



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería de la Salud



INGENIERÍA
DE LA SALUD

**TFG del Grado en Ingeniería de la
Salud**

**Modelado determinista de
epidemias
Documentación Técnica**

Presentado por Lucía Segura Benito
en Universidad de Burgos

6 de julio de 2025

Tutores: Daniel Sarabia Ortiz – Alejandro Merino
Gómez

Índice general

Índice general	i
Índice de figuras	iii
Índice de tablas	iv
Apéndice A Plan de Proyecto Software	1
A.1. Introducción	1
A.2. Planificación temporal	1
A.3. Planificación económica	8
A.4. Viabilidad legal	11
Apéndice B Documentación	15
B.1. Requisitos software y hardware para ejecutar el proyecto . .	15
B.2. Instalación / Puesta en marcha	16
B.3. Manual de usuario	22
Apéndice C Manual del programador	31
C.1. Estructura de directorios	31
C.2. Compilación, instalación y ejecución del proyecto	33
C.3. Pruebas del sistema	34
C.4. Instrucciones para la modificación o mejora del proyecto . .	35
Apéndice D Descripción de adquisición y tratamiento de datos	37
D.1. Descripción formal de los datos	37
D.2. Descripción clínica de los datos	38
D.3. Datos utilizados	39

D.4. Información relevante de los simulaciones	44
Apéndice E Manual de especificación del diseño	47
E.1. Diagramas de bloques	47
E.2. Script MATLAB	52
E.3. Diseño arquitectónico	53
Apéndice F Especificación de Requisitos	57
F.1. Diagrama de casos de uso	57
F.2. Explicación casos de uso	58
Apéndice G Anexo de sostenibilidad curricular	65
Bibliografía	67

Índice de figuras

A.1. Diagrama de Gantt para este proyecto.	8
B.1. Página de inicio de MATLAB.	19
B.2. Instrucción de MATLAB para el paso 3.	20
B.3. Instrucción de MATLAB para abrir la aplicación.	20
B.4. Botón de Run de MATLAB.	21
B.5. Interfaz inicial de la aplicación.	23
B.6. Elección del modelo, en este caso SIR.	26
B.7. Comprobación de que salen los datos por defecto.	27
B.8. Simulación del modelo SIR en la aplicación.	28
B.9. Elección del modelo, en este caso SEIRV.	29
B.10. Simulación del modelo SEIRV en la aplicación con los datos predefinidos.	30
E.1. Diagrama de bloques en Simulink para el modelo SI.	48
E.2. Diagrama de bloques en Simulink para el modelo SIS.	48
E.3. Diagrama de bloques en Simulink para el modelo SIR.	49
E.4. Diagrama de bloques en Simulink para el modelo SIR con regu- lador PID.	50
E.5. Diagrama de bloques en Simulink para el modelo SEIR.	50
E.6. Diagrama de bloques en Simulink para el modelo SIRV.	51
E.7. Diagrama de bloques en Simulink para el modelo SEIRV.	52
E.8. Definición de parámetros en el script MATLAB.	53
E.9. Implementación del regulador PID en el script MATLAB.	53
E.10. Diagrama de clases de la aplicación realizada.	55
F.1. Diagrama de casos de uso de la aplicación realizada.	58

Índice de tablas

A.1. Etiquetas temáticas (labels) utilizadas en la planificación del proyecto.	3
A.2. Costes de software imputados al TFG.	9
A.3. Cálculo de coste económico personal mensual.	10
A.4. Coste económico total para 6 meses.	10
A.5. Resumen económico total del TFG.	10
B.1. Requisitos mínimos y recomendados para la ejecución del proyecto.	16
F.1. CU-1: Seleccionar modelo.	59
F.2. CU-2: Introducir parámetros.	60
F.3. CU-3: Simular modelo.	61
F.4. CU-4: Visualización de resultados.	62
F.5. CU-5: Instalar RunTimes.	63
F.6. CU-6: Instalar .exe.	64

Apéndice A

Plan de Proyecto Software

A.1. Introducción

En este proyecto se ha llevado a cabo el diseño y simulación de modelos epidemiológicos deterministas utilizando Simulink, así como el desarrollo de una aplicación interactiva en App Designer para visualizar los resultados. Para organizar adecuadamente el trabajo, se ha realizado una planificación que contempla los aspectos temporales (distribución de tareas), una estimación básica del coste, y una breve revisión de los aspectos legales relacionados con el uso del software empleado.

A.2. Planificación temporal

Para la planificación temporal del proyecto se ha seguido una metodología ágil inspirada en Scrum, adaptada a un entorno de trabajo individual. Se han utilizado herramientas de GitHub, como los issues y los milestones. Esta metodología ha favorecido una entrega progresiva de resultados, una mejora continua a lo largo de las iteraciones, y una documentación detallada de cada fase del desarrollo.

Todas las tareas y etapas del proyecto han sido organizadas y gestionadas a través del repositorio de GitHub¹. En él se encuentran documentados los milestones, issues, versiones del código y los avances del proyecto de forma detallada.

¹Link para entrar en el repositorio <https://github.com/Luciasegura/TFG>

La estructura del proyecto se basa en la definición de milestones, issues y labels, que permiten dividir el trabajo en fases, tareas y categorías específicas, se explica cada una a continuación.

- **Milestones:** son las grandes fases del proyecto. Cada milestone tiene una fecha límite y agrupa tareas relacionadas. Por ejemplo, la Investigación teórica es un milestone que va del 19 de febrero al 4 de marzo.
- **Issues:** son las tareas concretas que hay que hacer dentro de cada milestone. Por ejemplo, “estudiar modelos” es una issue dentro del milestone Investigación teórica.
- **Labels:** las etiquetas ayudan a identificar el tipo de tarea, por ejemplo investigación, implementación, documentación, etc. Así se sabe rápidamente de qué va cada issue.

Durante el desarrollo del proyecto, se utilizaron distintas *labels* para clasificar y organizar las *issues*, lo que facilitó la gestión del trabajo y el seguimiento del progreso dentro de cada *milestone*. Estas etiquetas temáticas, descritas en la Tabla A.1, permitieron categorizar las tareas según su naturaleza técnica, conceptual o funcional.

Etiqueta	Descripción
análisis	Tareas enfocadas al estudio de resultados obtenidos y su interpretación epidemiológica o matemática.
aplicación	Actividades relacionadas con el diseño e implementación de la app interactiva en App Designer .
conceptos epidemiológicos	Estudio de conceptos fundamentales como el número reproductivo básico, tasas de transmisión o inmunidad.
documentation	Redacción de la memoria, anexos, bibliografía y otros documentos del TFG.
investigación	Fase teórica del proyecto: revisión de literatura, estudio de modelos y recopilación de información.
matlab	Uso de MATLAB para implementar modelos, generar gráficas o scripts de simulación.
mejora modelo	Cambios y ajustes realizados para optimizar el comportamiento de los modelos implementados.
modelado matemático	Tareas relacionadas con la formulación y análisis matemático de los modelos compartimentales.
modelo epidemiológico	Implementación y análisis de modelos como SI, SIS, SIR, SEIR, SIRV y SEIRV.
PID	Aplicación del regulador PID para el control de la propagación epidémica en el modelo.
regulador	Estudio y aplicación de técnicas de control para estabilizar o reducir los contagios.
simulación	Ejecución de experimentos computacionales en MATLAB o Simulink .
simulink	Desarrollo de diagramas de bloques en Simulink para representar los modelos dinámicos.
vacunación	Inclusión del efecto de la vacuna en los modelos y análisis del impacto en la propagación.
variables y parámetros	Estudio, definición y ajuste de los parámetros clave (como β , γ , ν) y variables del sistema.

Tabla A.1: Etiquetas temáticas (labels) utilizadas en la planificación del proyecto.

Estas etiquetas ayudaron a identificar con claridad el objetivo de cada tarea, y permiten entender el enfoque multidisciplinar del trabajo: desde la teoría matemática hasta la aplicación práctica en simuladores y herramientas digitales.

A continuación se explican las milestones con sus issues correspondientes.

Milestone 1: Investigación teórica

Periodo: 19 de febrero – 14 de mayo

Durante esta fase se llevó a cabo el estudio de los fundamentos del modelado matemático de epidemias. Se analizaron las diferencias entre los modelos compartimentales (SI, SIS, SIR y SEIR), sus variables y parámetros, y se profundizó en los conceptos epidemiológicos necesarios para entender su comportamiento.

- Estudio de diferencias entre modelos compartimentales y análisis de variables y parámetros. Se estudian las distintas variantes de modelos compartimentales (SI, SIR, SEIR, etc.) y cómo se definen e interpretan sus variables y parámetros, lo que permite entender mejor su estructura y aplicabilidad.

(24 de febrero – 9 de abril)

- Estudio de conceptos básicos de epidemiología. Se investigan los fundamentos teóricos de la epidemiología y su relación con los modelos matemáticos, incluyendo tasas de infección, recuperación, periodo de incubación, y más.

(19 de febrero – 4 de marzo)

- Investigación del compartimento de vacunación y su efecto en la dinámica del modelo. Se analiza cómo introducir el efecto de la vacunación dentro de los modelos, especialmente en variantes como SIRV y SEIRV, y se investiga su impacto epidemiológico.

(19 de febrero – 14 de mayo)

Milestone 2: Implementación en Simulink

Periodo: 6 de marzo – 16 de mayo

Se implementaron los modelos epidemiológicos en **Simulink**, analizando el efecto de los parámetros y añadiendo el compartimento de vacunación. Esta fase permitió simular distintos escenarios de propagación.

- Implementación de modelos epidemiológicos. Se lleva a cabo la implementación de los modelos básicos en el entorno gráfico de Simulink, permitiendo visualizar la evolución de las epidemias a través de diagramas de bloques.

(6 de marzo – 1 de abril)

- Análisis del efecto de los parámetros y mejora del modelo. Se experimenta cómo pequeñas variaciones en parámetros clave (como beta, gamma, sigma) afectan la dinámica del modelo, lo que permite afinar su comportamiento y ajustarlo a escenarios realistas.

(6 de marzo – 18 de abril)

- Incorporación del compartimento de vacunación en los modelos. Se integra la vacunación en los modelos ya implementados, ajustando ecuaciones y bloques para simular campañas vacunales constantes.

(6 de marzo – 16 de mayo)

Milestone 3: Análisis de datos reales

Periodo: 23 de marzo – 16 de mayo

En esta etapa se buscaron enfermedades reales susceptibles de ser modeladas mediante los modelos implementados, y se realizaron simulaciones comparativas.

- Buscar enfermedades e investigar modelos aplicables. Se buscan datos de enfermedades reales para identificar cuáles se ajustan a los modelos creados y permiten una validación básica de su comportamiento.

(10 de abril – 16 de mayo)

- Simulación de modelos relacionados con enfermedades reales mediante **Simulink**. Se utilizan los modelos implementados para simular el comportamiento de ciertas enfermedades y comparar sus curvas teóricas con datos reales.

(23 de marzo - 16 de mayo)

Milestone 4: Control

Periodo: 16 de mayo – 29 de junio

Se estudió el impacto de medidas de control como la vacunación y el distanciamiento social, y se implementó un controlador PID en **MATLAB** para la gestión dinámica de los infectados.

- Análisis de la vacunación como medida de control. Se evalúa el impacto de la vacunación dentro del modelo como una forma de mitigar o evitar la propagación epidémica.

(16 – 19 de mayo)

- Estudio de medidas de control no farmacológicas. Se investigan otros mecanismos de control (cuarentenas, reducción de contactos, etc.) y cómo se pueden representar matemáticamente en el modelo.

(19 de mayo – 3 de junio)

- Implementación y simulación del controlador PID. Se aplica un controlador PID en **MATLAB** para actuar sobre la tasa de infección o el número de infectados, simulando medidas de control automáticas sobre el sistema.

(5 – 6 de junio)

- Mejora regulador PID. Se implementa el modelo SIR con el PID en Simulink. A parte se han mejorado las simulaciones de los modelos en **MATLAB**.

(16 - 29 de junio)

Milestone 5: Aplicación

Periodo: 16 de mayo – 3 de junio

Se desarrolló una interfaz interactiva en **App Designer** para facilitar la visualización y manipulación de los modelos, promoviendo la comprensión intuitiva del comportamiento epidémico.

- Diseño inicial de la aplicación en **MATLAB**. Se define cómo debe funcionar la interfaz gráfica de usuario para que permita modificar parámetros y visualizar las simulaciones de forma interactiva.

(16 – 19 de mayo)

- Implementación funcional de la aplicación. Se desarrolla la app que integra los modelos epidemiológicos con una interfaz usable, permitiendo simular, visualizar y analizar distintos escenarios epidémicos.

(19 de mayo – 3 de junio)

- Ejecución de la aplicación. Se ha implementado una versión independiente de la aplicación, permitiendo su instalación y uso sin necesidad de tener MATLAB previamente instalado.

(16 - 25 de junio)

Milestone 6: Redacción en LaTeX

Periodo: 7 de mayo – 30 de junio

Durante esta etapa se elaboró la memoria del proyecto y sus respectivos anexos, documentando todos los aspectos teóricos, prácticos y experimentales desarrollados.

- Redacción de la memoria principal del TFG. Se escribe la documentación del proyecto, incluyendo introducción, objetivos, metodología, resultados y conclusiones, utilizando LaTeX.

(7 de mayo – 6 de julio)

- Preparación y revisión de los anexos. Se desarrollan los anexos donde se detallan aspectos más técnicos del proyecto como diseño en Simulink, scripts de MATLAB, estructuras de la aplicación y sostenibilidad.

(16 de mayo – 6 de julio)

Para organizar y visualizar de manera clara las fases y tareas, se ha elaborado un diagrama de Gantt que muestra la distribución temporal de cada etapa del proyecto. Este diagrama permite identificar los periodos dedicados a la investigación, desarrollo, redacción y revisión, facilitando así el seguimiento y control del progreso del trabajo. A continuación, se presenta el diagrama de Gantt, figura [A.1](#) correspondiente a la planificación temporal del TFG.

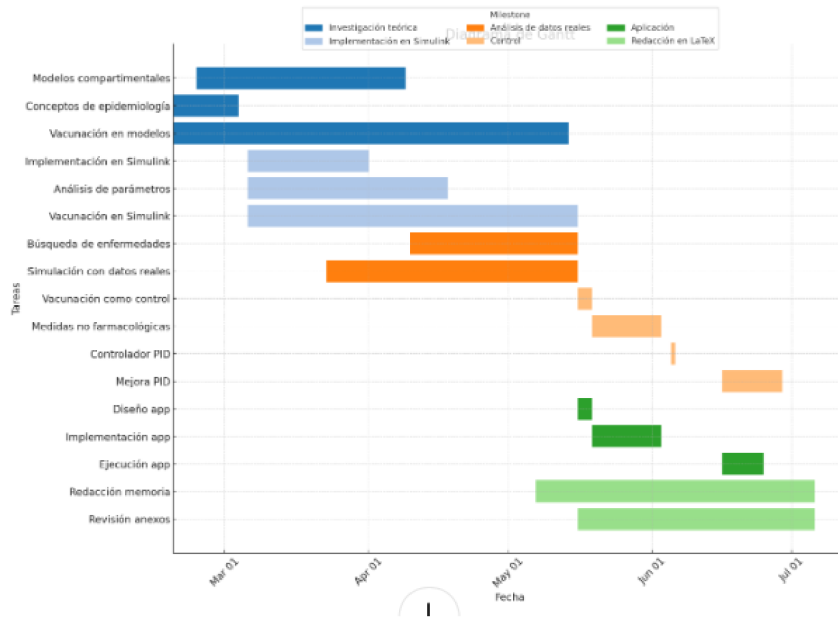


Figura A.1: Diagrama de Gantt para este proyecto.

A.3. Planificación económica

La planificación económica de este trabajo tiene como finalidad estimar y justificar los recursos utilizados durante su desarrollo, tanto materiales como humanos. Se han clasificado los costes en tres grandes categorías: hardware, software y mano de obra. Además, se incluye un resumen económico total que refleja el coste imputado al TFG, es decir, el valor proporcional de cada recurso directamente relacionado con el proyecto. Esta estimación proporciona una visión global del esfuerzo económico asociado y permite valorar la inversión requerida para llevar a cabo un proyecto de estas características.

Costes de hardware

El equipo informático utilizado para el desarrollo del TFG es un Asus Vivobook X409JB, con un valor inicial de 650 €. Para imputar un coste proporcional a su uso específico en el proyecto, se ha considerado una vida útil estimada de 3 años (36 meses), un periodo estándar para la amortización de equipos informáticos debido a la rápida evolución tecnológica.

Según [Ministerio de Hacienda de España, 2024] la amortización mensual se calcula dividiendo el valor del equipo entre los meses de vida útil:

$$\text{Amortización mensual} = \frac{650 \text{ €}}{36 \text{ meses}} \approx 18,06 \text{ €}$$

Dado que el tiempo de uso exclusivo para el desarrollo del TFG ha sido de 6 meses, el coste imputado correspondiente es:

$$18,06 \text{ €} \times 6 \text{ meses} = 108,36 \text{ €}$$

Por lo tanto, el coste imputado del equipo informático asignado al proyecto es de aproximadamente 108,36 €, reflejando un uso proporcional y razonable del equipo durante el periodo de trabajo en el TFG.

Costes de Software

Se incluyen en la Tabla A.2 las licencias de software necesarias para el desarrollo del TFG, como MATLAB, Simulink [MathWorks, nd], Office [Microsoft, nda] y el sistema operativo Windows [Microsoft, ndb]. En cada caso, se ha estimado el grado de uso del software en relación al TFG para calcular el coste imputado.

Concepto	Licencia	Coste (€)	Uso TFG	Imputado (€)
Windows	Windows 10 Pro	150	50 %	75
Office	Office 365 (1 año)	69	100 %	69
MATLAB	Académica (estudiante)	150	100 %	150
Simulink	En MATLAB	0	100 %	0

Tabla A.2: Costes de software imputados al TFG.

Costes de mano de obra

El desarrollo del Trabajo Fin de Grado ha sido llevado a cabo por un estudiante del Grado en Ingeniería de la Salud. A efectos del cálculo económico, se asume que el trabajo ha sido realizado por un ingeniero en prácticas, con un salario bruto mensual estimado de 1.700, valor que se encuentra dentro del rango habitual para un ingeniero recién graduado en el ámbito de la salud [Instituto Nacional de Estadística (INE), 2024].

A continuación, se presenta el desglose del coste mensual de personal teniendo en cuenta las cotizaciones sociales y otros gastos asociados. En la Tabla A.3 se muestra el coste mensual estimado, y en la Tabla A.4 se calcula el coste total correspondiente a 6 meses de trabajo [Project Management Institute, 2017].

Concepto	Coste mensual (€)
Salario bruto medio mensual	1.700,00
Cuota general a la Seguridad Social	401,20
Cotización por contingencias profesionales	10,20
Prestaciones por desempleo	93,50
Fondo de Garantía Salarial (FOGASA)	5,10
Formación	10,00
Total personal mensual	2.220,00

Tabla A.3: Cálculo de coste económico personal mensual.

Concepto de coste	Coste estimado (€)
Coste personal (6 meses)	13.320,00

Tabla A.4: Coste económico total para 6 meses.

Resumen económico total

En el resumen económico representado en la tabla A.5 se agrupan todos los costes imputados al proyecto, reflejando el coste total estimado para el desarrollo del TFG. Esta cifra proporciona una valoración económica global del proyecto teniendo en cuenta todos los recursos utilizados.

Concepto	Coste estimado (€)
Hardware	108,36
Software	294,00
Coste de personal	13.320,00
Total general	13.722,36

Tabla A.5: Resumen económico total del TFG.

El análisis económico realizado pone de manifiesto la implicación de recursos tanto técnicos como humanos en su desarrollo. Gran parte del material y software ya estaba disponible previamente, el coste imputado ofrece una estimación realista del valor del proyecto en un contexto profesional. Permite también reforzar la visión del trabajo como un proceso completo que, más allá del contenido académico, implica una gestión eficiente de recursos.

A.4. Viabilidad legal

Esta sección analiza los aspectos legales relacionados con el desarrollo del proyecto, con el fin de garantizar que se ha actuado conforme a la normativa vigente en materia de propiedad intelectual, uso de software, protección de datos y cumplimiento del reglamento académico.

Propiedad intelectual y licencias

Durante el desarrollo del trabajo se ha hecho uso de diversas herramientas y recursos, tanto propios como de terceros. Para asegurar la legalidad del uso de estos elementos, se ha revisado la licencia de cada uno de ellos:

- **MATLAB y Simulink:** se han utilizado bajo licencia educativa proporcionada por la universidad, lo cual permite el uso no comercial y con fines académicos del software. Esta licencia limita su uso a entornos de formación e investigación.
- **Microsoft Windows 10 y Office 365:** se han empleado versiones con licencia personal, válidas para su uso en el ámbito educativo y no profesional, sin vulnerar las condiciones de uso del proveedor.
- **Repositorios y herramientas de desarrollo:** el código fuente del proyecto se encuentra alojado en GitHub. Se han seguido las recomendaciones de uso de la plataforma y se ha seleccionado una licencia de software libre (**MIT License**) que permite la reutilización, modificación y distribución del código, siempre que se otorgue reconocimiento al autor original.
- **Librerías y recursos externos:** todas las librerías de software utilizadas en el desarrollo de la aplicación han sido seleccionadas por estar bajo licencias de código abierto, lo que permite su uso legal en proyectos académicos. En los casos donde se han empleado fragmentos

de código o documentación externa, se ha incluido la correspondiente cita y fuente.

Respecto al software desarrollado en este proyecto, se ha decidido licenciarlo bajo una licencia **Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0)** [[Creative Commons, 2013](#)].

Esta licencia permite a cualquier persona **copiar, distribuir, comunicar públicamente, ejecutar y adaptar** el software, incluyendo su transformación o creación de obras derivadas, **siempre que se cumplan las siguientes condiciones** [[Commons, 2013](#)]:

1. **Reconocimiento:** Se debe otorgar el crédito correspondiente al autor original, de forma adecuada, proporcionando una referencia clara a la autoría del software. Este reconocimiento no debe sugerir que el autor respalda el uso que se haga del software.
2. **Uso no comercial:** El software **no puede ser utilizado con fines comerciales**. Esto incluye cualquier uso cuya finalidad principal sea obtener una ventaja comercial directa o indirecta o una compensación monetaria.
3. **Sin restricciones adicionales:** No se pueden aplicar términos legales o medidas tecnológicas que restrinjan legalmente a otros de hacer cualquier uso permitido por la licencia.

Esta licencia promueve la libre difusión del conocimiento, permitiendo el uso del software en contextos **académicos, educativos, personales o de investigación**, mientras protege los derechos del autor y evita su explotación económica sin autorización previa.

Protección de datos

Durante el desarrollo del proyecto se han utilizado datos relacionados con enfermedades reales, obtenidos a partir de fuentes públicas y abiertas disponibles en internet, como artículos científicos, bases de datos institucionales y documentos divulgativos.

En todo momento se ha respetado la legalidad vigente en materia de protección de datos. Los datos empleados no contienen información personal

identificable, por lo que no se consideran datos personales según el Reglamento (UE) 2016/679 (RGPD) [Unión Europea,] y la Ley Orgánica 3/2018 (LOPDGDD) [España,].

Además, se ha prestado especial atención a que las fuentes de donde se han extraído dichos datos sean de carácter público, accesibles libremente por cualquier usuario y con fines divulgativos, científicos o educativos. No se ha requerido consentimiento alguno al no tratarse de información sensible vinculada a individuos concretos, sino a descripciones generales de patologías, síntomas, prevalencias u otras variables de carácter epidemiológico. Por tanto, se concluye que el uso de estos datos es legal y éticamente aceptable en el marco de un trabajo académico con fines formativos y no comerciales.

Normativa académica y originalidad

El trabajo ha sido realizado de forma íntegra y original por la autora del TFG, sin incurrir en plagio, copia de código sin atribución o uso indebido de materiales de terceros. Todo el desarrollo se ha realizado conforme a las normas establecidas por la UBU² en relación con la elaboración de Trabajos Fin de Grado, asegurando la autoría propia, la honestidad académica y el cumplimiento del código ético de la institución.

²Universidad de Burgos.

Apéndice B

Documentación

B.1. Requisitos software y hardware para ejecutar el proyecto

Todo esto se encuentra en la referencia [\[MathWorks, 2025b\]](#).

Requisitos de software

Sistema operativo compatible: Windows 10 (64 bits) o superior. También compatible con macOS Mojave o superior y algunas distribuciones Linux (Ubuntu 18.04+)

Entorno principal de desarrollo: MATLAB R2019b (versión 9.7.0, actualización 9 – Build 1737446). Fecha de instalación usada en este proyecto: 5 de agosto de 2021

Toolboxes de MATLAB requeridos:

- Simulink: para diseño y simulación visual de sistemas dinámicos.
- Control System Toolbox: para diseñar y ajustar el regulador PID.
- App Designer: entorno visual para crear la interfaz de usuario (GUI).

Todas las herramientas fueron utilizadas bajo licencia académica, válida para investigación y docencia.

Requisitos hardware

En la tabla B.1¹ se encuentran los requisitos mínimos y recomendados para el hardware.

Recurso	Mínimo	Recomendado
Procesador	Intel Core i3 / AMD Ryzen 3	Intel Core i5 o superior
Memoria RAM	4 GB	8–16 GB (para simulaciones en Simulink)
Espacio en disco	2 GB libres	5 GB (con resultados y backups)
GPU (opcional)	No requerida	Compatible con OpenGL
Resolución de pantalla	1366×768	1920×1080

Tabla B.1: Requisitos mínimos y recomendados para la ejecución del proyecto.

B.2. Instalación / Puesta en marcha

Este apartado explica el procedimiento para instalar el software necesario, así como las instrucciones para ejecutar tanto la aplicación como las simulaciones desarrolladas.

Para ejecutar las simulaciones es imprescindible tener MATLAB instalado en el sistema, ya que el código depende del entorno de desarrollo de MATLAB.

En cambio, para ejecutar la aplicación desarrollada con App Designer, no es necesario disponer de una licencia de MATLAB, siempre que se utilice el archivo ejecutable correspondiente. En este caso, basta con descargar e instalar los componentes del MATLAB Runtime, que son de libre distribución y permiten ejecutar aplicaciones desarrolladas en MATLAB sin necesidad del software completo.

Ejecución de la aplicación

La carpeta **aplicación** contiene todos los archivos necesarios para ejecutar la **Aplicación de Modelos Epidemiológicos**. Existen dos formas de ejecutar la aplicación, según se disponga o no de MATLAB:

1. **Con MATLAB instalado:** ejecutando directamente el archivo `.mlapp` desde App Designer.
2. **Sin MATLAB instalado:** ejecutando el archivo compilado `.exe`, previa instalación del MATLAB Runtime.

¹Las simulaciones complejas en Simulink con controladores PID y grandes datos pueden requerir mayor capacidad de RAM.

Archivos incluidos en la carpeta aplicación

- `aplicacion_modelos.mlapp`: interfaz gráfica de usuario desarrollada con App Designer. Ejecutable desde MATLAB.
- `aplicacion_modelos.exe`: versión compilada de la aplicación para sistemas Windows (64 bits). Ejecutable independiente que requiere los runtimes de MATLAB.
- `MyAppInstaller_web.exe`: instalador automático de los *MATLAB Runtime* (MCR), necesarios para ejecutar el archivo `.exe` si no se dispone de MATLAB instalado.
- Carpeta `aplicacion_modelos/`: contiene los archivos auxiliares generados durante la compilación de la aplicación.
- `README.txt`: archivo de texto con instrucciones detalladas sobre instalación, ejecución y requisitos del sistema.

Requisitos del sistema

- **Con MATLAB instalado:**
 - MATLAB R2019b (o superior)
 - App Designer
 - No requiere instalar MATLAB Runtime adicionales
- **Sin MATLAB instalado:**
 - Sistema operativo Windows 64 bits
 - Espacio libre en disco: mínimo 500 MB
 - Instalación de MATLAB Runtime mediante `MyAppInstaller_web.exe`

Instrucciones de instalación y ejecución

- **Opción 1: Ejecución con MATLAB**
 1. Abrir MATLAB.
 2. Acceder al directorio donde se encuentra el archivo `aplicacion_modelos.mlapp`.

3. En la línea de comandos de MATLAB, ejecutar: `>> open('aplicacion_modelos.mlapp')`.
4. Pulsar en el botón “Run” dentro de App Designer.
5. La aplicación se iniciará y permitirá interactuar con los modelos epidemiológicos.

■ Opción 2: Ejecución sin MATLAB

1. Instalar los runtimes de MATLAB:
 - Ejecutar `MyAppInstaller_web.exe`.
 - Seguir los pasos del asistente de instalación.
2. Ejecutar la aplicación compilada:
 - Hacer doble clic en `aplicacion_modelos.exe`.
 - La aplicación se iniciará automáticamente.

Notas adicionales

- Si ya se dispone de MATLAB instalado, no es necesario ejecutar `MyAppInstaller_web.exe` ni utilizar la versión compilada.
- La instalación de los runtimes puede tardar varios minutos dependiendo de la velocidad de conexión y del sistema.
- La desinstalación de los MATLAB Runtime puede realizarse desde el Panel de control de Windows → “Programas y características” → “MATLAB Runtime”.

Instalación de MATLAB

Para la ejecución del proyecto es necesario disponer de una instalación funcional de MATLAB R2019b o superior. A continuación, se detallan los pasos básicos para su instalación con licencia académica:

1. Acceder a la página oficial [[MathWorks, 2025a](#)] y crear una cuenta usando el correo de la universidad.
2. Tras iniciar sesión, buscar la opción ‘Get MATLAB’ y seguir las instrucciones para asociar la cuenta a la licencia de la universidad, la UBU si dispone de ella.

3. Descargar el instalador correspondiente al sistema operativo (Windows, macOS o Linux).
4. Ejecución del instalador. Iniciar sesión con la cuenta MathWorks y seleccionar la licencia asociada. Elegir los productos a instalar: MATLAB, Simulink, App Designer, Control System Toolbox, etc. Finalizar la instalación siguiendo las instrucciones del asistente.
5. Abrir MATLAB (figura B.1) desde el menú de inicio y comprobar su funcionamiento.

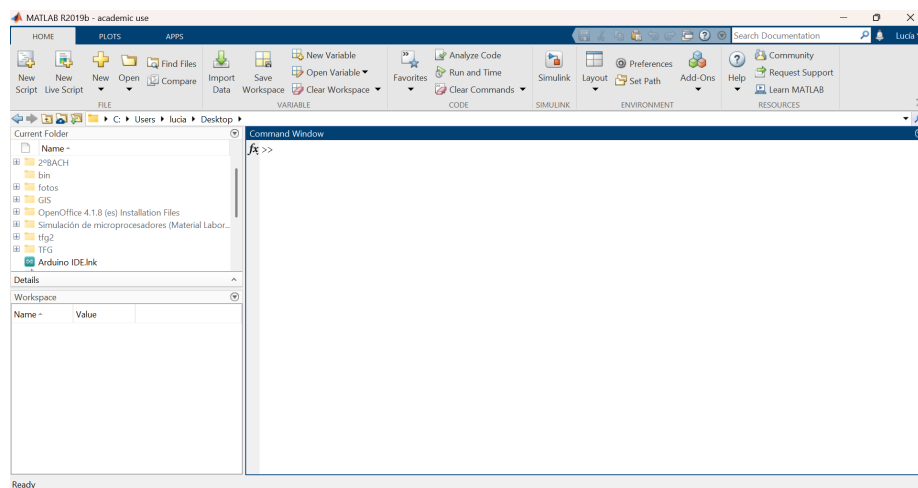


Figura B.1: Página de inicio de MATLAB.

Ejecución del proyecto

Una vez instalado MATLAB, se recomienda seguir los siguientes pasos para ejecutar correctamente el proyecto:

1. Abrir MATLAB R2019b.
2. Descargar o clonar el proyecto en una carpeta local.
3. Añadir la carpeta raíz al path de MATLAB, como se ve en la imagen B.2, escribir en la ventana de comandos

```
addpath(genpath('ruta/del/proyecto'));
```

Figura B.2: Instrucción de MATLAB para el paso 3.

Cambiar 'ruta/del/proyecto' por la ruta en la que se tenga el proyecto. Sirve para añadir al path² de MATLAB la carpeta principal del proyecto y todas sus subcarpetas, de modo que MATLAB pueda acceder a todos los archivos, funciones, scripts y modelos sin errores. No hay que estar navegando carpeta por carpeta o copiando archivos manualmente.

4. Para lanzar la aplicación con interfaz gráfica, abrir el archivo .mlapp o ejecutar en la ventana de comandos de MATLAB lo que se ve en la figura B.3

```
appdesigner('aplicación/aplicacion_modelos.mlapp')
```

Figura B.3: Instrucción de MATLAB para abrir la aplicación.

5. Los modelos desarrollados en Simulink se encuentran organizados dentro de la carpeta Simulink/Modelos Simulink, con extensión .slx. Para abrir cualquiera de ellos, basta con hacer doble clic sobre el archivo correspondiente.

Por otro lado, los scripts de MATLAB se localizan en la carpeta matlab y tienen extensión .m. Al igual que los modelos, estos pueden abrirse directamente haciendo doble clic sobre el archivo. Una vez abiertos en el entorno, la ejecución se realiza pulsando el botón Run, que se puede observar en la figura B.4 donde se encuentra.

²Ruta de búsqueda, conjunto de carpetas donde MATLAB busca archivos cuando se ejecutan comandos o funciones.

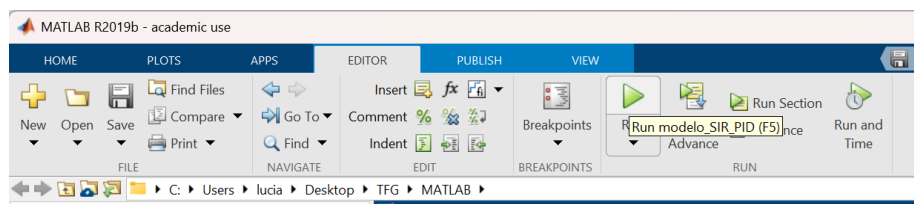


Figura B.4: Botón de Run de MATLAB.

Apertura de modelos en Simulink

Los modelos desarrollados con Simulink se encuentran organizados en la carpeta `simulink/modelos de simulink`, dentro del directorio principal del proyecto. Todos los archivos tienen extensión `.slx`, correspondiente al formato de modelos de Simulink.

Para abrirlos desde MATLAB, se puede proceder de las siguientes maneras:

1. Abrir MATLAB.
2. Acceder a la carpeta `simulink/modelos de simulink`.
3. Localizar el archivo `.slx` correspondiente al modelo deseado.
4. Abrirlo de una de las siguientes formas:
 - Doble clic sobre el archivo desde el explorador de archivos de MATLAB.
 - O bien, escribir en la consola de MATLAB:

```
open('simulink/modelos de simulink/modelo_SI.slx')
```

Sustituir `modelo_SI.slx` por el nombre del archivo que se quiera abrir.
5. Una vez abierto, el modelo puede editarse o simularse desde el entorno de Simulink.

Se recomienda añadir la carpeta al *path* de MATLAB para evitar errores de acceso.

Ejecución de scripts en MATLAB

Los scripts desarrollados en lenguaje MATLAB se encuentran en la carpeta **MATLAB**, dentro del directorio raíz del proyecto. Uno de los principales archivos es `modelo_SIR_PID.m`, que implementa un modelo epidemiológico con control PID.

Para abrir y ejecutar este archivo, seguir los siguientes pasos:

1. Abrir MATLAB.
2. Navegar hasta la carpeta **MATLAB**.
3. Abrir el archivo de una de estas formas:
 - Doble clic sobre `modelo_SIR_PID.m`.
 - O bien, ejecutar en la consola de MATLAB:
`open('MATLAB/modelo_SIR_PID.m')`
4. Para ejecutar el script, pulsar el botón **Run** en el editor, o escribir en la consola:
`modelo_SIR_PID`

Es recomendable que la carpeta **MATLAB** esté añadida al *path* para facilitar el acceso a funciones auxiliares o archivos relacionados.

B.3. Manual de usuario

Este apartado proporciona una guía práctica para el uso de la aplicación desarrollada en el proyecto. Se describen las funcionalidades principales, la estructura de la interfaz gráfica, y los pasos necesarios para interactuar con los modelos epidemiológicos disponibles.

El objetivo es que cualquier usuario pueda comprender el funcionamiento de la aplicación y utilizarla de forma autónoma. Se incluyen capturas de pantalla, descripciones de los controles y recomendaciones de uso para facilitar la interpretación de resultados y garantizar una experiencia de usuario satisfactoria.

Interfaz principal de la aplicación

La interfaz principal de la aplicación, desarrollada con App Designer de MATLAB, presenta un entorno intuitivo para simular diferentes modelos epidemiológicos. La pantalla inicial se muestra en la figura B.5.

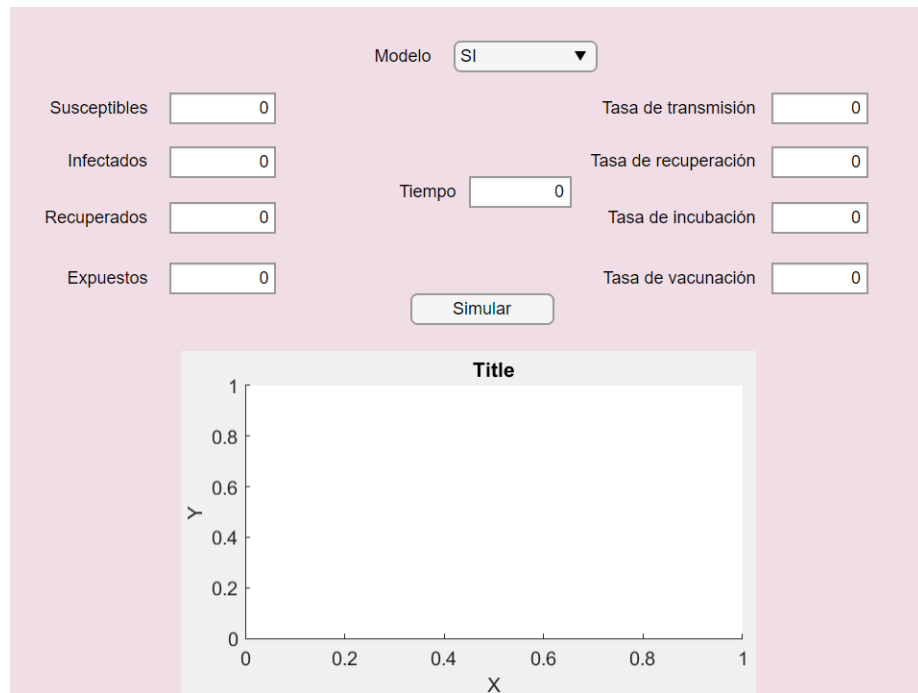


Figura B.5: Interfaz inicial de la aplicación.

Desde el menú desplegable superior se puede seleccionar el modelo deseado entre SI, SIS, SIR, SEIR. A continuación, el usuario puede introducir los valores iniciales para cada grupo de población:

- Susceptibles
- Infectados
- Recuperados
- Expuestos

Así como las tasas características del modelo:

- Tasa de transmisión

- Tasa de recuperación
- Tasa de incubación
- Tasa de vacunación

También se introduce el tiempo de simulación total (en días o unidades temporales equivalentes). Una vez completados los parámetros, el botón "Simular" permite ejecutar la simulación. Los resultados se visualizan automáticamente en una gráfica situada en la parte inferior.

Esta interfaz facilita el análisis comparativo entre modelos y permite observar el comportamiento dinámico del sistema epidemiológico en función de las distintas condiciones iniciales.

Selección del modelo

La aplicación permite seleccionar el tipo de modelo epidemiológico a simular mediante un menú desplegable ubicado en la parte superior central de la interfaz. Los modelos disponibles son:

- **SI**: modelo simple de contagio sin recuperación.
- **SIS**: permite reinfección tras la recuperación.
- **SIR**: incluye inmunidad permanente tras la recuperación.
- **SEIR**: añade una fase de exposición previa al contagio.
- **SIRV**: añade una tasa de vacunación al modelo SIR.
- **SEIRV**: añade tasa de vacunación al modelo SEIR.

Al seleccionar un modelo, la interfaz se adapta automáticamente, activando o desactivando los campos de entrada correspondientes. Por ejemplo, el campo "Expuestos" sólo se habilita si se selecciona el modelo SEIR, y la "Tasa de vacunación" se emplea únicamente en aquellos modelos que incorporan estrategias de vacunación como SIRV o SEIRV. Además al seleccionar un modelo, se cargan automáticamente valores por defecto en todos los parámetros y campos de entrada. Esto permite ejecutar simulaciones de ejemplo directamente, sin necesidad de introducir manualmente todos los datos. Estos valores predefinidos están pensados para ofrecer resultados representativos del comportamiento típico de cada modelo.

Ejecución de simulaciones

Una vez configurados los valores iniciales de la población y los parámetros del modelo, el usuario debe presionar el botón **Simular**. Al hacerlo, se ejecuta la resolución del sistema de ecuaciones diferenciales asociado al modelo epidemiológico. Las simulaciones se realizan empleando funciones de MATLAB basadas en ode45, y los resultados se visualizan directamente en el eje gráfico de la interfaz. Las curvas generadas permiten analizar el comportamiento de cada grupo poblacional en el tiempo.

El sistema también incluye controles para reiniciar los parámetros y ejecutar nuevas simulaciones con diferentes configuraciones.

Casos de uso prácticos

A continuación, se presentan dos ejemplos prácticos de uso de la aplicación, explicados paso a paso. Estos permiten comprobar cómo se comporta la simulación en diferentes escenarios epidemiológicos.

Ejemplo 1 – Simulación con modelo SIR

Pasos:

1. Seleccionar el modelo **SIR** en el menú desplegable [B.6](#).

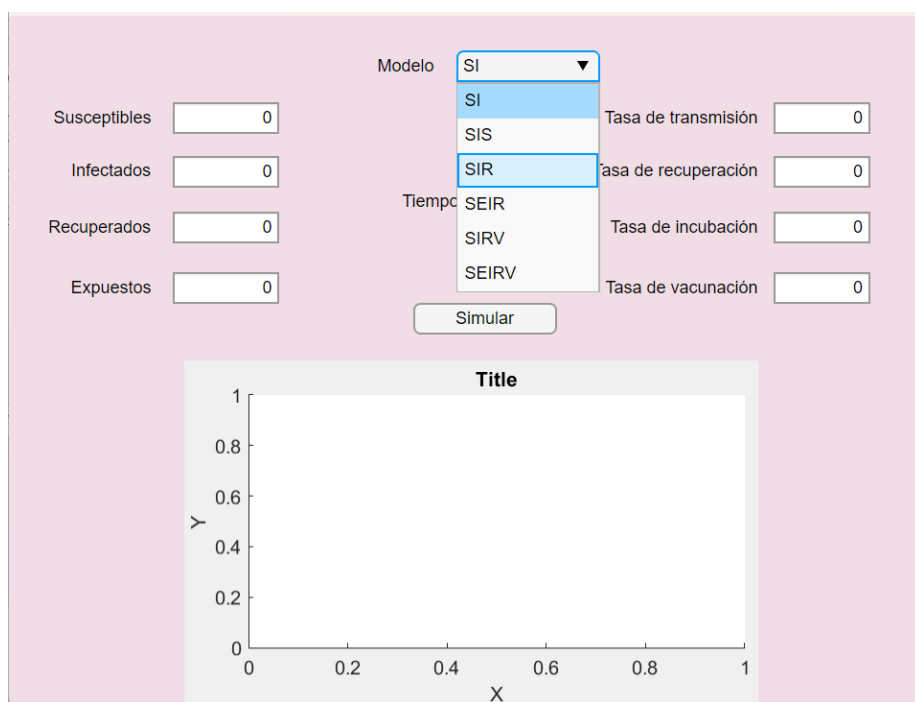


Figura B.6: Elección del modelo, en este caso SIR.

2. Verificar que los valores por defecto se han cargado correctamente (se pueden modificar si se desea) [B.7](#).

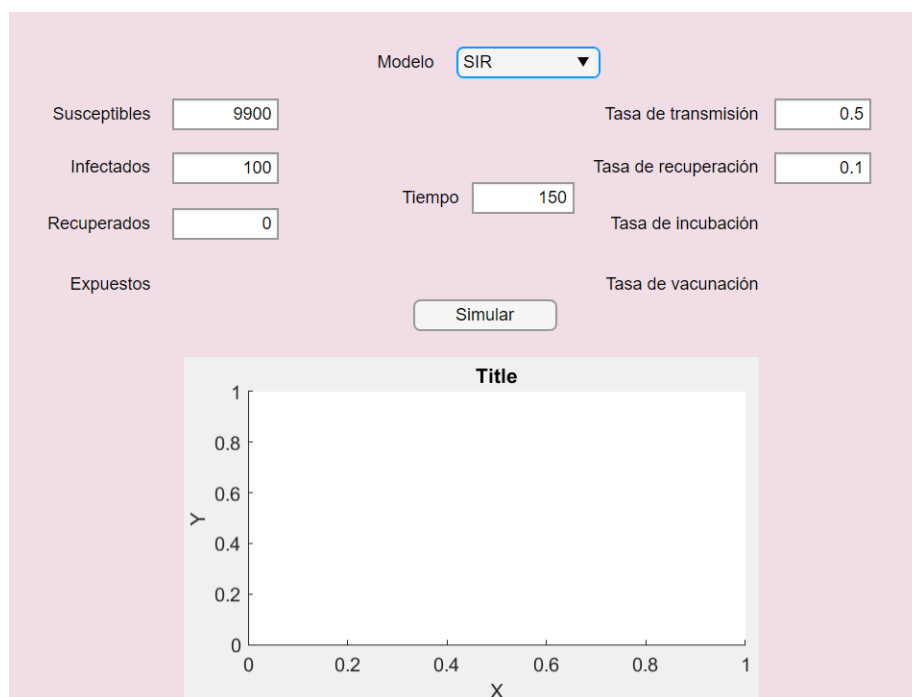


Figura B.7: Comprobación de que salen los datos por defecto.

3. Pulsar el botón **Simular**.
4. Observar la evolución temporal de las tres poblaciones en la gráfica [B.8](#)

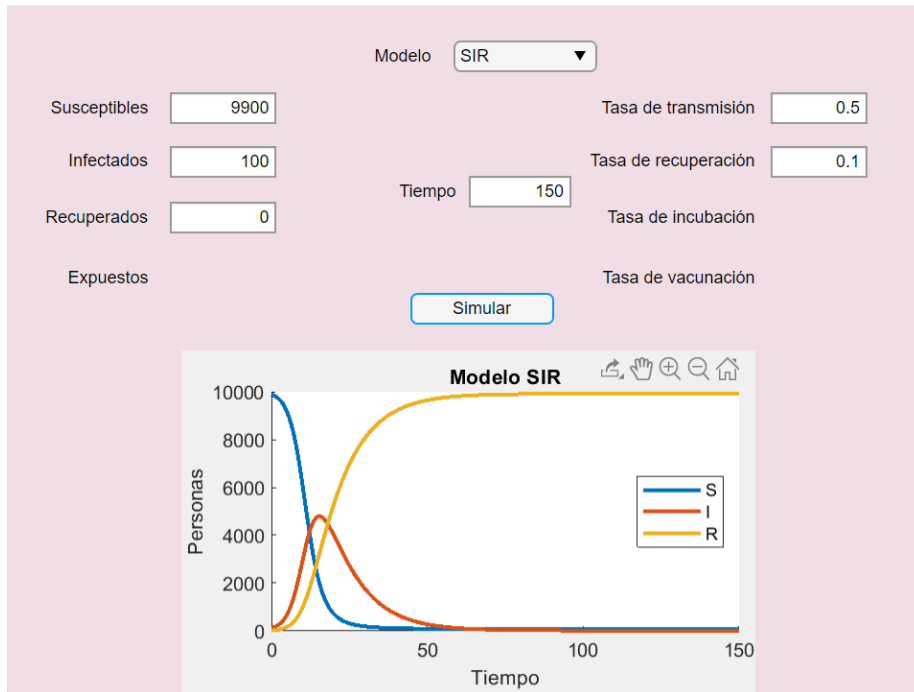


Figura B.8: Simulación del modelo SIR en la aplicación.

Este ejemplo permite visualizar cómo aumenta inicialmente el número de infectados, hasta alcanzar un pico, seguido de una disminución conforme los individuos se recuperan y se inmunizan.

Ejemplo 2 – Simulación con modelo SEIR

Se siguen los mismos pasos que antes:

1. Seleccionar el modelo **SEIR** [B.9](#).

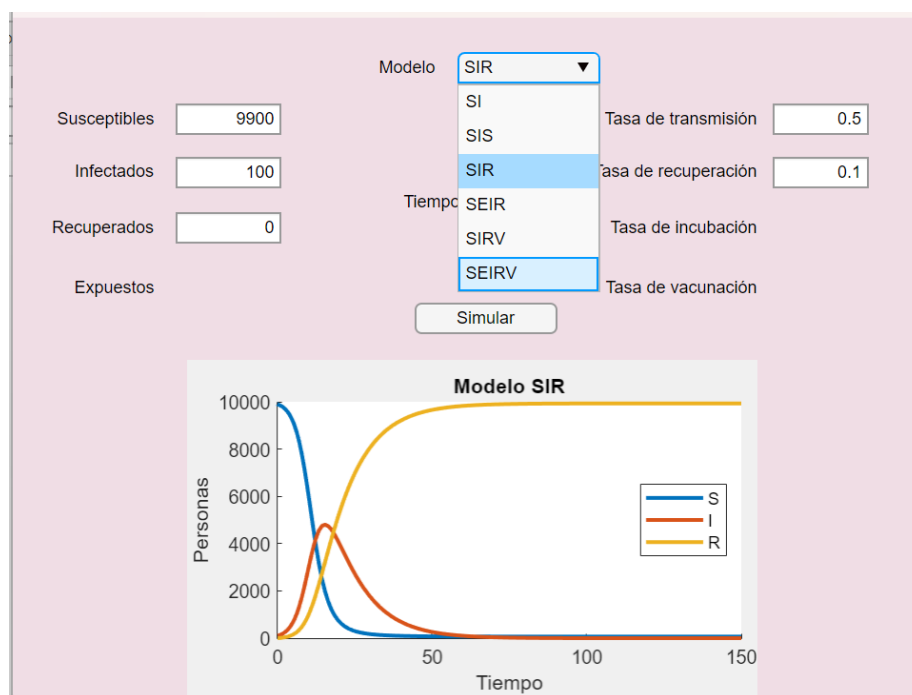


Figura B.9: Elección del modelo, en este caso SEIRV.

2. Comprobar que los valores se han precargado automáticamente, se pueden cambiar si se desea.
3. Hacer clic en **Simular**.
4. Observar el efecto del regulador PID: el número de infectados se mantiene bajo control y disminuye más rápidamente [B.10](#).



Figura B.10: Simulación del modelo SEIRV en la aplicación con los datos predefinidos.

Manual del programador

C.1. Estructura de directorios

A continuación se describe la estructura del proyecto, así como el contenido y propósito de cada carpeta y archivo relevante, todo está en el repositorio de GitHub. El proyecto está organizado en una estructura de carpetas que separa claramente los distintos componentes como se describe a continuación.

■ Carpeta aplicación

Contiene los archivos necesarios para la ejecución y distribución de la aplicación desarrollada en *App Designer* de MATLAB. A continuación, se describen los elementos incluidos:

- `aplicacion_modelos.mlapp`: archivo principal de la aplicación, editable y ejecutable desde MATLAB con App Designer.
- `aplicacion_modelos.exe`: versión compilada de la aplicación para ejecutarse de forma autónoma sin necesidad de abrir MATLAB.
- `MyAppInstaller_web.exe`: instalador web proporcionado por MATLAB que permite instalar las dependencias necesarias para ejecutar la aplicación si el usuario no dispone de una instalación completa de MATLAB.
- Carpeta `aplicacion_modelos`: contiene los archivos auxiliares generados tras la compilación de la aplicación. Son necesarios para su correcto funcionamiento sin entorno MATLAB.

- **README.txt**: archivo de texto que actúa como guía de usuario. Describe los componentes anteriores y cómo instalar y ejecutar la aplicación.
- **Carpeta bibliografía**
Documentos y artículos científicos en .pdf utilizados para entender y fundamentar los modelos epidemiológicos.
- **Carpeta código**
Contiene los archivos fuente del software desarrollado. Esta carpeta incluye los siguientes archivos en formato .txt:
 - **aplicacion.txt**: contiene el código fuente de la interfaz gráfica de usuario desarrollada en *App Designer* de MATLAB.
 - **PID.txt**: contiene el código del controlador PID implementado en MATLAB.
- **Carpeta documentación overleaf**
Carpeta que contiene los archivos fuente del TFG escritos en LaTeX:
 - **Carpeta img**: imágenes utilizadas en la memoria y anexos.
 - **Carpeta tex**: secciones y capítulos del TFG, tanto memoria como anexos.
 - **anexos.tex**, **memoria.tex**: documentos principales del trabajo.
 - **bibliografia.bib**, **bibliografiaAnexos.bib**: bases de datos bibliográficas en formato BibTeX.
- **Carpeta informes entrega**
Archivos PDF del trabajo final entregado:
 - **anexos_Lucía_Segura.pdf**: anexos en formato .pdf.
 - **memoria_Lucía_Segura.pdf**: memoria en formato .pdf.
- **Carpeta MATLAB**
Archivos y resultados de las simulaciones en entorno MATLAB:
 - **/resultados/**: gráficas generadas desde scripts.
 - **modelo_SIR_PID.m**: código de simulación del modelo SIR con control PID.
 - Imágenes del controlador y parámetros PID.

- **Carpeta Simulink**

Contiene los modelos diseñados en Simulink y sus resultados visuales:

- **Carpeta modelos Simulink:** archivos `.slx` de los modelos SI, SIS, SIR, SEIR, SIRV y SEIRV.
- **carpeta Otras simulaciones:** contiene capturas de pantalla y gráficos generados al modificar distintos parámetros en los modelos matemáticos. Estas simulaciones adicionales permiten realizar un análisis comparativo entre diferentes escenarios y evaluar la sensibilidad del modelo ante variaciones en sus condiciones iniciales o parámetros clave.
- **Carpeta resultados:** imágenes de resultados (`.jpg`).
- Imágenes de los diagramas (`.png`).

- **README.md**

Archivo de introducción al proyecto, útil para usuarios que lo descarguen por primera vez.

Esta organización permite acceder rápidamente a cada componente del proyecto según su función: código, documentación, simulaciones o entregas finales.

C.2. Compilación, instalación y ejecución del proyecto

El proceso completo de instalación del entorno *MATLAB* y la ejecución de la aplicación se encuentra detallado en el Apéndice B, concretamente en el apartado B.2. A continuación, se resumen las consideraciones técnicas más relevantes desde el punto de vista del desarrollo y la distribución del software.

El proyecto ha sido desarrollado en **MATLAB R2019b**, por lo que se recomienda utilizar esta misma versión o una superior compatible con los archivos generados.

Requisitos del entorno de desarrollo

Para ejecutar los modelos y la aplicación, se requiere disponer de los siguientes componentes instalados:

- **MATLAB** (versión R2019b o superior)
- **Simulink**
- **Control System Toolbox**
- **App Designer**

Los modelos matemáticos implementados en Simulink se encuentran en la carpeta `/Simulink/modelos_Simulink/` con extensión `.slx`, y pueden abrirse directamente desde el entorno MATLAB. Por su parte, los scripts, incluyendo el controlador PID desarrollado sobre el modelo SIR, están ubicados en la carpeta `/MATLAB/` con extensión `.m`.

La aplicación desarrollada con App Designer se encuentra en la carpeta `/aplicación/`, dentro del archivo `aplicacion_modelos.mlapp`.

C.3. Pruebas del sistema

Durante el desarrollo del proyecto se llevaron a cabo diferentes pruebas para asegurar el correcto funcionamiento de los modelos implementados y de la interfaz gráfica desarrollada:

- Se validó que los modelos epidemiológicos (SI, SIS, SIR, SEIR y SIR con control PID) producen resultados coherentes para diferentes combinaciones de parámetros.
- En los modelos SIR y SEIR se añadió un componente de **vacunación**. Se realizaron pruebas con tasas de vacunación constantes, observando cómo influye en la evolución de los individuos. Como resultado, se observaron curvas más planas o reducción del pico epidémico, en línea con lo esperado teóricamente.
- Se ejecutaron simulaciones con **datos reales de enfermedades infecciosas** contrastando los resultados generados por el modelo con los datos empíricos, para verificar la validez de la estructura matemática.
- Se realizaron pruebas específicas con el modelo **SIR regulado mediante un controlador PID**, comprobando que el sistema responde adecuadamente a variaciones en el número de infectados y que el regulador permite mitigar oscilaciones o picos no deseados.

- En cuanto a la aplicación gráfica (.mlapp) se verificó que cada modelo se carga correctamente, que los valores por defecto permiten una simulación directa y que la visualización gráfica es coherente con la salida del modelo.

Estas pruebas confirman la robustez funcional del sistema, tanto desde el punto de vista técnico como de su aplicabilidad para visualizar dinámicas epidemiológicas complejas.

C.4. Instrucciones para la modificación o mejora del proyecto

El proyecto ha sido desarrollado de forma modular, por lo que resulta relativamente sencillo introducir mejoras. A continuación, se describen algunas recomendaciones para su modificación o ampliación:

- **Ampliación de modelos:** Se pueden implementar nuevos modelos epidemiológicos (por ejemplo, SEIRS, modelos estocásticos o con movilidad) reutilizando la estructura existente en la aplicación.
- **Incorporación de nuevas funcionalidades:** Sería interesante añadir funciones como el cálculo automático del número básico de reproducción R_0 , análisis de sensibilidad, o exportación de resultados en CSV o PDF.
- **Mejora de la interfaz:** La interfaz puede ampliarse con pestañas o menús desplegables para facilitar la navegación. También se podría implementar la carga de datos desde archivos externos.
- **Uso de datos reales actualizados:** El proyecto puede conectarse con fuentes de datos abiertas (como la OMS o Our World in Data) para actualizar automáticamente los valores utilizados en las simulaciones.

Estas sugerencias pueden servir de base para futuras mejoras, tanto a nivel técnico como funcional.

Apéndice D

Descripción de adquisición y tratamiento de datos

D.1. Descripción formal de los datos

Los datos utilizados en este trabajo han sido extraídos de diversos estudios científicos y publicaciones especializadas, todos ellos debidamente citados en la bibliografía de la memoria. No se han empleado conjuntos de datos descargables o bases oficiales, sino que los valores han sido recogidos manualmente o digitalizados a partir de tablas y gráficas publicadas en dichos estudios.

Los datos empleados incluyen:

- Número de personas infectadas, susceptibles, recuperadas y población total.
- Parámetros epidemiológicos clave:
 - Tasa de transmisión (β)
 - Tiempo medio de recuperación (γ^{-1})
 - Tiempo medio de incubación (en modelo SEIR)
 - Cobertura de vacunación.
 - Número reproductivo básico (R_0).

En algunos casos, los parámetros han sido directamente obtenidos de los estudios. En otros, han sido estimados o deducidos a partir de las relaciones matemáticas de los modelos, como por ejemplo:

$$R_0 = \frac{\beta}{\gamma} \quad \text{o} \quad \beta = R_0 \cdot \gamma$$

Los datos han sido tratados en MATLAB para ajustarlos al formato de entrada de los modelos. Estos datos han servido tanto para calibrar los modelos (SI, SIR, SEIR con y sin vacunación) como para comparar los resultados simulados con los patrones observados en situaciones reales.

D.2. Descripción clínica de los datos

Desde un punto de vista clínico, los datos empleados reflejan la evolución temporal de una enfermedad infecciosa en una población concreta, segmentada en distintos estados epidemiológicos:

- **Susceptibles (S):** individuos sanos sin inmunidad previa, que pueden infectarse al entrar en contacto con agentes infecciosos. Clínicamente, representan la población en riesgo de contraer la enfermedad.
- **Expuestos (E):** personas que han sido infectadas recientemente y que se encuentran en un período de incubación durante el cual aún no son contagiosas. Clínicamente, este grupo representa la fase pre-sintomática o latente de la infección.
- **Infectados (I):** individuos que actualmente tienen el virus y son capaces de transmitirlo a los susceptibles. Este grupo clínico puede incluir desde casos asintomáticos hasta pacientes con síntomas severos.
- **Recuperados (R):** se agrupan tanto los individuos que han superado la infección y desarrollado inmunidad, como aquellos que han adquirido protección a través de la vacunación. En ambos casos, se asume que ya no pueden infectarse ni transmitir la enfermedad.

También es importante hacer una descripción clínica de las principales tasas epidemiológicas que se utilizan en este estudio.

- **Tasa de transmisión, β :** Representa la probabilidad o velocidad con la que una persona susceptible se infecta al entrar en contacto con un individuo infectado. Clínicamente, esta tasa está influenciada por la virulencia del patógeno, el comportamiento social (contactos, movilidad),

y medidas de prevención (uso de mascarillas, higiene, distanciamiento social). Un aumento en β puede indicar mayor contagiosidad o relajación de medidas sanitarias.

- **Tasa de incubación, σ :** indica la velocidad con la que los individuos en el estado expuesto (infectados pero no contagiosos) pasan a ser infectantes. Clínicamente, esta tasa está relacionada con el período de incubación del virus, es decir, el tiempo que tarda en manifestarse la capacidad de contagiar tras la infección.
- **Tasa de recuperación, γ :** refleja la velocidad con la que los infectados superan la enfermedad y pasan al estado de recuperados, asumiendo inmunidad. Clínicamente, esta tasa depende de la duración media de la enfermedad y del acceso a tratamientos médicos eficaces. Un aumento en γ puede indicar mejoras en la atención sanitaria o en la respuesta inmune.
- **Tasa de vacunación, ν :** representa la velocidad a la que los individuos susceptibles reciben la vacuna y adquieren inmunidad. Clínicamente, esta tasa refleja la capacidad y cobertura de vacunación en la población, afectando directamente al tamaño del grupo susceptible y, por tanto, al control de la epidemia.

Esta descripción clínica de las variables y parámetros permite entender cómo se traduce la información epidemiológica en las ecuaciones del modelo, y facilita la interpretación de los resultados para la toma de decisiones sanitarias.

D.3. Datos utilizados

A continuación, se describen los datos utilizados para cada modelo, incluyendo su origen, características y el procedimiento seguido para su obtención y tratamiento.

Modelo SI

Se va a estudiar la enfermedad en la región MENA¹, cuya población total se estima en aproximadamente 400 millones de personas. En cuanto a las personas infectadas con VIH es de alrededor de 240000 personas [Khorrami et al., 2023].

¹Oriente Medio y Norte de África.

La tasa de transmisión (β_f) es de 0,105 [Shakiba et al., 2021] para esta enfermedad. Sin embargo, para utilizarlo correctamente es necesario calcular beta, como se muestra en la ecuación (D.7).

$$\beta = \frac{\beta_f}{N} = \frac{0,105}{400\,000\,000} = 2,625 \times 10^{-10} \quad (\text{D.1})$$

Modelo SIS

Se ha tomado como referencia la situación epidemiológica en Estados Unidos durante el 2019. En ese año, la población total del país era aproximadamente 328 millones de personas, en cuanto a los casos de personas infectadas ascendía a 1603473 personas [Pollock et al., 2023].

La tasa de la enfermedad puede variar en función del comportamiento social, el uso de métodos de protección y otros factores. Pero para este estudio se ha estimado una tasa de 0.25, hay que calcular la beta (D.2).

$$\beta = \frac{\beta_f}{N} = \frac{0,25}{328\,000\,000} = 7 \times 10^{-10} \quad (\text{D.2})$$

Respecto a la tasa de recuperación, se considera que, en un país con un sistema sanitario desarrollado como Estados Unidos, los individuos infectados reciben tratamiento de forma relativamente rápida. Aunque sin tratamiento la infección puede durar semanas o incluso meses, con tratamiento, el tiempo medio de recuperación se estima en unos 7 días. Por lo tanto, la tasa de recuperación se calcula como se observa en (D.3).

$$\gamma = \frac{1}{\text{tiempo de recuperación}} = \frac{1}{7} \quad (\text{D.3})$$

Modelo SIR

Se ha elegido analizar la propagación del sarampión en Estados Unidos antes de la introducción de la vacuna en 1963. El país contaba con una población de aproximadamente 189 millones de personas [Datosmacro.com, 2025], mientras que se registraban alrededor de 500 mil casos anuales. Datos que reflejan la alta incidencia del sarampión. Uno de los parámetros más relevantes para modelar esta enfermedad es el número básico de reproducción, que representa el número medio de personas susceptibles que puede llegar a contagiar un solo individuo infectado en una población completamente susceptible. En el caso del sarampión, este valor es excepcionalmente alto, estimándose en torno a $R_0 = 12$ [Solomon, 2019]. Significa que, en ausencia

de medidas de control, cada persona infectada podría contagiar a doce individuos, lo que explica su elevada transmisibilidad y su capacidad para provocar grandes brotes epidémicos.

Además, se conoce que el tiempo de recuperación de la enfermedad es de 14 días [Organización Panamericana de la Salud, sf]. Esto implica que la tasa de recuperación (γ) se calcula en la ecuación (D.4)

$$\gamma = \frac{1}{\text{tiempo medio de recuperación}} = \frac{1}{14} = 0,071 \text{ días}^{-1} \quad (\text{D.4})$$

Dado que la relación entre R_0 , beta y gamma viene dada por la fórmula (D.5)

$$R_0 = \frac{\beta_f}{\gamma} \quad (\text{D.5})$$

Se despeja beta para poder calcular la tasa de transmisión (D.6)

$$\beta = R_0 \times \gamma = 12 \times 0,071 = 0,852 \quad (\text{D.6})$$

Una vez se ha calculado beta (β_f) (D.6), hay que calcular la beta para este modelo (D.7)

$$\beta = \frac{\beta_f}{N} = \frac{0,852}{189\,000\,000} = 4,5 \times 10^{-9} \quad (\text{D.7})$$

Modelo SIRV

La única diferencia introducida es la incorporación de una tasa de vacunación correspondiente a una cobertura del 92,7 % de la población en un periodo de seis meses, lo cual se ajusta a datos reales observados en campañas de vacunación contra el sarampión. Este porcentaje se ha convertido en un valor de tasa (ν) para integrarlo dentro del sistema de ecuaciones del modelo SIRV, y así simular cómo evolucionaría la epidemia bajo una estrategia de inmunización masiva [Centers for Disease Control and Prevention, 2025].

De esta manera, se busca evaluar cuantitativamente cómo la vacunación modifica la evolución del brote, en particular en términos de reducción de contagios y aceleración de la inmunidad colectiva, y contrastar estos resultados con los obtenidos previamente en ausencia de intervenciones. Se tiene la cobertura de vacunación, pero hay que calcular la tasa de vacunación con los datos que se han obtenido. Se calcula a partir de la siguiente fórmula (D.8).

$$\nu = \frac{-\ln(1 - 0,927)}{183} \approx 0,014 \quad (\text{D.8})$$

Esto significa que aproximadamente el 1,4 % de la población susceptible se vacuna cada día durante ese periodo, adquiriendo inmunidad y pasando directamente al compartimento de recuperados. Esta tasa se incorpora en el modelo como un nuevo flujo desde susceptibles a recuperados, simulando el efecto de la vacunación masiva sobre la dinámica de propagación del sarampión.

Modelo SEIR

Se ha elegido como caso de estudio la evolución de la COVID-19 en España, desde la aparición del primer caso confirmado, registrado el 31 de enero de 2020 [EFE, 2025]. Desde esa fecha hasta el 28 de diciembre de 2020 a las 14:00 horas, se notificaron oficialmente un total de 1879413 casos acumulados de personas infectadas por SARS-CoV-2 [de Sanidad, 2025]. En ese mismo año, la población total de España era de aproximadamente 47370000 habitantes [Data Commons, 2025].

En las primeras fases de la pandemia, el número básico de reproducción del virus SARS-CoV-2 se situaba, según estimaciones epidemiológicas, entre 2 y 3 [Álvaro García, 2020]. Esto indica que cada persona infectada transmitía el virus entre 2 y 3 personas, lo que implica una tendencia clara al crecimiento exponencial del brote en ausencia de medidas de control o inmunidad previa en la población.

En cuanto a los parámetros clínicos de la enfermedad, el periodo de incubación, entendido como el intervalo desde la exposición al virus hasta el momento en que el individuo se vuelve contagioso, se estima con una media de 5.4 días [Lauer et al., 2020]. Hay que calcular la tasa de incubación (σ) (D.9)

$$\sigma = \frac{1}{\text{tiempo de incubación}} = \frac{1}{5,4} = 0,185 \quad (\text{D.9})$$

Por otro lado, el tiempo medio de recuperación depende de la gravedad del cuadro clínico. En los casos leves, suele rondar las 2 semanas, mientras que en los casos moderados o graves puede extenderse entre 3 y 6 semanas. Para simplificar el análisis y mantener la coherencia en el modelo, se considerará un tiempo medio de recuperación de 14 días [Ada Health, 2025]. Hay que calcular la tasa de recuperación (D.10)

$$\gamma = \frac{1}{\text{tiempo medio de recuperación}} = \frac{1}{14} = 0,071 \text{ días}^{-1} \quad (\text{D.10})$$

Para finalizar, hay que calcular la tasa de transmisión, para ello utilizamos el R_0 , ya que relaciona la gamma y beta. Se aplica la fórmula del R_0 . Se va a

optar por utilizar el valor de $R_0 = 3$, ya que es un escenario más desfavorable y representativo, es más contagioso. La ecuación (D.5) se utiliza para calcular beta.(D.11)

$$\beta = R_0 \times \gamma = 3 \times 0,071 = 0,213 \quad (\text{D.11})$$

Una vez se ha calculado la beta (β_f), hay que calcular la beta (D.12).

$$\beta = \frac{\beta_f}{N} = \frac{0,213}{47\,370\,000} \approx 4,5 \times 10^{-9} \quad (\text{D.12})$$

Modelo SEIRV

Con el fin de ampliar el análisis realizado sobre el COVID-19 mediante el modelo SEIR, se ha decidido incorporar la vacunación en la simulación a través del modelo SEIRV. Permite estudiar cómo una campaña de vacunación influye en la dinámica de propagación del virus, proporcionando una visión más realista del contexto epidemiológico vivido en España durante la pandemia. Para mantener la coherencia entre escenarios, se han conservado los mismos parámetros epidemiológicos utilizados en el modelo SEIR básico: la tasa de transmisión, el tiempo de incubación y la tasa de recuperación. La diferencia es la inclusión de un parámetro, la tasa de vacunación, que representa el efecto de la inmunización masiva iniciada en España a finales de 2020. Según los datos oficiales, la campaña de vacunación comenzó el 27 de diciembre de 2020, y hasta el 15 de enero de 2021 se habían administrado 768.950 dosis [Ministerio de Sanidad, 2021]. Posteriormente, la cobertura vacunal alcanzó el 92,6 % de la población en un año [Ministerio de Sanidad, 2025]. Para incorporar esta información en el modelo, es necesario calcular la tasa diaria de vacunación, lo cual se realiza con la siguiente fórmula (D.13).

$$\nu = \frac{-\ln(1 - 0,926)}{365} \approx 0,0071 \quad (\text{D.13})$$

Este valor indica que aproximadamente el 0,71 % de la población susceptible se vacuna cada día durante ese periodo. En el modelo, estas personas abandonan directamente el compartimento de susceptibles y pasan al de recuperados, ya que se considera que la vacunación confiere una inmunidad funcional equivalente a la adquirida tras superar la infección. Se mantiene, la misma estructura conceptual del modelo SEIR. La única diferencia es la inclusión del flujo adicional introducido por la vacunación. Así, el modelo SEIRV permite simular con mayor precisión cómo las campañas de inmunización modifican la evolución de la pandemia, en particular reduciendo el número de contagios y acelerando el logro de la inmunidad colectiva.

D.4. Información relevante de los simulaciones

Las simulaciones realizadas permiten analizar la evolución temporal de la epidemia bajo distintos escenarios y condiciones iniciales. A través de estas simulaciones se observan dinámicas clave de la propagación del virus, lo que facilita la evaluación de posibles estrategias de control.

Variaciones en parámetros fundamentales, provocan cambios significativos tanto en la magnitud del brote como en su duración. Por ejemplo, un incremento en la tasa de vacunación reduce notablemente el número máximo de infectados y acorta el tiempo total de propagación de la epidemia. La curva de infectados presenta un pico cuya altura y momento dependen directamente de los parámetros epidemiológicos y de las intervenciones aplicadas. Estas simulaciones permiten identificar puntos críticos a partir de los cuales es posible controlar o incluso erradicar la epidemia mediante intervenciones adecuadas. Los resultados obtenidos aportan una visión cuantitativa del comportamiento dinámico del sistema epidemiológico, lo que resulta útil para diseñar estrategias de mitigación que minimicen el impacto sanitario.

No obstante, es importante tener en cuenta que el modelo se basa en una serie de supuestos que simplifican la realidad para hacer el sistema tratable desde el punto de vista matemático. Entre las principales suposiciones adoptadas se encuentran:

- La población total se considera constante durante el periodo analizado; no se contemplan nacimientos, muertes naturales ni movimientos migratorios.
- La inmunidad adquirida, ya sea por recuperación o por vacunación, se asume permanente durante el horizonte temporal de la simulación.
- La tasa de vacunación (ν) es constante y homogénea, sin variaciones geográficas ni sociales.
- Los individuos en el estado de expuestos (E) no son contagiosos, y el período de incubación es constante y conocido.
- No se contempla la aparición de nuevas variantes del patógeno con características diferentes a las iniciales.
- Las medidas de control (como el uso de mascarillas, confinamientos o distanciamiento social) se consideran constantes o se reflejan mediante ajustes en parámetros como β .

Además, se ha implementado un modelo SIR con control PID en Simulink, lo que permite estudiar la aplicación de estrategias automáticas de regulación sobre la tasa de transmisión. Este enfoque simula la implementación dinámica de medidas de control (como confinamientos o campañas de prevención) en función del número de infectados, actuando como una retroalimentación negativa. Las simulaciones muestran cómo el uso de un regulador PID permite reducir el pico de infecciones y estabilizar el sistema en torno a un nivel deseado de contagios. Esta metodología resulta especialmente útil para analizar escenarios de respuesta adaptativa, donde las medidas se ajustan de forma continua según la evolución de la epidemia. La herramienta Simulink proporciona una interfaz visual intuitiva para modelar estos sistemas dinámicos, facilitando tanto la comprensión del comportamiento del modelo como su modificación para distintos casos prácticos.

Estas condiciones permiten obtener resultados consistentes y comparables entre escenarios, pero también limitan la capacidad del modelo para reflejar situaciones más complejas o variables. Por ello, los resultados deben interpretarse en el contexto de estas hipótesis, y siempre en combinación con datos reales y conocimiento clínico-epidemiológico actualizado.

Apéndice *E*

Manual de especificación del diseño

En este trabajo se ha realizado la implementación de diferentes modelos epidemiológicos deterministas utilizando diversas herramientas del entorno de MATLAB.

E.1. Diagramas de bloques

Se han utilizado diagramas de bloques en **Simulink** para representar los distintos modelos epidemiológicos (SI, SIS, SIR, SEIR, SIRV y SEIRV). Estos diagramas no constituyen planos técnicos tradicionales, pero permiten visualizar de forma clara y esquemática la estructura dinámica de cada modelo, mostrando las interacciones entre los compartimentos y facilitando la comprensión y simulación del sistema.

A continuación se muestran los diagramas de bloques para cada modelo.

Modelo SI representado en la figura E.1, representa la dinámica básica entre individuos susceptibles e infectados, asumiendo que una vez infectados, permanecen en ese estado sin recuperación.

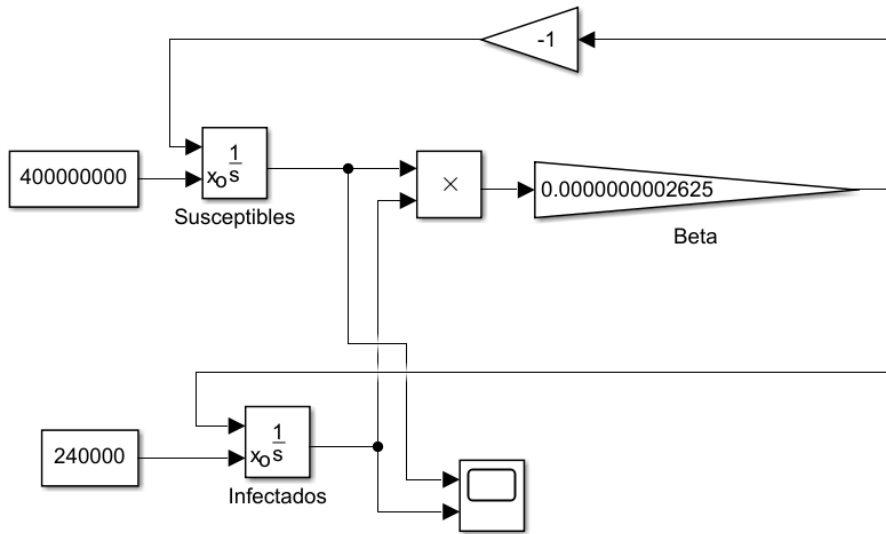


Figura E.1: Diagrama de bloques en Simulink para el modelo SI.

Modelo SIS representado en la figura E.2, contempla que los individuos infectados pueden recuperarse pero sin adquirir inmunidad, volviendo al estado susceptible.

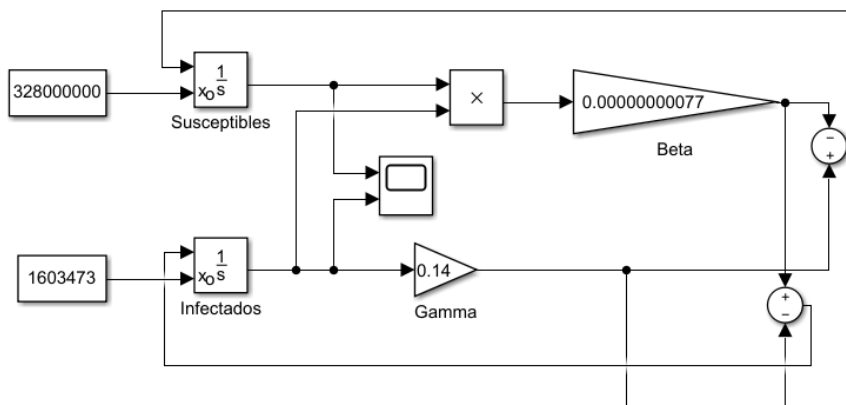


Figura E.2: Diagrama de bloques en Simulink para el modelo SIS.

Modelo SIR representado en la figura E.3, los individuos susceptibles pueden infectarse, pasar a infectados y posteriormente recuperarse con inmunidad permanente.

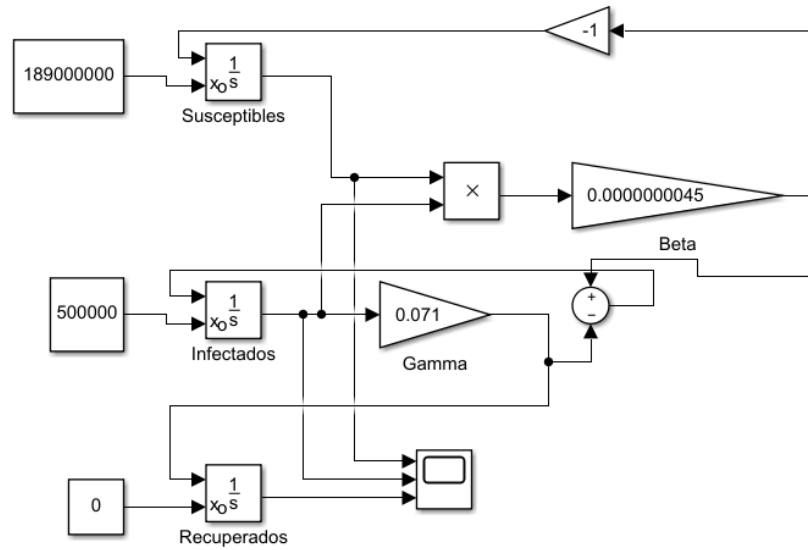


Figura E.3: Diagrama de bloques en Simulink para el modelo SIR.

Modelo SIR con regulador PID representado en la figura E.4 corresponde a una extensión del modelo SIR clásico en la que se incorpora un controlador PID. Este regulador actúa sobre la tasa de transmisión con el objetivo de mantener la proporción de individuos infectados por debajo de un valor de referencia, simulando medidas de control sanitario como confinamientos, campañas de vacunación o restricciones de movilidad. El sistema ajusta dinámicamente la intensidad de dichas medidas en función del error entre el número de infectados reales y el deseado, permitiendo evaluar la efectividad de estrategias de intervención automática en la evolución de la epidemia.

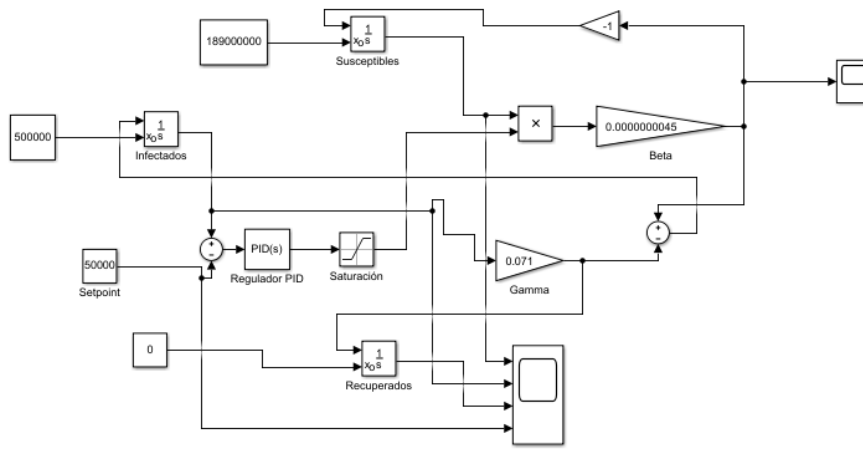


Figura E.4: Diagrama de bloques en Simulink para el modelo SIR con regulador PID.

Modelo SEIR representado en la figura E.5, extensión del modelo SIR que incluye una fase de exposición, donde los individuos están infectados pero no son contagiosos durante un período de incubación.

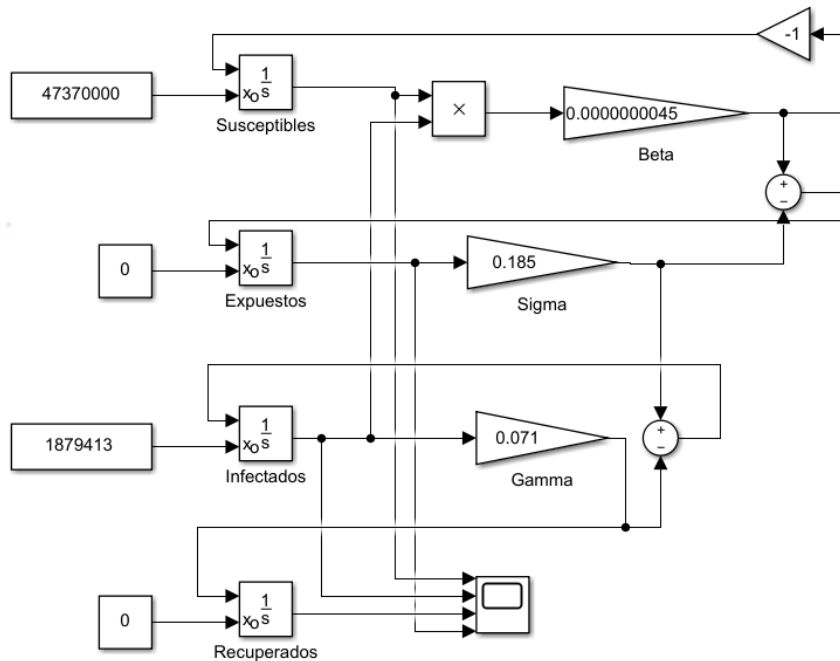


Figura E.5: Diagrama de bloques en Simulink para el modelo SEIR.

Modelo SIRV representado en la figura E.6, modelo SIR que incorpora la vacunación como una vía adicional para pasar de susceptibles a inmunes sin pasar por la infección.

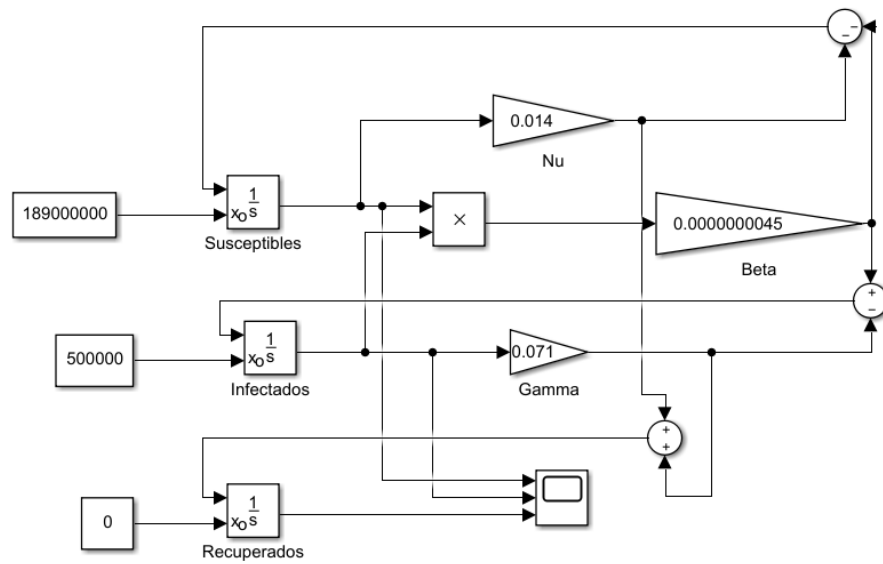


Figura E.6: Diagrama de bloques en Simulink para el modelo SIRV.

Modelo SEIRV representado en la figura E.7, modelo SEIR que añade un compartimento para vacunados, permitiendo estudiar el impacto de la vacunación en la dinámica de la epidemia.

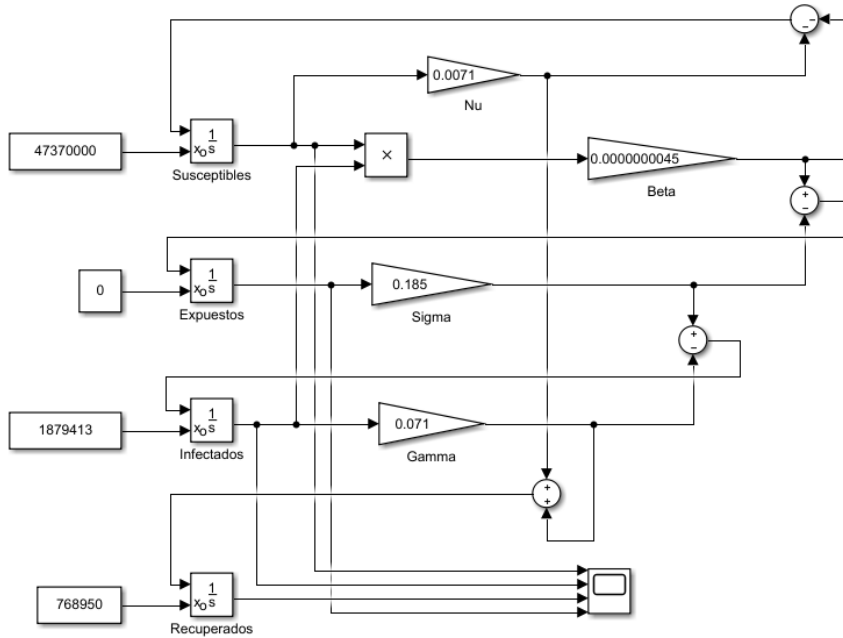


Figura E.7: Diagrama de bloques en Simulink para el modelo SEIRV.

E.2. Script MATLAB

A continuación, se muestran los fragmentos clave del script desarrollado en MATLAB que complementa el modelo SIR con control PID.

En la Figura E.8 se presentan las líneas del script donde se definen los parámetros del modelo epidemiológico, como las tasas de transmisión y recuperación, así como las condiciones iniciales para cada una de las variables del sistema (Susceptibles, Infectados, Recuperados, etc.).

Por otro lado, la Figura E.9 muestra la implementación del controlador PID en el script. En este fragmento se define el error como la diferencia entre la referencia deseada y la cantidad de infectados, y se calcula la señal de control que actúa sobre el sistema para minimizar dicho error. Esta lógica permite estudiar el comportamiento del sistema bajo distintas configuraciones del controlador.

Estas imágenes reflejan cómo el uso de código MATLAB permite ajustar dinámicamente los parámetros del sistema y aplicar estrategias de control sin modificar directamente los modelos de Simulink.

```
% Parámetros del regulador PID
Kp = 0.01;
Ki = 0.001;
Kd = 0.01;
error_prev = 0;
error_int = 0;
```

Figura E.8: Definición de parámetros en el script MATLAB.

```
% Control PID
error = I_setpoint - I;
error_int = error_int + error * dt;
d_error = (error - error_prev) / dt;
beta = beta_base + Kp * error + Ki * error_int + Kd * d_error;
beta = max(beta, 0); % Limitar a valores positivos
```

Figura E.9: Implementación del regulador PID en el script MATLAB.

E.3. Diseño arquitectónico

El sistema desarrollado consta de tres componentes principales:

- **Modelos en Simulink:** cada modelo epidemiológico se implementa mediante diagramas de bloques que representan las ecuaciones diferenciales del sistema. Esto permite simular la dinámica de la epidemia de manera visual e intuitiva.
- **Scripts en MATLAB:** se utiliza para la implementación del controlador PID en el modelo SIR, actuando sobre la tasa de transmisión para mitigar la propagación.
- **Interfaz gráfica en App Designer:** diseñada para permitir al usuario modificar parámetros epidemiológicos y del controlador, ejecutar simulaciones y visualizar resultados de forma interactiva mediante gráficas dinámicas.

En cuanto a la aplicación ha sido desarrollada en MATLAB App Designer, lo que proporciona una arquitectura de programación basada en eventos y

de programación declarativa para la interfaz de usuario, donde la interfaz gráfica y la lógica de control están encapsuladas en un único archivo ‘mlapp’.

El diseño se estructura en los siguientes componentes principales:

- **Interfaz de usuario (UI):** compuesta por elementos gráficos como botones, menús desplegables, campos de entrada numérica y ejes para visualización. Permite al usuario seleccionar el modelo epidemiológico, definir los parámetros iniciales, ejecutar la simulación y visualizar los resultados.
- **Controladores (callbacks):** funciones asociadas a eventos de la interfaz gráfica, tales como la selección del modelo o la pulsación del botón de simulación. Estas gestionan la lógica de interacción entre la UI y el núcleo del programa.
- **Módulo de simulación:** implementado mediante funciones anónimas de MATLAB y el solucionador `ode45`, permite resolver los distintos modelos epidemiológicos (SI, SIS, SIR, SEIR, SIRV, SEIRV) a partir de los parámetros definidos por el usuario.
- **Visualización de resultados:** Utiliza la función `plot` para representar la evolución temporal de las variables del modelo. Las leyendas y los ejes se ajustan dinámicamente según el modelo seleccionado.

Diagrama de clases. La clase principal `aplicacion_modelos` agrupa todos los elementos de la interfaz de usuario (UI), así como las funciones necesarias para gestionar eventos y simular los modelos epidemiológicos. Se emplea un enfoque orientado a eventos, donde la lógica de simulación se encuentra encapsulada en funciones de devolución de llamada (callbacks). El diagrama de clases se puede ver en la figura [E.10](#).

aplicación_modelos
<ul style="list-style-type: none"> + UIFigure : matlab.ui.Figure + UIAxes : matlab.ui.control.UIAxes + SusceptiblesEditFieldLabel : matlab.ui.control.Label + SusceptiblesEditField : matlab.ui.control.NumericEditField + InfectadosEditFieldLabel : matlab.ui.control.Label + InfectadosEditField : matlab.ui.control.NumericEditField + RecuperadosEditFieldLabel : matlab.ui.control.Label + RecuperadosEditField : matlab.ui.control.NumericEditField + ExpuestosEditFieldLabel : matlab.ui.control.Label + ExpuestosEditField : matlab.ui.control.NumericEditField + TasadetransmisinEditFieldLabel : matlab.ui.control.Label + TasadetransmisinEditField : matlab.ui.control.NumericEditField + TasaderecuperacinEditFieldLabel : matlab.ui.control.Label + TasaderecuperacinEditField : matlab.ui.control.NumericEditField + TasadeincubacionEditFieldLabel : matlab.ui.control.Label + TasadeincubacionEditField : matlab.ui.control.NumericEditField + TasadevacunacinEditFieldLabel : matlab.ui.control.Label + TasadevacunacinEditField : matlab.ui.control.NumericEditField + ModeloDropDownLabel : matlab.ui.control.Label + ModeloDropDown : matlab.ui.control.DropDown + SimularButton : matlab.ui.control.Button + TiempoEditFieldLabel : matlab.ui.control.Label + TiempoEditField : matlab.ui.control.NumericEditField
<ul style="list-style-type: none"> + aplicacion_modelos() + delete(app) --- - ModeloDropDownValueChanged(app, event) - SimularButtonPushed(app, event) - createComponents(app)

Figura E.10: Diagrama de clases de la aplicación realizada.

Especificación de Requisitos

F.1. Diagrama de casos de uso

En la Figura F.1 se presenta el diagrama de casos de uso de la aplicación realizada, en el que se describen las distintas interacciones que pueden realizar los usuarios, dependiendo de si tienen MATLAB instalado o no.

- El **usuario con MATLAB** puede abrir directamente la aplicación desde el archivo `.mlapp`, seleccionar el modelo epidemiológico, introducir los parámetros, ejecutar la simulación y visualizar los resultados, también se puede simular después de elegir el modelo ya que .
- El **usuario sin MATLAB** debe primero instalar los *MATLAB Run-times* mediante el instalador `myAppInstaller_web.exe`, y posteriormente ejecutar la aplicación compilada `.exe`. Una vez completados estos pasos, podrá seguir el mismo flujo funcional que el usuario con MATLAB.
- Los casos de uso principales compartidos por ambos tipos de usuario son: *Seleccionar modelo*, *Introducir parámetros*, *Simular* y *Visualización de resultados*.

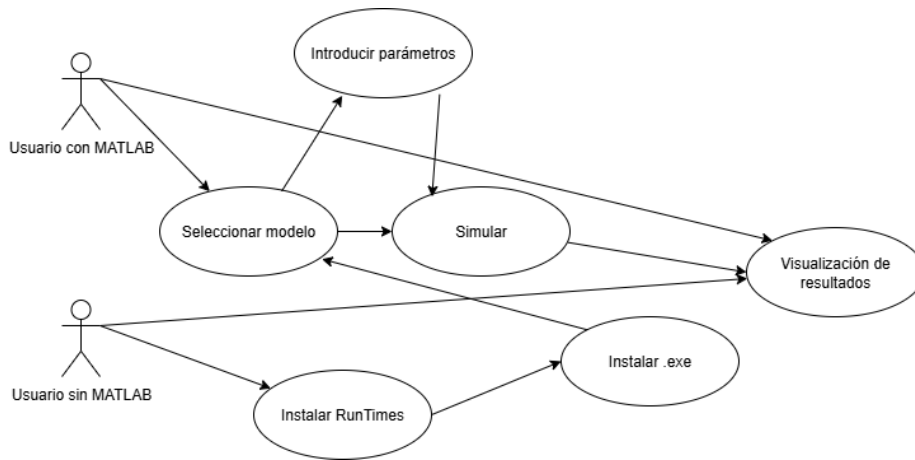


Figura F.1: Diagrama de casos de uso de la aplicación realizada.

F.2. Explicación casos de uso

A continuación en las diferentes tablas (F.1)(F.2)(F.3)(F.4)(F.5)(F.6) se detalla la explicación de los casos de usos utilizados en el diagrama de casos de uso del apartado anterior.

CU-1	Seleccionar modelo
Versión	1.0
Autor	Lucía Segura Benito
Requisitos asociados	RF-01
Descripción	El usuario selecciona el tipo de modelo epidemiológico a utilizar: SI, SIS, SIR, SEIR, SIRV o SEIRV.
Precondición	La aplicación debe estar abierta.
Acciones	<ol style="list-style-type: none"> 1. El usuario abre el menú desplegable de modelos. 2. Selecciona uno de los modelos disponibles. 3. El sistema guarda la selección para la simulación posterior.
Postcondición	Modelo seleccionado queda registrado en la aplicación.
Excepciones	No se produce ninguna si el desplegable está bien cargado.
Importancia	Alta

Tabla F.1: CU-1: Seleccionar modelo.

CU-2	Introducir parámetros
Versión	1.0
Autor	Lucía Segura Benito
Requisitos asociados	RF-02
Descripción	El usuario introduce los valores numéricos requeridos para simular el modelo seleccionado.
Precondición	Un modelo debe estar previamente seleccionado.
Acciones	<ol style="list-style-type: none">1. El usuario accede a los campos de parámetros.2. Introduce valores que se piden dependiendo del modelo seleccionado.3. El sistema almacena los parámetros para usarlos en la simulación.
Postcondición	Parámetros definidos correctamente para la simulación.
Excepciones	Parámetros vacíos o no válidos (el sistema muestra mensaje de error).
Importancia	Alta

Tabla F.2: CU-2: Introducir parámetros.

CU-3	Simular modelo
Versión	1.0
Autor	Lucía Segura Benito
Requisitos asociados	RF-03, RF-04
Descripción	El usuario ejecuta la simulación del modelo epidemiológico seleccionado utilizando los parámetros introducidos.
Precondición	El modelo debe estar seleccionado y los parámetros correctamente definidos.
Acciones	<ol style="list-style-type: none"> 1. El usuario pulsa el botón "Simular". 2. El sistema verifica que los datos de entrada sean válidos. 3. El sistema ejecuta el modelo epidemiológico seleccionado. 4. Se muestran los resultados en pantalla mediante gráficas.
Postcondición	Los resultados de la simulación se visualizan en la interfaz de usuario.
Excepciones	Parámetros incompletos o inválidos (el sistema muestra un mensaje de error).
Importancia	Alta

Tabla F.3: CU-3: Simular modelo.

CU-4	Visualización de resultados
Versión	1.0
Autor	Lucía Segura Benito
Requisitos asociados	RF-05
Descripción	La aplicación muestra al usuario los resultados de la simulación mediante gráficos.
Precondición	Simulación completada exitosamente.
Acciones	<ol style="list-style-type: none">1. El sistema genera gráficos de las variables.2. Los gráficos se muestran en la interfaz.
Postcondición	Gráficas epidemiológicas visibles para el usuario.
Excepciones	Error al renderizar gráficos (muy raro).
Importancia	Media

Tabla F.4: CU-4: Visualización de resultados.

CU-5	Instalar RunTimes
Versión	1.0
Autor	Lucía Segura Benito
Requisitos asociados	RF-06
Descripción	El usuario sin MATLAB instala los MATLAB Runtime necesarios para ejecutar la aplicación compilada.
Precondición	No disponer de MATLAB instalado. Tener el archivo <code>myAppInstaller_web.exe</code> .
Acciones	<ol style="list-style-type: none"> 1. El usuario ejecuta el archivo <code>myAppInstaller_web.exe</code>. 2. El instalador descarga e instala los runtimes de MATLAB (MCR).
Postcondición	Los runtimes de MATLAB quedan correctamente instalados.
Excepciones	Error en la instalación por permisos, red o espacio en disco.
Importancia	Alta

Tabla F.5: CU-5: Instalar RunTimes.

CU-6	Instalar .exe
Versión	1.0
Autor	Lucía Segura Benito
Requisitos asociados	RF-07
Descripción	El usuario sin MATLAB ejecuta el archivo compilado de la aplicación tras instalar los runtimes.
Precondición	MATLAB Runtime (MCR) instalado previamente.
Acciones	<ol style="list-style-type: none"> 1. El usuario ejecuta <code>aplicacion_modelos.exe</code>. 2. Se lanza la aplicación de forma autónoma, sin requerir MATLAB.
Postcondición	Aplicación ejecutándose de forma normal.
Excepciones	Error al lanzar el .exe, falta de runtimes o permisos insuficientes.
Importancia	Alta

Tabla F.6: CU-6: Instalar .exe.

Anexo de sostenibilidad curricular

Este proyecto está centrado en el modelado matemático de epidemias mediante modelos compartimentales y su implementación en **MATLAB** y **Simulink**, se vincula de forma directa con diversos ODS¹ establecidos en [Organización de las Naciones Unidas, 2015] la Agenda 2030 de las Naciones Unidas.

ODS relacionados

- **ODS 3: Salud y Bienestar.** El proyecto contribuye al entendimiento de la propagación de enfermedades infecciosas y la evaluación de estrategias de control como la vacunación y las cuarentenas, aspectos clave para mejorar la resiliencia de los sistemas sanitarios ante futuras pandemias [Organización Mundial de la Salud, 2020].
- **ODS 4: Educación de Calidad.** La creación de una aplicación interactiva para la visualización y manipulación de los modelos permite democratizar el acceso al conocimiento técnico, facilitando la comprensión de dinámicas epidemiológicas incluso a personas sin formación especializada. Esto potencia el aprendizaje autónomo y el desarrollo de competencias digitales [UNESCO, 2020].
- **ODS 10: Reducción de las Desigualdades.** Al permitir la simulación y análisis de medidas sanitarias en distintos escenarios, el trabajo también reflexiona sobre cómo las epidemias afectan de manera desigual a distintas poblaciones. El desarrollo de herramientas accesibles

¹Objetivos de Desarrollo Sostenible.

y adaptables puede contribuir a reducir brechas tecnológicas y de conocimiento [World Health Organization, 2021].

Contribuciones a la sostenibilidad curricular

El trabajo fomenta diversas competencias clave para el desarrollo sostenible, alineadas con los principios de la sostenibilidad curricular en educación superior [Lozano et al., 2017]. Entre ellas destacan:

- **Pensamiento crítico y sistémico:** al modelar fenómenos complejos como la dinámica epidémica desde un enfoque cuantitativo e interdisciplinar.
- **Innovación responsable:** mediante el diseño de herramientas digitales orientadas a la educación y la toma de decisiones en salud pública.
- **Conciencia ética y social:** al abordar la dimensión humana y desigual de las pandemias, así como la necesidad de respuestas colaborativas y equitativas.

En conjunto, este trabajo de fin de grado no solo aporta un enfoque técnico y analítico al estudio de epidemias, sino que también contribuye a una visión integral de la sostenibilidad, reconociendo la interdependencia entre ciencia, tecnología, salud pública y justicia social. De este modo, el TFG se alinea con los principios de una educación superior comprometida con la Agenda 2030 y con la formación de profesionales capaces de abordar los desafíos globales desde una perspectiva ética y sostenible.

Bibliografía

- [Ada Health, 2025] Ada Health (2025). ¿cuántos días dura el covid? una guía completa. <https://ada.com/es/covid/cuantos-dias-dura-el-covid/>. Consultado el 2 de junio de 2025.
- [Centers for Disease Control and Prevention, 2025] Centers for Disease Control and Prevention (2025). Measles cases and outbreaks. <https://www.cdc.gov/measles/data-research/index.html>. Consultado el 2 de junio de 2025.
- [Commons, 2013] Commons, C. (2013). Licencia creative commons atribución-nocomercial 4.0 internacional (cc by-nc 4.0). <https://creativecommons.org/licenses/by-nc/4.0/>. Accedido el 6 de julio de 2025.
- [Creative Commons, 2013] Creative Commons (2013). Licencia Creative Commons Atribución-NoComercial-CompartirIgual 4.0 Internacional (CC BY-NC-SA 4.0). <https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>. Consultado el 6 de julio de 2025.
- [Data Commons, 2025] Data Commons (2025). España – demografía. https://datacommons.org/place/country/ESP?utm_medium=explore&prop=count&popt=Person&hl=es. Consultado el 2 de junio de 2025.
- [Datosmacro.com, 2025] Datosmacro.com (2025). Estados unidos - población 1963. <https://datosmacro.expansion.com/demografia/poblacion/usa?anio=1963>. Recuperado el 2 de junio de 2025.
- [de Sanidad, 2025] de Sanidad, M. (2025). Informes de situación publicados durante la crisis sanitaria. <https://www.sanidad.gob.es/areas/>

- [alertasEmergenciasSanitarias/alertasActuales/nCov/evolucion/situacion.htm](#). Consultado el 2 de junio de 2025.
- [EFE, 2025] EFE, A. (2025). Cinco años del primer caso de covid en españa: un turista alemán, en la gomera. <https://www.20minutos.es/noticia/5678158/0/cinco-anos-primer-caso-covid-19-espana-turista-aleman-la-gomera/>. Consultado el 2 de junio de 2025.
- [España,] España. Ley orgánica 3/2018, de 5 de diciembre, de protección de datos personales y garantía de los derechos digitales. *Boletín Oficial del Estado*, núm. 294, 6 de diciembre de 2018, pp. 119788–119857.
- [Instituto Nacional de Estadística (INE), 2024] Instituto Nacional de Estadística (INE) (2024). Encuesta trimestral de coste laboral (etcl). <https://www.ine.es/dynt3/inebase/index.htm?padre=517&capsel=518>. Accedido el 29 de junio de 2025.
- [Khorrami et al., 2023] Khorrami, Z., Hasankhani, M. B., Khezri, M., Jafari-Khounigh, A., Jahani, Y., and Sharifi, H. (2023). Trends and projection of incidence, mortality, and disability-adjusted life years of hiv in the middle east and north africa (1990–2030). *Scientific Reports*, 13(1):13859.
- [Lauer et al., 2020] Lauer, S. A., Grantz, K. H., Bi, Q., Jones, F. K., Zheng, Q., Meredith, H. R., Azman, A. S., Reich, N. G., and Lessler, J. (2020). The incubation period of coronavirus disease 2019 (covid-19) from publicly reported confirmed cases: estimation and application. *Annals of internal medicine*, 172(9):577–582.
- [Lozano et al., 2017] Lozano, R., Ceulemans, K., and Scarff Seatter, C. (2017). Teaching organisational change management for sustainability: Designing and delivering a course at the university of leeds to better prepare future sustainability change agents. *Journal of Cleaner Production*, 172:4357–4366.
- [MathWorks, 2025a] MathWorks (2025a). Matlab. <https://es.mathworks.com/products/matlab.html>. Recuperado el 6 de junio de 2025.
- [MathWorks, 2025b] MathWorks (2025b). System requirements for matlab and simulink. <https://www.mathworks.com/support/requirements/matlab-system-requirements.html>. Recuperado el 6 de junio de 2025.
- [MathWorks, nd] MathWorks (n.d.). Matlab para estudiantes. <https://es.mathworks.com/products/matlab/student.html>. Recuperado el 8 de junio de 2025.

- [Microsoft, nda] Microsoft (n.d.a). Cómo obtener windows 11 para un pc compatible. <https://www.microsoft.com/es-es/windows/get-windows-11>. Recuperado el 8 de junio de 2025.
- [Microsoft, ndb] Microsoft (n.d.b). Office 365 personal. <https://www.microsoft.com/es-es/microsoft-365/p/office-365-personal/cfq7ttc0k5dm>. Recuperado el 6 de junio de 2025.
- [Ministerio de Hacienda de España, 2024] Ministerio de Hacienda de España (2024). Tablas de amortización fiscal (impuesto de sociedades). <https://www.agenciatributaria.es>. Establecen un coeficiente lineal máximo del 33 % anual para equipos informáticos, equivalente a 3 años.
- [Ministerio de Sanidad, 2021] Ministerio de Sanidad (2021). Informe de actividad diario. gestión integral de vacunación frente al covid-19 en España. https://www.sanidad.gob.es/profesionales/saludPublica/ccayes/alertasActual/nCov/documentos/Informe_GIV_comunicacion_20210115.pdf. Consultado el 2 de junio de 2025.
- [Ministerio de Sanidad, 2025] Ministerio de Sanidad (2025). Histórico de coberturas de vacunación frente a la covid-19. <https://www.sanidad.gob.es/areas/promocionPrevencion/vacunaciones/coberturas/historico/vacunacionCovid19/home.htm>. Consultado el 2 de junio de 2025.
- [Organización de las Naciones Unidas, 2015] Organización de las Naciones Unidas (2015). *Transformar nuestro mundo: la Agenda 2030 para el Desarrollo Sostenible*. Naciones Unidas. Consultado el 6 de junio de 2025.
- [Organización Mundial de la Salud, 2020] Organización Mundial de la Salud (2020). *A World in Disorder: Global Preparedness Monitoring Board Annual Report 2020*. OMS. Consultado el 6 de junio de 2025.
- [Organización Panamericana de la Salud, sf] Organización Panamericana de la Salud (s.f.). Sarampión. <https://www.paho.org/es/temas/sarampion#:~:text=No%20hay%20tratamiento%20espec%C3%ADfico%20para,en%202%20o%203%20semanas>. Recuperado el 1 de junio de 2025.
- [Pollock et al., 2023] Pollock, E. D., Clay, P. A., Kreisel, K. M., and Spicknall, I. H. (2023). Estimated incidence and prevalence of gonorrhea in the united states, 2006–2019. *Sexually transmitted diseases*, 50(4):188–195.
- [Project Management Institute, 2017] Project Management Institute (2017). *Guía de los Fundamentos para la Dirección de Proyectos (Guía PMBOK)*. PMI, 6 edition.

- [Shakiba et al., 2021] Shakiba, E., Ramazani, U., Mardani, E., Rahimi, Z., Nazar, Z. M., Najafi, F., and Moradinazar, M. (2021). Epidemiological features of hiv/aids in the middle east and north africa from 1990 to 2017. *International journal of STD & AIDS*, 32(3):257–265.
- [Solomon, 2019] Solomon, C. G. (2019). Peter m. strebel, mb, ch. b., mph, and walter a. orenstein, md. *Vaccine*, 500:1200.
- [UNESCO, 2020] UNESCO (2020). *Education in a Post-COVID World: Nine Ideas for Public Action*. International Commission on the Futures of Education. Consultado el 6 de junio de 2025.
- [Unión Europea,] Unión Europea. Reglamento (ue) 2016/679 del parlamento europeo y del consejo, de 27 de abril de 2016, relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos, y por el que se deroga la directiva 95/46/ce (reglamento general de protección de datos). *Diario Oficial de la Unión Europea*, L119, 4.5.2016, pp. 1–88.
- [World Health Organization, 2021] World Health Organization (2021). Covid-19 and health inequality: A global perspective. Consultado el 6 de junio de 2025.
- [Álvaro García, 2020] Álvaro García (2020). Coronavirus: ¿qué es el número r_0 y cuál es su papel en la desescalada? https://as.com/diarioas/2020/05/05/actualidad/1588680293_938747.html. Consultado el 2 de junio de 2025.