
Especificación de Requisitos de Software

para

Sistema Web para la Gestión de Accesos por COVID-19

Version 2.0

Presentado por: 1. Juan Manuel Cuellar Borrero
2. Nicolás Ibagón Rivera
3. Santiago Uribe Pastás

Presentado a: Gustavo Salazar Garzón
Lector

Octubre de 2020

Índice general

1	Introducción	3
1.1	Propósito	3
1.2	Alcance	3
1.3	Descripción general del producto	4
1.3.1	Perspectiva del producto	4
1.3.2	Características de usuario	4
1.3.3	Funcionalidades del producto	5
1.3.4	Limitaciones	7
1.4	Definiciones y Abreviaturas	7
2	Especificación de Requisitos	8
2.1	Documentación de Requisitos	8
2.2	Educción de Requisitos	8
2.3	Requisitos Funcionales	9
2.4	Requisitos No Funcionales	9
2.5	Priorización de Requisitos	10
2.6	Restricciones de Diseño	10
2.7	Cumplimiento de Estándares	11
3	Verificación	12
3.1	Criterios de validación	12
4	Apéndices	13
4.1	Identificación de los Stakeholders	13
4.2	Arquitectura del sistema	13
4.3	Casos de uso	14
4.4	Prototipos Visuales	14
4.5	Suposiciones y Dependencias	14

1 Introducción

El presente documento de especificación de requisitos de software brindará a los lectores una comprensión adecuada de las características más relevantes del sistema web para la gestión de accesos por COVID-19.

1.1. Propósito

El sistema web para la gestión de accesos por COVID-19 tiene como propósito dar un sistema confiable, eficaz y eficiente, para poder dar un manejo a las visitas por parte de civiles a los establecimientos públicos, de tal manera que se preserve la seguridad e integridad de todas la personas implicadas en dicho proceso de ingreso, permitiendo así un ambiente en el que las probabilidades de contagio del virus COVID-19 se vean drásticamente reducidas.

1.2. Alcance

El sistema web para la gestión de accesos por COVID-19 realizará un proceso de verificación de las condiciones de ingreso a establecimientos públicos, en dicho proceso se hará toma de datos necesarios como la temperatura del civil y el uso de tapabocas, y la validación de su estado de salud de acuerdo a los datos almacenados en la base de datos de las entidades de salud.

El software permitirá que la forma de ingresar a los establecimientos públicos durante esta coyuntura global sea mucho más eficiente, esto debido a que lo único que se necesitaría para ello, sería el estar registrado en el sistema. El objetivo principal del software, a nivel social, es poder ser una herramienta que no sólo ayude a preservar la seguridad biológica de cada persona, sino también, que el proceso de ingreso sea lo menos tedioso posible. Por otro lado a nivel de funcionamiento, el sistema pretende ser lo más rápido posible en cuanto a tiempos de respuesta, de igual forma pretende ser lo mas seguro en términos de disponibilidad, siendo capaz de soportar a por lo menos 5 mil (5000) de personas de forma concurrente.

El sistema en cuestión será desarrollado por un grupo de estudiantes de séptimo semestre de la materia *Procesos Y Diseño de Software* en la Pontificia Universidad Javeriana Cali, razón por la cual, si bien este no será liberado y utilizado a nivel nacional por un público común, sí se creará con la intención de satisfacer los requisitos propuestos, poniendo en práctica los temas vistos durante todo el semestre, en un desarrollo más

cercano a la realidad; esto intentado que el sistema mismo sea capaz de manejar un cantidad razonable de usuarios, todo esto tomando en cuenta limitaciones tales como el no uso de API's de pago o servidores de pago donde montar el sistema.

1.3. Descripción general del producto

1.3.1. Perspectiva del producto

El software será un producto completamente autónomo diseñado para trabajar en entornos web, el cual contará con el acceso a registros de entidades de salud locales, las cuales estén registradas en el sistema, para ayudar a determinar de manera mas efectiva y precisa cual es el veredicto de acceso de un civil a un entorno determinado (entorno publico).

1.3.2. Características de usuario

El sistema web para la gestión de accesos por COVID-19 contará con 4 usuarios (roles) que interactuarán con el sistema, como se muestra en la Figura 1.1.

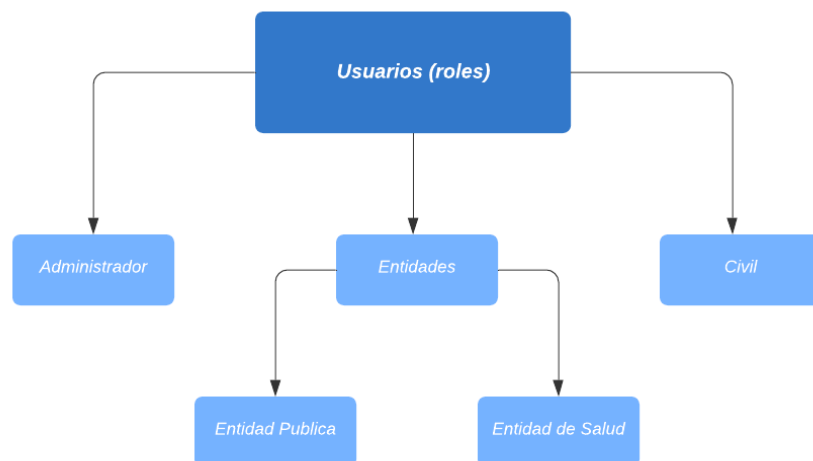


Figura 1.1: Usuarios que pueden acceder y hacer uso del sistema

A continuación se presentarán las características generales del perfil de los usuarios del producto. Cada usuario tendrá un perfil específico para que su interacción con el sistema sea correcto y no conlleve a fallos.

- **Administrador:** Persona que interactuará con el sistema continuamente. Se requiere que sea una persona con dominio en la utilización de las TIC y conocimiento técnico.

- Civil: Persona que interactuará con el sistema, su educación no puede ser definida, únicamente debe tener un conocimiento básico sobre navegación en paginas web.
- Entidad de Publica: Persona que interactuará continuamente con el sistema, su educación no puede ser definida y debe tener una capacitación con respecto al sistema.
- Entidad de Salud: Persona que interactuará continuamente con el sistema, su educación debe ser respecto a personal de salud y debe tener una capacitación con respecto al sistema.

1.3.3. Funcionalidades del producto

A continuación se presenta el resumen de las funcionalidades del sistema de acuerdo a los roles que influyen sobre este.

Funcionalidades administrador

- El administrador podrá iniciar sesión en el sistema.
- El administrador podrá crear nuevos usuarios de tipo administrador.
- El administrador podrá modificar los parámetros del sistema.
- El administrador podrá revisar los historiales de visitas de los civiles.
- El administrador podrá revisar los historiales de pruebas de las entidades de salud.
- El administrador podrá filtrar los historiales anteriormente mencionados
- El administrador podrá descargar los historiales anteriormente mencionados.
- El administrador podrá editar su perfil.
- El administrador podrá cerrar sesión.

Funcionalidades civil

- El civil podrá registrarse en el sistema.
- El civil podrá iniciar sesión en el sistema (una vez este registrado).
- El civil podrá generar su código QR.
- El civil podrá visualizar los motivos de la denegación de ingreso a los establecimientos.
- El civil podrá generar un historial de sus visitas a los establecimiento públicos.
- El civil podrá filtrar su historial de ingresos con respecto a fechas específicas.

- El civil podrá descargar su historial de ingresos.
- El civil podrá editar su perfil.
- El civil podrá cerrar sesión.

Funcionalidades entidad publica

- La entidad pública podrá registrarse en el sistema.
- La entidad pública podrá iniciar sesión en el sistema (una vez este registrada).
- La entidad pública podrá registrar el ingreso de un civil a su gestor de accesos.
- La entidad pública podrá registrar los resultados de las condiciones de ingreso de un civil.
- La entidad pública podrá generar un historial de los civiles que han ingresado en su establecimiento.
- La entidad pública podrá filtrar dicho historial por fechas.
- La entidad pública podrá descargar dicho historial.
- La entidad pública podrá editar su perfil.
- La entidad pública podrá cerrar sesión.

Funcionalidades entidad de salud

- La entidad de salud podrá registrarse en el sistema.
- La entidad de salud podrá iniciar sesión en el sistema (una vez este registrada).
- La entidad de salud podrá registrar cuando se le ha hecho una prueba de COVID-19 a un civil.
- La entidad de salud podrá registrar el resultado de una prueba de COVID-19 de un civil en el sistema.
- La entidad de salud podrá generar un historial de las pruebas COVID-19 que ha realizado a los civiles.
- La entidad de salud podrá filtrar su historial de pruebas con respecto a fechas específicas.
- La entidad de salud podrá descargar el historial de pruebas COVID-19.
- La entidad de salud podrá editar su perfil.
- La entidad de salud podrá cerrar sesión.

1.3.4. Limitaciones

A continuación se presentan las limitaciones que tendrá el sistema frente a diferentes tópicos.

- Limitaciones de hardware: Para esta aplicación el sistema será montado únicamente en el servidor de Amazon AWS.
- Interfaces con otras aplicaciones: Debido a que el sistema no interactúa con otros sistemas no se desarrollarán interfaces con otras aplicaciones.
- Políticas reguladoras: El sistema se desarrollará mediante software de licencia libre, por lo tanto, la utilización de estos programas se hará mediante las políticas establecidas por este tipo de licenciamiento.
- Funcionamiento paralelo: No aplica en el sistema.
- Idioma: Todo el material que se realice debe de estar en lenguaje español.
- Seguridad: Cada usuario deberá autenticarse y su acceso verificado. Todas las contraseñas deberán estar encriptadas en la base de datos para dar una buena seguridad al sistema y su información.

1.4. Definiciones y Abreviaturas

- AWS: Amazon Web Services.
- COVID-19: Enfermedad infecciosa causado por el virus SARS-CoV-2.
- CU: Caso de Uso
- RDE: Requisito de Desempeño.
- RDI: Requisito de Diseño.
- RF: Requisito Funcional
- RNF: Requisito No Funcional
- RS: Requisito de Sistema.
- RSG: Requisito de Seguridad.
- RU: Requisito de Usuario.
- TIC: Tecnologías de la Información y Comunicación.

2 Especificación de Requisitos

En esta sección se define la manera en que fueron educidos los requisitos del sistema, la manera en que estos fueron especificados, para que así el lector pueda identificar claramente lo que significa cada uno de los ítems de las especificaciones.

2.1. Documentación de Requisitos

La documentación de cada uno de los requisitos esta basada en los campos que se pueden observar en el Tabla 2.1.

ID	Tipo	Nombre	Estado	Responsable	Prioridad	Descripción
----	------	--------	--------	-------------	-----------	-------------

Tabla 2.1: Esquema de definición de requisitos

- ID: Identificación única asignada a cada requisito.
- Tipo: Clasificación del nivel de descripción del requisitos.
- Nombre: Identificación en lengua natural asignada a cada requisito.
- Estado: Clasificación del nivel de progreso del requisito, haciendo uso de código de colores. Se tienen los siguientes estados de requisitos. En revisión (naranja), no aprobado (rojo), aprobado (verde), en desarrollo (azul) y finalizado (morado).
- Responsable: Asignación de la persona encargada de que el sistema cumpla el requisito y su implementación.
- Prioridad: Nivel de importancia del requisito para el funcionamiento óptimo del sistema.
- Descripción: Explicación breve y en lengua natural de lo que debe cumplir el sistema.

2.2. Educción de Requisitos

Debido a que la educación de requisitos no es trivial, pues nunca se puede estar seguro de obtener todos los requisitos del usuario y del cliente con solo preguntarles qué debe hacer o no debe hacer el sistema, como técnicas de educación se utilizaron las siguientes:

- Entrevistas: Estas fueron llevadas a cabo los días lunes y miércoles a las 8:30 am con duración de 15 a 20 minutos con el cliente. Estas fueron grabadas para su posterior análisis.
- Encuestas: Se realizaron 2 encuestas que fueron enviadas por correo electrónico al cliente para que las respondiera.
- Lluvia de ideas: Fue realizada entre los integrantes del equipo para dar ideas de posibles preguntas para las encuestas. Se fueron anotando, posteriormente analizando y seleccionando los mejores.
- Cambio de perspectiva: Cada integrante del equipo hizo un cambio de perspectiva con los 4 roles que tiene el sistema, para así los otros 2 integrantes hacer una toma de requisitos.

2.3. Requisitos Funcionales

Los requisitos funcionales se encuentran en el anexo como un archivo excel bajo el nombre de *Requisitos.xlsx* en la hoja de cálculo 1 (R_Funcionales)

2.4. Requisitos No Funcionales

Los requisitos no funcionales se encuentran en el anexo como un archivo excel bajo el nombre de *Requisitos.xlsx* en la hoja de cálculo 2 (R_No_Funcionales)

2.5. Priorización de Requisitos

Como técnica de priorización de requisitos el equipo decidió usar la técnica MoSCoW [1]. A diferencia de las técnicas numéricas, esta técnica plantea una categorización en función de palabras que tengan un significado concreto, para ello esta técnica propone utilizar las siguientes etiquetas:

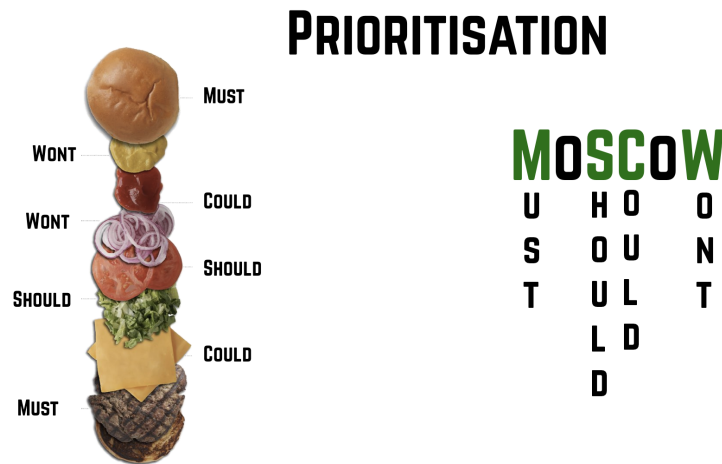


Figura 2.1: Técnica MoSCoW

Significado de las etiquetas de la Figura 2.1:

- **M:** Esta funcionalidad debe estar (MUST).
- **S:** Esta funcionalidad debería estar (SHOULD).
- **C:** Esta funcionalidad podría estar (COULD).
- **W:** Esta funcionalidad no estará por ahora, quizás en un futuro (WON'T).

La priorización de cada requisito se encuentra en la columna "Prioridad" en el archivo *Requisitos.xlsx*.

2.6. Restricciones de Diseño

- **Lenguajes de software :** Los lenguajes utilizados para la realización de este software serán Python, HTML/CSS, Javascript, y CQL (Cassandra Query Language).
- **Servidores:** Se debe contar con servidores estables que soporten una cantidad máxima de 5000 usuarios concurrentes en el sistema. El servidor a usar son los proporcionados por AWS.

- Sistema Operativo: Para una óptima utilización de software se puede utilizar los sistemas operativos Windows, MacOS, y Linux.
- Requisitos de desempeño: El sistema no deberá demorarse más de 30 segundos en renderizar el Historial de visitas en cada tipo de usuario, EL sistema no deberá demorarse más de 2 segundos en registrar e iniciar sesión en cualquier tipo de usuario.
- Para el uso del software se ve necesaria el uso de internet.

2.7. Cumplimiento de Estándares

Debido al almacenamiento, validación y manejo de datos personales en el software propuesto, este debe regirse bajo Ley 1581 de 2012 y el Decreto 1377 de 2013. Esta ley, conocida como Habeas Data, establece que es un derecho constitucional el que todas las personas puedan conocer, eliminar, actualizar y rectificar, cualquier clase de dato personal almacenado en bases de datos de entidades públicas o privadas. Además el artículo 15 de la Constitución Política establece que: Todas las personas tienen derecho a su intimidad personal y familiar y a su buen nombre, y el Estado debe respetarlos y hacerlos respetar.

Según el ministerio de educación Colombiano "En atención a la mencionada Ley se estableció el Registro Nacional de Bases de Datos - RNBD, el cual es un directorio público de las bases de datos con información personal sujetas a tratamiento que operan en el país, acorde con lo establecido en la Ley 1581 de 2012."

Esto implica que el software debe contar con un sistema de privacidad y seguridad de datos en el que, los datos personales manejados en el sistema sólo puedan ser vistos por aquella persona con la que están relacionados, de igual manera se plantea una clausula de confidencialidad para los administradores del sistema, en donde estos no tienen permitido el uso de los datos con fines maliciosos, de igual manera se les es prohibido la distribución de los mismos.

3 Verificación

En esta sección se define la forma y los criterios que se tendrán en cuenta para validar y/o verificar los requisitos, esto con el objetivo de encontrar errores en la especificación de requisitos. El archivo de validación de requisitos se encuentra en el anexo como *Validación Requisitos.xlsx*.

3.1. Criterios de validación

La técnica usada para la validación de requisitos es usar lista de chequeo, por ende, a continuación se presentan los criterios que se tendrán en cuenta para verificar los requisitos propuestos en el documento de requisitos (*Requisitos.xlsx*).

- Completo: El requisito debe contener en sí mismo toda la información necesaria, y no remitir a otras fuentes externas que le expliquen con más detalle.
- Consistente: Ningún requisito debe entrar en conflicto con otro requisito diferente, ni con parte de otro.
- Alcanzable: El requisito debe ser un objetivo realista, posible de ser alcanzado con el dinero, el tiempo y los recursos disponibles.
- No Ambiguo: La descripción del requisito debe ser clara, precisa y tener una única interpretación posible.
- Comprensible: Todo tipo de lectores (clientes, usuarios, desarrolladores, equipo de pruebas, etc.) pueden entender fácilmente el requisito.
- Verificable: Para el requisito existe un procedimiento de prueba finito y no costoso para demostrar que el sistema lo satisface.

4 Apéndices

4.1. Identificación de los Stakeholders

La identificación de los stakeholders se encuentra adjunta en el anexo como un archivo excel bajo el nombre *Registro de Interesados.xlsx*.

4.2. Arquitectura del sistema

El planteamiento de la estructura de software escogido se hizo principalmente pensando en un bajo acoplamiento teniendo una buena resistencia a fallas, esto debido a que al separar el sistema de la forma en que lo hicimos, logramos que cualquier clase de falla en alguno de los niveles planteados, sólo sea necesario modificar el nivel en cuestión. En la figura 4.1 se muestra el diagrama de la arquitectura del sistema.

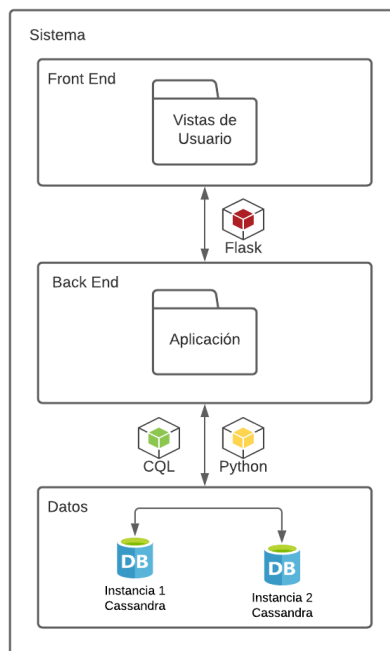


Figura 4.1: Arquitectura del sistema

4.3. Casos de uso

Los diagramas de caso de uso se encuentran adjuntos en el anexo como un archivo PDF con el nombre *Casos de Uso.pdf* junto con su respectiva documentación en un archivo PDF con el nombre *Documentación Casos de Uso.pdf*.

4.4. Prototipos Visuales

Los prototipos visuales de algunas de las interfaces el sistema se encuentran adjuntos en el anexo como un archivo PDF con el nombre *Prototipos Visuales.pdf*.

4.5. Suposiciones y Dependencias

Como primera suposición tenemos que los lectores de este documento presentan conocimientos técnicos frente a los temas tratados en el mismo. De igual manera ya que este sistema es desarrollado con fines académicos se hace la suposición de que no abra un despliegue a nivel nacional del sistema.

Bibliografía

- [1] C. Pérez, "Técnicas de priorización: el desafío de conseguir un orden para las funcionalidades", atsistemas.com. [Online]. Available: <https://www.atsistemas.com/en/blog/tcnicas-de-priorizacin-el-desafio-de-conseguir-un-orden-para-las-funcionalidades>.
- [2] "Protección de Datos Personales Ministerio de Educación Nacional de Colombia", mineducacion.gov.co. [Online]. Available: <https://bit.ly/35iLtXM>