

Especificación de Requerimientos

Descripción del Diseño

Terra

Apellidos, Nombres	Correo electrónico	Rol
Alejandra Montealegre	mariamontealegreb@gmail.com	Back end
Miguel Fernandes	miguelangelantoniofernandez@gmail.com	Front end
Andres Camilo Rojas	camilin.andres90@gmail.com	Tester
Omar Castañeda	omarricardocc@gmail.com	Master scrum
Cristian Rodriguez		Bases de datos

Fecha de presentación: XX/XX/2016

Contenido

1	INTRODUCCIÓN.....	¡ERROR! MARCADOR NO DEFINIDO.
1.1	PROPÓSITO	3
1.2	ALCANCE O ÁMBITO DEL SISTEMA	3
1.3	DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS	3
1.3.4	<i>Referencias</i>	3
1.4	PERSPECTIVA GENERAL DEL DOCUMENTO.....	4
2	DESCRIPCIÓN GENERAL DE LA APLICACIÓN	4
2.1	PERSPECTIVA DE LA APLICACIÓN.....	4
2.2	FUNCIONES DE LA APLICACIÓN	4
2.3	CARACTERÍSTICAS DE LOS USUARIOS	4
2.4	RESTRICCIONES.....	4
2.5	SUPOSICIONES Y DEPENDENCIAS	4
2.6	REQUERIMIENTOS DIFERIDOS	5
3	REQUERIMIENTOS ESPECÍFICOS.....	5
3.1	REQUERIMIENTOS	5
3.1.1	<i>Product Backlog</i>	5
3.1.2	<i>Ciclo de Sprints del proyecto</i>	6
3.1.3	<i>Sprint Backlog</i>	6
3.1.4	<i>Historias de usuario (Tareas y Subtareas)</i>	7
3.1.5	<i>Mecánica de organización del grupo. (Reuniones, evidencias/artefactos)</i>	7
3.2	MODELO DE REQUERIMIENTOS	7
3.2.1	<i>Modelo de Casos de Uso</i>	7
4	DESCRIPCIÓN DEL DISEÑO	8
4.1	INTERFAZ GRÁFICA (MOCKUPS)	8
5	GESTIÓN DE LA CONFIGURACIÓN	8
6	PRUEBAS	¡ERROR! MARCADOR NO DEFINIDO.
6.1	DESCRIPCIÓN DE PRUEBAS UNITARIAS	8
6.2	DESCRIPCIÓN DE PRUEBAS DE ACEPTACIÓN	8
7	GLOSARIO	9
8	ANEXO(S)	9

1 INTRODUCCIÓN

El presente documento tiene como finalidad presentar la documentación, descripción, explicación y detalles sobre el uso configuración y código empleado para realizar una aplicación amigable para el usuario y operador en la interminable tarea de almacenar, vender prestar o remover de un inventario un libro

1.1 Propósito

El documento tiene como fin presentar al cliente un resumen ejecutivo de las principales características del programa del cual se va a hacerse acreedor, para llevar la tarea de inventarios y prestamos de una biblioteca de manera más amigable y fácil.

1.2 Alcance o Ámbito del Sistema

Terra es un programa que integra los diferentes beneficios a la hora de almacenar, vender, eliminar o agregar un insumo a su inventario, permitiendo a los tenderos definir precios, hacer cortes diarios y mensuales además de llevar un control de inventarios

1.3 Definiciones, Acrónimos y Abreviaturas

PHP: Es un lenguaje de programación enfocado en el desarrollo web, que permite a los usuarios cambiar modificar y agregar opciones al entorno de trabajo visual del programa.

CSS: Es un lenguaje de programación muy usado para establecer el diseño visual de los documentos web, e interfaces de usuario escritas en HTML o XHTML.

Mongo BD: Es una base de datos de documentos que ofrece una gran escalabilidad y flexibilidad, y un modelo de consultas e indexación avanzado.

Express JS: Proporciona una delgada capa de características de aplicación web básicas

React JS: Es una biblioteca Javascript de código abierto diseñada para crear interfaces de usuario con el objetivo de facilitar el desarrollo de aplicaciones en una sola página. Es mantenido por Facebook y la comunidad de software libre

Node JS: Es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor basado en el lenguaje de programación JavaScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos

1.3.1 Referencias

- 1) IEEE Std-830-1998.
- 2) ISO-IEC-IEEE-298148
- 3) IEEE Std-1016-2009
- 4) ISO/IEC/IEEE 29148:2011
- 5) OMG Unified Modeling Language
- 6) Schwinger, W.; Koch, N. "Modeling Web Applications", Chapter 3 en: Kappel, G.; Pröll,

- 7) Koch, N.; Knapp, A.; Zhang, G.; Baumeister, H. "UML-Based Web Engineering. An Approach Based on Standards", Chapter 7 en: Rossi, G.; Pastor
- 8) Python a Fondo Oscar Ramirez Jimenez.

1.4 Perspectiva General del Documento

El documento está dirigido para el que operador y el usuario tengan una idea general de cómo funciona Terra y que ventajas sobre otros programas de almacenamiento interactivo de insumos en cuanto a tiempo y requerimientos a partir de una interfaz minimalista.

2 DESCRIPCIÓN GENERAL DE LA APLICACIÓN

2.1 Perspectiva de la Aplicación

Terra es una aplicación que permite el almacenamiento por medio de bases de datos de inventarios ya sea de insumos como de clientes, además de estar enfocando en la venta de productos, lo que permitiera al usuario llevar de manera mas ágil las cuentas y de esta forma determinar sus veneficios diarios y mensuales.

2.2 Funciones de la Aplicación

Terra cuenta con las siguientes funciones:

- 1) Venta de productos y actualización de precios.
- 2) Manejo de inventarios.
- 3) Hojas de deudas de clientes.
- 4) Pedidos y domicilios.
- 5) Guanacias y ventas globales.

2.3 Características de los Usuarios

Todas aquellas personas que tengan un inventario, negocio o su principal nicho de mercado sea el comercio.

2.4 Restricciones

Las siguientes son las restricciones con las cuales Terra cuenta.

2.5 Suposiciones y Dependencias

Las siguientes son las suposiciones de Terra para asegurar su optimo funcionamiento.

2.6 Requerimientos Diferidos

Actualmente Terra tiene como propósito implementar en versiones futuras la conexión a diferentes bases de datos almacenadas en la nube, que no requieran un servidor físico ni de un tipado de base de datos específico, además se busca la interacción con diferente host que permitan la estimar cuales generan menor latencia y mayor rendimiento en la generación de resultados.

3 REQUERIMIENTOS ESPECÍFICOS

3.1 Requerimientos

Los requerimientos específicos para Terra actualmente son los siguientes:

- Interacción con las diferentes bases de datos usuarios, inventario, fechas, costos, clientes ect
- Guardar información acerca de la venta o un nuevo insumo en el inventario bajo solicitud de un usuario o administrados de la base datos
- Informar a los usuarios de manera clara y sencilla sobre las ganancias diarias o mensuales.

3.1.1 Product Backlog

- 1) Creación de Repositorio de GitHub
- 2) Enlace de repositorios
- 3) Creación de Trello
- 4) Documento de gestión de configuración
- 5) Historias de usuario en Trello
- 6) Acta de entrega y reunión
- 7) búsqueda y creación de plantilla
- 8) Montaje html y css de la página principal y secundarias
- 9) Montaje js de las paginas
- 10) Conexión backend
- 11) Entrega para test
- 12) Corrección errores informados por las pruebas
- 13) Configuración y alistamiento de ambiente (validando si se realiza en atlas
- 14) Creación de la base de datos
- 15) Creación de la colección en DB mongo
- 16) Insertar documento de prueba
- 17) Configuración del entorno
- 18) Conexión con el localhost
- 19) Conexión BD Mongo
- 20) Creación de los esquemas
- 21) Creación de los controladores
- 22) Creación de los Reuters
- 23) Conexión con el Front
- 24) Instalación de herramientas indicadas para pruebas

- 25) Generación de Script de Pruebas
- 26) Pruebas Unitarias Back
- 27) Pruebas Unitarias Front
- 28) Pruebas de Acceso
- 29) Pruebas de Integración Back-BD
- 30) Pruebas Funcionales API

3.1.2 Ciclo de Sprints del proyecto

A continuación, se relacionarán los sprints y sus fechas para evaluación.

- Sprint 1 23/10/2022
- Sprint 2 06/11/2022
- Sprint 3 20/11/2022
- Sprint 4 27/11/2022

3.1.3 Sprint Backlog

Sprint 1

- Creación de Repositorio de Github
- Enlace de repositorios
- Creación de Trello
- Documento de gestión de configuración (avance)
- Historias de usuario en Trello
- Historia de usuario
- Plantilla Front End

Sprint 2

- Interfaz Grafica
- Documento de gestión de configuración (avance)
- búsqueda y creación de plantilla
- Montaje html y css de la página principal y secundarias
- Montaje js de las paginas
- Conexión backend
- Configuración del entorno
- Conexión con el localhost
- Conexión BD Mongo
- Creación de los esquemas
- Entrega para test Historias de usuario en Trello (Actualizadas)
- Historia de usuario (Actualizadas)
- Acta de entrega y reunión (Actualizadas)

Sprint 3

- Corrección errores informados por las pruebas
- Configuración y alistamiento de ambiente (validando si se realiza en atlas)
- Creación de la base de datos
- Creación de la colección en DB mongo
- Insertar documento de prueba
- Configuración del entorno
- Conexión con el localhost

- Conexión BD Mongo
- Creación de los esquemas
- Creación de los controladores
- Creación de los Reuters
- Conexión con el Front
- Instalación de herramientas indicadas para pruebas
- Historias de usuario en Trello (Actualizadas)
- Historia de usuario (Actualizadas)
- Acta de entrega y reunión (Actualizadas)

Sprint 4

- Generación de Script de Pruebas
- Pruebas Unitarias Back
- Pruebas Unitarias Front
- Pruebas de Acceso
- Pruebas de Integración Back-BD
- Pruebas Funcionales API
- Historias de usuario en Trello (Actualizadas)
- Historia de usuario (Actualizadas)
- Acta de entrega y reunión (Actualizadas)

3.1.4 Historias de usuario (Tareas y Subtareas)

3.1.5 Mecánica de organización del grupo. (Reuniones, evidencias/artefactos)

Para la evaluación de Terra se llevarán a cabo reunión semanales y reuniones para acta de entrega de los diferentes product backlog y de esta forma llamar un alto grado de control en el desarrollo de la aplicación.

3.2 Modelo de Requerimientos

A continuación, se validarán los métodos de caso que serán usados en los diferentes métodos a evaluar o información de entrada.

3.2.1 Modelo de Casos de Uso

CU-01: "Nombre del Caso de Uso"

Iniciador	
Otros actores	
Precondiciones	
Flujo básico	
Actor	Sistema
1. ...	
	2. ...

	3. ...
4. ...	
Flujo alternativo 1	
Flujo alternativo 2	
...	
Flujo alternativo n	
Poscondiciones	

CU-02: "Nombre del Caso de Uso"

CU-nn: "Nombre del Caso de Uso"

4 DESCRIPCIÓN DEL DISEÑO

4.1 Interfaz gráfica (Mockups)

5 GESTIÓN DE LA CONFIGURACIÓN

6 PRUEBAS

6.1 Descripción de pruebas unitarias

6.2 Descripción de pruebas de aceptación

7 GLOSARIO

8 ANEXO(S)