



**UNIVERSIDAD SIMÓN BOLÍVAR
DECANATO DE ESTUDIOS PROFESIONALES
COORDINACIÓN DE INGENIERÍA DE LA COMPUTACIÓN**

**DESARROLLO DEL MÓDULO PRINCIPAL Y ESTADÍSTICAS DE LA
LIBRERÍA AUDITORÍAS TURPIAL**

Por:
Stefani Carolina Castellanos Torres

INFORME DE PASANTÍA

Presentado ante la Ilustre Universidad Simón Bolívar
como requisito parcial para optar al título de
Ingeniero de la Computación

Sartenejas, Enero 2018



**UNIVERSIDAD SIMÓN BOLÍVAR
DECANATO DE ESTUDIOS PROFESIONALES
COORDINACIÓN DE INGENIERÍA DE LA COMPUTACIÓN**

**DESARROLLO DEL MÓDULO PRINCIPAL Y ESTADÍSTICAS DE LA
LIBRERÍA AUDITORÍAS TURPIAL**

Por:
Stefani Carolina Castellanos Torres

Realizado con la asesoría de:
Tutor Académico: Angela Di Serio
Tutor Industrial: Ing. Pedro Romero

INFORME DE PASANTÍA
Presentado ante la Ilustre Universidad Simón Bolívar
como requisito parcial para optar al título de
Ingeniero de la Computación

Sartenejas, Enero 2018

RESUMEN

Este es el resumen

DEDICATORIA

RECONOCIMIENTOS Y AGRADECIMIENTOS

First of all, I would like to thank

ÍNDICE

RESUMEN	iii
ÍNDICE	vi
LISTA DE TABLAS	vii
LISTA DE FIGURAS	viii
LISTA DE SÍMBOLOS	x
LISTA DE ABREVIACIONES	xi
INTRODUCCIÓN	1
1 ENTORNO EMPRESARIAL	2
1.1 Descripción	2
1.2 Misión	2
1.3 Visión	2
1.4 Estructura	2
2 DEFINICIÓN DEL PROBLEMA	4
2.1 Antecedentes	4
2.2 Planteamiento del problema	4
2.3 Justificación	4
2.4 Objetivo general	4
2.5 Objetivos específicos	4
3 MARCO TEÓRICO	5
3.1 Auditoría	5
3.2 Acciones auditables	5
3.3 Microservicio	5
3.4 Integración Continua	5
3.5 Pruebas automatizadas	5
3.6 Patrón Modelo-Vista-Controlador	5
3.7 Patrón Modelo-Vista-Plantilla	5
3.8 Señales	5
3.9 Mixins	5

4	MARCO TECNOLÓGICO	6
4.1	Python	6
4.2	Django	6
4.3	HTML	6
4.4	Javascript	6
4.5	Pytest	6
4.6	Django-Graphos o Chart.js	7
4.7	PostgreSQL	7
4.8	MySQL	7
4.9	SQLite	7
4.10	JSON	7
4.11	Git	7
4.12	Jenkins	7
5	MARCO METODOLÓGICO	8
5.1	Descripción de la metodología TAUP	8
5.2	Fase de Concepción	8
5.2.1	Historias de Usuario (HU)	8
5.3	Fase de Construcción	9
5.4	Fase de Transición	9
6	DESARROLLO	10
6.1	Fase de investigación	10
6.2	Fase de concepción	10
6.3	Fase de construcción del núcleo	10
6.4	Fase de construcción del módulo de estadísticas	10
6.5	Fase de transición	10
	CONCLUSIONES	11
	Conclusiones	11
	Recomendaciones	11
	REFERENCIAS	12

LISTA DE TABLAS

LISTA DE FIGURAS

LISTA DE SÍMBOLOS

símbolos

LISTA DE ABREVIACIONES

HTTP	Hipertext Transfer Protocol. Protocolo de Transferencia de Hipertexto
------	---

INTRODUCCIÓN

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

CAPÍTULO 1

Entorno empresarial

1.1 Descripción

Turpial Development es una empresa mediana, con 4 años en el mercado que está enfocada en el desarrollo de sistemas y aplicaciones Web y Móviles. Fundada e integrada por jóvenes venezolanos, ofrece soluciones que cumplen con altos estándares de usabilidad, diseño y funcionalidad.

1.2 Misión

La empresa tiene como misión “prestar servicios y consultoría en diseño y desarrollo de soluciones web, a la medida del cliente, caracterizadas por una alta calidad, excelente soporte y experiencia de usuario”.

1.3 Visión

Su visión es “servir de plataforma para el desarrollo y éxito de nuevos emprendimientos en el área web”.

1.4 Estructura

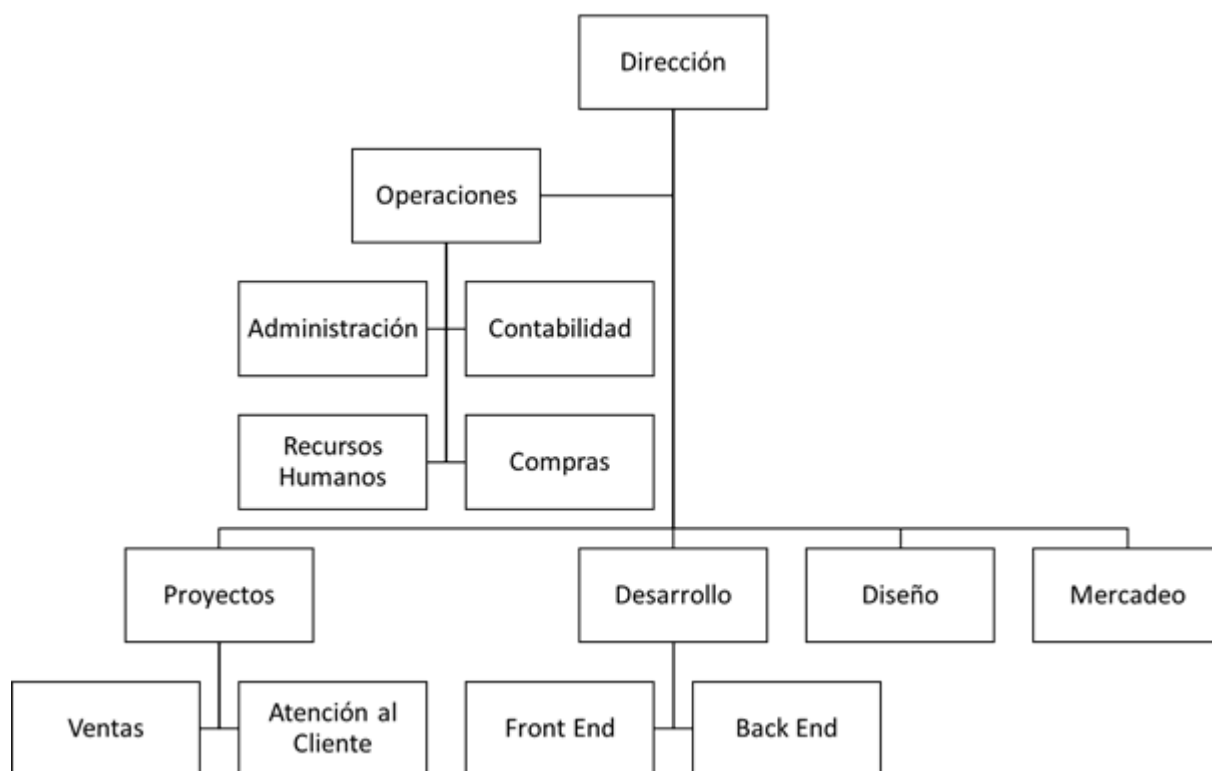
Dirección de Operaciones

Dirección de Proyectos

Dirección de Desarrollo

Dirección de Diseño

Dirección de Mercadeo



El desarrollo de la pasantía fue llevado a cabo en la Dirección de Desarrollo de la empresa. El pasante fue asignado al departamento de *Backend* bajo el cargo de Pasante, para cumplir tareas tales como: levantar requerimientos, diseñar y desarrollar las funcionalidades de la solución propuesta para el sistema. El Tutor Industrial ocupa el cargo de Desarrollador Estándar del área de Desarrollo y supervisó al pasante.

CAPÍTULO 2

Definición del problema

2.1 Antecedentes

2.2 Planteamiento del problema

2.3 Justificación

2.4 Objetivo general

Implementar, probar y presentar las funcionalidades de selección, gestión y listados de auditorías y todas las funcionalidades del módulo Estadísticas de la librería de Auditorías Turpial e implantar un sistema de integración continua con el repositorio.

2.5 Objetivos específicos

CAPÍTULO 3

Marco teórico

3.1 Auditoría

3.2 Acciones auditables

3.3 Microservicio

3.4 Integración Continua

3.5 Pruebas automatizadas

3.6 Patrón Modelo-Vista-Controlador

3.7 Patrón Modelo-Vista-Plantilla

3.8 Señales

3.9 Mixins

CAPÍTULO 4

Marco tecnológico

4.1 Python

Es un lenguaje de programación interpretado, poderoso y fácil de entender. Posee estructuras de datos de alto nivel y una aproximación sencilla a el paradigma de programación Orientado a Objetos. También es altamente extensible, cuenta con tipos de datos dinámicos y una sintaxis simple para facilitar el rápido desarrollo de aplicaciones. (Python Software Foundation, 2017)

4.2 Django

Es un *framework* de código abierto, escrito en Python y está basado en el patrón Modelo-Plantilla-Vista. Proporciona diversas funcionalidades reutilizables para desarrollar, rápidamente, aplicaciones Web escalables.

4.3 HTML

El lenguaje de marcado de hipertexto, *HyperText Markup Language* por sus siglas en inglés es un formato de datos simple usado, para crear documentos portables de una plataforma a otra y ha sido utilizada ampliamente en la World Wide Web desde 1990. (Berners, L., 1995)

4.4 Javascript

Es un lenguaje interpretado, multi-paradigma y dinámico que soporta estilos de programación funcional, orientado a objetos e imperativa, así como funciones de primera clase. Es comúnmente utilizado como el lenguaje de *script* para páginas Web. (Mozilla, 1995)

4.5 Pytest

Es un *framework* escrito en Python, que facilita la escritura de complejas pruebas de funcionalidad para aplicaciones y librerías y así, asegurar la calidad del software que será

entregado. (Pytest, 2017)

4.6 Django-Graphos o Chart.js

4.7 PostgreSQL

4.8 MySQL

4.9 SQLite

4.10 JSON

JavaScript Object Notation es un formato para intercambiar datos que es independiente del lenguaje, está basado en textos y es ligero. Define un pequeño conjunto de reglas para la representación de datos estructurados que sean portables. JSON puede representar cuatro tipos primitivos: cadenas de caracteres, números, booleanos y null; y dos tipos estructurados: objetos y arreglos. (RFC, 1)

4.11 Git

Es un sistema de control de versiones, gratis y de código abierto, diseñado para administrar cualquier tipo de proyecto con rapidez y eficiencia. Dispone facilidades para llevar el seguimiento de los cambios realizados, soportar para el desarrollo no-lineal, cambiar fácilmente de contexto y realizar experimentos sin afectar las versiones entregables del proyecto lo que mejora la colaboración entre equipos. (Git, 1)

4.12 Jenkins

CAPÍTULO 5

Marco metodológico

5.1 Descripción de la metodología TAUP

TAUP o Turpial Agile Unified Process es una metodología ágil creada por la empresa Turpial Development basada en los principios del "manifiesto ágil" que brinda a su equipo de trabajo una manera eficaz de llevar a cabo el desarrollo de *software*.

TAUP considera que la prioridad es la satisfacción del cliente mediante entregas continuas las cuales deben realizarse de manera frecuente, entre 2 semanas y 2 meses. Está conformada por tres fases de desarrollo: creación, construcción y transición.

5.2 Fase de Concepción

El objetivo de esta etapa es que el equipo de desarrollo profundice la comprensión de los requisitos del sistema y validación de la arquitectura propuesta.

5.2.1 Historias de Usuario (HU)

Es una técnica para expresar requerimientos que se caracteriza por ser sencilla de escribir, leer y evaluar. El usuario final o cliente le dará valor a esta HU ya que describe una funcionalidad que él necesita.

Las historias de usuarios se escriben en oraciones que poseen tres componentes indispensables: el rol del usuario que quiere realizar la acción, qué desea hacer y por qué. Ejemplo: *ejemplo*. Una buena HU está caracterizada por ser independiente, negociable, valiosa, estimable, pequeña y verificable (INVEST).

Adicionalmente, deben contar con criterios de aceptación bien definidos que brinden más detalle respecto a los aspectos de la Historia de Usuario que validan que se ha logrado completar.

5.3 Fase de Construcción

5.4 Fase de Transición

CAPÍTULO 6

Desarrollo

6.1 Fase de investigación

6.2 Fase de concepción

6.3 Fase de construcción del núcleo

6.4 Fase de construcción del módulo de estadísticas

6.5 Fase de transición

CONCLUSIONES

Conclusiones

Recomendaciones

REFERENCIAS

- Berners, L. (1995). Hypertext markup language - 2.0. <https://tools.ietf.org/html/rfc1866>. Consultado el 15/07/2017.
- Git (1). Pytest. <https://git-scm.com/>. Consultado el 17/07/2017.
- Mozilla (1995). Javascript. <https://developer.mozilla.org/es/docs/Web/JavaScript>. Consultado el 15/07/2017.
- Pytest (2017). Pytest. <https://docs.pytest.org/en/latest/>. Consultado el 17/07/2017.
- Python Software Foundation (2017). The python tutorial. <https://docs.python.org/3/tutorial/index.html>. Consultado el 15/07/2017.
- RFC (1). Pytest. <https://tools.ietf.org/html/rfc7159>. Consultado el 17/07/2017.