

**Universidad de Camagüey “Ignacio Agramontes Loynaz”**

**Facultad de Informática y Ciencias Exactas**

**Desarrollo de la plataforma de debate universitario “Debatiendo”**

**Autores:**

**Juan Ariel Estrada Moreno**

**Dariel Argüello Gutiérrez**

**Dainel Martínez Cañizares**

**Abimael Cobas Hernández**

**Raúl López Hernández**

**Daniels Alejandro Cordovés Figueredo**

**Curso 2024**

**Tabla de contenido**

[Introducción 7](#_Toc163068968)

[Capítulo 1. Modelo de negocio 10](#_Toc163068969)

[1.1 Descripción detallada del negocio 10](#_Toc163068970)

[1.2 Reglas del negocio 12](#_Toc163068971)

[1.3 Actores del negocio 14](#_Toc163068972)

[1.4 Trabajadores del negocio 14](#_Toc163068973)

[1.5 Diagrama de Casos de Uso del Negocio 15](#_Toc163068974)

[1.6 Casos de Uso del Negocio 15](#_Toc163068975)

[1.6.1 CUN1: Gestionar Debates 16](#_Toc163068976)

[1.6.1.1 Descripción literal del CUN1: Gestionar Debates 16](#_Toc163068977)

[1.6.1.2 Diagrama de actividades del CUN1: Gestionar Debate 17](#_Toc163068978)

[1.6.1.3 Diagrama de objetos del CUN1: Gestionar Debate 17](#_Toc163068979)

[1.6.2 CUN2: Gestionar Acuerdos 18](#_Toc163068980)

[1.6.2.1 Descripción literal del CUN2: Gestionar Acuerdos 18](#_Toc163068981)

[1.6.2.2 Diagrama de actividades del CUN2: Gestionar Acuerdos 19](#_Toc163068982)

[1.6.2.3 Diagrama de objetos del CUN2: Gestionar Acuerdos 19](#_Toc163068983)

[1.6.3 CUN3: Generar Acta de Debate 19](#_Toc163068984)

[1.6.3.1 Descripción literal del CUN3: Generar Acta de Debate 19](#_Toc163068985)

[1.6.3.2 Diagrama de actividades del CUN3: Generar Acta de Debate 20](#_Toc163068986)

[1.6.3.3 Diagrama de objetos del CUN3: Generar Acta de Debate 21](#_Toc163068987)

[1.6.4 CUN4: Generar Acta de Acuerdo 21](#_Toc163068988)

[1.6.4.1 Descripción literal del CUN4: Generar Acta de Acuerdo 21](#_Toc163068989)

[1.6.4.2 Diagrama de actividades del CUN4: Generar Acta de Acuerdo 22](#_Toc163068990)

[1.6.4.3 Diagrama de objetos del CUN4: Generar Acta de Acuerdo 22](#_Toc163068991)

[1.6.5 CUN5: Participación en Debate 22](#_Toc163068992)

[1.6.5.1 Descripción literal del CUN5: Participación en Debate 23](#_Toc163068993)

[1.6.5.2 Diagrama de actividades del CUN5: Participación en Debate 23](#_Toc163068994)

[1.6.5.3 Diagrama de objetos del CUN5: Participación en Debate 24](#_Toc163068995)

[1.6.6 CUN6: Votación en los acuerdos 24](#_Toc163068996)

[1.6.6.1 Descripción literal del CUN6: Votación en los acuerdos 24](#_Toc163068997)

[1.6.6.2 Diagrama de actividades del CUN6: Votación en los acuerdos 25](#_Toc163068998)

[1.6.6.3 Diagrama de objetos del CUN6: Votación en los acuerdos 25](#_Toc163068999)

[Capítulo 2. Propuesta de solución 26](#_Toc163069000)

[2.1 Visión del producto. 26](#_Toc163069001)

[2.2 Definición del producto 27](#_Toc163069002)

[2.2.1 Definición de los usuarios del sistema 27](#_Toc163069003)

[2.2.2 Requisitos funcionales 28](#_Toc163069004)

[2.2.3 Requisitos no funcionales 30](#_Toc163069005)

[2.2.4 Diagrama de Casos de Uso del sistema 32](#_Toc163069006)

[2.2.5 Descripción de los casos de uso del sistema 32](#_Toc163069007)

[2.2.6 Diagrama de paquetes del sistema 39](#_Toc163069008)

[2.3 Análisis y diseño del sistema 40](#_Toc163069009)

[2.3.1 Realizaciones de los Casos de Usos del sistema 40](#_Toc163069010)

[2.3.2 Descripción de la arquitectura 43](#_Toc163069011)

[2.3.3 Realizaciones de los Casos de Usos del Diseño 44](#_Toc163069012)

[2.3.4 Diagrama de secuencia 46](#_Toc163069013)

[2.3.5 Definición del Modelo de Datos 47](#_Toc163069014)

[2.4 Implementación del sistema 49](#_Toc163069015)

[2.4.1 Despliegue del sistema 49](#_Toc163069016)

[2.4.2 Diagrama de componentes 50](#_Toc163069017)

[Capítulo 3. Validación de la propuesta 52](#_Toc163069018)

[3.1 Análisis de Factibilidad 52](#_Toc163069019)

[3.2 Planificación temporal y seguimiento del proyecto 53](#_Toc163069020)

[3.2.1 Principios generales de la planificación 53](#_Toc163069021)

[3.2.2 Actividades a planificar, hitos y fases del proyecto 54](#_Toc163069022)

[3.3 Prototipo del Sistema 56](#_Toc163069023)

[3.3.1 Forma general y principios de la protección y seguridad 58](#_Toc163069024)

[3.3.2 Forma general del tratamiento de errores. 59](#_Toc163069025)

[3.4 Diseño y resultado de la Estrategia de Pruebas 61](#_Toc163069026)

[3.4.1 Estrategia de prueba 62](#_Toc163069027)

[3.4.2 Resultados de las pruebas 65](#_Toc163069028)

[Conclusiones 67](#_Toc163069029)

[Recomendaciones 68](#_Toc163069030)

[Bibliografía 69](#_Toc163069031)

[Anexos 70](#_Toc163069032)

**Índice de Tablas**

[Tabla 1: Estimación de puntos por CU 52](#_Toc163067827)

[Tabla 2: Fases del proyecto y sus iteraciones 55](#_Toc163067828)

[Tabla 3: Las fases y sus hitos. 55](#_Toc163067829)

[Tabla 4: Estrategia de Pruebas 62](#_Toc163067830)

[Tabla 5: Caso de Prueba Crear Debate 63](#_Toc163067831)

[Tabla 6: Caso de Prueba Recuperar Acuerdo 63](#_Toc163067832)

[Tabla 7: Caso de Prueba Generar Acta de Acuerdo 64](#_Toc163067833)

[Tabla 8: Caso de prueba Opinar 64](#_Toc163067834)

**Índice de Ilustraciones**

[Ilustración 1: Diagrama de Caso de Uso del Sistema 32](#_Toc163067790)

[Ilustración 2: Prototipo de interfaz Gestionar Debate 34](#_Toc163067791)

[Ilustración 3: Prototipo de interfaz Gestionar Acuerdo 36](#_Toc163067792)

[Ilustración 4: Prototipo de Interfaz Generar Acta de Acuerdo 37](#_Toc163067793)

[Ilustración 5: Prototipo de interfaz Gestionar Acuerdo 39](#_Toc163067794)

[Ilustración 6: Diagrama de Paquetes del Sistema 40](#_Toc163067795)

[Ilustración 7: Diagrama de clases del análisis del CU Gestionar Debate 40](#_Toc163067796)

[Ilustración 8: Diagrama de Comunicación del CU Gestionar Debate (Crear Debate) 41](#_Toc163067797)

[Ilustración 9: Diagrama de clases del análisis del CU Gestionar Acuerdo 41](#_Toc163067798)

[Ilustración 10: Diagrama de Comunicación del CU Gestionar Acuerdo (Recuperar Acuerdo) 41](#_Toc163067799)

[Ilustración 11: Diagrama de clases del análisis del CU Generar Acta de Acuerdo 42](#_Toc163067800)

[Ilustración 12: Diagrama de Comunicación del CU Generar Acta de Acuerdo 42](#_Toc163067801)

[Ilustración 13: Diagrama de clases del análisis del CU Participación en Debates 42](#_Toc163067802)

[Ilustración 14: Diagrama de Comunicación del CU Participación en Debate 43](#_Toc163067803)

[Ilustración 15: Diseño arquitectónico de la propuesta de solución basado en MVT 43](#_Toc163067804)

[Ilustración 16: Diagrama de CUD de Gestionar Debate 44](#_Toc163067805)

[Ilustración 17: Diagrama de CUD de Gestionar Acuerdos 44](#_Toc163067806)

[Ilustración 18: Diagrama de CUD de Generar Acta de Acuerdo 45](#_Toc163067807)

[Ilustración 19: Diagrama de CUD de Participación en Debate 45](#_Toc163067808)

[Ilustración 20: Diagrama de Secuencia del CU Gestionar Debate (Crear Debate) 46](#_Toc163067809)

[Ilustración 21:Diagrama de Secuencia del CU Gestionar Acuerdo (Recuperar Acuerdo) 46](#_Toc163067810)

[Ilustración 22:Diagrama de Secuencia del CU Generar Acta de Acuerdo (Generar Acta de Acuerdo) 47](#_Toc163067811)

[Ilustración 23: Diagrama de Secuencia del CU Participar en Debate (Opinar) 47](#_Toc163067812)

[Ilustración 24: Diagrama de Clases Persistentes 48](#_Toc163067813)

[Ilustración 25: Diagrama de Entidad Relación 48](#_Toc163067814)

[Ilustración 26: Diagrama de Despliegue del sistema 49](#_Toc163067815)

[Ilustración 27: Diagrama de Componentes 50](#_Toc163067816)

[Ilustración 28: Prototipo de Pantalla Principal 56](#_Toc163067817)

[Ilustración 29: Prototipo de Pantalla en rol usuario 1 57](#_Toc163067818)

[Ilustración 30: Prototipo de Pantalla en rol de usuario 2 57](#_Toc163067819)

[Ilustración 31: Prototipo de Participar en Debate 58](#_Toc163067820)

[Ilustración 32: Mensaje de error 60](#_Toc163067821)

[Ilustración 33: Mensaje de advertencia 60](#_Toc163067822)

[Ilustración 34: Mensaje de información 61](#_Toc163067823)

[Ilustración 35: Resultados de las pruebas funcionales 65](#_Toc163067824)

[Ilustración 36: Prueba 1 65](#_Toc163067825)

[Ilustración 37: Prueba 2 66](#_Toc163067826)

# Introducción

El debate en una comunidad universitaria es una herramienta fundamental para fomentar el intercambio de ideas, el pensamiento crítico y el desarrollo de habilidades comunicativas. La importancia del debate radica en la posibilidad de exponer diferentes puntos de vista, confrontar opiniones y llegar a consensos a través del diálogo constructivo.

La Universidad de Camagüey está compuesta por 10 facultades, 52 carreras, 117 proyectos y una matrícula de más de 15000 estudiantes. La comunicación efectiva es esencial en este contexto, ya que permite que los participantes expresen sus ideas de manera clara y argumentada, escuchen activamente a los demás y busquen soluciones en conjunto. El debate en una comunidad universitaria no solo enriquece el aprendizaje académico, sino que también promueve la tolerancia, el respeto y la empatía entre sus miembros.

La comunidad universitaria de la Universidad de Camagüey carece de una plataforma efectiva para generar y llevar a cabo debates sobre temas específicos de interés. La ausencia de un sistema centralizado y estructurado ha ocasionado dificultades para facilitar la participación activa de los miembros de la comunidad universitaria, expresar sus opiniones, llevar a cabo debates abiertos y cerrados, así como tomar acuerdos en base a los votos de los participantes.

Esto ha resultado en una falta de seguimiento y registro adecuado de las opiniones emitidas, así como de los acuerdos alcanzados durante los debates, lo que ha impactado negativamente en la toma de decisiones y en la transparencia dentro de la comunidad.

Algunos de los desafíos específicos incluyen:

1. **Falta de Plataforma Efectiva para Debates:** La falta de una plataforma específica ha generado dificultades para planificar, organizar y realizar debates centrados en temas de interés para la comunidad universitaria. La ausencia de esta herramienta ha impedido la participación significativa y estructurada en la generación de opiniones colectivas.
2. **Registro y Seguimiento de Opiniones:** La falta de un sistema centralizado ha impedido el registro y la sistematización adecuada de las opiniones emitidas durante los debates, limitando la capacidad de documentar y hacer un seguimiento claro de las posturas expresadas por los participantes.
3. **Falta de Mecanismo de Toma de Acuerdos:** La carencia de un mecanismo específico para tomar acuerdos en base a los votos de los participantes ha generado dificultades para alcanzar consensos basados en opiniones colectivas.

La situación problemática anterior da paso al siguiente **problema a resolver**: ¿cómo contribuir al proceso de debate y toma de acuerdos sobre temas de interés para la comunidad universitaria?

Lo cual nos plantea el **objetivo general**: desarrollar una aplicación web que permita la generación de debates abiertos y cerrados, el registro de actas de debates, seguimiento de opiniones y la toma de acuerdos en base a votos. Desglosándose las siguientes **tareas investigativas:**

* Realización de la fundamentación teórica como resultado de la revisión bibliográfica e investigación del estado del arte.
* Definición de los aspectos que distinguirán la propuesta de solución.
* Levantamiento de los requisitos funcionales y no funcionales.
* Planificación del proceso de desarrollo de la aplicación web.
* Diseño e implementación de las nuevas funcionalidades de la aplicación web.
* Realización de las pruebas de calidad a la aplicación web.

Para el desarrollo del trabajo se emplearon los siguientes métodos de investigación:

Métodos teóricos:

* **Análisis Sintético**: Permitió realizar un estudio sobre los elementos más importantes relacionados con el proceso de debates abiertos y cerrados y toma de acuerdos, para posteriormente sintetizar la información necesaria para la gestión de debates y acuerdos.
* **Análisis Inductivo-Deductivo**: Permitió emplear los fundamentos teóricos más generales relacionados con los procesos de la gestión de debates y toma de acuerdos enriqueciendo el marco teórico referencial de la investigación con los conocimientos surgidos a partir de la experiencia particular.
* **Modelación**: Se utilizó para representar por medio de diagramas caso de uso del negocio. Lo que permitirá una mayor comprensión de los procesos y lineamientos de desarrollo a seguir en cada una de las fases, a partir de su utilización para el modelado del sistema.

Métodos empíricos:

* **Observación**: Se usó para verificar el proceso de debate y toma de acuerdos realizados actualmente en la comunidad universitaria, para extraer los parámetros fundamentales a tener en cuenta.
* **Entrevista**: Con este método fue posible la recolección de información útil a la investigación a través de entrevistas realizadas a la comunidad universitaria, especialmente el estudiantado.

El documento está conformado por Resumen, Introducción, tres capítulos, Recomendaciones, Bibliografía, Glosario de términos y Anexos. A continuación, se realiza un breve resumen de cada capítulo:

**Capítulo 1. Modelado del negocio:** en este capítulo se realiza una descripción detallada del negocio, sus reglas y los actores que influyen en él. Se confecciona el diagrama de caso de uso del negocio (CUN) y la descripción de los casos de uso del negocio junto a su diagrama de actividades y de objetos.

**Capítulo 2. Propuesta de solución:** en este capítulo se abarca la visión del producto y su definición. Se enuncian los usuarios del sistema y se denotan los requisitos funcionales y no funcionales del sistema. Se diagrama los casos de uso del sistema (CUS), su descripción y el diagrama de paquetes de datos del sistema. Se realiza el análisis y diseño del sistema, el cual engloba la confección del diagrama de clases del análisis, casos de uso del diseño (CUD), descripción de la arquitectura y modelado del modelo de datos. Para finalizar se realiza el diagrama de componentes (DC) y el despliegue del sistema (DS).

**Capítulo 3. Validación de la repuesta:** se realiza el análisis de factibilidad junto con la planificación temporal y seguimiento del proyecto. Se diseña el prototipo del sistema y se llega a la conclusión de las estrategias de pruebas y sus resultados.

# Modelo de negocio

## Descripción detallada del negocio

El modelo de negocio planteado trata de una plataforma donde se puedan debatir temas de interés para la comunidad universitaria. Además, en estos debates se pueden llevar a cabo acuerdos, que se deciden en base a una votación hecho por los participantes. Dividido en procesos fundamentales seria:

**Proceso de creación de Debate:**

1. Participantes:

* Moderador: Persona encargada de crear los temas de debate

1. Producto:

* Debate: Espacio para que los participantes en el debate puedan dar sus opiniones y llegar a consensos.

1. Pasos para elaborarlo:

* El moderador genera un nuevo debate, eligiendo el tema a tratar y el tipo de debate que será: abierto o cerrado.
* Si es un debate cerrado, deberá elegir los participantes del mismo.

**Proceso de Participación en Debate:**

1. Participantes:

* Participante: Participantes principales en los debates.
* Moderador: Controla el curso del debate y sus resultados.

1. Productos:

* Opiniones: Un debate estará conformado por las opiniones emitidas por los participantes.
* Proponer un acuerdo
* Crear encuesta
* Participar en una encuesta
* Bloquear a un participante

1. Pasos para elaborarlo:

* Si poseen permiso para participar, los estudiantes podrán emitir opiniones sobre el tema propuesto y debatir entre ellos.
* Podrán responder directamente en el tema principal o responderse directamente en el espacio de debate.
* Cuando el moderador considere que el tiempo de debate ha terminado, cerrara el tema de debate.

1. Reglas

**Proceso de Generación de Acta de Debate:**

1. Participantes:

* Moderador: Se encarga de elaborar el acta de debate después que un tema ha sido cerrado.

1. Producto:

* Acta de debate: Documento que guarda todas las opiniones emitidas por los estudiantes en el debate,votaciones y aprobaciones de acuerdos.

1. Pasos para elaborarlo:

* Cuando se cierra un debate, el moderador puede generar un acta de debate, que guardara todos los datos del debate: tema a tratar, total de participantes, fecha de creada y de cierre, opiniones omitidas.
* Se guardará en un documento en la base de datos para poder ser exportado o leído de necesitarse.

**Proceso de Generación de Acuerdos**

1. Participantes:

* Participante: Persona encargada de crear el tema a acordar en el debate y el recuento de votos.

1. Productos:

* Acuerdo: Se guarda las votaciones creadas, su total de participantes y fecha de creación y fin.

1. Pasos para elaborarlo:

* Un moderador podrá crear un tema para votar en un debate creado previamente o de manera individual.
* Se elige los participantes capaces de votar.

**Proceso de Votación en los Acuerdos:**

1. Participantes:

* Participantes: Dan su voto en el tema a decidir en el acuerdo
* Moderador: Controla el curso del acuerdo y sus resultados.

1. Productos:

* Votación: A partir de unas opciones dadas, los participantes pueden votar la opción preferida.

1. Pasos para elaborarlo:

* Los participantes cuentan con un voto único para cada tema.
* Los moderadores después de un tiempo acordado cerrarán las votaciones y emitirán los resultados.
* Moderador: Se encarga de elaborar el acta de acuerdo después de una El modelo de negocio planteando busca garantizar el seguimiento y almacenamiento de las opiniones emitidas por los participantes de la comunidad. Garantiza que los acuerdos sean ms efectivos y optimiza el tiempo al cual se llega a una conclusión y respuesta.

## Reglas del negocio

Las reglas de negocio guían la toma de decisiones diarias dentro de las empresas trazando las relaciones entre los objetos, como los nombres de cliente y sus pedidos correspondientes. Esta conversión de las actividades de negocio de una organización en lógica empresarial concreta permite a los ingenieros de software y analistas de negocio aplicar estas reglas en herramientas de flujos de trabajo u otras aplicaciones para habilitar la automatización de procesos. Sin ellas, los procesos de actualización pueden volverse más pesados y lentos, y los documentos están más expuestos a error humano e inconsistencias. Una empresa que implementa reglas de negocio puede ahorrar tiempo y dinero gracias a la optimización del trabajo y un menor abandono (IBM, 2024).

* **Reglas de estructura:**
* Termino:
* Debate, participante, Acuerdo, Moderador, votación
* Modelo de datos:
* La cantidad de moderadores de un debate debe ser mayor que 0.
* Relación:
* Entre los participantes se generan debates.
* Un acuerdo ocurre entre los participantes que votan.
* Un acuerdo puede ocurrir durante un debate.
* **Reglas de derivación:**
* Inferencia:
* Un acuerdo terminado genera un acta .
* Cálculo:
* El resultado de un acuerdo se calcula sumando los votos de cada una de las opciones dadas y luego comparándolas entre sí, quedándose con el mayor.
* **Reglas de acción:**
* Flujo:
* Un Participante genera un debate y/o acuerdo
* Los participantes pueden opinar y votar con un voto único en el tema del debate o acuerdo creado.
* El moderador finaliza el debate y/o acuerdo cuando considere que haya pasado el tiempo necesario.
* **Reglas de validación:**
* Restricción de operaciones:
* Para que una votación sea aprobada debe tener más de 1 voto.
* Estímulo y respuesta:
* Cuando hayan votado todos los participantes en el acuerdo, se le notificara al moderador para que lo finalice y de los resultados.

Estas reglas rigen las políticas y condiciones que posee el negocio a informatizar, aportando claridad en los procesos y en los actores que influyen en ellos.

## Actores del negocio

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados (Gonzáles, 2004).

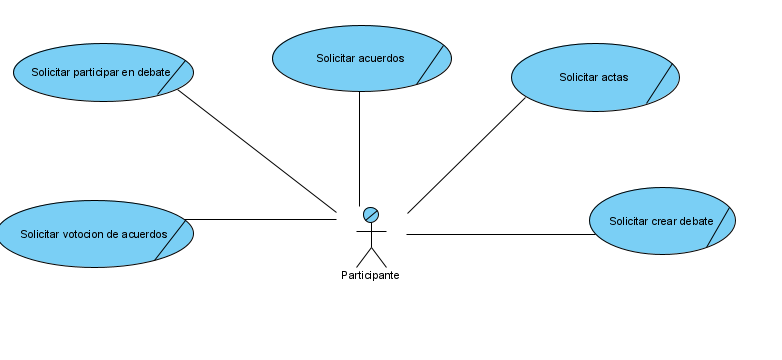
|  |  |
| --- | --- |
| **Actor** | **Descripción** |
| **Participante** | Puede dar opiniones, responder debates y votar en los acuerdos creados. |

## Trabajadores del negocio

Representa a una persona, conjunto de personas, hardware, software y a cualquier componente que interactúa con la solución informática basado en una necesidad (Mueras, 2010).

|  |  |
| --- | --- |
| **Trabajador** | **Descripción** |
| **Moderador** | Encargado de gestionar los usuarios del sistema y sus roles asignados. |

## Diagrama de Casos de Uso del Negocio



## Casos de Uso del Negocio

El modelo de casos de uso del negocio es un modelo que describe los procesos de un negocio (casos de uso del negocio) y su interacción con elementos externos, tales como socios y clientes, es decir, describe las funciones que el negocio pretende realizar y su objetivo básico es describir cómo el negocio es utilizado por sus clientes y socios. Implica la determinación de los actores y casos de uso del negocio.

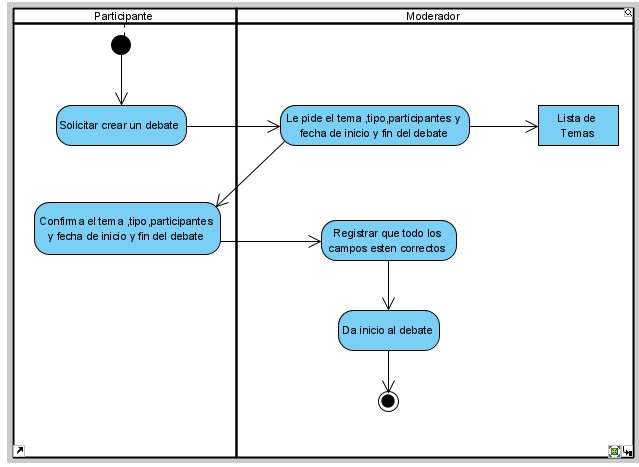
Con esta actividad se pretende: Identificar los procesos en el negocio, definir las fronteras del negocio que van a modelarse, Definir quién y qué interactuarán con el negocio y crear diagramas del modelo de casos de uso del negocio (Gonzáles, 2004).

### CUN1: Solicitar Crear Debates

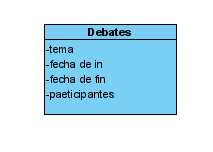
### Descripción literal del CUN1: Solicitar Crear Debates

|  |  |
| --- | --- |
| **Caso de uso del negocio** | Solicitar Crear Debates |
| Actores | Participante |
| Resumen | El Participante solicita crear un tema de debate. Luego este es aprobado o no por el moderador, y una vez que los participantes hayan dado su opinión, el moderador cierra el tema y guarda la información en un acta. |
| Casos de uso asociados | Solicitar Acta |
| **Acción del actor** | **Respuesta del proceso de negocio** |
| 1. El Participante solicita crear un debate.   3. El participante confirma el tema, tipo, participantes y fecha de inicio y fin del debate. | 1. El moderador le da a elegir el tema, tipo, participantes y fecha de inicio y fin del debate.   4. El moderador verifica que todos los campos este correcto.  5. Da inicio al debate. |
|  |  |
| Otras secciones | Si el debate no posee opiniones, no se crea un acta. |

### Diagrama de actividades del CUN1: Solicitar Crear Debate



### Diagrama de objetos del C0055N1: Solicitar Crear Debate

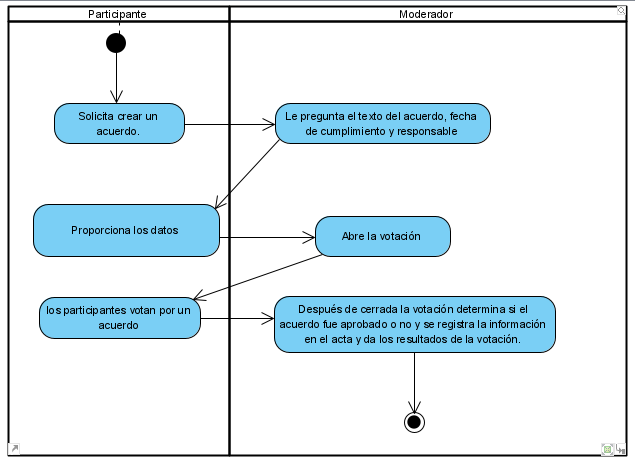


### CUN2: Solicitar Acuerdos

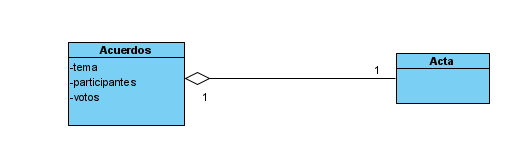
### Descripción literal del CUN2: Solicitar Acuerdos

|  |  |
| --- | --- |
| **Caso de uso del negocio** | **Solicitar Acuerdos** |
| Actores | Participante |
| Resumen | El Participante solicita crear un acuerdo para llevarlo a votación. El moderador pregunta el texto del acuerdo, fecha de cumplimiento y responsable. El participante proporciona los datos y el moderador abre la votación para que los participantes digan si están de acuerdo o no. Después de cerrada la votación determina si el acuerdo fue aprobado o no y se registra la información en el acta y da los resultados de la votación. |
| Casos de uso asociados | Solicitar Acta. |
| **Acción del actor** | **Respuesta del proceso de negocio** |
| 1. El participante solicita crear un acuerdo.   3. El participante proporciona los datos  3.1 Para que los participantes digan si están de acuerdo o no. | 2. El moderador pregunta el texto del acuerdo, fecha de cumplimiento y responsable  4. El moderador abre la votación  5. Después de cerrada la votación determina si el acuerdo fue aprobado o no y se registra la información en el acta y da los resultados de la votación. |
|  |  |
| Otras secciones | Si no hay votos, no se genera acta |

### Diagrama de actividades del CUN2: Solicitar Acuerdos



### Diagrama de objetos del CUN2: Solicitar Acuerdos

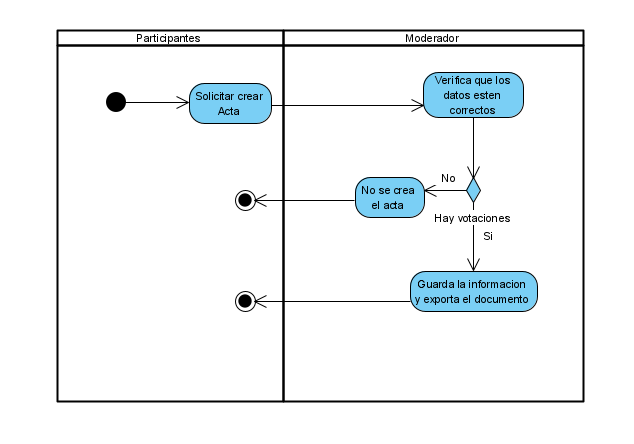


### CUN3: Solicitar Acta.

### Descripción literal del CUN3: Solicitar Acta

|  |  |
| --- | --- |
| **Caso de uso del negocio** | **Solicitar Acta** |
| Actores | Participante |
| Resumen | El participante solicita el acta cuando se cierra un debate, el moderador encargado del mismo, genera un acta con todos los datos del debate y todas las opiniones emitidas. Si se efectuó un acuerdo durante el mismo también estará reflejado en el acta. |
| Casos de uso asociados | Solicitar crear Debate, Solicitar acuerdos |
| **Acción del actor** | **Respuesta del proceso de negocio** |
| 1. El participante solicita un acta. | 2. Cierra el tema de debate.  3.Guarda la información en un documento PDF con el nombre del tema del debate realizado y los acuerdos tomados  4. Exporta el documento. |
| Otras secciones | Si el debate no posee opiniones, no se crea un acta. |

### Diagrama de actividades del CUN3: Solicitar Acta

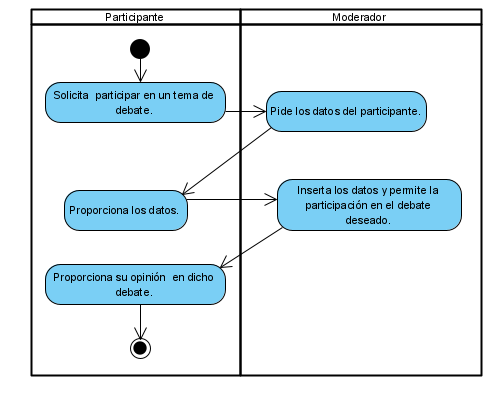


### CUN4: Solicitar participar en debate

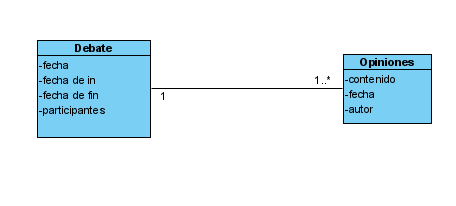
### Descripción literal del CUN4: Solicitar participar en debate

|  |  |
| --- | --- |
| **Caso de uso del negocio** | Solicitar participar en debate |
| Actores | Participante |
| Resumen | El participante puede opinar en un tema creado de su interés. Puede responder directamente al tema, o responderse a otros usuarios. |
| Casos de uso asociados | Solicitar crear Debate |
| **Acción del actor** | **Respuesta del proceso de negocio** |
| 1. El participante solicita participar en un tema de debate. | 2. El moderador pide los datos del participante.  4. El moderador inserta los datos y permite la participación en el debate deseado. |
| 3. El participante proporciona los datos.  5. Proporciona su opinión en dicho debate. |  |
| Otras secciones | N/A |

### Diagrama de actividades del CUN4: Solicitar Participar en Debate



### Diagrama de objetos del CU4: Solicitar Participar en Debate

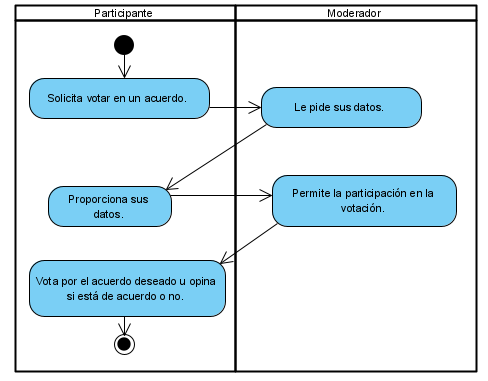


### CUN5: Solicitar Votación de acuerdos

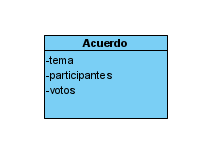
### Descripción literal del CUN5: Solicitar Votación de acuerdos

|  |  |
| --- | --- |
| **Caso de uso del negocio** | Solicitar Votación de acuerdos |
| Actores | Participante |
| Resumen | El Participante solicita votar en un acuerdo creado. El moderador le pide sus datos y luego permite la participación en la votación, luego el participante vota por el acuerdo deseado u opina si está de acuerdo o no. |
| Casos de uso asociados | Solicitar Acuerdos, Crear Debate |
| **Acción del actor** | **Respuesta del proceso de negocio** |
| 1. El participante solicita votar en un acuerdo.   3. El participante proporciona sus datos.  5.El participante vota por el acuerdo deseado u opina si está de acuerdo o no. | 2. El moderador le pide sus datos.  4. Permite la participación en la votación. |
| Otras secciones | N/A |

### Diagrama de actividades del CUN5: Solicitar Votación de acuerdos



### Diagrama de objetos del CUN5: Solicitar Votación de acuerdos



# Propuesta de solución

## Visión del producto.

El producto a desarrollar es una plataforma informática que permita que sus usuarios debatan en base a un tema en específico de interés y que se puedan llevar a cabo acuerdos de manera rápida y colectiva de forma más rápida y mayoritaria. Se basará en un sistema con una interfaz amigable y fácil de usar para el cliente, donde las funciones fundamentales serán fácilmente accesibles.

Se plantea el desarrollo de una aplicación web para abordar estos desafíos, empleando Django en el backend y NextJS en el frontend. Esta plataforma tecnológica permitirá una implementación modular y una arquitectura escalable que pueda adaptarse a la complejidad de las funcionalidades requeridas para este sistema de gestión de debates en la comunidad universitaria.

Esta plataforma tecnológica propuesta, implementada con un enfoque modular y de alta capacidad escalable, permitirá cumplir con las necesidades actuales y futuras de la comunidad universitaria de la Universidad de Camagüey en lo que respecta a la generación de debates proactivos, la toma de acuerdos y la documentación de decisiones colectivas.

**Clientes u usuarios potenciales del producto:**

* Comunidad en línea en necesidad de debatir y llegar a acuerdos de forma rápida y organizada.
* Empresa que quiera tener una plataforma donde sus trabajadores puedan hacer reuniones virtuales sobre un orden del día y poder tomar una decisión colectiva.

**Necesidad del cliente que genera la necesidad del producto:**

* Falta de una herramienta para debatir entre la comunidad universitaria sobre temas de interés y tomar decisiones de forma colectiva y transparentes.
* Falta de un medio para documentar las decisiones que puedan afectar a la comunidad y su vida diaria.

**Nombre del producto:**

* Debatiendo.

**Tipo de producto y caracterización:**

* Aplicación web de gestión de debates y toma de acuerdos, que genera documentación de los mismos.

**Factores distintivos:**

* Estructura modular: El proyecto contará con módulos claramente definidos para la gestión de debates, toma de acuerdos y documentación de decisiones.
* Tecnología empleada: La combinación de Django en el backend y NextJS en el frontend permitirá una arquitectura robusta y escalable.
* Complejidad del proyecto: La necesidad de gestionar debates y acuerdos dentro de una institución educativa como la Universidad de Camagüey implica un alto nivel de complejidad en términos de funcionalidades y requerimientos.

**Principales beneficios que obtendrá el cliente con el producto y características o funcionalidades del producto que lo soportan:**

* Facilitación de la toma de decisiones colectivas.
* Documentación transparente y accesible de acuerdos y debates.
* Mejora en la comunicación y participación dentro de la comunidad universitaria.

**Perspectivas de este producto relativo a otros módulos, productos y/o sistemas.**

Este producto se posiciona como un componente clave para mejorar la gestión interna y la participación en decisiones dentro de la Universidad de Camagüey. En términos de interacción con otros sistemas, se podrían identificar interfaces relevantes con sistemas de gestión académica, sistemas de comunicación interna y plataformas de colaboración existentes en la universidad. La integración efectiva con estos sistemas permitirá una experiencia más completa y eficiente para los usuarios finales.

## Definición del producto

### Definición de los usuarios del sistema

1. **Participante:**

* Descripción: Usuario que utiliza el sistema para acceder a los debates, participar en los acuerdos ,dar opiniones y votar
* Justificación: Los estudiantes son el usuario principal del sistema, ya que su objetivo principal es facilitarle el acceso a la información y participar en debates y acuerdos

1. **Moderador:**

* Descripción: Usuario encargado de supervisar y gestionar el contenido generado por los participantes, garantizando que cumpla con las normas y políticas establecidas. Gestiona todo lo referente a los debates y acuerdos en el sistema.
* Justificación: Los moderadores son necesarios para mantener un ambiente seguro y respetuoso en la plataforma, evitando la difusión de contenido inapropiado o perjudicial. Son los encargados de manejar los temas tratados en la plataforma y el control de los acuerdos llegados durante los debates.

**Administrador del sistema**

* Descripción: Usuario con permisos avanzados para configurar y administrar el sistema, gestionar usuarios, realizar copias de seguridad, entre otras tareas técnicas.
* Justificación: El administrador de sistema es fundamental para garantizar el correcto funcionamiento de la plataforma, así como para realizar mantenimiento y actualizaciones necesarias.

**Jerarquía:**

**1. Administrador del sistema**

2. Moderador

3. Participante

Esta jerarquía establece que el administrador de sistema tiene el mayor nivel de acceso y control sobre el sistema, seguido por los moderadores que tienen la responsabilidad de supervisar el contenido generado por los estudiantes y la gestión de los debates y acuerdos. Por último, los estudiantes tienen un nivel básico de acceso para utilizar las funcionalidades del sistema.

### Requisitos funcionales

Los requisitos funcionales (RF) de un sistema, son aquellos que describen cualquier actividad que este deba realizar, en otras palabras, el comportamiento o función particular de un sistema cuando se cumplen ciertas condiciones. Por lo general, estos deben incluir funciones desempeñadas por pantallas específicas, descripciones de los flujos de trabajo a ser desempeñados por el sistema y otros requerimientos de negocio (Pressman, 2013).

1. **Autenticar usuario**
2. **Actualizar perfil de usuario**
3. **Gestionar usuario**
   1. Crear usuario
   2. Listar usuarios
   3. Gestionar roles de usuario
   4. Gestionar permisos de usuario
   5. Eliminar usuarios
4. **Gestionar Debate** 
   1. Crear debate
   2. Modificar debate
   3. Eliminar debate
   4. Listar debate
   5. Detallar debate
5. **Gestionar Acuerdo**
   1. Crear acuerdo
   2. Modificar acuerdo
   3. Eliminar acuerdo
   4. Listar acuerdo
   5. Detallar acuerdo
6. **Participación en Debates**
   1. Crear comentarios y opiniones
   2. Eliminar comentarios y opiniones
   3. Modificar comentarios y opiniones
7. **Sistema de Votación**
   1. Votación en acuerdo
   2. Contabilizar votos
   3. Mostrar resultados de votación
   4. Registro de votos
8. **Registro de opiniones**
   1. Almacenar opiniones
   2. Organizar opiniones por debate
   3. Buscar opiniones
9. **Notificaciones y recordatorios**
   1. Notificar recordatorios sobre debates y acuerdos
   2. Configuración de preferencia de notificación
   3. Alertar sobre participación en debates o acuerdos nuevos

### Requisitos no funcionales

#### Usabilidad:

**RNF 1: Tiempo de Entrenamiento:**

Descripción: El sistema debe ser intuitivo y fácil de usar para que los usuarios normales y avanzados puedan operarlo productivamente con un tiempo de entrenamiento mínimo.

#### Fiabilidad:

**RNF 2: Disponibilidad del Sistema:**

Descripción: El sistema debe tener una disponibilidad del 99.9% para garantizar que esté disponible la mayor parte del tiempo para los usuarios. Se deben establecer horarios de mantenimiento programado para minimizar interrupciones.

**RNF 3: Tiempo Medio Entre Fallos (MTBF):**

Descripción: Se debe especificar un MTBF que indique cuánto tiempo en promedio puede funcionar el sistema sin presentar fallas. Esto garantiza la estabilidad y confiabilidad del sistema.

#### Eficiencia:

**RNF 4: Tiempo de Respuesta por Transacción:**

Descripción: El sistema debe tener un tiempo de respuesta promedio por transacción de menos de 2 segundos para mantener una experiencia fluida y eficiente para los usuarios.

**RNF 5: Rendimiento del Sistema:**

Descripción: Se debe especificar el rendimiento del sistema en términos de transacciones por segundo para garantizar que pueda manejar un alto volumen de usuarios simultáneos sin degradar su funcionamiento.

#### Soporte:

**RNF 6: Normas de Codificación y Mantenimiento:**

Descripción: Se deben establecer normas de codificación claras y buenas prácticas de mantenimiento para facilitar futuras actualizaciones y correcciones en el software.

#### Restricciones de diseño:

**RNF 7: Lenguaje de programación y tecnologías a utilizar:**

Descripción: El sistema debe ser desarrollado utilizando el lenguaje de programación Python y tecnologías asociadas como Django para el backend y React para el frontend. Debe usar una metodología RUP y un servidor web para el hosting de la aplicación.

#### Requisitos para la documentación de usuarios en línea y ayuda del sistema:

**RNF 8: Documentación en línea:**

Descripción: Se debe proporcionar una documentación en línea detallada para los usuarios, que incluya manuales de usuario, tutoriales y preguntas frecuentes para facilitar su uso del sistema.

#### Interfaz:

##### **Interfaces de usuario:**

**RNF 9: Interfaz de usuario amigable:**

Descripción: La interfaz de usuario debe ser intuitiva, amigable y escalable para adaptarse a diferentes dispositivos y pantallas.

##### **Interfaces Software:**

**RNF 10: Integración con servicios externos:**

Descripción: El sistema debe poder integrarse con servicios externos como sistemas de autenticación para garantizar la seguridad y la interoperabilidad.

#### Requisitos de Licencia:

**RNF 11: Requisitos de licencia:**

Descripción: El software desarrollado debe cumplir con las licencias de código abierto utilizadas en los componentes de terceros, y se deben respetar los términos de uso correspondientes.

#### Requisitos Legales, de Derecho de Autor y otros:

**RNF 12: Cumplimiento legal:**

Descripción: El sistema debe cumplir con las leyes de protección de datos, privacidad y cualquier otra normativa legal aplicable en el ámbito de la plataforma de debates universitarios.

#### Estándares Aplicables:

**RNF 13: Estándares de seguridad:**

Descripción: El sistema debe cumplir con los estándares de seguridad establecidos para proteger la información sensible de los usuarios y garantizar la confidencialidad y la integridad de los datos.

### Diagrama de Casos de Uso del sistema

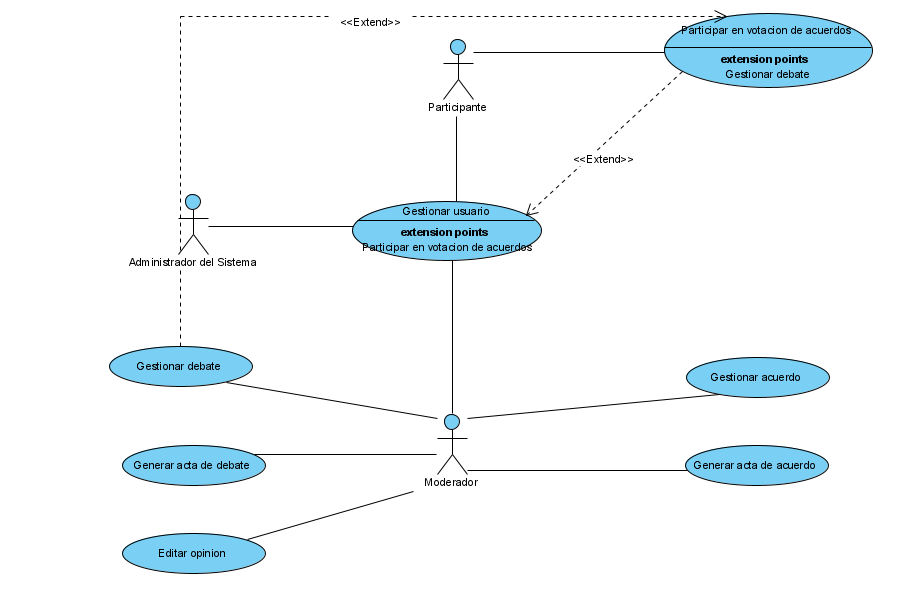


Ilustración 1: Diagrama de Caso de Uso del Sistema

### Descripción de los casos de uso del sistema

|  |  |  |
| --- | --- | --- |
| **Caso de Uso:** | Gestionar Debate | |
| **Actores:** | Participante | |
| **Resumen:** | Este caso de uso permite a los participantes gestionar debates, creando, recuperando, actualizando y eliminando debates. | |
| **Precondiciones:** | El sistema debe tener al menos un debate creado. | |
| **Referencias** | Requerimientos funcionales relacionados con la gestión de debates. | |
| **Prioridad** | Crítico | |
| **Flujo Normal de Eventos** | | |
| **Sección “Crear Debate”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 1. El participante solicita crear un nuevo debate. | | 1.1 El sistema muestra el registro con los datos a llenar del nuevo debate en la base de datos. |
| 2.El participante introduce los datos requeridos | | 2.1- El sistema valida que los datos entrados estén completos y correctos.  2.2 El sistema registra el nuevo debate en la base de datos. |
| **Sección “Recuperar Debate”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 3. El participante busca un debate existente. | | 3.1 El sistema muestra la información del debate solicitado. |
| **Sección “Actualizar Debate”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 4. El participante modifica un debate existente. | | 4.1 Muestra una plantilla solicitando los datos que se desea modificar. |
| 5.El participante modifica la plantilla con nuevos datos.  5. El participante cancela la modificación | | 5.1 El sistema valida que los datos modificados estén completos y correctos.  5.2 Flujo alterno 1 |
| **Sección “Eliminar Debate”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 6. El participante elimina un debate existente. | | 6.1 Muestra un mensaje que si está seguro que desea eliminar el debate |
| 7.El participante confirma la eliminación  7.El participante cancela la eliminación | | 7.1 El sistema elimina el debate de la base de datos.  7.2 Flujo alterno 1 |
| **Flujos Alternos** | | |
| F1: Se vuelve a la pantalla de lista de debate  E1: Si los datos adicionados no son válidos, el sistema envía un mensaje de error  E2: Si el debate que se intenta recuperar, actualizar o eliminar no existe, se muestra un mensaje de error. | | |
| **Poscondiciones** | El debate se ha creado, actualizado o eliminado correctamente en el sistema. | |
| Prototipo de Interfaz    Ilustración 2: Prototipo de interfaz Gestionar Debate | | |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso:** | Gestionar Acuerdo | |
| **Actores:** | Participante | |
| **Resumen:** | Este caso de uso permite a los moderadores gestionar acuerdos, creando, recuperando, actualizando y eliminando acuerdos. | |
| **Precondiciones:** | El sistema debe tener al menos un acuerdo creado. | |
| **Referencias** | Requerimientos funcionales relacionados con la gestión de acuerdos. | |
| **Prioridad** | Alta | |
| **Flujo Normal de Eventos** | | |
| **Sección “Crear Acuerdo”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 1. El participante solicita crear un nuevo acuerdo. | | 1.1 El sistema muestra el registro con los datos a llenar del nuevo acuerdo en la base de datos. |
| 2.El participante introduce los datos requeridos | | 2.1- El sistema valida que los datos entrados estén completos y correctos.  2.2 El sistema registra el nuevo acuerdo en la base de datos. |
| **Sección “Recuperar Acuerdo”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 3. El participante busca un acuerdo existente. | | 3.1 El sistema muestra la información del acuerdo solicitado. |
| **Sección “Actualizar Acuerdo”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 4. El participante modifica un acuerdo existente. | | 4.1 Muestra una plantilla solicitando los datos que se desea modificar. |
| 5.El participante modifica la plantilla con nuevos datos.  5. El participante cancela la modificación | | 5.1 El sistema valida que los datos modificados estén completos y correctos.  5.2 Flujo alterno 1 |
| **Sección “Eliminar Acuerdo”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 6. El participante elimina un acuerdo existente. | | 6.1 Muestra un mensaje que si está seguro que desea eliminar el acuerdo |
| 7.El participante confirma la eliminación  7.El participante cancela la eliminación | | 7.1 El sistema elimina el acuerdo de la base de datos.  7.2 Flujo alterno 1 |
| **Flujos Alternos** | | |
| F1: Se vuelve a la pantalla de lista de acuerdo  E1: Si los datos adicionados no son válidos, el sistema envía un mensaje de error  E2: Si el acuerdo que se intenta recuperar, actualizar o eliminar no existe, se muestra un mensaje de error. | | |
| **Poscondiciones** | El acuerdo se ha creado, actualizado o eliminado correctamente en el sistema. | |
| Prototipo de Interfaz    Ilustración 3: Prototipo de interfaz Gestionar Acuerdo | | |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso:** | Generar Acta | |
| **Actores:** | Participante | |
| **Resumen:** | Este caso de uso describe el proceso mediante el cual se genera un acta basada en los votos y opiniones emitidas durante un debate,acuerdo. | |
| **Precondiciones:** | Los participantes han emitido sus votos y opiniones durante el acuerdo. | |
| **Referencias** | Requerimientos funcionales relacionados con la generación de actas de acuerdo. | |
| **Prioridad** | Crítico | |
| **Flujo Normal de Eventos** | | |
| **Sección “Generar Acta ”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 1. El moderador selecciona la opción para generar un acta de acuerdo. | | 1.1 El sistema recopila automáticamente los votos y opiniones emitidas durante el debate y genera un documento de acta de acuerdo. |
| **Sección “Eliminar Acta ”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 6. El moderador elimina un acta de acuerdo. | | 6.1 Muestra un mensaje que si está seguro que desea eliminar el acuerdo |
| 7.El moderador confirma la eliminación  7.El moderador cancela la eliminación | | 7.1 El sistema elimina el acta seleccionada de la lista de actas generadas.  7.2 Flujo alterno F1 |
| **Flujos Alternos** | | |
| F1: Se vuelve a la pantalla de lista de Acta de acuerdo generadas  E1: Si los datos adicionados no son válidos, el sistema envía un mensaje de error | | |
| **Poscondiciones** | Se ha generado un documento de acta de acuerdo basado en los votos y opiniones emitidas durante el debate. | |
| Prototipo de Interfaz    Ilustración 4: Prototipo de Interfaz Generar Acta | | |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso:** | Participar en Debate | |
| **Actores:** | Participante | |
| **Resumen:** | El caso de uso describe el proceso mediante el cual un participante se une a un debate específico dentro de la plataforma, expresando su opinión y votando en base a las propuestas presentadas. | |
| **Precondiciones:** | 1. El participante ha iniciado sesión en la plataforma.  2. Existe un debate activo al cual el participante desea unirse. | |
| **Referencias** | Caso de Uso relacionado: Gestionar Debate | |
| **Prioridad** | Alta | |
| **Flujo Normal de Eventos** | | |
| **Sección “Opinar”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 1. El estudiante selecciona el debate al que desea unirse. | | 1.1 1.1 El sistema muestra la información del debate seleccionado |
| 2.El estudiante opina en la caja de opiniones | | 2.1- El sistema valida que los datos entrados estén completos y correctos.  2.2 El sistema registra la nueva opinión en la base de datos. |
| **Sección “Recuperar Opinión”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 3. El estudiante busca una opinión especifica en el debate | | 3.1 El sistema muestra la información del acuerdo solicitado. |
| **Sección “Responder”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 4. El participante visualiza las opiniones expresadas por otros participantes. | | 4.1 El sistema muestra las opiniones registradas durante el debate |
| 5.El estudiante ingresa una respuesta a una opinión antes emitida. | | 5.1 El sistema valida que los datos estén completos y correctos.  5.2 El sistema guarda la respuesta como una nueva opinión. |
| **Sección “Eliminar Opinión”** | | |
| **Acción del Actor** | | **Respuesta del sistema** |
| 6. El estudiante decide eliminar una opinión | | 6.1 Muestra un mensaje que si está seguro que desea eliminar la opinión |
| 7.El estudiante confirma la eliminación  7.El estudiante cancela la eliminación | | 7.1 El sistema elimina la opinión de la base de datos.  7.2 Flujo alterno 1 |
| **Flujos Alternos** | | |
| F1: Se vuelve a la pantalla de debate actual.  E1: Si los datos adicionados no son válidos, el sistema envía un mensaje de error  E2: El participante no puede emitir su opinión si no ha ingresado al debate. | | |
| **Poscondiciones** | - La opinión del participante se registra en el debate.  - El sistema actualiza la lista de opiniones y participantes del debate. | |
| Prototipo de Interfaz    Ilustración 5: Prototipo de interfaz Participar en Debate | | |

### Diagrama de paquetes del sistema

Los diagramas de paquetes son diagramas estructurales que se emplean para mostrar la organización y disposición de diversos elementos de un modelo en forma de paquetes. Se usan con frecuencia para proporcionar una organización visual de la arquitectura en capas dentro de cualquier clasificador UML, por ejemplo, un sistema de software (Lucidchart, 2024).

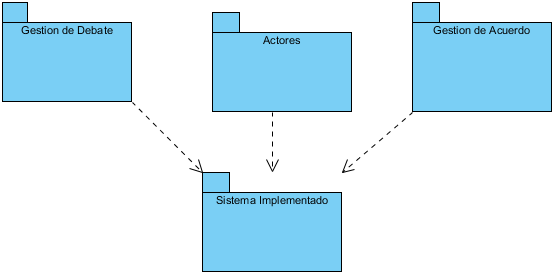


Ilustración 6: Diagrama de Paquetes del Sistema

## Análisis y diseño del sistema

### Realizaciones de los Casos de Usos del sistema

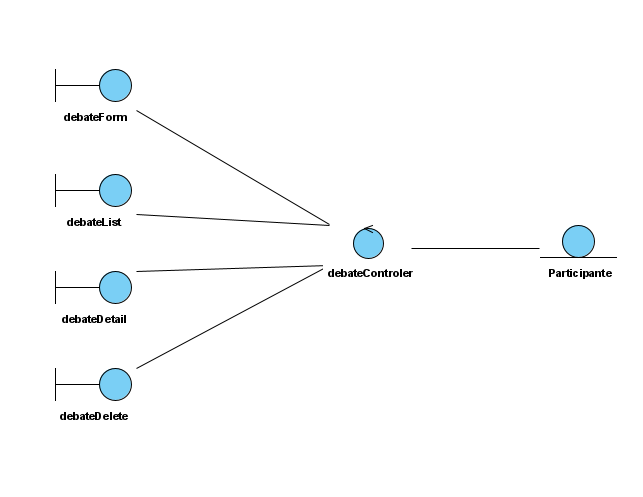


Ilustración 7: Diagrama de clases del análisis del CU Gestionar Debate

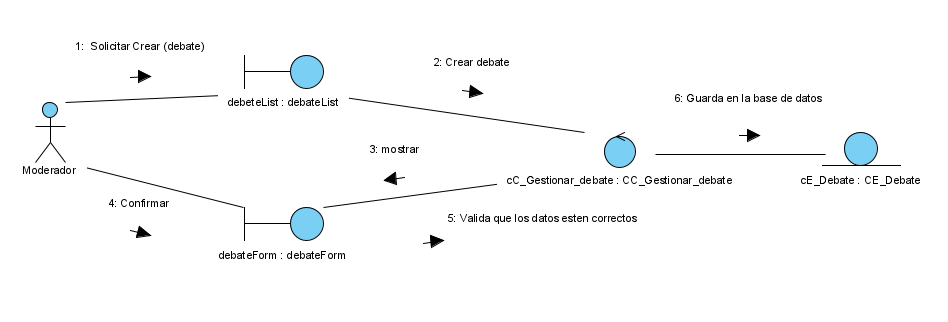


Ilustración 8: Diagrama de Comunicación del CU Gestionar Debate (Crear Debate)

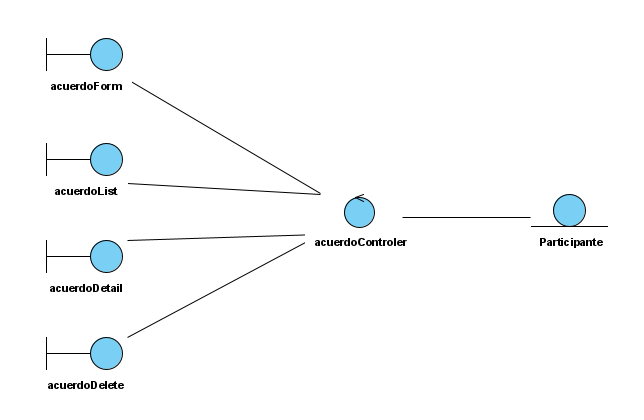


Ilustración 9: Diagrama de clases del análisis del CU Gestionar Acuerdo

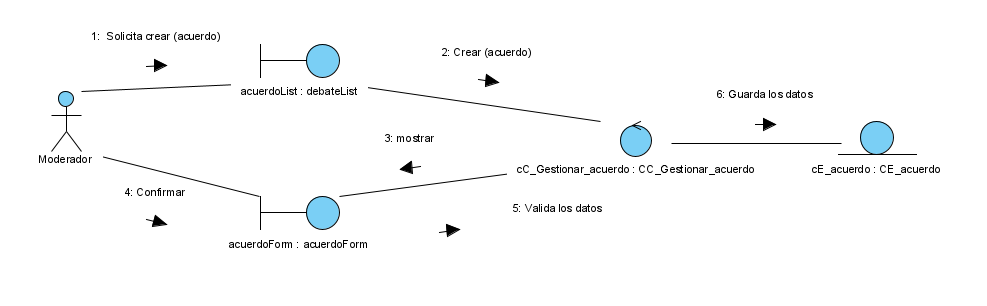


Ilustración 10: Diagrama de Comunicación del CU Gestionar Acuerdo (Crear Acuerdo)

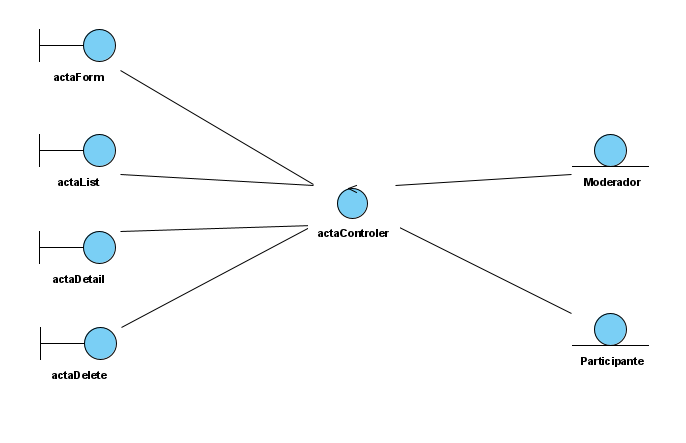


Ilustración 11: Diagrama de clases del análisis del CU Generar Acta (Eliminar acta)

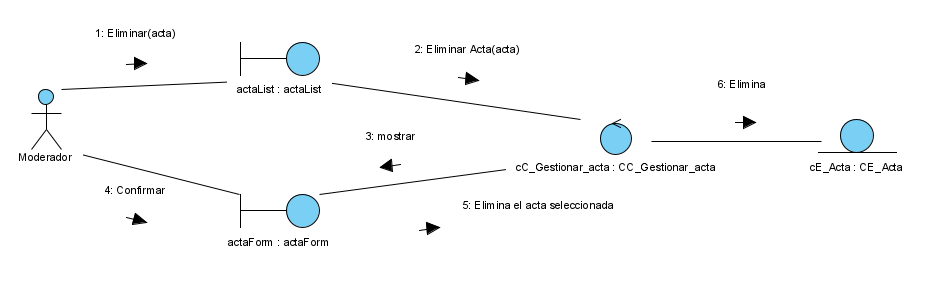


Ilustración 12: Diagrama de Comunicación del CU Generar Acta (Eliminar acta)

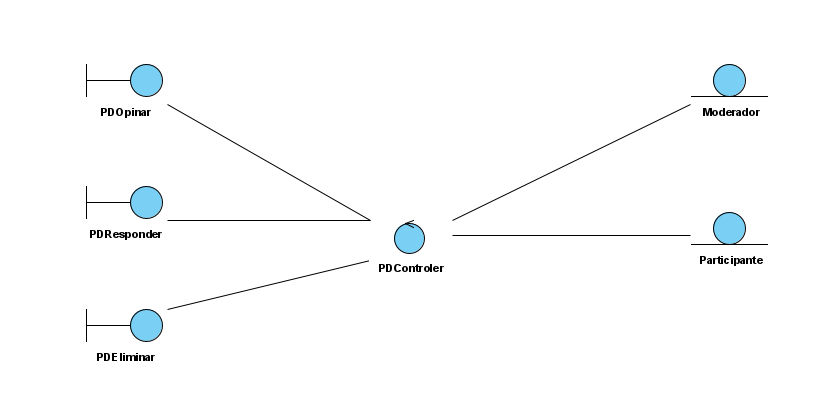


Ilustración 13: Diagrama de clases del análisis del CU Participación en Debates

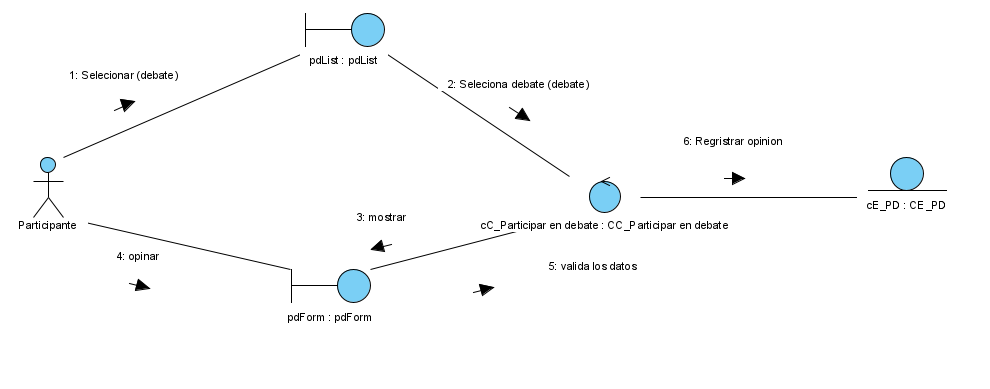


Ilustración 14: Diagrama de Comunicación del CU Participación en Debate (Opinar)

### Descripción de la arquitectura

La arquitectura de software es un conjunto de patrones que proporcionan un marco de referencia necesario para guiar la construcción de un software, permitiendo a los programadores, analistas y todo el conjunto de desarrolladores del software compartir una misma línea de trabajo y cubrir todos los objetivos y restricciones de la aplicación (Acevedo, 2015).

Se utilizó el patrón arquitectónico de “n capas” el cual divide una aplicación en capas lógicas y niveles físicos, para estructurar el software de la siguiente manera:

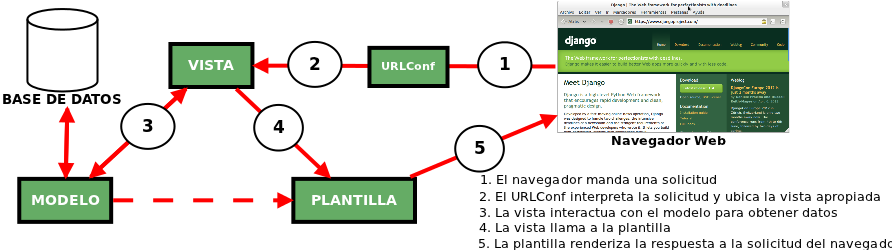


Ilustración 15: Diseño arquitectónico de la propuesta de solución basado en MVT

El patrón de arquitectura MVC convierte el desarrollo de aplicaciones complejas en un proceso mucho más manejable. Permite a varios desarrolladores trabajar simultáneamente en la aplicación.

* El **Modelo** es el que contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.
* La **Vista** **o interfaz de usuario** compone la información que se envía al cliente y los mecanismos interacción con éste.
* El **Controlador (*Template*)** es extraer, modificar y proporcionar datos al usuario. Esencialmente, el controlador es el enlace entre la vista y el modelo. A través de las funciones *getter* y *setter*, el controlador extrae datos del modelo e inicializa las vistas (Hernández, 2021).

### Realizaciones de los Casos de Usos del Diseño

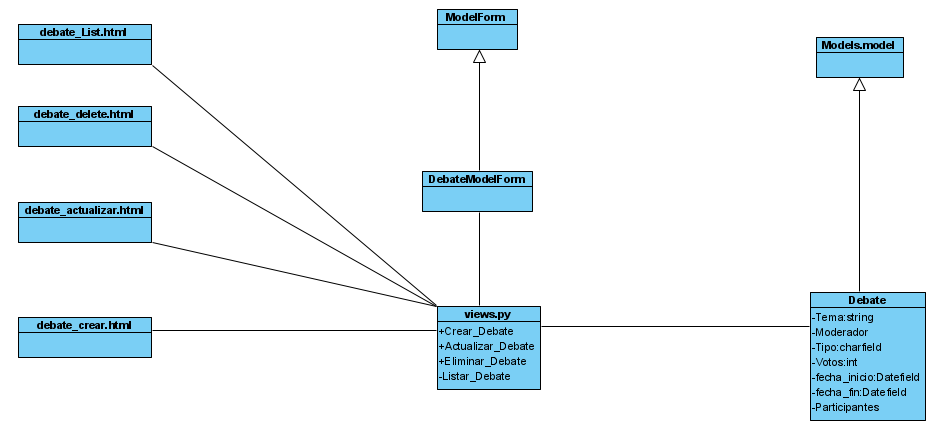


Ilustración 16: Diagrama de CUD de Gestionar Debate

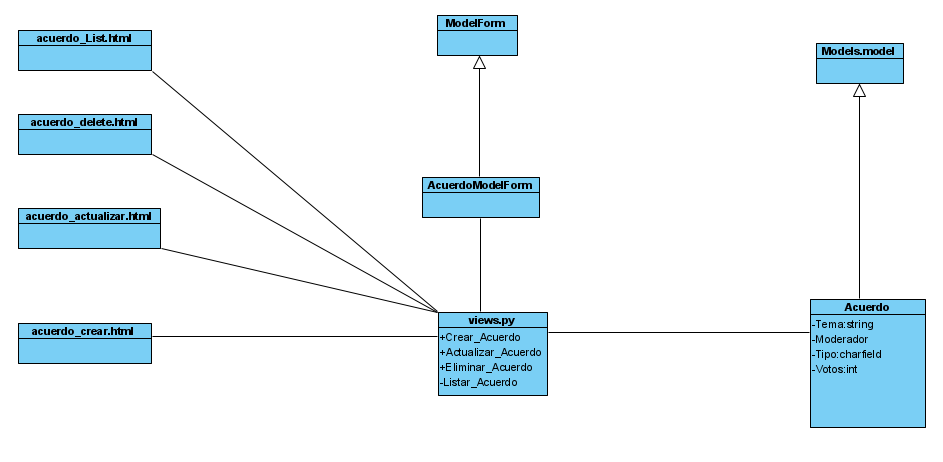


Ilustración 17: Diagrama de CUD de Gestionar Acuerdos

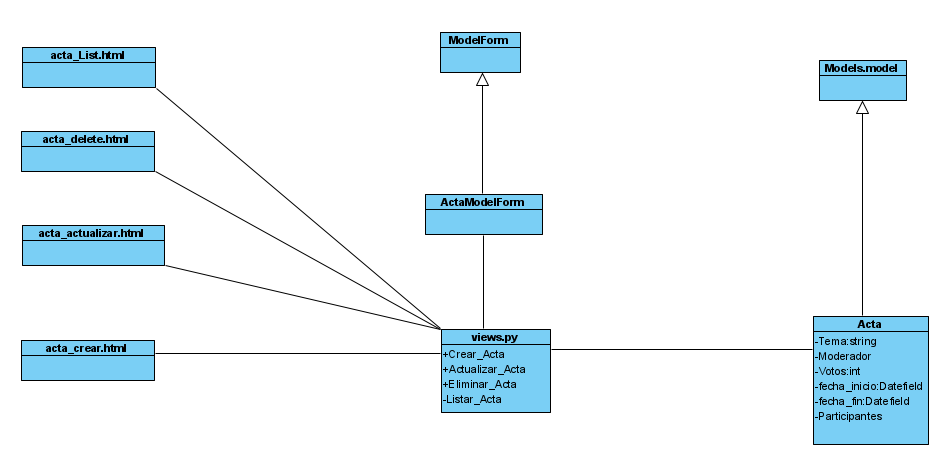
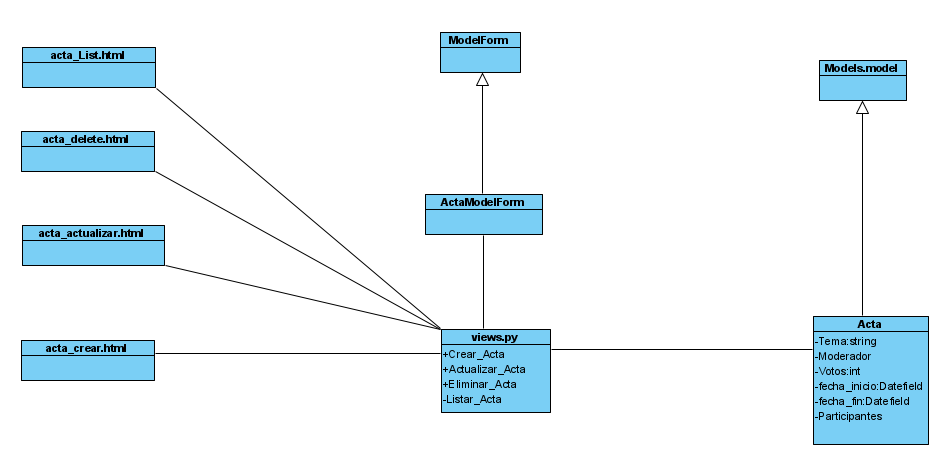


Ilustración 18: Diagrama de CUD de Generar Acta de Acuerdo

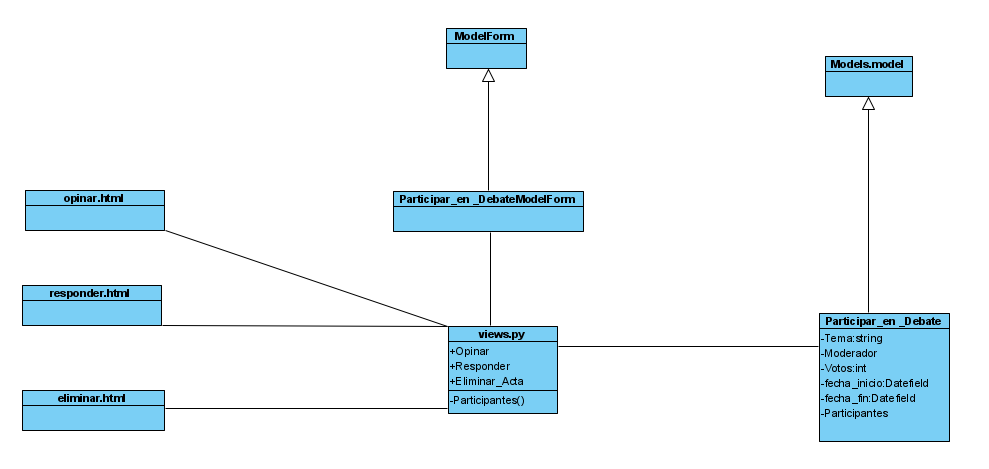


Ilustración 19: Diagrama de CUD de Participación en Debate

### Diagrama de secuencia

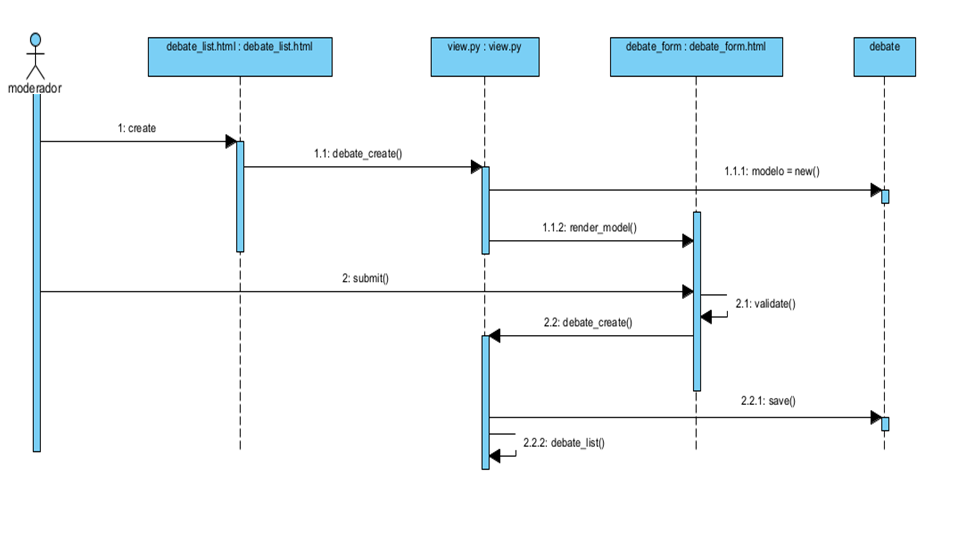


Ilustración 20: Diagrama de Secuencia del CU Gestionar Debate (Crear Debate)

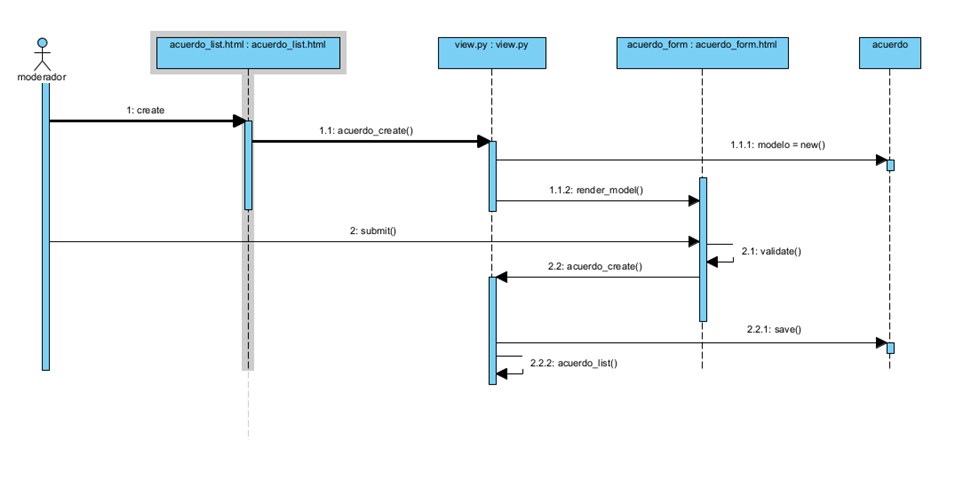


Ilustración 21: Diagrama de Secuencia del CU Gestionar Acuerdo (Crear Acuerdo)

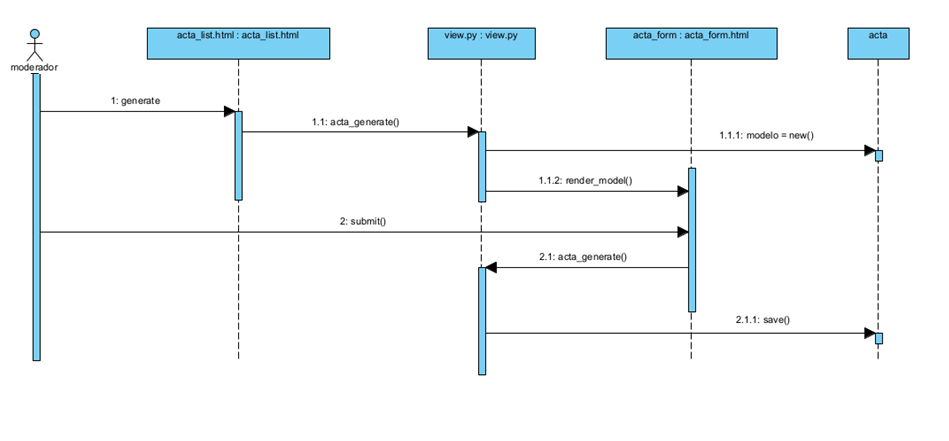


Ilustración 22: Diagrama de Secuencia del CU Generar Acta (Generar Acta)

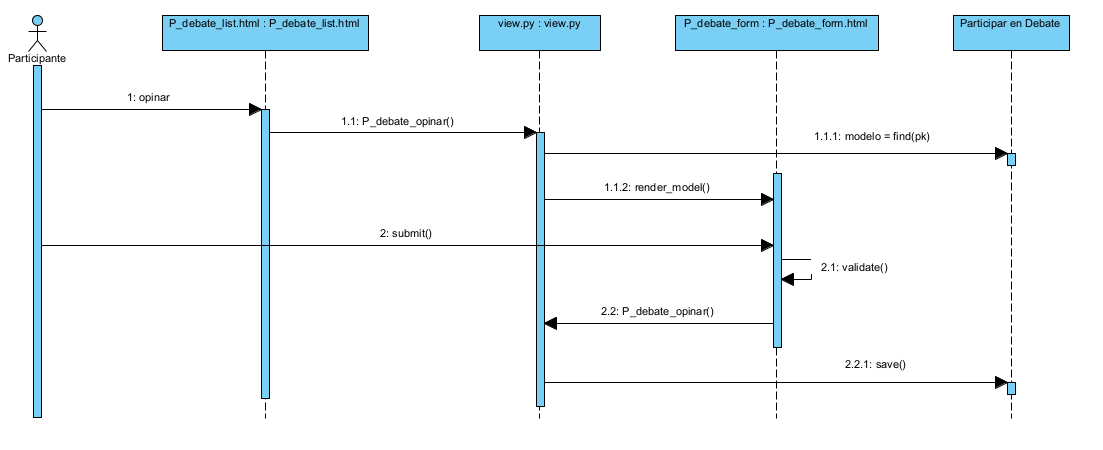


Ilustración 23: Diagrama de Secuencia del CU Participar en Debate (Opinar)

### Definición del Modelo de Datos

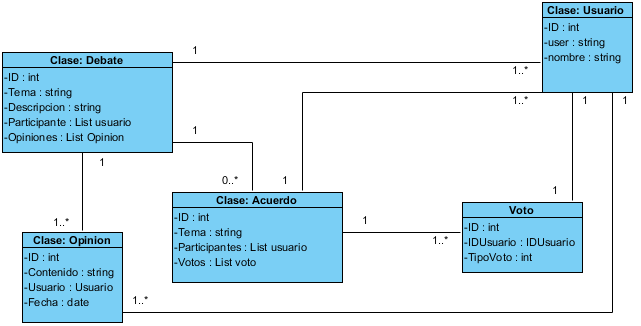


Ilustración 24: Diagrama de Clases Persistentes

Estas clases representan entidades clave en el sistema propuesto para facilitar debates y toma de decisiones dentro de la comunidad universitaria. Cada clase tiene atributos y métodos específicos para gestionar la información relevante y permitir la interacción entre los usuarios.

#### 2.3.5.1 Diagrama de Entidad Relación

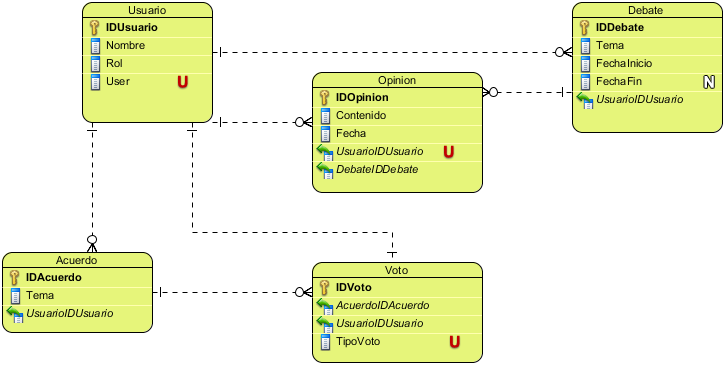


Ilustración 25: Diagrama de Entidad Relación

Este diagrama de entidad relación proporciona una visión clara de la estructura de datos del sistema propuesto y cómo se relacionan las diferentes entidades entre sí. Con esta representación visual, se facilita el diseño y la implementación de la base de datos necesaria para respaldar la plataforma de debates y toma de decisiones en la comunidad universitaria.

## Implementación del sistema

### Despliegue del sistema

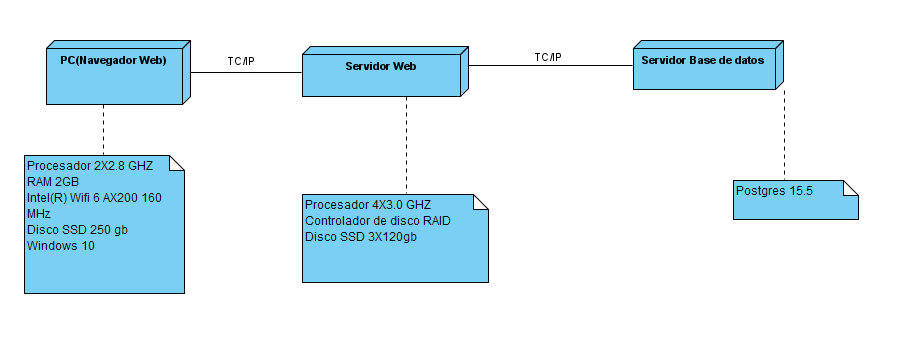


Ilustración 26: Diagrama de Despliegue del sistema

**Componentes físicos:**

* Cliente Web: Representa los dispositivos (computadoras, tablets, smartphones) desde los cuales los usuarios acceden a la plataforma a través de un navegador web.
* Servidor Web: Es el servidor donde se aloja la aplicación web de la plataforma. Aquí se ejecutan las operaciones del lado del servidor y se gestionan las solicitudes de los clientes.
* Base de Datos: Representa el sistema de gestión de base de datos donde se almacenan y recuperan los datos de la plataforma.

**Componentes lógicos:**

* Aplicación Web: Es la aplicación principal que se ejecuta en el servidor web y proporciona la interfaz de usuario para interactuar con la plataforma.
* Controlador de Base de Datos: Es el componente encargado de gestionar las operaciones de lectura y escritura en la base de datos, garantizando la integridad y seguridad de los datos.
* API REST: Es una interfaz que permite la comunicación entre la aplicación web y la base de datos, facilitando el intercambio de información de manera eficiente.

**Conexiones:**

* La aplicación web se comunica con el controlador de base de datos a través del API REST para acceder y manipular los datos almacenados en la base de datos.
* Los clientes web envían solicitudes al servidor web para interactuar con la aplicación web, y este a su vez se comunica con la base de datos para recuperar o almacenar información según sea necesario.

### Diagrama de componentes

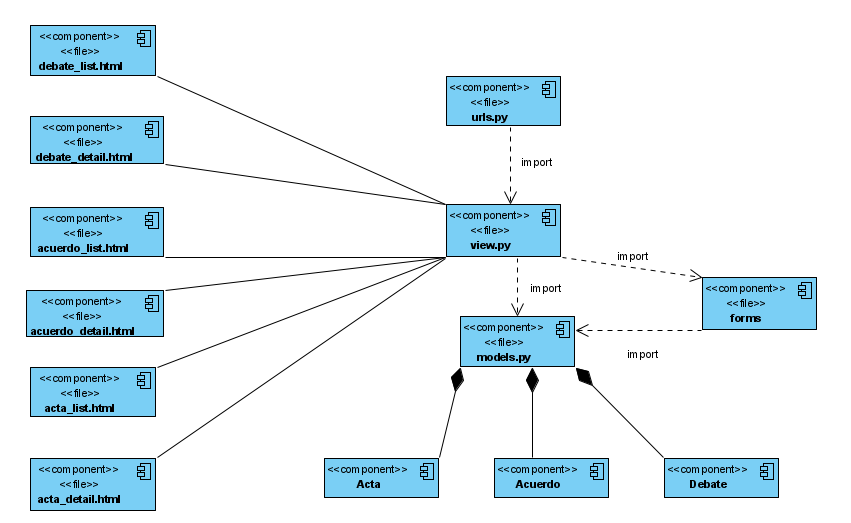


Ilustración 27: Diagrama de Componentes

**Componente Django:**

* Modelo: Define la estructura de la base de datos y las relaciones entre las entidades.
* Vista: Controla la lógica de negocio y la interacción con los modelos y las plantillas.
* Plantilla: Define la presentación visual de las páginas web generadas por la aplicación
* URLConf: Gestiona las rutas y redireccionamientos de las URLs dentro de la aplicación.

**Componente Python:**

* Controlador: Contiene la lógica de negocio que coordina la interacción entre los modelos y las vistas.
* Servicios: Encapsula funcionalidades específicas y reutilizables, como la gestión de usuarios, debates, opiniones, acuerdos, votos, etc.
* Utilidades: Contiene funciones auxiliares y herramientas para facilitar el desarrollo y mantenimiento del sistema.

**Componente Base de Datos:**

* Base de Datos Relacional: Utilizada para almacenar los datos de usuarios, debates, opiniones, acuerdos, votos, etc.
* Gestor de Base de Datos: Django se integra con diferentes gestores de base de datos como SQLite, MySQL, PostgreSQL, entre otros.

**Componente Interfaz de Usuario:**

* Frontend: Desarrollado con HTML, CSS y JavaScript para crear una interfaz amigable y atractiva para los usuarios.
* Interacción con el Backend: Se comunica con el backend a través de peticiones HTTP para obtener y enviar datos.

Este diagrama de componentes representa la arquitectura general del sistema basado en Django y Python, mostrando los diferentes módulos y tecnologías utilizadas en su desarrollo. Cada componente desempeña un papel importante en el funcionamiento del sistema y en la interacción entre las diferentes capas.

# Validación de la propuesta

## Análisis de Factibilidad

Para estimar los puntos por casos de uso en el desarrollo de esta plataforma, se puede aplicar el método de estimación de puntos de caso de uso. Este método consiste en asignar puntos a cada caso de uso basándose en la complejidad y el esfuerzo requerido para implementarlo. A continuación, se muestra una tabla de estimación de puntos por casos de uso:

Tabla 1: Estimación de puntos por CU

|  |  |  |
| --- | --- | --- |
| Casos de uso | Complejidad | Puntos estimados |
| Gestionar Debate | Alta | 8 |
| Gestionar Acuerdo | Alta | 8 |
| Generar Acta de Debate | Media | 5 |
| Generar Acta de Acuerdo | Media | 5 |
| Participar en Debate | Media | 5 |
| Votación en Acuerdos | Media | 5 |

**Gestionar Debate (Alta complejidad - 8 puntos):** Se considera un caso de uso de alta complejidad debido a que implica la creación, modificación y eliminación de debates, así como la gestión de participantes, temas y fechas. Requiere un alto nivel de interacción con la base de datos y la lógica de negocio, lo que implica un mayor esfuerzo de desarrollo.

**Gestionar Acuerdo (Alta complejidad - 8 puntos):** Similar al caso anterior, la gestión de acuerdos implica operaciones complejas como la creación, modificación y eliminación de acuerdos, así como la asignación de votos y seguimiento del estado de los acuerdos. Requiere una integración profunda con otros módulos del sistema y un manejo cuidadoso de la información.

**Generar Acta de Debate (Media complejidad - 5 puntos):** Aunque menos complejo que los anteriores, generar un acta de debate implica la recopilación y presentación estructurada de la información discutida durante el debate. Requiere un procesamiento adecuado de los datos para presentarlos de manera clara y comprensible.

**Generar Acta de Acuerdo (Media complejidad - 5 puntos):** Similar al caso anterior, generar un acta de acuerdo implica recopilar y presentar la información relevante sobre los acuerdos alcanzados durante las discusiones. Requiere un procesamiento específico para presentar los acuerdos de manera detallada y accesible.

**Participación en Debates (Media complejidad - 5 puntos):** Este caso de uso implica la interacción de los usuarios con los debates, incluyendo la visualización, participación activa y seguimiento de debates en curso. Requiere una interfaz intuitiva y eficiente para facilitar la participación de los usuarios.

**Votación en Acuerdos (Media complejidad - 5 puntos):** La votación en acuerdos implica la participación activa de los usuarios en la toma de decisiones mediante la votación sobre los acuerdos propuestos. Requiere un manejo preciso de los votos y un sistema seguro para garantizar la integridad del proceso de votación.

Estas estimaciones se basan en la complejidad percibida de cada caso de uso, considerando factores como la interacción con la base de datos, la lógica de negocio requerida, la integración con otros módulos del sistema y la experiencia del usuario. Es importante recordar que estas estimaciones son una guía inicial y pueden ajustarse a medida que se avanza en el desarrollo del proyecto y se obtiene más información sobre los requisitos específicos y la complejidad real de cada funcionalidad.

## Planificación temporal y seguimiento del proyecto

### Principios generales de la planificación

1. Identificación de los objetivos y alcance del proyecto.
2. Definición de las fases del proyecto: Inicio, Elaboración, Construcción y Transición.
3. Asignación de recursos y tiempos a cada actividad.
4. Establecimiento de hitos para medir el avance del proyecto.
5. Seguimiento continuo del progreso y ajuste de la planificación según sea necesario.

### Actividades a planificar, hitos y fases del proyecto

**Fase: Inicio**

1. Definición del alcance y objetivos del proyecto
2. Identificación de *stakeholders* y definición de roles
3. Elaboración del plan de proyecto
4. Creación del equipo de trabajo
5. Revisión y aprobación del plan de proyecto por el jefe de proyecto

**Hitos:**

* Aprobación del plan de proyecto por el jefe de proyecto

**Fase: Elaboración**

1. Definición detallada de requisitos del sistema
2. Diseño arquitectónico de la plataforma
3. Desarrollo de prototipos y maquetas
4. Planificación detallada de actividades y recursos para la fase de Construcción
5. Validación de requisitos con los *stakeholders*

**Hitos:**

* Aprobación de los requisitos del sistema
* Validación de los prototipos y maquetas con los *stakeholders*

**Fase: Construcción**

1. Implementación del software según diseño arquitectónico
2. Realización de pruebas de Caja Negra y Caja Blanca
3. Ajustes y correcciones en base a pruebas realizadas
4. Capacitación del personal en el uso de la plataforma
5. Preparación para la fase de Transición

**Hitos:**

* Finalización de la implementación del software
* Aprobación de las pruebas unitarias y de integración

**Fase: Transición**

1. Despliegue de la plataforma en entorno de producción
2. Monitoreo y soporte post-implementación
3. Evaluación del proyecto y lecciones aprendidas
4. Cierre del proyecto y entrega de documentación final

**Hitos:**

* Despliegue exitoso de la plataforma en entorno de producción
* Evaluación final del proyecto y lecciones aprendidas

Tabla 2: Fases del proyecto y sus iteraciones

|  |  |  |
| --- | --- | --- |
| **Fase** | **Número de Iteraciones** | **Finalización** |
| Inicio | 1 | Semana 04, 17/10/23 |
| Elaboración | 2 | Semana 09, 20/11/23 |
| Construcción | 3 | Semana 19, 11/03/24 |
| Transición | 4 | Semana 26, 08/04/24 |

Tabla 3: Las fases y sus hitos.

|  |  |  |
| --- | --- | --- |
| **Fase** | **Descripción** | **Hitos** |
| **Inicio** | Establecer el alcance y objetivos del proyecto, identificar stakeholders y roles, elaborar el plan de proyecto, formar el equipo y obtener la aprobación del plan por el comité de dirección. | 1. Aprobación del plan de proyecto por el jefe de proyecto. |
| **Elaboración** | Definir detalladamente los requisitos del sistema, diseñar la arquitectura, desarrollar prototipos, planificar actividades para la fase de Construcción y validar requisitos con stakeholders. | 2.Aprobación de los requisitos del sistema. 3.Validación de los prototipos y maquetas con los *stakeholders.* |
| **Construcción** | Implementar el software según diseño, realizar pruebas, ajustes y correcciones, capacitar al personal y prepararse para la Transición. | 4.Finalización de la implementación del software.  5.Aprobación de las pruebas unitarias y de integración. |
| **Transición** | Desplegar la plataforma en producción, brindar soporte post-implementación, evaluar el proyecto, aprender lecciones y cerrar el proyecto satisfactoriamente. | 6.Despliegue exitoso de la plataforma en entorno de producción.  7.Evaluación final del proyecto y lecciones aprendidas. |

## Prototipo del Sistema



Ilustración 28: Prototipo de Pantalla Principal

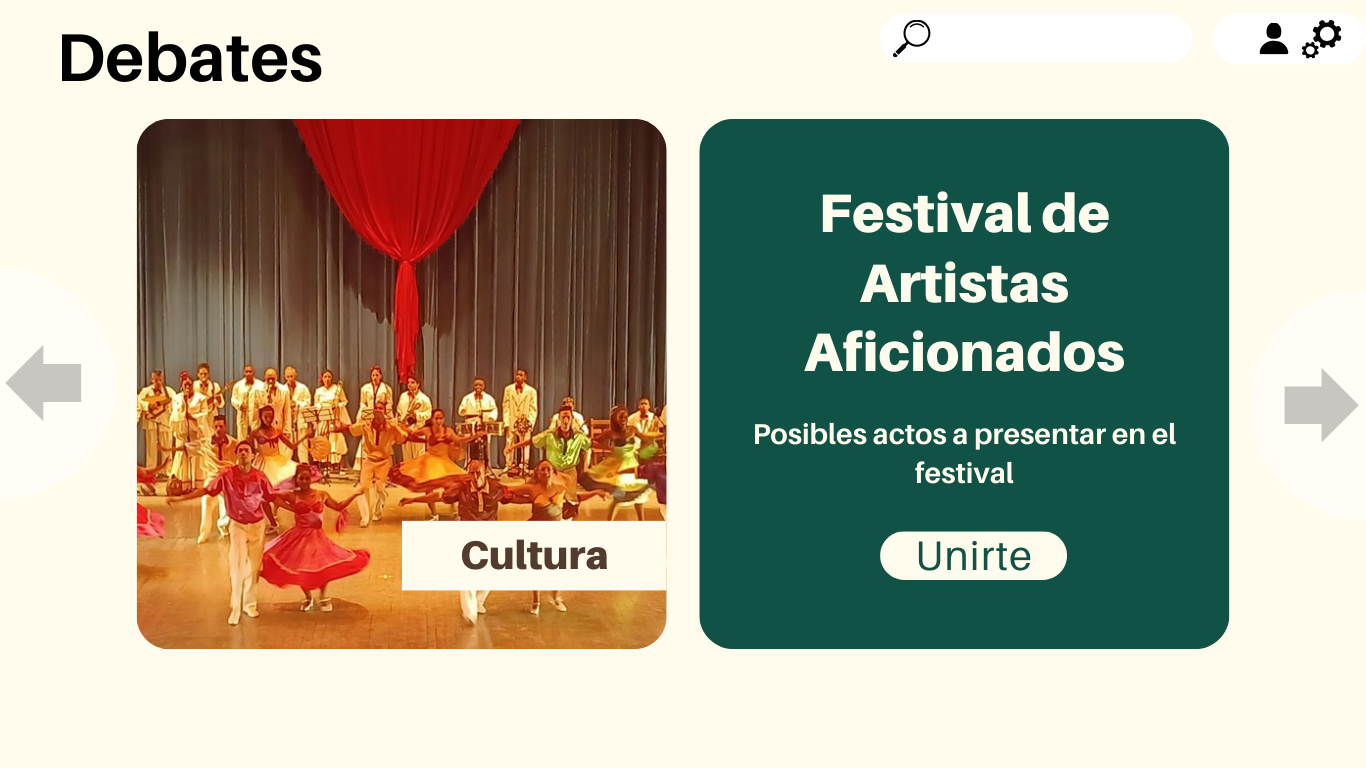


Ilustración 29: Prototipo de Pantalla en rol usuario 1



Ilustración 30: Prototipo de Pantalla en rol de usuario 2



Ilustración 31: Prototipo de Participar en Debate

### Forma general y principios de la protección y seguridad

**Seguridad hacia la interacción con los usuarios**:

1. **Autenticación y autorización:** Se implementan mecanismos de autenticación sólidos (contraseñas seguras, autenticación de dos factores, etc.) para verificar la identidad de los usuarios. Además, se establecen roles y permisos para controlar el acceso a diferentes funcionalidades del sistema.
2. **Encriptación de datos:** Se utilizan protocolos de encriptación para proteger la confidencialidad de la información transmitida entre el sistema y los usuarios, evitando posibles ataques de interceptación.
3. **Validación de entradas:** Se implementan controles de validación en todos los formularios y entradas de datos para prevenir ataques de inyección de código (SQL injection, XSS), asegurando que los datos introducidos por los usuarios sean seguros y válidos.

**Seguridad hacia lo interno**:

1. **Monitoreo y detección de anomalías:** Se implementan herramientas de monitoreo en tiempo real para detectar posibles anomalías en el funcionamiento del sistema, como picos inusuales de tráfico, comportamientos sospechosos, etc.
2. **Copias de seguridad y recuperación ante fallos:** Se establecen políticas de copias de seguridad periódicas para garantizar la disponibilidad y la integridad de los datos en caso de fallos o incidentes. Además, se implementan mecanismos de recuperación ante desastres para minimizar el tiempo de inactividad del sistema.
3. **Pruebas de penetración y auditorías de seguridad:** Se realizan pruebas de penetración periódicas para identificar posibles vulnerabilidades en el sistema y corregirlas antes de que puedan ser explotadas por atacantes. Se llevan a cabo auditorías de seguridad internas y externas para evaluar el cumplimiento de las políticas de seguridad y mejorar continuamente la protección del sistema.

### Forma general del tratamiento de errores.

**Principios:**

* **Claridad y precisión:** Los mensajes de error y validación deben ser claros y precisos, de modo que el usuario pueda comprender fácilmente qué salió mal y cómo corregirlo.
* **Consistencia:** Es importante mantener la consistencia en la presentación de mensajes de error y validación en todo el sistema para que los usuarios puedan anticipar cómo se comunicarán los problemas.
* **Feedback inmediato:** Los mensajes de error deben aparecer de manera inmediata después de que se haya producido un error, para que el usuario pueda corregirlo rápidamente.
* **Jerarquía de mensajes:** Es útil establecer una jerarquía de mensajes, donde los errores más críticos se muestren de manera más prominente que los menos críticos.
* **Evitar la jerga técnica:** Los mensajes de error y validación deben estar redactados en un lenguaje claro y sencillo, evitando la jerga técnica que pueda resultar confusa para el usuario.

En cuanto a los tipos de mensajes utilizados en el sistema, se pueden distinguir tres tipos principales:

1. **Mensajes de error:**

Indican que se ha producido un problema que impide que se complete una acción. Estos mensajes suelen ser de color rojo y deben explicar claramente cuál es el error y cómo corregirlo.

**Ejemplo:** "La contraseña ingresada es incorrecta. Por favor, inténtalo de nuevo."

**Color:** Rojo

**Descripción:** Este mensaje indica que se ha producido un error al ingresar la contraseña y se le pide al usuario que la ingrese nuevamente.

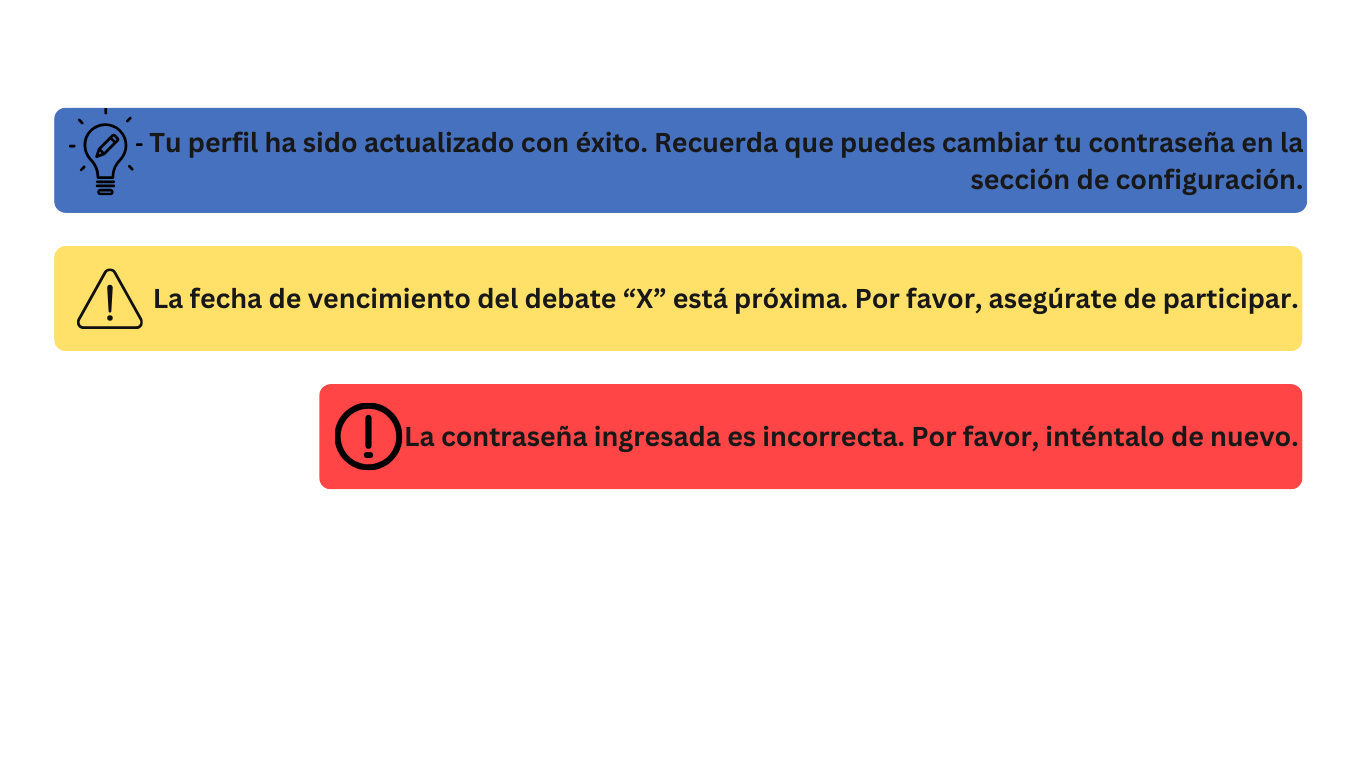


Ilustración 32: Mensaje de error

1. **Mensajes de advertencia:**

Se utilizan para alertar al usuario sobre posibles problemas o situaciones que requieren atención, pero que no impiden completar una acción. Suelen ser de color amarillo y deben ofrecer información sobre las posibles consecuencias.

**Ejemplo:** "La fecha de vencimiento del debate “X” está próxima. Por favor, asegúrate de participar."

**Color:** Amarillo

**Descripción:** Este mensaje advierte al usuario sobre una situación que requiere atención, en este caso, la proximidad de la fecha de vencimiento de la tarjeta de crédito.

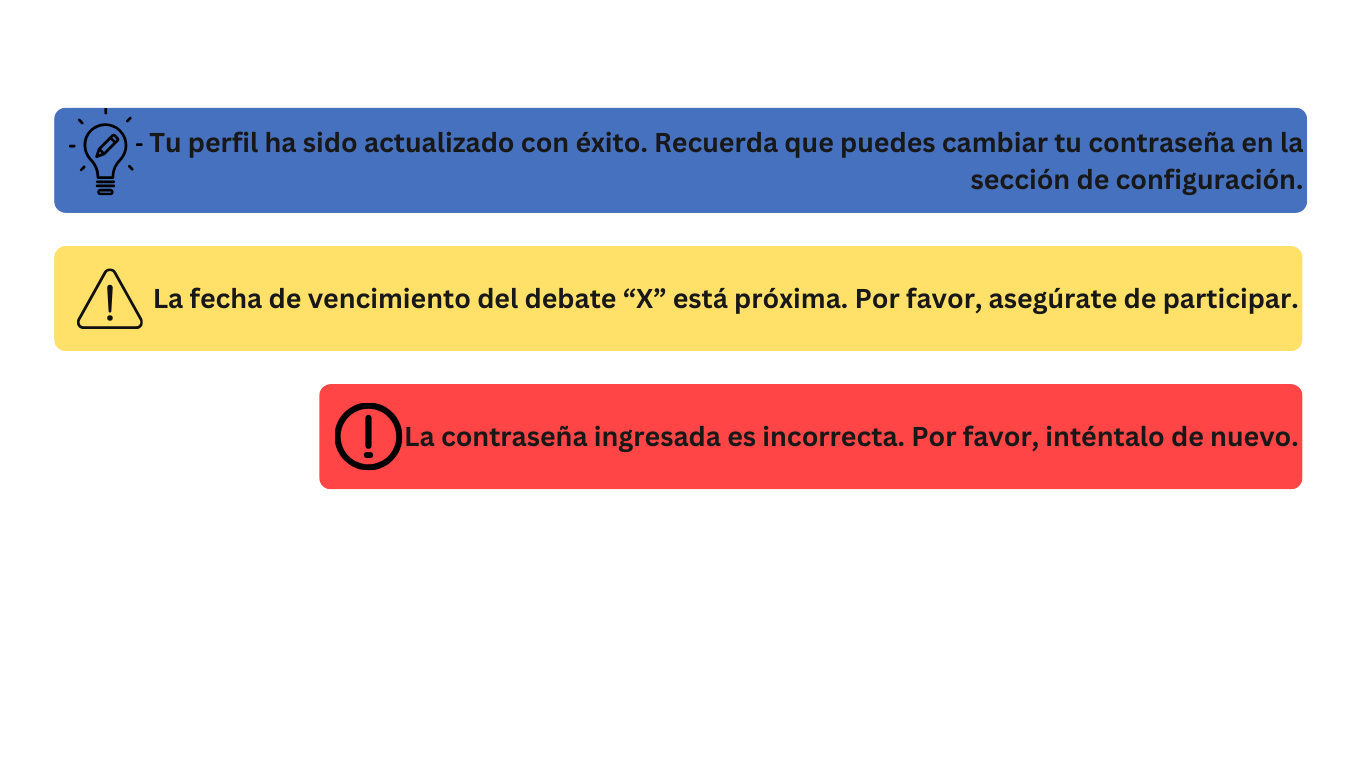


Ilustración 33: Mensaje de advertencia

1. **Mensajes de información:**

Proporcionan información adicional al usuario sobre ciertos aspectos del sistema o del proceso en curso. Suelen ser de color azul y su objetivo es educar al usuario o proporcionarle información relevante.

**Ejemplo:** "Tu perfil ha sido actualizado con éxito. Recuerda que puedes cambiar tu contraseña en la sección de configuración."

**Color:** Azul

**Descripción:** Este mensaje proporciona información relevante al usuario sobre una acción completada con éxito, y además le ofrece una sugerencia adicional sobre cómo realizar otra tarea relacionada.

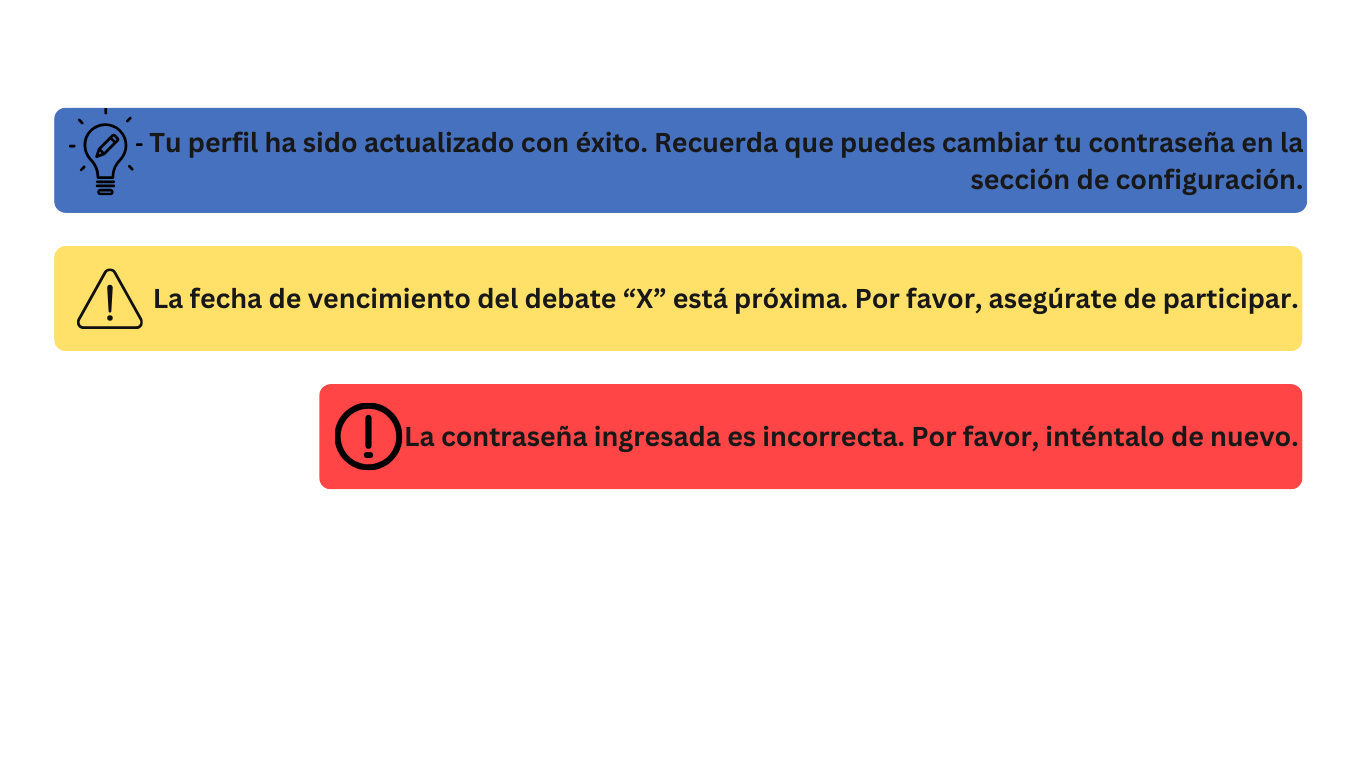


Ilustración 34: Mensaje de información

## Diseño y resultado de la Estrategia de Pruebas

La estrategia fundamentalmente trabajará sobre los principios de las pruebas tempranas y la dependencia de las del contexto asociado (Diaz, Casañola y Hidalgo, 2020).

El plan de pruebas trazado para este sistema se basa en la realización de pruebas unitarias, de integración, sistema y rendimiento, con el objetivo de verificar el correcto funcionamiento de los requisitos funcionales y no funcionales del sistema. Se realizarán un total de 58 pruebas, distribuidas de la siguiente manera:

**Pruebas unitarias:** se realizarán 20 pruebas para verificar el cumplimiento de los requisitos funcionales del sistema. Cada prueba estará enfocada en un requisito específico y se utilizará la técnica de caja blanca para su ejecución.

**Pruebas de integración:** se realizarán 15 pruebas para verificar la correcta interacción entre los diferentes módulos del sistema. Se utilizará la técnica de caja negra para identificar posibles fallos en la integración de los módulos.

**Pruebas de sistema:** se realizarán 15 pruebas para verificar el cumplimiento de los requisitos no funcionales del sistema, como la usabilidad, rendimiento y seguridad. Se utilizará la técnica de aceptación del usuario final para validar que el sistema cumple con las expectativas del cliente.

**Prueba de Rendimiento:** se realizarán 8 pruebas para verificar el rendimiento del sistema bajo diferentes cargas de trabajo, se agruparán de la siguiente manera:

* **Prueba de carga:** se simulará una carga de trabajo normal para evaluar la capacidad del sistema para manejar un número esperado de usuarios concurrentes. Se medirá el tiempo de respuesta y el rendimiento del sistema bajo esta carga.
* **Prueba de estrés:** se aumentará gradualmente la carga de trabajo hasta alcanzar el límite de capacidad del sistema. Se evaluará cómo responde el sistema en situaciones de sobrecarga y cuál es su comportamiento ante picos de demanda.

Las pruebas se realizarán sobre los 9 requisitos funcionales y los 13 requisitos no funcionales del sistema. En cada iteración del software se probarán diferentes módulos, siguiendo la estrategia de prueba establecida en el plan. Por ejemplo, en la primera iteración se probarán los módulos de autenticación y registro de usuarios, utilizando pruebas unitarias y de integración. En la segunda iteración se probará el módulo de gestión de debate, utilizando pruebas de sistema y de rendimiento para verificar su funcionamiento correcto.

### Estrategia de prueba

Una estrategia de pruebas es un conjunto de definiciones de alto nivel que determinan cómo se van a llevar a cabo los procesos de pruebas de software, impulsando nuestras decisiones sobre cómo invertir nuestros esfuerzos de pruebas (Abstracta Team, 2022).

A continuación, se muestra una tabla resumen de la estrategia de pruebas llevada a cabo según su tipo y niveles:

Tabla 4: Estrategia de Pruebas

|  |  |
| --- | --- |
| Tipos | Niveles |
| Funcionalidad (Pruebas de casos de uso) | **Prueba de Desarrollador:**  Pruebas unitarias  **Prueba de Unidad:**  Verificación de módulos individuales. |
| Usabilidad (Pruebas de interfaz de usuario.) | **Prueba de Aceptación:**  Verificación con el cliente para aceptación final.  **Prueba de Sistema:**  Verificación del sistema completo. |
| Rendimiento (Pruebas de carga y estrés) | **Pruebas de integración:**  Evaluar tiempo de respuesta del sistema  Medir capacidad de carga de trabajo del sistema |

**Verificación de Seguridad:**

Se realizarán pruebas específicas para verificar el cumplimiento de los requisitos de seguridad, incluyendo pruebas de penetración, pruebas de autenticación y autorización, y pruebas de encriptación.

Tabla 5: Caso de Prueba Crear Debate

|  |  |
| --- | --- |
| **Diseño de caso de prueba para escenario Crear Debate** | |
| Descripción | Verificar que el sistema permita a los usuarios crear un nuevo debate en la plataforma. |
| Condiciones | El usuario debe estar autenticado en la plataforma y tener los permisos necesarios para crear debates. |
| Entradas | Tema, Fecha de Inicio, Tipo, Fecha Fin, Participantes, Descripción |
| Resultados | El debate se crea exitosamente y se muestra en la lista de debates activos. |

Tabla 6: Caso de Prueba Recuperar Acuerdo

|  |  |
| --- | --- |
| **Diseño de caso de prueba para escenario Recuperar Debate** | |
| Descripción | Validar que los usuarios puedan recuperar acuerdos previamente establecidos en la plataforma. |
| Condiciones | El usuario debe tener acceso a los acuerdos previos y estar autorizado para visualizarlos. |
| Entradas | Identificador del acuerdo o término de búsqueda. |
| Resultados | El acuerdo solicitado se muestra correctamente con todos los detalles asociados. |

Tabla 7: Caso de Prueba Generar Acta de Acuerdo

|  |  |
| --- | --- |
| **Diseño de caso de prueba para escenario Generar Acta de Acuerdo** | |
| Descripción | Comprobar que el sistema permita generar un acta detallada de un acuerdo específico. |
| Condiciones | El usuario debe ser parte del acuerdo y tener los permisos necesarios para generar el acta. |
| Entradas | Identificador del acuerdo, contenido adicional para el acta (si es necesario). |
| Resultados | Se genera un documento con el resumen del acuerdo, participantes, fecha y otros detalles relevantes. |

Tabla 8: Caso de prueba Opinar

|  |  |
| --- | --- |
| **Diseño de caso de prueba para escenario Opinar** | |
| Descripción | Verificar que los usuarios puedan expresar su opinión sobre un debate en la plataforma. |
| Condiciones | El usuario debe estar autenticado y tener acceso al debate sobre el cual desea opinar. |
| Entradas | Comentario u opinión sobre el tema en cuestión. |
| Resultados | El comentario se publica correctamente y se muestra en la sección de opiniones relacionadas. |

### Resultados de las pruebas

En las pruebas funcionales se tuvieron cuenta el método de caja negra y la técnica de partición equivalente. Para la aplicación de la técnica de partición equivalente se retomaron los casos de prueba diseñados en el epígrafe 3.4.1 “Estrategia de Pruebas”. En la figura 12 se presenta un gráfico con los resultados de las pruebas funcionales realizadas.

En total, se planificaron tres iteraciones de prueba. Como resultado final de las pruebas funcionales se obtuvo en una primera iteración cuatro no conformidades de funcionalidad y dos de validación. En la segunda iteración se resolvieron dos de las no conformidades de funcionalidad y en la tercera iteración se resolvieron todas y no se detectaron nuevas no conformidades.

Ilustración 35: Resultados de las pruebas funcionales

Ejemplos de las pruebas funcionales realizados:

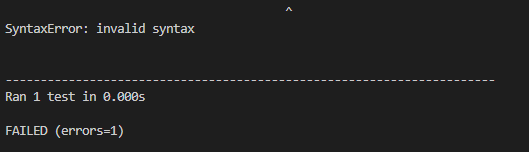


Ilustración : Prueba 1

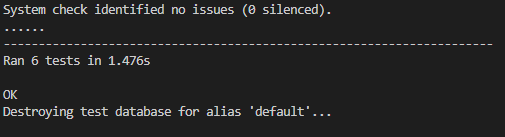


Ilustración : Prueba 2

En las figuras anteriores (Figura 36 y 37) se demuestran dos de las tres iteraciones realizados para probar si las pruebas funcionales fueron correctas.

# Conclusiones

Como conclusiones podemos decir que los tres objetivos fundamentales los cuales fueron:

1. Desarrollar una plataforma para el debate universitario
2. Implementar una funcionalidad para el registro y seguimiento de opiniones.
3. Implementar una funcionalidad para el mecanismo de toma de Acuerdos.

Fueron cumplidos con satisfactoriamente teniendo en cuenta los siguientes parámetros:

1. **Cumplimiento de Objetivos**:

* Se logró un modelado detallado del negocio, identificando reglas y actores clave.
* La propuesta de solución abarcó la visión del producto y definición, evidenciando requisitos funcionales y no funcionales.
* El análisis y diseño del sistema incluyó casos de uso, arquitectura, modelo de datos y despliegue efectivo.

2. **Beneficios del Producto**:

* Beneficios Técnicos: Mejora en la gestión de debates y acuerdos, facilitando la documentación.
* Beneficios Intangibles: Impulso a la participación estudiantil y la generación de conocimiento colectivo.

3. **Validación de la Respuesta**:

* El análisis de factibilidad y planificación temporal garantizaron el éxito del proyecto.
* El diseño del prototipo del sistema permitió evaluar y ajustar funcionalidades de manera efectiva.
* Las estrategias de pruebas aplicadas demostraron la robustez y efectividad de la plataforma.

En resumen, la elaboración de la aplicación web de debate universitario ha demostrado su viabilidad y beneficios. La integración de funcionalidades clave y la planificación meticulosa han resultado en una solución efectiva que promueve la participación y eficiencia en los debates académicos, generando un impacto positivo en el ambiente universitario y potenciando el intercambio de ideas de forma innovadora.

# Recomendaciones

Se recomienda que:

1. Este trabajo sea guardado para futuras referencias de desarrollos de software parecidos, donde nuevos desarrolladores puedan apoyarse de los resultados del proyecto.
2. Desarrollar en el futuro mas módulos o funcionalidades para así enriquecer el producto ya creado llevándolo a versiones superiores de su diseño original.

# Bibliografía

ABSTRACTA TEAM, 2022. ¿Cómo crear la estrategia de pruebas de software adecuada para tu proyecto? - Abstracta. *Blog de Testing, QA y Calidad de Software | Abstracta* [en línea]. [consulta: 3 abril 2024]. Disponible en: https://es.abstracta.us/blog/guia-crear-estrategia-pruebas-software-adecuada/.

ACEVEDO, C.A.G., 2015. Guía pedagógica Arquitectura de software. En: Accepted: 2016-03-15T16:36:28Z [en línea], [consulta: 22 marzo 2024]. Disponible en: http://ri.uaemex.mx/handle/20.500.11799/33827.

DIAZ, A.M., CASAÑOLA, Y.T. y HIDALGO, D.B., 2020. Estrategia de pruebas para organizaciones desarrolladoras de software. [en línea], vol. 14, no. no.3, [consulta: 3 abril 2024]. Disponible en: http://scielo.sld.cu/scielo.php?script=sci\_arttext&pid=S2227-18992020000300083.

GONZÁLES, A.H., 2004. *IDENTIFICACIÓN DE PROCESOS DE NEGOCIO* [en línea]. 2004. S.l.: s.n. [consulta: 15 marzo 2024]. Disponible en: https://rii.cujae.edu.cu/index.php/revistaind/article/view/168/152.

HERNANDEZ, R.D., 2021. El patrón modelo-vista-controlador: Arquitectura y frameworks explicados. *freeCodeCamp.org* [en línea]. [consulta: 22 marzo 2024]. Disponible en: https://www.freecodecamp.org/espanol/news/el-modelo-de-arquitectura-view-controller-pattern/.

IBM, 2024. ¿Qué son las reglas de negocio? | IBM. *IBM* [en línea]. [consulta: 13 marzo 2024]. Disponible en: https://www.ibm.com/es-es/topics/business-rules.

LUCIDCHART, 2024. Tutorial sobre diagramas de paquetes. *Lucidchart* [en línea]. [consulta: 21 marzo 2024]. Disponible en: https://www.lucidchart.com/pages/es/que-es-un-diagrama-de-paquetes-uml.

MUERAS, R.M., 2010. Modelo de Negocio y Análisis de Requerimientos Basado en el Proceso Unificado. *Ingenium*, vol. No 03, no. No 01,

PRESSMAN, R.S., 2013. *Ingeniería del software: un enfoque práctico* [en línea]. S.l.: s.n. [consulta: 2 julio 2020]. ISBN 978-1-4562-1836-2. Disponible en: http://www.ingebook.com/ib/NPcd/IB\_BooksVis?cod\_primaria=1000187&codigo\_libro=4272.

# Anexos

**Anexo 1: Entrevista realizada a estudiantes de la Universidad de Camagüey para saber los puntos mas importantes durante la realización de un acuerdo y/o debate.**

**Objetivo:** Conocer como es el proceso de un debate y una toma de acuerdo en la realidad.

**Preguntas:**

¿Se escoge un tema a tratar durante un debate o el tema puede ser libre?

¿Los acuerdos son llevados a cabo mediante qué manera?

Existe un encargado de guiar estos debates y/o acuerdos?

Se guarda una documentación física de lo hablado y acordado en estos debates

¿Se puede llegar a un acuerdo sin necesidad de un debate previo?

¿Se toma mucho tiempo llegar a un acuerdo cuando hay mucha masividad de participantes? ¿El proceso resulta innecesariamente engorroso?

Sugerencia de que le gustaría que tuviera una plataforma enfocada en el debate y la toma de acuerdo en una comunidad universitaria.

**Anexo 2:** **Guía de observación para verificar el proceso de acuerdo y debate en la Universidad de Camagüey**

**Observador:** Juan Ariel

**Objetivo:** Identificar los elementos fundamentales para la realización de un debate y acuerdo en la comunidad universitaria.

1. **Datos sobre debates y acuerdos**

* Tema
* Fecha de realizado
* Asistencia de participantes
* Votos a favor y en contra
* Descripción del tema a tratar
* Documentación manual del proceso

¿Como se realizan en la actualidad los debates?

¿De qué manera se llega a un acuerdo?

¿Como son recogidas las opiniones de los asistentes?