



## SimuDest – Simulador de destinos para oposiciones

**José Alberto Castillo Jiménez**  
Grado de Ingeniería Informática

**Antoni Oller Arcas**

01/06/2022



Esta obra está sujeta a una licencia de  
Reconocimiento-NoComercial-CompartirIgual  
[3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-sa/3.0/es/)

## FICHA DEL TRABAJO FINAL

<b>Título del trabajo:</b>	SimuDest – Simulador de destinos para oposiciones.
<b>Nombre del autor:</b>	José Alberto Castillo Jiménez
<b>Nombre del consultor:</b>	Antoni Oller Arcas
<b>Fecha de entrega (mm/aaaa):</b>	06/2022
<b>Área del Trabajo Final:</b>	Java EE
<b>Titulación:</b>	<i>Grado de Ingeniería Informática</i>
<b>Resumen del Trabajo:</b>	
<p>SimuDest es una plataforma online destinada a opositores que han superado un proceso de oposición, y que desean saber su plaza asignada antes de que sean publicados los resultados definitivos.</p> <p>Para ello, la aplicación les provee un entorno, que les permite organizarse y realizar una simulación, sobre cuál será el destino que se les asigne en base a sus elecciones, y al puesto obtenido en base a sus resultados en el proceso de oposición.</p> <p>A nivel tecnológico, SimuDest es una aplicación web desarrollada en Java, conforme al modelo de 3 capas y al patrón modelo-vista-controlador. Para su implementación se han usado entre otras las siguientes tecnologías: Spring, Spring Boot, Spring Security, Spring Data JPA, Spring Web MVC, Thymeleaf, Hibernate, PostgreSQL, Bootstrap, Git.</p> <p>A nivel educativo, se han aplicado los conocimientos adquiridos en las diferentes asignaturas del grado, ya sea en materia de planificación del trabajo, análisis y especificación de requisitos, diseño de base de datos, diseño y construcción de código, logarítmica o calidad de código.</p> <p>Los conocimientos aplicados se encuentran especialmente relacionados con las asignaturas pertenecientes al itinerario de ‘Ingeniería del software’.</p> <p>Igualmente a nivel educativo, hay que considerar la experiencia obtenida en el proyecto en cuanto a todas las fases de un proyecto software desde el planteamiento hasta su construcción y puesta en producción.</p>	

**Abstract:**

SimuDest is an online platform designed for candidates who have passed a competition examination process, and who want to know their work place before the final results of the process are published.

To achieve this objective, the application provides an environment, that allows them to organise themselves and simulate the work place that will be assigned to them based on their choices, and the position obtained based on their results in the competition process.

At a technological level, SimuDest is a web application developed in Java, according to the 3-layer model and the model-view-controller pattern. The following technologies, among others, have been used for its implementation: Spring, Spring Boot, Spring Security, Spring Data JPA, Spring Web MVC, Thymeleaf, Hibernate, PostgreSQL, Bootstrap, Git.

At an educational level, the knowledge acquired in the different subjects of the degree has been applied, whether in work planning, requirements analysis and specification, database design, code design and construction, logarithmics or code quality.

The applied knowledge is especially related to the subjects belonging to the 'Software Engineering' itinerary.

It should also be taken into account, the experience gained in the project in terms of all the phases of a software project from its planning to its construction and implementation in production.

**Palabras clave (entre 4 y 8):**

Simulador, Destinos, Plazas, Oposiciones

# Índice

<b>1. Introducción</b> .....	1
<b>1.1 Contexto y justificación del Trabajo</b> .....	1
<b>1.2 Objetivos del Trabajo</b> .....	2
1.2.1 Objetivos funcionales .....	2
1.2.2 Objetivos didácticos .....	3
<b>1.3 Enfoque y método seguido</b> .....	3
<b>1.4 Planificación del Trabajo</b> .....	4
<b>1.5 Breve resumen de productos obtenidos</b> .....	1
<b>1.6 Breve descripción de los otros capítulos de la memoria</b> .....	1
<b>2. Análisis y Diseño</b> .....	2
<b>2.1 Casos de uso</b> .....	2
2.1.1 Actores principales .....	2
2.1.2 Diagrama de casos de uso .....	3
2.1.3 Fichas de casos de uso .....	5
<b>2.2 Diagrama relacional de base de datos</b> .....	14
<b>2.3 Diagrama de clases</b> .....	14
<b>2.4 Diagrama de arquitectura</b> .....	15
2.4.1 Visión global .....	16
2.4.2 Capa de presentación .....	17
2.4.3 Capa de lógica de negocio .....	19
2.4.4 Capa de datos .....	21
<b>2.5 Modelo de pantallas</b> .....	22
2.5.1 Diagrama de navegación .....	22
2.5.2 Prototipos .....	23
<b>3. Implementación del proyecto</b> .....	29
<b>3.1 Propuesta tecnológica</b> .....	29
<b>3.2 Estructura general del proyecto</b> .....	29
<b>3.3 Capa de presentación</b> .....	30
<b>3.4 Capa de lógica de negocio</b> .....	30
<b>3.5 Capa de datos</b> .....	31
<b>3.6 Autenticación</b> .....	31
<b>3.7 Base de datos</b> .....	31
<b>3.8 Servidor de aplicaciones</b> .....	32
<b>3.9 IDE</b> .....	32
<b>3.10 Control de versiones</b> .....	32
<b>4. Conclusiones</b> .....	33
<b>5. Glosario</b> .....	34
<b>6. Bibliografía</b> .....	35
<b>7. Anexos</b> .....	36
<b>7.1 Manual de instalación</b> .....	36
7.1.1 Base de datos .....	36
7.1.2 Aplicación.....	36
<b>7.2 Manual de usuario</b> .....	37
7.2.1 Registro .....	37

7.2.2 Login.....	37
7.2.3 Ventana principal .....	37
7.2.4 Solicitar acceso .....	37
7.2.5 Crear convocatoria .....	38
7.2.7 Listado de opositores.....	38
7.2.8 Validar y rechazar opositores.....	38
7.2.9 Listado de plazas .....	39
7.2.10 Añadir una plaza.....	39
7.2.11 Selección de plazas .....	39
7.2.12 Consulta del resultado de la simulación .....	40

# 1. Introducción

## 1.1 Contexto y justificación del Trabajo

Los procesos selectivos de las administraciones públicas terminan con la asignación de las plazas ofertadas. En este punto, se tiene una serie de plazas ofertadas con distintas condiciones y localizaciones, y una lista de opositores ordenada por nota. Estos opositores deberán entonces elegir las plazas deseadas en orden de prioridad. Posteriormente, al término del plazo establecido, se publicarán las plazas asignadas a cada opositor, y estos tendrán 30 días naturales para tomar posesión de la plaza.

Este sistema tiene un inconveniente, el plazo desde que se adjudican definitivamente los destinos hasta que se toma posesión de la plaza es muy corto.

En ese plazo el opositor debe:

- Decidir si la plaza que le ha sido asignada es de su agrado, para empezar con los trámites.
- Comunicar la baja a su empresa, respetando el tiempo de comunicación, que normalmente es de 15 días naturales o laborables, pero en algunos casos más amplio, pudiendo llegar a ser mayor a los 30 días.
- Buscar alojamiento y preparar una mudanza en caso de que la plaza sea en una ciudad distinta a la del opositor.

En la actualidad, con las redes sociales, los opositores aprobados de una misma convocatoria suelen estar en contacto mediante algún grupo de mensajería instantánea. Es común que se intente contactar con todos los aprobados de una oposición para poder calcular cuál de las plazas ofertadas será asignada a cada opositor de modo que se pueda saber cuál es la plaza que definitivamente será asignada a cada uno.

Actualmente, se suele realizar una simulación mediante preguntas uno a uno, tablas de Excel y otros métodos, realmente poco eficientes, que suponen de bastante trabajo y esfuerzo, y a menudo introducen fallos en las posibles elecciones de los opositores.

Mediante este simulador de selección de plazas se pretende implementar un sistema para los grupos auto organizados que les permita determinar qué plaza será otorgada a cada opositor antes de la fecha en que se haga efectiva esta elección.

Para ello permitirá registrar en la aplicación oposiciones con sus respectivas plazas, así mismo permitirá registrar y validar a los opositores que pertenezcan a una misma oposición, y permitirá a los opositores realizar la elección de sus plazas deseadas. De modo que desde el primer día de publicación de las

posibles plazas, la comunidad de opositores aprobados pueda conocer o bien de manera exacta o bien aproximada, la plaza asignada a cada opositor.

Según mi experiencia en los procesos selectivos que he participado, este simulador es una herramienta muy útil, que aunque no tiene carácter oficial, sirve de mucha ayuda a los grupos auto organizados.

El simulador no está libre de problemas, a menudo existen algunos opositores que o bien no son encontrados por los grupos organizados, o bien no quieren participar en el simulador de destinos. Esto implica que los puestos designados más allá del puesto de este opositor no sean 100% fidedignos, y puedan diferir del resultado real. Este problema es minimizado por el hecho que se suelen ofertar varios destinos iguales dentro de un mismo organismo, de modo que si un opositor selecciona 5 destinos iguales uno tras otro, al opositor le será indiferente obtener el 1º 2º o 5º destino.

## **1.2 Objetivos del Trabajo**

Dentro de los objetivos del trabajo de fin de grado se deben distinguir por una parte los objetivos de la propia aplicación que se quiere implementar y por otra parte los objetivos técnicos o didácticos que como alumno se esperan obtener.

### **1.2.1 Objetivos funcionales**

El principal objetivo del trabajo ha sido proveer un simulador plenamente funcional, que contuviera todas las funcionalidades básicas y necesarias para el funcionamiento de la plataforma.

Se han incluido todos los perfiles necesarios, y todas las funcionalidades básicas, desde la creación de convocatorias, creación de plazas, validación de opositores, selección de destinos o la obtención del resultado de la simulación.

Se han incluido algunas funcionalidades secundarias, como la gestión de los datos maestros por parte del administrador y una zona de administración.

Otras funcionalidades secundarias no ha sido posible implementarlas por la falta de tiempo, y se posponen para una implementación futura, como por ejemplo la invitación mediante enlace, invitación por email, la carga de destinos mediante fichero CSV, o un formulario de contacto para el administrador o el organizador de cada convocatoria.



### **1.2.2 Objetivos didácticos**

Los objetivos didácticos de este trabajo se dividir en varios bloques bien diferenciados.

Por una parte la planificación del proyecto, la estimación en tiempo del trabajo a desarrollar y la ejecución de las diferentes fases del proyecto siempre planificado en consideración del corto espacio de tiempo para un proyecto software.

Por otra parte, la elección de las tecnologías a usar dentro del proyecto, que se adecuen al tipo de proyecto y a las peculiaridades de este, al tipo de trabajo desarrollado y la limitación de tiempo y de personal de este, y las compatibilidades entre las diferentes tecnologías elegidas.

Finalmente la implementación del proyecto, creado desde cero, con el uso de una serie de tecnologías totalmente nuevas y en otro casos tecnologías que aunque teniendo conocimiento de ellas, no se habían aplicado tan en profundidad, con un tiempo considerable de aprendizaje para cada una de ellas.

### **1.3 Enfoque y método seguido**

Uno de los primeros pasos a la hora de llevar a cabo un proyecto software es verificar si efectivamente el producto a desarrollar va a tener un ROI (Return On Investment) adecuado, considerar si existe en el mercado otro producto que pueda usarse directamente, o en todo caso pueda adaptarse mediante un desarrollo si las licencias lo permiten, o si es viable el desarrollo del producto en el tiempo y coste especificados por el cliente.

En este proyecto el retorno de inversión es claramente nulo, ya que es un producto gratuito que se va a entregar a la comunidad de opositores, con la esperanza de que se use y se mejore y amplíe su código en un futuro por parte de otros opositores.

Se ha estudiado la posibilidad de usar software existente, en la actualidad esta tarea se viene realizando mediante hojas Excel y contactos uno a uno con los opositores, lo cual no se puede considerar una alternativa válida.

Existe una herramienta desarrollada por otro opositor como simulador de destinos, pero está desarrollada en PHP, está desarrollada únicamente para una oposición en concreto, y el autor de la herramienta no tiene intención de compartir el código fuente, así que también se ha descartado esa opción.

Finalmente se ha decidido realizar una aplicación desde cero, abierta a todo tipo de oposiciones, desarrollada en JEE y con un alcance básico que permita realizar la simulación desde el inicio hasta el final. Se ha estimado el tiempo

necesario para las funcionalidades y se han priorizado estas en primarias, secundarias y opcionales, y se ha acotado la carga de trabajo para el tiempo disponible y la cantidad de desarrolladores (uno en este caso).

De este modo, el resto de funcionalidades secundarias y opcionales fuera del alcance podrán ser realizadas por la comunidad de opositores en un futuro si se estiman necesarias, mientras que la plataforma puede funcionar plenamente desde el primer momento.

#### 1.4 Planificación del Trabajo

Se ha dividido el trabajo en 4 grandes bloques:

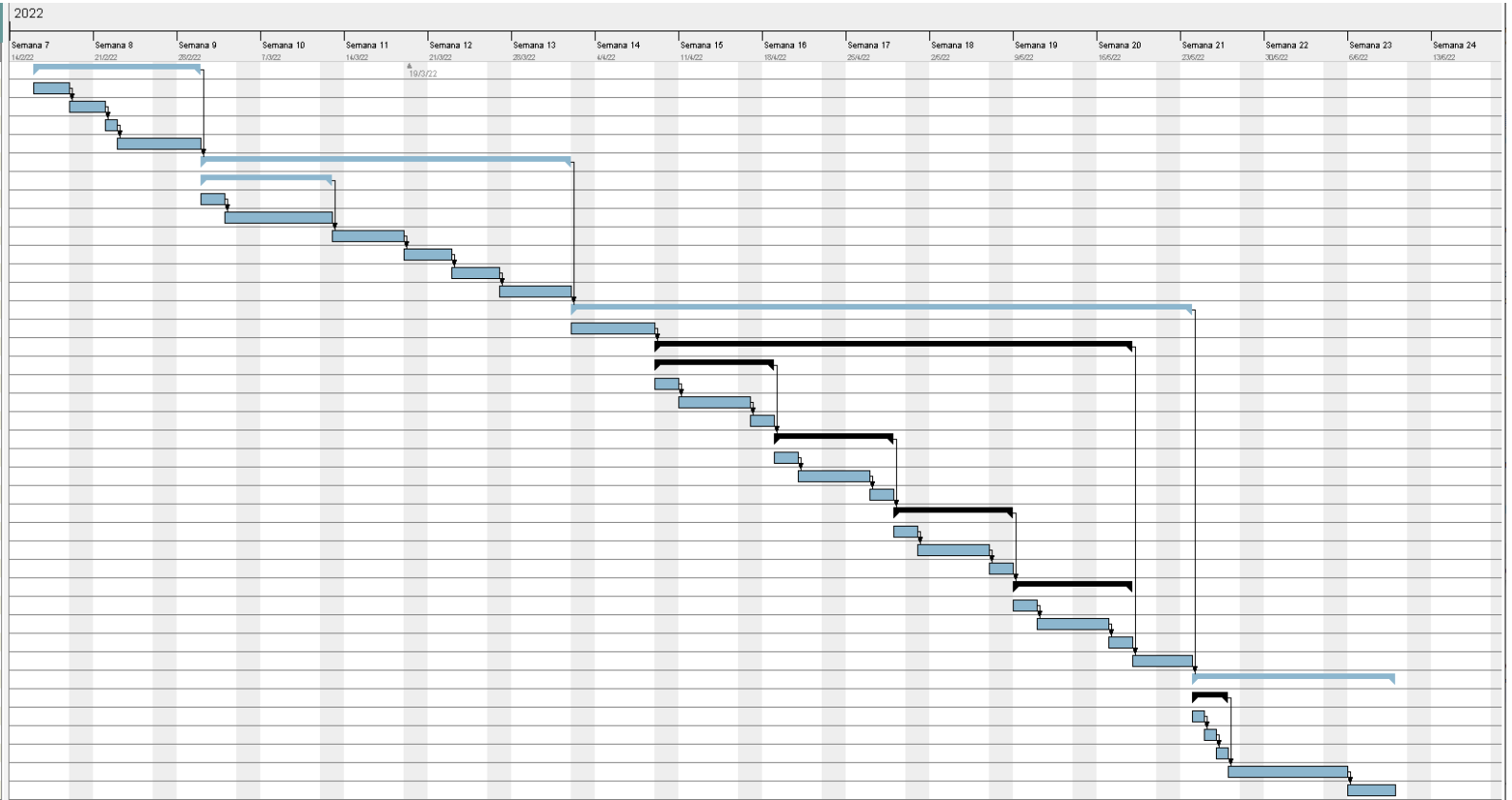
- **Planificación:**  
Todo lo referente a la planificación inicial del trabajo de fin de grado, información inicial sobre el proyecto, alcance de este, primeras ideas sobre las tecnologías a utilizar.
- **Análisis y diseño:**  
Creación de las fichas de los casos de uso, del diagrama de casos de uso, diagrama de base de datos, diagrama de clases, diversos diagramas de arquitectura, diagrama de navegación, y de los prototipos necesarios.
- **Construcción:**  
Generación del código de la aplicación, de la base de datos, estructura del proyecto, pruebas e integración. Todo ello se ha separado en 4 iteraciones de 10 días, para seguir el modo de trabajo de las metodologías ágiles.
- **Memoria y presentación:**  
Revisión de los entregables, finalización y revisión de la memoria, generación del documento para la presentación del trabajo de fin de grado.

Los bloques han sido divididos a su vez en tareas menores, a las que se le ha asignado una cantidad de tiempo (en días) para realizarse, distribuyéndose así el tiempo total disponible para la realización del trabajo de fin de grado.

Para su mejor comprensión se ha realizado un **diagrama de Gantt** que se adjunta a continuación.



Nombre	Fecha de i...	Fecha d...	Dur...
Planificación	16/2/22	1/3/22	14
Recopilación de información sobre el proyecto	16/2/22	18/2/22	3
Decisión sobre frameworks y tecnologías	19/2/22	21/2/22	3
Preparación de la reunión inicial	22/2/22	22/2/22	1
Creación Gantt	23/2/22	1/3/22	7
Análisis y Diseño	2/3/22	1/4/22	31
Casos de uso	2/3/22	12/3/22	11
Planteamiento casos de uso	2/3/22	3/3/22	2
Creación Fichas de casos de uso	4/3/22	12/3/22	9
Prototipos	13/3/22	18/3/22	6
Diseño de la base de datos	19/3/22	22/3/22	4
Diagrama de Clases	23/3/22	26/3/22	4
Diagrama de arquitectura	27/3/22	1/4/22	6
Construcción	2/4/22	23/5/22	52
Construcción estructura del proyecto	2/4/22	8/4/22	7
Construcción del proyecto	9/4/22	18/5/22	40
Iteración 1	9/4/22	18/4/22	10
Frameworks y tecnologías	9/4/22	10/4/22	2
Codificación	11/4/22	16/4/22	6
Integración y pruebas	17/4/22	18/4/22	2
Iteración 2	19/4/22	28/4/22	10
Frameworks y tecnologías	19/4/22	20/4/22	2
Codificación	21/4/22	26/4/22	6
Integración y pruebas	27/4/22	28/4/22	2
Iteración 3	29/4/22	8/5/22	10
Frameworks y tecnologías	29/4/22	30/4/22	2
Codificación	1/5/22	6/5/22	6
Integración y pruebas	7/5/22	8/5/22	2
Iteración 4	9/5/22	18/5/22	10
Frameworks y tecnologías	9/5/22	10/5/22	2
Codificación	11/5/22	16/5/22	6
Integración y pruebas	17/5/22	18/5/22	2
Revisión del proyecto	19/5/22	23/5/22	5
Memoria y Presentación	24/5/22	9/6/22	17
Revisión y Entrega	24/5/22	26/5/22	3
Script base de datos	24/5/22	24/5/22	1
Codigo fuente y empaquetado	25/5/22	25/5/22	1
Servidor de aplicaciones y otras herramientas	26/5/22	26/5/22	1
Revisión y finalización de la memoria	27/5/22	5/6/22	10
Creación de la presentación	6/6/22	9/6/22	4



## **1.5 Breve resumen de productos obtenidos**

- Código del proyecto JEE para su puesta en producción y su posible modificación y mejora.
- Script de base de datos con la creación del esquema y los datos maestros necesarios para el correcto funcionamiento de la aplicación.
- Documentación de análisis y diseño (diagramas, fichas de casos de uso) para las posibles modificaciones y mejoras de la aplicación.
- Pequeño manual sobre la instalación y puesta en marcha de la aplicación.

## **1.6 Breve descripción de los otros capítulos de la memoria**

A continuación se detallan los distintos aspectos del proyecto divididos en apartados.

En el apartado de Análisis y Diseño se detallan y adjuntan todos los productos obtenidos en dicha fase, en concreto los casos de uso, los diferentes diagramas y los prototipos realizados.

En el apartado de Implementación del proyecto se concretan las tecnologías utilizadas en el proyecto, las razones de la elección de cada tecnología sobre otras posibles y se explica la estructura del proyecto.

El apartado de Conclusiones explica los resultados y la experiencia personal obtenida con el proyecto.

En el apartado de glosario se pueden consultar los términos más importantes usados en la aplicación.

En el apartado de Bibliografía se enlazan las fuentes más importantes de consulta del trabajo.

Finalmente en los anexos se pueden encontrar unos pequeños manuales de instalación y usuario de la aplicación.

## 2. Análisis y Diseño

A la hora de llevar a cabo un proyecto como este es indispensable realizar un trabajo previo de análisis y diseño.

Es preciso analizar los requisitos funcionales y no funcionales que va a tener la aplicación. Especificar que funcionalidades son necesarias en la aplicación, priorizarlas y estimarlas y de este modo saber qué cantidad de trabajo se va a abordar y que funcionalidades quedarán pospuestas para más adelante. Por otra parte existen unos requisitos no funcionales que se deben cumplir, como tiempos de respuesta, escalabilidad, seguridad o posibilidad de adaptación a nuevas tecnologías o mejoras.

Aparte del análisis, es importante el diseño de la solución software, un diseño robusto no fallará con el tiempo, será mantenible y escalable, y permitirá a los nuevos desarrolladores (ya que uno de los objetivos de este proyecto es que futuros opositores puedan mantenerlo y ampliarlo) puedan disponer de la documentación técnica del proyecto.

### 2.1 Casos de uso

Los casos de uso son una excelente forma de mostrar al cliente cuales serán las funcionalidades de la aplicación, y por otra parte de acotar el alcance de la implementación de esta.

Aunque en este proyecto en concreto no existe un cliente del que recabar los casos de uso, ni mostrárselos para su aprobación, sí que se ha considerado útil su utilización de una manera académica.

#### 2.1.1 Actores principales

Existen tres actores principales en la aplicación:

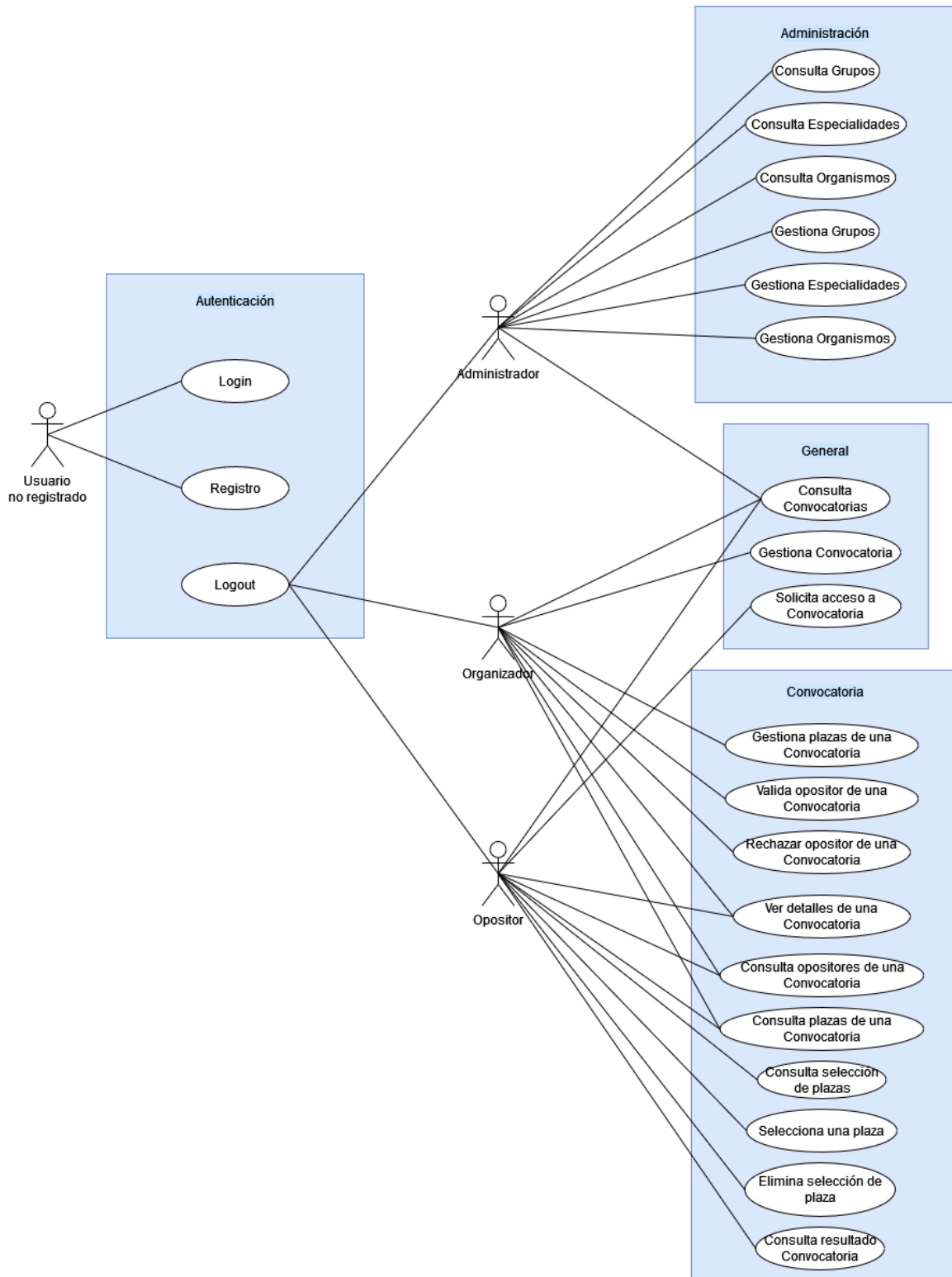
- **Administrador:** es el administrador del sitio, se encarga de gestionar los grupos, especialidades y organismos, y por otra parte puede administrar y realizar las mismas acciones sobre todas las convocatorias, opositores y plazas que pueden realizar los organizadores de cada convocatoria.
- **Organizador:** es un usuario normal, que ha registrado una convocatoria en la aplicación, tendrá poder sobre esta para realizar todas las acciones de administración de esta, como puede ser añadir, quitar y modificar plazas, validar y rechazar opositores, o cambiar el estado de la convocatoria.
- **Opositor:** es cada uno de los usuarios normales de la aplicación, pueden inscribirse en las convocatorias, y una vez que hayan sido validados por el organizador de una convocatoria, podrán seleccionar plazas y consultar el resultado de la simulación.

Los **usuarios no registrados** por su parte no tienen ningún permiso en especial, y no podrán acceder a ninguna ventana de la aplicación salvo la ventana de Login y de Registro.

### **2.1.2 Diagrama de casos de uso**

Muestra las diferentes funcionalidades (casos de uso) a las que puede acceder cada actor de la aplicación. Para su mejor entendimiento, se han agrupado los distintos casos de uso por componente.

El propósito del diagrama de casos de uso es por una parte tener un listado de todas las funcionalidades que debe cumplir el sistema y por otra parte poder mostrarlos mediante un esquema visual sencillo que pueda entender el cliente.



Todos los casos de uso pertenecientes a Administración, General y Convocatoria tienen una relación de `<<include>>` con el caso de uso de Login, es decir, para realizar cualquiera de esas acciones, el usuario deberá haberse autenticado en la aplicación. No se ha añadido este hecho al diagrama para una mejor visualización y entendimiento de este.

### 2.1.3 Fichas de casos de uso

En cada ficha se muestra con detalle para cada caso de uso los datos más significativos, el flujo habitual de ejecución y los posibles flujos alternativos que pueden ocurrir.

Algunos casos de uso han sido agrupados en uno solo para simplificar este documento, ya que tienen similitud con otros casos de uso de los que sí que existe ficha.

En otros casos de uso no se ha desarrollado su flujo principal, ni flujos alternativos por mayor simplicidad, ya que presentaban igualmente gran similitud con otros existentes.

<b>ID - Nombre</b>	<b>CU001 – Login</b>
<b>Descripción</b>	Acceso a la aplicación
<b>Actor principal</b>	Usuario registrado
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Registro, Logout
<b>Precondición</b>	El usuario está registrado.
<b>Postcondición</b>	El usuario se habrá autenticado.
<b>Flujo principal de eventos</b>	
1	El usuario introduce su usuario y contraseña y pulsa login.
2	El usuario accede a la ventana principal.
<b>Flujos alternativos de eventos</b>	
1a1	El usuario introduce un login o contraseña incorrectos.
1a2	Se muestra un error.

<b>ID - Nombre</b>	<b>CU002 – Logout</b>
<b>Descripción</b>	Desconexión de usuario de la aplicación
<b>Actor principal</b>	Usuario registrado
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Portada, Login
<b>Precondición</b>	El usuario está registrado y se ha autenticado en la aplicación.
<b>Postcondición</b>	El usuario habrá terminado su sesión y se mostrará la portada.
<b>Flujo principal de eventos</b>	
1	El usuario pincha en el link de Logout.
2	Se cierra la sesión del usuario.
3	Se redirige al usuario a la portada.
<b>Flujos alternativos de eventos</b>	

<b>ID - Nombre</b>	<b>CU003 – Registro</b>
<b>Descripción</b>	Registro de usuarios en la aplicación
<b>Actor principal</b>	Usuario no registrado



<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Portada, Login
<b>Precondición</b>	El usuario no está registrado.
<b>Postcondición</b>	El usuario se habrá registrado.
<b>Flujo principal de eventos</b>	
1	El usuario introduce sus datos correctamente.
2	Se crea el usuario en la aplicación.
3	El usuario accede a la aplicación.
<b>Flujos alternativos de eventos</b>	
1a1	El usuario introduce algún dato incorrecto o falta algún dato.
1a2	Se muestra un error al usuario para que complete los datos correctamente.
1b1	El usuario introduce los datos de un usuario existente.
1b2	Se indica al usuario que dicho usuario ya existe.

<b>ID - Nombre</b>	<b>CU004 – Consulta Convocatorias</b>
<b>Descripción</b>	Ventana principal para usuarios autenticados, muestra dos listados de convocatorias, en el primero se muestran las convocatorias a las que el usuario está aceptado o es organizador, y en el segundo listado se muestran el resto de convocatorias. Permite acceder al resto de funcionalidades.
<b>Actor principal</b>	Usuario registrado
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Todos
<b>Precondición</b>	El usuario está registrado y se ha autenticado.
<b>Postcondición</b>	El usuario ha podido visualizar sus convocatorias y las convocatorias activas, y ha podido realizar las acciones permitidas para su perfil (administrador, organizador u opositor)
<b>Flujo principal de eventos</b>	
1	El usuario accede al sistema.
2	Se muestra la ventana principal con los dos listados de convocatorias.
<b>Flujos alternativos de eventos</b>	

<b>ID - Nombre</b>	<b>CU005 – Consulta grupos</b>
<b>Descripción</b>	Listado de los grupos existentes
<b>Actor principal</b>	Administrador
<b>Tipo</b>	Opcional
<b>Casos relacionados</b>	Gestiona grupos, Consulta escalas, Consulta organismos
<b>Precondición</b>	El usuario ha accedido como administrador
<b>Postcondición</b>	El usuario ha podido ver todos los grupos.
<b>Flujo principal de eventos</b>	
1	El usuario accede al listado de grupos.
2	El usuario visualiza los grupos existentes.

Flujos alternativos de eventos	

ID - Nombre	CU006 – Gestiona grupos
Descripción	Creación, modificación y borrado de grupos
Actor principal	Administrador
Tipo	Opcional
Casos relacionados	Consulta grupos, Gestiona escalas, Gestiona organismos
Precondición	El usuario ha accedido como administrador
Postcondición	El usuario ha podido crear, modificar o borrar grupos.
Flujo principal de eventos	
1	Desde el listado de grupos el usuario pulsa en crear, modificar o eliminar un grupo.
2	El usuario rellena al menos los datos necesarios.
3	El usuario guarda los cambios.
4	El usuario visualiza los grupos actualizados en el listado.
Flujos alternativos de eventos	
3a1	El usuario introduce datos incorrectos o no introduce datos obligatorios.
3a2	Se muestra un mensaje de error al usuario.
3b1	El usuario intenta borrar un grupo que tiene especialidades.
3b2	Se muestra un mensaje de error al usuario.

ID - Nombre	CU007 – Consulta escalas
Descripción	Listado de las escalas existentes
Actor principal	Administrador
Tipo	Opcional
Casos relacionados	Gestiona escalas, Consulta grupos, Consulta organismos
Precondición	El usuario ha accedido como administrador
Postcondición	El usuario ha podido ver todas las escalas.

ID - Nombre	CU008 – Gestiona escalas
Descripción	Creación, modificación y borrado de escalas
Actor principal	Administrador
Tipo	Opcional
Casos relacionados	Consulta escalas, Gestiona grupos, Gestiona organismos
Precondición	El usuario ha accedido como administrador
Postcondición	El usuario ha podido crear, modificar o borrar escalas.

ID - Nombre	CU009 – Consulta organismos
Descripción	Listado de los organismos existentes
Actor principal	Administrador
Tipo	Opcional

<b>Casos relacionados</b>	Gestiona organismos, Consulta escalas, Consulta grupos
<b>Precondición</b>	El usuario ha accedido como administrador
<b>Postcondición</b>	El usuario ha podido ver todos los organismos.

<b>ID - Nombre</b>	<b>CU010 – Gestiona organismos</b>
<b>Descripción</b>	Creación, modificación y borrado de organismos
<b>Actor principal</b>	Administrador
<b>Tipo</b>	Opcional
<b>Casos relacionados</b>	Consulta organismos, Gestiona grupos, Gestiona escalas
<b>Precondición</b>	El usuario ha accedido como administrador
<b>Postcondición</b>	El usuario ha podido crear, modificar o borrar organismos.

<b>ID - Nombre</b>	<b>CU011 – Gestión Convocatorias</b>
<b>Descripción</b>	Creación, modificación y borrado de convocatorias
<b>Actor principal</b>	Administrador, Organizador
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Consulta convocatorias
<b>Precondición</b>	El usuario está autenticado y es administrador u organizador y quiere operar sobre una convocatoria suya o crear una nueva.
<b>Postcondición</b>	EL usuario habrá creado, modificado o eliminado una convocatoria.
<b>Flujo principal de eventos</b>	
1	El usuario crea, modifica o elimina una convocatoria.
2	Los datos se guardan correctamente.
3	El usuario visualiza las convocatorias actualizadas.
<b>Flujos alternativos de eventos</b>	
1a1	El usuario introduce datos incorrectos o no introduce datos obligatorios.
1a2	Se muestra un mensaje de error al usuario.
1b1	El usuario intenta borrar una convocatoria con opositores o plazas.
1b2	Se marca como inactiva la convocatoria, permaneciendo invisible para los opositores.

<b>ID - Nombre</b>	<b>CU012 – Solicita acceso a convocatoria</b>
<b>Descripción</b>	Un opositor solicita acceso a alguna de las convocatorias existentes.
<b>Actor principal</b>	Opositor
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Consulta convocatorias, Consulta opositores convocatoria, Valida opositor, Rechaza opositor
<b>Precondición</b>	El usuario está autenticado y no tiene acceso a la convocatoria.
<b>Postcondición</b>	El usuario realiza una petición de acceso a la convocatoria.

<b>Flujo principal de eventos</b>	
1	El usuario accede al listado de convocatorias activas.
2	El usuario rellena la palabra secreta en una convocatoria.
3	El usuario pincha en Solicitar acceso a la convocatoria.
4	Se guarda la solicitud de acceso a la convocatoria
5	Se muestra el listado de convocatorias
<b>Flujos alternativos de eventos</b>	
3a1	El usuario solicita acceso a una convocatoria de la que es propietario o ya tiene acceso.
3a2	Se muestra un mensaje de error indicando el problema.
3b1	El usuario no rellena la palabra secreta.
3b2	No se permite al usuario pinchar en el botón de solicitar acceso.
3c1	El usuario ha indicado una palabra secreta incorrecta
3c2	Se muestra un mensaje de error indicando el problema y se le insta a preguntar al organizador la palabra secreta.

<b>ID - Nombre</b>	<b>CU013 – Ver detalles de una convocatoria</b>
<b>Descripción</b>	Vista principal de la convocatoria con los datos generales.
<b>Actor principal</b>	Organizador, Opositor
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Consulta convocatorias, Solicita acceso a convocatoria
<b>Precondición</b>	Ser organizador de la convocatoria a la que se accede o ser un opositor validado.
<b>Postcondición</b>	El usuario habrá visualizado los datos generales de la convocatoria y los accesos al resto de funcionalidades de la convocatoria.
<b>Flujo principal de eventos</b>	
1	El usuario pincha en una convocatoria en el listado de convocatorias.
2	Se muestra una ventana con los datos principales de la convocatoria.
<b>Flujos alternativos de eventos</b>	
1a1	El usuario no tiene acceso a la convocatoria.
1a2	Se muestra un mensaje indicando al usuario que no tiene acceso.

<b>ID - Nombre</b>	<b>CU014 – Consulta opositores de una convocatoria</b>
<b>Descripción</b>	Listado con los opositores que participan en una convocatoria o han solicitado acceso a esta.
<b>Actor principal</b>	Organizador
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Ver convocatoria, Valida opositor, Rechaza opositor
<b>Precondición</b>	Ser organizador de la convocatoria a la que se accede o ser un opositor validado.
<b>Postcondición</b>	El usuario habrá visto un listado con los opositores que tienen acceso a la convocatoria o han solicitado este.
<b>Flujo principal de eventos</b>	
1	El usuario pincha en una convocatoria en el botón de listado de opositores de la

	convocatoria
2	Se muestra una ventana con un listado con los opositores que tienen acceso a la convocatoria, ordenados de manera descendente por su orden, y otro listado con los opositores que han solicitado el acceso.
<b>Flujos alternativos de eventos</b>	

<b>ID - Nombre</b>	<b>CU015 – Validar opositor</b>
<b>Descripción</b>	Un organizador valida un opositor que ha solicitado el acceso a una convocatoria, permitiendo así que acceda a la convocatoria, seleccione plazas y consulte el resultado de la simulación.
<b>Actor principal</b>	Organizador
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Consulta opositores de una convocatoria, Rechaza opositor.
<b>Precondición</b>	Ser organizador de la convocatoria a la que se accede al listado, o ser administrador.
<b>Postcondición</b>	El opositor estará validado para poder acceder a la convocatoria, a seleccionar plaza y consultar el resultado, también se le habrá asignado un orden de selección de plaza.
<b>Flujo principal de eventos</b>	
1	El usuario organizador de una convocatoria accede al listado de opositores que han solicitado el acceso a la convocatoria.
2	El usuario rellena el orden del opositor.
3	El usuario pincha en el botón validar opositor de un opositor.
4	Se guarda el opositor como validado.
5	Se muestra el listado de opositores de la convocatoria, mostrando al opositor como validado. El opositor ya no aparece en el listado de opositores solicitantes
<b>Flujos alternativos de eventos</b>	
2a1	Se indica un orden inválido, ya sea un valor no numérico, un orden existente o uno mayor al máximo, o negativo.
2a2	Se muestra un mensaje indicando el error correspondiente.
3b1	El opositor ya estaba validado.
3b2	Se muestra un mensaje indicando el error.

<b>ID - Nombre</b>	<b>CU016 – Rechaza opositor</b>
<b>Descripción</b>	Un organizador rechaza un opositor que ha solicitado el acceso a una convocatoria.
<b>Actor principal</b>	Organizador
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Consulta opositores convocatoria, Valida opositor.
<b>Precondición</b>	Ser organizador de la convocatoria a la que se accede al listado, o ser administrador.
<b>Postcondición</b>	El opositor desaparecerá del listado de opositores solicitantes

	en la convocatoria, y no podrá acceder a la convocatoria.
<b>Flujo principal de eventos</b>	
1	El usuario organizador de una convocatoria accede al listado de opositores que han solicitado el acceso a la convocatoria.
2	El usuario pincha en el botón rechazar opositor de un opositor.
5	Se elimina el opositor como solicitante.
6	Se muestra el listado de opositores de la convocatoria, y el listado de opositores solicitantes de la convocatoria, el opositor ya no aparece como solicitante.
<b>Flujos alternativos de eventos</b>	

<b>ID - Nombre</b>	<b>CU017 – Consulta plazas de una convocatoria</b>
<b>Descripción</b>	Listado con las plazas de una oposición, permite acceder a los usuarios al resto de funcionalidades relacionadas con las plazas de la convocatoria.
<b>Actor principal</b>	Organizador y Opositor
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Gestiona plazas, Consulta selección de plazas, Consulta resultado convocatoria
<b>Precondición</b>	El usuario tiene permiso para acceder a la convocatoria de las plazas
<b>Postcondición</b>	El usuario habrá visualizado las plazas de la oposición.
<b>Flujo principal de eventos</b>	
1	El usuario pincha sobre el botón de listado de plazas de una convocatoria.
2	Se muestra el listado de plazas de la convocatoria.
<b>Flujos alternativos de eventos</b>	

<b>ID - Nombre</b>	<b>CU018 – Gestiona plazas</b>
<b>Descripción</b>	Crea, modifica y borra las plazas de una convocatoria
<b>Actor principal</b>	Organizador
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Consulta de plazas, Consulta selección de plazas, consultar resultado convocatoria
<b>Precondición</b>	El usuario es organizador de la convocatoria a la que pertenecen las plazas sobre las que quiere operar.
<b>Postcondición</b>	Se habrán guardado correctamente los cambios en las plazas.
<b>Flujo principal de eventos</b>	
1	El usuario accede a la ventana de listado de plazas de una convocatoria.
2	El usuario pincha sobre crear, modificar o borrar una plaza.
3	El usuario realiza los cambios necesarios sobre la plaza y pincha en guardar.
4	La plaza queda guardada correctamente con los nuevos cambios.
5	Se visualiza el listado de plazas con los nuevos cambios reflejados.
<b>Flujos alternativos de eventos</b>	

3a1	El usuario no ha rellenado correctamente los datos o ha olvidado datos obligatorios.
3a2	No se guardan los cambios y se muestra un mensaje explicativo al usuario.
2b1	El usuario intenta borrar una plaza que ya está seleccionada por algún opositor
2b2	Se borran todas las elecciones de los opositores antes de borrar la plaza.

<b>ID - Nombre</b>	<b>CU019 – Consulta selección de plazas</b>
<b>Descripción</b>	El usuario opositor ve la lista de elecciones que ha realizado sobre las plazas de la convocatoria.
<b>Actor principal</b>	Opositor
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Listado de plazas, Selecciona plaza, Eliminar Selección
<b>Precondición</b>	El opositor tiene acceso a la convocatoria de las plazas y la convocatoria tiene las plazas ya creadas.
<b>Postcondición</b>	El opositor habrá consultado su selección de plazas.
<b>Flujo principal de eventos</b>	
1	Desde el listado de plazas, el opositor pincha sobre el botón de seleccionar plazas.
2	Se muestra la ventana del listado de elecciones de plazas del opositor en orden ascendente del 1 al total de plazas de la convocatoria.
<b>Flujos alternativos de eventos</b>	

<b>ID - Nombre</b>	<b>CU020 – Selecciona plaza</b>
<b>Descripción</b>	El usuario selecciona una plaza para el orden de selección que ha elegido.
<b>Actor principal</b>	Opositor
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Listado de plazas, Consulta selección de plazas, Eliminar Selección
<b>Precondición</b>	El opositor tiene acceso a la convocatoria de las plazas y la convocatoria tiene las plazas ya creadas.
<b>Postcondición</b>	El opositor habrá realizado una selección de una plaza, que quedará correctamente guardada en su orden.
<b>Flujo principal de eventos</b>	
1	El opositor accede al listado de elecciones de plazas y pincha sobre una de las líneas de la tabla, esté ocupada o vacía.
2	Se muestran las plazas existentes.
3	El opositor pincha sobre una plaza.
4	La elección de plaza se guarda correctamente.
5	Se muestra el listado de elecciones actualizado con la elección realizada.
<b>Flujos alternativos de eventos</b>	
2a1	No existen plazas creadas para la convocatoria actual.
2a2	No se permite al opositor seleccionar ninguna plaza

3b1	El opositor elige una plaza que ya ha sido elegida en otro orden.
3b2	Se muestra un mensaje explicativo al opositor, indicándole como se deben elegir las plazas.
4c1	El opositor guarda una elección sobre una línea que ya tenía una elección
4c2	Se sobrescribe la elección anterior con la actual.

<b>ID - Nombre</b>	<b>CU021 – Elimina selección de plaza</b>
<b>Descripción</b>	El usuario elimina una selección de plaza para el orden de selección que ha elegido.
<b>Actor principal</b>	Opositor
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Listado de plazas, Consulta selección de plazas, Selecciona plaza
<b>Precondición</b>	El opositor ya ha realizado alguna selección de plaza.
<b>Postcondición</b>	El opositor habrá eliminado una selección de una plaza, quedando la elección para ese orden vacía.
<b>Flujo principal de eventos</b>	
1	El opositor accede al listado de elecciones de plazas y pincha sobre una de las líneas de la tabla ocupada con una elección.
2	Se muestra las plazas existentes y un botón de eliminar selección.
3	El opositor pincha sobre el botón de eliminar selección.
4	La elección de plaza se elimina correctamente.
5	Se muestra el listado de elecciones actualizado con la elección eliminada.
<b>Flujos alternativos de eventos</b>	
1a1	El opositor pincha sobre una fila no ocupada.
2a2	El opositor pincha sobre el botón eliminar selección
2a3	Se muestra un mensaje al opositor indicándole que no hay una elección aún en ese orden.

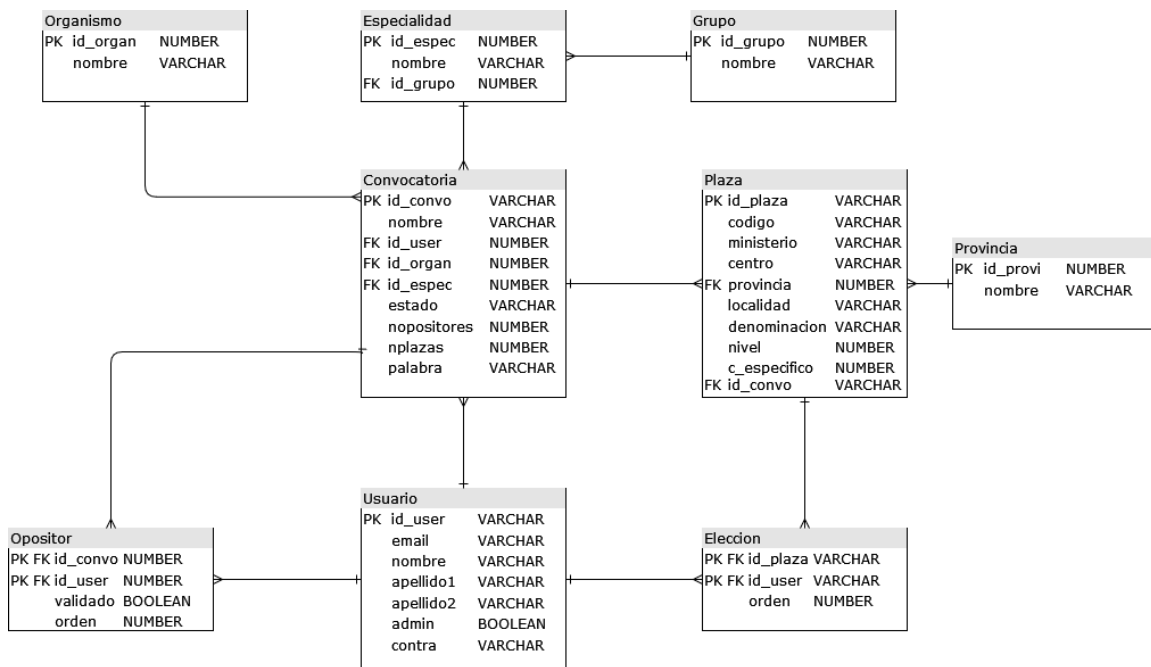
<b>ID - Nombre</b>	<b>CU022 – Consulta resultado convocatoria</b>
<b>Descripción</b>	El opositor consulta el resultado de la asignación de las plazas en el estado actual, es decir, se muestra al opositor, la plaza que tendría asignada en este momento con las elecciones del resto de opositores. También se muestra con sombreados verdes, grises y rojos, las posibles plazas que le pueden tocar al opositor, las que no lo pueden tocar, y las que otros opositores con un orden de elección anterior al él han elegido, y por tanto no puede optar a ellas.
<b>Actor principal</b>	Opositor
<b>Tipo</b>	Primario
<b>Casos relacionados</b>	Consulta de plazas, Gestiona plazas, Consulta selección de plazas
<b>Precondición</b>	El usuario tiene acceso a la convocatoria y ha realizado su selección de plazas.



<b>Postcondición</b>	El usuario habrá podido consultar el resultado de la simulación y sabrá la plaza que le va a ser asignada en el proceso selectivo.
<b>Flujo principal de eventos</b>	
1	El usuario pincha en el botón de consultar resultado plazas
2	Se muestra la ventana con las plazas ordenadas por la selección del opositor, mostrándose en verde las posibles plazas del opositor, en rojo las que han elegido opositores con mejor orden que él, y en gris las que no le serán asignadas por haber sido elegidas en un orden posterior.
<b>Flujos alternativos de eventos</b>	

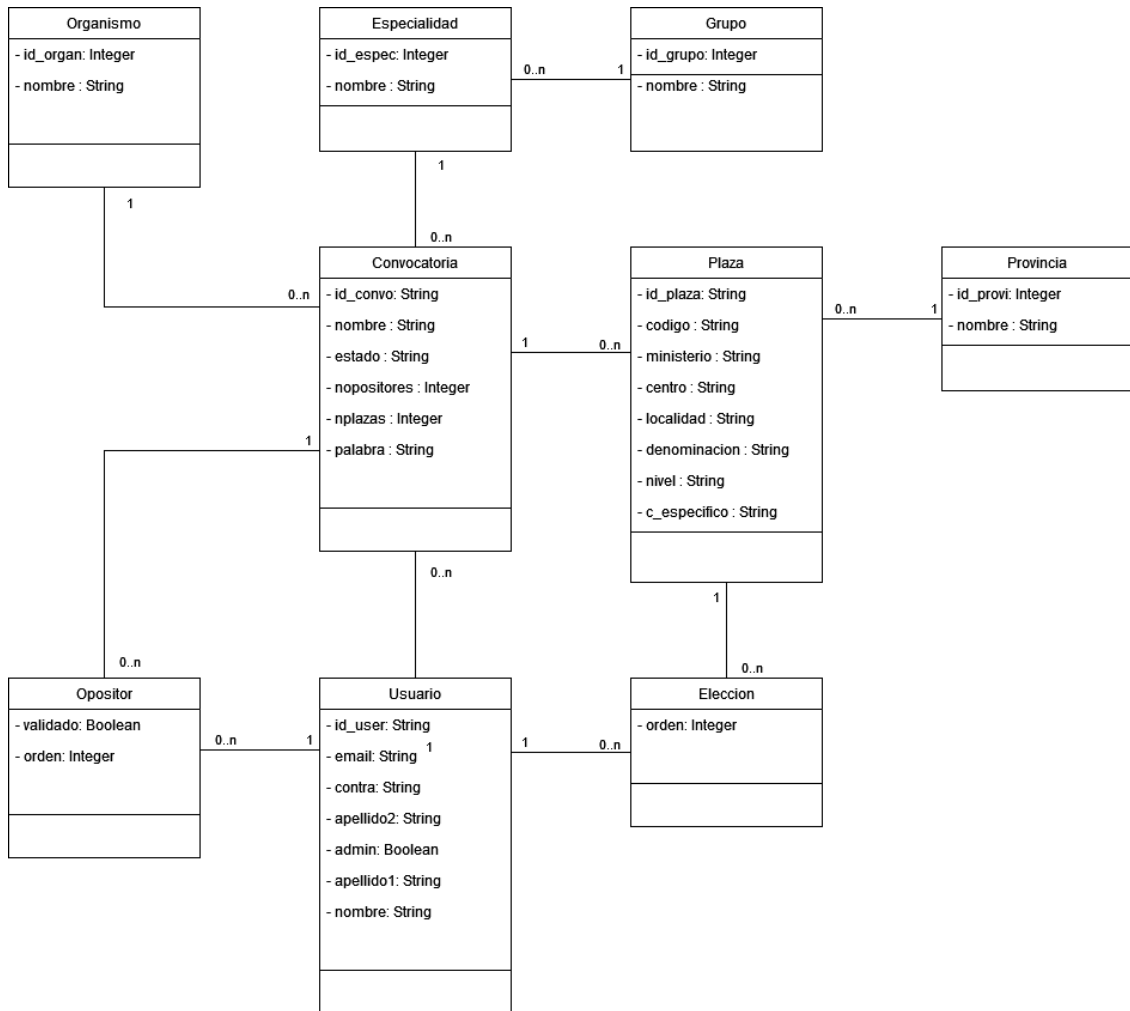
## 2.2 Diagrama relacional de base de datos

Representación de la estructura de la base de datos con las tablas y sus restricciones de integridad y claves ajenas.



## 2.3 Diagrama de clases

Representación de las diferentes clases que conforman la aplicación y como se relacionan entre ellas. Por simplicidad se ha decidido no añadir los métodos de cada clase.



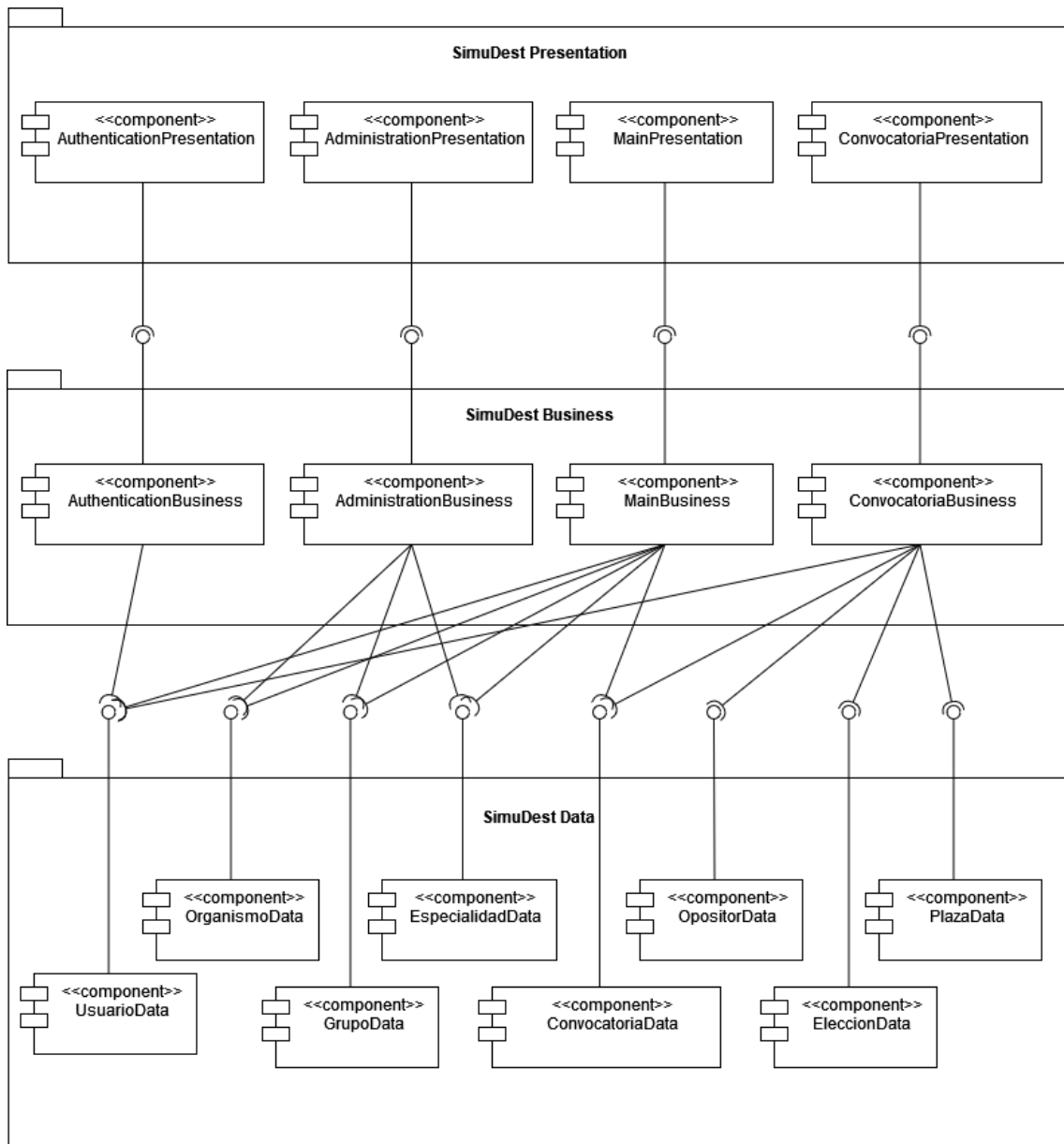
## 2.4 Diagrama de arquitectura

La aplicación se desarrollará según el paradigma de 3 capas: presentación, lógica de negocio y datos.

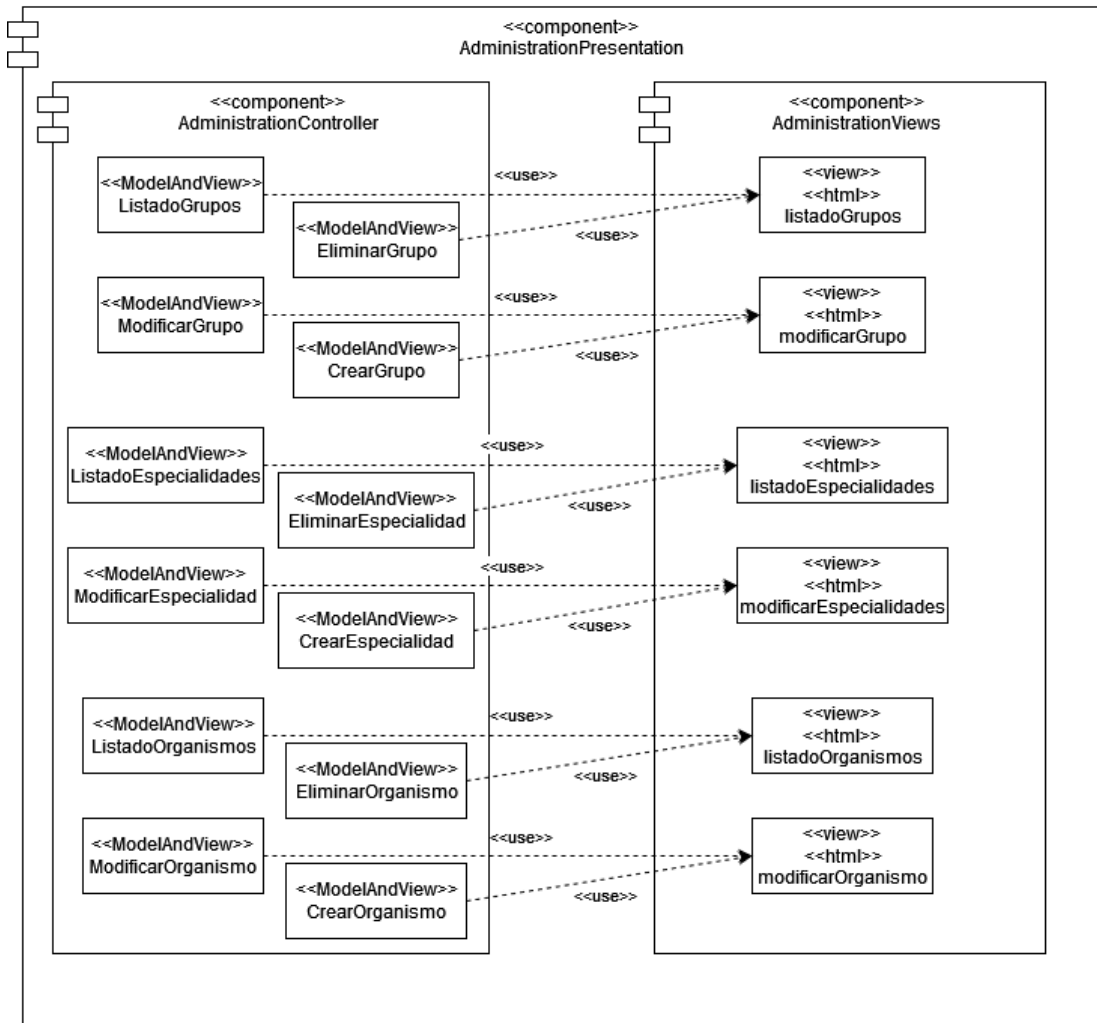
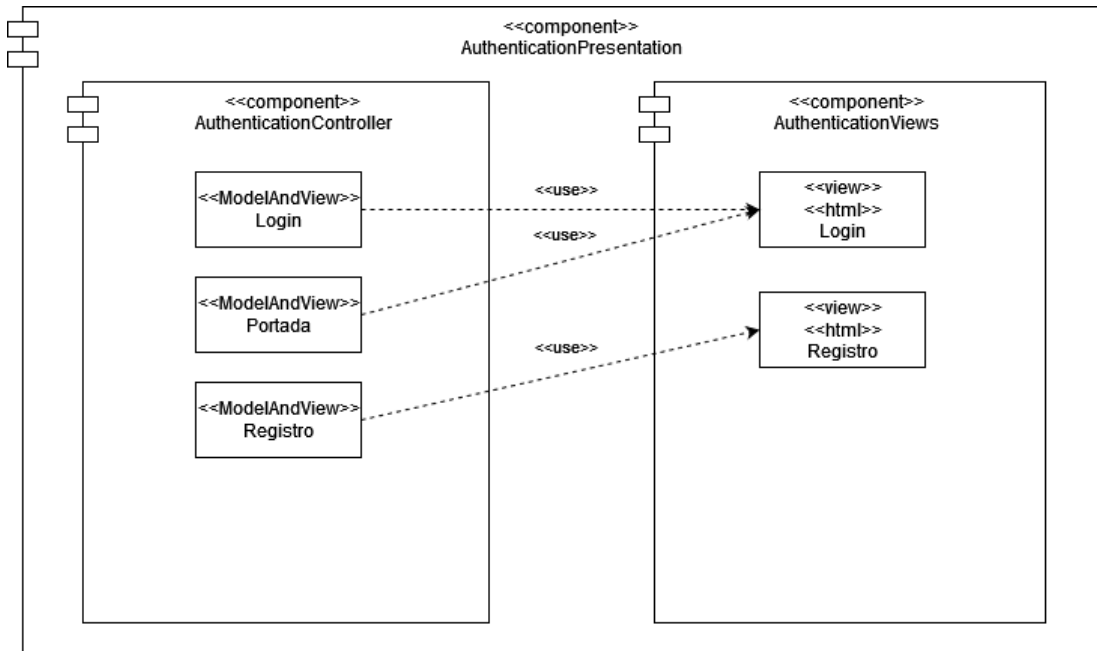
En este diagrama se muestran cada una de estas 3 capas y los distintos componentes que las forman y como se relacionan entre ellos.

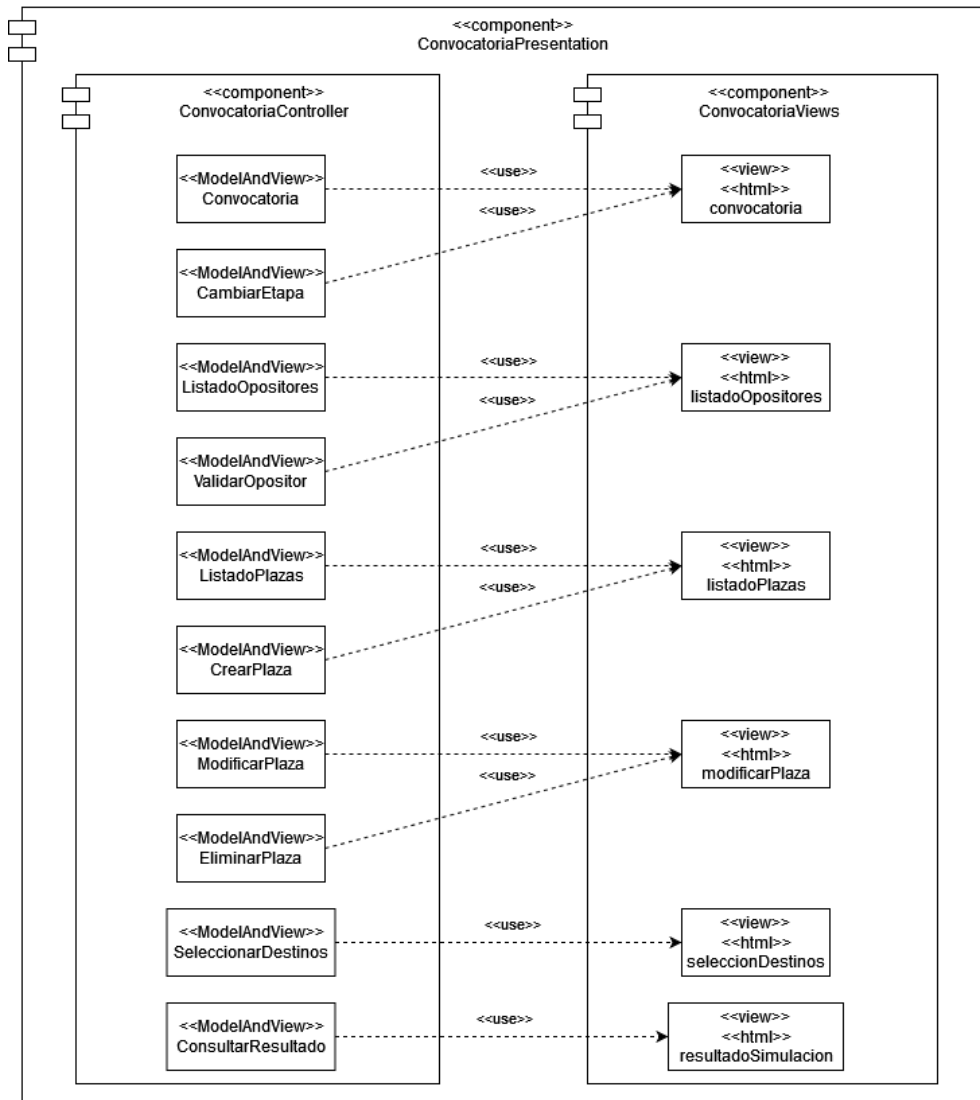
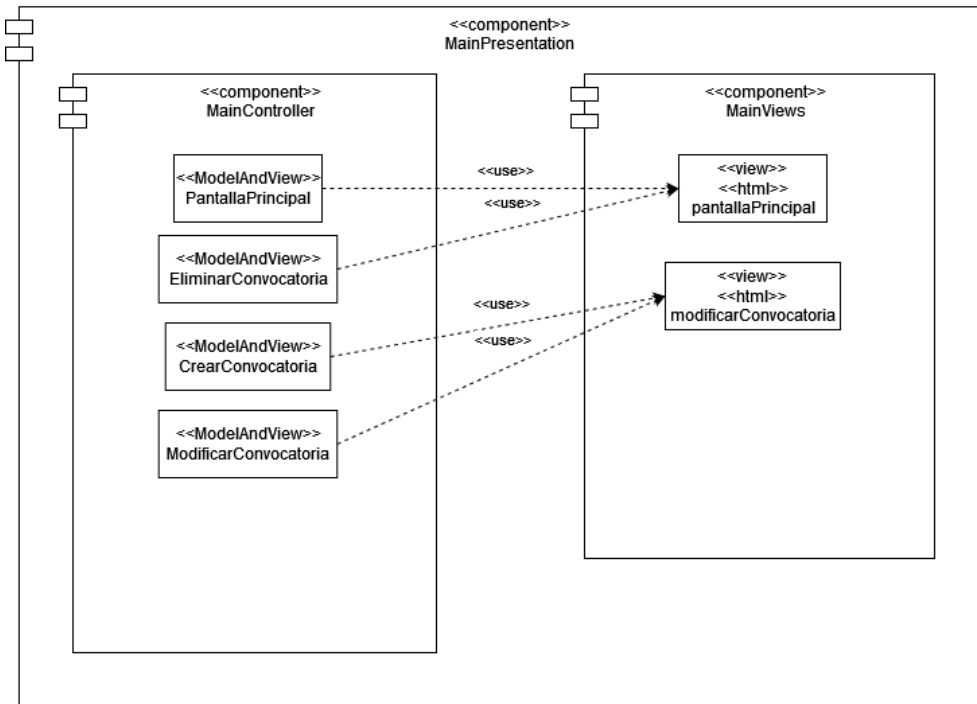
Este diagrama se ha dividido para mejor entendimiento, mostrando primero una visión global del sistema, y posteriormente varios diagramas más específicos de cada uno de los componentes en cada una de las capas de la aplicación.

## 2.4.1 Visión global

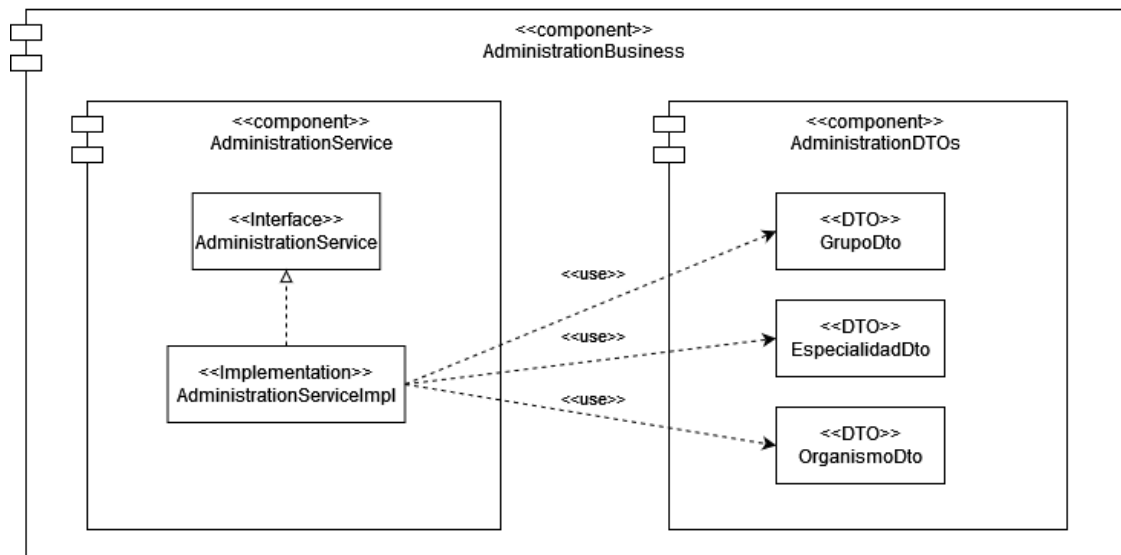
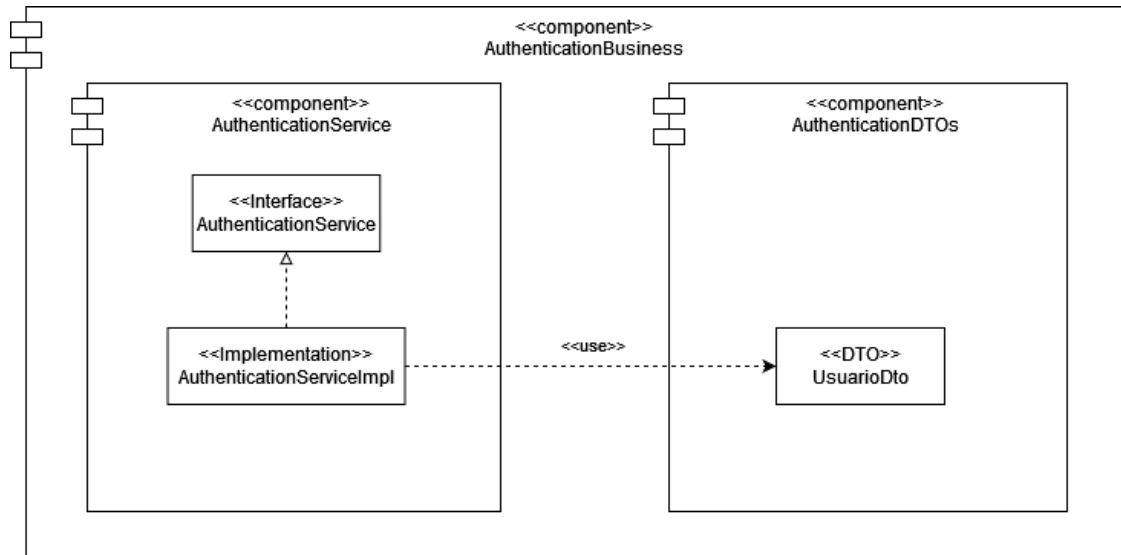


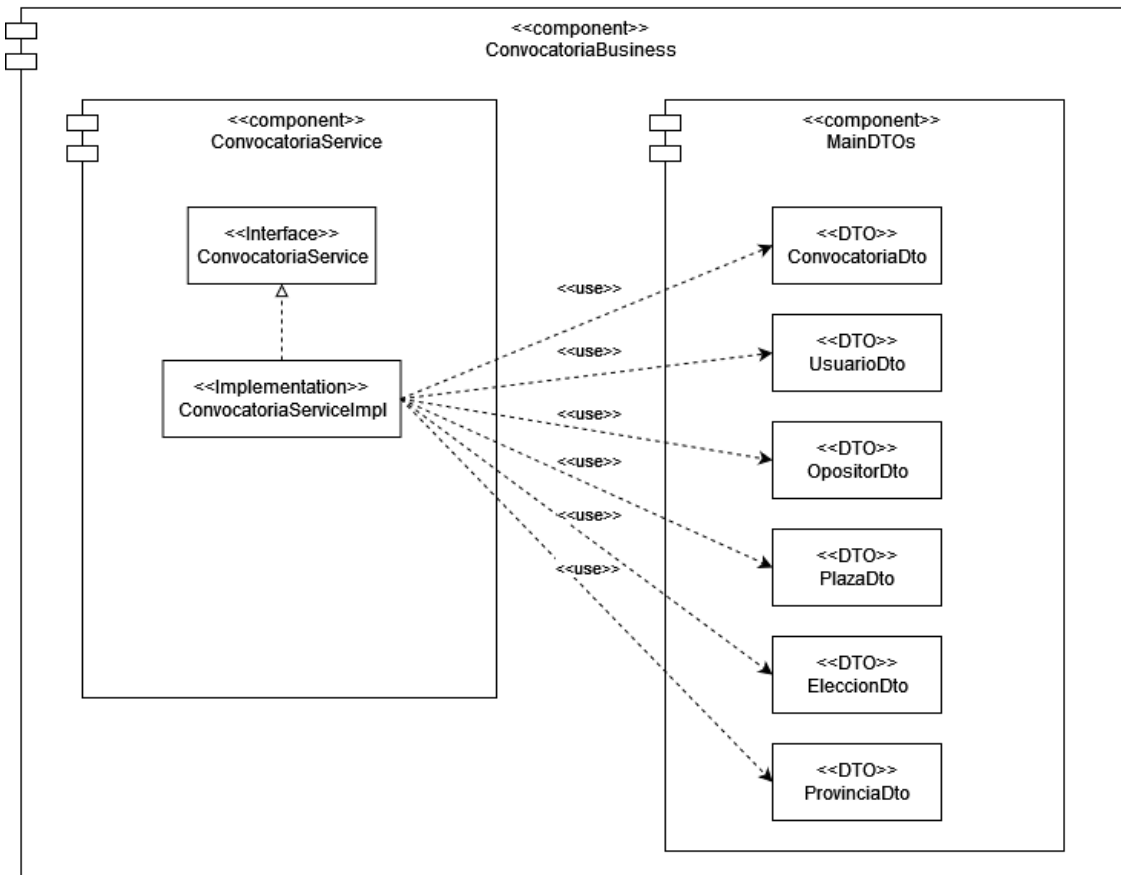
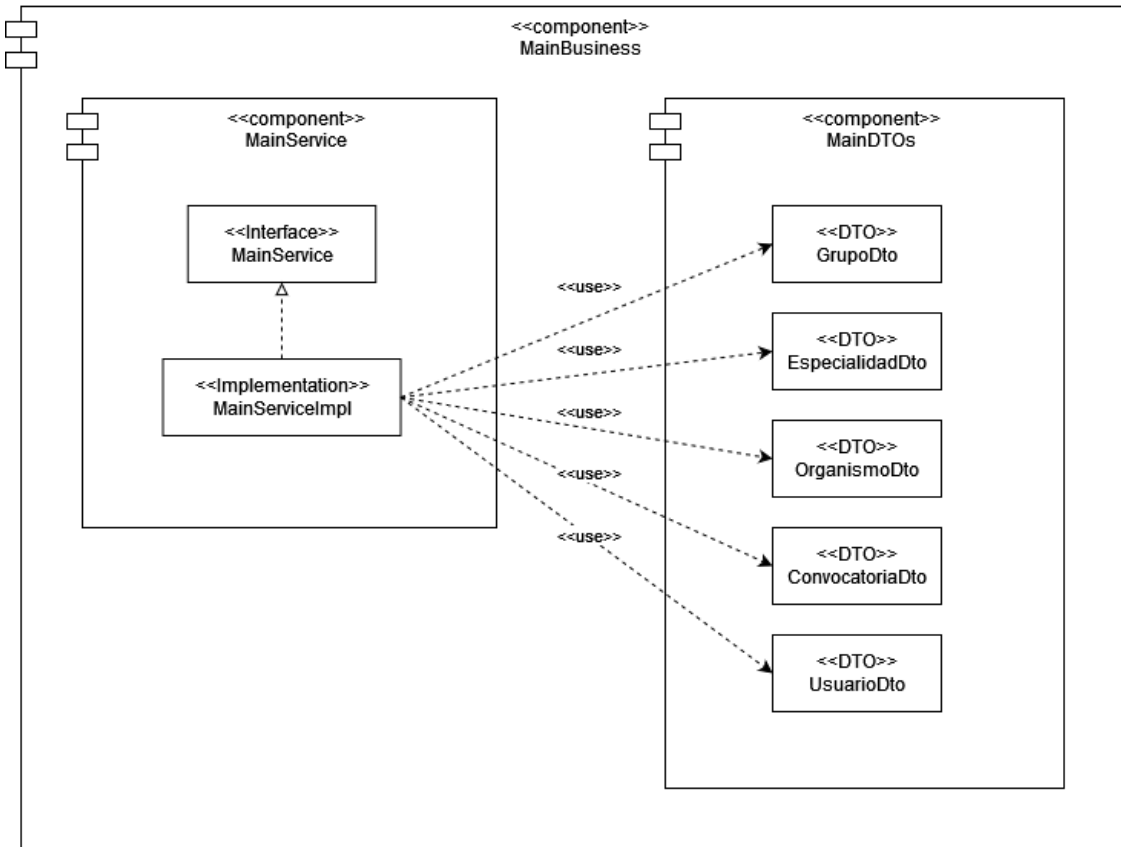
## 2.4.2 Capa de presentación



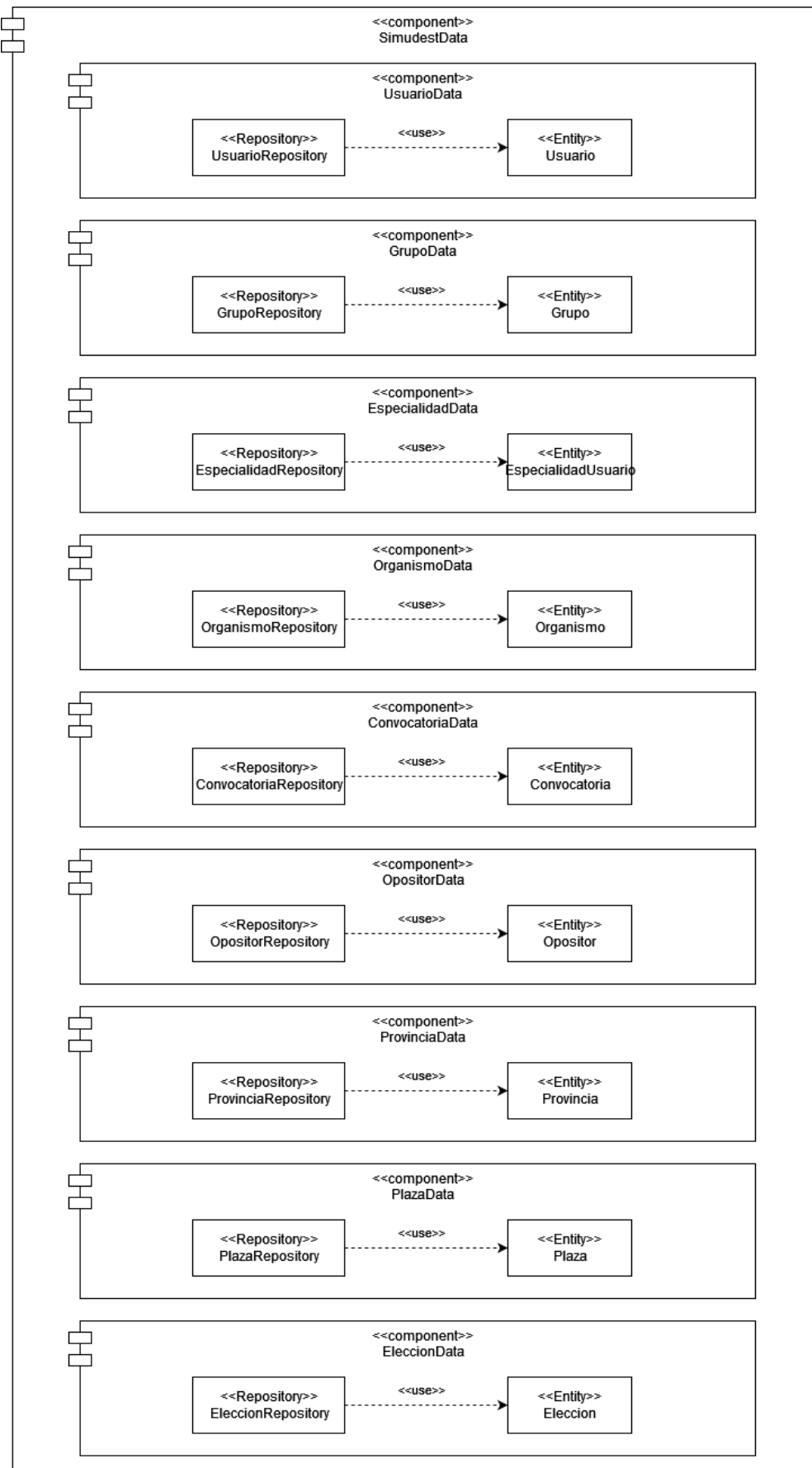


### 2.4.3 Capa de lógica de negocio





## 2.4.4 Capa de datos





## 2.5 Modelo de pantallas

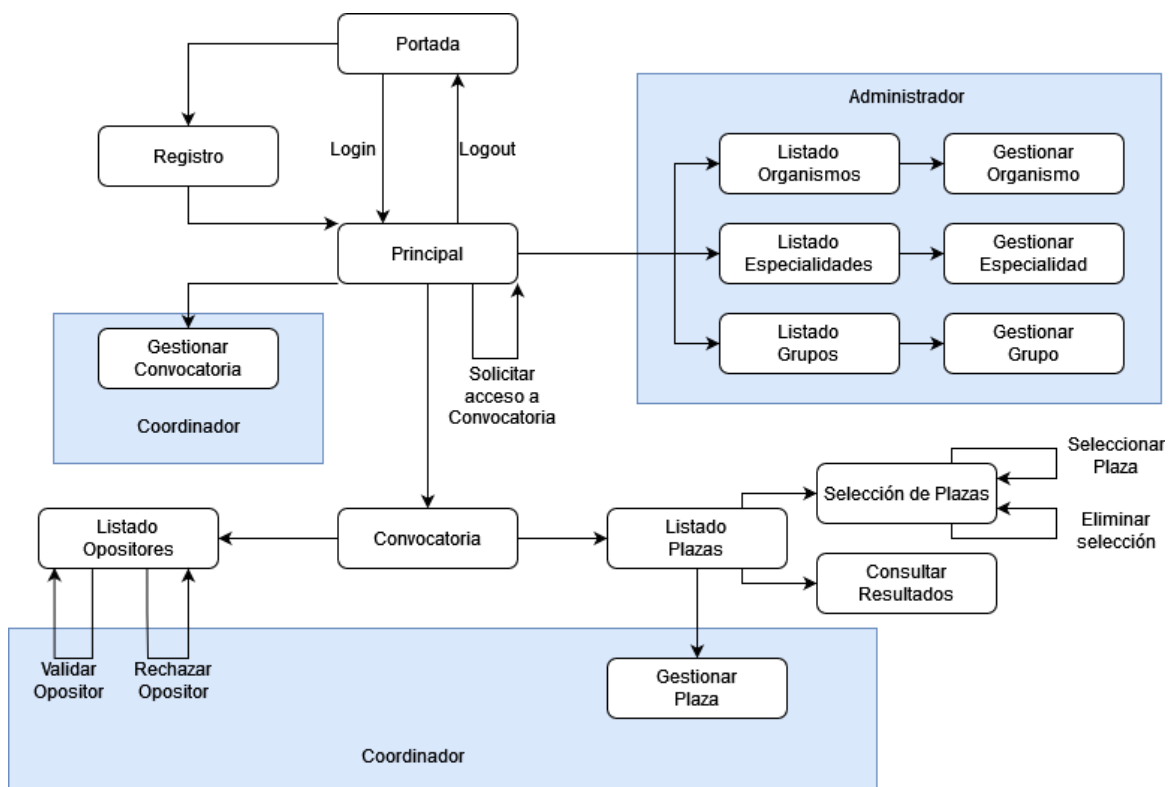
Dentro del modelado de pantallas se encuentra la visión más cercana al cliente, la más entendible para él, por una parte con el prototipado de pantallas y por otra parte con el diagrama de navegación.

### 2.5.1 Diagrama de navegación

A continuación se muestran las posibles opciones de navegación entre las distintas ventanas de la aplicación. Se han agrupado algunas ventanas y funcionalidades por el actor que tiene permiso para realizar dichas acciones.

El propósito del diagrama de navegación es doble, por una parte, para que el cliente entienda cual es el funcionamiento general de la aplicación, y por otra parte para el propio desarrollador o equipo de desarrolladores poder tener una visión global de la aplicación, y si se ha tenido en cuenta la navegación hasta todas las posibles funcionalidades, y comprobar que constan todas en el diseño.

Hay que señalar que este diagrama no es un diagrama UML, y no es frecuentemente utilizado, sin embargo he decidido utilizarlo por las razones planteadas arriba.



## 2.5.2 Prototipos

Cada prototipo es una imagen del aspecto que se pretende obtener para cada ventana de la aplicación. Es posible que este difiera ligeramente de la implementación final debido a aspectos técnicos.

El propósito del prototipo es dar una idea al cliente de cómo se verá la aplicación para su aprobación, o para que indique los posibles cambios necesarios.

### PORTADA



Simulador de destinos

Login

password

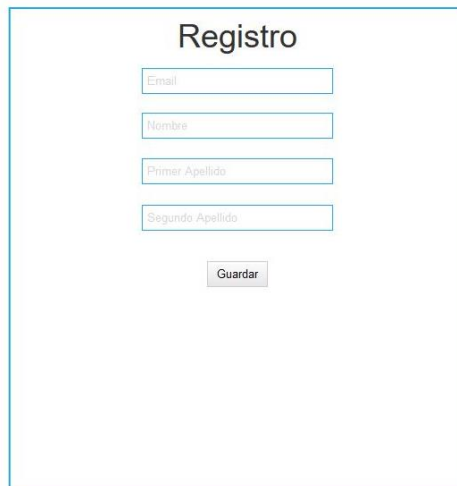
Entrar

¿No tienes cuenta? Regístrate

Detailed description: The image shows a wireframe of a login page for an application titled 'Simulador de destinos'. The page has a white background with a blue border. At the top center, the title 'Simulador de destinos' is displayed in a black sans-serif font. Below the title, there are two input fields: the first is labeled 'Login' and the second is labeled 'password'. Both fields have a light blue border. Below the password field is a grey button with the text 'Entrar' in white. At the bottom center, there is a link that says '¿No tienes cuenta? Regístrate'.

Portada de la aplicación, solo permite el acceso a la aplicación, o al registro de usuario, es la única ventana visible como usuario no registrado. En la versión final contendrá un pequeño texto explicativo sobre el propósito de la aplicación y en caso de ser necesario por ley o norma, un pie con accesos directos a “Protección de datos” y “Términos de uso”

## REGISTRO



Registro

Email

Nombre

Primer Apellido

Segundo Apellido

Guardar

Ventana de registro de la aplicación, en principio se pedirán los mínimos datos posibles para poder identificar al usuario, ya que cuantos más datos se almacenen, mayor será el nivel de seguridad requerido por protección de datos.

## PANTALLA PRINCIPAL

Crear Convocatoria Gestionar Grupos Gestionar Escalas

Mis convocatorias Administrador u Organizador de la convocatoria

Organismo	Convocatoria	Escala	Grupo
Administración General del Estado	Convocatoria 2020-2021	Escala Administrativa de informática	C1
Junta de Comunidades de Castilla la Mancha	Convocatoria 2020-2021	Escala Superior de informática	A1

Administrador

Convocatorias activas

Organismo	Convocatoria	Escala	Grupo
Administración General del Estado	Convocatoria 2020-2021	Escala Administrativa de informática	C1
Junta de Comunidades de Castilla la Mancha	Convocatoria 2020-2021	Escala Superior de informática	A1
Administración General del Estado	Convocatoria 2021	Escala Técnica de informática	C1
Administración General del Estado	Convocatoria 2021	Ingeniería Técnica Industrial	A1
Administración General del Estado	Convocatoria 2020-2021	Especialidad Jurídica	C1
Universidad de Granada	Convocatoria 2020	Auxiliar administrativa	A1
Ayuntamiento de Madrid	Convocatoria 2020	Gestión administrativa	C1
Junta de Comunidades de Castilla la Mancha	Convocatoria 2021	Escala Administrativa de informática	A1

1 2 3...10

Ventana principal que visualizan los usuarios al entrar a la aplicación.

En la cabecera se encuentra un menú solo visible por el administrador, que le permite acceder a las funcionalidades de administración y mantenimiento de la aplicación.

La tabla superior incluirá todas las convocatorias a las que está inscrito un opositor, o de las que este sea organizador. Como opositor podrá acceder a la convocatoria pinchando en ella. En caso de ser administrador además podrá modificarla o borrarla.

La tabla inferior incluirá todas las convocatorias activas en la aplicación, de modo que los opositores puedan buscar convocatorias y consultar los detalles de cada una para poder inscribirse. Una vez inscritos deberán ser validados por el organizador de la convocatoria para que esta aparezca en el listado superior y puedan acceder al resto de opciones de esta.

# CONVOCATORIA



Ventana principal de la convocatoria, en ella se muestran los datos principales de la convocatoria, como los opositores ya registrados del total y el número de plazas.

Desde esta ventana los opositores validados y el organizador de la convocatoria podrán acceder al listado de opositores y al listado de plazas.

## LISTADO OPOSITORES

### Opositores Validados

1	Juan Lopez Perez
2	Manuel Santiago Luz
3	Marta Trigo Trillo
4	John Smith
6	Lope Perez Garcia
7	Santiago Lopez Lopez
13	Maria Gil Noves
14	Juan Sinto Torras
17	Alberto Castro Castro

1 2 3...10

### Solicitantes

Juan Lopez Perez	Orden: 5	Validar	Rechazar
Manuel Santiago Luz	Orden: 5	Validar	Rechazar
Marta Trigo Trillo	Orden: 5	Validar	Rechazar
John Smith	Orden: 5	Validar	Rechazar

Administrador u Organizador de la convocatoria

Se muestran los opositores ya validados de la convocatoria ordenados por su posición en el proceso selectivo. Según este orden podrán seleccionar su plaza de destino.

En la tabla de debajo se muestran los opositores pendientes de validar, aquí el organizador los rechazará o validará, indicando en este caso la posición en la que quedaron en el proceso selectivo para tener en cuenta por prioridad su selección de plazas.

## LISTADO PLAZAS

Plazas							
Codigo Plaza	Ministerio / Consejeria	Centro	Provincia	Localidad	Puesto de trabajo	Nive	C.Especifico
0012345	MINISTERIO DE INCLUSION, SEGURIDAD SOCIAL Y MIGRACIONES	SEGURIDAD SOCIAL Y PENSIONES. GERENCIA DE INFORMATICA DE LA SEGURIDAD SOCIAL	MADRID	MADRID	TECNICO / TECNICA AUXILIAR DE INFORMATICA	16	3.556,14
0012346	MINISTERIO DE INCLUSION, SEGURIDAD SOCIAL Y MIGRACIONES	SEGURIDAD SOCIAL Y PENSIONES. GERENCIA DE INFORMATICA DE LA SEGURIDAD SOCIAL	MADRID	MADRID	TECNICO / TECNICA AUXILIAR DE INFORMATICA	16	3.556,14
9876541	MINISTERIO DE POLITICA TERRITORIAL Y FUNCION PUBLICA	DELEGACIÓN DEL GOBIERNO EN CASTILLA LA MANCHA	TOLEDO	TOLEDO	TECNICO / TECNICA AUXILIAR DE INFORMATICA	15	3.556,14
0012345	MINISTERIO DE POLITICA TERRITORIAL Y FUNCION PUBLICA	SEGURIDAD SOCIAL Y PENSIONES. GERENCIA DE INFORMATICA DE LA SEGURIDAD SOCIAL	MADRID	MADRID	TECNICO / TECNICA AUXILIAR DE INFORMATICA	16	4.500,12
0012345	MINISTERIO DE INCLUSION, SEGURIDAD SOCIAL Y MIGRACIONES	SEGURIDAD SOCIAL Y PENSIONES. GERENCIA DE INFORMATICA DE LA SEGURIDAD SOCIAL	BARCELONA	BARCELONA	TECNICO / TECNICA AUXILIAR DE INFORMATICA	15	5.646,48

Administrador u Organizador de la convocatoria

Muestra un listado con las plazas de la convocatoria y los datos específicos de cada plaza, como su localización, nivel, y complemento específico.

El organizador podrá gestionar las plazas, añadiendo, modificando o eliminando las existentes.

Desde esta ventana cada opositor podrá acceder a la selección de plazas y a consultar el resultado de la simulación.

## 3. Implementación del proyecto

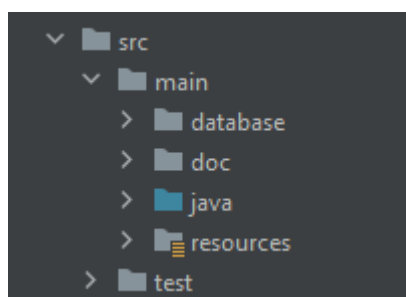
### 3.1 Propuesta tecnológica

Para la implementación del simulador de selección de plazas se ha decidido utilizar **Java Enterprise Edition (JEE)** como lenguaje de programación, con una arquitectura en 3 capas, siguiendo el patrón modelo-vista-controlador.

### 3.2 Estructura general del proyecto

Para la gestión del proyecto se utiliza **Maven**, que permite tanto su compilación, empaquetado y despliegue como la gestión e importación de librerías.

Maven utiliza una estructura de directorios que permite mantener el proyecto ordenado y estructurado. También se han incluido en esta estructura algunos directorios para almacenar los diagramas, y algunos scripts de base de datos necesarios para el despliegue de la aplicación. A continuación se detallan los directorios más importantes de la estructura.



#### ***/src/main/java/***

Contiene el código Java del proyecto.

#### ***/src/main/resources/static/***

Contiene los archivos estáticos de la aplicación (css, js, json, imágenes, etc.). También contiene algunos frameworks, librerías o templates de la capa web, como pueden ser JQuery, Bootstrap o adminlte.

#### ***/src/main/resources/templates/***

Contiene la parte web de la aplicación, es decir, todos los ficheros.html están contenidos aquí.

#### ***/src/main/database/***

Contiene varios scripts para la creación de la base de datos:

- **simudest\_schema.sql**: script de creación del esquema de la base de datos.



- **simudest\_data\_master.sql**: script con datos maestros necesarios para la ejecución de la aplicación, como pueden ser las provincias, grupos, especialidades, organismos...
- **simudest\_data\_test.sql**: script con datos de prueba para poder ver el programa con datos no verídicos. Incluye 200 usuarios, 5 convocatorias, opositores, selecciones de plazas, etc. Contiene todo lo necesario para poder hacer una demo completa del programa sin tener que insertar más datos.

#### **/src/main/doc/diagramas/**

En este directorio se encuentran todos los diagramas del programa. Están hechos con [draw.io](https://draw.io) que tiene una integración con GitHub para poder trabajar directamente con ellos desde allí.

### **3.3 Capa de presentación**

La capa de presentación consta principalmente de dos partes, las vistas y los controladores.

Para los controladores se ha usado el framework **Spring MVC**, específicamente los ModelAndView para la elección de la vista y el paso de los datos del modelo a las vistas.

Para las vistas se ha elegido **Thymeleaf** como motor de plantillas HTML. Inicialmente se consideró el uso de Server Faces (JSF) y su implementación Primefaces, pero por su mejor integración con todo el entorno de Spring Boot y el resto de frameworks utilizados, finalmente se decidió utilizar Thymeleaf.

También se han utilizado para las vistas varias herramientas y plantillas que deben mencionarse:

- **Jquery** (biblioteca javascript muy conocida y potente)
- **Bootstrap** (biblioteca javascript y css para el diseño web y diseño responsivo)
- **Datatables** (plugin basado en Jquery para la generación y manipulación de tablas)
- **AdminLTE** (template basado en Bootstrap)

### **3.4 Capa de lógica de negocio**

Para la lógica de negocio se ha utilizado la familia de frameworks de **Spring**, que nos provee inversión de control mediante la inyección de dependencias y otra serie de ventajas muy deseables en un proyecto software.

Se ha optado por una implementación mediante servicios utilizando la tecnología aportada por Spring, y el uso de DTOs (Data Transfer Object) que

permiten obtener un menor acoplamiento entre los Entities y la capa de presentación. Para este último objetivo también se han implementado unos Mappers que permiten mapear y transformar los atributos de cada entidad en su correspondiente DTO y viceversa.

### 3.5 Capa de datos

La capa de acceso a datos se ha implementado mediante el uso de **Java Persistence API (JPA)**, en concreto con la implementación de **Hibernate**, se ha elegido este ORM por ser una tecnología madura y potente, y por su fácil integración dentro del ecosistema de Spring Boot.

Para mayor facilidad de implementación, y dado que la complejidad de los accesos a base de datos por parte de la aplicación no es muy grande, se ha hecho uso también del framework **Spring Data JPA** que permite la realización de consultas y actualizaciones en base de datos de manera muy sencilla e intuitiva sin necesidad de escribir casi código.

En concreto las clases de la capa de datos son los Entity, que realizan el mapeo entre las columnas de base de datos y la clase java mediante anotaciones, además de otras características, como pueden ser las claves primarias y foraneas. Y las clases Repository que se ocupan de los accesos a base de datos, ya sean actualizaciones o consultas.

### 3.6 Autenticación

Para la autenticación de usuarios se utiliza **Spring Security**, un framework de sobrada experiencia, fácil de implementar y que cubre las necesidades del proyecto. Además de integrarse perfectamente con el resto de frameworks de la familia Spring.

### 3.7 Base de datos

Como Sistema de Gestión de Base de Datos (SGBD) se utiliza **PostgreSql** por ser de código abierto, de fácil uso, instalación y uso en equipos no necesariamente potentes. Si en un futuro se decidiese cambiar a otro SGBD no sería preciso realizar una gran cantidad de cambios, debido a la similitud de lenguaje entre los diferentes SGDB relacionales y debido al muy bajo acoplamiento con el resto del proyecto.

### 3.8 Servidor de aplicaciones

El servidor de aplicaciones utilizado es **Tomcat**, ya que otra de las decisiones tecnológicas es el uso del framework **SpringBoot**, de gran popularidad en la actualidad, y Tomcat viene embebido en este último.

Hay que especificar que Tomcat no es un servidor de aplicaciones propiamente dicho, sino un contenedor de servlets o un contenedor web.

### 3.9 IDE

Como entorno de desarrollo se han utilizado dos IDEs distintos, debido a que el trabajo ha sido realizado en dos ordenadores distintos con distintas necesidades.

En un equipo se ha decidido utilizar **Eclipse** por ser una solución software de código abierto, y por poseer gran cantidad de plugins para todas las tecnologías usadas en el proyecto, y existir una solución portable que no precisa instalación.

En el segundo equipo se ha utilizado **IntelliJ IDEA** software no gratuito en su versión empresarial, pero que dispone una versión gratuita para la comunidad.

### 3.10 Control de versiones

Como control de versiones se ha utilizado **Git**, en concreto la plataforma **GitHub**, que permite llevar la gestión de versiones sin necesidad de instalar un servidor para ello, y además nos permite compartir el código del programa de manera pública, ya que uno de los objetivos del proyecto es que este pueda ser mantenido y ampliado en el futuro por la comunidad de opositores.

El proyecto se ha implementado desde dos ordenadores distintos, así que el uso de control de versiones ha sido crucial para poder trabajar desde ambos equipos sobre el mismo código.

Además de utilizar el control de versiones para el código ha sido usado para tener un control sobre las versiones de los scripts de base de datos y para la realización de los diagramas, debido a la integración entre draw.io y GitHub.

## 4. Conclusiones

He adquirido conocimientos de mucho valor a lo largo del trabajo de fin de grado, desde las fases más tempranas como la planificación, la especificación de requisitos, el análisis y diseño de la aplicación, o la elección de las distintas tecnologías.

He tenido que modificar sobre la marcha algunas ideas que estaban claras, como diagramas, diseño de algunas columnas de la base de datos, o tecnologías a usar, ya que como se suele decir 'en el papel todo funciona'.

En general he aprendido muchas tecnologías Java que tenía o bien olvidadas, o bien no había usado nunca, o bien había usado, pero no había aplicado desde cero en un proyecto.

Funcionalmente el objetivo principal del trabajo era desarrollar una plataforma que permitiera a opositores de una convocatoria organizarse de manera autónoma para realizar la simulación de destinos, este objetivo se ha cumplido, ya que la aplicación cumple los requisitos mínimos para poder funcionar.

Otro objetivo secundario del trabajo era el uso de la aplicación para la convocatoria de ingreso libre de la oferta de empleo público de 2019 del Cuerpo de Técnicos Auxiliares de Informática de la Administración del Estado. Este objetivo no se ha podido cumplir, ya que el proceso se ha resuelto antes de lo previsto, realizándose la elección real de destinos antes de estar la aplicación finalizada.

En lo concerniente a la planificación se han hecho modificaciones en cuanto a la cuantía, duración y fechas y de los sprints. Debido a aspectos ajenos al trabajo de fin de grado, he habido fechas en las que no se ha podido avanzar en proyecto y otras fechas en las que se ha realizado mucho más trabajo del esperado, por este motivo las fechas indicadas en la planificación no se han ajustado a la reales.

Sí se ha respetado la metodología de trabajo, orientada a las metodologías ágiles, realizando integraciones del trabajo en GIT de manera diaria, estimando y dividiendo las tareas en partes asumibles para un solo desarrollador, y generando contenido de valor en iteraciones cortas.

Existen bastantes mejoras y funcionalidades secundarias que han quedado pendientes de implementarse. En concreto el envío de emails en el registro, la invitación a una convocatoria mediante un enlace, o la carga masiva de destinos. Estas funcionalidades, quedan pendientes para ser desarrolladas, bien por mi o bien por otro opositor que en un futuro decida colaborar con la plataforma, ya que la idea es que este proyecto sea de de naturaleza colaborativa.

## 5. Glosario

- **Convocatoria:** se entiende por convocatoria a cada uno de los procesos selectivos organizado por un organismo, sea cual sea su naturaleza (oposición, concurso/oposición, concurso), en el que una serie de opositores con un número de orden asignado compiten por una serie de plazas.
- **Organismo:** cada uno de los entes públicos que puede realizar una convocatoria.
- **Opositor de una convocatoria:** usuario registrado en la plataforma, que ha solicitado el acceso a una convocatoria y que una vez validado realizará una selección de destinos y consultará el resultado de la simulación.
- **Organizador de la convocatoria:** usuario registrado en la plataforma, que ha creado una convocatoria y que se encarga de las tareas de administración sobre dicha convocatoria, como por ejemplo la creación de las plazas y la validación de los opositores.
- **Administrador:** usuario con poderes en la plataforma que puede acceder a la zona de administración, y operar sobre todas las convocatorias para realizar las mismas tareas que tiene cada administrador de cada convocatoria.
- **Plaza:** una plaza es un puesto de trabajo, que estará asociada a un ministerio o consejería, a un centro, y tendrá un complemento monetario específico y un nivel, que definirán el salario de esta.

## 6. Bibliografía

- **Spring.io** [Sitio oficial de la familia de frameworks Spring, en concreto ha sido consultada la documentación sobre Spring Boot, Spring Data y Spring Security y varias guías y tutoriales]  
<https://spring.io/projects/>  
<https://spring.io/guides/>
- **Stackoverflow** [Sitio de consultas y respuestas sobre programación en general, se han consultado gran cantidad de artículos]  
<https://stackoverflow.com/>
- **Thymeleaf** [Documentación oficial sobre la biblioteca java Thymeleaf]  
<https://www.thymeleaf.org/documentation.html>
- **Baeldung** [Sitio web con gran cantidad de cursos e información sobre Java y Spring]  
<https://www.baeldung.com/>
- **Adminlte** [Documentación oficial del template construido en Bootstrap Adminlte]  
<https://adminlte.io/docs/>
- **Datatables** [Manual de este plugin basado en Jquery para la generación y manipulación de tablas]  
<https://datatables.net/manual/index>
- **Arteco-Consulting** [Tutorial sobre Spring Boot y Spring Security]  
<https://www.arteco-consulting.com/tu-primera-aplicacion-con-spring-boot/>  
<https://www.arteco-consulting.com/securizando-una-aplicacion-con-spring-boot/>
- **programandointentandolo.com** [Tutorial sobre Spring MVC, Spring Boot y Thymeleaf]  
<https://programandointentandolo.com/2018/12/spring-mvc-con-spring-boot-thymeleaf.html>
- **Javadevjournal** [Tutorial sobre customización del registro en Spring Security]  
<https://www.javadevjournal.com/spring-security/registration-with-spring-security/>

## 7. Anexos

### 7.1 Manual de instalación

Se adjunta un breve manual de instalación de la aplicación y de la base de datos en un entorno local, que permita la realización de una demo de la aplicación.

#### 7.1.1 Base de datos

El SGDB elegido es PostgreSQL, este pequeño manual asume que el PostgreSQL está instalado en el equipo local.

Para la creación de la base de datos, en el proyecto de GitHub (<https://github.com/Castillo82/simudest>), concretamente dentro del directorio /simudest/src/main/database/ se encuentran 3 scripts, uno para la creación del esquema, otro para los datos maestros, y otro para los datos de prueba.

Antes de nada, es necesario crear un usuario de base de datos simudest con contraseña 888888

En caso de no querer usar este usuario, habría que modificar el archivo application.properties las líneas:

```
spring.datasource.username= simudest
spring.datasource.password=888888
```

Y también los diferentes scripts de base de datos para indicar el nombre de usuario elegido.

El script **simudest\_schema.sql** es el primero que se debe ejecutar, e incluye la creación de la base de datos y del esquema, en caso de haber creado ya la base de datos, se debe obviar la primera parte del script hasta la sentencia:

```
CREATE SCHEMA simudest;
```

Y ejecutar el resto del script desde la creación del schema.

Posteriormente, se debe ejecutar **simudest\_data\_master.sql** que es el script de datos maestros.

Para finalizar es conveniente ejecutar **simudest\_data\_test.sql** que contiene los datos de prueba, no es indispensable, ya que incluye una serie de usuarios, convocatorias, plazas, opositores y elecciones de plazas.

#### 7.1.2 Aplicación

El proyecto está implementado con Spring Boot y Maven, para su ejecución en un entorno local los pasos deberían ser:

- Descargar y descomprimir el proyecto desde GitHub (<https://github.com/Castillo82/simudest>)
- Abrir una línea de comandos y situarse en el directorio raíz del proyecto.
- Ejecutar el comando “./mvnw spring-boot:run” o “mvnw spring-boot:run” según el sistema operativo usado.

La ejecución en un servidor dependerá de varios factores, por ejemplo se puede realizar mediante un docker compuesto con Tomcat y PostgreSQL, o mediante un Tomcat en un servidor no virtualizado.

En todo caso la ejecución en servidor queda fuera del alcance de este manual.

## **7.2 Manual de usuario**

### **7.2.1 Registro**

Desde la portada, pulsando en [¿No tienes cuenta?](#) se accede a la ventana de registro.

Se deben rellenar todos los campos, salvo ‘Segundo Apellido’ y posteriormente pulsar en registrarse para acabar el registro.

### **7.2.2 Login**

Desde la portada se deben rellenar el login y contraseña para poder acceder a la plataforma y al resto de funcionalidades.

### **7.2.3 Ventana principal**

En la ventana principal se muestran dos listados de convocatorias, en el primero se muestran las convocatorias a las que el usuario está aceptado o es organizador, y en el segundo listado se muestran el resto de convocatorias.

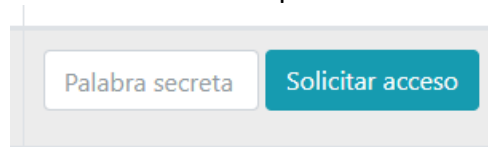
Desde aquí se pueden gestionar las convocatorias de las que uno es organizador, solicitar acceso a alguna convocatoria activa o acceder a alguna convocatoria en la que ha sido aceptado.

### **7.2.4 Solicitar acceso**

Para solicitar acceso a una convocatoria, previamente tenemos que haber contactado con el organizador de la convocatoria para que nos provea la



palabra secreta de acceso. Posteriormente, desde la ventana principal rellenamos podemos solicitar el acceso aquí:



Un formulario con un campo de texto que contiene el texto "Palabra secreta" y un botón azul a su derecha con el texto "Solicitar acceso".

Una vez hayamos solicitado el acceso correctamente, el organizador deberá validarnos como opositor de dicha convocatoria, y posteriormente podremos acceder a ella.

### 7.2.5 Crear convocatoria

Podemos crear una convocatoria, accediendo desde la cabecera mediante el botón 'Crear Convocatoria'.

En el formulario de crear convocatoria debemos rellenar todos los campos que se solicitan, si en los desplegados de Organismo, Grupo o Especialidad, no existe alguno de los datos de la convocatoria, deberemos contactar con un administrador para que los dé de alta en la aplicación.

Se sugiere no utilizar nombres de convocatoria muy extensos, o muy genéricos que puedan inducir a error.

Igualmente se sugiere utilizar una palabra secreta complicada y entregarla uno a uno a los posibles opositores.

### 7.2.7 Listado de opositores

Desde el listado de opositores se puede consultar los opositores que han sido validados, los que están pendientes de validar, y el orden de cada uno en la convocatoria.

### 7.2.8 Validar y rechazar opositores

Como organizador de una convocatoria, una de las tareas es validar o rechazar los opositores que solicitan acceso y otorgarles un orden dentro de esta. Esta operación se realiza desde el listado de opositores.



Un formulario que muestra el nombre "Alejandra Morales Villafana" a la izquierda y tres botones a la derecha: "Orden" (gris), "Validar" (azul) y "Rechazar" (rojo).

Una vez validado un opositor, ya podrá acceder a la convocatoria.

En caso de ser rechazado, tendrá que solicitar de nuevo el acceso a la convocatoria.

## 7.2.9 Listado de plazas


Desde el listado de plazas, como organizador se pueden crear, modificar y borrar las plazas existentes en la convocatoria.

Como opositor se pueden consultar los detalles de las plazas, ordenando o filtrando por cualquiera de los criterios.

Desde esta ventana se puede acceder a la selección de plazas y acceder al resultado de la simulación.

### 7.2.10 Añadir una plaza

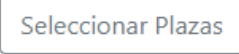
Como organizador, una de las tareas a realizar es la creación de las plazas de la convocatoria. Esta tarea se realiza desde la ventana del listado de plazas,

pinchando en el botón 

En el formulario de creación de plazas podemos rellenar los datos que estimemos necesarios, ya que no existen datos obligatorios, esta peculiaridad se debe a que según el organismo que convoque el proceso selectivo las plazas pueden tener o no ciertos datos, queda a libre disposición del organizador estimar que campos serán necesarios.

### 7.2.11 Selección de plazas

Para la selección de plazas debemos acceder desde la ventana de plazas, en el

botón 

En la siguiente ventana se muestra una tabla con cada uno de los órdenes de la selección de plazas que debemos hacer, inicialmente todos vacíos.

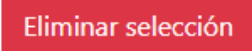
1	Seleccionar plaza
---	-------------------

Pincharemos sobre cualquiera de las líneas de la tabla, se abrirá un modal emergente que mostrará todas las plazas de la convocatoria. Si pinchamos en alguna de las plazas se elegirá como plaza para orden seleccionado y volveremos a la vista anterior, donde podremos ver que la línea ha sido ocupada por la plaza que hemos seleccionado.

1	91	HACIENDA Y ADMINISTRACIONES PÚBLICAS	SECRET. GRAL. HACIENDA Y ADMONES. PCAS.	Toledo	TOLEDO	AUXILIAR REGISTRO	15	8175.24
---	----	--------------------------------------	---	--------	--------	-------------------	----	---------

En caso de querer modificar una selección, basta con pinchar sobre la fila ocupada y seleccionar otra plaza.

En caso de querer seleccionar una plaza que ya hemos seleccionado en otro orden, primero deberemos eliminar la selección, pinchando en el orden

anterior, y pinchando en el botón  para eliminar la selección.

### 7.2.12 Consulta del resultado de la simulación

Se accede desde la ventana del listado de plazas, en el botón

Consultar resultado simulación

Aparecerá una ventana con el listado de las elecciones de plazas que hemos realizado, y un código de colores.

<b>Plaza no posible</b> Otro opositor con mejor orden tiene asignada esta plaza	<b>Plaza posible</b> Esta plaza es una de las posibles que puede obtener el opositor.	<b>Plaza descartada</b> Se ha asignado alguna de las plazas anteriores al opositor.
--	--	--

Cada plaza que hayamos elegido aparecerá en un color distinto, ya sea rojo, verde o gris.

El color **rojo** indica que otro opositor con mejor orden que el usuario ha elegido dicha plaza y le ha sido asignada, por tanto, no podremos optar por esa plaza.

El color **verde** indica que la plaza está en el rango de plazas asignables. Idealmente el rango de plazas asignables debería ser una sola plaza, pero dependiendo de la cantidad de opositores que no se hayan registrado en la plataforma y la cantidad de opositores que no hayan realizado su elección de plazas, tendremos un rango de plazas de incertidumbre.

Finalmente en color **gris** quedarán las plazas que hemos elegido en últimas posiciones, y por tanto no nos han sido asignadas, ya que se nos ha asignado alguna plaza mejor.