comandos.

- sudo apt-get update
- sudo apt-get upgrade

10. Configuración dominio

En la siguiente liga se encuentra un tutorial paso a paso para poder configurar el dominio en el portal de Digital Ocean en caso de que el dominio llegara a cambiar:

<https://github.com/CaveLabs-1/Wiki/blob/master/Arquitectura/Manuales/Manual%20Arquitectura%20General.md#dominio>

11. Habilitar HTTPS: Certificado SSL

En la siguiente liga se encuentra un tutorial paso a paso para poder configurar el certificado de SSL y habilitar el HTTPS por medio de una sesión por SSH.

<https://github.com/CaveLabs-1/Wiki/blob/master/Arquitectura/Manuales/Manual%20Arquitectura%20General.md#seguridad>