



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

9 de junio 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	7
A.4. Viabilidad legal	9
Apéndice B Documentación	11
B.1. Requisitos software y hardware para ejecutar el proyecto . . .	11
B.2. Instalación / Puesta en marcha	12
B.3. Manuales y/o demostraciones prácticas	14
Apéndice C Manual del investigador	21
C.1. Estructura de directorios	21
C.2. Compilación, instalación y ejecución del proyecto	22
C.3. Pruebas del sistema	23
C.4. Instrucciones para la modificación o mejora del proyecto . . .	24
Apéndice D Descripción de adquisición y tratamiento de datos	25
D.1. Descripción formal de los datos	25
D.2. Descripción clínica de los datos	26
D.3. Información relevante de los simulaciones	27

Apéndice E Manual de especificación del diseño	29
E.1. Planos	29
E.2. Diseño arquitectónico	33
Apéndice F Estudio experimental	35
Apéndice G Anexo de sostenibilidad curricular	37
Bibliografía	39

Índice de figuras

B.1. Instrucción de MATLAB para el paso 3	13
B.2. Instrucción de MATLAB para abrir la aplicación	13
B.3. Interfaz inicial de la aplicación	14
B.4. Interfaz inicial de la aplicación	15
B.5. Elección del modelo, en est caso SIR	17
B.6. Comprobación de que salen los datos por defecto	18
B.7. Simulación del modelo SIR en la aplicación	18
B.8. Elección del modelo, en este caso SEIRV	19
B.9. Simulación del modelo SEIRV en la aplicación con los datos predefinidos	20
E.1. Diagrama de bloques en Simulink para el modelo SI	30
E.2. Diagrama de bloques en Simulink para el modelo SIS	30
E.3. Diagrama de bloques en Simulink para el modelo SIR	31
E.4. Diagrama de bloques en Simulink para el modelo SEIR	31
E.5. Diagrama de bloques en Simulink para el modelo SIRV	32
E.6. Diagrama de bloques en Simulink para el modelo SEIRV	33

Índice de tablas

A.1. Etiquetas temáticas (labels) utilizadas en la planificación del proyecto	3
A.2. Costes de hardware imputados al TFG	7
A.3. Costes de software imputados al TFG	8
A.4. Costes de mano de obra imputados al TFG	8
A.5. Resumen económico total del TFG	8
B.1. Requisitos mínimos y recomendados para la ejecución del proyecto	12

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 – 6 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)

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)

Milestone 6: Redacción en LaTeX

Periodo: 7 de mayo – 8 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 junio)

- 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 – 8 de junio)

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

Se recoge en la Tabla A.2 el coste del equipo informático utilizado durante el desarrollo del TFG. Aunque el equipo ya estaba disponible, se ha imputado un coste proporcional en función del uso específico para el proyecto

Concepto	Descripción	Coste total (€)	Coste imputado (€)
Portátil	Asus Vivobook X409JB	650	350

Tabla A.2: Costes de hardware imputados al TFG

Costes de Software

Se incluyen en la Tabla A.3 las licencias de software necesarias para el desarrollo del TFG, como MATLAB, Simulink, Office y el sistema operativo Windows. 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 ²	150	50 %	75
Office	Office 365 (1 año) ³	69	100 %	69
MATLAB	Educativa estudiante ⁴	150	100 %	150
Simulink	Incluida en MATLAB	0	100 %	0

Tabla A.3: Costes de software imputados al TFG

Costes de Mano de Obra

Este apartado refleja en la Tabla A.4 el valor estimado del trabajo realizado por el estudiante desde una perspectiva profesional. Se ha considerado un total de 360 horas de trabajo, valoradas a razón de 20/hora, correspondiente a la tarifa media de un ingeniero informático junior [Glassdoor, nd]. Aunque no representa un coste real, sirve para valorar el esfuerzo implicado.

Concepto	Detalle	Horas	Precio/hora (€)	Coste total (€)
Desarrollo app	Ingeniero	360	20	7.200

Tabla A.4: Costes de mano de obra imputados al TFG

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 imputado (€)
Hardware	450
Software	294
Mano de obra	7.200
Total general	7.944

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 (como MIT o GPL), 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.

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, 2016] y la Ley Orgánica 3/2018 (LOPDGDD) [España, 2018].

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 UBU⁵ 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.

⁵Universidad 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.

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

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

B.2. Instalación / Puesta en marcha

Este apartado explica cómo instalar el software necesario y cómo ejecutar la aplicación y simulaciones desarrolladas.

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 MATLABz 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.

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

5. Abrir MATLAB desde el menú de inicio y comprobar su funcionamiento.

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.1](#), escribir en la ventana de comandos

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

Figura B.1: 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 desde o ejecutar en la ventana de comandos de MATLAB lo que se ve en la figura [B.2](#)

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

Figura B.2: Instrucción de MATLAB para abrir la aplicación

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

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.3 donde se encuentra.

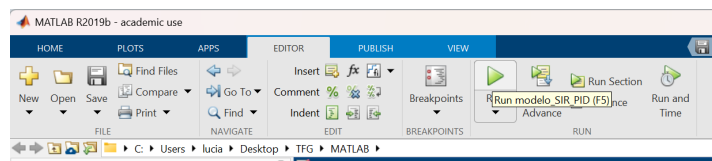


Figura B.3: Interfaz inicial de la aplicación

B.3. Manuales y/o demostraciones prácticas

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.4

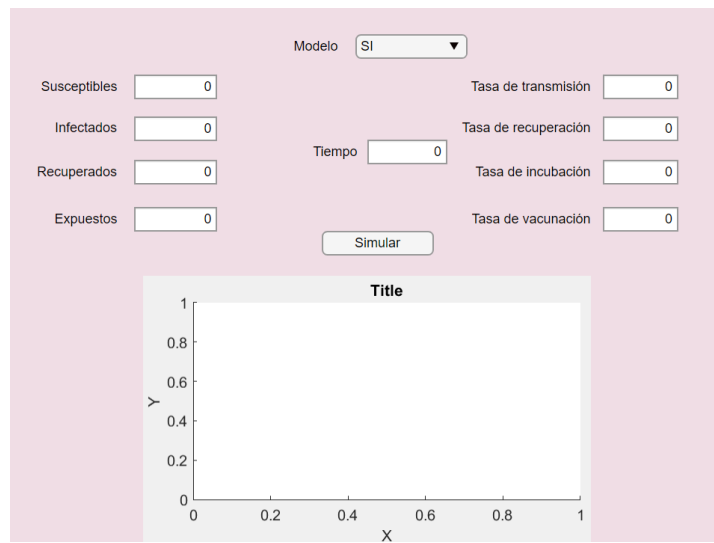


Figura B.4: Interfaz inicial de la aplicación

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.5](#).

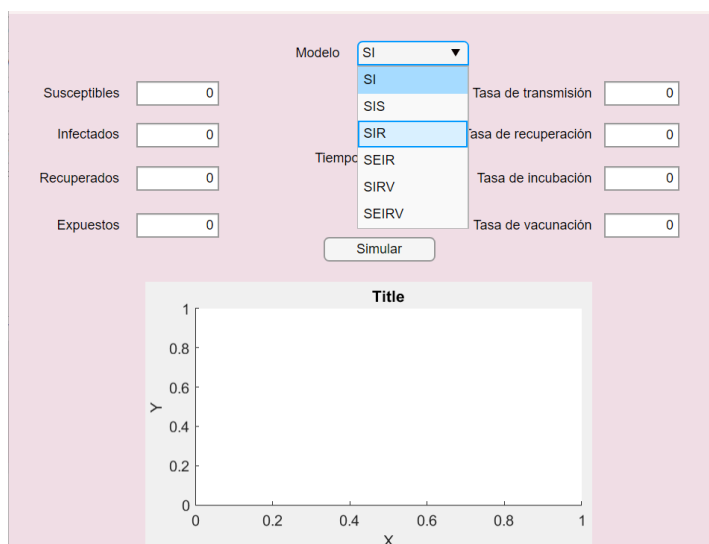


Figura B.5: Elección del modelo, en est caso SIR

2. Verificar que los valores por defecto se han cargado correctamente (se pueden modificar si se desea) [B.6](#).

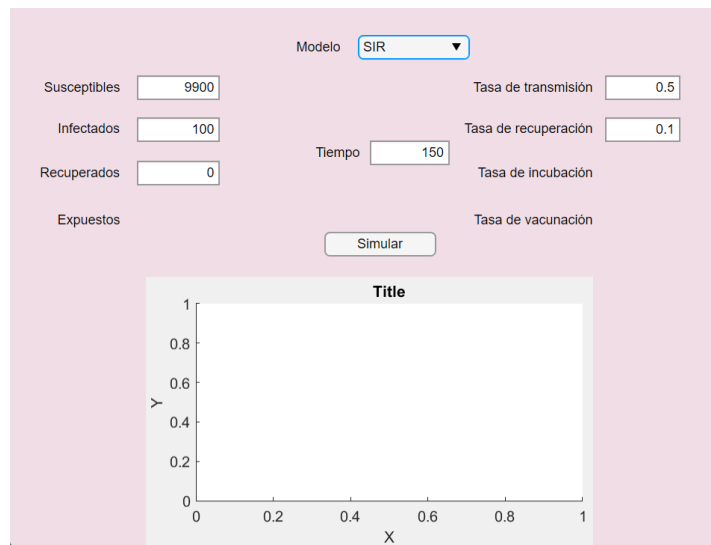


Figura B.6: Comprobación de que salen los datos por defecto

3. Pulsar el botón **Simular**.
 4. Observar la evolución temporal de las tres poblaciones en la gráfica
- B.7**

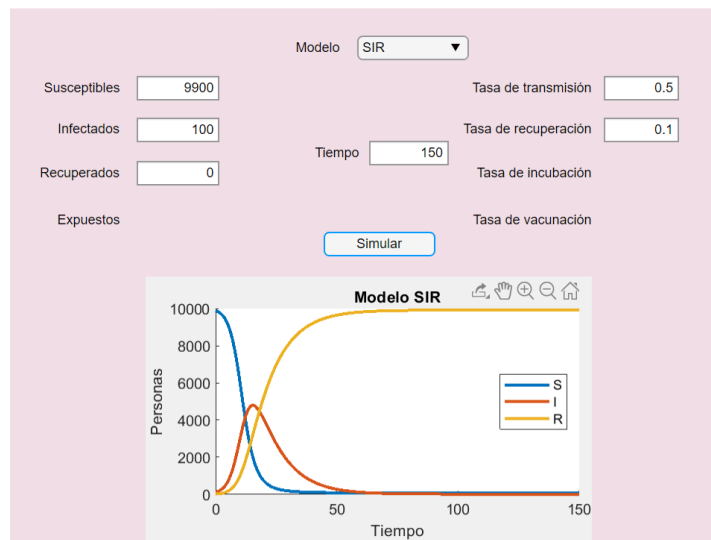


Figura B.7: Simulación del modelo SIR en la aplicación

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.8.

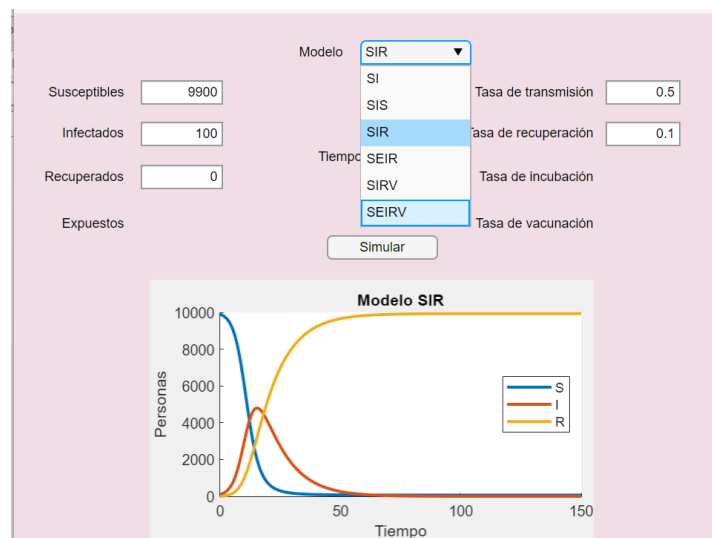


Figura B.8: Elección del modelo, en este caso SEIRV

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.9.

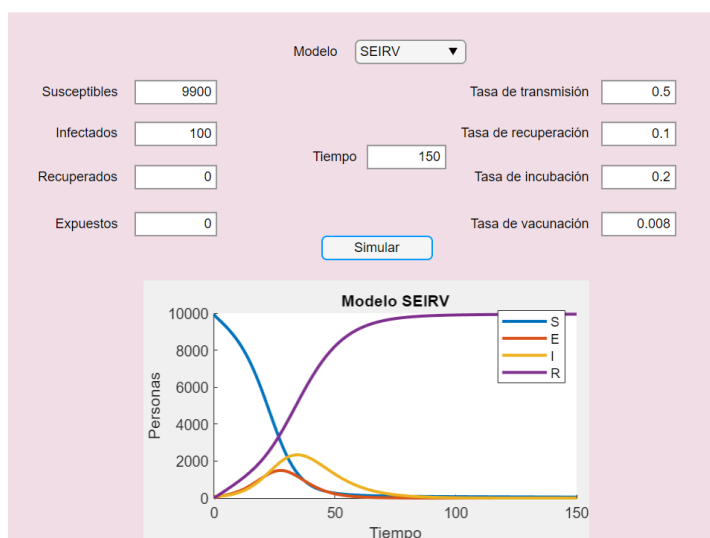


Figura B.9: Simulación del modelo SEIRV en la aplicación con los datos predefinidos

Manual del investigador

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 el archivo de la aplicación desarrollada en App Designer:

- `aplicacion_modelos.mlapp`: interfaz principal para ejecutar simulaciones.

- **Carpeta bibliografía**

Documentos y artículos científicos en .pdf utilizados para entender y fundamentar los modelos epidemiológicos.

- **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 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 recogen únicamente algunas consideraciones relevantes desde el punto de vista del desarrollo:

- El proyecto ha sido desarrollado en **MATLAB R2019b**. Es recomendable utilizar esta misma versión o una superior compatible con los archivos generados.

- Se requiere disponer de los siguientes componentes:
 - MATLAB (R2019b o superior)
 - Simulink
 - Control System Toolbox
 - App Designer
- Los modelos se encuentran en la carpeta `/Simulink/modelos_Simulink/` con extensión `.slx`, y pueden abrirse directamente desde el entorno MATLAB haciendo doble clic sobre ellos.
- Los scripts inc el controlador PID implementado sobre el modelo SIR, se encuentran en la carpeta `/MATLAB/` con extensión `.m`.
- La aplicación se encuentra en `/aplicación/aplicacion_modelos.mlapp`
- Para cualquier detalle adicional sobre configuración del entorno, instalación o estructura de archivos, se remite al Apéndice B.

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.
- **Portabilidad:** Para permitir su uso fuera de MATLAB, podría considerarse una migración parcial a otros lenguajes como Python o una versión web del visualizador.

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. 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 .

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. Planos

Aunque no se han elaborado planos arquitectónicos formales, se han utilizado diagramas de bloques en **Simulink** para representar los distintos modelos epidemiológicos (SI, SIS, SIR, SEIR, SIRV y SEIR). 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.

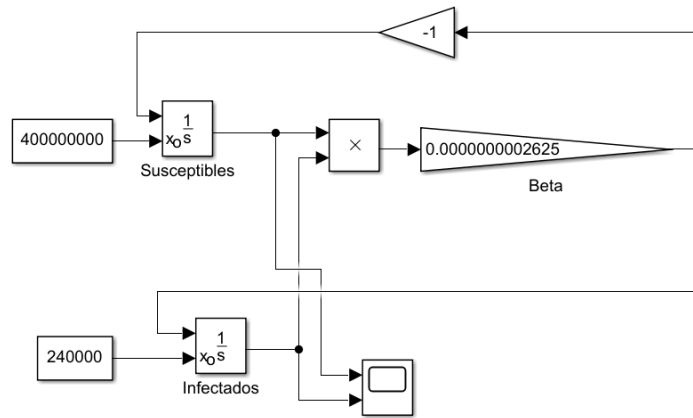


Figura E.1: Diagrama de bloques en Simulink para el modelo SI

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

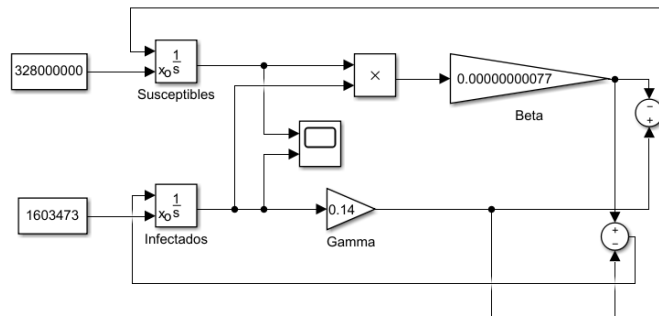


Figura E.2: Diagrama de bloques en Simulink para el modelo SIS

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

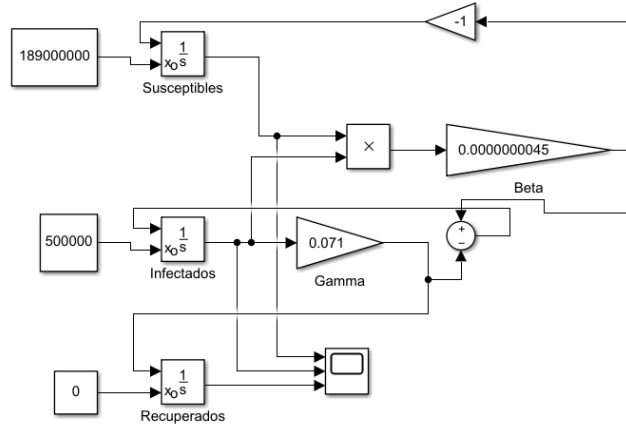


Figura E.3: Diagrama de bloques en Simulink para el modelo SIR

Modelo SEIR representado en la figura E.4, extensión del modelo SIR que incluye una fase de exposición, donde los individuos están infectados pero no son contagiosos durante un período de incubación.

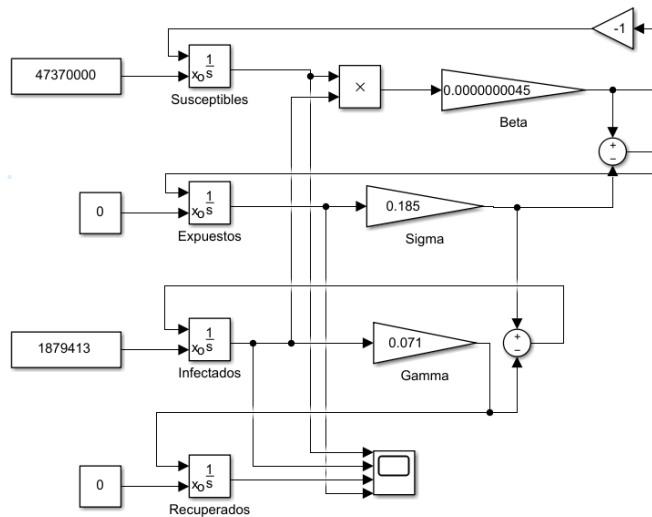


Figura E.4: Diagrama de bloques en Simulink para el modelo SEIR

Modelo SIRV representado en la figura E.5, modelo SIR que incorpora la vacunación como una vía adicional para pasar de susceptibles a inmunes sin pasar por la infección.

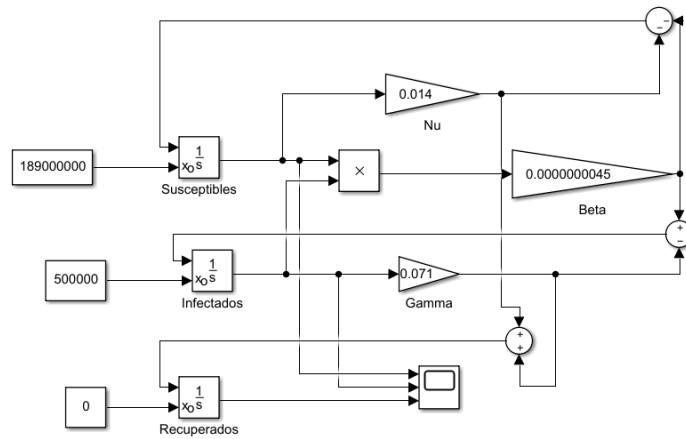


Figura E.5: Diagrama de bloques en Simulink para el modelo SIRV

Modelo SEIRV representado en la figura E.6, modelo SEIR que añade un compartimento para vacunados, permitiendo estudiar el impacto de la vacunación en la dinámica de la epidemia.

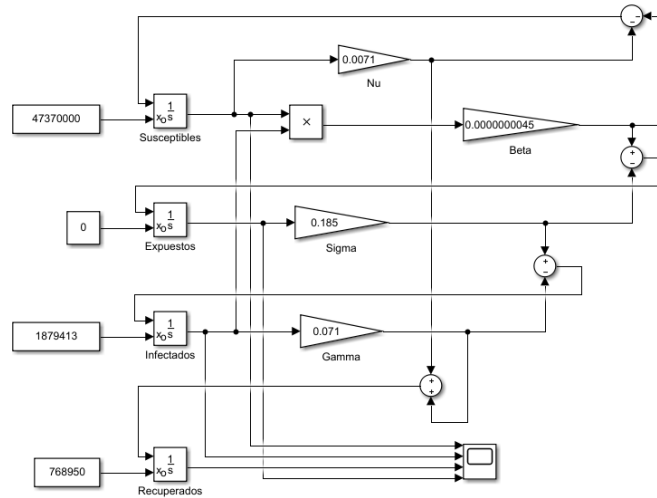


Figura E.6: Diagrama de bloques en Simulink para el modelo SEIRV

E.2. 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.

No se han desarrollado diagramas de clases ni de despliegue, dado que el proyecto no utiliza programación orientada a objetos ni arquitectura distribuida. La estructura es modular, basada en scripts, modelos de Simulink y la aplicación gráfica integrada.

Apéndice F

Estudio experimental

La información correspondiente al estudio experimental, parametrización, resultados obtenidos y su análisis se encuentra desarrollada en la sección de Resultados y Discusión de la memoria. Por tanto, no se incluye de nuevo en este apéndice.

Apéndice *G*

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 la Agenda 2030 de las Naciones Unidas [[Organización de las Naciones Unidas, 2015](#)].

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

- [España, 2018] España (2018). 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. Consultado el 6 de junio de 2025.
- [Glassdoor, nd] Glassdoor (n.d.). Ingeniero informático junior – sueldos en España. Recuperado el 8 de junio de 2025.
- [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. Recuperado el 6 de junio de 2025.
- [MathWorks, 2025b] MathWorks (2025b). System requirements for matlab and simulink. Recuperado el 6 de junio de 2025.
- [MathWorks, nd] MathWorks (n.d.). Matlab para estudiantes. Recuperado el 8 de junio de 2025.
- [Microsoft, nda] Microsoft (n.d.a). Cómo obtener windows 11 para un pc compatible. 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.

- [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.
- [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, 2016] Unión Europea (2016). 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. Consultado el 6 de junio de 2025.
- [World Health Organization, 2021] World Health Organization (2021). Covid-19 and health inequality: A global perspective. Consultado el 6 de junio de 2025.