



**Universidad
Tecnológica
de Pereira**

Facultad de ingenierías

Prototipo de portal web para el acceso a modelos de redes neuronales.

Entregado por:

Johan Felipe Marín González

Director del proyecto:

Juan David Hincapié Zea

Universidad Tecnológica de Pereira

Pereira - Risaralda

2020

Contenido

1. Introducción	1
1.1. Presentación del proyecto	1
1.2. Planteamiento del problema.....	1
1.3. Objetivos	2
1.3.1. Objetivo general	2
1.3.2. Objetivos específicos	2
1.4. Justificación	2
1.5. Límites	3
1.6. Metodología	3
1.6.1. El recurso Humano.....	3
1.6.2. Gitlab y el board de tareas.....	4
1.6.3. Estructura de descomposición del trabajo.....	4
1.6.4. Herramientas tecnológicas.....	5
2. Marco teórico	5
2.1. Ingeniería del software.	6
2.2. Metodologías ágiles.....	7
2.3. Páginas web 3.0	8
2.4. Inteligencia artificial	9
3. Ejecución del proyecto	10
3.1. Funciones del producto.	10
3.2. La historia del portal web.....	10
3.3. Análisis y desarrollo.....	11
3.3.1. Identificación tipos de Usuario.	11
3.3.2. Empresas operadoras.....	12
3.3.3. Funcionamiento red neuronal.	12
3.3.4. Formulario de simulación de casos.	13
3.3.5. Modelo vista controlador	13
3.4. Pruebas de software.....	14
3.5. Documentación.....	14
3.5.1. Mockups del prototipo	14
3.5.2. Diseño base de datos.	14
3.5.3. Proceso ejecución de casos.	15
3.5.4. Manual de instalación del entorno de desarrollo.	15
3.5.5. Manual de instalación de despliegue en intranet	15

4. Conclusiones y recomendaciones.....	15
4.1. Conclusiones	15
4.2. Recomendaciones.....	16
4.3. Trabajos futuros.....	16
5. Anexos.....	17
6. Referencias bibliográficas.....	18

1. Introducción

1.1. Presentación del proyecto

El desarrollo de la ciencia y la tecnología viene en rápido aumento en los últimos años, creando así perspectivas o caminos para el desarrollo social y económico sustentable, es importante resaltar que en la actualidad las máquinas nos acompañan en nuestras labores diarias y pueden tomar decisiones importantes que son imperceptibles para la mayoría de las personas, estas decisiones son tomadas la mayoría de las veces por modelos de inteligencia artificial mediante el uso de la información.

Para que las personas del común puedan nutrir de información estos modelos de IA y obtengan una especie de beneficio o valor, se deben crear unas interfaces que estén disponibles, sean confiables y mantengan la integridad de la información, estas interfaces son las encargadas de recibir la información del usuario, entenderse con la inteligencia artificial y entregar algún resultado a las personas interesadas.

1.2. Planteamiento del problema

Existe un modelo de redes neuronales ya implementado en Matlab el cual está listo para su uso, para el cual no existe una herramienta que sirva de interfaz a las personas con pocos conocimientos en computación, se resalta que antes existía una herramienta que alojaba este servicio, pero ha quedado obsoleta y resulta muy costosa su actualización.

se ha llegado a conclusión que existe una brecha entre la comunidad interesada y la tecnología, este modelo debe ser usado por varios tipos de usuarios de diferentes entidades y con distintos privilegios, esto con la finalidad de ejecutar casos en el modelo de redes

neuronales, autorizar a otros usuarios y permitir a las diferentes entidades ejecutar casos en la inteligencia artificial.

1.3. Objetivos

1.3.1. Objetivo general

Diseñar e implementar un prototipo de portal web con registro, ingreso y autenticación de usuarios para el uso de herramientas de inteligencia artificial.

1.3.2. Objetivos específicos

- Identificar los requerimientos funcionales y no funcionales del portal web
- Diseñar e implementar módulo para registro e ingreso de usuarios
- Diseñar e implementar módulo para registro e ingreso de empresas.
- Diseñar e implementar módulo para el registro de la información de entrada del modelo de inteligencia artificial.

1.4. Justificación

Existe un software muy antiguo el cual sirve de interfaz entre una red neuronal que hace unas predicciones específicas y unos usuarios que hacen uso de esos resultados, esta interfaz se ha quedado rezagado en actualizaciones y presenta fallas en seguridad, entre otras.

A priori se ha hecho el análisis y se ha determinado que es mejor crear una nueva interfaz, entre la red neuronal y las personas que usan la aplicación, por lo tanto, se plantea en este proyecto la creación de un prototipo de portal web con sistema de usuarios y logueo para satisfacer todas las necesidades del negocio.

Se ha evitado la actualización de la anterior plataforma, ya que la versión actual le lleva dos versiones de ventaja al sistema actual y su estructura ha cambiado por completo hasta el punto de que en sus comunidades se ha dejado de dar soporte a la versión actual.

1.5. Límites

En este proyecto contempla la construcción de un prototipo de portal web, con módulos de usuarios, empresas y casos cuyo propósito es servir de interfaz entre diferentes tipos de usuarios de distintas entidades y una red neuronal ya prefabricada para que con los datos que estos usuarios entregan por medio de un formulario se pueda ejecutar un caso y notificar por correo el fin de la transacción, almacenando todos los datos representativos de manera permanente en una base de datos.

Este proyecto de aplicación se lleva a cabo en la Universidad Tecnológica de Pereira con el apoyo del docente guía Juan David Hincapié Zea en el periodo de tiempo que comprende el segundo semestre del año 2019.

1.6. Metodología

1.6.1. El recurso Humano

En el desarrollo de este proyecto estuvieron involucradas tres personas, las cuales se distribuyen tareas de forma sistemática para poder construir el prototipo, se plantearon reuniones semanales para hacer evaluaciones globales de los avances, y día a día se asignan tareas cortas para realizar en el menor tiempo posible, y cumplir con objetivos concretos, cada que se terminan las tareas planteadas se replantean nuevas y se va iterando de manera muy ágil creando un prototipo de manera incremental.

1.6.2. Gitlab y el board de tareas

Gitlab es un servicio web de control de versiones y desarrollo de software colaborativo que se basa en git, este servicio web brinda una board con cuatro secciones para crear tareas, tareas **“open”** en las cuales se ubican tareas que se deben realizar pero tiene baja prioridad, las tareas **“to do”**, las cuales se plantean como tareas a realizar a corto plazo, estas tareas esperan ser resueltas en un día o dos, máximo un tiempo de una semana; existe una sección llamada **“doing”** en la cual cada miembro pone lo que hace actualmente, se espera que cuando una persona abre una tarea y termina está, desplace la tarea a la sección **“close”** indicando que esa tarea ha sido terminada, si no la termina se espera que cuando deje de trabajar desplace esta tarea a la sección **“to do”** para que después pueda retomarla

Cuando una persona empieza a hacer una tarea del **“to do”**, se espera que mueva esta a la sección del doing y aquí cree todas las tareas complementarias para realizar esa tarea, teniendo así la descripción atómica de las actividades que se realizan.

1.6.3. Estructura de descomposición del trabajo

Las tareas que se crean en el board son en base al EDT, la EDT representa el panorama completo de tareas y funcionalidades a desarrollar, tanto tareas de diseño, de levantamiento de requerimientos, documentación, pruebas, entre otras se encuentran expresadas aquí, asignando a cada tarea una respectiva dificultad, esta herramienta nos ayuda a obtener la ruta crítica y marca el camino sobre qué tareas atacar primero, cuales son indispensables y cuáles pueden quedar por fuera.(ver anexo 1)

1.6.4. Herramientas tecnológicas

Para el desarrollo del prototipo se consideró hacer uso de herramientas modernas a nivel de programación y tecnología, para lo cual se cuenta con las herramientas de software que se listan a continuación:

- JavaScript
- nodejs
- nodemailer
- sequelize
- express
- PostgreSQL
- ejs
- git
- gitlab
- css
- bootstrap
- psexec
- máquina virtual de Matlab
- http
- red neuronal de Matlab.

2. Marco teórico

A continuación, se hace referencia a los conceptos que se manejan de forma implícita y explícita el desarrollo del prototipo, se parte siempre del conocimiento existente.

2.1. Ingeniería del software.

Para hablar de la ingeniería del software hay que tener en cuenta el crecimiento en la importancia del software frente al hardware. García-Peñalvo (2018) afirma que. “a mediados de los años 60 los costes del software ascendieron hasta un 40-50% del coste total del sistema, y su influencia fue creciendo hasta niveles en los que el coste del hardware ya representaba tan solo el 20% del total” (cap. 7, p 2), convirtiendo en un hecho el crecimiento de la importancia del software frente al hardware.

En la actualidad las empresas desarrollan o compran software para optimizar sus procesos de ahí la necesidad de una disciplina que se dedique a crear soluciones efectivas, con poco costo y que cumpla con su función, acompañando a los diferentes sistemas del mundo digital que buscan la homeostasis.

El software es parte de nuestra realidad, es usado por múltiples personas en el mundo y estas se vuelven cada vez más críticas, esperan que cumplan sus necesidades, que sean rápidas, que sean seguras, por eso el tema de crear software requiere de un análisis, diseño, construcción y una implementación de una manera sistemática, lo cual podemos lograrlo con la aplicación de la ingeniería del software, esta nos brinda las herramientas para construir un producto de software que sea usable, confiable, que cumpla unos requerimientos previos, también nos da las pautas para mantenerlo durante toda su vida útil.

Entonces podemos ver la ingeniería del software como el arte de crear soluciones computacionales y gracias a las necesidades de las personas y la aceptación del software se mantiene ubicua la computación.

El software y la tecnología son cambiantes, por lo tanto, debemos mantenerlo a la vanguardia, por eso si un software no se mantiene a lo largo de su producción este podría llegar al punto de ser obsoleto, inseguro, en pocas palabras pierde su estabilidad y la mejor solución puede ser volverlo a construir.

2.2. Metodologías ágiles

Hay que tener en cuenta que se tiene una manera ya planteada de hacer las cosas, con el avance de las tecnologías se han generado formas de crear software como lo son las metodologías.

Las metodologías tradicionales imponen una disciplina de trabajo sobre el proceso de desarrollo del software, con el objetivo de conseguir un software más eficiente y predecible. Para ello, se hace un especial hincapié en la planificación total de todo el trabajo a realizar y una vez que está todo detallado, comienza el ciclo de desarrollo del producto software. . (Cáceres, P., Marcos, E., & Kybele, G., 2001. p.2)

En este orden de ideas debemos planificar todo para poder construirlo, siendo muy rigurosos y burocráticos en el proceso haciendo que las personas eviten el uso de metodologías a la hora de programar y trabajen al paso, con el objetivo de ahorrar tiempo y dinero.

Gracias al estudio de la ingeniería del software se han creado nuevas formas de desarrollar el software, una de estas formas son las metodologías ágiles, que a diferencia de las metodologías tradicionales incluye más al cliente ya que este puede ver como su aplicación va tomando forma, poco a poco, en plazos pequeños, puede sugerir ediciones y puede darle uso inmediato, dando así motivación y seguridad, por otro lado a las personas encargadas de crear la aplicación se les permite ir construyendo un software que sea susceptible a los cambios, no se arriesga tanto como en las metodologías tradicionales, ya que no tenemos que esperar que todo el proceso de ingeniería termine para recibir o generar las observaciones y cambios que pueden resultar muy complejos de solucionar.

La gran ventaja de estas metodologías es que nos permite iterar en las entregas del producto final, con el objetivo de obtener un incremento en cada iteración, generando así la confianza a todos los involucrados, también es muy buena para la creación de prototipos

funcionales ya que como siempre se itera de manera ágil , se pueden obtener resultados funcionales con pocas iteraciones y en poco tiempo, analizando requerimientos, diseñando, construyendo, probando, dando el tiempo de corregir errores entre iteración y generando siempre una adición en las funcionalidades.

Según Pressman (1988) “El desarrollo ágil proporciona beneficios importantes, pero no es aplicable a todos los proyectos, productos, personas y situaciones” (p 55 - 56). por eso hay que ser muy cuidadoso cuando se decida trabajar con una metodología de este tipo; para el problema que quiere resolver este proyecto es beneficioso ya que se puede iterar sobre componentes y funcionalidades.

2.3. Páginas web 3.0

En la actualidad nos hablan de las web como algo general, pero esta ha venido evolucionando a lo largo de la historia, empezamos el mundo de las páginas web con la web 1.0 en la cual solo se disponía de páginas estáticas escritas por lo general con HTML, luego pasamos a las web 2.0 en la cual Martín, A. R., & Martín, M. J. R.(2014) refiere que a mediados de los noventa las empresas empezaron a ofertar sus productos por medio de la web, en las que destaca a Yahoo, EBay y Amazon, lo que hicieron fue crear páginas web dinámicas con sistemas de gestión de contenidos, un gran aporte para lo que empezó a convertirse la web.

Ahora planteamos tener una red inteligente, la cual podemos acceder de manera inmediata con el simple hecho de tener un celular en nuestros bolsillo, según Martín, A. R., & Martín, M. J. R.(2014) la Web 3.0 supone una revolución en la web basado en datos, los datos se convierten en información y dan valor a las personas, podemos integrar datos y conocimientos; y nuevas formas de conectarse a la web a través de una variedad de máquinas capaces de la búsqueda, la combinación y análisis de datos.

Dicho de esta manera, en la actualidad no solo los humanos van a usar la web para buscar cosas, también tendremos tipos de inteligencias, o procesos programados que se encargan de interactuar con las personas y con otros softwares para solucionar problemas de la vida cotidiana, teniendo la posibilidad de tomar decisiones en ambientes controlados y generar algún tipo de valor.

2.4. Inteligencia artificial

Como recalca. Takeyas (2007) “Los seres humanos, a diferencia de otras especies, tienen la capacidad de razonar sobre una serie de percepciones de hechos y proposiciones estableciendo relaciones entre sí. A esta capacidad se le llama inteligencia.” (p. 1). aplicando esta lógica y definición el ser humano a llevado la computación hasta el punto de crear máquinas “inteligentes”, capaces de resolver problemas específicos o hacer predicciones con datos que existen en su entorno, con la diferencia que una máquina es muy potente y su nivel de respuesta es mucho más rápido que el de un ser humano.

en este documento se recoge la IA como “una rama de las ciencias computacionales encargada de estudiar modelos de cómputo capaces de realizar actividades propias de los seres humanos con base a dos de sus características primordiales: el razonamiento y la conducta” (Takeyas,2007, p1), la cual ha permitido el uso de herramientas como las redes neuronales las cuales aprenden de la información que se almacena de alguna forma, y luego quedan listas para hacer predicciones sobre lo que fueron entrenadas, entender esto es importante ya que el fin del prototipo es servir de interfaz para un modelo de red neuronal, pero no es importante entender a fondo que es una red neuronal ni ser un experto en IA, cabe resaltar que las aplicaciones de inteligencia artificial son las que complementan la web 3.0, ahora la web nos ofrece soluciones en tiempo real y muchas veces se vuelve imperceptible el hecho que las soluciones son entregadas por “inteligencias prefabricadas”.

3. Ejecución del proyecto

3.1. Funciones del producto.

El producto es un portal web el cual debe tener un servicio de logueo y autenticación para los diferentes tipos de usuario los cuales son usuarios administradores, operadores y privilegiados, donde pueden ejecutar casos en la red neuronal, algunos usuarios tienen la posibilidad de crear nuevos usuarios con diferentes restricciones y asociarlos o no a una entidad.

Los usuarios pueden intentar ingresar casos en cualquier momento, mientras no supere la cantidad máxima de casos que el sistema pueda encolar, si en algún momento la cantidad de casos en cola supera el tope de lo que permite el sistema se le debe informar al usuario que no debe ingresar más casos hasta que el servidor esté desocupado y pueda recibir más casos.

Cuando el modelo termina de ejecutar un caso, el portal web deberá enviar un mensaje de correo electrónico a la persona que mando los datos y a los usuarios root que son los encargados de garantizar las disponibilidad, confiabilidad y seguridad del sistema.

Algunos usuarios pueden ver que usuarios existen, a qué entidad pertenece, sus casos ejecutados entre otros podrán también agregar/editar entidades y asignarles unas fechas disponibles para que las personas asociadas a dichas entidades puedan ejecutar casos.

3.2. La historia del portal web

La importancia de este desarrollo es crear un prototipo de interfaz web que le sirva a los distintos usuarios que por el contexto del negocio en el que se desenvuelven necesitan obtener simulaciones de un modelo de redes neuronales ya prefabricado en Matlab, esta interfaz ya tiene una versión en un sistema de gestión de contenido (CMS) llamado drupal

versión 6, este desarrollo fue hecho hace más de 8 años y drupal a lo largo de este tiempo como herramienta de software libre ha pasado por sus versiones 7,8 y se encuentra próximo al despliegue de su versión 9, hace un tiempo el soporte dado por la comunidad a drupal 6 ha acabado y actualizarlo es una labor tediosa.

El software culmina su vida útil debido a la falta de soporte y los avances tecnológicos, si no llevamos a cabo tareas repetitivas de mantenimiento podemos tener situaciones en las cuales nuestro software empieza a ser obsoleto, como el caso de la interfaz que antes comunicaba la red neuronal.

Se evaluó la posibilidad de rehacer el portal en la última versión de drupal, pero se llegó a la conclusión que sería más ágil y eficiente el uso de una herramienta en la que los miembros del equipo se desarrollaron mejor como lo es node js, asumiendo el riesgo de crear un propio prototipo de sistema con manejo de usuarios y toda la lógica para ejecutar casos.

3.3. Análisis y desarrollo

3.3.1. Identificación tipos de Usuario.

Se resaltan los siguientes tipos de usuario, todos estos usuarios pueden ejecutar casos.

3.3.1.1. Administradores

Son los usuarios que se encargados de dar mantenimiento, soporte y probar todas las funcionalidades del prototipo, tiene todas las funciones de un usuario privilegiado y un usuario operador, adicional se deja la brecha abierta en el prototipo para que al módulo de usuario administrador se le puedan agregar más funcionalidades que se crean pertinentes.

3.3.1.2. Privilegiados

Estos usuarios son ajenos al desarrollo, pueden ejecutar casos y por lo general ponen a prueba la aplicación, son aquellos encargados de ingresar usuarios operadores, empresas o entidades y dar tiempos de ejecución a esas empresas.

3.3.1.3. Operadores

Estos usuarios tienen pocos privilegios, pueden ejecutar y ver sus casos, ver su información de cuenta, cambiar su correo electrónico y contraseña.

3.3.2. Empresas operadoras.

Las empresas operadores o entidades pueden tener muchos usuarios “inscritos en ellas” y un usuario solo puede pertenecer a una entidad, estas entidades se componen por un nombre y siglas único, irrepetible por otras entidades, un valor de pérdidas técnicas para la ejecución de casos, y un rango de fechas que indica en qué periodo de tiempo los usuarios que pertenecen a esa entidad pueden hacer ejecuciones de la red neuronal.

3.3.3. Funcionamiento red neuronal.

Fue clave entender el funcionamiento de la red neuronal como un módulo externo el cual no podemos modificar ni volver a programar, cabe resaltar que este software también fue hecho hace más de ocho años, pero con el trabajo de ingeniería pudo ponerse a punto dentro del prototipo.

Esta red neuronal necesita un equipo donde se instale la máquina virtual de Matlab, esta máquina virtual será su ambiente de ejecución, además necesita unas plantillas de Excel ya prefabricadas las cuales son accedidas por la red neuronal para construir el resultado, la red

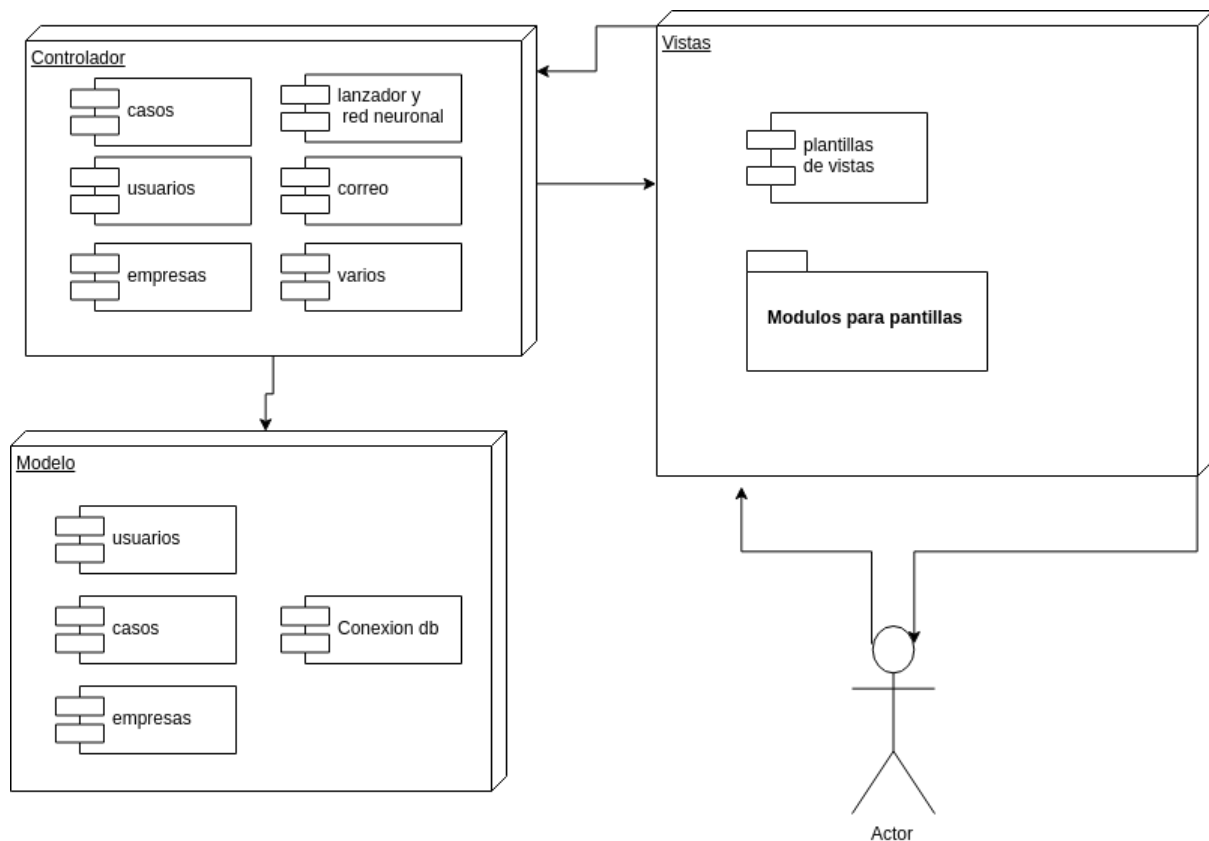
neuronal recibe un archivo de texto el cual debe ser creado por el prototipo con la ayuda de un formulario diligenciado por los distintos usuarios del aplicativo.

3.3.4. Formulario de simulación de casos.

El formulario de simulación de casos es importante, el esquema HTML fue tomado del antiguo portal y se le aplicó css por medio del framework bootstrap, también todas las validaciones que hacían por código a las variables fueron reunidas en un documento (ver anexo 2) y se codificaron en el formulario.

3.3.5. Modelo vista controlador

Este proyecto se desarrolló bajo el modelo, vista, controlador, el cual se puede resumir en la siguiente imagen.



El uso de este patrón de diseño es apropiado para la metodología, ya que podemos dividir muchos componentes, estos componentes pueden ser realizados por separado sin intervenir de manera directa en la construcción de otros componentes, pero siempre pensando en el objetivo de acoplarlos posteriormente.

3.4. Pruebas de software.

Se diseñó un manual de pruebas, que funciona como un checklist del prototipo, en este se analizan los módulos creados, aspectos de visualización, seguridad, rendimiento, compatibilidad, usabilidad y funcionalidad. (ver anexo 3).

3.5. Documentación

3.5.1. Mockups del prototipo

Los mockups son los diseños que se dieron a las vistas, estos mockups se desarrollan para tener una idea de cómo se debe crear la vista en cuestión, estos mockups son solo de documentación y no tienen ninguna funcionalidad. (ver anexo 4).

3.5.2. Diseño base de datos.

Para el prototipo se realizó el diagrama entidad-relación. (ver anexo 5) la tabla operador red representa las empresas, los tipos usuarios surge de normalizar las tablas para evitar redundancia, la tabla usuario almacena todos los tipos diferentes de usuario y la tabla caso almacena toda la información de los casos que se ejecuten por los diferentes usuarios.

3.5.3. Proceso ejecución de casos.

EL proceso para ejecutar un caso es el proceso crítico de todo el prototipo, ya que sin esto no tendría ningún sentido la realización del portal web, por este motivo se ha realizado un mapa al proceso, que comprende su activación hecha por el usuario mediante el envío del formulario de casos y los pasos que tiene este proceso por los diferentes actores del prototipo hasta el envío del correo que notifica el fin correcto o incorrecto de la ejecución.

3.5.4. Manual de instalación del entorno de desarrollo.

Ver anexo 6.

3.5.5. Manual de instalación de despliegue en intranet

Ver anexo 7.

4. Conclusiones y recomendaciones

4.1. Conclusiones

- El uso del board es un buen medio de control, esto sirve para ver qué tareas desarrollan los miembros del equipo y evaluar cómo distribuir las tareas entre estos.
- La planificación de un EDT sirve para definir rutas críticas y guía al equipo de trabajo para cumplir con los objetivos planteados.
- Aunque el board es una buena herramienta en pocas ocasiones se pasaba por alto el uso de esta, luego de que fue asimilada por el equipo de trabajo estos incidentes pasaron con menos frecuencia.

- El trabajo en equipo es primordial, es importante motivar al equipo, la comunicación es primordial en una metodología ágil.
- El uso de ramas de git permite que los desarrolladores trabajen en paralelo sin temor a dañar las cosas ya desarrolladas o que otros están desarrollando.

4.2. Recomendaciones

- Se recomienda una revisión exhaustiva a la red neuronal, si es posible volverla a producir con tecnologías a la vanguardia y replantear con el equipo de desarrollo del prototipo la forma de entregar las salidas.
- Se recomienda hacer un refactoring al código en las vistas, se puede reducir el número de líneas y de archivos
- Se recomienda crear un nuevo diseño para las vistas, estos diseños son tomados del portal antiguo.

4.3. Trabajos futuros.

Es bueno recalcar que se ha diseñado un prototipo funcional, pero aún se pueden realizar varios trabajos a este:

- Crear una vista para ver los datos de un caso específico, ya que existe las opciones de listarlos todos juntos o cargarlo, no existe una opción para listar un caso específico.
- Los módulos de usuario administrador y usuario privilegiado tienen las mismas funcionalidades, un usuario privilegiado no podría hacerle modificaciones al usuario administrador.

- Crear una funcionalidad de despachador de casos dentro del módulo usuario administrador.
- Realizar pruebas con owasp y openvas
- Hacer la configuración del dominio
- Hacer la configuración de los certificados digitales.
- Configurar opciones de eliminar usuarios y empresas teniendo en cuenta las restricciones de la base de datos.
- Rediseño de vistas.

5. Anexos

Todos los anexos se encuentran en la carpeta “**anexos**” que se encuentra en el directorio bajo los siguientes nombres.

- Anexo 1: edt.pdf
- Anexo 2: tablaCasos.pdf
- Anexo 3: pruebasSoftware.pdf
- Anexo 4: mockups.pdf
- Anexo 5: modeloEntidadRelacion.png
- Anexo 6: procesoCritico.png
- Anexo 7: EntornoDesarrollo.pdf
- Anexo 8: EntornoProduccion.pdf

6. Referencias bibliográficas

García-Peñalvo, F. J. (2018). Ingeniería del Software. Grupo GRIAL.

Pressman, R. S., & Troya, J. M. (1988). Ingeniería del software.

Cáceres, P., Marcos, E., & Kybele, G. (2001). Procesos ágiles para el desarrollo de aplicaciones Web. *Taller de Web Engineering de las Jornadas de Ingeniería del Software y Bases de Datos de, 2001.*

Martín, A. R., & Martín, M. J. R. (2014). *Aplicaciones web*. Ediciones Paraninfo, SA.

Takeyas, B. L. (2007). Introducción a la inteligencia artificial.

P. Bourque y R. E. Fairley Eds., "Guide to the Software Engineering Body of

Knowledge. Version 3.0. SWEBOK®." USA: IEEE, 2014. Disponible en:

<https://goo.gl/UphKi1>.