

Universidad Nacional Pedro Henríquez Ureña
Facultad de Ciencias y Tecnología
Escuela de Informática

Modelado y Simulación de un Dispositivo Inteligente de Medición y
Prevención de Alergias con Matlab y Simulink



Trabajo de Grado presentado por

Jaison Alexander Cabral Calderón

para la obtención del grado:

Ingeniería en Sistemas Computacionales

Santo Domingo, D.N.

2021

Dedicatorias

En primer lugar, va dedicado a Dios por mantenerme firme, enfocado y brindarme las fuerzas necesarias para continuar en este camino de aprendizaje y poder lograr con tanto esfuerzo esta gran meta en mi vida.

A mi familia, especialmente a mis padres Carlos Cabral y Argentina Calderón y mi hermana Marina Cabral por su apoyo incondicional en todo este recorrido, gracias por estar ahí en los momentos más difíciles, en los retos que atravesaba y por siempre inspirarme a dar lo mejor de mí, espero estén orgullosos de mí. Este logro no es solo mío, también es de ustedes.

A mi novia Elizabeth Piña, quien ha sido participe en este largo camino y sabe que no ha sido nada fácil, por estar cuando más la necesitaba, por siempre creer en mí, sus palabras y su amor incondicional.

A mis primas Glennys Rosario Duval, Glennys Elisa, mi tío Carlos Duval y a Victor Batista que durante todo este proceso de una manera u otra me brindaron sus conocimientos y por enseñarme que por más largo que sea el camino al final tiene su recompensa.

A mis amigos, por creer en mí, por depositar su confianza en mí, por estar desde el principio conmigo de este gran camino recorrido, me siento muy afortunado y agradecido con Dios por tan excelentes amigos que pudo darme y la oportunidad de establecer tan buena amistad.

De igual forma este logro va dedicado a los maestros por transmitir sus conocimientos y guiarme por el camino correcto, siempre mantendré sus enseñanzas y lecciones que ustedes me brindaron.

Jaison A. Cabral Calderón

AGRADECIMIENTOS

Si bien es cierto, que todo sacrificio tiene su recompensa, es necesario destacar que nada de esto hubiese sido posible sin la ayuda de Dios y de mis padres, Carlos Cabral y Argentina Calderón, gracias por siempre estar ahí para mí, por los valores inculcados, por tan buena educación, por enseñarme que hay que sacrificarse para obtener lo que se quiere lograr y que nada que valga la pena en esta vida es fácil, gracias infinitas, esto es por y para ustedes.

Agradezco a mis amigos, con ustedes he compartido muchos de los mejores y peores momentos, ustedes son los que realmente conocen la cantidad de horas de estudio, sudor y lágrimas derramadas en toda esta travesía que hemos pasado. Gracias por siempre darme fuerzas y darme a entender que ustedes son importantes en todo este camino que hemos recorrido juntos.

Agradezco de manera especial a mi asesor José Ramón Romero. Excelente amigo, maestro y profesional, gracias por su dedicación y su esfuerzo empleado en guiarme por el camino correcto. Estoy muy agradecido con usted por su total honestidad y profesionalismo que ha demostrado en cada una de las actividades como tutor y también de siempre enseñar a sus alumnos el camino correcto.

RESUMEN

Por medio de las adversidades que personas con hipersensibilidades han estado padeciendo, se ha implementado simulaciones de un dispositivo capaz de prevenir y medir las reacciones (IgE alta) que luego de un tiempo se tornan visibles en distintas partes del cuerpo, el Smart Analyzer (Análisis Inteligente) surge como tema central de este trabajo de grado, con el objetivo de llevar esta tecnología a nuevas áreas de salud, mediante predicciones de su funcionamiento. Incentivando a estudiantes a aportar nuevas tecnologías para el cuidado de seres queridos y demás personas.

Tras poner en práctica los conocimientos de Anaconda Jupyter con el lenguaje de programación Python, Matlab y Simulink, este proyecto ha logrado crear una solución que permite ayudar a los médicos en gran manera utilizando inteligencia artificial (Machine Learning), de tal manera que el dispositivo analiza por si solo la inmunoglobina E del paciente, el Smart Analyzer es pequeño, es un brazalete el cual cualquier persona puede ajustárselo en su muñeca y el mismo analizará la IgE del individuo de forma automática.

Empleando estas simulaciones y viendo los resultados, se podrá comprobar que el uso de este nuevo dispositivo es vital para ciertas personas que tienen una hipersensibilidad alta.

ABSTRACT

Through the adversities that people with hypersensitivity have been suffering, simulations of a device capable of preventing and measuring reactions (high IgE) have been implemented that after a time become visible in different parts of the body, the Smart Analyzer (Analysis Intelligent) emerges as the central theme of this degree work, with the aim of taking this technology to new areas of health, through predictions of its operation. Encouraging students to contribute new technologies for the care of loved ones and other people.

After putting into practice the knowledge of Anaconda Jupyter with the Python, Matlab and Simulink programming language, this project has managed to create a solution that allows doctors to greatly help using artificial intelligence (Machine Learning), in such a way that the device It analyzes the immunoglobulin E of the patient by itself, the Smart Analyzer is small, it is a bracelet which anyone can adjust to their wrist and it will analyze the individual's IgE automatically.

Using these simulations and seeing the results, it can be seen that the use of this new device is vital for certain people who have high hypersensitivity.

TABLA DE FIGURAS

Ilustración 1 Modelo SIS	10
Ilustración 2 Modelo Tridimensional trasero del Smart Analyzer	20
Ilustración 3 Modelo Tridimensional Lateral del Smart Analyzer	20
Ilustración 4 Model Tridimensional Frontal del Smart Analyzer.....	21
Ilustración 5 Modelo físico del Smart Analyer ángulo 1	21
Ilustración 6 Modelo físico del dispositivo Smart Analyzer ángulo 2	22
Ilustración 7 Modelo físico del Smart Analyzer ángulo 3	22
Ilustración 8 Modelo del Dispositivo Smart Analyzer	23
Ilustración 9 Simulación Proceso Inyección de Dosis	26
Ilustración 10 Resultados del Proceso de Inyección de Dosis	27
Ilustración 11 Proceso de Entrenamiento con Machine Learning	28
Ilustración 12 Proceso de Entrenamiento con Machine Learning	29
Ilustración 13 Resultados del Entrenamiento	31
Ilustración 14 Importación de librerías y definición de variables	36
Ilustración 15 Definición de variables y creación de método	36
Ilustración 16 Definición de las variables para el tiempo de contacto de la alergia al cuerpo y el tiempo de recuperación	37
Ilustración 17 Función de Actualización	37
Ilustración 18 Método de ejecutar simulación	38
Ilustración 19 Método de ejecutar simulación utilizando TimeSeries para almacenar el estado del sistema en cada variable	38
Ilustración 20 Método para las gráficas de los resultados	39
Ilustración 21 DataFrame	40

Ilustración 22 Resultados en un DataFrame	40
Ilustración 23 Grafica de los resultados del DataFrame acerca del tiempo de contacto de la alergia vs el tiempo de recuperación.....	41
Ilustración 24 Ejemplo de Ataque de Anafilaxia y utilizando DataFrame	42
Ilustración 25 Grafica de los resultados de Ataque de Anafilaxia	42
Ilustración 26 Importación de librerías y definiendo métodos.....	43
Ilustración 27 Métricas cuantificativas para saber el número total de personas enfermas.....	44
Ilustración 28 Método para aplicar el Smart Analyzer	44
Ilustración 29 Resultados del uso del Smart Analyzer vs No uso del Smart Analyzer	45
Ilustración 30 Recorrido de eficiencia del uso del dispositivo	45
Ilustración 31 Organizando los resultados en una barrido "Sweep" o tabla	46
Ilustración 32 Grafica de impacto del uso del dispositivo Smart Analyzer	46

CONTENIDO

CAPÍTULO 1	Introducción.....	1
1.1.	Planteamiento del Problema	2
1.1.1.	Antecedentes.....	2
1.2.	Definición del Problema	3
1.3.	Objetivos	5
1.4.	Justificación.....	5
1.4.1.	Originalidad.....	5
1.4.2.	Profundidad	6
1.4.3.	Impacto	6
1.4.4.	Efecto a nivel nacional	7
1.4.5.	Alcances	7
1.5.	Limitaciones	7
CAPÍTULO 2	Marco teórico	9
2.1	Inteligencia Artificial (Machine Learning)	9
2.2	Modelado y Simulación	9
2.2.1	Modelo SIS (Susceptible-infectado-susceptible).....	10
2.2.2	Modelo matemático SIS.....	10
2.3	Dispositivos Inteligentes	11
2.4	Anaconda Jupyter	11
2.5	Alergias.....	11
2.6	Clasificación de alergias	12
2.6.1	Alergia alimentaria	13
2.6.2	Alergia a las picaduras de insectos.....	13
2.6.3	Rinitis.....	13

2.6.4	Alergia a fragancias	14
2.6.5	Alergia a la humedad	14
2.7	Impresora 3D	14
2.8	Matlab y Simulink	14
2.9	Sensor y equipamiento	15
CAPÍTULO 3	Marco metodológico	16
3.1	Metodología del proyecto de grado.....	16
3.2	Tipo de investigación.....	16
3.2.1	Investigación aplicada:	16
3.2.2	Investigación cuasi experimental	16
3.3	Técnicas de recolección de datos	17
3.3.1	Entrevistas.....	17
3.3.2	Observación	17
3.3.3	Hipótesis.....	17
CAPÍTULO 4	Resultados	18
4.1	Cumplimiento de objetivos	18
4.2	¿Por qué Anaconda Jupyter, Matlab y Simulink?	18
4.3	Ecuaciones utilizadas para seleccionar muestras aleatorias (Modelo Matemático)	19
4.4	Definición de las variables del modelo matemático	19
4.5	Modelo Digital del Smart Analyzer	20
4.5.1	Modelo físico de Smart Analyzer.....	21
4.5.2	Modelo del dispositivo Smart Analyzer	23
4.5.3	Tabla de comandos de Matlab y Simulink	23
4.5.4	Proceso de dosis Simulación con Matlab y Simulink	26

4.5.5	Resultados de la simulación con Matlab y Simulink	27
4.5.6	Proceso de entrenamiento con Inteligencia Artificial (Machine Learning) 28	
4.5.7	Resultados de la simulación del entrenamiento de la IgE con Inteligencia Artificial.....	31
4.6	Tabla de comandos de Anaconda Jupyter y su función	32
4.7	Proceso de Simulación con Anaconda Jupyter	36
CAPÍTULO 5	Conclusiones	47
CAPÍTULO 6	Recomendaciones	48
CAPÍTULO 7	Referencias.....	49
CAPÍTULO 8	ANEXOS	52
8.1	Preguntas de las entrevistas	52
8.2	Diagrama de flujo	53
8.3	Diagrama de Actividades.....	54
8.4	Casos de Uso	55
8.4.1	Caso de uso con alérgeno detectado	55

CAPÍTULO 1 INTRODUCCIÓN

Las enfermedades alérgicas constituyen un problema de salud pública a nivel global cuya incidencia, según las previsiones científicas más fiables, aumenta a mediano y largo plazo, es esperable, pues, un incremento de la demanda sanitaria por estos procesos, con el consiguiente impacto sobre la salud pública y los recursos sanitarios posibles.

El desconocimiento de las enfermedades alérgicas y la falta de un diagnóstico y un tratamiento adecuados conducen a una clara disminución de la calidad de vida relacionada con la salud, el aumento de las complicaciones y mortalidad, y un incremento considerable en los costes. Por todo ello, profundizar en su conocimiento y difundirlo son tareas que corresponden a las instituciones, públicas o privadas, conscientes de esta importante misión.

La alergia es una hipersensibilidad inmunológica adversa a sustancias extrañas inofensivas que puede dar lugar a varios tipos de enfermedades, en su mayoría proteínas, se denominan alérgenos y se pueden hallar en diversos tejidos, partículas, alimentos y organismos.

Los alérgenos pueden causar una reacción alérgica cuando caen sobre la piel o en un ojo, o cuando se inhalan, se ingieren o se inyectan. Se puede producir una reacción alérgica en diferentes circunstancias.

Las manifestaciones de las alergias varían desde síntomas relativamente leves, como estornudos hasta graves y potencialmente mortales como la anafilaxia.

1.1. Planteamiento del Problema

1.1.1. Antecedentes

El término alergia proviene en 1905, Pirquet sugiere que la enfermedad del suero, un fenómeno de hipersensibilidad tiene relación directa con la producción de anticuerpos contra antígenos en el suero inyectado, introduciendo el término “alergia” para referirse a “La reacción inmunológica alterada” en la cual suele presentarse una fuerte liberación de histaminas”. (Cardona, 2016)

Se presentó un estudio observacional en el que se determina la presencia de antecedentes alérgicos en madres, padres y hermanos de por lo menos 168 recién nacidos consecutivos durante el primer trimestre del año 2009 en el Hospital San Juan de Dios en Costa Rica, en los resultados se encontró que el 22% de las madres tienen antecedentes alérgicos, en tanto que en los hijos previos de estas madres la incidencia de alergia es de un 32% lo que esto significa un incremento del 10% de una generación a la siguiente. ((Rolando Celada Ramon, 2010)

En conclusión, a estos resultados se compararon estos datos de incidencia de los hijos de madres con antecedentes atópicos con la de los hijos de las madres sin alergia se comprobó que el riesgo relativo se duplica. En 56 niños (3,3%) hubo sospecha de reacción adversa, que se confirmó en 6 niños (0,36%) un cuadro de alergia a la proteína de la leche de vaca (APLV). El 83 % (5/6) de los niños con APLV tenían antecedentes familiares de primer grado de enfermedad. ((Celada Quezada, 2010)

También conocido como síndrome de alergia al polen y los alimentos, el síndrome de la alergia oral afecta a muchas personas que tienen rinitis alérgica. En esta afección, ciertas frutas, vegetales frescos o nueces y especias pueden desencadenar una reacción alérgica que causa cosquilleo o picazón en la boca. En casos graves, la reacción produce inflamación de la garganta. (MayoClinic, 1998 - 2020)

Desde hace tiempo se ha buscado la manera de herramientas de tecnología para la mejoría de la salud, en este renglón cabe mencionar **La bomba de insulina** creada y

desarrollada por la compañía irlandesa Medtronic en el 2017. (Alvarez, ComputerHoy, 2016)

Entre innovaciones destacadas sobre dispositivos capaces de inyección automática para insulina se encuentra **Enlite de Medtronic**, sistemas de monitorización continua propiamente dicha que dan una lectura directa a un transmisor en tiempo real y de 24 horas y permiten la existencia de alarmas de límite de glucosa alta o baja (entre otras). (Cardona, 2016)

Entre estos sistemas se encuentran el sensor **Enlite de Medtronic** para monitorizar en tiempo real. (Cardona, 2016)

Igual de importante se encuentra el sensor **Dexcom**, con su versión de lectura en monitor y en pantalla de bomba de insulina. Ambos sistemas requieren calibración mediante controles de glucemia capilar. (Cardona, 2016)

También esta **bomba flash** que es un medidor de glucosa en tiempo real. (Cardona, 2016)

1.2. Definición del Problema

Las enfermedades alérgicas van en aumento en todo el mundo tanto en los países desarrollados como en los países en desarrollo. Estas enfermedades incluyen asma, rinitis, anafilaxia, alergia a alimentos e insectos. Este aumento es específicamente en niños y adolescentes, en quienes se observa la mayor tendencia de aumento desarrollado. A pesar de este aumento con enfermedades alérgicas muy pocos países poseen servicios adecuados en este campo de la medicina. (Rubby Pawankar, 2001)

Casi no hay servicios especializados para personas alérgicas en muchos países, debido a que la prevalencia de alergias ha aumentado tanto, se debe considerar como un problema de salud. Según la Organización Mundial de la Salud (OMS), cientos de millones de sujetos en el mundo sufren de rinitis y se estima que 300 millones tienen asma siendo esto el 3.95% de la población a nivel mundial, también sujetos con alergias ocasionadas por alimentos y animales. (Canonica, 2001)

En este trabajo se explora una de las posibilidades de innovación para uso de la salud humana con un analizador inteligente, identificado como “Smart Analyzer” en este proyecto, tipo brazalete, sujetado en el brazo, específicamente en la muñeca.

Un dispositivo diseñado con la funcionalidad de prevenir y/o avisar reacciones alérgicas a las personas que tienen distintos tipos de hipersensibilidad, este dispositivo contiene un compartimiento para cápsulas y agujas que, al detectar la reacción interna de la alergia, le aplica la inyección en la muñeca, le indica y envía una notificación a su dispositivo advirtiéndole que existen anomalías alérgicas en su cuerpo ocasionando reacciones secundarias. Está dotado con un software que se mantiene activo en el dispositivo asegurando así la salud del individuo.

Al poseer compartimientos con pequeñas agujas y capsulas cuando se aplica la dosis regulada, Smart Analyzer cambia automáticamente a una aguja y capsula nueva para estar preparado para la siguiente reacción alérgica que pueda ocurrir.

La codificación de Smart Analyzer desarrolla el pensamiento crítico computacional que será de uso vital. En estos escenarios, la implementación de Smart Analyzer incorpora una práctica de la tecnología moderna enfocado a un entorno de salud, basado en aspectos, tales como:

- Aprender sobre ingeniería en aportar en el cuidado de la salud.
- Crear experiencias en la ingeniería y el mundo informático aportando nuevas facilidades médicas.
- Motivar a la innovación y creatividad de nuevos recursos tecnológicos.
- Mejorar el sistema de salud.
- Motivar a estudiantes de tecnología a ser parte de un gran cambio para el mundo aplicando sus talentos de programación para mejorar el sistema de salud.

Este dispositivo estará diseñado en Matlab y Simulink. Referente a esto los estudiantes tendrán acceso gratuito a este modelador y simulador de dispositivos para el desarrollo de nuevos programas/sistemas incentivándolos a crear nuevas aplicaciones o funcionalidades basadas en la salud o distintas ciencias.

Las instituciones de salud utilizan informáticos y/o poseen estudiantes de término del área de Ingeniería de Sistemas con el propósito de crear aplicaciones y funcionalidades

para el bienestar de las personas, esto incentiva al informático de manera intelectual a expandir su creatividad y desarrolla sus habilidades de ingeniería apoyado de su lógica de programación al momento de resolver problemas.

1.3. Objetivos

Objetivo general: Diseñar un dispositivo ciberfísico inteligente para la medición y prevención de alergias utilizando Matlab y Simulink.

Objetivos específicos:

1. Concebir, mediante la especificación de requerimientos, el dispositivo, incluyendo la definición de los tipos de alergias.
2. Modelar el dispositivo en Matlab y Simulink.
3. Simular el dispositivo.
4. Hacer una impresión 3D del dispositivo simulado.
5. Analizar los resultados, particularmente el algoritmo de medición y prevención.

1.4. Justificación

La investigación acerca de las alergias y sus reacciones secundarias (asfixia o falta de aire, garganta cerrada, sarpullido, hinchazón, etc.), permitirá determinar si los dispositivos inteligentes son vitales en la actualidad en procesos médicos tanto para analizar o prevenir reacciones alérgicas, por ejemplo, alergias a ciertos alimentos como a los mariscos o nueces. También, puede atribuirse a componentes presentes en la fórmula de fármacos frecuentemente recetados a los pacientes tras una consulta médica.

En la sección siguiente se definirán elementos que se tomaron en cuenta para el desarrollo del proyecto, entre estos están la originalidad, profundidad, efecto a nivel nacional, el impacto del proyecto y los alcances.

1.4.1. Originalidad

El modelo original en el cual está basado el dispositivo Smart Analyzer tiene la sistematización de medir los niveles de azúcar y el marcapasos para medir los latidos irregulares del corazón la cual esta predeterminado en un uso del cuidado de la salud.

El aporte original que brinda este proyecto, es brindar al individuo un dispositivo inteligente que pueda asegurar su salud y le notifique con antelación acerca de anomalías o reacción inmunológica alterada en su cuerpo y al mismo tiempo el dispositivo le pueda proporcionar el antialérgico que estará ubicado en una pequeña capsula integrado en el Smart Analyzer con agujas para la inserción del mismo contra la alergia de la cual padece, evitando así que los efectos secundarios de su hipersensibilidad reaccionen a un 100% prologando una resistencia hasta que reciba atención médica.

1.4.2. Profundidad

Este proyecto abarca varias etapas, las cuales son, el diseño y una simulación de como funcionaria el Smart Analyzer. El diseño incluye una pequeña impresión en 3D y el aparato simulado.

Este proyecto pretende motivar e incentivar lo importante que es aplicar la informática y ciertas técnicas que en ella están, como, por ejemplo, la programación, diseño y simulación de sistemas con el simple hecho de innovar parte del campo de salud.

1.4.3. Impacto

El dispositivo inteligente encargado de medición y prevención de alergias (Smart Analyzer) puede servir de referencia para otras personas que quieran aportar nuevos modelos y aparatos inteligentes para el uso del bienestar propio. Una vez terminado, el proyecto estará público en páginas Open Source, como GitHub y GitLab, también disponible para estudiantes con ideas para mejorar la calidad de salud de las personas creando más dispositivos útiles en las diferentes ramas del campo de la medicina, solo es necesario una impresora 3D para tener un modelo físico de cómo se vería el dispositivo, el uso de programas informáticos para diseñarlo y para su simulación.

El dispositivo inteligente está enfocado para tres tipos de usuarios:

- Estudiantes que aprendan como aplicar la lógica de programación, diseño y simulación de sistemas para crear innovaciones y tener una idea con la simulación de que tan fructífero podría ser su impacto en el país o en el mundo.

- Doctores con experiencia en el campo de medicina especializados a ayudar a los pacientes con hipersensibilidad a indicar el dispositivo para que sus pacientes estén preparados para cualquier reacción no esperada de la cual padece.
- Creadores e innovadores con un interés en la ingeniería de sistemas enfocándola al campo médico, ya sean profesionales o como un pasatiempo.

1.4.4. Efecto a nivel nacional

El efecto a nivel nacional trae consigo la posibilidad de introducir nuevas tecnologías al campo de medicina, entre los cuales se encuentran en estudios e investigaciones, esta tecnología en un dispositivo que pueda prevenir una situación no agradable para una familia por reacciones hipersensibles para compuestos en formulas en pastillas ayudara al sistema de salud a reducir casos críticos de personas en cuidados intensivos y reduciría una gran cantidad de accidentes causadas por ciertos alimentos que son hipersensibles algunas personas.

1.4.5. Alcances

El desarrollo de este proyecto está dirigido a las personas con hipersensibilidades con el propósito de brindarles mejor calidad de vida, protección y cuidado de su vida y salud.

Entre los alcances del proyecto están:

- Modelo 3D del dispositivo Smart Analyzer.
- Un reloj inteligente capaz de suministrar el antialérgico una vez detectada la reacción alérgica.
- Funciones de Matlab con la particularidad de mostrar el funcionamiento de Smart Analyzer.
- Solo se mostrará la simulación, no el funcionamiento implementado en el modelado 3D.
- Resultados de las simulaciones.

1.5. Limitaciones

El proyecto consta de ciertos factores que deben señalarse con el objetivo de que el usuario pueda utilizar el dispositivo y su función sin salirse de los márgenes establecidos.

Estos factores son necesarios para que, tanto la configuración establecida en el dispositivo, como la utilidad no entren en conflicto con el usuario.

- No es una aplicación móvil que se puede descargar de una tienda digital.
- El dispositivo Smart Analyzer no necesita configurarse más de una (1) vez.
- No es compatible con agua.
- Solo es compatible con la persona que lo utilizará.

CAPÍTULO 2 MARCO TEÓRICO

En esta sección se definen los conceptos que se utilizaron en la implementación para la creación de Smart Analyzer. A continuación, se tratan los términos de Inteligencia artificial (Machine Learning), modelo SIS, dispositivos inteligentes, Anaconda, Impresora 3D, Matlab y Simulink, sensor y equipamiento, alergias, clasificación de las alergias, modelado y simulación.

2.1 Inteligencia Artificial (Machine Learning)

La inteligencia artificial es el campo científico de la informática que se centra en la creación de programas y mecanismos que pueden mostrar comportamientos considerados inteligentes. En otras palabras, la IA es el concepto según el cual “las máquinas piensan como seres humanos”. (SalesForce, 2020)

Normalmente, un sistema de inteligencia artificial es capaz de analizar datos en grandes cantidades (big data), identificar patrones y tendencias y, por lo tanto, formular predicciones de forma automática, con rapidez y precisión. Para nosotros, lo importante es que la IA permite que nuestras experiencias cotidianas sean más inteligentes. El aprendizaje automático es la ciencia que se encarga de hacer que las computadoras realicen acciones sin necesidad de programación explícita. (SalesForce, 2020)

2.2 Modelado y Simulación

El modelado es la creación de una representación o imagen de un objeto real o imaginario. El modelado se refiere generalmente a la creación manual de una imagen tridimensional, o bien, una idea de un objeto. Un modelo es una representación parcial o simplificada de la realidad que recoge aquellos aspectos de relevancia para las intenciones del modelador, ya que pretende extraer conclusiones de tipo predictivo, se modela para comprender o explicar mejor un proceso o unas observaciones. (EcuRed, n.d.)

La simulación es un acto consiste en imitar o fingir que se está realizando una acción cuando en realidad no se está llevando a cabo. La simulación es aplicada en campos de la investigación como la química, física y estudios comparativos de elementos de la naturaleza que se requiere evaluar el comportamiento. (ConceptoDefinicion, s.f.)

2.2.1 Modelo SIS (Susceptible-infectado-susceptible)

Los modelos SIS (Susceptible-infectado-susceptible) son modelos epidemiológicos de compartimientos, que se definen a partir de clases y subclases. (al., 2015)

Los modelos SIS se usan para enfermedades en las que no hay inmunidad, pues, una vez que las personas infectadas se recuperan, pasan a ser de nuevo susceptibles. Por lo tanto, la progresión de la enfermedad desde el punto de vista de un individuo es susceptible-infectado-susceptible. (al., 2015)

2.2.2 Modelo matemático SIS

En este modelo se estudia la población total N dividida en dos grupos, susceptibles e infectados, que evolucionan con el tiempo t . las variaciones entre ellos son consecuencia del contagio por parte de individuos infectados a individuos susceptibles, e individuos infectados que se recuperan y vuelven a ser susceptibles. El contagio se produce a través de una **tasa de contagio $B > 0$** mientras que la recuperación tiene lugar a través de una **tasa de recuperación $Y > 0$** que solo depende de la población de individuos que haya en cada momento.

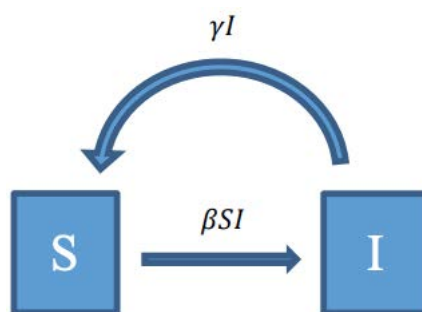


Ilustración 1 Modelo SIS

2.3 Dispositivos Inteligentes

Son aparatos de pequeño tamaño, con algunas capacidades de procesamiento, con conexión permanente o intermitente a una red, con memoria limitada diseñados específicamente para una función, pero pueden llevar a cabo otras funciones más generales. (Farfán, 2013)

Un dispositivo inteligente es un dispositivo electrónico, generalmente conectado a otros dispositivos o redes de diferentes protocolos inalámbricos como el Bluetooth y wifi. Los dispositivos inteligentes pueden diseñarse para admitir muchos factores de forma diferente, una variedad de propiedades relacionadas con la informática ubicua y para ser utilizado en tres entornos del sistema principal: mundo físico, entornos centrados en el ser humano y entornos informáticos distribuidos. (Moko Smart, 2020)

El término también puede referirse a un dispositivo que exhibe algunas propiedades de la computación ubicua, incluyendo, aunque no necesariamente, inteligencia artificial.

2.4 Anaconda Jupyter

El proyecto Jupyter es una organización sin ánimo de lucro creada para “desarrollar software de código abierto, estándares abiertos y servicios para computación interactiva en docenas de lenguajes de programación”. El nombre del proyecto jupyter es una referencia a los tres lenguajes de programación principales soportados por Jupyter, que son Julia, Python y R. (project, n.d.)

Anaconda es una distribución libre y abierta de los lenguajes Python y R, utilizada en ciencia de datos, y aprendizaje automático (machine learning). Esto incluye grandes volúmenes de información, análisis predictivo y cómputos científicos. Está orientado a simplificar el despliegue y administración de los paquetes de software. (Anaconda, n.d.)

2.5 Alergias

Las alergias aparecen cuando el sistema inmunitario reacciona ante una sustancia extraña como el polen, el veneno de abejas o la caspa de las mascotas, y también a un

alimento que no provoca una reacción en la mayoría de las personas. (MayoClinic, 1998 - 2020)

Los síntomas de la alergia dependen de la sustancia involucrada. Una alergia por picadura de insecto puede causar anafilaxia, tos, opresión en el pecho, falta de aire o una zona de gran hinchazón en el lugar de la picadura. (MayoClinic, 1998 - 2020)

Una alergia causada por un medicamento puede provocar picazón en la piel, erupción cutánea, hinchazón en la cara y anafilaxia. (MayoClinic, 1998 - 2020)

Está presente la rinitis alérgica que puede provocar estornudos, picazón en la nariz, ojos irritados o hinchados (conjuntivitis), picazón en el paladar o nariz congestionada. (MayoClinic, 1998 - 2020)

Este dispositivo estará basado en las alergias causadas por alimentos. La alergia a las comidas es una respuesta anormal a un alimento desencadenada por el sistema inmunitario del cuerpo. Los síntomas de una alergia a un alimento incluyen picazón o hinchazón de la boca, vómito, cólicos abdominales y dolor, sarpullido, sentir la garganta apretada y dificultad para respirar, también se puede incluir la disminución de la presión sanguínea. (MedlinePlus)

2.6 Clasificación de alergias

El organismo humano está <<condenado>> a vivir en un entorno más o menos hostil, en permanente contacto con elementos extraños. Para defenderse de las agresiones externas y evitar que se pongan en peligro la supervivencia humana, la naturaleza le dotó un sistema defensivo: el sistema inmunitario. No obstante, en ocasiones este mecanismo se descontrola y reacciona ante sustancias mediante respuestas anómalas de una intensidad desmesurada, a esto se le denomina hipersensibilidad o alergia (por estar mediadas por anticuerpos del tipo IgE o relacionados) y que de hecho los alergólogos tratan con mayor frecuencia (aunque no de forma exclusiva), son las siguientes. (Bonet, s.f.)

2.6.1 Alergia alimentaria

La transición epidemiológica ha determinado un aumento en la aparición y diagnóstico de las alergias alimentarias en general. El término reacción alimentaria adversa se refiere a cualquier síntoma presente en forma constante y repetida después de la ingestión de un determinado alimento, reacción que puede deberse a una alergia o intolerancia alimentaria. En la alergia alimentaria, las manifestaciones clínicas observadas son consecuencia de una respuesta **inmunológica alterada**. (Scielo, n.d.)

Los tipos de alergia se clasifican para denotar causa, severidad, administración y prevención posibles.

2.6.2 Alergia a las picaduras de insectos

La hipersensibilidad a veneno o saliva de insectos mediada por un mecanismo inmunitario se denomina alergia a veneno o saliva, como en el caso de la alergia al veneno de abeja. Se puede denominar alergia a veneno mediada por la IgE. La gran cantidad de alérgeno de veneno existente en una sola picadura se puede comparar con años de inhalación del polen. El sistema inmunitario reacciona por todos los medios para hacer frente al invasor. Esto provoca una reacción alérgica de una forma desproporcionada a las proteínas que contiene el veneno de insecto, y cuando sufre una picadura de insecto su cuerpo interpreta que esas proteínas son invasoras y trata de eliminarlas. (Hirsch, s.f.)

2.6.3 Rinitis

Se denomina rinitis alérgica cuando los síntomas típicos como congestión nasal, rinorrea, estornudos o prurito nasal son el resultado de una reacción de hipersensibilidad (en la mayoría, de tipo IgE). Se caracteriza por la inflamación de la mucosa nasal, como por ejemplo la rinitis alérgica inducida por el polen. La presencia de rinitis eleva al triple el riesgo de padecer de asma un paciente, esto afecta a un 40% a los niños. (Sáez, 2020)

2.6.4 Alergia a fragancias

Las sensibilidades a los perfumes o en fragancias generales accionan diversas reacciones desagradables en gente. Algunos síntomas comunes de la alergia del perfume incluyen suave dolor de cabeza severo, irritación, picazón y erupciones.

2.6.5 Alergia a la humedad

La alergia a la humedad es un subtipo de alergia respiratoria superior e inferior producida por la inhalación de esporas de hongos presentes en el aire y que necesitan altos niveles de humedad para subsistir y multiplicarse. Todas las personas están expuestas al moho, pero en algunas se produce una respuesta desproporcionada de su sistema inmunológico, expresándose en síntomas respiratorios. (SAVIA, 2019)

Al inhalar las esporas del moho que se encuentran en el aire, el sistema inmunológico del cuerpo las reconoce como no propias, por lo que se generan anticuerpos para combatirlos y estos crean memoria para los agentes invasores, de modo que cualquier contacto posterior desencadena una nueva reacción mediada por histamina, que produce los síntomas típicos de la alergia. (SAVIA, 2019)

2.7 Impresora 3D

Las impresoras 3D permiten elaborar objetos a partir de un modelo digital, en su mayoría, a través de procesos de fabricación aditiva, donde el material se añade capa por capa hasta conformar la pieza final. Esta tecnología está siendo ampliamente abordada en diversos ámbitos, incluido el sanitario, donde se destaca el desarrollo de implantes e incluso la impresión de tejidos u órganos. (Marson, 2016)

2.8 Matlab y Simulink

Matlab es un programa interactivo orientado para llevar a cabo proyectos en donde se encuentren implicados cálculos matemáticos y la visualización de estos.

Es una herramienta basada en matrices para cálculos científicos y de ingeniería. Desde el punto de vista del control, Matlab se puede considerar como un entorno matemático de simulación que puede utilizarse para modelar y analizar sistemas. Permite el estudio

de sistemas continuos, discretos, lineales y no lineales, mediante descripción interna y externa, en el dominio temporal y de frecuencia. Un factor muy importante de Matlab es que es una herramienta usada para el análisis, investigación y desarrollo de nuevos productos tecnológicos. (MATLAB, s.f.)

Simulink es un entorno de programación especial de Matlab que sirve para simular el comportamiento de los sistemas dinámicos. Puede simular sistemas lineales y no lineales, modelos en tiempo continuo y tiempo discreto y sistemas híbridos de todos los anteriores. Es un entorno gráfico en el cual el modelo a simular se constituye clicando y arrastrando los diferentes bloques que lo constituyen. (ETSETB, s.f.)

2.9 Sensor y equipamiento

Biosensor: este sensor consta de un receptor biológico con un anticuerpo que detecta el alérgeno que causa la hipersensibilidad, un transductor que convierte esa reacción en una señal cuantificable, y un instrumento electrónico que la procesará y mostrará en el sistema. (Corso, 2015)

Este sensor, por ejemplo, es capaz de detectar entre 0 y 10 partes por millón de caseína en la leche, principal alérgeno alimentario. “Esa precisión es importante, porque algunas personas son alérgicas incluso a los utensilios sucios de alimentos.”, según detalla (Bozzola, 2015)

Se mantiene en constante análisis de la inmunoglobina E, programado con estándares de medidas de la IgE (Inmunoglobina E) dependiendo de la persona, niño o adolescente. Teniendo los intervalos de IgE correspondiente a las edades será más fácil de suministrar la dosis adecuada.

Mini agujas ADVOCATE Pen: estas agujas son de calibre ultrafino, con la menor molestia posible. Cuentan con una aguja extra lubricada que proporciona menos fricción y una penetración suave. Las agujas ADVOCATE son compatibles con todos los dispositivos de inyección. (ADVOCATE, n.d.)

Acompañado del biosensor y las mini agujas advocate, estará incluida las capsulas que estarán cargadas con el antialérgico correspondiente.

CAPÍTULO 3 MARCO METODOLÓGICO

3.1 Metodología del proyecto de grado

En el siguiente capítulo se presentan los lineamientos metodológicos relacionados así como también la técnica de recolección de datos, el cual presenta la guía para obtener la información necesaria, para el tratamiento de la información y procedimientos para el análisis, los métodos de investigación son cuasi experimental y aplicada debido que a partir de lo que ya se ha descrito y explicado se centra en predecir de lo que va a pasar en el futuro haciendo énfasis en los datos que fueron analizados para el desarrollo de esta investigación.

3.2 Tipo de investigación

A continuación, se describen los métodos utilizados.

3.2.1 Investigación aplicada:

El objetivo de esta investigación es encontrar estrategias que puedan ser empleadas en el abordaje de un problema en específico. La investigación aplicada se nutre de la teoría de generar conocimiento práctico, y su uso es muy común en ramas del conocimiento como la ingeniería y medicina.

La investigación aplicada tiene fines predictivos. A través de este tipo de investigación se pueden medir ciertas variables para pronosticar comportamientos con un impacto en la vida cotidiana.

3.2.2 Investigación cuasi experimental

El objetivo de esta investigación es controlar algunas variables del fenómeno a estudiar, por ello no llega ser totalmente experimental. En este caso, podría ponerse en peligro la salud de un paciente para este tipo de situación ya que el alérgeno que podría tener no es del todo controlable.

3.3 Técnicas de recolección de datos

La recolección de los datos a utilizar es de la siguiente manera:

3.3.1 Entrevistas

Para la recopilación de los datos del dispositivo Smart Analyzer se realizaron entrevistas al personal especializado en alergias, o mejor conocidos como alergólogos.

Los datos obtenidos fueron a través de una serie de preguntas abiertas y cerradas, el método de la entrevista seleccionado se basó en forma libre. Las entrevistas fueron llevadas a cabo por separado, por consiguiente, obtuve respuestas muy similares en cuanto el tema de las hipersensibilidades.

3.3.2 Observación

El proceso de observación consistió en tomar como paciente a mi padre, el cual recientemente le ha surgido una alergia desconocida (No era alérgico a nada, pero aún se desconoce esa hipersensibilidad) junto con uno de los doctores entrevistados el cual fue el Dr. Hibarioné Alcántara para tratar el caso, por lo que actualmente no existe una cura total para las hipersensibilidades ni un diagnóstico exacto, en esta etapa, a través de la observación directa se determinó que sus niveles de inmunoglobina E (IgE) estaban en aumento y comenzaron a mostrarse en su cuerpo los resultados de esta reacción.

3.3.3 Hipótesis

Personas desde su nacimiento padecen de alérgenos, otras personas pueden adquirirlas sin ningún motivo o causa probable, pero esto se debe a la inmunoglobulina E que cada individuo tiene en su cuerpo, sin embargo, el porcentaje de hipersensibles van en aumento cada día, este aumento podría estar relacionado por varios factores a lo largo del tiempo, por ejemplo, picaduras de insectos desde temprana edad y/o constante polvo en su entorno.

El contacto de distintos alérgenos ocasiona reacciones inmunológicamente alteradas las cuales producen irritaciones en la piel o dificultad para respirar.

CAPÍTULO 4 RESULTADOS

4.1 Cumplimiento de objetivos

Durante el desarrollo de este trabajo de grado se alcanzaron los objetivos específicos establecidos anteriormente.

- Se diseñó el modelado 3D.
- La simulación se realizó.
- Se definieron los tipos de alergias.
- Se realizó la impresión 3D del dispositivo Smart Analyzer.
- Se analizó el funcionamiento del dispositivo.

4.2 ¿Por qué Anaconda Jupyter, Matlab y Simulink?

Anaconda es un entorno de programación abierto basado en tres lenguajes principales, los cuales son:

- Julia.
- Python.
- R.

Este es utilizado para la ciencia de datos y aprendizaje automático (Machine Learning). Esto incluye grandes procesamientos de grandes volúmenes de datos, análisis predictivo y de cómputos científicos. Esta distribución de datos es utilizada por más de 6 millones de usuarios e incluye más de 250 paquetes de ciencias de datos.

Matlab es un sistema basado en matrices que ofrece su propio entorno de desarrollo integrado con una gran variedad de funciones, ha sido seleccionado para este proyecto debido a que este sistema es el indicado para modelar y simular dispositivos y/o funcionamientos. Matlab cuenta con su propio lenguaje de programación (Lenguaje M) utilizado para coordinar y conectar los datos para poder analizar y extraer la información más detallada y predecir procesos a futuro.

Simulink es un entorno de programación de más alto nivel de abstracción que el lenguaje interpretado Matlab.

4.3 Ecuaciones utilizadas para seleccionar muestras aleatorias (Modelo Matemático)

El modelo SIS puede ser formulado como un sistema de dos ecuaciones diferenciales:

$$dS/dt = -\beta(t)I(t) + \gamma I(t), \quad S(0) = S_0 > 0$$

Ecuación 1

$$dI/dt = \beta(t)I(t) - \gamma I(t), \quad I(0) = I_0 > 0.$$

Ecuación 2

4.4 Definición de las variables del modelo matemático

γ	Tasa de recuperación.
β	Tasa de contagio (probabilidad por individuo y unidad de tiempo de contraer la enfermedad).
dt/ds	Diferencial de t respecto a s .
t	tiempo
S	Número de individuos susceptibles
I	Número de individuos infectados.
N	Número de individuos totales.

4.5 Modelo Digital del Smart Analyzer



Ilustración 2 Modelo Tridimensional trasero del Smart Analyzer

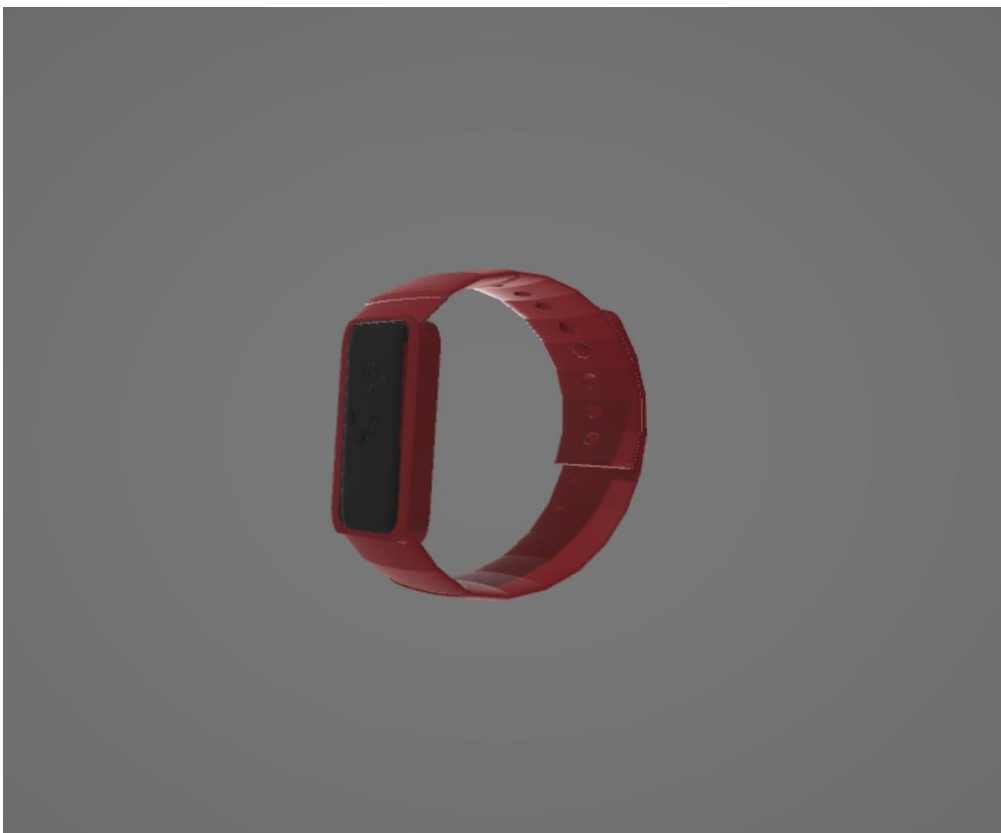


Ilustración 3 Modelo Tridimensional Lateral del Smart Analyzer



Ilustración 4 Model Tridimensional Frontal del Smart Analyzer

4.5.1 Modelo físico de Smart Analyzer



Ilustración 5 Modelo físico del Smart Analyzer ángulo 1



Ilustración 6 Modelo físico del dispositivo Smart Analyzer ángulo 2



Ilustración 7 Modelo físico del Smart Analyzer ángulo 3

4.5.2 Modelo del dispositivo Smart Analyzer

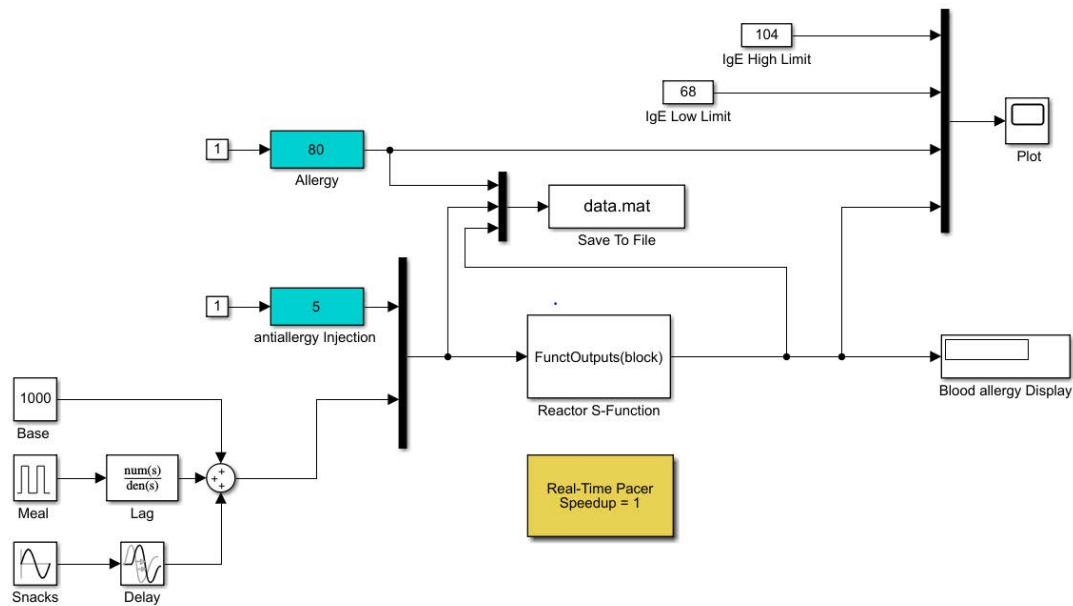


Ilustración 8 Modelo del Dispositivo Smart Analyzer

4.5.3 Tabla de comandos de Matlab y Simulink

Comando	Función
dataURL	Variable donde se almacenará una url de Matlab para descargar archivos de prueba.
datasetfolder	Carpeta de datos.
If - end	Condicional si – fin. Para toma de decisiones.
Websave(zipfile)	Método que guarda un archivo

sds	(Signal Data Store) señal de almacenamiento.
data	Variable.
signalMask	Mascara de la señal.
Plotsigroi(param1, param2)	Graficar en Matlab.
rng	Rango.
trainIdx	Entrenamiento por el índice.
trainDs	Entrenamiento por índice y señal de almacenamiento.
Type getMask.m	Se obtiene la máscara para saber su entrenamiento.
Preview()	Muestra cómo se vería el entrenamiento.
actionFlag	Funciona para entrenar una red o descargarla nuevamente como lo utiliza dataURL.
load	Cargar.
Option = trainingOptions	En option se almacena el entrenamiento con las iteraciones.
layers	Capas.

maxEpochs	Máximas recorridas.
miniBatchSize	Tamaño del lote mínimo
initialLearnRate	Tasa de aprendizaje inicial.
LearnRateDropPeriod	Aprendizaje del periodo de caída de la tasa.
LearnRateSchedule, piecewise	Aprendizaje tasa de calendario por partes
gradientThreshold	Umbral de gradiente.
Training-progress	Progreso de entrenamiento.
Shuffle, every-epoch	Saltar cada época al terminar una época.
dispatchInBackground	Despacho en segundo plano.
trainData	Datos de entrenamiento.
tallTrainSet	Alto conjunto de entrenamientos.
tallTestSet	Alto conjunto de entrenamiento de prueba.
testData	Datos de prueba.
rawNet	Líneas de red.
trainNetwork	Red entrenada

4.5.4 Proceso de dosis Simulación con Matlab y Simulink

En este proceso, el código hace una simulación del comportamiento del alérgeno y luego ya suministrada la dosis.

```
1 %Se cargan los datos a utilizar
2 load Allergy.mat
3
4 time = Allergy(1,:);
5 alergia_sp = Allergy(2,:);
6 dosis = Allergy(3,:);
7 meals = Allergy(4,:);
8 alergia = Allergy(5,:);
9
10 figure(1)
11 hold off
12
13 %Grafica 1
14 subplot(2,1,1)
15 hold off
16 plot(time,dosis,'b-','LineWidth',2)
17 axis([min(time) max(time) 1 5]);
18 legend('IgE Injection')
19 ylabel('dosis (\muU/min)')
20 xlabel('Suministrando dosis')
21
22
23 subplot(2,1,2)
24 hold off
25 plot(time,alergia,'b:','LineWidth',2)
26 legend('Blood IgE')
27 axis([min(time) max(time) ...
28 min(min(alergia),min(alergia_sp))-10 ...
29 max(max(alergia),max(alergia_sp))+10]);
30 ylabel('Allergy (mg/dl)')
31 xlabel('Time (min)')
32
33
34 %Grafica 3
35
36 figure(2)
37 subplot(3,1,1)
38 hold off
39 plot(time,alergia_sp,'b-','LineWidth',2)
40 legend('Sin IgE alta')
41 ylabel('allergy without IgE (\muU/min)')
42 xlabel('min')
43 % save data to text file
44 data = alergia';
45
```

Ilustración 9 Simulación Proceso Inyección de Dosis

4.5.5 Resultados de la simulación con Matlab y Simulink

En la primera gráfica (la que está detrás) se muestra el resultado una vez la dosis del antialérgico ha sido inyectado, mientras que la segunda gráfica (la que está delante) muestra el comportamiento de la alergia contra la dosis.

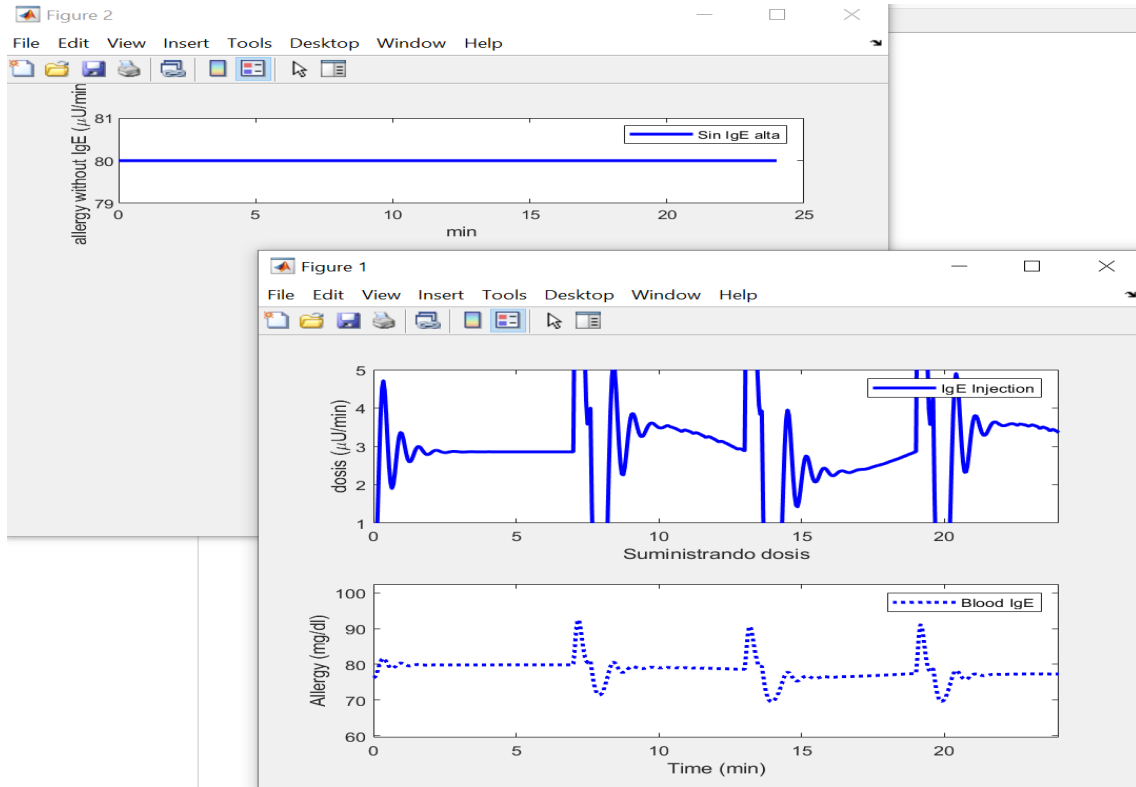


Ilustración 10 Resultados del Proceso de Inyección de Dosis

4.5.6 Proceso de entrenamiento con Inteligencia Artificial (Machine Learning)

En esta captura de pantalla definimos las variables, cargamos los datos y esa información la almacenamos (ya descomprimido) en un directorio temporal para luego iniciar el proceso de análisis inteligente.

```
1  dataURL = 'https://www.mathworks.com/supportfiles/SPT/data/QTDatabaseECGData1.zip'
2  datasetFolder = fullfile(tempdir,'QTDataset');
3  zipFile = fullfile(tempdir,'QTDatabaseECGData.zip');
4  if ~exist(datasetFolder,'dir')
5      websave(zipFile,dataURL);
6      unzip(zipFile,tempdir);
7  end

8
9
10 data = preview(sds)

11 M = signalMask(data{2}); % Asignar la mascara de senal a la variable M
12 plotsigroi(M,data{1}(1:1000)) % Mostrar el grafico.

13 rng default
14 [trainIdx,~,testIdx] = dividerand(numel(sds.Files),0.7,0,0.3);
15 trainDs = subset(sds,trainIdx);
16 testDs = subset(sds,testIdx);

17 type getmask.m % type es para saber que tipo es ese metodo o para saber que sabe.

18 trainDs = transform(trainDs, @getmask);
19 testDs = transform(testDs, @getmask);
20
21 transformedData = preview(trainDs)

22 plot(transformedData{2}(1:1000))
```

Ilustración 11 Proceso de Entrenamiento con Machine Learning

Condicionamos la red de entrenamiento de los datos cargados anteriormente, esto es para evitar conflictos en la primera fase de los datos de entrenamiento, valida si hay que cargar nuevamente los datos, sino solo carga los datos que están en el directorio.

```
22     plot(transformedData{2}(1:1000))

23     trainDs = transform(trainDs,@resizeData);
24     testDs = transform(testDs,@resizeData);
25     preview(trainDs)
26

27     actionFlag = ;
28     if actionFlag == "Download networks"
29         % Download the pre-trained networks
30         dataURL = 'https://ssd.mathworks.com/supportfiles/SPT/data/QTDatabaseECGSegmen';
31         modelsFolder = fullfile(tempdir,'QTDatabaseECGSegmentationNetworks');
32         modelsFile = fullfile(modelsFolder,'trainedNetworks.mat');
33         zipFile = fullfile(tempdir,'QTDatabaseECGSegmentationNetworks.zip');
34         if ~exist(modelsFolder,'dir')
35             websave(zipFile,dataURL);
36             unzip(zipFile,fullfile(tempdir,'QTDatabaseECGSegmentationNetworks'));
37         end
38         load(modelsFile)
39     end

40     layers = [ ...
41         sequenceInputLayer(1)
42         lstmLayer(200,'OutputMode','sequence')
43         fullyConnectedLayer(4)
44         softmaxLayer
45         classificationLayer];
```

Ilustración 12 Proceso de Entrenamiento con Machine Learning

En esta captura de pantalla se definen las recorridas del entrenamiento que le dará a la red neuronal y almacena en las variable los resultados que al final se van visualizando mediante va ejecutando el entrenamiento.

```
40 layers = [ ...
41     sequenceInputLayer(1)
42     lstmLayer(200,'OutputMode','sequence')
43     fullyConnectedLayer(4)
44     softmaxLayer
45     classificationLayer];

46 options = trainingOptions('adam', ...
47     'MaxEpochs',1, ...
48     'MiniBatchSize',50, ...
49     'InitialLearnRate',0.01, ...
50     'LearnRateDropPeriod',3, ...
51     'LearnRateSchedule','piecewise', ...
52     'GradientThreshold',1, ...
53     'Plots','training-progress',...
54     'shuffle','every-epoch',...
55     'Verbose',0,...
56     'DispatchInBackground',true);

57 tallTrainSet = tall(trainDs);
58 tallTestSet = tall(testDs);

59 trainData = gather(tallTrainSet);
60 trainData(1,:)
61 testData = gather(tallTestSet);

62 if actionFlag == "Train networks"
63     rawNet = trainNetwork(trainData(:,1), ...
64         trainData(:,2), ...
65         layers,options);
66 end
67
68
```

Ilustración 10 Proceso de Entrenamiento con Machine Learning

4.5.7 Resultados de la simulación del entrenamiento de la IGE con Inteligencia Artificial

Visualizamos los resultados de la red neuronal, la frecuencia azul indica lo que aprendió la red y el naranja son los datos que pudo completar totalmente, por eso se visualiza en descenso y el azul en ascenso.

A la derecha se muestran los resultados del entrenamiento y las leyendas.

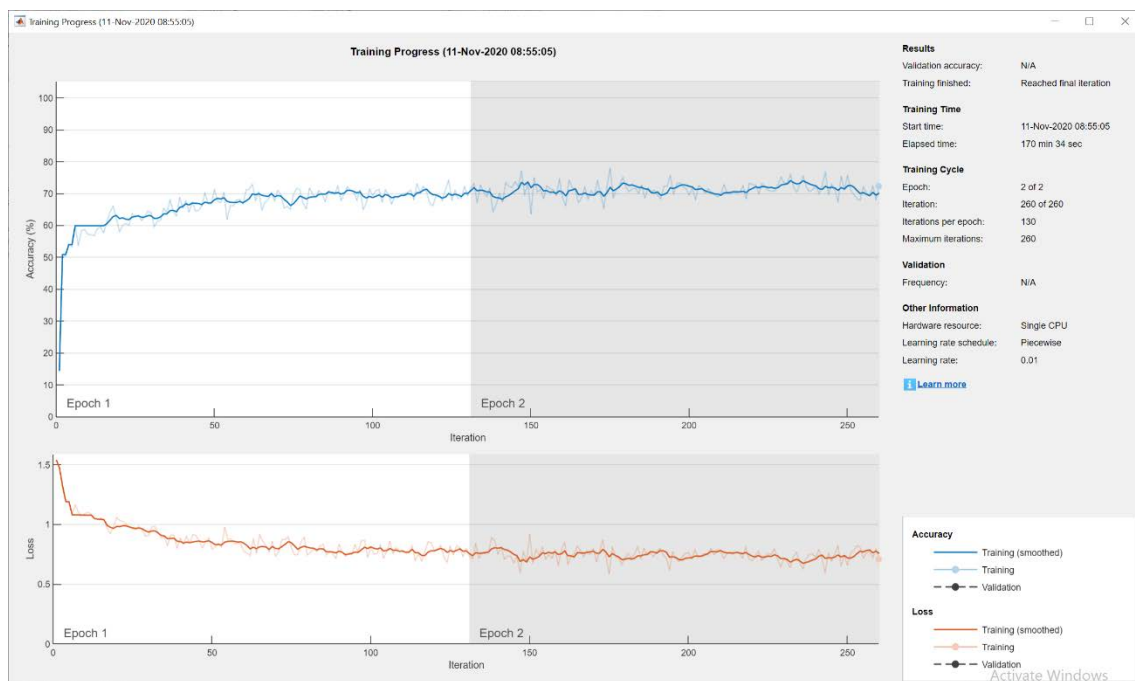


Ilustración 13 Resultados del Entrenamiento

4.6 Tabla de comandos de Anaconda Jupyter y su función

Comandos	Aplicación
<code>%matplotlib inline</code>	Importando módulo de gráficas.
<code>%config InteractiveShell.ast_node_interactivity='last_expr_or_assign'</code>	Configuración para mostrar los valores asignados.
<code>Import modsim</code>	Importando funciones de un módulo de python
<code>From matplotlib import PercentFormatter</code>	Importando gráficos de barras para representar resultados.
<code>Init = State(S,I,S)</code>	Definimos una variable para almacenar los resultados de un objeto de "State" estado para representar el número o fracción de

	personas en cada comportamiento.
<code>init /= sum(init)</code>	Dividimos el resultado que se almacenó en la variable <code>init</code> y se suma los datos fraccionarios.
<code>def make_system(beta, gamma)</code>	Método que crea un objeto con los parámetros a recibir y retorna el objeto.

<code>tc</code>	Tiempo de contacto
<code>tr</code>	Tiempo de recuperación
<code>beta</code>	Duración de contacto por días
<code>gamma</code>	Tiempo de recuperación por días
<code>System</code>	Variable que almacena un objeto.
<code>def update_func(state, t, system)</code>	Método que actualiza el estado durante el paso de tiempo actual y devuelve el estado durante el siguiente paso de tiempo.

state	Estado con las variables S, I, S
t	Paso de tiempo

State = update_func(init, 0, system)	Ejecutamos un solo paso de tiempo.
def run_simulation(system, update_func)	Ejecuta una simulación de un sistema.
T0	Tiempo de inicio.
T_end	Tiempo final.
infected	infectados
Recovered	Recuperados
TimeSeries()	Time Series representa la evolución en el tiempo de una población y se utiliza para identificar, modelar y pronosticar patrones y comportamientos en los datos que se muestran en intervalos de tiempo discretos.
Linrange(param1, param2, param3)	
Linesapce(param1, param2, param3)	Rango de tiempo, inicio, número de pasos, pasos a tomar.
Dataframe	Se almacenan los para visualizar mejor los datos.

<code>Results.head()</code>	Muestra los datos en una tabla de filas y columnas
<code>results</code>	Almacena los resultados de la ejecución del método de <code>run(simulation(system, update_func))</code> .
<code>def calc_toal_infected(results)</code>	Fracción de infectados durante la simulación.
<code>def compute_factor(spending)</code>	Método de que calcula un factor lógico.
<code>logistica (spending, m, K, B)</code>	Método que recibe parámetros de variación para determinar un valor lógico.
<code>def add_device(system, fraction)</code>	Método de inmunización de las personas sin el dispositivo.
<code>plot</code>	Visualizar valores en una gráfica.
<code>decorate</code>	Marca leyendas en una gráfica.
<code>Inmunize_array</code>	Arreglo de datos de inmunización.
<code>Infected_sweep</code>	Variable que almacena un barrido de inmunización.
<code>For variable in arreglo_de_datos:</code>	Ciclo for para recorrer arreglos de datos.

4.7 Proceso de Simulación con Anaconda Jupyter

Este proceso de simulación es estático, con datos fijos. Primero importamos las librerías a utilizar y definimos las variables con sus respectivos valores y el modelo a utilizar, el modelo es "SIS", se repite de esta manera "SISS" porque en Python no se puede definir la misma variable con distinto valor. En los valores del modelo "SIS" es la muestra de personas a utilizar. S = susceptibles, I = Inmunes, SS = susceptibles.

Modeling Python

```
In [1]: # Configure Jupyter so figures appear in the notebook
%matplotlib inline

# Configure Jupyter to display the assigned value after an assignment
%config InteractiveShell.ast_node_interactivity='last_expr_or_assign'

# import functions from the modsim.py module
from modsim import *

from matplotlib.ticker import PercentFormatter
```

SIS Implementacion

Vamos a usar un estado un objeto "State" a representar el numero (o fraccion) de personas en cada comportamiento. We'll use a `State` object to represent the number (or fraction) of people in each compartment.

SIS = Susceptible - Inmune - Susceptible

```
In [2]: init = State(S=105, I=25, SS=80)
```

Out[2]:

	values
S	105
I	25
SS	80

Ilustración 14 Importación de librerías y definición de variables

En esta captura convertimos los datos del modelo SIS a términos decimales para mejor comprensión y creamos un método que crea un objeto con los parámetros que reciba.

Para convertir de numero de personas a fracciones, dividimos por el total.

```
In [3]: init /= sum(init)
```

Out[3]:

	values
S	0.500000
I	0.119048
SS	0.380952

make_system crea un objeto System con los parámetros dados.

```
In [4]: def make_system(beta, gamma):
        """Make a system object for the SIS model.

        beta: contact rate in hours
        gamma: recovery rate in hours

        returns: System object
        """
        init = State(S=105, I=25, SS=80)
        init /= sum(init)

        t0 = 1
        t_end = 7 * 24

        return System(init=init, t0=t0, t_end=t_end,
                    beta=beta, gamma=gamma)
```

Ilustración 15 Definición de variables y creación de método

Aquí definimos las variables para el tiempo de contacto de la alergia y el tiempo de recuperación, luego se convierte a decimal en “beta y gamma” para que todos los datos estén en el mismo sistema numérico.

Aquí se definen las variables para el tiempo de contacto de la alergia al cuerpo y el tiempo de recuperación.

```
In [5]: tc = 0.5      # time between contacts in hours by allergy
        tr = 2       # recovery time in hours of the allergy

        beta = 1 / tc # contact rate in per day
        gamma = 1 / tr # recovery rate in per day

        system = make_system(beta, gamma)
```

Out[5]:

	values
init	S 0.500000 I 0.119048 SS 0.380952 d...
t0	1
t_end	168
beta	2
gamma	0.5

Ilustración 16 Definición de las variables para el tiempo de contacto de la alergia al cuerpo y el tiempo de recuperación

Aquí se crea una función que va actualizando la información de infectados y recuperado a medida que se van introduciendo los datos, y luego devuelve un estado actualizado del modelo respecto a los datos recibidos devolviendo un estado para las variables del modelo.

La función de actualización toma el estado durante el paso de tiempo actual y devuelve el estado durante el siguiente paso de tiempo.

```
In [6]: def update_func(state, t, system):
        """Update the SIS model.

        state: State with variables S, I, S
        t: time step
        system: System with beta and gamma

        returns: State object
        """
        s, i, ss = state

        infected = system.beta * i * s
        recovered = system.gamma * i

        s -= infected
        i += infected - recovered
        ss += recovered

        return State(S=s, I=i, SS=ss)
```

Para ejecutar un solo paso de tiempo, lo llamamos así:

```
In [7]: state = update_func(init, 0, system)
```

Out[7]:

	values
S	0.380952
I	0.178571
SS	0.440476

Ilustración 17 Función de Actualización

En este paso se define una función de “run_simulation” para ejecutar la simulación con los datos almacenados del modelo actualizado obtenido del método “update_func”. Para cada paso en el tiempo de la iteración del bucle for() que se va almacenando en una variable que se le está asignando al método de actualización.

Ahora podemos ejecutar una simulación llamando a la función de actualización para cada paso de tiempo.

```
In [8]: def run_simulation(system, update_func):
        """Runs a simulation of the system.

        system: System object
        update_func: function that updates state

        returns: State object for final state
        """
        state = system.init

        for t in linrange(system.t0, system.t_end):
            state = update_func(state, t, system)

        return state
```

El resultado es el estado del sistema en t_end

```
In [9]: run_simulation(system, update_func)
```

Out[9]:

	values
S	2.756464e-02
I	5.071920e-43
SS	9.724354e-01

Ilustración 18 Método de ejecutar simulación

TimeSeries es una sucesión de datos medidos en determinados momentos y ordenados con un orden cronológico, por lo tanto, lo que hace es que se define el método utilizado de “run_simulation”, pero en este caso como un TimeSeries para que luego se visualicen mejor los datos.

Usando objetos TimeSeries

Si queremos almacenar el estado del sistema en cada paso de tiempo, podemos usar un objeto `TimeSeries` para cada variable de estado.

```
In [10]: def run_simulation(system, update_func):
        """Runs a simulation of the system.

        Add three Series objects to the System: S, I, S

        system: System object
        update_func: function that updates state
        """
        S = TimeSeries()
        I = TimeSeries()
        SS = TimeSeries()

        state = system.init
        t0 = system.t0
        S[t0], I[t0], SS[t0] = state

        for t in linrange(system.t0, system.t_end):
            state = update_func(state, t, system)
            S[t+1], I[t+1], SS[t+1] = state

        return S, I, SS
```

Ilustración 19 Método de ejecutar simulación utilizando TimeSeries para almacenar el estado del sistema en cada variable

Entonces aquí definimos los datos a utilizar con el TimeSeries para luego definir un método que nos mostrará los resultados en una gráfica.

```

Así es como lo llamamos

In [11]: tc = 0.5 # time between contacts in hours
         tr = 2 # recovery time in hours

         beta = 1 / tc # contact rate in per hours
         gamma = 1 / tr # recovery rate in per hours

         system = make_system(beta, gamma)
         S, I, SS = run_simulation(system, update_func)

Y luego podemos graficar los resultados.

In [12]: def plot_results(S, I, SS):
         """Plot the results of a SIS model.

         S: TimeSeries
         I: TimeSeries
         S: TimeSeries
         """
         plot(S, '--', label='Susceptible')
         plot(I, '-', label='Allergy')
         plot(SS, ':', label='Recovered')
         decorate(xlabel='Time (days)',
                  ylabel='Fraction of recovery')

```

Ilustración 20 Método para las gráficas de los resultados

Así visualizamos los datos. Determinando en el eje horizontal las horas de contacto de la alergia ilustrado con las barras azules y en el eje vertical las horas de recuperación ilustrado con la línea naranja.



Ilustración 18 Gráfica de tiempo de contacto vs tiempo de recuperación de la alergia.

Un data frame es una estructura de datos en forma de tabla de doble entrada ordenados en filas y columnas. Definimos el método nuevamente de ejecutar la simulación, pero utilizando esta estructura y que el método devuelva un cuadro de datos.

Usando un DataFrame

En lugar de hacer tres objetos `TimeSeries`, podemos usar un `DataFrame` para mejor visualización.

Tenemos que usar `fila` para seleccionar filas, en lugar de columnas. Pero entonces Pandas hace lo correcto, haciendo coincidir las variables de estado con las columnas del `DataFrame`.

```
In [16]: def run_simulation(system, update_func):
         """Runs a simulation of the system.

         system: System object
         update_func: function that updates state

         returns: TimeFrame
         """
         frame = TimeFrame(columns=system.init.index)
         frame.row[system.t0] = system.init

         for t in linrange(system.t0, system.t_end):
             frame.row[t+1] = update_func(frame.row[t], t, system)

         return frame
```

Ilustración 21 DataFrame

Definimos las variables a utilizar (recordando que como este tipo de simulación es estática, los datos deben estar cargados en memoria previamente).

Se visualiza el dataFrame con los datos.

Así es como lo ejecutamos y cómo se ve el resultado.

```
In [17]: tc = 0.5      # time between contacts in hours by allergy
         tr = 2        # recovery time in hours of the allergy

         beta = 1 / tc  # contact rate in per hours
         gamma = 1 / tr # recovery rate in per hours

         system = make_system(beta, gamma)
         results = run_simulation(system, update_func)
         results.head()
```

```
Out[17]:
```

	s	I	SS
1	0.500000	0.119048	0.380952
2	0.380952	0.178571	0.440476
3	0.244898	0.225340	0.529762
4	0.134527	0.223041	0.642432
5	0.074517	0.171531	0.753952

Ilustración 22 Resultados en un DataFrame

Entonces así se ilustran esos datos obtenidos. En el eje horizontal el tiempo de contacto de la alergia en horas ilustrado con las barras azules y en el eje vertical ilustrado con el color naranja es el tiempo de recuperación.

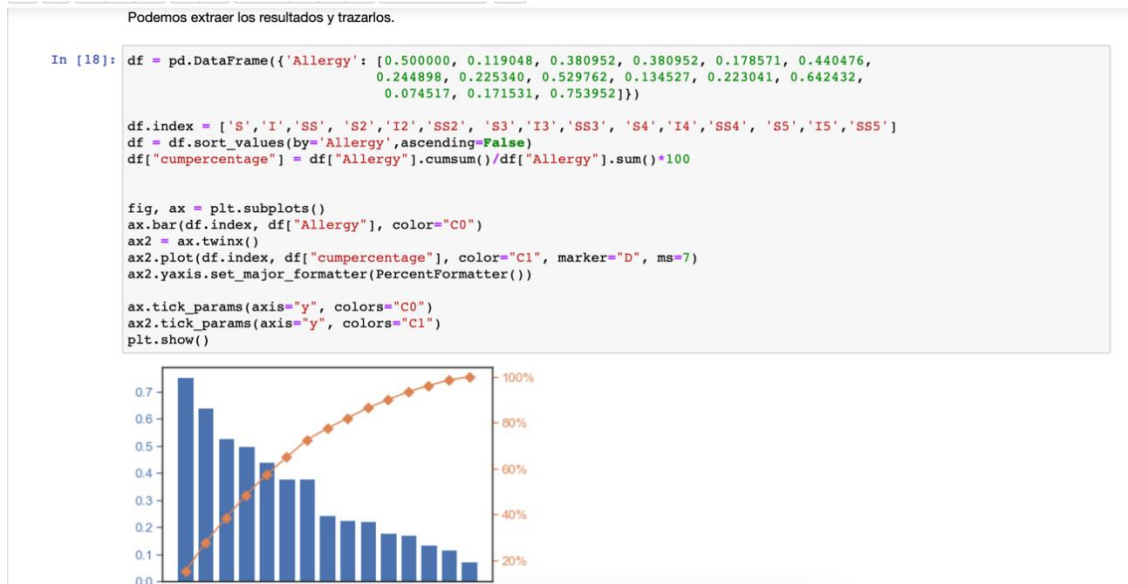


Ilustración 23 Grafica de los resultados del DataFrame acerca del tiempo de contacto de la alergia vs el tiempo de recuperación.

Un ataque de anafilaxia es una reacción alérgica severa-grave en todo el cuerpo ante un alérgeno. Suponiendo aquí que tenemos un ataque de anafilaxia definimos los datos de la simulación a utilizar, con un tiempo de contacto en 1 hora y un tiempo de recuperación en 7 días y esto luego lo mostramos en un dataframe.

Ejemplo de ataque de anafilaxia

Suponga que el tiempo entre contactos es de 1 o 2 horas y el tiempo de recuperación es de 5 días.

Y tiene un ataque de "Anafilaxia"

```
In [19]: t_Contact = 1
t_recovery = 7

beta = 1/t_Contact
gamma = 1/t_recovery

system = make_system(beta, gamma)
results = run_simulation(system, update_func)
results.head()
```

Out[19]:

	S	I	SS
1	0.500000	0.119048	0.380952
2	0.440476	0.161565	0.397959
3	0.369311	0.209649	0.421040
4	0.291885	0.257125	0.450990
5	0.216834	0.295444	0.487722

Ilustración 24 Ejemplo de Ataque de Anafilaxia y utilizando DataFrame

Mostramos los resultados en una gráfica y podemos notar que el ataque de anafilaxia es más severo que una reacción alérgica simple.

```
In [22]: df = pd.DataFrame({'Allergy': [0.500000, 0.119048, 0.380952, 0.440476, 0.161565, 0.397959,
0.369311, 0.209649, 0.421040, 0.291885, 0.257125, 0.450990,
0.216834, 0.295444, 0.487722]})
df.index = ['S', 'I', 'SS', 'S2', 'I2', 'SS2', 'S3', 'I3', 'SS3', 'S4', 'I4', 'SS4', 'S5', 'I5', 'SS5']
df = df.sort_values(by='Allergy', ascending=False)
df["cumpercentage"] = df["Allergy"].cumsum()/df["Allergy"].sum()*100

fig, ax = plt.subplots()
ax.bar(df.index, df["Allergy"], color="C0")
ax2 = ax.twinx()
ax2.plot(df.index, df["cumpercentage"], color="C1", marker="D", ms=7)
ax2.yaxis.set_major_formatter(PercentFormatter())

ax.tick_params(axis="y", colors="C0")
ax2.tick_params(axis="y", colors="C1")
plt.show()
```

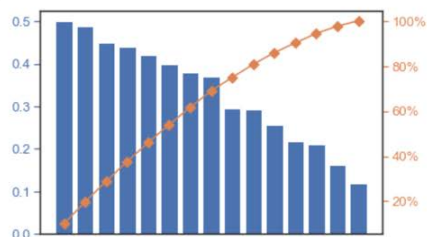


Ilustración 25 Grafica de los resultados de Ataque de Anafilaxia

En esta captura definimos nuevamente los métodos y librerías utilizados para simular el ataque alérgico contra el dispositivo Smart Analyzer.

Modeling and Simulation in Python

```
In [1]: # Configure Jupyter so figures appear in the notebook
%matplotlib inline

# Configure Jupyter to display the assigned value after an assignment
%config InteractiveShell.ast_node_interactivity='last_expr_or_assign'

# import functions from the modsim.py module
from modsim import *
```

Code

Continuación del modelo. Definimos nuevamente el método que crea el objeto `System` con los parámetros dados.

```
In [2]: def make_system(beta, gamma):
        """Make a system object for the SIS model.

        returns: System object
        """
        init = State(S=89, I=1, SS=0)
        init /= sum(init)

        t0 = 0
        t_end = 7 * 24

        return System(init=init, t0=t0, t_end=t_end,
                      beta=beta, gamma=gamma)
```

```
In [3]: def update_func(state, t, system):
        """Update the SIS model.

        state: State with variables S, I, SS
        t: time step
        system: System with beta and gamma

        returns: State object
        """
        s, i, ss = state

        infected = system.beta * i * s
        recovered = system.gamma * i

        s -= infected
        i += infected - recovered
        ss += recovered

        return State(S=s, I=i, SS=ss)
```

```
In [4]: def run_simulation(system, update_func):
        """Runs a simulation of the system.

        system: System object
        update_func: function that updates state

        returns: TimeFrame
        """
        frame = TimeFrame(columns=system.init.index)
        frame.row[system.t0] = system.init

        for t in linrange(system.t0, system.t_end):
            frame.row[t+1] = update_func(frame.row[t], t, system)

        return frame
```

Ilustración 26 Importación de librerías y definiendo métodos.

Aquí definimos métodos que permiten calcular el número total de personas enfermas o que padecen de una hipersensibilidad.

```
Metrics

Dados los resultados, podemos calcular métricas que cuantifiquen lo que nos interesa, como el número total de personas enfermas, por ejemplo.

In [5]: def calc_total_infected(results):
        """Fraction of infected during the simulation.

        results: DataFrame with columns S, I, Ss

        returns: fraction of sick
        """
        return get_first_value(results.S) - get_last_value(results.S)

Aqui en ejemplo previo

In [6]: beta = 0.5
        gamma = 0.25
        system = make_system(beta, gamma)

        results = run_simulation(system, update_func)
        print(beta, gamma, calc_total_infected(results))

0.5 0.25 0.804506446944698

In [7]: def compute_factor(spending):
        return logistica(spending, m=400, K = 0.5, B=0.3)
```

Ilustración 27 Métricas cuantificativas para saber el número total de personas enfermas.

Este método se basa en el uso del dispositivo Smart Analyzer, su funcionamiento es medir y prevenir la reacción alérgica. Vemos que hay un 80% de probabilidad de que las personas hipersensibles sean muy susceptibles a enfermarse constantemente.

```
Podemos utilizar este modelo para evaluar escenarios de "qué pasaría si". Por ejemplo, esta función modela el efecto de la inmunización moviendo una fracción de la población de S a Ss antes de que comience la simulación.

In [8]: def add_device(system, fraction):
        """Immunize a fraction of the people without the device.
        SISS = SIS

        Moves the given fraction from S to Ss.

        system: System object
        fraction: number from 0 to 1
        """
        system.init.S -= fraction
        system.init.SS += fraction

Ejecutando el modelo sin el dispositivo.

In [9]: results = run_simulation(system, update_func)
        calc_total_infected(results)

Out[9]: 0.804506446944698
```

Ilustración 28 Método para aplicar el Smart Analyzer

Entonces si ese 80% utiliza el dispositivo, un 66% pueden tener mejor calidad de vida y estar con menos preocupaciones por su condición hipersensible.

Debajo se puede visualizar el “No uso del dispositivo” y la reducción de los casos de alergia y/o de anafilaxia.

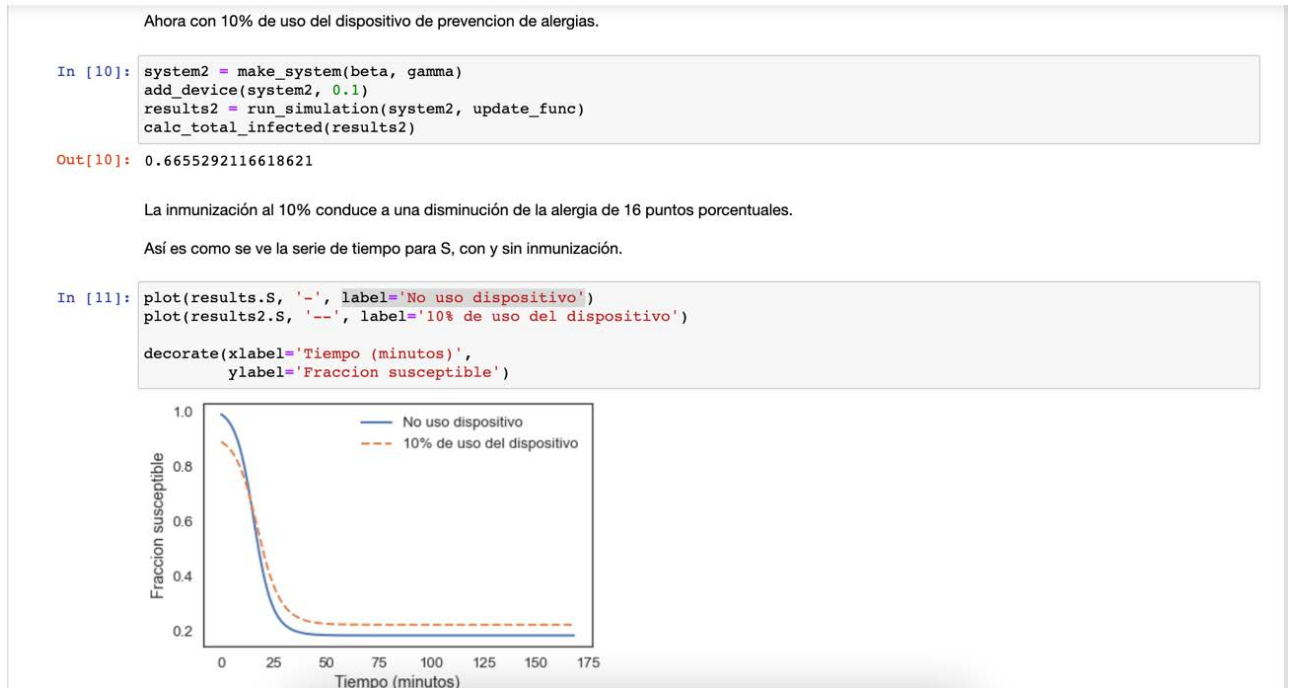


Ilustración 29 Resultados del uso del Smart Analyzer vs No uso del Smart Analyzer

Analizando los resultados de la simulación, podemos ver que en términos porcentuales según muestra la tabla, si frecuentemente las personas con hipersensibilidad utilizan el dispositivo podrán tener mejor calidad de vida, mientras que las que no lo usan estarán más susceptibles a padecer de una reacción alérgica o seguir en su constante lucha con su hipersensibilidad.

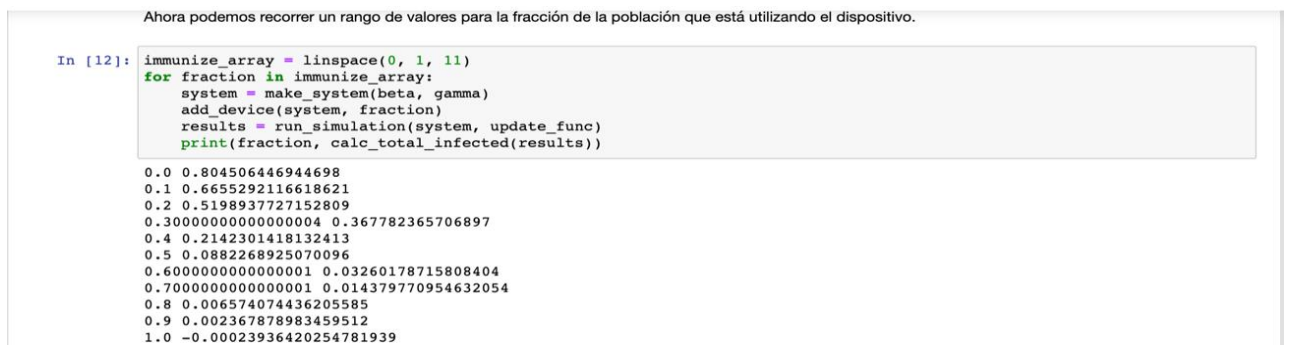


Ilustración 30 Recorrido de eficiencia del uso del dispositivo

Utilizando un método de barrido que almacena los resultados en un objeto y lo muestra en una tabla podemos notar con mayor claridad la factibilidad del dispositivo.

Esta función hace lo mismo y almacena los resultados en un objeto "Sweep".

```
In [13]: def sweep_immunity(immunize_array):
    """Sweeps a range of values for immunity.

    immunize_array: array of fraction immunized

    returns: Sweep object
    """
    sweep = SweepSeries()

    for fraction in immunize_array:
        system = make_system(beta, gamma)
        add_device(system, fraction)
        results = run_simulation(system, update_func)
        sweep[fraction] = calc_total_infected(results)

    return sweep
```

Así es como lo ejecutamos

```
In [14]: immunize_array = linspace(0, 1, 21)
infected_sweep = sweep_immunity(immunize_array)
```

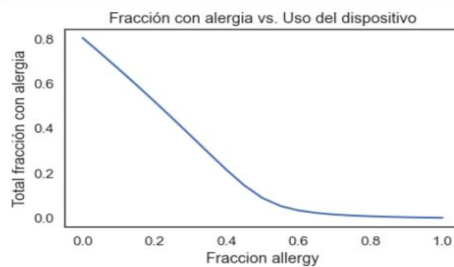
0.40	0.214230
0.45	0.144047
0.50	0.088227
0.55	0.052530
0.60	0.032602
0.65	0.021303
0.70	0.014380
0.75	0.009799
0.80	0.006574
0.85	0.004193
0.90	0.002368
0.95	0.000927
1.00	-0.000239

Ilustración 31 Organizando los resultados en un barrido "Sweep" o tabla

Ilustración de simulación tomando el 40% de los datos suministrados del modelo SIS.

Y así es como se ven los resultados.

```
In [17]: plot(infected_sweep)
decorate(xlabel='Fracción alergia',
        ylabel='Total fracción con alergia',
        title='Fracción con alergia vs. Uso del dispositivo',
        legend=True)
```



Si el 40% de la población está protegida con el dispositivo, menos del 4% de la población se enferma por causa de alguna alergia de la cual padece.

Ilustración 32 Grafica de impacto del uso del dispositivo Smart Analyzer

CAPÍTULO 5 CONCLUSIONES

El objetivo fundamental de este trabajo de grado fue crear una solución que beneficie a personas hipersensibles a ciertos cambios que se manifestaban en sus cuerpos, creando un dispositivo inteligente con la funcionalidad adecuada para medir y prevenir reacciones alérgicas.

Debido a la naturaleza del proyecto, la creación del modelo tridimensional fue diseñada para tener una percepción física más clara del dispositivo Smart Analyzer, es importante resaltar que el uso que se le espera dar solo es para tratamientos que tengan que ver con alergias y/o ataques alérgicos.

Sin embargo, existe al menos un paralelo con otro proyecto de un dispositivo capaz de inyectar insulina a personas que su condición es diabetes, pero no lo hace de manera analítica como está pensado hacer el Smart Analyzer, por lo que las posibilidades de comparación con ese proyecto y más similares es reducida.

El proyecto presentado en este trabajo de grado ha sido orientado a un área en específico, la cual pertenece a la medicina, con el objetivo de salvaguardar, cuidar y brindar una mejor calidad de vida a personas con hipersensibilidades a través de un dispositivo que estará colocado en la muñeca como un brazalete y que puede ser utilizado en todo momento.

CAPÍTULO 6 RECOMENDACIONES

En el presente apartado se detalla el conjunto de lineamientos que deben de ser tomados en cuenta tan pronto se ponga en marcha el proyecto en este trabajo de grado, las recomendaciones a seguir son las siguientes:

- A pesar de ser simulaciones de un dispositivo, es importante que se les explique al personal médico cómo funciona el dispositivo para futuras indicaciones.
- Implementar el funcionamiento real del dispositivo.
- Implementar funcionamientos adicionales para detectar más anomalías en el cuerpo humano, por ejemplo, fiebre, dolor de cabeza y células cancerígenas.
- Realizar mantenimientos para que el dispositivo pueda seguir funcionando.
- Se debe considerar que al momento de realizar el dispositivo funcional es solo de uso único puesto que esto con la inteligencia artificial analiza a la persona y por motivos de seguridad no se puede prestar ya que esto inyecta automáticamente una dosis y no es debido que por higiene y evadir cualquier infección o enfermedad otra persona lo utilice.
- Implementar funcionalidad para analizar de igual manera la glucosa para las personas diabéticas.
- Aunque está basado en hipersensibilidades, también se debe considerar añadirle una funcionalidad para medir la presión arterial de manera que esto pueda evitar complicaciones para el usuario.

CAPÍTULO 7 REFERENCIAS

(Celada Quezada, R. (2010). Obtenido de

<https://repositorio.binasss.sa.cr/xmlui/handle/20.500.11764/355>

(Rolando Celada Ramon, V. (2010). Obtenido de

<https://repositorio.binasss.sa.cr/xmlui/handle/20.500.11764/355>

ADVOCATE. (s.f.). Obtenido de <https://www.medicalexpo.es/prod/pharma-supply-inc/product-107000-845274.html>

al., B. e. (2015). *Modelo SIS*. Septiembre.

Alvarez, E. (2016). *ComputerHoy*. Obtenido de

<https://computerhoy.com/noticias/life/primera-bomba-automatica-insulina-venta-2017-51926>

Anaconda. (s.f.).

Bonet, A. G. (s.f.). *ELSEVIER*. Obtenido de <https://www.elsevier.es/es-revista-offarm-4-articulo-alergias-antialergicos-causas-tipos-tratamiento-13059410>

Bozzola, M. (2015). Obtenido de <https://www.scidev.net/america-latina/seguridad-alimentaria/noticias/sensor-detecta-alimentos-que-producen-alergias.html>

Canonica, P. G. (2001). *Libro Blanco sobre Alergia de la WAO*. World Federation of Allergy, asthma & Clinical Immunology Societics.

Cardona, R. (30 de Mayo de 2016). *fdnn*. Obtenido de

<https://www.fundaciondiabetes.org/general/articulo/173/sistemas-de-monitorizacion-continua-de-glucosa>

Celada Quezada, C. R. (Abril de 2010). *Repositorio CCSS*. Obtenido de

<https://repositorio.binasss.sa.cr/xmlui/handle/20.500.11764/355>

- Corso, P. (2015). *SciDevNet*. Obtenido de <https://www.scidev.net/america-latina/seguridad-alimentaria/noticias/sensor-detecta-alimentos-producen-alergias.html>
- EcuRed*. (s.f.). Obtenido de https://www.ecured.cu/Modelado#.C2.BFQu.C3.A9_es_un_modelo.3F
- Elsevier. (2001). *Anales de pediatria*. Madrid, España: ScienceDirect. Obtenido de <https://www.sciencedirect.com/science/article/abs/pii/S169540330177590X>
- ETSETB. (s.f.). *upcommons*. Obtenido de https://upcommons.upc.edu/bitstream/handle/2117/192916/tema_5_simulink-5156.pdf?sequence=7&isAllowed=y
- Farfán, R. (20 de agosto de 2013). *prezi*. Obtenido de <https://prezi.com/yvqk4c05lqul/1-que-es-un-dispositivo-inteligente/?frame=d9d91499f820316bc1b0e340ee71d9293a48e42b>
- Hirsch, L. (s.f.). *Kidshealth*. Obtenido de <https://kidshealth.org/es/parents/sting-allergy-esp.html#:~:text=%C2%BFQu%C3%A9%20ocurre%20en%20la%20alergia,contiene%20el%20veneno%20de%20insecto.>
- Marson, N. (2016). <https://revistas.psi.unc.edu.ar/index.php/Bitacora/article/view/16317>. *Bitacoradigital*, 1.
- Mathworks. (s.f.). <https://la.mathworks.com/products/simulink.html>. Obtenido de Mathworks.
- MATLAB. (s.f.). Obtenido de <https://juancarlosusomatlab2015.weebly.com/definicion-matlab.html>
- MayoClinic. (1998 - 2020). Obtenido de <https://www.mayoclinic.org/es-es/diseases-conditions/allergies/symptoms-causes/syc-20351497>

Medical Expo. (2020). Obtenido de <https://www.medicalexpo.es/prod/pharma-supply-inc/product-107000-845274.html>

MedlinePlus. (s.f.). Obtenido de <https://medlineplus.gov/spanish/foodallergy.html#:~:text=Los%20s%C3%ADntomas%20de%20una%20alergia,Sarpullido%20o%20eccema>

Moko Smart. (2020). Obtenido de Moko Smart: <https://www.mokosmart.com/es/what-is-a-smart-device/>

Pediatr, R. C. (2002). *Alergia à proteína do leite de vaca.*

project, J. (s.f.). Obtenido de https://es.wikipedia.org/wiki/Proyecto_Jupyter

Rubby Pawankar, G. W. (2001). *Libro Blanco sobre Alergia de la WAO.* World Federation of Allergy, asthma & Clinical Immunology Societics.

Sáez, F. C. (diciembre de 2020). *PMC.* Obtenido de <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7151823/>

SalesForce. (2020). *SalesForce.* Obtenido de <https://www.salesforce.com/mx/blog/2017/6/Que-es-la-inteligencia-artificial.html>

SAVIA. (14 de 05 de 2019). *alergologia.* Obtenido de savia: <https://www.saludsavia.com/contenidos-salud/enfermedades/alergia-a-la-humedad>

Scielo. (s.f.). *Scielo.* Obtenido de https://scielo.conicyt.cl/scielo.php?script=sci_arttext&pid=S0370-41062002000400011

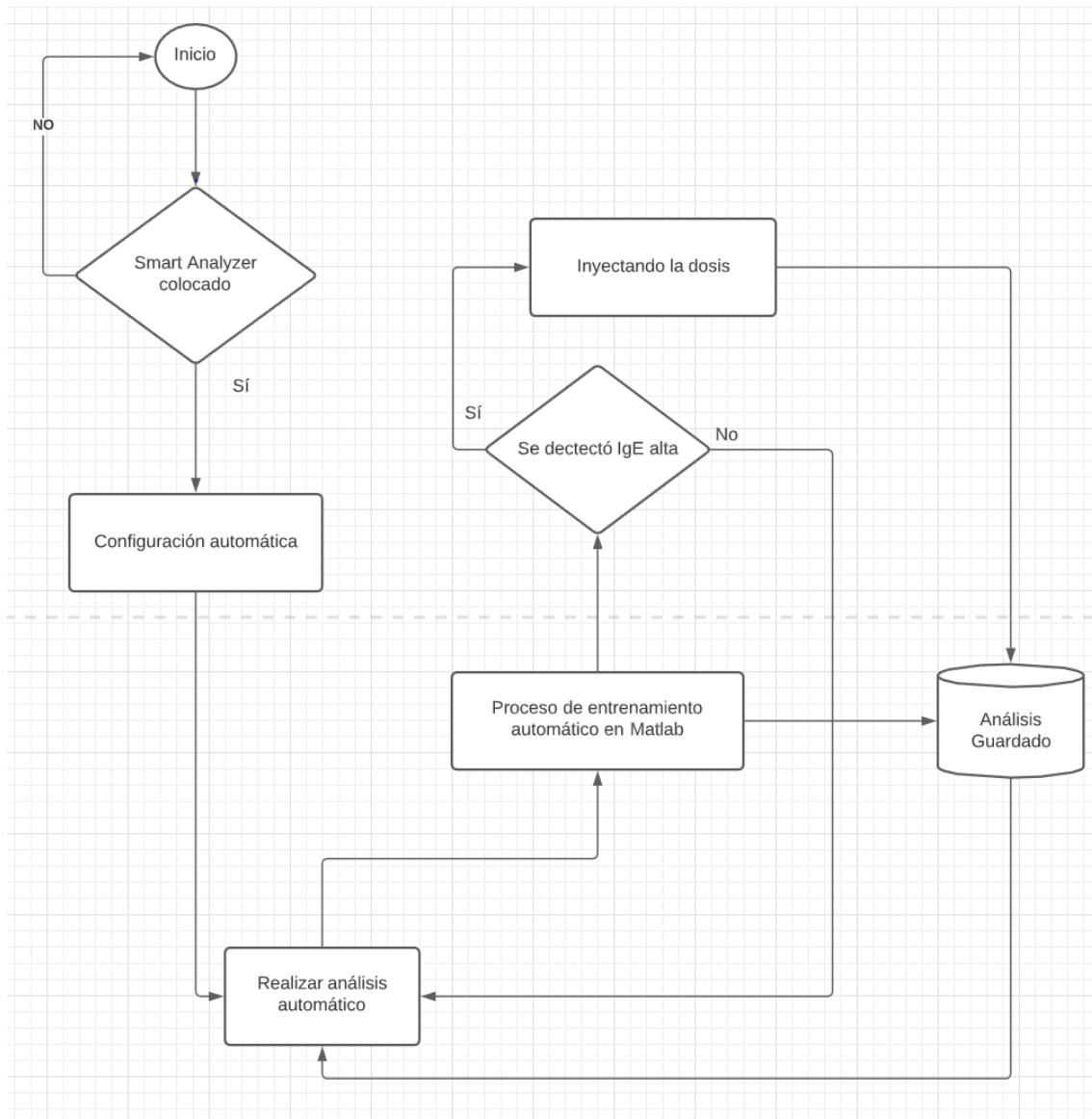
Valenzuela, H. (12 de Agosto de 2020). IgE (Inmunoglobina E). (J. Cabral, Entrevistador)

CAPÍTULO 8 ANEXOS

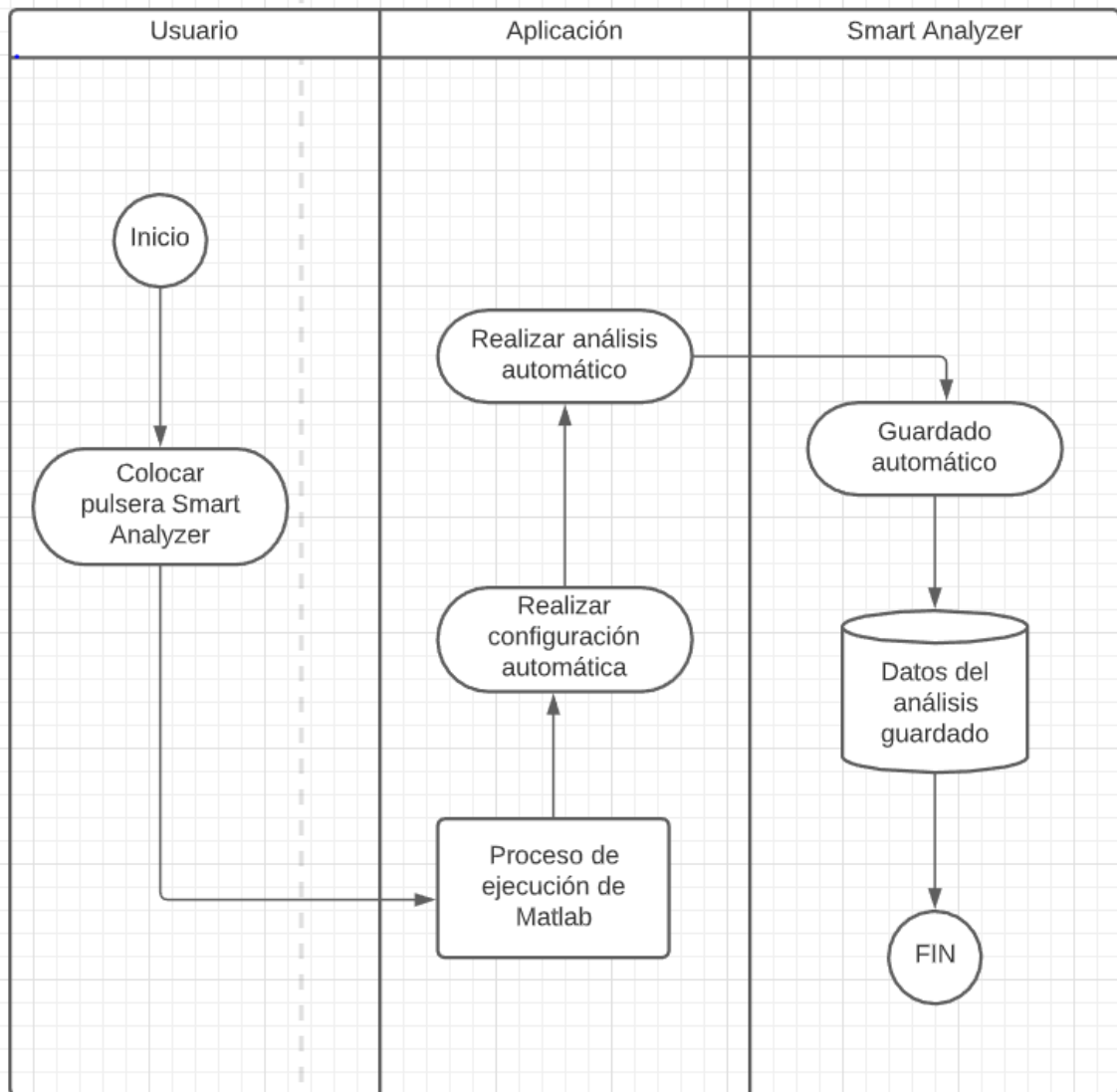
8.1 Preguntas de las entrevistas

1. ¿Qué es una alergia?
2. ¿Qué tipos de alergias existen?
3. ¿Puede una persona tener más de dos tipos de alergias?
4. ¿Cualquier antialérgico es funcional para cualquier alergia?
5. ¿Cómo funciona la alergia en el cuerpo humano?
6. ¿A qué edad normalmente se comienzan a desarrollar las alergias?
7. ¿La sangre puede ser una variante para la causa de que una persona sea alérgica o no?
8. ¿Qué sucede si una persona no es atendida durante las próximas 2 horas en el proceso de una reacción alérgica?
9. ¿Usted ha presenciado casos de anafilaxia en alguno de sus pacientes?
10. ¿Cómo puede tratarse un caso de una persona en estado anafiláctico?
11. Después del proceso alérgico, ¿el cuerpo queda afectado permanentemente? ¿Por ejemplo, algún sarpullido?
12. ¿Qué lugar del cuerpo es la vía más directa para que el antialérgico reaccione inmediatamente?
13. ¿Qué usted piensa de un dispositivo capaz de suministrar el antialérgico automáticamente detecte una subida anormal de la IgE (Inmunoglobina E)?
14. ¿Cree usted que un dispositivo inteligente con la capacidad de medir constantemente la IgE del cuerpo puede ser importante en el cuidado de personas con hipersensibilidad?

8.2 Diagrama de flujo



8.3 Diagrama de Actividades



8.4 Casos de Uso

8.4.1 Caso de uso con alérgeno detectado

CU-01	Caso de uso de alérgeno detectado	
Actor	Usuario/Paciente	
Precondición	Colocar el Smart Analyzer en la muñeca	
Descripción	El usuario debe tener el dispositivo colocado en su muñeca para efectuar su funcionalidad.	
Flujo normal	Paso	Acción
	1	Realizar análisis automático.
	2	Notificar del análisis.
Postcondición	El dispositivo analiza la IgE del propietario del dispositivo	
Flujos Alternos	Paso	Acción
	2	Detectar alérgeno.
	3	Notificar acerca de niveles elevados de la IgE.
	4	Autoinyectar dosis.

UNIVERSIDAD NACIONAL PEDRO HENRÍQUEZ UREÑA
FACULTAD DE CIENCIA Y TECNOLOGÍA
ESCUELA DE INFORMÁTICA

Modelado y Simulación de un Dispositivo Inteligente de Medición y
Prevención de Alergias con Matlab y Simulink.



Proyecto de grado presentado por
Jaison Alexander Cabral Calderón

Ing. Mario Mesa
Miembro del Jurado

Lic. José Aníbal Tolentino
Miembro del Jurado

Ing. Pantaleón Mueses
Miembro del Jurado

José Ramón Romero (Feb 9, 2021 10:57 AST)

Ing. José Ramón Romero
Consejero

Ulises Agüero. Ph.D.
Director



Fecha de sustentación: 08/02/2021