

RESUMEN

El sistema “Qualita”, se realizó en la empresa “Digital Orbe”, tiene como objetivo realizar capturas de tipos de defectos en varias inspecciones de sus piezas y tener un control del tiempo empleado en revisarlas, ya que ellos trabajaban todo en Excel.

Se crearon los usuarios con 7 roles diferentes para la administración del sistema, fue desarrollado a través de la metodología SCRUM, utilizando las herramientas de Software MongoDB, Postman (Para las pruebas de Api Rest), NodeJs y React JS, con ayuda de estas herramientas se logró generar reportes específicos y generales solicitados por la empresa, así mismo la administración de empleados, visualización de piezas defectuosas o con control de calidad.

ABSTRACT

The "Qualita" system was carried out in the company "Digital Orbe", its objective is to capture the types of defects in various inspections of its parts and to have control of the time spent in reviewing them, since they usually worked everything in Excel.

Users with 7 different roles for system administration were created, it was developed through the SCRUM methodology, using the MongoDB Software tools, Postman (for Api Rest tests), NodeJs and React JS, with the help of these tools. It was possible to generate specific and general reports requested by the company, as well as the administration of employees, visualization of defective parts or with quality control.

Revisado por Sibma
Ma. Trinidad Gonzaga
Hernández
21/Abril/2022

ÍNDICE GENERAL

Contenido

CAPÍTULO I	2
DATOS GENERALES DE LA ORGANIZACIÓN	2
1. DATOS GENERALES DE LA ORGANIZACIÓN.....	3
1.1. <i>DATOS GENERALES.....</i>	3
1.1.1 Nombre de la Organización.....	3
1.1.2 Razón social.....	3
1.1.3 Logotipo.....	3
1.1.4 Giro.....	3
1.1.5 Dirección.....	3
1.1.6 Teléfonos.....	3
1.1.7 Página Web	3
1.1.8 Croquis de Localización.....	4
1.2. <i>ANTECEDENTES DE LA ORGANIZACIÓN.....</i>	4
1.3. <i>MISIÓN, VISIÓN Y OBJETIVOS DE LA ORGANIZACIÓN</i>	5
1.4. <i>ORGANIGRAMA.....</i>	6
1.5. <i>DESCRIPCIÓN DEL DEPARTAMENTO DE ESTADÍA.....</i>	6
CAPÍTULO II	7
METODOLOGÍA BÁSICA	7
2 ANTECEDENTES.....	8
2.1 <i>PLANTEAMIENTO DEL PROBLEMA.....</i>	8
2.2 <i>OBJETIVOS GENERAL Y ESPECÍFICOS</i>	8
2.2.1 <i>OBJETIVO GENERAL</i>	8
2.2.2 <i>OBJETIVOS ESPECÍFICOS.....</i>	8
2.3 <i>JUSTIFICACIÓN</i>	9
2.4 <i>ALCANCES Y DELIMITACIONES</i>	9
2.5 <i>CRONOGRAMA DE ACTIVIDADES (actividades de todo el proyecto)</i>	10
CAPÍTULO III	11
MARCO TEÓRICO	11
3 MARCO TEÓRICO Y CONCEPTUAL	12
3.1 <i>CONCEPTOS BÁSICOS</i>	12

3.2	<i>METODOLOGÍA UTILIZADA</i>	14
3.3	<i>TÉCNICAS Y HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO DEL PROYECTO</i>	19
CAPÍTULO IV.....		22
DESARROLLO		22
4 INTRODUCCIÓN		23
4.1 ACTIVIDADES REALIZADAS DURANTE EL PERÍODO DE ESTADÍA.....		23
4.2 DESARROLLO DEL PROYECTO.....		23
<i>Entregables intermedios con características detalladas y tiempos de entrega.....</i>		25
<i>Pruebas.....</i>		46
• <i>Test de usabilidad</i>		46
• <i>Plan de pruebas.....</i>		47
• <i>Casos de uso de pruebas.....</i>		51
• <i>Ejecución de pruebas</i>		53
<i>Implementación</i>		54
• <i>Plan de implementación</i>		54
<i>Manual de usuario</i>		54
<i>Manual de Usuario</i>		54
CAPÍTULO V.....		55
RESULTADOS.....		55
5 RESULTADOS OBTENIDOS.....		56
Referencias Hemerográficas.....		107
Referencias Digitales		108
Referencias Bibliográficas		110

ÍNDICE DE FIGURAS Y/O TABLAS

Ilustración 1 – Logotipo	3
Ilustración 2 - Croquis de Localización	4
Ilustración 3 - Organigrama	6
Ilustración 4 - Cronograma de Actividades	10
Ilustración 5 - Software.....	12
Ilustración 6 - Sistema Web.....	12
Ilustración 7 - Hardware	13
Ilustración 8 - Front-End	13
Ilustración 9 - Back-End	14
Ilustración 10 - Diagrama SCRUM.....	18
Ilustración 11 - Visual Studio Code.....	20
Ilustración 12 - Postman.....	21
Ilustración 13 - MongoDB	21
Ilustración 14 – EDT	24
Ilustración 15 - Diagrama de Roles.....	26
Ilustración 16 - Caso de uso	30
Ilustración 17 - Diagrama de Secuencia	31
Ilustración 18 - Diagrama de Colaboración.....	32
Ilustración 19 - Diagrama de Estado.....	33
Ilustración 20 - Diagrama de Actividades.....	34
Ilustración 21 - Diagrama de Clases.....	35
Ilustración 22 - Diagrama de Componentes.....	36
Ilustración 23 - Diagrama de Despliegue	37
Ilustración 24 - Diagrama Relacional	38
Ilustración 25 - Qualita Login Antes	57
Ilustración 26 - Qualita Login Después	57
Ilustración 27 - Qualita Antes - Mi cuenta	58
Ilustración 28 - Qualita Despues - Mi cuenta	58
Ilustración 29 - Qualita Antes - Employees	59
Ilustración 30 - Qualita Después - Employees	59
Ilustración 31 - Qualita Antes - Package Types	60
Ilustración 32 - Qualita Después - Package Types	60
Ilustración 33 - Qualita Antes - Destinations	61

Ilustración 34 - Qualita Después - Destinations	61
Ilustración 35 - Qualita Antes - Machines	62
Ilustración 36 - Qualita Después - Machines	62
Ilustración 37 - Qualita Antes - Inspection Activities	63
Ilustración 38 - Qualita Después - Inspection Activities.....	63
Ilustración 39 - Qualita Antes - Locations	64
Ilustración 40 - Qualita Después - Locations	64
Ilustración 41 - Qualita Antes - Departaments	65
Ilustración 42 - Qualita Después - Departaments	65
Ilustración 43 - Qualita Antes - Defect Types.....	66
Ilustración 44 - Qualita Después - Defect Types.....	66
Ilustración 45 - Qualita Antes - Inspections.....	67
Ilustración 46 - Qualita Después - Inspections.....	67
Ilustración 47 - Qualita Antes - Inventory (Pallet).....	68
Ilustración 48 - Qualita Después - Inventory (Pallet).....	68
Ilustración 49 - Qualita Antes - Part Numbers.....	69
Ilustración 50 - Qualita Después - Part Numbers.....	69
Ilustración 51 - Qualita Antes - Dashboard	70
Ilustración 52 - Qualita Después - Dashboard	70
Ilustración 53 - Qualita Antes - Crear Employees	71
Ilustración 54 - Qualita Después - Crear Employees	71
Ilustración 55 - Qualita Antes - Editar Employees.....	72
Ilustración 56 - Qualita Después - Editar Employees.....	72
Ilustración 57 - Qualita Antes - Crear Package Types	73
Ilustración 58 - Qualita Después - Crear Package Types	73
Ilustración 59 - Qualita Antes - Editar Package Types	74
Ilustración 60 - Qualita Después - Editar Package Types	74
Ilustración 61 - Qualita Antes - Crear Destinations	75
Ilustración 62 - Qualita Después - Crear Destinations	75
Ilustración 63 - Qualita Antes - Editar Destinations	76
Ilustración 64 - Qualita Después - Editar Destinations	76
Ilustración 65 - Qualita Antes - Crear Machines.....	77
Ilustración 66 - Qualita Después - Crear Machines.....	77
Ilustración 67 - Qualita Antes - Editar Machines	78

Ilustración 102 - Qualita Después - sprint 7	95
Ilustración 103 - Qualita Después - sprint 8	96
Ilustración 104 - Qualita Después - sprint 9	96
Ilustración 105 - Qualita Después - sprint 10	97
Ilustración 106 - Qualita nuevo - borrados logicos	98
Ilustración 107 - Qualita nuevo - snackbars	98
Ilustración 108 - Qualita nuevo - Activities	99
Ilustración 109 - Qualita nuevo - Activities Reports.....	100
Ilustración 110 - Qualita nuevo - Activities Reports sprint	100
Ilustración 111 - Qualita nuevo - Activities Reports sprint 2	101
Ilustración 112 - Qualita nuevo - Roles y Permisos	101
Ilustración 113 - Qualita nuevo - Roles y Permisos Crear.....	102
Ilustración 114 - Qualita nuevo - Roles y Permisos Editar	103
Ilustración 115 - Qualita nuevo - Reporte Excel.....	104
 Tabla 1 - Sofware y Hardware	25
Tabla 2 - Requerimientos Humanos	25
Tabla 3 - Roles de Desarrolladores	29
Tabla 4 - Tabla de Inspection	39
Tabla 5 - Tabla de Pallets.....	40
Tabla 6 - Tabla de Part Numbers.....	40
Tabla 7 - Tabla de Activity Reports.....	41
Tabla 8 de Defects Type	42
Tabla 9 de Employees.....	42
Tabla 10 de Destinations.....	43
Tabla 11 de Machines	43
Tabla 12 de Activities	44
Tabla 13 de Activities Inspection	44
Tabla 14 de Package Types	44
Tabla 15 de Locations	45
Tabla 16 de Departaments	45
Tabla 17 de Roles	46
Tabla 18 de Permissions	46
Tabla 19 Tabla de Usabilidad	47

INTRODUCCIÓN

El objetivo principal de realizar el sistema Qualita, es con el fin de proporcionar una mayor organización de los tiempos de revisión e inspección de los defectos que resultan de la fabricación de sus “part numbers”, se pueden tardar, segundos, minutos y horas, el objetivo es mostrar los defectos que tienen los “part numbers” es el detectar el defecto más común y calcular tiempos de inspección.

Este documento describe el proyecto en cinco capítulos importantes, los cuales son:

Capítulo I: Datos generales de la organización en el cual se mostrará los datos de la organización donde se realizó el proyecto.

Capítulo II: Metodología Básica se describe la problemática, los objetivos generales y específicos, justificación, los alcances que se lograran y las delimitaciones que se presentan. Además, se incluye el plan de trabajo para el desarrollo del proyecto.

En el capítulo III: Marco Teórico se exponen los conceptos básicos que son esenciales para el proyecto, así como la metodología, las técnicas y herramientas utilizadas que fueron necesarias para el desarrollo.

El capítulo IV: Presenta el desarrollo del proyecto, el cual se incluyen los diagramas UML y los requerimientos del sistema contemplando el diseño de la base de datos y el plan de trabajo.

El capítulo V: Se muestran los resultados obtenidos del proyecto.

CAPÍTULO I

DATOS GENERALES DE LA ORGANIZACIÓN

CAPÍTULO II

METODOLOGÍA BÁSICA

2 ANTECEDENTES

2.1 PLANTEAMIENTO DEL PROBLEMA

La empresa Qualita tiene la necesidad de obtener un mayor control en la revisión e inspección de los posibles defectos que tienen las piezas “part numbers” y mejorar el control de calidad, para que le permita visualizar las piezas defectuosas y cuantas cumplen con los estándares de calidad, monitoreando los tiempos que tardan en revisar los “part numbers”, así mismo solicitan que se generen roles diferentes de los empleados en el sistema.

2.2 OBJETIVOS GENERAL Y ESPECÍFICOS

2.2.1 OBJETIVO GENERAL

Realizar sistema para registrar los diferentes tipos de defectos en los “part numbers”, mediante las herramientas Nuxt JS como Front-End, NodeJS como Back-End y base de datos MongoDB localizado en un servidor Mongo Atlas, con el fin de calcular los tiempos de realización e inspección, para mejorar el control de calidad de las piezas fabricadas.

2.2.2 OBJETIVOS ESPECÍFICOS

- Realizar CRUDs y/o Catálogos para la administración , se realizará con lenguaje de programación JavaScript con el IDE Visual Studio Code, con el fin de optimizar tiempos con las extensiones del IDE.
- Replicar los “inspections” en “Activities Reports”, utilizando el modelo y controlador, para ahorrar tiempo de programación y solo calcular tiempos sin necesidad de capturar defectos.
- Reprogramar las “inspections” para poder registrar 2 “inspections” conocidos como “sprints” en una sola “inspection”.
- Realizar reportes en Excel, utilizando librerías externas adaptadas a Javascript en formato JSON, para que puedan generar sus reportes por fechas.

2.3 JUSTIFICACIÓN

Qualita con el sistema, la empresa podrá gestionar de manera adecuada las inspecciones e incluso tener un mayor control de calidad, esto beneficiará ya que podrá gestionar el tiempo de sus inspectores de sus horas de trabajo, podrán generar reportes en Excel de los días que ellos lo soliciten, al principio solo serán algunos reportes, así mismo podrá gestionar de manera adecuada las inspecciones y a sus inspectores, por otro lado, el control de calidad mejorará, ya que podrán obtener información de sus piezas con defectos y que piezas cumplen con los estándares de calidad.

2.4 ALCANCES Y DELIMITACIONES

Alcances:

- Tener un mayor control con roles y permisos.
- Los reportes serán fáciles de obtener.
- El administrador podrá modificar los usuarios y roles.
- Los reportes del sistema son en tiempo real
- Se puede filtrar por fechas
- Automáticamente se filtra por el mes actual, pero se puede modificar.
- El diseño es amigable para los usuarios

Delimitaciones:

- Si no hay conexión a internet, no se podrá trabajar, ya que la base de datos está en la nube.
- No se utilizará responsive ya que se obtendrá una aplicación en Android y/o iOS
- Los respaldos de Base de Datos no son automáticos.

2.5 CRONOGRAMA DE ACTIVIDADES (actividades de todo el proyecto)

Ilustración 4 - Cronograma de Actividades

CAPÍTULO III

MARCO TEÓRICO

3 MARCO TEÓRICO Y CONCEPTUAL

3.1 CONCEPTOS BÁSICOS

Software: Es el equipamiento o soporte lógicos de una computadora digital, el cual conjunta instrucciones y datos regulados para ser leídas e interpretadas por una computadora. Estas instrucciones y datos fueron concebidos para el procesamiento electrónico de datos. (Computer, 2002).



Ilustración 5 - Software

(Fuente: Computer, 2002)

Sistemas web: También conocido como “aplicaciones Web” son aquellos que están creados e instalados no sobre una plataforma o sistemas operativos (Windows, Linux). Sino que se aloja en un servidor en Internet o sobre una intranet (red local). Su aspecto es muy similar a páginas Web que vemos normalmente, pero en realidad los ‘sistemas Web’ tienen funcionalidades muy potentes que brindan respuestas a casos particulares. (sistemas web, 2013)



Ilustración 6 - Sistema Web

(Fuente: milanuncions, 2020)

Hardware: El hardware es la parte que puedes ver del computador, es decir todos los componentes de su estructura física algunos ejemplos son: la pantalla, el teclado, la torre y el ratón hacen parte del hardware de tu equipo. (devcode, 2018)



Ilustración 7 - Hardware

(Fuente: dispositivospalopoliperdiguero, 2008)

Front-End: Se enfoca en el usuario, en todo con lo que podemos interactuar y lo que vemos mientras navegamos, nuestra web busca causar una buena impresión y agradar al usuario, para lo cual utiliza HTML, CSS y JAVASCRIPT. Buena experiencia de usuario, inmersión y usabilidad, son algunos de los objetivos que busca. (devcode, 2018)



Ilustración 8 - Front-End

(Fuente: devcode, 2018)

Backend: Enfocado en hacer que todo lo que está detrás de un sitio web funcione correctamente. Toma los datos, los procesa y los envía al usuario, además de encargarse de las consultas o peticiones a la Base de Datos, la

conexión con el servidor, entre otras tareas. Cuenta con una serie de lenguajes y herramientas que le ayudan a cumplir con su trabajo como PHP, Ruby, Python, JavaScript, SQL, MongoDB, MySQL, etc., estos son usados para crear sitios dinámicos. (devcode, 2018)



Ilustración 9 - Back-End

(Fuente: devcode, 2018)

3.2 METODOLOGÍA UTILIZADA

Es de suma importancia conocer el modo como se interrelacionan metodologías con estándares y herramientas siguiendo un único propósito, el cual consiste en la elaboración de aplicaciones de manera eficientes, ordenadas y con el menor número de defectos. (Procesos de Software, 2017)

Por ello se utilizará la Metodología SCRUM la cual consta de 5 fases:

1. Reunión de planificación de Sprint

El trabajo para realizar en el Sprint se prevé en la **Reunión de Planificación del Sprint**. Este plan se crea con la colaboración de todo el Equipo Scrum.

La reunión de planificación de un Sprint es un evento de tiempo variable. Para un Sprint de un mes tiene ocho horas de duración. Para Sprints más cortos, el evento es proporcionalmente más corto. Por ejemplo, para un Sprint de dos semanas, las reuniones de planificación de Sprint son de cuatro horas de duración.

En esta reunión se define la funcionalidad en el incremento planeado y cómo el Equipo de Desarrollo creará este incremento y la salida de este trabajo es definir el Objetivo del Sprint.

La reunión de planificación de Sprint tradicionalmente consta de dos partes, cada una de la mitad de tiempo de duración de la Reunión de Planificación respondiendo a las siguientes dos preguntas:

- ¿Qué va a ser entregado en el incremento resultante del próximo Sprint?
- ¿Cómo se va a realizar el trabajo seleccionado?

A destacar que el objetivo del Sprint puede ser un hito en el objetivo más amplio de la hoja de ruta (roadmap) del producto.

2. El Scrum Diario

Es un evento de 15 minutos, cuyo objetivo es que el equipo de desarrollo sincronice actividades, y cree un plan para las próximas 24 horas. Esto se realiza mediante la inspección del trabajo desde el último Scrum Diario, y la previsión del trabajo que se puede hacer antes del próximo. El Scrum Diario se lleva a cabo en la misma hora y lugar cada día para reducir la complejidad.

El equipo de desarrollo utiliza el Scrum Diario para evaluar el progreso hacia la meta del Sprint y evaluar la tendencia del progreso en finalizar el trabajo en el Sprint Backlog. Cada día, el equipo de desarrollo debe ser capaz de explicar al dueño del producto y al Scrum Máster como van a trabajar juntos como un equipo auto-organizado para lograr el objetivo y crear el incremento previsto en el resto del Sprint.

Un beneficio adicional es que los Scrums diarios mejoran las comunicaciones, eliminan otras reuniones, identifican y eliminan obstáculos para el desarrollo, destacan y promueven la rápida toma de decisiones, y mejoran el nivel de

conocimiento del proyecto del equipo de desarrollo. Esta es una reunión clave de inspección y adaptación.

3. Trabajo de desarrollo durante el Sprint

Cuando el sprint está en curso, debemos asegurar que:

- No se realizan cambios que afectan al objetivo del Sprint.
- No disminuyen los objetivos de calidad.
- El Alcance podrá aclararse y renegociarse entre el propietario del producto y el Equipo de Desarrollo a medida que se va aprendiendo.

Cuando un Sprint es demasiado largo, la definición de lo que se está construyendo puede cambiar, puede aumentar la complejidad y puede aumentar el riesgo. Los Sprints permiten previsibilidad al garantizar la inspección y la adaptación de los avances hacia una meta de por lo menos cada mes de calendario.

4. Revisión del Sprint

Se lleva a cabo al final del Sprint, para inspeccionar el incremento y adaptar, si es necesario, el Product Backlog. El Equipo Scrum y las partes interesadas colaboran durante la revisión de lo que se hizo en el Sprint. Basado en ese y cualquier cambio en el Product Backlog durante el Sprint, los asistentes trabajan en las próximas cosas que se podrían hacer. Esta es una reunión informal, y la presentación del incremento está destinada a obtener retroalimentación y fomentar la colaboración.

La revisión de Sprint incluye los siguientes elementos:

- Los asistentes son el Equipo Scrum y los interesados clave invitados por el Dueño de Producto.
- El propietario del producto identifica lo que se ha "hecho" y lo que no se ha "hecho";
- El equipo de desarrollo discute lo que anduvo bien durante el Sprint, qué problemas hubo y cómo se resolvieron.
- El equipo de desarrollo demuestra el trabajo que se ha "hecho" y responde preguntas sobre el Incremento.
- El propietario del producto analiza el estado actual del Product Backlog, y estima fechas de finalización basado en el progreso hasta la fecha.
- Todo el grupo colabora en qué hacer a continuación, de modo que la revisión del Sprint ofrece valiosos aportes a las subsiguientes reuniones de planificación de Sprint.
- Se hace una revisión de cómo el mercado o el uso potencial del producto podría haber cambiado lo que es de más valor para hacer a continuación.
- Se hace una revisión de la línea de tiempo, presupuesto, capacidades potenciales y mercado para la próxima entrega prevista del producto.

El resultado de la revisión del Sprint es un Product Backlog revisado que define los ítems del Product Backlog de mayor valor o probables para el siguiente Sprint. El Product Backlog también se puede ajustar en general para satisfacer las nuevas oportunidades.

5. Retrospectiva del Sprint

Es una oportunidad para el Equipo Scrum de inspeccionarse a sí mismo y crear un plan de mejoras para ejecutar durante el siguiente sprint. El propósito de la retrospectiva de Sprint es:

- Revisar cómo fue el último Sprint en lo que respecta a las personas, relaciones, procesos y herramientas.
- Identificar y ordenar los temas principales que salieron bien y las potenciales mejoras.
- Crear un plan para la implementación de mejoras con respecto a cómo el Equipo Scrum hace su trabajo.

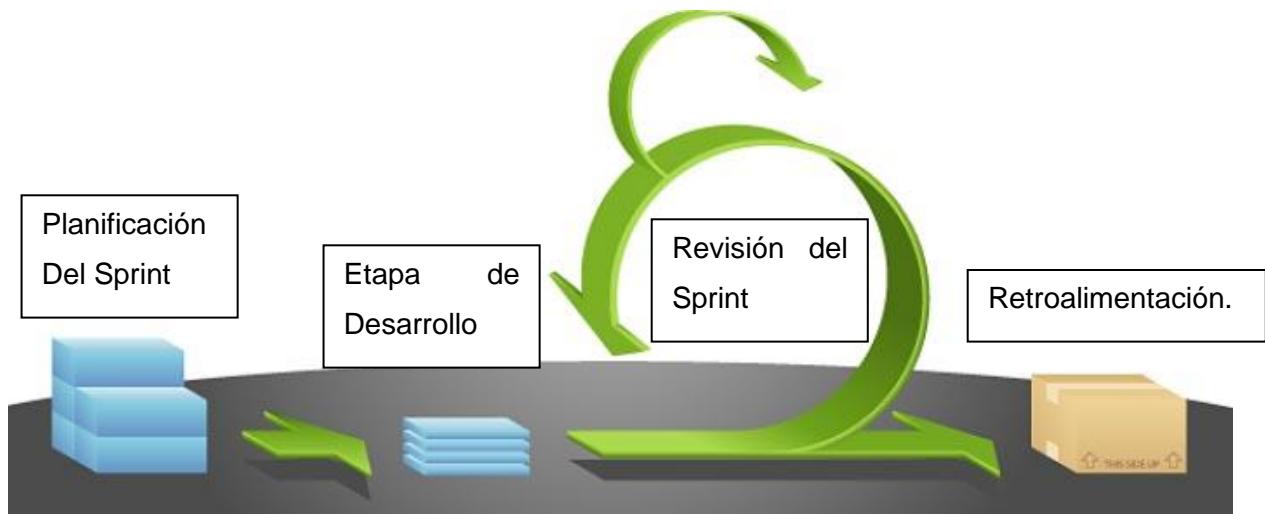


Ilustración 10 - Diagrama SCRUM

(Fuente: EALDE, 2016)

3.3 TÉCNICAS Y HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO DEL PROYECTO

En el análisis del sistema se recurre a las siguientes técnicas que se utilizaron para el desarrollo del proyecto:

- Reuniones con el encargado del área de sistemas para proponer mejoras al sistema, o ver modificaciones.
- Pruebas reales del sistema (Funcionamiento y efectividad al momento de realizar un pedido).

Las herramientas utilizadas son:

Hardware

- Equipo de cómputo
- Procesador AMD Ryzen 7 3700U with Radeon Vega Mobile Gfx 2.30 GHz
- Memoria RAM 12GB
- Disco Duro 1 TB, Disco de Estado Solido 256 GB

Software

Visual Studio Code: Es un editor de código multiplataforma concebida para programar sin distracciones. Algunas características que ofrece son:

- Interfaz de color oscuro y la riqueza de coloreado de la sintaxis (Se puede personalizar, en mi caso tengo un tema de Atom).
- Permite tener varios documentos abiertos mediante pestañas, e incluso emplear varios paneles para aquellos que utilicen más de un monitor.
- Dispone de modo de pantalla completa, para aprovechar al máximo el espacio visual disponible de la pantalla.
- El sistema de resaltado de sintaxis de Sublime Text soporta un gran número de lenguajes (C, C++, C#, CSS, D, Erlang, HTML, Groovy, Haskell, HTML, Java,

JavaScript, LaTeX, Lisp, Lua, Markdown, Matlab, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile, XML y Vue).

- El programa dispone de autoguardado, muchas opciones de personalización, cuenta con un buen número de herramientas para la edición del código y automatización de tareas.

```

    <div className="row">
      {(rows.find((row) => row._id === value).endTime && (
        (isPermitted(user.role, "inspection.start_inspection") ||
        isPermitted(
          user.role,
          "inspection.continue_inspection"
        )) || (
          <div className="col">
            <StartButton
              inspection={rows.find((row) => row._id === value)}
              iniciarPermiso={isPermitted(
                user.role,
                "inspection.start_inspection"
              )}
              continuarPermiso={isPermitted(
                user.role,
                "inspection.continue_inspection"
              )}
            />
          </div>
        )} || (
          rows.find((row) => row._id === value).startTime && (
            (isPermitted(
              user.role,
              "inspection.finish_inspection"
            ) && (
              <div className="col">
                <FinishButton
                  inspection={rows.find((row) => row._id === value)}
                  refreshData={refreshData}
                  user={user}
                />
              </div>
            )) || (
              rows.find((row) => row._id === value).endTime && (
                (isPermitted(user.role, "inspection.update") && (
                  <div className="d-flex flex-row">
                    <div>
                      <Edit
                        refreshData={refreshData}
                      />
                    </div>
                  </div>
                ))
              )
            )
          )
        )
      )
    )
  
```

Ilustración 11 - Visual Studio Code

(Fuente: Elaboración Propa)

Postman: Es una extensión gratuita para el navegador Google Chrome que permite probar servicios web fácilmente, basta con indicar la URL, el método HTTP (POST, GET, etc.) y los parámetros de la petición.

Conforme usamos Postman, automáticamente se van guardando las peticiones más recientes en el historial. También permite definir colecciones en donde podemos guardar una serie de métodos para reutilizarlos más fácilmente y poder compartirlos con otras personas en formato JSON. Además, podemos crear ambientes y definir variables específicas para esos ambientes. (Postman, 2018)

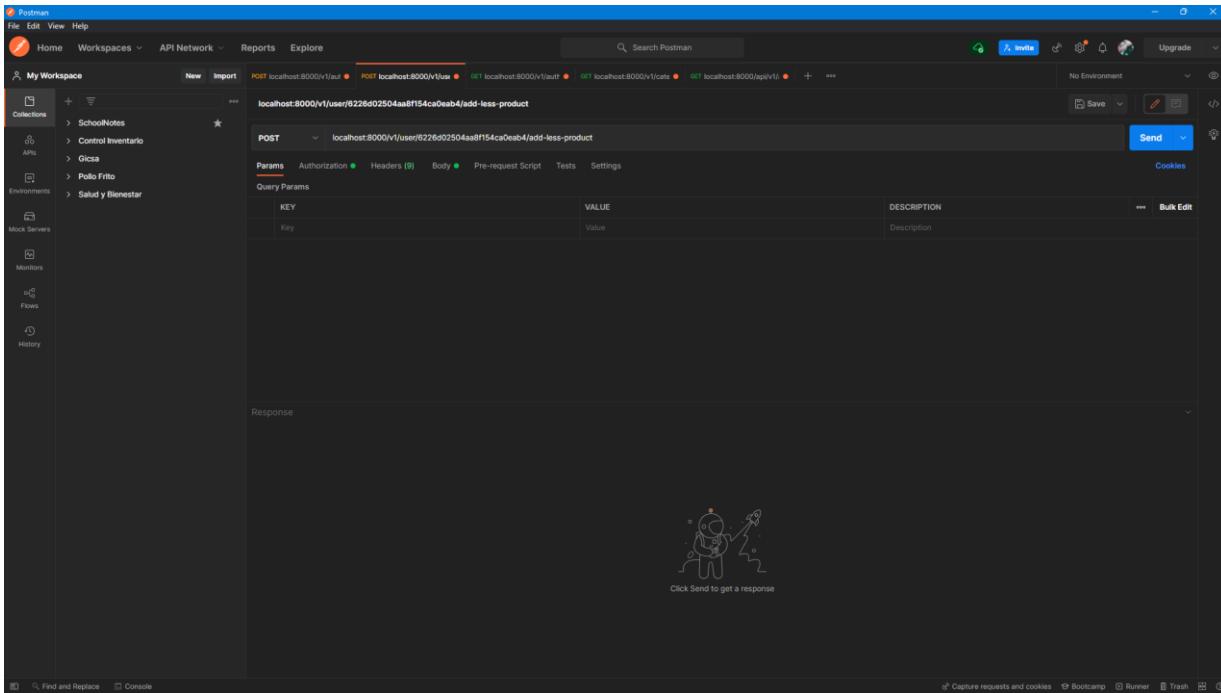


Ilustración 12 - Postman

(Fuente: Elaboración Propia)

MongoDB: Es un sistema de base de datos NoSQL, orientado a documentos y de código abierto. ... El código fuente está disponible para los sistemas operativos Windows, GNU/Linux, OS X y Solaris.

MongoDB (del inglés humongous, "enorme") es un sistema de base de datos NoSQL orientado a documentos de código abierto y escrito en C++, que en lugar de guardar los datos en tablas lo hace en estructuras de datos BSON (similar a JSON) con un esquema dinámico. Al ser un proyecto de código abierto, sus binarios están disponibles para los sistemas operativos Windows, GNU/Linux, OS X y Solaris y es usado en múltiples proyectos o implementaciones en empresas como MTV Network, Craigslist, BCI o Foursquare.



Ilustración 13 - MongoDB

(Fuente: thinkbif, 2014)

CAPÍTULO IV

DESARROLLO

4 INTRODUCCIÓN

4.1 ACTIVIDADES REALIZADAS DURANTE EL PERÍODO DE ESTADÍA

Durante todo el periodo de estadía, se efectuaron diferentes actividades para la elaboración del proyecto:

Primero se hizo análisis de la problemática y las necesidades que se tiene que resolver. Diariamente hablamos sobre el proyecto que me toco, lo nuevo que se puede integrar y las modificaciones que pide el jefe, ya que cada semana hay reuniones con él para ver puntos fuertes y débiles del sitio, que es lo que le hace falta o que se debería modificar.

Una vez que se obtuvo toda la información necesaria, se eligió la metodología SCRUM la cual se adaptó durante el desarrollo del proyecto.

Posteriormente se hizo un análisis comparando diferentes Frameworks de desarrollo tanto Back-End y tanto como Front-End, para elegir cuál de ellos se adapta a las necesidades del desarrollo del proyecto.

Se desarrollaron y analizaron los diagramas UML, esto con el fin de darnos una visión más general de como seria el funcionamiento del sistema.

Finalmente se realizó el desarrollo de todo el sistema de administración, tomando en cuenta correcciones, retroalimentaciones e implementaciones de nuevos módulos de programación propuestos por los Ejecutivos de dicha empresa.

También en otro proyecto se ayudó a evaluar mejor el sitio web de excite.com.mx, hacer que mejore el nivel de estadísticas en SEO.

4.2 DESARROLLO DEL PROYECTO

El proyecto debe contener por lo menos:

1. PLAN DE DESARROLLO EDT(estructura de desglose de trabajo)

El plan de desarrollo se realizó bajo la metodología SCRUM en cinco fases que se describen a continuación.



Ilustración 14 – EDT

(Fuente: Elaboración Propia)

Entregables intermedios con características detalladas y tiempos de entrega

2. ANÁLISIS DE REQUERIMIENTOS.

A continuación, se muestran los requerimientos que fueron utilizados para el desarrollo del sistema:

Software	Hardware
Windows 10.	Equipo de Cómputo.
Visual Studio Code.	
Google Chrome.	
Node JS.	
MongoDB.	
GanntProject.	
Windows Terminal	

Tabla 1 - Sofware y Hardware

(Fuente: Elaboración Propia)

En cuanto a los requerimientos para el usuario son los siguientes:

Requerimientos Humanos (Para el cliente).
Hosting.
Internet o Datos.
Área de Trabajo.
Luz.
Computadora o Smartphone.

Tabla 2 - Requerimientos Humanos

(Fuente: Elaboración Propia)

En el siguiente diagrama se muestra el equipo propuesto por el líder del proyecto para el desarrollo del sistema.

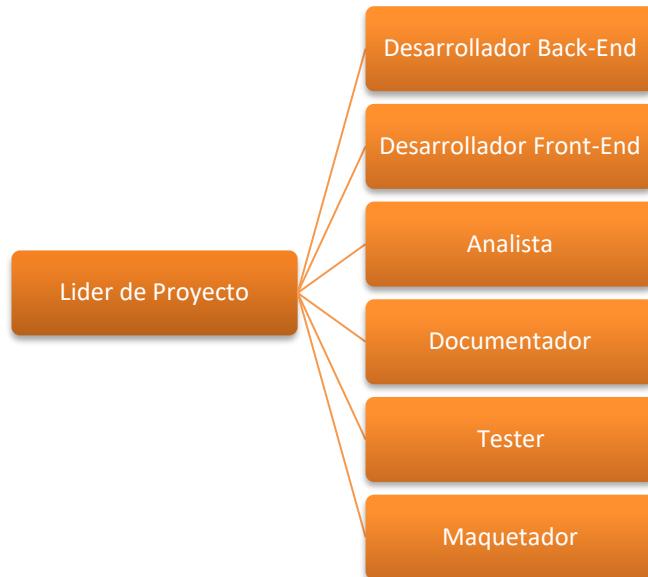


Ilustración 15 - Diagrama de Roles

(Fuente: Elaboración Propia)

Los roles que deberán cumplir cada integrante del equipo en el transcurso del desarrollo del sistema se observan en la siguiente tabla.

Rol	Funciones
Líder de Proyecto	<ul style="list-style-type: none">• Gestiona y asigna recursos humanos y de otro tipo.• Define las prioridades de las tareas dentro y/o relacionadas con el proyecto.• Coordina las interacciones con los clientes y los usuarios finales.• Planifica las iteraciones.

	<ul style="list-style-type: none"> • Planifica y asigna las tareas de la forma más razonable posible. • Define la organización y estructura del proyecto. • Establece las líneas de trabajo a seguir para garantizar la calidad e integridad de los artefactos del proyecto.
Desarrollador Back-End	<ul style="list-style-type: none"> • Motiva y organiza el equipo de trabajo para lograr un objetivo definido. • Establece los horarios de trabajo del equipo de desarrollo. • Planifica y realiza las reuniones de control del equipo de desarrollo en el tiempo establecido. • Mantiene el control del resultado de estas reuniones. • Informa sobre el estado actual del proyecto. • Mantiene el plan del proyecto. • Suministra al equipo el informe del ciclo de desarrollo. <ul style="list-style-type: none"> • Buscar todas las soluciones posibles a un problema, analizarlas y determinar la más efectiva y eficiente.

	<ul style="list-style-type: none"> • Aplicar las mejores soluciones para programar acciones y funciones. • Mantener siempre una actitud de autoaprendizaje y actualización de conocimientos, estudiando los lenguajes de programación vigentes. • Indagar las necesidades de los clientes y ofrecer las mejores soluciones en forma de proyectos factibles. • Identificar problemas en la cadena productiva, analizando todos los procesos, para definir dónde se generó y cómo solucionarlo. • Proponer soluciones en etapas tempranas para evitar problemas.
Desarrollador Front-End	<ul style="list-style-type: none"> • Desarrollará la parte visual de la web. • Diseña la estructura, la tipografía, la colorimetría, imágenes, banners, etc. • Diseñara un sitio web atractivo e intuitivo para el receptor. • Empleará los 3 principales lenguajes en el desarrollo web React JS, JS, HTML.
Analista	<ul style="list-style-type: none"> • Gestiona el proyecto en general. • Comunicación interpersonal.

	<ul style="list-style-type: none"> • Conocimientos de la Organización. • Capacidad de Negociación. • Adaptación al Cambio. • Investigación y Proactividad. • Identificación de roles y perfiles.
Documentador	<ul style="list-style-type: none"> • Documentara los manuales de usuario y administrador • Implementara Metodologías para la administración de proyectos
Tester	<ul style="list-style-type: none"> • Se realizarán pruebas en el dominio de pruebas para corroborar bugs o fallos. • Comprobar que los roles y validaciones funcionen correctamente. • Correcciones de ortografía y tipografía.
Maquetador	<ul style="list-style-type: none"> • Diseñara propuestas de diseño en adobe Photoshop. • Diseño de imágenes y corrección y tipografía.

Tabla 3 - Roles de Desarrolladores

(Fuente: Elaboración Propia)

3. DIAGRAMAS UML

4. Para diseño de sistemas:

- Casos de uso

Descripción: En este caso de uso, se visualiza el manejo del Sistema en General y la interacción que tiene cada actor sobre él, mostrando los 6 tipos de usuario y las actividades que cada uno de ellos ejercerá, los permisos los defino el Admin de Qualita, así que puede variar, pero actualmente es así como lo tienen.

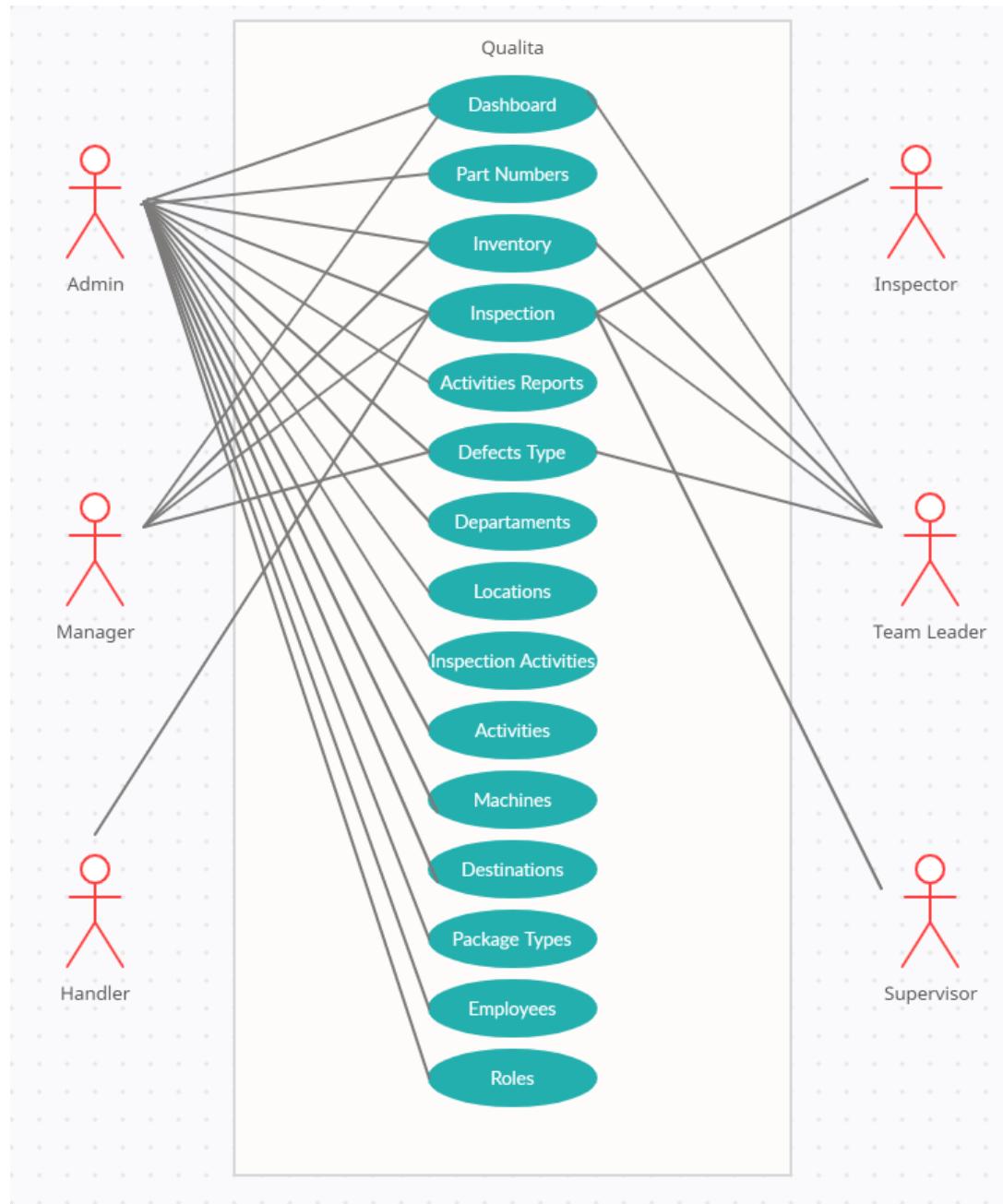


Ilustración 16 - Caso de uso

(Fuente: Elaboración Propia)

- **Secuencia**

Descripción: En este diagrama se representa el funcionamiento general del sistema de horarios, en donde una vez terminada deberá hacer estas interacciones con cada uno de los servicios integrados al proyecto, con la única finalidad de devolver la petición de datos solicitadas por el usuario.

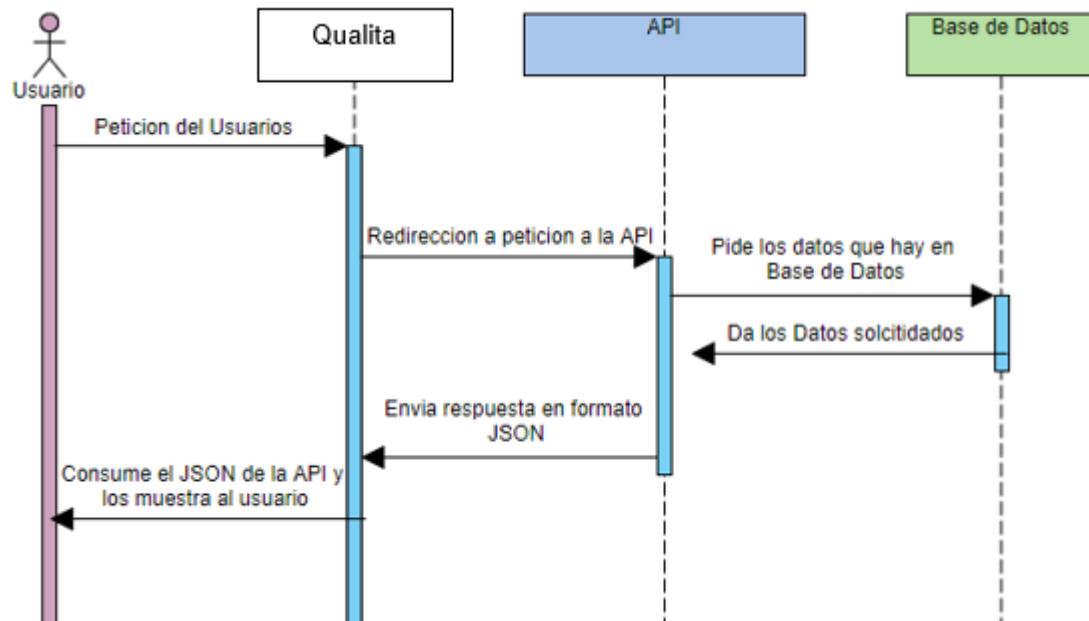


Ilustración 17 - Diagrama de Secuencia

(Fuente: Elaboración Propia)

• Colaboración

Descripción: En este diagrama se muestra el comportamiento dinámico del sistema, mostrando como interactúan todo el sitio Qualita

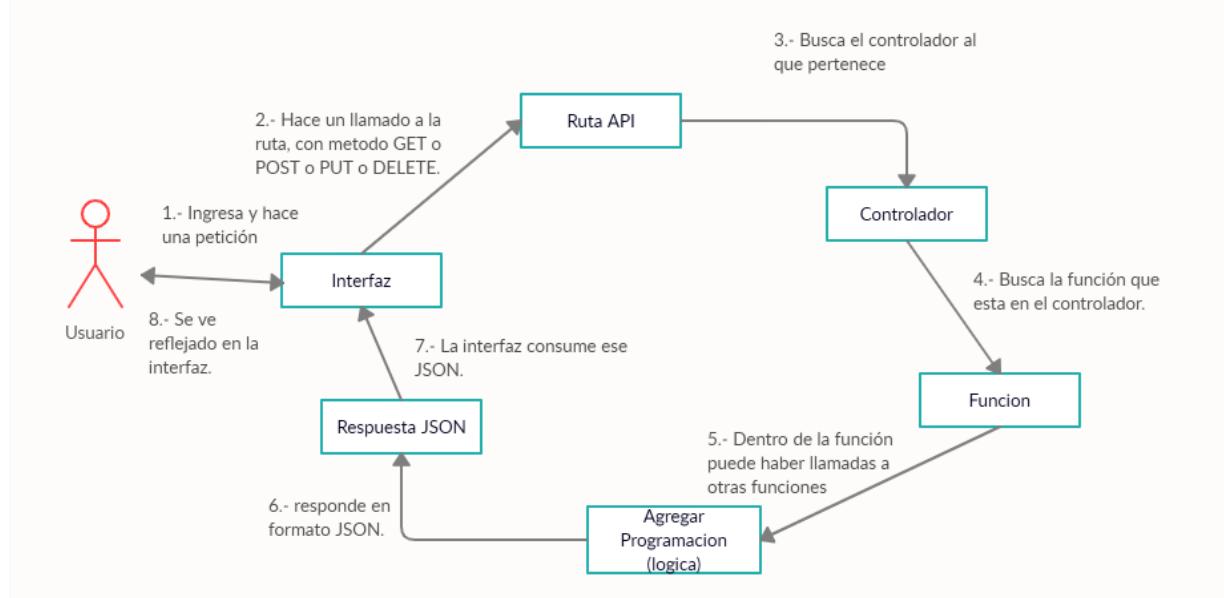


Ilustración 18 - Diagrama de Colaboración

(Fuente: Elaboración Propia)

- **Estado**

Descripción: En este diagrama se puede observar el funcionamiento de un alta de prospectos. Parte en un punto negro, donde se dará un alta, llenando los datos correspondientes en el formulario. Una vez concluido, se hará validación de los datos ingresados, los cuales si son correctos permitirá dar de alta de un prospecto y en dado caso de que haya un error lo mostrara en mensaje, en este caso correo electrónico y números de teléfonos tienen que ser únicos.

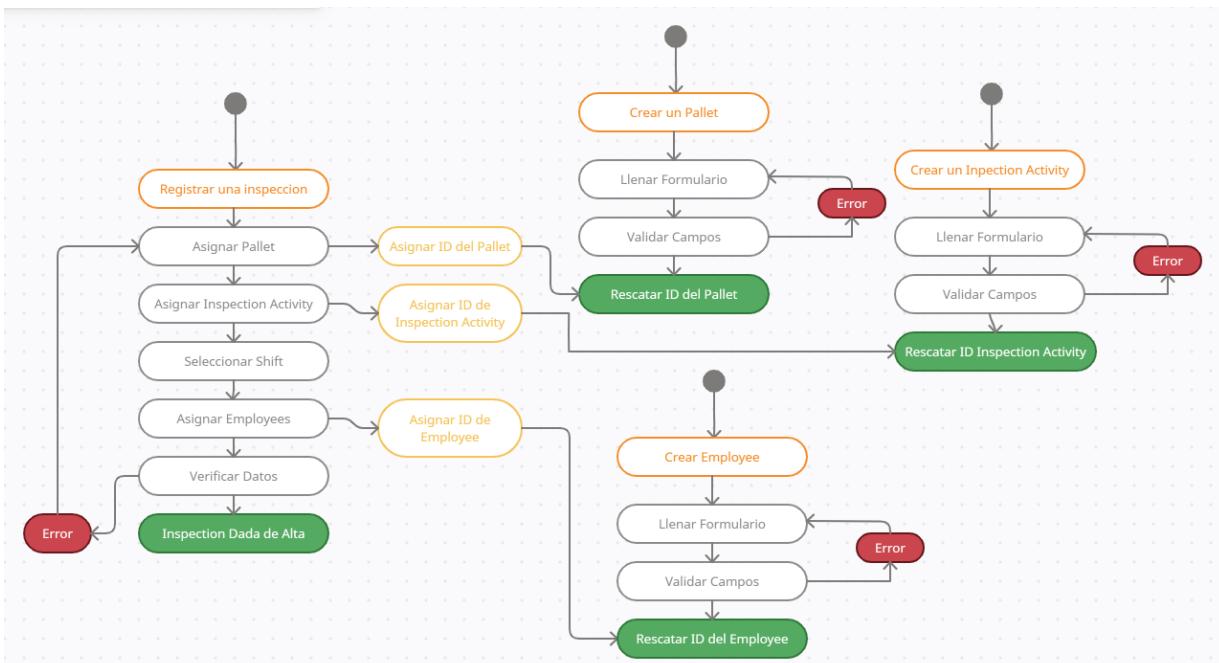


Ilustración 19 - Diagrama de Estado

(Fuente: Elaboración Propia)

- **Actividades**

Descripción: En el siguiente diagrama de flujo se ilustra el flujo de trabajo entre el usuario y el sistema, describiendo de mejor manera la lógica que se implementara detrás del sistema.

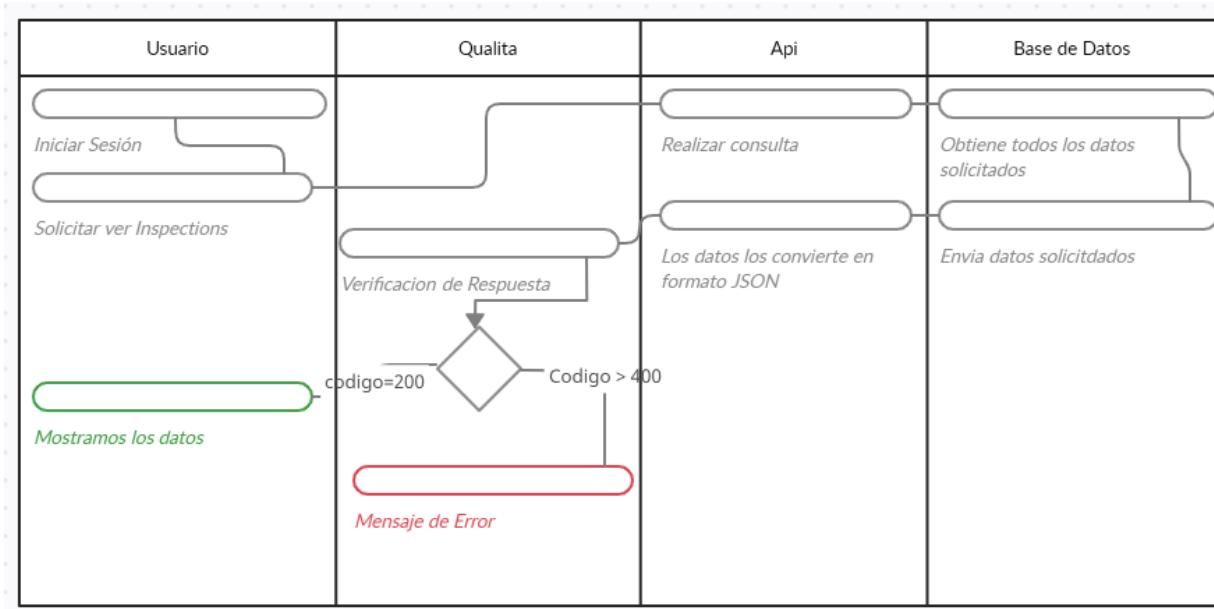


Ilustración 20 - Diagrama de Actividades

(Fuente: Elaboración Propia)

• Clases

Descripción: En este diagrama se puede observar las relaciones entre cada uno de los módulos que involucran el sistema, mostrando los atributos de cada módulo y sus operaciones.

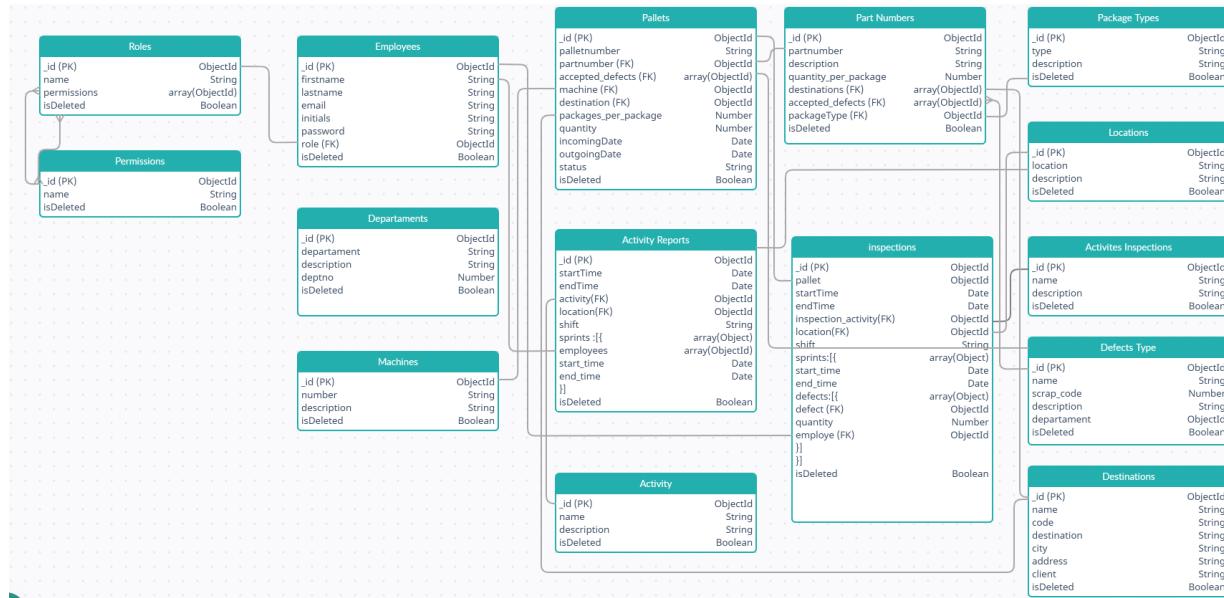


Ilustración 21 - Diagrama de Clases

(Fuente: elaboración Propia)

Para implementación

• Componentes

Descripción: en este diagrama se pueden observar los componentes que usa cada tipo de usuario, y así se reutilizan componentes a nivel Front-End solo con algunas validaciones dentro del componente para no tener errores. Las vistas también dependen de los roles, ya que el administrador del sistema podrá modificar las opciones de roles y permisos.

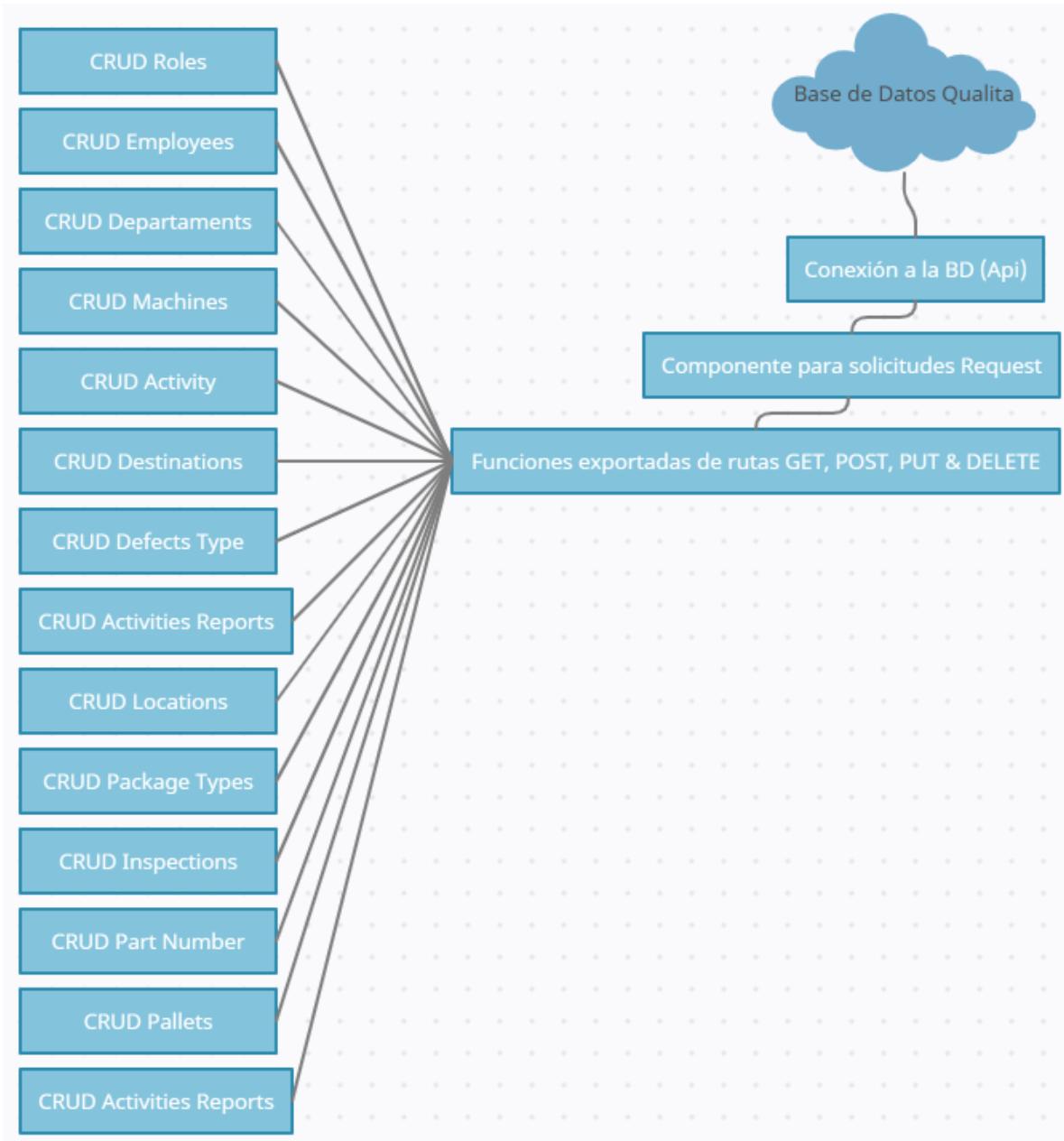


Ilustración 22 - Diagrama de Componentes

(Fuente: Elaboración propia)

- **Despliegue**

Descripción: En este diagrama se representa la distribución física de los componentes del aplicativo distribuidos en la red

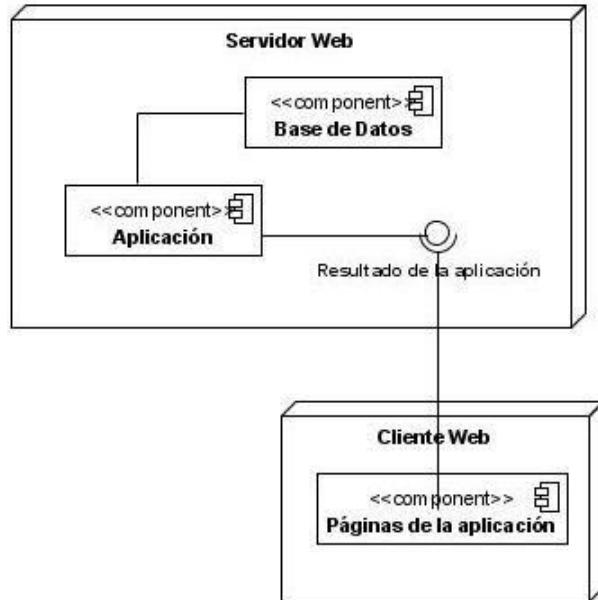


Ilustración 23 - Diagrama de Despliegue

(Fuente: Elaboración Propia)

Diagramas de la base de datos

- Diagrama relacional

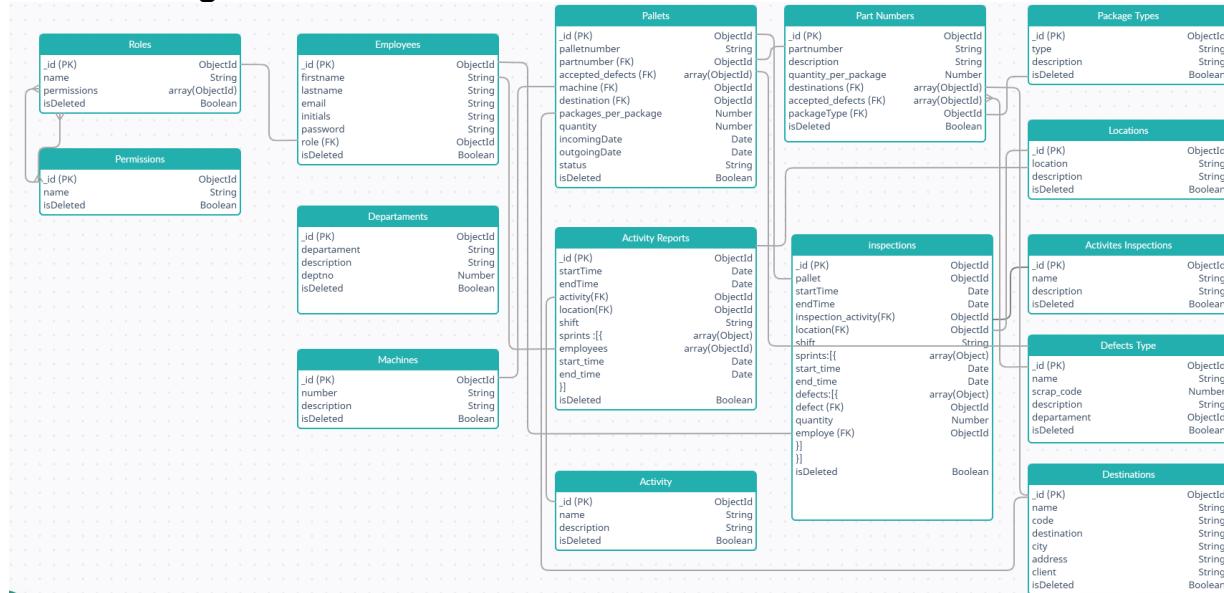


Ilustración 24 - Diagrama Relacional

(Fuente: Elaboración Propia)

- Diccionario de datos

Como es una Base de Datos NoSQL no tiene longitudes, y los tipos de datos son diferentes, por ejemplo, Varchar que es para texto, en MongoDB se maneja String

Tabla de Inspection

Campo	Tipo de Dato	PK/FK	Descripción.
<code>_id</code>	<code>ObjectId</code>	PK	Identificador
<code>pallet</code>	<code>ObjectId</code>	FK	Id del Pallet
<code>startTime</code>	<code>Date</code>		Fecha de cuando se inició la inspection
<code>endTime</code>	<code>Date</code>		Fecha de cuando se terminó la inspection
<code>activity</code>	<code>ObjectId</code>	FK	Id del Activity
<code>location</code>	<code>ObjectId</code>	FK	Id de Location
<code>shift</code>	<code>String</code>		Turno en que se creó la inspection ['First', 'Second', 'Third'].
<code>sprints</code>	<code>Array(Object)</code>		Array de objetos

sprints.employees	Array(ObjectId)	FK	Array de Ids de employees
sprints.start_time	Date		Tiempo en que se inició el sprint
sprints.end_time	Date		Tiempo en que se terminó el sprint
sprints.defects	Array(Object)		Array de Objetos
sprints.defects.defect	ObjectId	FK	Id del defect type
sprints.defects.quantity	Number		Cantidad de defectos
sprints.defects.employee	ObjectId	FK	Id del employee
updated_at	DateTime		Ultima fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 4 - Tabla de Inspection

(Fuente: Elaboración Propia)

Tabla de Pallet

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador
palletnumber	String		Nombre del Pallet
partnumber	ObjectId	FK	Id del PartNumber
accepted_defects	Array(ObjectId)	FK	Array de Ids de Defects Type
machine	ObjectId	FK	Id del Machine
destination	ObjectId	FK	Id de Destination
packages_per_pallet	Number		Numero de paquetes por pallet.
parts_per_package	Number		Numero de partes por paquete
quantity	Number		Cantidad total.
incomingDate	Date		Tiempo de entrada

outgoingDate	Date		Tiempo de Salida
status	String		Opciones ['Received', 'Certified', 'Shipped', 'On Hold']
updated_at	DateTime		Ultima fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 5 - Tabla de Pallets

(Fuente: Elaboración Propia)

Tabla de Part Numbers

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador
partnumber	String		Nombre del part number
description	String		Una breve descripción
quantity_per_package	Number		Cantidad por paquetes
destinations	Array(ObjectId)	FK	Ids de Destinations
accepted_defects	Array(ObjectId)	FK	Ids de Defects Type
packageType	ObjectId	FK	Id del Package Type
updated_at	DateTime		Ultima fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 6 - Tabla de Part Numbers

(Fuente: Elaboración Propia)

Tabla de Activity Reports

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador
startTime	Date		Fecha de cuando se inició la inspection
endTime	Date		Fecha de cuando se terminó la inspection
activity	ObjectId	FK	Id del Activity
location	ObjectId	FK	Id de Location
shift	String		Turno en que se creó la inspection ['First', 'Second', 'Third'].
sprints	Array(Object)		Array de objetos
sprints.employees	Array(ObjectId)	FK	Array de Ids de employees
sprints.start_time	Date		Tiempo en que se inició el sprint
sprints.end_time	Date		Tiempo en que se terminó el sprint
updated_at	DateTime		Ultima fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 7 - Tabla de Activity Reports

(Fuente: Elaboración Propia)

Tabla de Defects Type

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador
name	String		Nombre del defecto
scrap_code	Number		Numero de Scrap Code
description	String		Descripción del defecto

departament	ObjectId	FK	Id del Departament
updated_at	DateTime		Ultima fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 8 de Defects Type

(Fuente: Elaboración Propia)

Tabla de Employees

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador de la empresa
firtname	String		Primer nombre
lastname	String		Segundo nombre
email	String		Correo electrónico
initials	String		Iniciales del usuario
password	String		Contraseña
role	ObjectId	FK	Id del Rol
updated_at	DateTime		Ultima fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 9 de Employees

(Fuente: Elaboración Propia)

Tabla de Destinations

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador de la empresa
name	String		Nombre del destino
code	String		Código
destination	String		Breve Descripción
city	String		Nombre de la ciudad

address	String		Dirección
client	String		Nombre del cliente
updated_at	DateTime		Ultima fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 10 de Destinations

(Fuente: Elaboración Propia)

Tabla de Machines

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador de la empresa
number	String		Numero de la maquina (Puede contener otros caracteres)
description	string		Breve descripción
updated_at	DateTime		Ultima fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 11 de Machines

(Fuente: Elaboración Propia)

Tabla de Activities

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador de la empresa
name	String		Nombre de la Actividad
description	String		Breve Descripción
updated_at	DateTime		Ultima fecha de actualización
created_at	DateTime		Fecha de creación

isDeleted	Boolean	Validador de borrado lógico
-----------	---------	-----------------------------

Tabla 12 de Activities

(Fuente: Elaboración Propia)

Tabla de Activities Inspection

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador de la empresa
name	String		Nombre de la actividad inspection
description	String		Breve descripción
updated_at	DateTime		Última fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 13 de Activities Inspection

(Fuente: Elaboración Propia)

Tabla de Package Types

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador de la empresa
type	String		Nombre del paquete
description	String		Breve descripción
updated_at	DateTime		Última fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 14 de Package Types

(Fuente: Elaboración Propia)

Tabla de Locations

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador de la empresa
location	String		Nombre de la localidad
description	String		Breve description
updated_at	DateTime		Ultima fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 15 de Locations

(Fuente: Elaboración Propia)

Tabla de Departamentos

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador de la empresa
departament	String		Nombre del Departamento
description	String		Breve Descripción
deptno	Number		Numero del departamento
updated_at	DateTime		Ultima fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 16 de Departaments

(Fuente: Elaboración Propia)

Tabla de Roles

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador de la empresa
name	String		Nombre del Rol
permissions	ObjectId	FK	Ids de Permisos

updated_at	DateTime		Última fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 17 de Roles

(Fuente: Elaboración Propia)

Tabla de Permissions

Campo	Tipo de Dato	PK/FK	Descripción.
_id	ObjectId	PK	Identificador de la empresa
name	String		Nombre del Permiso
updated_at	DateTime		Última fecha de actualización
created_at	DateTime		Fecha de creación
isDeleted	Boolean		Validador de borrado lógico

Tabla 18 de Permissions

(Fuente: Elaboración Propia)

De la interfaz

- **Guía de estilos**

Guía de Estilos

- **Documentación de código fuente**

Documentación de Código Fuente

Pruebas

- **Test de usabilidad**

En la siguiente tabla se mostrarán los siguientes comentarios, quienes han probado Qualita.

1	La navegación cumple con lo solicitado, el home está bien distribuido para la cantidad de información que contiene
2	Las secciones están distribuidas correctamente, no hay problema alguno
3	La página está muy bien, solo falta revisar la ortografía.

4 Qualita es un sistema en inglés, aún hay pequeños detalles de redacción

Tabla 19 Tabla de Usabilidad

(Fuente: Elaboración Propia)

- **Plan de pruebas**

Nombre de la Prueba	Descripción de la Prueba	Fecha de la Prueba	Responsabilidad
CRUD Pallet	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	15/03/2022 17/03/2022	- Edgar González Martínez Axel
CRUD PartNumber	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	18/03/2022 20/03/2022	- Edgar González Martínez Axel
CRUD Inspection	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	20/03/2022 25/03/2022	- Edgar González Martínez Axel

Implementación de Sprints en Inspections	Corroborar que los Sprints estén realizados de una manera adecuada.	26/03/2022 01/04/2022	-	Edgar González Martínez	Axel
CRUD de Activity Reports	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	28/02/2022 05/03/2022	-	Edgar González Martínez	Axel
Implementación de Sprints en Activity Reports	Corroborar que los Sprints estén realizados de una manera adecuada.	06/03/2022 14/03/2022	-	Edgar González Martínez	Axel
CRUD de Activity Inspections	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios..	20/02/2022 26/02/2022	-	Edgar González Martínez	Axel
CRUD de Activities	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	11/02/2022 19/02/2022	-	Edgar González Martínez	Axel

CRUD de Defects Types	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	04/02/2022 10/02/2022	-	Edgar González Martínez	Axel
CRUD de Departaments	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	29/01/2022 03/02/2022	-		
CRUD Destinations	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	23/01/2022 29/01/2022	-		
CRUD Locations	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	16/01/2022 22/01/2022	-		
CRUD Machines	Corroborar que todo el CRUD funcione, sin falla	10/01/2022 15/01/2022	-		

		alguna, validaciones, campos obligatorios.		
CRUD de Package Types		Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	07/01/2022 10/01/2022	-
CRUD de Permissions	de	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	03/01/2022 05/01/2022	-
CRUD de Roles		Corroborar que todo el CRUD funcione, sin falla alguna, validaciones, campos obligatorios.	05/01/2022 06/01/2022	-
CRUD Employees	de	Corroborar que todo el CRUD funcione, sin falla alguna, validaciones,	06/01/2022 07/01/2022	-

	campos obligatorios.		
--	----------------------	--	--

Tabla 20 - Tabla de Plan de Pruebas

(Fuente: Elaboración Propia)

- **Casos de uso de pruebas**

Descripción del paso	Resultado Esperado
Abrir https://stone-semiotics-235117.web.app/	Debería mostrar un formulario para iniciar sesión
Ingresar Correo Electrónico y Contraseña	Si es administrador, debería mandarlo a la vista del administrador y en dado caso de que no, le mostraría la vista para el asesor
Crear Roles	Debería Crear Roles, ya con sus ciertos permisos.
Crear Employees	Debería Crear el employee.
Editar Employees	Debería Editar la información de la data seleccionada
Eliminar Employees	Debería Eliminar al usuario de una manera lógica para no perder los datos
Crear Package Types	Debería Crear el package type.
Editar Package Types	Debería Editar la información de la data seleccionada
Eliminar Package Types	Debería Eliminar de una manera lógica para no perder los datos
Crear Destination	Debería Crear el destination.
Editar Destination	Debería Editar la información de la data seleccionada
Eliminar Destination	Debería Eliminar de una manera lógica para no perder los datos

Crear Machines	Debería Crear el machine.
Editar Machines	Debería Editar la información de la data seleccionada
Eliminar Machines	Debería Eliminar de una manera lógica para no perder los datos
Crear Activities	Debería Crear el activities.
Editar Activities	Debería Editar la información de la data seleccionada
Eliminar Activities	Debería Eliminar de una manera lógica para no perder los datos
Crear Inspection Activities	Debería Crear el inspection activity.
Editar Inspection Activities	Debería Editar la información de la data seleccionada
Eliminar Inspection Activities	Debería Eliminar de una manera lógica para no perder los datos
Crear Locations	Debería Crear la location.
Editar Locations	Debería Editar la información de la data seleccionada
Eliminar Locations	Debería Eliminar de una manera lógica para no perder los datos
Crear Departaments	Debería Crear el departament.
Editar Departaments	Debería Editar la información de la data seleccionada
Eliminar Departaments	Debería Eliminar de una manera lógica para no perder los datos
Crear Defects Type	Debería Crear el defect type.
Editar Defects Type	Debería Editar la información de la data seleccionada
Eliminar Defects Type	Debería Eliminar de una manera lógica para no perder los datos
Crear Part Numbers	Debería Crear el part number.

Editar Part Numbers	Debería Editar la información de la data seleccionada
Eliminar Part Numbers	Debería Eliminar de una manera lógica para no perder los datos
Crear Pallet	Debería Crear el pallet.
Editar Pallet	Debería Editar la información de la data seleccionada
Eliminar Pallet	Debería Eliminar de una manera lógica para no perder los datos
Crear Inspection	Debería Crear el inspection.
Crear Sprints para Inspection	Debería crear crear varias sprints en una sola inspections
Editar Sprints de Inspections	Podrá editar los sprints en de una inspection
Crear Activities Reports	Debería Crear Activities Reports.
Crear Sprints para Activities Reports	Debería crear crear varias sprints en una sola activities Reports
Editar Sprints de Activities Reports	Podrá editar los sprints en de una activity report

Tabla 21 - Tabla de Uso de Pruebas

(Fuente: Elaboración Propia)

- Ejecución de pruebas**

1	Es responsive para móviles	No
2	Los usuarios están satisfechos	Si
3	El diseño es atractivo	Si
4	El sistema es entendible para los usuarios	Si

Tabla 22 - Ejecución de Pruebas

(Fuente: Elaboración Propia)

Implementación

- Plan de implementación**

Actividades	Acción	Duración	Responsable
Implementación de Borrado lógico	Se implementará un borrado lógico, para aun conservar los datos en base de datos y obtener estadísticas	2 días	Edgar Axel González Martínez
Implementación de alertas en cuentas suspendidas	En dado caso de borrar a un usuario, les mostrara una alerta diciendo que la cuenta fue eliminada	2 días	Edgar Axel González Martínez

Tabla 23 - Tabla de Plan de Implementación

(Fuente: Elaboración Propia)

5. DOCUMENTACIÓN

Manual de usuario

[Manual de Usuario](#)

CAPÍTULO V

RESULTADOS

5 RESULTADOS OBTENIDOS

5.1 RESULTADOS

El Sistema de Qualita, hubo demasiados cambios, se tomaron previas capturas antes de remodelar el sistema en cuestión de diseño, a continuación se observa el antes y el después .

LOGIN DEL SISTEMA

ANTES:

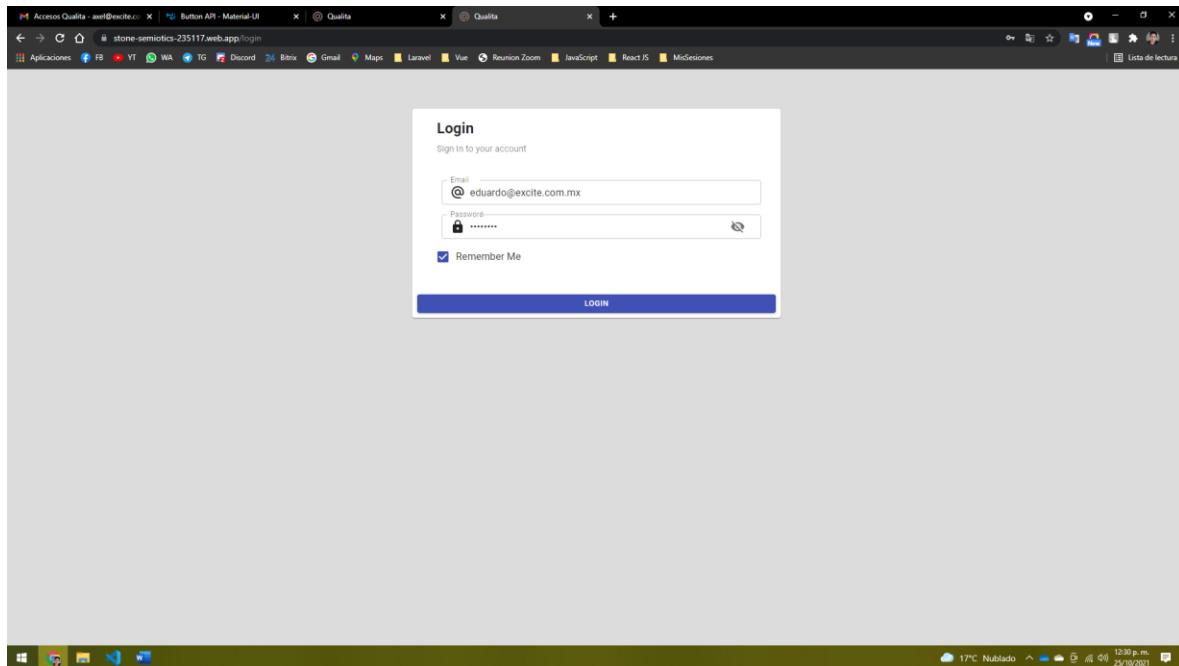


Ilustración 25 - Qualita Login Antes

(Fuente: Elaboración Propia)

DESPUES:

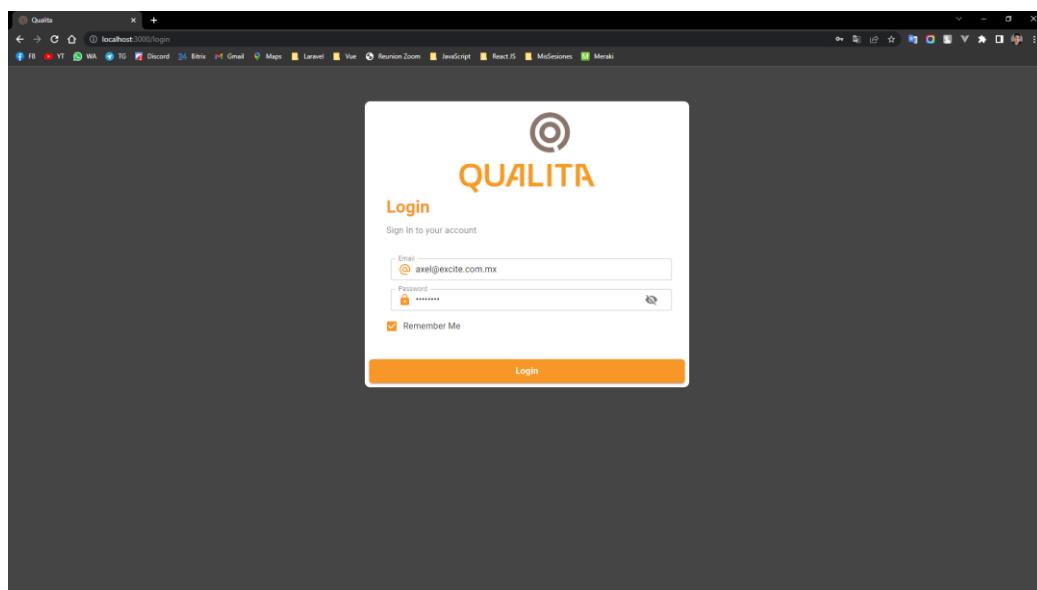


Ilustración 26 - Qualita Login Despues

(Fuente: Elaboración Propia)

SECCION DE MY ACCOUNT

ANTES:

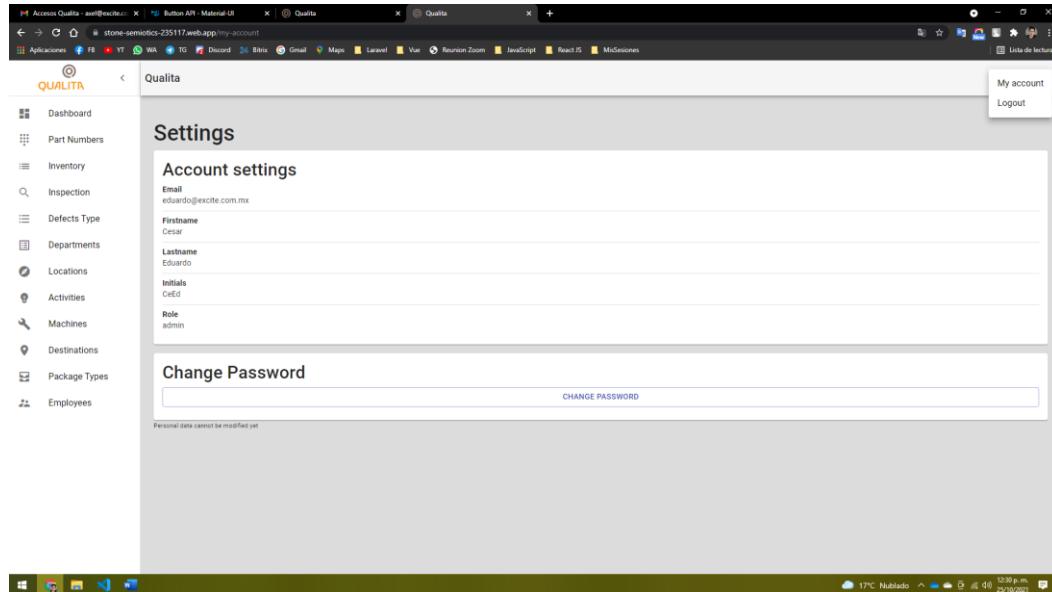


Ilustración 27 - Qualita Antes - Mi cuenta

(Fuente: Elaboración Propia)

DESPUES:

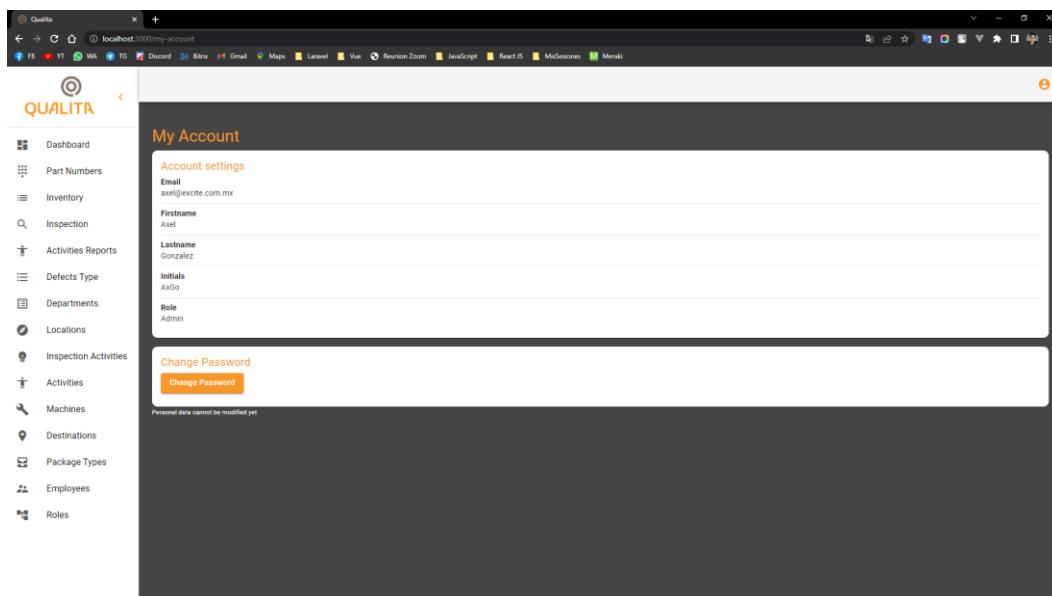


Ilustración 28 - Qualita Despues - Mi cuenta

(Fuente: Elaboración Propia)

SECCION EMPLOYEES LIST

ANTES:

The screenshot shows a web application interface for 'Employees List'. On the left is a sidebar with icons for Dashboard, Part Numbers, Inventory, Inspection, Defects Type, Departments, Locations, Activities, Machines, Destinations, Package Types, and Employees. The main area has a title 'Employees List' and a blue '+ ADD' button. A table lists employees with columns: Email, Initials, Firstname, Role, and Actions (represented by edit and delete icons). The data includes:

Email	Initials	Firstname	Role	Actions
info@qualita.net	ADMIN	Info	admin	[Edit] [Delete]
axel@excite.com.mx	AxGo	Axel	admin	[Edit] [Delete]
eduardo@excite.com.mx	CeEd	Cesar	admin	[Edit] [Delete]
manager@excite.com.mx	MaQu	Manager	manager	[Edit] [Delete]
teamleader@excite.com.mx	TeQu	Team Leader	team_leader	[Edit] [Delete]
supervisor@excite.com.mx	SuQu	Supervisor	supervisor	[Edit] [Delete]

At the bottom right of the table, there are buttons for 'Rows per page' (set to 10), '1-8 of 8', and navigation arrows.

Ilustración 29 - Qualita Antes - Employees

(Fuente: Elaboración Propia)

DESPUES:

The screenshot shows the same web application interface for 'Employees List' after changes. The sidebar remains the same. The main area now has a title 'Employees' and an orange '+ Add' button. A message 'Show Deleted Employees' is displayed above the table. The table structure and data are identical to the previous screenshot, showing the same list of employees with their respective roles and actions.

Ilustración 30 - Qualita Despues - Employees

(Fuente: Elaboración Propia)

SECCION PACKAGE TYPE

ANTES:

The screenshot shows a web browser window with the Qualita application open. The left sidebar contains navigation links: Dashboard, Part Numbers, Inventory, Inspection, Defects Type, Departments, Locations, Activities, Machines, Destinations, Package Types, and Employees. The main content area is titled 'Package Type' and displays a table with three rows. The columns are 'Type' and 'Description'. The first row is 'Box' with the description: 'Caja de carton, cantidad de piezas por caja diferente en cada numero de parte y un pallet completo tiene un determinado numero de cajas. Aquí podriamos tener la opción de pallet completo o 'other' y meter la cantidad de piezas o cajas manualmente.' The second row is 'Bins' with the description: 'Tinas de material No ensamblado, la cantidad varia cada vez y se tendría que ingresar manual unicamente.' The third row is 'Totes' with the description: 'Pequeños contenedores de plastico con cantidad especifica de piezas dependiendo del numero de parte. Estos se sortean y se re empacan en cajas de carton de acuerdo a la especificación de Toyota. Igualmente tendriamos que dar la opción de pallet completo o 'other' para meter la cantidad manualmente.' At the bottom of the table, there is a 'Rows per page' dropdown set to 10, and a message '1-8 of 3'.

Ilustración 31 - Qualita Antes - Package Types

(Fuente: Elaboración Propia)

DESPUES:

The screenshot shows the same web browser window with the Qualita application open, but the interface has been redesigned. The left sidebar remains the same. The main content area is titled 'Package Types' and displays a table with five rows. The columns are 'Type', 'Description', and 'Actions'. The first three rows are identical to the ones in the previous screenshot. The fourth row is 'package Delete' with the description 'DELETEEE'. The fifth row is also 'package Delete' but is highlighted with a gray background. At the bottom of the table, there is a 'Rows per page' dropdown set to 10, and a message '1-4 of 4'.

Ilustración 32 - Qualita Despues - Package Types

(Fuente: Elaboración Propia)

SECCION DESTINATIONS

ANTES:

The screenshot shows the 'Destinations List' page in the Qualita application. On the left is a sidebar with icons for Dashboard, Part Numbers, Inventory, Inspection, Defects Type, Departments, Locations, Activities, Machines, Destinations (which is selected), Package Types, and Employees. The main area has a title 'Destinations List' and a button '+ Add'. A table lists destinations with columns: Name, Destination Code, Destination, City, Address, and Actions (with edit and delete icons). The table contains six rows of data. At the bottom are pagination controls for 'Rows per page: 10' and '1-10 of 53'.

Ilustración 33 - Qualita Antes - Destinations

(Fuente: Elaboración Propia)

DESPUES:

The screenshot shows the same 'Destinations List' page after improvements. The sidebar remains the same. The main area now has a title 'Destinations' and a checkbox 'Show Deleted Destinations'. The table structure is identical to the previous version, showing the same six destination entries. The 'Actions' column includes edit and delete icons, which are now colored orange.

Ilustración 34 - Qualita Despues - Destinations

(Fuente: Elaboración Propia)

SECCION MACHINES LIST

ANTES:

Machine Number	Description	Actions
98	Enter a machine description	
101	Enter a machine description	
99	Enter a machine description	
100	Enter a machine description	
102	Enter a machine description	
103	Enter a machine description	
		Rows per page: 10 ▾ 1-10 of 25

Ilustración 35 - Qualita Antes - Machines

(Fuente: Elaboración Propia)

DESPUES:

Machine Number	Description	Actions
98	Enter a machine description	
101	Enter a machine description	
99	Enter a machine description	
100	Enter a machine description	
102	Enter a machine description	
103	Enter a machine description	
108	Enter a machine description	
111	Enter a machine description	
115	Enter a machine description	

Ilustración 36 - Qualita Después - Machines

(Fuente: Elaboración Propia)

SECCIONES ACTIVITIES LIST

ANTES:

Name	Description	Actions
Nuts	Material no ensamblado que se maneja en tinas, a granel	
Nuts Bolt Sort	Material no ensamblado que se maneja en tinas, a granel. Se inserta un tornillo en el bulto para verificar integridad de la cuerda.	
Caps	Material no ensamblado que se maneja en tinas, a granel	
Washers	Material no ensamblado que se maneja en tinas, a granel	
Black Parts General Sort	Partes negras que se someten a un proceso especial de sorteo	
Black Parts Re-pack	Partes negras que ya fueron sorteadas y solo se re empacan	

Ilustración 37 - Qualita Antes - Inspection Activities

(Fuente: Elaboración Propia)

DESPUES:

Name	Description	Actions
Nuts	Material no ensamblado que se maneja en tinas, a granel	
Nuts Bolt Sort	Material no ensamblado que se maneja en tinas, a granel. Se inserta un tornillo en el bulto para verificar integridad de la cuerda.	
Caps	Material no ensamblado que se maneja en tinas, a granel	
Washers	Material no ensamblado que se maneja en tinas, a granel	
Black Parts General Sort	Partes negras que se someten a un proceso especial de sorteo	
Black Parts Re-pack	Partes negras que ya fueron sorteadas y solo se re empacan	
Bolt Sort	Proceso de sorteo en el cual se inserta un tornillo en el bulto para verificar integridad de la cuerda.	
Making Boxes	Se arman cajas para produccion	
Down Time	Tiempo muerto en espera de material	

Ilustración 38 - Qualita Despues - Inspection Activities

(Fuente: Elaboración Propia)

SECCION LOCATION LIST

ANTES:

Location	Description	Actions
Qualita Global Troy	Qualita Global Troy	
MacLean-Fogg Component Solutions	Description	

Ilustración 39 - Qualita Antes - Locations

(Fuente: Elaboración Propia)

DESPUES:

Location	Description	Actions
Qualita Global Troy	Qualita Global Troy	
MacLean-Fogg Component Solutions	Description 2	
location	desc	

Ilustración 40 - Qualita Despues - Locations

(Fuente: Elaboración Propia)

SECCION DEPARTMENTS

ANTES:

Departments List

Department	Department No.	Description	Actions
Assembly	827	827	
Forming	822	822	
General	891	891	
Heat Treat	999	999	
ICOP	826	826	
Knurling	822	822	

Ilustración 41 - Qualita Antes - Departaments

(Fuente: Elaboración Propia)

DESPUES:

Departments

Department	Department No.	Description	Actions
Assembly 2	827	827	
Forming	822	822	
General	891	891	
Heat Treat	999	999	
ICOP	826	826	
Knurling	822	822	
Multiple	891	891	
Plating	999	999	
Supplier	999	999	

Ilustración 42 - Qualita Despues - Departaments

(Fuente: Elaboración Propia)

SECCION DEFECTS LIST

ANTES:

Name	Scrap code	Department	Department No.	Actions
Bad Plating/Coating	44	Plating	999	
Bolt Thread Damage	33	Tapping	822	
Burnt Cones	9	Heat Treat	999	
Burrs	49	Forming	822	
Cap Flange Damage	50	ICOP	826	
Cone Damage	53	Multiple	891	

Ilustración 43 - Qualita Antes - Defect Types

(Fuente: Elaboración Propia)

DESPUES:

Name	Scrap code	Department	Department No.	Actions
Bad Plating/Coating	44	Plating	999	
Bolt Thread Damage	33	Tapping	822	
Burnt Cones	9	Heat Treat	999	
Burrs	49	Forming	822	
Cap Flange Damage	50	ICOP	826	
Cone Damage	53	Multiple	891	
Damaged Dome	51	ICOP	826	
Damaged Hex	0			
Damaged Threads	0			

Ilustración 44 - Qualita Despues - Defect Types

(Fuente: Elaboración Propia)

Actualmente son 92 permisos, pero tiene apertura para crecer, dependiendo de las necesidades a futuro.

También se ha implementado un reporte en Excel, en el Dashboard

The screenshot shows a Microsoft Excel spreadsheet titled "Report_Dashboard (1).csv". The data is organized into several sections:

- Header Rows:** Contains column headers such as Week Month, Location, Part Number, Description, Machine ID, Date, Scrape Reason, Qty, Scrape, Scrap Code, Defect Number, Shift, Name, Team, Total Start Spent Time, End Spent Time, Total Spent Time, Work Spent, Sprint Number, and more.
- Body Data:** Detailed inspection records for various parts (Axle, Teeth, etc.) across different locations (Fábrica, MacLean-Fogg Component Solutions) and dates (February 2024).
- Calculated Columns:** Includes columns for Total Start Spent Time, End Spent Time, Total Spent Time, Work Spent, and Sprint Number.
- Summary Row:** A row at the bottom provides a summary of the total inspection time for the entire period.

Ilustración 115 - Qualita nuevo - Reporte Excel

(Fuente: Elaboración Propia)

5.2 TRABAJO FUTURO

Con este proyecto yo recomendaría realizar graficas en el Dashboard, de cuantas piezas hay defectuosas en un mes y cuantas están en perfecto estado, esto para saber el control de calidad que están teniendo, también una gráfica de tiempos, cuanto tiempo es que se están tardando en realizar estas graficas.

Los administradores puedan organizar de manera adecuada los roles y permisos para una mayor organización del sistema.

CONCLUSIONES

Con este proyecto aprendí a desarrollar en un lenguaje nuevo React JS, aprendí a implementar los roles y permisos, básicamente con cada actualización o implementación que se le agrego al sistema, se crearon nuevos permisos. El desarrollo de este proyecto me dejo a tener una mayor organización por componentes ya que se pueden volver a reutilizar en un futuro, aprendí que los componentes entre más cortos sean mucho mejor ya que serán mucho más sencillos de leer.

REFERENCIAS DE CONSULTA

Referencias Hemerográficas

Pimienta García, Rodrigo; Aguilar Torres, Gualberto; Ramírez Flores, Manuel; Gallegos García, Gina. (3, noviembre, 2014). Métodos de programación segura en Java para aplicaciones móviles en Android. *Métodos de programación segura en Java para aplicaciones móviles en Android*, 21, 7.

ARBOLEDA, ORLANDO. (agosto, 2011). DEFINICIÓN DEL LENGUAJE DE PROGRAMACIÓN EPLOAM PARA LA EJECUCIÓN DE PSEUDOCÓDIGO Y SU COMPILADOR. *Scientia Et Technica*, XVI, pp. 116-121.

Botero Tabares, Ricardo de Jesús; Castro Castro, Carlos Arturo; Parra Castrillón, José Eucario. (febrero-abril, 2006). Método integrado de programación secuencial y programación orientada a objetos para el análisis, diseño y elaboración de algoritmos - MIPSOO. *Revista Virtual Universidad Católica del Norte*, 17, 26.

Referencias Digitales

12207, I. (2008). International Standard: Information Technology. Software Life Cycle Processes. *ISO/IEC*.

Alfonso, R. (2005). *Metodología para Estructurar Proyectos*. Caracas, Venezuela.

Alvarez, M. A. (08 de Abril de 2008). *Desarrolloweb*. Obtenido de Desarrolloweb: <https://desarrolloweb.com/articulos/notepad.html>

Computer. (2002). Obtenido de Computer.

devcode. (s.f.). Obtenido de *devcode*: <https://devcode.la/blog/frontend-y-backend/>

devcode. (2018). Obtenido de *devcode*: <https://devcode.la/blog/frontend-y-backend/>

EcuRed Conocimiento con todos y para todos. (08 de Agosto de 2017). Obtenido de EcuRed Conocimiento con todos y para todos: <https://www.ecured.cu/SQLyog>

EcuRed Conocimiento con todos y para todos. (08 de Agosto de 2017). Obtenido de EcuRed Conocimiento con todos y para todos: <https://www.ecured.cu/SQLyog>

HAINE, A. B. (2011). *HTML5 MASTERY*. friendsof.

Handbook, I. V. (2009). Ashley20Hanna, John Windebank, Simon Adams, John Sowerby, Stuart Rance. *The Stationery Office*.

IBM,neobux,maxcdn,famo.us. (2017). *jquery write less, do more*. Obtenido de jquery write less, do more: <https://jquery.com/>

IEEE. (2004). *SWEBOK. Knowledge*.

IEEE, S. (1998). IEEE. IEEE Std 1074-1997. *IEEE*.

informaticabasica. (2016). Obtenido de *informaticabasica*: https://www.gcfaprendelibre.org/tecnologia/curso/informatica_basica/empezando_a_usar_un_computador/1.do

Kruchten, P. (2003). *The Rational Unified Process An Introduction*. USA: Addison Wesley.

Mantenimientodeequiposeinstalaciones. (13 de septiembre de 2017). Obtenido de Mantenimientodeequiposeinstalaciones:

<https://www.talent.upc.edu/esp/professionals/presentacio/codi/300600/mantenimiento-equipos-instalaciones/>

MARCOVITCH, J. (1993). *Estrategia, planificación y gestión de ciencia y tecnológica.*

Northware Software and Apps. (22 de Enero de 2016). Obtenido de Norrhware Software and Apps: www.northware.mx/perfiles-y-sus-funciones-en-proyectos-de-ti/

PMBOK Guide, I. P. (2004). A Guide to the Project Management Body of Knowledge. *Project Management Institute Inc. USA.*

Postman. (04 de Octubre de 2018). Obtenido de Postman: <https://www.getpostman.com/>

Procesos de Software. (2017). Obtenido de Procesos de Software.

SANDERS, D. H. (2001). *Informática: Presente Y Futuro.* México: McGraw Hill.

sistemasweb. (20 de junio de 2013). Obtenido de sistemasweb: <http://fraktalweb.com/blog/sistemas-web-para-que-sirven/>

Trachtenberg, D. s. (2014). *PHP Cookbook.* O'Reilly Media.

University, C. M. (2006). CMMI. Capability Maturity Model Integration Version 1.2. CMMI-DEV for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing. *Software Engineering Institute, USA.*

zapata, c. (2011). *Mantenimiento de una computadora.* Obtenido de Mantenimiento de una computadora: <http://mantenimientosdeunapc.blogspot.mx/2011/11/que-es-xampp-y-para-que-sirve.html>

zapata, c. (2011). *Mantenimiento de una Computadora.* Obtenido de Mantenimiento de una Computadora: <http://mantenimientosdeunapc.blogspot.mx/2011/11/que-es-xampp-y-para-que-sirve.html>

EALDE. (2016). *El diagrama del modelo académico Scrum.* 11-08-2020, de Web Sitio web: <https://www.ealde.es/diagrama-modelo-academico-scrum/>

Referencias Bibliográficas

Marijn Haverbeke. (2018). Eloquent Javascript, 3rd Edition: A Modern Introduction to Programming. United States: No Starch Press.

Diego Gauchat Juan. (2021). LIBRO AO El Gran Libro De Html5, Css3 Y Javascript 3^a Ed.. México: Alfaomega, Marcombo.

J. Francisco. (2016). Desarrollo web con HTML 5. Mexico: Alfaomega.

ANEXOS

[Clic Para ver el Video](#)