



UNIVERSIDAD CENTRAL DE VENEZUELA
FACULTAD DE CIENCIAS
ESCUELA DE COMPUTACIÓN
CENTRO DE ENSEÑANZA ASISTIDA POR COMPUTADOR - CENEAC

Desarrollo de un Sistema de Gestión de Arbitraje de Artículos de Investigación

**Trabajo Especial de Grado presentado ante la
Ilustre Universidad Central de Venezuela por
el Br. Albert J. Urbina U. (C.I. 20.616.775)
y el Br. Rabindranath Ferreira (C.I. 19.820.078)**

Tutora: Profa. Yusneyi Y. Carballo Barrera

Ciudad Universitaria de Caracas, Octubre de 2019

UNIVERSIDAD CENTRAL DE VENEZUELA
FACULTAD DE CIENCIAS
ESCUELA DE COMPUTACIÓN
CENTRO DE ENSEÑANZA ASISTIDA POR COMPUTADOR - CENEAC

ACTA

Quienes suscriben, miembros del jurado designado por el Consejo de la Escuela de Computación, para examinar el Trabajo Especial de Grado titulado “**Desarrollo del Sistema de Gestión de Arbitraje de Artículos de Investigación**” y presentado por el Br. Albert J. Urbina U. (C.I. V-20.616.775) y el Br. Rabindranath Ferreira (C.I. V-19.820.078), a los fines de optar al título de **Licenciado en Computación**, dejamos constancia de lo siguiente:

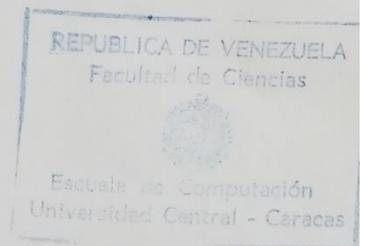
Leído como fue dicho trabajo, por cada uno de los miembros del jurado, se fijó el día 24 de octubre de 2019, a las 02:30 pm horas, para que los autores lo defendieran en forma pública, lo que estos hicieron en la Sala de Exposiciones del Centro de Computación, mediante una presentación oral de su contenido, luego de lo cual respondieron a las preguntas formuladas. El bachiller Rabindranath Ferreira realizó su exposición y defensa del TEG desde la ciudad de Santiago de Chile, Chile, respondiendo a las preguntas formuladas mediante video llamada. El jurado evaluador no manifestó ninguna objeción al respecto.

Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió aprobar con la nota de 19 puntos.

En fe de lo cual se levanta la presente Acta, en Caracas el día 24 de octubre de 2019.

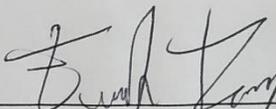


Profa. Yusneyi Carballo Barrera
Tutora





Prof. Robinson Rivas
Jurado



Prof. Ernesto Fuenmayor
Jurado

Agradecimientos y Dedicatorias

En primer lugar, le doy gracias a Dios por darme salud, sabiduría y la madurez necesaria para afrontar los retos que se me presentan.

A nuestras madres y abuelas por estar siempre ahí guiándonos y apoyándonos en las decisiones que he tomado.

A mi madrina Emilia por siempre darme buenos consejos y motivarme a seguir siempre adelante sin importar lo difícil que parezca el camino.

A mi primo Richard que, aunque ya no se encuentre entre nosotros, siempre estuvo pendiente de mis logros, me motivó a ser mejor en todo lo que hacía y siempre estuvo ahí para darme buenos consejos.

A mi prima Iraima, por siempre apoyarme, darme buenos consejos y creer en mí.

A mi tía-madrina Yngrid por estar allí y ayudarme a sobre llevar las dificultades y darme ánimos para continuar.

A mis grandes amigas Iranid del Carmen, Miriali Lopez por apoyarme y animarme para continuar sin importar el tamaño del avance.

Así como también a nuestros demás familiares y amigos por acompañarnos a lo largo de nuestra carrera y por siempre estar pendiente de nuestros logros. En especial a esos compañeros que nos apoyaron dándonos consejos, y asesorándonos para poder llegar a una mejor solución.

De igual manera estamos agradecidos con todos los profesores que intervinieron en nuestra formación ya que muchas de esas enseñanzas nos han permitido llegar aquí. Su consultoría, enseñanzas y conocimientos nos formaron como buenos profesionales.

A nuestra tutora, Yusneyi Carballo Barrera, por darnos la oportunidad de realizar este proyecto y guiarnos a lo largo del desarrollo del mismo.



Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación
Centro de Enseñanza Asistida por Computador - CENEAC

Desarrollo de un Sistema de Gestión de Arbitraje de Artículos de Investigación

Autores:

Albert Johan Urbina Urbina, C.I.: 20.616.775, johan50693@gmail.com
Rabindranath Ferreira, C.I.: 19.820.078, rabindranathucv@gmail.com

Tutora: Profa. Yusneyi Carballo Barrera
yusneyi.carballo@ciens.ucv.ve

Fecha: 24 de octubre de 2019

RESUMEN

El siguiente Trabajo Especial de Grado tiene como objetivo el desarrollo de un sistema de gestión de arbitraje para artículos de investigación, tomando como caso de estudio la postulación y evaluación de los resúmenes de trabajos libres de la Convención Anual de la Asociación Venezolana para el Avance de la Ciencia (AsoVAC). En la actualidad, la AsoVAC no cuenta con una aplicación informática que permita la gestión integrada de todo el proceso de arbitraje. Hasta la edición del año 2018, se han utilizado diferentes herramientas, lo cual implica invertir tiempo en la revisión y actualización de múltiples archivos en distintos formatos con datos dispersos en múltiples tablas, hojas de cálculo, documentos y carpetas. La propuesta planteada en esta investigación se orienta a desarrollar un sistema que contemple cinco (5) procesos principales: registro de trabajos, asignación de áreas de arbitraje, arbitraje, registro de eventos, e información de los resultados de la evaluación. Se han considerados los procesos, metadatos y especificaciones de varias aplicaciones con un fin similares, en especial, el *Open Journal Systems* y el *Open Conf*. Ambas son útiles para realizar la revisión por pares (*peer-review*), permitiendo la evaluación de un contenido académico por más de un experto en un área. Otra ventaja del proceso de revisión por pares es ayudar a mantener estándares de calidad y de credibilidad en cuanto al rigor científico de los trabajos postulados y de su revisión. En el **Sistema de Gestión de Arbitraje de Artículos de Investigación para AsoVAC** se contempla el modelo de revisión por pares y la participación de varios actores: administrador del sistema, autores (de resúmenes o de eventos), coordinador general, coordinadores de área y árbitros. Estos actores intervienen en varias actividades durante el proceso del arbitraje con funciones y estados bien definidos. Adicionalmente, el sistema desarrollado también apoya el **proceso de postulación y registro de eventos**, la planificación de las sesiones de presentación de trabajos y de eventos, la elaboración del libro de resúmenes de la Convención Anual, así como la elaboración de los certificados de participación. Como tecnologías y lenguajes de desarrollo se combinó Python, JavaScript, PostgreSQL y los *frameworks* Django y Bootstrap, aplicando el método de desarrollo Scrumban.

Palabras clave: Convención Anual de AsoVAC, arbitraje de artículos de investigación, gestión de publicaciones, revisión por pares, *Peer review*, *Open Journal Systems*, *OpenConf*.

Índice de Contenido

INTRODUCCIÓN	i
CAPÍTULO I. Planteamiento del Problema.....	1
1.1 Situación Actual y Necesidad.....	1
1.2 Propuesta de Solución y Justificación	2
1.3 Objetivos de la Investigación del Trabajo Especial de Grado.....	2
1.3.1 Objetivo General	2
1.3.2 Objetivos Específicos.....	2
CAPÍTULO II. Marco Teórico y Metodológico	3
2.1 Antecedentes	3
2.1.1 La Asociación Venezolana para el Avance de la Ciencia, AsoVAC	3
2.1.2 <i>Sociedade Brasileira para o Progresso da Ciência</i>	6
2.1.3 Sistemas para la Revisión por Pares	7
2.1.4 Métodos de Evaluación de Artículos	11
2.2 Fundamentos del Desarrollo Web y Herramientas Tecnológicas	12
2.2.1 Definición de Aplicación Web.....	12
2.2.2 Ventajas de las Aplicaciones Web	12
2.2.3 Desventajas de las Aplicaciones Web.....	13
2.2.4 Arquitectura Cliente-Servidor.....	13
2.2.5 Características de la Arquitectura Cliente-Servidor.....	13
2.2.6 Ventajas de la Arquitectura Cliente-Servidor	14
2.2.7 Limitaciones de la Arquitectura Cliente-Servidor	15
2.2.8 Patrón de Diseño MVC (Modelo-Vista-Controlador)	15
2.2.9 Herramientas Tecnológicas para el Desarrollo de la Aplicación Web	16
2.2.10 HTML (Lenguaje de Marcado de Hipertexto).....	17
2.2.11 Hojas de Estilos en Cascada (CSS).....	17

2.2.12 Bootstrap.....	19
2.2.13 JavaScript.....	19
2.2.14 jQuery.....	19
2.2.15 Django.....	20
2.2.16 Tecnologías del lado del Servidor de Bases de Datos	21
2.2.17 PostgreSQL.....	21
2.2.18 Ventajas de PostgreSQL.....	21
2.2.19 Desventajas de PostgreSQL.....	22
2.3 Resumen del Capítulo	22
CAPÍTULO III. Marco Metodológico	24
3.1 Metodología Scrum	24
3.1.1 Teoría de Scrum	24
3.1.2 El Equipo Scrum	25
3.1.3 Artefactos	26
3.1.4 Eventos	26
3.2 Metodología Kanban	27
3.2.1 Ciclo de Vida.....	28
3.2.2 Tableros Kanban.....	29
3.2.3 Revisión de Tableros Kanban.....	30
3.2.4 Tableros Virtuales	30
3.2.5 Ventajas y Desventajas de los Tableros Kanban.....	30
3.3 Razones que justifican la selección de la metodología	31
3.4 Resumen del Capítulo	32
CAPÍTULO IV. Marco Aplicativo.....	33
4.1 Requerimientos del Sistema de Gestión de Arbitraje de Artículos Académicos	33
4.1.1 Requerimientos Funcionales	33

4.1.2 Requerimientos no Funcionales	33
4.2 Arquitectura del Sistema	33
4.3 Roles de los actores	34
4.4 Aplicación de la Metodología combinada Scrumban.....	35
4.4.1 Cronograma tentativo de actividades (Tablero Kanban)	35
4.4.2 Descripción de las iteraciones y listas de tareas en cada iteración	38
4.5 Descripción de Flujos Principales	41
4.5.1 Flujo coordinador de general	41
4.5.2 Flujo de coordinador de área.....	45
4.5.3 Flujo de árbitro de subárea.....	50
4.5.4 Flujo de autor	52
4.6 Análisis del modelado de datos y definición.....	56
4.6.1 Listado de Tablas	56
4.6.2 Modelado de datos	58
4.7 Descripción de las principales interfaces de la aplicación	59
4.7.1 Interfaces de la aplicación.....	59
4.8 Fase de Pruebas	81
4.8.1 Pruebas funcionales.....	81
4.8.2 Pruebas de integración entre módulos	84
4.9 Resultados	86
CONCLUSIONES.....	92
REFERENCIAS BIBLIOGRÁFICAS	93

Índice de Figuras

Figura N° 1. Reunión anual de la Sociedad Brasileña para el Progreso de la Ciencia.	6
Figura N° 2. Logo de Open Journal System.	7
Figura N° 3. Número de revistas que usan Open Journal Systems.....	10
Figura N° 4. Arquitectura Cliente-Servidor.	14
Figura N° 5. Patrón de Diseño MVC.	16
Figura N° 6. Arquitectura Aplicaciones Web.	17
Figura N° 7. Tablero de Kanban Básico.	28
Figura N° 8. Tablero de Kanban para desarrollo de software.....	29
Figura N° 9. Arquitectura General.	34
Figura N° 10. Flujo del coordinador general - estado iniciado.....	42
Figura N° 11. Flujo del coordinador General - estado selección y asignación de coordinadores de área – Editar datos básicos del arbitraje.....	42
Figura N° 12. Flujo del coordinador general - estado en selección y asignación de coordinadores de área – asignar árbitros de subárea	43
Figura N° 13. Flujo del coordinador general - estado en arbitraje – Arbitrar trabajo.....	44
Figura N° 14. Flujo del coordinador general - estado en arbitraje – Mostrar arbitrajes	44
Figura N° 15. Flujo del coordinador de área - estado iniciado	45
Figura N° 16. Flujo del coordinador de área - estado iniciado	46
Figura N° 17. Flujo del coordinador de área - estado en carga de trabajos – verificar pago.....	46
Figura N° 18. Flujo del coordinador de área - estado en carga de trabajos – status árbitro	47
Figura N° 19. Flujo del coordinador de área - estado en carga de trabajos – Asignar árbitros	47
Figura N° 20. Flujo del coordinador de área - estado en cierre de arbitraje – mostrar resultados	48
Figura N° 21. Flujo del coordinador de área - estado en asignación de sesión – cambiar modalidad de sesión.....	48
Figura N° 22. Flujo del coordinador de área - estado en asignación de sesión – crear sesión	49
Figura N° 23. Flujo del coordinador de área - estado en resumen – mostrar resultados	49
Figura N° 24. Flujo del coordinador de área - estado en resumen – exportar resultados	50
Figura N° 25. Flujo del coordinador de área - estado en resumen – generar certificados	50
Figura N° 26. Flujo del árbitro de subárea - estado en arbitraje – ver trabajo por arbitrar.....	51
Figura N° 27. Flujo del árbitro de subárea - estado en arbitraje – arbitrar trabajo	52

Figura N° 28. Flujo del autor - estado en carga de trabajos – crear un trabajo.....	53
Figura N° 29. Flujo del autor - estado en carga de trabajos – ver trabajo.....	53
Figura N° 30. Flujo del autor - estado en carga de trabajos – editar trabajo.....	54
Figura N° 31. Flujo del autor - estado en carga de trabajos – eliminar trabajo	54
Figura N° 32. Flujo del autor - estado en carga de trabajos – Postular trabajo.....	55
Figura N° 33. Flujo del autor - estado en resumen – ver mis resultados	55
Figura N° 34 Modelo de Base de Datos, parte I	56
Figura N° 35. Modelo de Base de Datos parte II.....	57
Figura N° 36. Interfaz para ingresar al sistema.....	59
Figura N° 37. Interfaz de registro	60
Figura N° 38. Interfaz de selección de aplicación	60
Figura N° 39. Interfaz de selección de arbitraje	61
Figura N° 40. Interfaz para crear arbitrajes	62
Figura N° 41. Interfaz para configuración general parte 1	62
Figura N° 42. Interfaz para configuración general parte 2	63
Figura N° 43. Interfaz para cambio de estado del arbitraje	63
Figura N° 44. Interfaz para mostrar lista de usuarios	64
Figura N° 45. Interfaz para editar usuarios	64
Figura N° 46. Interfaz de carga de usuarios.....	65
Figura N° 47. Interfaz para asignar coordinador general.....	65
Figura N° 48. Interfaz de carga de áreas y subáreas parte 1	66
Figura N° 49. Interfaz de carga de áreas y subáreas parte 2	66
Figura N° 50. Interfaz de carga de subáreas	67
Figura N° 51. Interfaz para listar autores.....	67
Figura N° 52. Interfaz de edición de datos del autor parte 1	68
Figura N° 53. Interfaz de edición de datos del autor parte 2	68
Figura N° 54. Interfaz para cargar autores.....	69
Figura N° 55. Interfaz para listar árbitros	69
Figura N° 56. Interfaz de asignación de sesiones	70
Figura N° 57. Interfaz de asignación de sesiones	71
Figura N° 58. Interfaz de asignación de árbitros	71

Figura N° 59. Interfaz de detalles del pago.....	72
Figura N° 60. Interfaz para listar las sesiones.....	72
Figura N° 61. Interfaz de edición de sesiones.....	73
Figura N° 62. Interfaz de asignación de coordinadores de sesión	73
Figura N° 63. Interfaz para mostrar resultados parte 1	74
Figura N° 64. Interfaz para mostrar resultados parte 2	74
Figura N° 65. Interfaz para certificados de autor.....	75
Figura N° 66. Modelo base de Certificado de autores	75
Figura N° 67. Interfaz para certificados relacionados a eventos	76
Figura N° 68. Interfaz para certificados comisión logística	76
Figura N° 69. Interfaz para certificados de organizadores.....	77
Figura N° 70. Interfaz para certificados de sesiones.....	78
Figura N° 71. Interfaz para certificados de autores por trabajo	78
Figura N° 72. Interfaz para certificados digitales de AsoVAC	79
Figura N° 73. Interfaz para porta nombre de comisión organizadora.....	79
Figura N° 74. Modelo de porta nombre de comisión organizador	80
Figura N° 75. Modelo de porta nombre de autor	81

Índice de Tablas

Tabla N° 1. Cronograma tentativo de actividades	35
Tabla N° 2. Cuadro de pruebas funcionales realizadas por <i>sprint</i>	81
Tabla N° 3. Resultados parte I	87
Tabla N° 4. Resultados parte II.....	88
Tabla N° 5. Resultados parte III	89
Tabla N° 6. Resultados parte IV	90

INTRODUCCIÓN

A medida que van pasando los años la tecnología va evolucionando y se adapta cada vez más a las necesidades de las personas, bien sea en tareas cotidianas o en la realización de procesos que permitan la culminación de tareas más complejas en el ámbito personal, laboral o académico. Es por esto que las empresas, comunidades o asociaciones deben mantenerse actualizadas y hacer uso de las nuevas tecnologías que permitan la automatización de los servicios que estas ofrecen.

En este orden de ideas, los avances científicos generan un impacto gigantesco en la vida de muchas personas, pero muchas de esas ideas no llegan a ser conocidas o percibidas por todas las personas, debido a que pasan por un exhaustivo proceso de selección en el que un investigador/científico propone una idea innovadora y la fundamenta a través de un conjunto de procesos como experimentos, planteamiento de ideas concreta, datos estadísticos, estudios anteriores, etc. Todo esto para ser sometido ante un jurado con el conocimiento pertinente para ratificar si la investigación es auténtica y si genera un conocimiento al hombre.

En el caso de las publicaciones científicas que son postuladas para congresos, universidades y asociaciones de investigación son evaluadas por otros investigadores con suficientes conocimientos en el área para determinar si existe o no un aporte real. Es por ello que surge la necesidad de crear un sistema para la gestión del arbitraje de artículos de investigación, tomando como caso de estudio el arbitraje de resúmenes de trabajos libres de la Asociación Venezolana para el Avance de la Ciencia (AsoVAC).

El punto principal de esta propuesta es buscar la manera de lograr que la fase de postulación, evaluación y veredicto para la publicación de los artículos científicos sea fluida, ya que actualmente se lleva de forma casi manual, tediosa y poco automatizada. Se apunta a una implementación a través de una aplicación que permita generar un proceso sencillo, fácil de controlar y computarizando.

A continuación, se describirá como se encuentra estructurado este documento:

Primero, en el Capítulo I se describe situación actual, el planteamiento del problema, la solución al problema, la justificación de la investigación del seminario y se describen los objetivos del seminario.

Segundo, en el Capítulo II se describen los antecedentes, los fundamentos del desarrollo web y herramientas tecnológicas y las metodologías de desarrollo de software.

Tercero, en el Capítulo III se describen las herramientas a usar para la realización del TEG, la metodología a usar y la justificación de su uso.

Cuarto, en el Capítulo IV se describe el proceso de desarrollo del Sistema de Gestión de Arbitraje de Artículos de Investigación, los requerimientos para la realización de este sistema, las diferentes herramientas usadas para aplicar la metodología de desarrollo seleccionada para este desarrollo, el modelo de datos usado, las interfaces principales creadas para el sistema y las pruebas realizadas al sistema para verificar el correcto funcionamiento de cada una de las funcionalidades realizadas.

Para finalizar se tienen las secciones de Conclusiones y las Referencias Bibliográficas.

CAPÍTULO I. Planteamiento del Problema

Un texto o una publicación científica, es uno de los últimos pasos de cualquier investigación a nivel académico, previo al debate externo para saber si es válida, correcta, corroborable y legítima.

Así mismo, todo inició como cartas personales entre los científicos, libros y publicaciones periódicas lo cual fue evolucionando hasta nuestros días. Actualmente, existe una herramienta que nos ha permitido compartir los conocimientos de manera más rápida y universal, esta es el internet (que justamente nació como un mecanismo para comunicar las distintas fases de las investigaciones científicas entre militares y científicos localizados en distintas partes del mundo).

En este orden de ideas, si el hallazgo científico es de gran trascendencia o actualidad, también se utilizan los medios de difusión masiva y las ruedas de prensa, aunque se considera poco respetable hacerlo antes de haberlo comunicado a la comunidad científica.

Por otro lado, suele denominarse específicamente como comunicación a un tipo de texto científico, más o menos breve, originalmente concebido para su transmisión oral; especialmente el remitido a un congreso o simposio para que esté a disposición de los asistentes dé lugar o no a una conferencia leída realmente en esa reunión, dicho documento fue sometido a revisiones antes de ser presentados en dicho evento. Este proceso se realiza de forma muy estructurada en la actualidad.

1.1 Situación Actual y Necesidad

Una realidad es que las universidades venezolanas tienen que hacer un esfuerzo por difundir los resultados de las investigaciones no sólo a través de revistas científicas especializadas. Sino también las publicaciones que tienen acceso solamente un número de personas relativamente minoritario, por lo general investigadores. Más aun, ni siquiera a todos sino a los de un área específica.

Por otro lado, tenemos otro factor que es la asistencia a congresos, simposios u otros eventos de origen académico, donde los receptores son más escasos mientras más grande sea el evento. Necesariamente, hay que difundir y hacer llegar a la sociedad venezolana los resultados de las investigaciones realizadas, y en los casos que sea posible dejar claro su pertinencia social, su aplicabilidad.

Aunado a esto, algunas organizaciones científicas en Venezuela realizan ciertos eventos donde se hacen las evaluaciones de esos avances científicos, publicaciones, “*abstracts*” o “*papers*”, los cuales son evaluadas en un proceso bien definido y conocido en la comunidad de investigadores, el cual puede convertirse en un proceso sumamente largo y trabajoso sino esta automatizado.

Caso particular es el que existe en La Asociación Venezolana para el Avance de la Ciencia (AsoVAC) el cual no posee aplicaciones informáticas que le permiten integrar y gestionar de una manera sencilla y eficiente las actividades que realiza. Si bien cuenta con blog y un perfil en redes sociales estas no dan basto para la cantidad de postulantes. El motivo principal es que una de sus más importantes actividades es la Convención Anual de AsoVAC en donde cada año desde 1951, un gran número de docentes, investigadores y estudiantes pueden hacer postulaciones de resúmenes de trabajos libres para ser evaluados por árbitros, realizar las correcciones correspondientes y luego ser presentados en el marco del evento.

En el año 2005 las inscripciones para este evento fueron realizadas en línea y era usado un sistema de arbitraje de forma irregular cuando las convenciones no eran organizadas por el capítulo de Caracas, este sistema fue actualizado en el año 2012 y todo el proceso era realizado en línea. En las convenciones siguientes se dejó de usar este sistema por lo complicado de su funcionamiento, en el 2015 y 2016 se utilizó un nuevo sistema de inscripción y arbitraje el cual funcionaba, mas no permitía al coordinador del evento el acceso oportuno y directo a las bases de datos.

En las convenciones AsoVAC realizadas en los años 2017 y 2018, para la realización de los diferentes procesos involucrados en el desarrollo de este evento utilizan herramientas como formularios de Google, perfiles en Facebook y Twitter, videos en YouTube, hojas de cálculo en Microsoft Excel además de manejar la información usando Google Drive y el Dropbox. Como podemos ver estas herramientas han sido útiles para la gestión de los contenidos y la divulgación de información sobre el evento, se requiere una manera de gestionar de forma más organizada el proceso de postulación de trabajos, su arbitraje y aceptación, así como la manipulación de la información de participantes y evaluadores.

1.2 Propuesta de Solución y Justificación

Como propuesta de solución a la problemática presentada anteriormente, se plantea el desarrollo del Sistema de Gestión de Arbitraje de Artículos de Investigación, el cual va a permitir la gestión del evento anual realizado por la AsoVAC.

Así mismo, esta solución tendrá como referencia otros sistemas de revisiones para artículos académicos o científicos como el OJS, y el Open Conf. Posteriormente, se presentarán sus características, su forma o manera de evaluación por parte de sus actores y los elementos que lo componen en futuros capítulos.

En este orden de ideas, el propósito de esta investigación es proponer una solución centrada en un sistema que permita gestionar los procesos involucrados en el arbitraje de resúmenes de trabajos libres de la convención anual de AsoVAC y de esta manera ofrecer una forma que permita facilitar la organización necesaria para la realización de este evento.

Este sistema es un aporte que va a permitir agilizar los procesos de organización y revisión de artículos científicos, dando de esta manera un apoyo a la actividad científica y a la divulgación que se realiza en el país.

1.3 Objetivos de la Investigación del Trabajo Especial de Grado

1.3.1 Objetivo General

Desarrollo del Sistema de Gestión de Arbitraje de Artículos de Investigación el cual va a permitir la gestión de la convención anual realizada por la AsoVAC.

1.3.2 Objetivos Específicos

1. Usar una metodología ágil que permita crear el Sistema de Gestión de Arbitraje de Artículos de Investigación de la convención anual realizada por la AsoVAC.
2. Crear los modelos correspondientes a los procesos involucrados en la gestión de la convención anual.
3. Usar los estándares y patrones de usabilidad en el diseño web para de esta manera poder lograr un diseño acorde a las actividades realizadas por la AsoVAC.
4. Realizar las pruebas correspondientes a cada proceso involucrado en la convención anual realizada por la AsoVAC.

CAPÍTULO II. Marco Teórico y Metodológico

El objetivo de este capítulo es dar a conocer el contexto en el cual se desarrolla el problema planteado en el capítulo anterior, la manera en la que son presentados los resúmenes, ejemplos de asociaciones que realizan eventos científicos de este tipo, la manera en la que funciona la revisión por pares, métodos de evaluación de artículos científicos y las diferentes herramientas que van a permitir desarrollar una solución.

2.1 Antecedentes

Las asociaciones para el avance de la ciencia fueron creadas en varios países europeos desde la tercera década del siglo XIX, la creación de la primera de estas asociaciones fue en Alemania en 1822 y con el transcurso de pocos años otros países europeos y Estados Unidos crearon sociedades que seguían el modelo alemán, un siglo después otros países fundaron sociedades de características similares a las europeas. Entre estas sociedades tenemos la de Brasil y Argentina, seguidas en 1950 por la venezolana.

En un principio, el número de miembros era muy reducido, pero a su vez era muy activo y consciente del papel que esperaban desempeñar, la AsoVAC no se definió como una asociación exclusiva para científicos en sus inicios, sino abierta a diversas categorías de personas que estuvieran vinculadas a la ciencia.

Parte del éxito obtenido se debió a ese punto de vista, pragmático, adoptado por la asociación. Sin embargo, esto no significó que perdiera de vista su objetivo el cual era el fomento de la ciencia básica, más en particular la investigación científica de tipo experimental (Juan José Martín Frechilla, 2005).

2.1.1 La Asociación Venezolana para el Avance de la Ciencia, AsoVAC

Esta asociación se originó en 1950. La Profa. Di Prisco destaca que desde su creación la AsoVAC estimuló la formación de una comunidad científica nacional más madura. Hasta entonces la actividad científica del país se había manifestado debido a la existencia de esfuerzos individuales que se agrupaban en algunas instituciones que surgieron en los años 1936 en las cuales se realizaban trabajos de investigación relacionados con las áreas médicas y del agro (Prisco, 1992).

En palabras dirigidas por su presidente, la Dra. Marisol Aguilera Meneses, en la LXVII Convención Anual, destaca que la Convención Anual se ha realizado todos los años desde la creación de la AsoVAC, con la presentación de 130 trabajos el primer año, este número fue creciendo con el pasar de los años hasta llegar a los 2.286 trabajos presentados en el año 1996. A inicios del siglo XXI estos trabajos fueron disminuyendo hasta llegar a 162 resúmenes presentados en la última convención realizada en el año 2017.

Las causas fundamentales de esta disminución es el deterioro del sistema científico tecnológico de Venezuela, causado por las políticas públicas ineficientes en este ámbito. La actividad científica no ha escapado de la destrucción a la cual han sido sometidas las personas, las ideas y las instituciones en nuestro país en los últimos años (AsoVAC, AsoVAC – Caracas, 2017).

Según la normativa general presentada en su LXVII Convención Anual, las áreas y sub-áreas que participan son las siguientes:

- **Área de Biociencias:** Contiene las sub-áreas de acuicultura, agronomía, antropología, arqueología, biofísica, biología celular, biología molecular, bioquímica, biotecnología, botánica, ecología animal, ecología vegetal, espeleología, estudios ambientales, farmacología, fisiología, genética, histología y morfología celular, inmunología, mastozoología, medicina, micología, microbiología, neurociencia, nutrición, odontología, parasitología, salud pública, tecnología de alimentos.

- **Área de Tecnología:** Contiene las sub-áreas de arquitectura y urbanismo, ciencias de los materiales, ciencias de la tierra, computación, ingeniería ambiental, ingeniería biomédica, ingeniería civil, ingeniería de materiales, ingeniería eléctrica, ingeniería electrónica, ingeniería industrial, ingeniería mecánica, ingeniería química, ingeniería metalúrgica, polímeros, sismología, turismo y hotelería, biotecnología, tecnología de alimentos.
- **Ciencias Exactas:** Contiene las sub-áreas de Física, Matemática, Química.
- **Ciencias Sociales y Humanísticas:** Contiene las sub-áreas de administración, administración de ciencias y tecnología, ciencias políticas, sociología, economía, educación, historia, lingüística, literatura, psicología, geografía, arte.

En su última edición se aceptaron 162 trabajos pertenecientes a 151 autores, estos trabajos fueron debidamente examinados por 92 árbitros. Fueron presentados en modalidad presencial (el 67% de los mismos=109) y en forma virtual (el 33% =53) los cuales se encontraron distribuidas dentro de las sesiones establecidas en las áreas de Biociencias (66), Ciencias Exactas (43), Tecnología (27) y Ciencias Sociales y Humanas (26) (AsoVAC, AsoVAC – Caracas, 2017).

En cuanto a las instituciones involucradas en el patrocinio del evento en el año 2017 se tienen las siguientes:

- Universidad Metropolitana
- Facultad de Ciencias de la Universidad Central de Venezuela
- FundaVAC
- Fundación Empresas Polar

Descripción de la Asociación

Los principales programas que canalizan la actuación de la AsoVAC y que han permitido el incremento de la actividad científica en el país son descritos por la Presidencia de AsoVAC de la siguiente manera:

- **La Convención Anual**

La Convención Anual de la AsoVAC constituye el foro principal para el análisis de la actividad científica en el país. En la actualidad la mayoría de los investigadores que trabajan en el país presentan sus trabajos, donde son discutidos con sentido crítico, lo que ha contribuido a elevar el nivel de la actividad investigativa.

En estas convenciones se llevan a cabo foros, mesas redondas, cursos, simposios y otras actividades afines, que versan sobre cuestiones de interés nacional y aspectos científicos básicos. Estas convenciones han dado lugar a la formación de otras sociedades y grupos a medida que los cultivadores de alguna disciplina se han sentido fortalecidos por su creciente número y la importancia de sus presentaciones.

El número de trabajos presentados y científicos participantes ha crecido en forma marcada a lo largo de los años y particularmente en la última década (Presidencia AsoVAC, 2018a).

- **Acta Científica Venezolana**

Es una revista de carácter multidisciplinario que, se publica desde 1950. Cuenta con un Cuerpo Editorial experimentado. Es financiada por varios organismos de promoción científica como los Consejos de Desarrollo Científico y Humanístico (CDCH) de varias universidades y

FUNDAVAC. Funciona con un sistema de arbitraje, lo que asegura la calidad de los trabajos aceptados (Presidencia AsoVAC, 2018b).

- **Festival Juvenil de la Ciencia**

Con el propósito de despertar vocaciones científicas en el seno del estudiantado de las escuelas Básicas, Media y Diversificada, la AsoVAC creó en el año 1968 el festival juvenil de la ciencia, que tiene lugar cada año en Caracas y otras ciudades del país.

En la preparación para el festival se forman equipos de alumnos bajo la orientación de profesores de educación secundaria y profesores universitarios. Estos equipos trabajan durante un tiempo determinado en la elaboración del proyecto de investigación, que es luego expuesto durante el festival. Un jurado se encarga de revisar cuidadosamente los diferentes trabajos y dictaminan acerca del otorgamiento de las distinciones a los trabajos más destacados (Presidencia Asovac, 2018c).

- **AsoVAC Juvenil**

Es un programa destinado a desarrollar actividades educativas científicas, tecnológicas y culturales por estudiantes de educación superior. Asesorados por docentes profesionales e investigadores universitarios pertenecientes a la AsoVAC (Ortega, 2015).

Presentación de Resúmenes de Trabajos Libres

Para realizar la presentación de los trabajos libres en el marco de la Convención Anual de AsoVAC se realiza un proceso en el cual se debe seguir una normativa establecida, con los siguientes pasos:

1. Se debe realizar el pago de la inscripción del trabajo que se desea presentar. Excepcionalmente se aceptan inscripciones mediante la modalidad de convenio, en donde el participante no realiza el pago, sino que es asumido por una institución.
2. Una vez realizado el pago, se procede a llenar la planilla de registro en la cual se indica información de los autores y sus adscripciones (universidades, centros, instituciones a las cuales están relacionados), del resumen de trabajo postulado y del pago.
3. En el transcurso de algunos días hábiles, se realiza la revisión de la información suministrada en el formulario y la verificación del pago. Se envía un comprobante de inscripción al correo electrónico del autor principal del resumen o se le informa de algún detalle en la información del registro.
4. Los resúmenes de trabajos libres participan en un proceso de arbitraje, en donde se verifica que se cumple con el formato y estilo indicado en la normativa, además de tener la pertinencia científico y académica de un resumen de trabajo de investigación a ser presentado en la Convención Anual de AsoVAC.
5. El resultado del arbitraje es enviado por correo electrónico según las fechas indicadas en el calendario. El certificado se entrega a nombre del trabajo y a nombre de los autores que hayan sido registrados en la planilla de inscripción.

Al momento de realizar la inscripción de resúmenes de trabajos libres, se tienen que tener en cuenta las siguientes consideraciones:

1. Para participar es necesario realizar la inscripción de una investigación desarrollada en cualquier área del conocimiento científico, tecnológico y social. La presentación de este trabajo se realizará en sesiones orales presenciales o en sesiones virtuales. En la convocatoria del año 2017 no se solicitó

para las presentaciones el uso de carteles impresos (posters). Las exposiciones se hacen con apoyo de presentaciones virtuales.

2. El resumen debe contener la información básica del trabajo: Título, introducción, objetivos, metodología, resultados y conclusiones. Escrito en un solo párrafo sin subtítulos.
3. La presentación de trabajos en sesiones presenciales o virtuales tendrá una duración máxima de diez minutos de exposición, con cinco minutos para las preguntas e intercambio de ideas.
4. Las presentaciones presenciales estarán dirigidas por un Coordinador, seleccionado por la Comisión Académica de la Conferencia.
5. Las presentaciones virtuales se realizarán con el apoyo de herramientas de video en vivo y/o carga de video, imágenes o fotos, en los sitios web y en perfiles en redes sociales que se destinen al apoyo de las actividades de la Conferencia.

2.1.2 Sociedade Brasileira para o Progresso da Ciência

La Sociedade Brasileira para o Progresso da Ciência, Sociedad Brasileña para el Progreso de la Ciencia (SBPC), es una asociación civil, sin fines de lucro o posición político-partidaria, enfocada en la defensa del avance científico, tecnológico y del desarrollo educacional y cultural de Brasil.

Con sede en la ciudad de São Paulo, la SBPC está presente en los todos estados brasileños a través de Secretarías Regionales. La SBPC representa más de 100 sociedades científicas asociadas y más de seis mil socios activos, entre investigadores, docentes, estudiantes e ciudadanos brasileños interesados en Ciencia y Tecnología (C&T).

En la actualidad la SBPC participa activamente de debates acerca de cuestiones que determinan el rumbo de las políticas de C&T y de la educación en Brasil. Crea sistemáticamente grupos de trabajo, compuestos por científicos reconocidos en sus especialidades, con el objetivo de estudiar y presentar propuestas para cuestiones específicas de interés nacional.

Anualmente, la SBPC realiza diversos eventos de carácter nacional y regional como la reunión anual de la Sociedad Brasileña para el Progreso de la Ciencia que se muestra en la Figura N° 1 la cual fue realizada del 21 al 26 de julio del año 2013.



Figura N° 1. Reunión anual de la Sociedad Brasileña para el Progreso de la Ciencia.

Fuente: <https://www.scidev.net/america-latina/financiamiento/noticias/brazil-s-science-investment-reaches-record-high-1.html>

Estos eventos se realizan con el objetivo de debatir políticas públicas de C&T y difundir los avances de la ciencia. La entidad también contribuye al debate permanente de cuestiones relacionadas al área de C&T por medio de diversas publicaciones como: el *Jornal da Ciência*, la revista *Ciência e Cultura*, su portal de internet y la edición de libros acerca de diversos temas vinculados a la ciencia brasileña (SBPC, 2017).

2.1.3 Sistemas para la Revisión por Pares

Para conocer el funcionamiento de la revisión por pares es necesario entender cómo se realiza dicho proceso en la actualidad, que herramientas pueden ser usados para dicha revisión y de qué manera pueden ser implementadas soluciones una vez entendido su funcionamiento.

A continuación, se presentan una serie de herramientas que van a permitir definir la mejor solución para la problemática que se desea resolver.

OpenConf

Es un software para la revisión por pares, resumen y gestión de conferencias conocido por su facilidad de uso, interfaz limpia y excelente soporte, OpenConf ha impulsado miles de eventos y revistas en más de 100 países. Actualmente se encuentra disponible en tres ediciones:

La edición de la comunidad es usada para funciones básicas como la presentación, tarea de revisión, revisiones en línea, puntuación y notificación, esta edición no requiere de la compra de una licencia para su uso.

La edición plus la cual incluye todos los elementos básicos de un sistema de revisión de resumen y revisión por pares, más soporte técnico, licitación de revisor, discusión de comité, carga de archivo de revisor, refutación de autor, procedimientos en línea, tipos de aceptación personalizados y la capacidad de agregar o eliminar campos pre definidos de formularios de envío y revisión.

Por último, tenemos la edición profesional la cual proporciona más de una docena de módulos para la revisión extendida por pares y funcionalidades de administración de eventos. Además de estas características incluye soporte técnico y la capacidad de personalizar formularios, aceptar pagos para presentarlos, asignar revisiones automáticas, entre otras funciones (Group, 2004-2017). Este software ofrece la posibilidad de crear módulos adaptados a las necesidades de cada problema e insertarlos dentro de las funcionalidades del software para ampliar sus funcionalidades.

Sistemas de Revistas abiertas u *Open Journal System*

Los *Open Journal Systems* (OJS) son aplicaciones para la gestión y publicación de revistas que han sido desarrollados por el *Public Knowledge Project*, a través de esfuerzos financiados por el gobierno federal para expandir y mejorar el acceso a la investigación (Simon Fraser University, 2014).

En la Figura N° 2 se puede observar el logo que identifica los sistemas de gestión de publicaciones de este tipo.



Figura N° 2. Logo de Open Journal System.

Fuente: <http://tecnologiasdocumentales.com/producto-recomendado>

Actualmente, OJS se encuentra en una tercera versión, la cual fue lanzada el 31 de agosto de 2016. Sus principales características son las siguientes:

- Se instala localmente y se gestiona localmente o a través de Internet.
- Los editores configuran requisitos, secciones, proceso de revisión, etc.
- Presentación en línea y administración de todo el contenido.
- Permite el envío y recibo de artículos en línea, incluyendo a los árbitros.
- Indexa los contenidos.
- Integración completa por contenido (revistas) como parte del sistema global. Específicamente la creación de *n* instancias llamadas como *journals* donde cada una es administrada de forma centralizada según el role asignado, permitiendo gestionar gran cantidad de contenido, procesos de evaluación, y segmentación de artículos científicos.
- Herramientas de lectura, revisión para el contenido, según el campo y la elección del editor.
- Notificación de correo electrónico y capacidad de comentarios para los lectores.
- Soporte de ayuda en línea.

Además de las características presentadas en la documentación oficial cabe destacar las siguientes:

- **Las funcionalidades ofrecidas por el sistema son gratis:** Una vez realizada la instalación de la aplicación se tiene acceso a todas las funcionalidades ofrecidas por el sistema sin ningún tipo de restricciones.
- **Las funcionalidades ofrecidas por el sistema son de código abierto:** Una vez son descargados los archivos de instalación de la aplicación se tiene acceso al código fuente de la aplicación lo cual permite realizar modificaciones a las funcionalidades básicas ofrecidas por el sistema.
- **Plugins:** Permite la instalación de *plugins* que permitan ampliar las funcionalidades de la aplicación.

OJS es un software de código abierto, el cual tiene el objetivo de hacer que las publicaciones de acceso libre sean una opción viable para más revistas, ya que el acceso abierto ayuda a aumentar el número de lectores, a divulgar la investigación y dar acceso al conocimiento (Simon Fraser University, 2014).

En este orden de ideas, el OJS es una aplicación creada por el *Public Knowledge Project* (PKP) desarrollado por *University of British Columbia*, el cual es la base para diferentes aplicaciones de revistas científicas existentes en el mundo de carácter privativo.

Este software posee funciones básicas para realizar el proceso de revisión en su instalación inicial, las cuales no incluyen funcionalidades para realizar respaldos a la base de datos o contenidos publicados en la revista, si se desea agregar nuevas funcionalidades al sistema se tiene la opción de buscar un *plugin* que cumpla con esta funcionalidad y realizar la instalación del mismo o se debe desarrollar dichas funcionalidades, tomando en cuenta la estructura base y las reglas aceptadas por el sistema para el desarrollo de nuevas funcionalidades.

Algunos de los ejemplos de revistas donde usan este software son los siguientes:

El Consejo Superior de Investigaciones Científica, en donde se tienen más de 30 revistas españolas usando este software al cual podremos acceder mediante el siguiente link: <http://revistas.csic.es/> y el Congreso de la Universidad de Murcia donde se tienen alrededor de 10 congresos españoles que usan este sistema y se pueden ver en <http://congresos.um.es>.

Roles de los usuarios

Para la realización de las diferentes actividades involucradas en el proceso editorial se crean los roles usados por cada usuario los cuales según la página Aprende en línea (2016), se definen de la siguiente manera:

- **Administrador general:** Es el encargado de realizar la configuración en el OJS y darle el estilo al portal para los usuarios.
- **Gestor de revista:** Es el encargado de iniciar la publicación con los parámetros preestablecidos, adicionalmente maneja a los usuarios y los roles en el proceso editorial.
- **Editor:** Es el encargado de supervisar todo el proceso editorial. Inicia el proceso mediante la asignación de los artículos que van llegando al sistema a los editores de sección para continuar su revisión.
- **Editor de sección:** Supervisa el envío, mediante su revisión y reenvío a pares académicos y correctores de estilo, diagramación y ortografía. Así mismo envía las novedades al autor para que esté enterado en todo momento del estado de su artículo.
- **Par evaluador:** Se encarga de la revisión analítica del artículo, su pertinencia y alcance investigativo; es la persona encargada de dar la aprobación acerca del contenido y emitir su concepto sobre la calidad del artículo que va a publicarse.
- **Corrector de sintaxis:** Se encarga de revisar la gramática y la claridad del autor para expresar las ideas, realiza preguntas al autor para encontrar posibles errores e inconsistencias gramaticales. Se asegura de que el artículo cumpla con los lineamientos bibliográficos y de estilo requeridos por la publicación.
- **Diagramador:** Se encarga de transformar el documento final ya corregido en un artículo gráfico que cumple los lineamientos de imagen institucional predefinidos en forma, fuentes, tamaños y colores.
- **Corrector de estilo:** Se encarga de leer los documentos finales para encontrar errores tipográficos y de formato previos a la publicación.
- **Autor:** Se encarga de realizar el envío de los artículos, este proceso contiene cinco pasos que incluyen la subida del documento donde se encuentra el artículo, la inclusión de los metadatos del mismo y ficheros adicionales. Estos artículos solo serán publicados cuando el editor lo decida.

Formatos para importación y exportación

Para la realización de importación y exportación de la data almacenada en el OJS, éste cuenta con un conjunto de módulos que vienen instalados por defecto, los cuales según la página web poli Papers (2012b), tienen las siguientes características:

- **Módulo XML usuarios/as:** Este módulo permite importar y exportar en formato XML los usuarios y sus roles.
- **Módulo de exportación de artículos de Erudit:** Este módulo permite la exportación individual de artículos para una indexación del texto completo mediante un XML.
- **Módulo XML artículos y números:** Este módulo permite la importación y exportación por lotes. Puede usarse para importar o exportar artículos de forma individual o colectiva en formato XML.

- **Módulo de exportación XML PubMed:** Este módulo permite la creación de información bibliográfica para los artículos del número actual según el estándar PubMed de publicación de datos en formato XML.
- **Módulo de exportación XML Crossref:** Este módulo de exportación permite crear información de metadatos para artículos.

Este software fue desarrollado usando las siguientes tecnologías:

- **Frameworks:** Bootstrap.
- **Lenguaje del lado del servidor:** PHP.
- **Lenguaje del lado del cliente:** JavaScript y JQuery.
- **Manejador de base de datos:** Solo se ha probado con MySQL y PostgreSQL aunque podría funcionar parcial o totalmente con otros gestores de bases de datos (València, 2012).

En la Figura N° 3 se puede observar el crecimiento en el número de revistas que usan OJS con el pasar de los años.

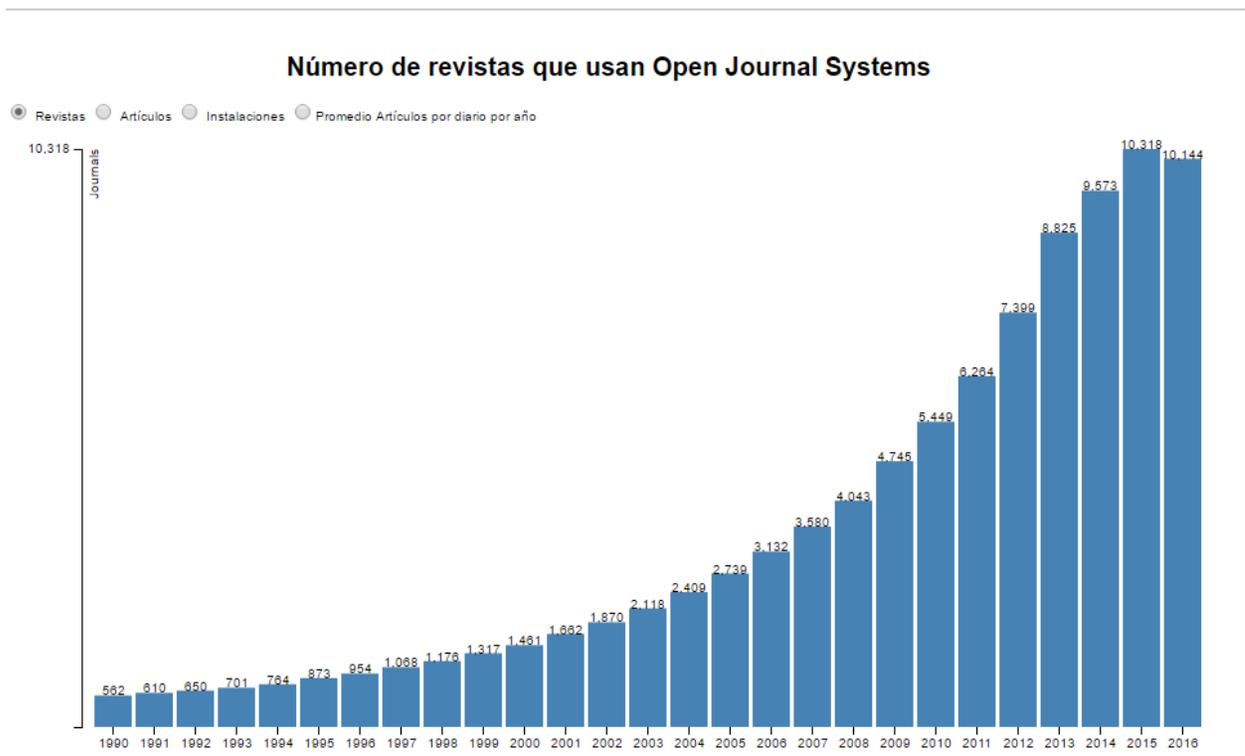


Figura N° 3. Número de revistas que usan Open Journal Systems

Fuente: <https://pkp.sfu.ca/ojs/ojs-usage/ojs-stats/>

A pesar de ser un software *open source* según Pablo (s.f.) , este tipo de sistemas presenta las siguientes desventajas:

1. La documentación es compleja lo cual dificulta la personalización de las plantillas para ofrecer una presentación formal la cual es característica de las revistas científicas.

2. No cuenta con una comunidad activa la cual publique ejemplos de desarrollos de nuevas funcionalidades lo cual significa que los editores y escritores comunes no podrán dominar la herramienta sin tener una capacitación previa.

Una vez realizado un análisis de las diferentes características de este software, se deben tomar en cuenta los pros y contras a la hora de tomar la decisión de usar este software para la publicación de artículos, en base a eso debemos tener en cuenta los siguientes puntos.

Pros

- El proceso de revisión de artículos se realiza de una manera rápida y sencilla.
- Posee módulos y *plugins* que permiten ampliar las funcionalidades del sistema.
- Permite realizar respaldo a la base de datos de forma manual.
- Facilita la comunicación entre los integrantes del proceso de revisión.

Contras

- A pesar de ser código abierto, aumentar las funcionalidades básicas del sistema puede llegar a ser costoso debido a la complejidad del software.
- Utiliza versiones de software que se encuentran sin soporte actualmente.

2.1.4 Métodos de Evaluación de Artículos

Revisión por Pares o *Peer Review*

Según Wikipedia (s.f.), la revisión por pares es un método usado para realizar la validación de trabajos escritos con el fin de evaluar su calidad, originalidad, factibilidad entre otras cosas antes de su publicación. Este método deja abierto el trabajo al escrutinio y frecuentemente a la anotación o modificación por varios evaluadores con igual o mayor rango académico o de formación que el autor. En muchas instituciones, sólo se considera válida una publicación cuando ha pasado por un proceso de revisión por pares, por ejemplo, la admisión para publicación en una revista arbitrada.

La razón principal para el uso de este mecanismo es que en varias ocasiones los autores o los equipos de investigación no pueden encontrar todos los errores o fallas que se pueden encontrar en un trabajo complejo. Al ser revisado por otros grupos igualmente capaces estos pueden encontrar aspectos, características o debilidades no observadas por los autores, lo que puede producir una mejora en el trabajo (Carmen Prado Arreaza).

Este método de evaluación de artículos según (Ladrón de Guevara Cervera, Hincapié, Jackman, & Caballero Uribe) (2008), funciona de la siguiente forma:

Este proceso inicia con la presentación del artículo por parte del autor al grupo editorial los cuales se van a encargar de verificar si el artículo cumple con los requisitos básicos de la revista, luego de esta verificación se define ya sea por 1, 2 o 3 autores o editores de un conjunto de voluntarios si el artículo pasa o no a ser revisado y de esta manera iniciar todo el proceso.

En segundo lugar, se procede a seleccionar los árbitros los cuáles se van a encargar de realizar la revisión y entregarle al editor las sugerencias para mejorar el escrito, este a su vez se encarga de reunir todas las críticas recibidas por parte de los árbitros implicados en la revisión para luego decidir si el artículo es publicable o no.

Cuando un artículo recibe puntos a favor e igualmente puntos en contra se podría considerar un empate y el editor puede elegir entre las siguientes opciones:

1. Los comentarios son comunicados al autor, quien puede explicarle al editor la razón de las críticas hechas por los árbitros.
2. Consultar con los miembros del comité científico y de redacción de la revista.
3. Por último, se le comentan las críticas hechas al autor para que éste opte por realizar las correcciones sugeridas y así poder publicar su artículo.

Este método de evaluación de artículos tiene múltiples beneficios los cuales se presentan a continuación:

- Da transparencia al proceso de publicación, en cuanto el juicio de si un artículo es publicado o no, recae en los científicos con mayor experiencia en un tema. Estos tienen una mejor capacidad crítica y tienen la capacidad de expresar de una manera muy objetiva su opinión.
- Mejora los artículos, pues el análisis hecho por personas sobresalientes en ese tema hace que sus sugerencias beneficien la interpretación que los autores hacen sobre sus hallazgos.
- Ayuda a aumentar la calidad de la revista al elegir solo aquellos artículos que son innovadores o que generan nuevos conocimientos o que presentan conceptos originales para las personas que leen la revista.

2.2 Fundamentos del Desarrollo Web y Herramientas Tecnológicas

Para llevar a cabo cualquier solución basada en tecnología que permita automatizar algún proceso o solucionar un problema mediante el uso de herramientas tecnológicas. Es necesario entrar en contexto sobre los elementos que interactúan para este tipo de soluciones. A continuación, se presentan los siguientes términos:

2.2.1 Definición de Aplicación Web

En la ingeniería de software se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. Al mismo tiempo son reflejados procesos de negocios de organizaciones de toda índole, dichos comportamiento definirán el comportamiento de la herramienta tecnológica.

En este orden de ideas, las aplicaciones web se basan en la arquitectura cliente-servidor, donde el cliente es un navegador web y el servidor es un servidor web que almacena las páginas de la aplicación. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador (Mora, 2002).

2.2.2 Ventajas de las Aplicaciones Web

Las principales ventajas que nos ofrecen las aplicaciones web son:

- Al ejecutarse a través de los navegadores, se puede acceder a ellas a través de cualquier computadora en la que se cuente con Internet.
- No hay incompatibilidades con los sistemas operativos porque todo se maneja en el navegador.
- No ocupan espacio en el disco duro porque se ejecutan a través de la web. aplicaciones web pueden ser utilizadas por múltiples usuarios al mismo tiempo.

- No necesitan ser descargadas, instaladas y configuradas. Se acceden vía en línea y están listas para trabajar sin importar cuál es su configuración o su hardware.
- Todos los cambios de interfaz o de funcionalidad que se deseen hacer en la aplicación web, se realizan en el código que reside en el servidor web y no en cada uno de los clientes.
- Se evitan problemas de inconsistencia de actualizaciones, ya que no existen clientes con distintas versiones de la aplicación.

2.2.3 Desventajas de las Aplicaciones Web

Algunas desventajas de las aplicaciones web son:

- Los múltiples accesos realizados en forma simultánea pueden degradar el rendimiento de las aplicaciones, debido a la sobrecarga del servidor donde se encuentran alojadas.
- Como el navegador es la herramienta utilizada para ejecutar las aplicaciones web, éstas dependen de la configuración del mismo, para que puedan trabajar de forma óptima, ya sea en cuanto a las *cookies*, JavaScript o archivos Flash y la versión que el usuario tenga instalada.
- Si el servidor donde está almacenada la aplicación web y la información no tiene los niveles de seguridad necesarios para resguardar los datos, éstas pueden ser vulnerables a los ataques de agentes maliciosos.

2.2.4 Arquitectura Cliente-Servidor

Las soluciones o implementaciones de software generalmente están relacionadas a conceptos como la arquitectura, lenguajes debido a las ventajas que estos ofrecen, esto refleja cómo están interconectados todos los elementos, como se comunicarán y que tipo de tecnologías y protocolos utilizarán. En este caso nos concentraremos en la arquitectura cliente/servidor es un modelo para el desarrollo de sistemas de información, en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos.

Así mismo, se denomina cliente al proceso que inicia el diálogo o solicita los recursos, y servidor al proceso que responde a las solicitudes. Es el modelo de interacción más común entre aplicaciones en una red (Mendoza M, 2015). A continuación, se presenta la Figura N° 4, la cual describe gráficamente el modelo:

2.2.5 Características de la Arquitectura Cliente-Servidor

A continuación, se explicarán características de la arquitectura cliente servidor para lograr un mejor entendimiento de la misma:

- La combinación de un cliente que interactúa con el usuario, y un servidor que interactúa con los recursos a compartir. El proceso del cliente proporciona la interfaz entre el usuario y el resto del sistema. El proceso del servidor actúa como un motor de software que maneja recursos compartidos (UCI).
- Se establece una relación entre procesos distintos, los cuales pueden ser ejecutados en la misma máquina o en máquinas diferentes distribuidas a lo largo de la red.
- La relación establecida puede ser de muchos a uno, en la que un servidor puede dar servicio a muchos clientes, regulando su acceso a los recursos compartidos.

- El ambiente es heterogéneo. La plataforma de hardware y el sistema operativo del cliente y del servidor no son siempre los mismos. Precisamente una de las principales ventajas de esta arquitectura es la posibilidad de conectar clientes y servidores independientemente de sus plataformas.
- No existe otra relación entre clientes y servidores que no sea la que se establece a través del intercambio de mensajes entre ambos. El mensaje es el mecanismo para la petición y entrega de solicitudes de servicios.
- El concepto de escalabilidad tanto horizontal como vertical es aplicable a cualquier sistema Cliente-Servidor. La escalabilidad horizontal permite agregar más estaciones de trabajo activas sin afectar significativamente el rendimiento. La escalabilidad vertical permite mejorar las características del servidor o agregar múltiples servidores.

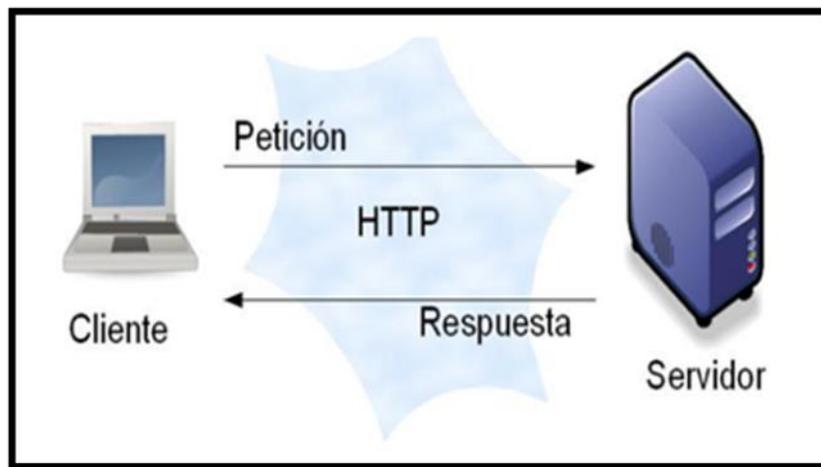


Figura N° 4. Arquitectura Cliente-Servidor.

Fuente: <http://grupo-701-marcenestore.wikispaces.com/file/view/WWW%5B1%5D.png/228568614/WWW%5B1%5D.png>

2.2.6 Ventajas de la Arquitectura Cliente-Servidor

En este orden de ideas, se presentan las ventajas y/o fortalezas que nos brinda la arquitectura cliente servidor:

- Centralización del control: los accesos, recursos y la integridad de los datos son controlados por el servidor de forma que un programa cliente defectuoso o no autorizado no pueda dañar el sistema (Dan Harkey, 2002).
- Escalabilidad: se puede aumentar la capacidad de clientes y servidores por separado. Cualquier elemento puede ser aumentado (o mejorado) en cualquier momento, o se pueden añadir nuevos nodos a la red (clientes y/o servidores).
- Fácil mantenimiento: al estar distribuidas las funciones y responsabilidades entre varios ordenadores independientes, es posible reemplazar, reparar, actualizar, o incluso trasladar un servidor, mientras que sus clientes no se verán afectados por ese cambio (o se afectarán mínimamente).

2.2.7 Limitaciones de la Arquitectura Cliente-Servidor

De igual manera, tenemos las siguientes limitaciones:

- Congestión del tráfico: cuando una gran cantidad de clientes envían peticiones simultáneas al mismo servidor, puede ser que cause muchos problemas para éste (a mayor número de clientes, más problemas para el servidor).
- Cuando un servidor está caído, las peticiones de los clientes no pueden ser satisfechas.
- El software y el hardware de un servidor son generalmente muy determinantes. Un hardware regular de un ordenador personal puede no servir a cierta cantidad de clientes. Normalmente se necesita software y hardware específico, sobre todo en el lado del servidor, para satisfacer el trabajo.

2.2.8 Patrón de Diseño MVC (Modelo-Vista-Controlador)

Los patrones son una forma, una manera, una técnica para resolver problemas en el desarrollo de software. Un patrón resulta ser una solución a un problema determinado, para que este sea considerado un patrón debe poseer ciertas características como lo son: haber comprobado su efectividad resolviendo problemas similares, y en qué circunstancias obtuvo mejores resultados, etc.

Un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. Las aplicaciones web consisten de tres tipos de objetos. El Modelo, es la capa donde se trabaja con los datos, por tanto, contiene mecanismos para acceder a la información. Cabe mencionar que en esta capa es habitual utilizar un ORM (*Object Relational Mapping*, Mapeo Objeto Relacional), que permita trabajar con abstracción de bases de datos y persistencia en objetos, en vez de usar directamente sentencias SQL (*Structured Query Language*, Lenguaje de Sentencias Estructuradas), que suele depender del motor de base de datos con el que se esté trabajando.

Segundo, La Vista especifica la visualización de los datos, algunas veces conocida como lógica de presentación, es decir, es la interfaz de usuario que permite mostrar los datos del modelo e interactuar con ellos mediante acciones o eventos

Tercero, El controlador es el coordinador entre estos dos últimos, es decir, define la forma en que la interfaz de usuario reacciona ante la entrada de usuario. MVC desacopla el concepto de interfaz de usuario y lógica de negocio para aumentar la flexibilidad y modularidad del software, posiblemente permitiendo que el código pueda ser reutilizado.

Finalmente, la idea es separar responsabilidades entre las personas que trabajan para un proyecto de desarrollo de software; es decir, descomponer el problema en módulos funcionales (entre ellos el diseño gráfico), lo que se traduce en enfocar de una forma reduccionista la solución de un proyecto de software.

Aunque se pueden encontrar diferentes implementaciones de MVC, el flujo de control generalmente es el siguiente:

- El usuario interactúa con la interfaz de alguna manera (ejemplo: presionando un botón, un enlace).
- El controlador recibe (por parte de los objetos de la interfaz vista) la notificación de la acción solicitada por el usuario.
- El controlador accede al modelo, posiblemente actualizando los datos enviados por el usuario.
- El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario.

- La vista usa el modelo para generar la interfaz apropiada para el usuario donde se reflejan los cambios en el modelo.
- En algunas implementaciones la vista no tiene acceso directo al modelo, dejando que el controlador envíe los datos del modelo a la vista.
- La interfaz espera por nuevas interacciones de usuario para iniciar nuevamente el ciclo.

A continuación, se presenta la Figura N° 5, la cual muestra el flujo de control general del patrón de diseño MVC:

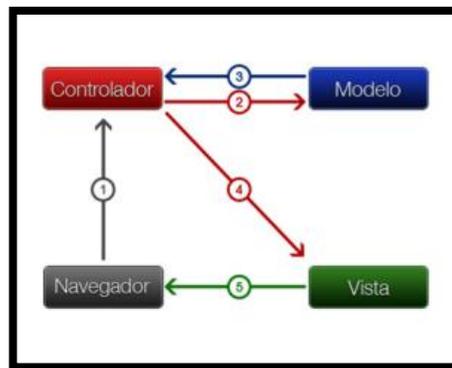


Figura N° 5. Patrón de Diseño MVC.

Fuente: <http://dwpalma.com/public/images/posts/MVC.jpg>

2.2.9 Herramientas Tecnológicas para el Desarrollo de la Aplicación Web

Una aplicación web está normalmente estructurada como una aplicación de tres-capas. En su forma más común, el navegador web (Cliente) ofrece la primera capa, un motor capaz de usar alguna tecnología web dinámica (Servidor) que constituye la capa intermedia. Por último, una base de datos constituye la tercera y última capa (Caivano, 2009) .

En este orden de ideas, el navegador web manda peticiones a la capa intermedia que ofrece servicios valiéndose de consultas y actualizaciones a la base de datos y a su vez proporciona una interfaz de usuario.

- Tecnologías del lado del Cliente: Aquellas que se ejecutan en el navegador del usuario, son las páginas dinámicas que se procesan en el cliente. En estas páginas toda la carga de procesamiento de los efectos y funcionalidades la soporta el navegador. Las tecnologías utilizadas en este caso son HTML, CSS, JavaScript, sus librerías y marcos de trabajo.
- Tecnologías del lado del servidor: Se encargan del procesamiento de peticiones de usuario mediante la interpretación de un script en el servidor web para generar contenido HTML dinámicamente como respuesta. Entre las más destacadas se encuentran ASP, PHP, Java (JSP), Python y Ruby.
- Sistema manejador de Base de Datos: Es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos.

A continuación, se presenta, en la Figura N° 6, la Arquitectura de Aplicaciones Web:

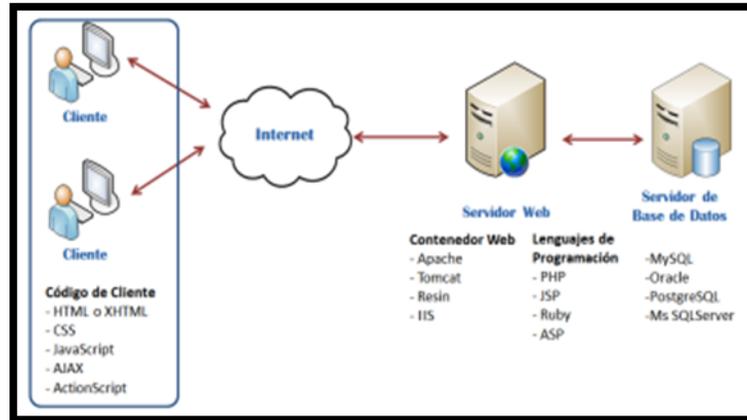


Figura N° 6. Arquitectura Aplicaciones Web.

Fuente: http://imagenes.mailxmail.com/cursos/imagenes/9/3/web-site-funcionamiento-tipos-de-programacion_22839_1_1.jpg

2.2.10 HTML (Lenguaje de Marcado de Hipertexto)

El Lenguaje de Marcas de Hipertexto, mejor conocido como HTML (*HiperText Markup Language*). HTML fue diseñado por el físico Tim Berners Lee en 1989, se publicó en un documento formal en 1991 y se convirtió en un estándar para el desarrollo de páginas web en 1995, ya que ofrece una estructura básica para la definición del contenido que por lo general se encuentra en una página web, como imágenes, texto, entre otros.

A partir de 1996, la World Wide Web Consortium (W3C), que es la organización que define la mayoría de los estándares para el desarrollo web, se ha encargado de publicar constantes revisiones (HTML 3.2, HTML 4, HTML 4.01) de este estándar, ampliándolo y depurándolo con cada versión. HTML 5 fue la última revisión aprobada por esta organización, en el año 2014.

HTML 5 establece una serie de nuevos elementos y atributos que reflejan el uso típico de los sitios web modernos. Incorpora nuevas etiquetas con mayor significado semántico, mejora la integración con las tecnologías de desarrollo web actual, permite desplegar objetos y animaciones en formato 3D (tridimensional), tiene un mejor soporte multimedia y ahorra la necesidad de tener que procesar medios de audio o video a través de complementos instalados en el navegador web.

2.2.11 Hojas de Estilos en Cascada (CSS)

Las Hojas de Estilo en Cascada o (*Cascading Style Sheets*, o sus siglas CSS) hacen referencia a un lenguaje de hojas de estilos usado para describir la presentación semántica (el aspecto y formato) de un documento escrito en lenguaje de marcas. Su aplicación más común es dar estilo a páginas webs escritas en lenguaje HTML y XHTML, pero también puede ser aplicado a cualquier tipo de documentos XML, incluyendo SVG y XUL.

Por ende, cuando se observan las distintas versiones de Las Hojas de Estilo en Cascadas o CSS sabemos que es un lenguaje cuya definición ha sido creada por el World Wide Web Consortium, cuyas siglas son W3C. Básicamente se trata de un consorcio internacional que produce especificaciones, recomendaciones, y herramientas en relación al desarrollo de internet. Por lo tanto, W3C trabaja continuamente para mejorar el lenguaje CSS, corrigiendo errores e incorporando nuevas funcionalidades. Antes de llegar a una especificación o recomendación oficial se trabajan numerosos borradores que son sometidos a revisión y corrección. (Consortium W. W., w3, 2018d).

Así mismo, las versiones de CSS a lo largo de su historia han sido las siguientes:

- CSS 1: publicada en 1996.
- CSS 2: publicada en 1998.
- CSS 2.1: publicada en 2004.
- CSS 3: publicada en 2011.
- CSS 4: se estima que pueda ser especificación oficial en 2019 (depende de la W3C)

Por ende la versión con los mayores estándares a nivel mundo es el CSS 3 el cual ha ido evolucionando, según la W3C con las definiciones oficiales realizadas hasta el año 2017 tenemos algunas de las siguientes mejoras a la especificación (Consortium W. W., W3, 2018c) (Consortium W. W., CCS 2017 Features, 2018a):

- *CSS Syntax Level 3*: Reemplaza los estándares CSS2 4.1, CSS2 4.2, CSS2 4.4 en como CSS es analizado en el navegador.
- *CSS Style Attributes*: Define como pueden ser añadidos las declaraciones de CSS en los atributos de marcado.
- *Media Queries Level 3*: Reemplaza el estándar CSS2 7.2 y expande la sintaxis para estilos, dimensiones específicas.
- *CSS Namespaces*: Introduce una regla basada en el uso de "@namespace" que permite tener selectores predefinidos en dicho espacio u alcance.
- *Selectors Level 3*: Reemplazan los estándares CSS2 5.0, CSS2 6.4.3 y definen un rango de extensión para los selectores.
- *CSS Color Level 3*: Reemplazan los estándares CSS2 4.3.6, CSS2 14.1, y definen un rango de colores extendidos. Al mismo tiempo que se introduce la propiedad *opacity*.
- *CSS Image Values and Replaced Content Level 3*: Proporciona un nuevo texto base para el tamaño de los elementos reemplazados (como las imágenes), agregando controles adicionales a su tamaño y orientación, e introduce la sintaxis de los degradados como imágenes en CSS
- *CSS Fonts Level 3*: Reemplaza el estándar CSS2 18.1 y define la propiedad '*cursor*', '*outline*' junto a conjunto de cambios que mejoran la interfaz de usuario.
- *CSS Writing Modes Level 3*: Define el soporte de CSS para varios modos de escritura internacionales, como de izquierda a derecha (por ejemplo, latín o índico), de derecha a izquierda (por ejemplo, hebreo o árabe), bidireccional (por ejemplo, mezcla de latín y árabe) y vertical (por ejemplo, secuencias de comandos asiáticas).
- *CSS Animation, Transitions*: Se definen un conjunto de propiedades que facilitan la animación, movimiento y agregación de dinamismo en los elementos que interactúan a través de CSS.

Así mismo, existen algunos inconvenientes o limitaciones aun en CSS3 que según la W3C serán puntos tomados en las mejoras para CSS4, algunos de ellos son:

- El uso de las tablas nos permitía crear diseños complejos de forma mucho más sencilla que utilizando CSS, aunque CSS3 está intentando facilitar dicho trabajo.
- Debido a malas prácticas de programación es muy fácil repetir propiedades de muchos elementos (*Mixing*).

Adicionalmente existe una lista de características estables y muchas otras que serán mejoradas en sus futuras versiones. (Consortium W. W., 2018b).

2.2.12 Bootstrap

Bootstrap o *twitter-bootstrap* es un *framework* creado originalmente por dos desarrolladores/diseñadores de *twitter* para acelerar el diseño de nuevas aplicaciones web.

“El *framework* proporciona clases CSS y código JavaScript para definir el *layout* de la página, crear componentes que respondan a eventos y estilizar los elementos HTML más habituales” (Otto, 2015).

La mayor ventaja es que podemos crear interfaces que se adapten a los distintos navegadores (*responsive design*) apoyándonos en un *framework* potente con numerosos componentes webs que nos ahorraran mucho esfuerzo y tiempo.

Podemos decir que los principios en los que se basa son:

- **Responsive Design:** consiste en que la página trata de “hacer lo correcto” al ser visualizada Independientemente del dispositivo y tamaño de la pantalla.
- **Mobile first:** al contrario que en la versión 2, en la 3, el diseño responsivo es la opción por defecto al trabajar con Bootstrap.
- **Cross Browser:** ser compatible con la mayoría de navegadores.
- **Integración con jQuery:** está muy integrado con esta librería, para la que define nuevos *plugins*.
- **Buenas prácticas:** Trata de emplear algunas de las prácticas más extendidas en cuanto a usabilidad, uso de CSS3/HTML5 generan una mejor organización del código.

2.2.13 JavaScript

JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos basado en prototipos, imperativo, débilmente “tipado” y dinámico (Newemage, 2011).

Se utiliza principalmente en su forma del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.

Entre las ventajas de JavaScript se destaca:

- Ligero de carga.
- Fácil de integrar.
- Cientos de aplicaciones disponibles para uso.
- Puede agregar interactividad a elementos web (menús desplegables, efectos visuales, manipular datos).

2.2.14 jQuery

- Es una librería JavaScript muy rápida y ligera que simplifica el desarrollo de la parte de cliente de las aplicaciones web (jQuery, 2015). En otras palabras, jQuery incluye muchas utilidades para crear fácilmente las páginas web de las aplicaciones dinámicas complejas.

- Consiste en un único archivo JavaScript que contiene las funcionalidades comunes de DOM, eventos, efectos y AJAX (*Asynchronous JavaScript And XM*, JavaScript Asíncrono y XML).
- La característica principal de la biblioteca es que permite cambiar el contenido de una página web sin necesidad de recargarla, mediante la manipulación del árbol DOM y peticiones AJAX. Entre las características de jQuery se encuentran las siguientes:
 - Selección de elementos DOM.
 - Interactividad y modificaciones del árbol DOM.
 - Eventos.
 - Manipulación de la hoja de estilos CSS.
 - Efectos y animaciones.
 - Animaciones personalizadas.
 - AJAX.
 - Soporta extensiones.
 - Obtención de información del navegador, operar con objetos y vectores, funciones para rutinas comunes.

2.2.15 Django

Holovaty indica que “Django es un *Framework* de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como Modelo-Vista-Controlador (MVC)”. (Holovaty, 2009)

La meta fundamental de Django es facilitar la creación de sitios web complejos. Django pone énfasis en el re-uso, la conectividad y extensibilidad de componentes, el desarrollo rápido y el principio No te repitas (DRY, del inglés *Don't Repeat Yourself*). Python es usado en todas las partes del *framework*, incluso en configuraciones, archivos, y en los modelos de datos. (Holovaty, 2009). Este *framework* fue desarrollado por Adrian Holovaty, Simon Willison, Jacob Kaplan-Moss y Wilson Miner mientras trabajaban en World Online, y originalmente se utilizó para administrar tres sitios web de noticias.

Los orígenes de Django en la administración de páginas de noticias son evidentes en su diseño, ya que proporciona una serie de características que facilitan el desarrollo rápido de páginas orientadas a contenidos. Por ejemplo, en lugar de requerir que los desarrolladores escriban controladores y vistas para las áreas de administración de la página.

La distribución principal de Django aglutina aplicaciones que proporcionan un sistema de comentarios, herramientas para direccionar contenido vía RSS y/o Atom, "páginas planas" que permiten gestionar el contenido sin necesidad de escribir controladores, y un sistema de redirección de URLs.

Otras características de Django son:

- Un “mapeador” objeto-relacional u *Object-Relational Mapping* (ORM).
- Aplicaciones modulares que pueden instalarse en cualquier página gestionada con Django.
- Una API de base de datos robusta.
- Un sistema incorporado de "vistas genéricas" que ahorra tener que escribir la lógica de ciertas tareas comunes.

- Un sistema extensible de plantillas basado en etiquetas, con herencia de plantillas.
- Un despachador de URLs basado en expresiones regulares.
- Un sistema "*middleware*" para desarrollar características adicionales; por ejemplo, la distribución principal de Django incluye componentes *middleware* que proporcionan cacheo, compresión de la salida, normalización de URLs, protección CSRF y soporte de sesiones.
- Soporte de internacionalización, incluyendo traducciones incorporadas de la interfaz de administración.
- Documentación incorporada accesible a través de la aplicación administrativa (incluyendo documentación generada automáticamente de los modelos y las bibliotecas de plantillas añadidas por las aplicaciones).

2.2.16 Tecnologías del lado del Servidor de Bases de Datos

Silberschatz indica que un Sistema Manejador de Base de Datos (SMBD) “consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos”, su objetivo principal “es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto práctica como eficiente”. (Silberschatz A., 2002)

En la actualidad existe una gran variedad de sistemas manejadores de bases de datos relacionales y no relacionales, muchos de ellos son propietarios o comerciales como Oracle, Microsoft SQL Server, IBM Informix, Sybase, etc. y otros no comerciales o de código abierto como MongoDB, Redis, DynamoDB, MySQL y PostgreSQL siendo este último el que se explica a continuación.

2.2.17 PostgreSQL

PostgreSQL es un potente Sistema Manejador de Base de Datos objeto-relacional, de código abierto, que cuenta con más de 15 años de desarrollo activo y una arquitectura probada que se ha ganado una sólida reputación por su fiabilidad, integridad de datos y corrección (Group., (1996-2015)).

Trabaja muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema, por lo que se considera una herramienta que favorece a los usuarios con sistemas empresariales de gran tamaño.

Así mismo, este SMBD se distribuye bajo la Licencia PostgreSQL, es una licencia similar a la de la Distribución de Software de Berkeley (BSD) y a la del Instituto de Tecnología de Massachusetts (MIT), que permite a los usuarios hacer cualquier cosa que quieran con el código, incluyendo la reventa de los binarios sin el código (Group., (1996-2015))

2.2.18 Ventajas de PostgreSQL

Las principales ventajas de PostgreSQL son:

- Es código abierto.
- La velocidad de respuesta se mantiene al aumentar el tamaño de la base de datos, cosa que no sucede con otros programas que se suelen poner lentos.
- Proporciona estabilidad y confiabilidad.
- Tiene una gran capacidad de almacenamiento.

- Soporta gran número de peticiones simultáneas a la base de datos de forma correcta. Puede operar sobre distintas plataformas, incluyendo Linux, Windows, Unix, Solaris y MacOS X.
- Provee un buen sistema de seguridad mediante la gestión de usuarios, grupos de usuarios y contraseñas.
- Soporta los tipos de datos, cláusulas, funciones y comandos de tipo estándar SQL92/SQL99 y extendidos propios de PostgreSQL.

2.2.19 Desventajas de PostgreSQL

Por otro lado, las desventajas de PostgreSQL son:

- En comparación con otros sistemas manejadores de base de datos, como por ejemplo MySQL, es más lento en inserciones y actualizaciones, ya que cuenta con cabeceras de intersección.
- Cuenta con muchos foros oficiales de ayuda, pero no con una documentación de ayuda obligatoria.
- La sintaxis de algunos comandos o sentencias no es tan intuitiva.

2.3 Resumen del Capítulo

Al momento de plantear una solución a un problema mediante el uso de algún lenguaje de programación, *framework*, etc. es vital contextualizarnos detenidamente en el problema, evaluar qué características requerimos en nuestra solución y cual(es) lenguajes o *framework* se adecuan mejor, o cual nos ofrece mejores ventajas en el proceso de desarrollo.

Primero, a nivel de lenguaje de programación la elección fue Python por la gran facilidad de uso, legibilidad, portabilidad, simplicidad. Dicho lenguaje fue diseñado para expresar en forma clara y directa las instrucciones que debe seguir un programa, sin necesidad de indicar detalles de bajo nivel tales como los tipos de variables, el tamaño de estructuras de datos o el manejo de la memoria. Todo eso es trabajo del intérprete, que es suficientemente "listo" como para inferir todos esos detalles por sí mismo. Así mismo, cuanto a versionado se refiere no posee grandes diferencias entre sus versiones 2.7 y 3.7, lo cual facilita la realización de mejoras en las funcionalidades existentes o la migración desde la versión 2.7 hacia la 3.7.

Segundo, en cuanto a *framework* la elección fue Django el cual es **con diferencia, el mayor *framework* web basado en Python**. Se apoya en una comunidad grande y activa. Ofrece formularios basados en modelos (*model-based*), tiene su propio lenguaje de plantillas y cuenta con una excelente documentación que está disponible de manera gratuita. Técnicamente, Django utiliza un ORM (*Object Relational Mapper*) para asignar sus objetos a tablas de base de datos. El mismo código funciona con diferentes bases de datos y facilita la portabilidad entre diferentes tipos de bases de datos.

Tercero, cuando nos referimos a bases de datos la elección fue PostgreSQL debido a la gran facilidad de integración con el ORM de Django, su estabilidad y confiabilidad, la gran adecuación para ambientes donde existan altos volúmenes de datos, la variedad de tipos de datos soportadas unido a los tipos de datos propios, y la contabilidad a través de las distintas plataformas y sistemas operativos.

Finalmente, Los *frameworks* y/o tecnologías elegidas para el desarrollo del lado del cliente fueron Bootstrap, JQuery y HTML5 todas poseen gran extensibilidad, facilidad de uso, documentación bien fundamentada (basada en estándares), son multiplataforma ya que dependen únicamente del navegador web. De igual manera, la arquitectura cliente servidor, el patrón de desarrollo MVC junto a lo antes mencionado conforman la base para realizar la implementación.

CAPÍTULO III. Marco Metodológico

La palabra metodología. Metodología es un término generado a partir de tres palabras de origen griego: *metà* (“más allá”), *odòs* (“camino”) y *logos* (“estudio”). Una metodología de desarrollo de software es la forma como se aborda el diseño de un sistema, este puede ser medido sobre el tiempo o sobre el alcance de un proyecto que se lleva a cabo, donde diferentes procesos o fases, que parten desde que se definen los requerimientos hasta que se coloca en producción una aplicación. Por otro lado, puede haber iteraciones sobre el mantenimiento y actualización del proyecto que incluyen cambios en los requerimientos.

En este orden de ideas, siempre es importante para los procesos ser actualizados constantemente para mejorar su rendimiento o para corregir fallas con el fin de generar un mejor producto final. Por ende, las metodologías de desarrollo en estas se basan en criterios como requerimientos del sistema, capacidad del equipo de desarrollo, flexibilidad del tiempo de las entregas, arquitectura del sistema, capital humano, herramientas que faciliten la implementación de la solución, entre otros criterios. Debido a lo descrito anteriormente se generan las metodologías ágiles de desarrollo de software.

Así mismo, Las metodologías ágiles son las más adecuadas para proyectos donde es necesaria la flexibilidad y la adaptación a los cambios, como es el caso de las aplicaciones web, que suelen tener requerimientos variables y usualmente se desea que estén disponibles en poco tiempo. Algunas metodologías ágiles de uso común en la actualidad son: Programación Extrema (XP), Proceso Unificado Ágil (AUP), Scrum, AgilUs, Kanban, entre otras.

3.1 Metodología Scrum

Primero, Scrum es “una metodología de desarrollo muy simple, que requiere trabajo duro, porque la gestión no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias de la evolución del proyecto”. Éste a su vez expresa que es un modo de desarrollo de carácter adaptable, orientado a las personas antes que a los procesos y emplea desarrollo ágil (iterativo e incremental) (Palacio, 2008).

Así mismo, como características adicionales tenemos que es ligero, fácil de entender y difícil de llegar a dominar. Muestra la eficacia relativa de las prácticas de gestión de producto y las prácticas de desarrollo, de modo que se pueda ir mejorando en el proceso de desarrollo. Dicha metodología intenta adaptarse al cambio en los proyectos de software, asumiendo que es bueno, necesario y que va a ocurrir. Su objetivo principal es lograr que el producto funcione satisfaciendo adecuadamente las necesidades del cliente.

Es por ello que se realizan planificaciones y estimaciones que tienen un periodo de tiempo de 2 a 4 semanas. El equipo de desarrollo toma decisiones importantes en las tareas a desarrollar y determina el tiempo de duración de cada una de ellas.

3.1.1 Teoría de Scrum

Las bases fundamentales de este marco de trabajo son: los Equipos Scrum, los roles, los eventos, los artefactos y las reglas asociadas. Cada uno de estos componentes sirve a un propósito específico y es esencial para el éxito de Scrum y para su uso. (Schwaber, 2013)

Scrum emplea un enfoque iterativo e incremental para optimizar la predictibilidad y el control del riesgo. A continuación, se describen los tres pilares que soportan toda la implementación del control de procesos empírico:

- **Transparencia:** los aspectos significativos del proceso deben ser visibles para aquellos que son responsables del resultado. La transparencia requiere que dichos aspectos sean definidos por un estándar común, de tal modo que se comparta un entendimiento común de lo que se está viendo.

- **Definición de “Terminado” (*Definition of “Done”*):** todos los miembros del Equipo deben tener un entendimiento compartido de lo que significa que un elemento de la Pila de producto se describa como “Terminado”, para asegurar la transparencia. El propósito de cada *sprint* es entregar incrementos de funcionalidad que se puedan poner en producción, y que se ajusten a la definición de “Terminado” actual del Equipo Scrum. Cada Incremento se integra con todos los Incrementos anteriores asegurando que todos los Incrementos funcionan en conjunto.
- **Inspección:** los usuarios de Scrum deben inspeccionar frecuentemente los artefactos de Scrum y el progreso hacia un objetivo, para detectar variaciones. Su inspección no debe ser tan frecuente como para que interfiera en el trabajo. Las inspecciones son más beneficiosas cuando se realizan de forma diligente por inspectores expertos, en el mismo lugar de trabajo.
- **Adaptación:** si un inspector determina que uno o más aspectos de un proceso se desvían de límites aceptables, y que el producto resultante no será aceptable, el proceso o el material que está siendo procesado deben ser ajustados. Dicho ajuste debe realizarse cuanto antes para minimizar desviaciones mayores. Scrum tiene cuatro eventos formales, contenidos dentro del *sprint*, para la inspección y adaptación los cuales son:
 - Reunión de planificación del *sprint* (*Sprint Planning Meeting*).
 - Scrum diario (*Daily Scrum*).
 - Revisión del *sprint* (*Sprint Review*).
 - Retrospectiva del *sprint* (*Sprint Retrospective*).

3.1.2 El Equipo Scrum

El Equipo Scrum está conformado por un Dueño del producto (*Product Owner*), el Equipo de desarrollo (*Development Team*) y un Facilitador (*Scrum Master*). Los Equipos Scrum son auto organizado y multifuncional. Los equipos auto organizados eligen la mejor forma de llevar a cabo su trabajo y no son dirigidos por personas externas al equipo. Los equipos multifuncionales tienen todas las competencias necesarias para llevar a cabo el trabajo sin depender de otras personas que no son parte del equipo. El modelo de equipo en Scrum está diseñado para optimizar la flexibilidad, la creatividad y la productividad.

A continuación, se describe cada uno de los roles pertenecientes al Equipo Scrum:

- **Equipo de desarrollo (*Development Team*):** está conformado por los profesionales que realizan el trabajo de entregar un Incremento de producto “Terminado”, que potencialmente se pueda poner en producción, al final de cada iteración. Solo deben haber cambio en los miembros del equipo entre las iteraciones.
- **Facilitador (*Scrum Master*):** es un líder que está al servicio del Equipo Scrum, responsable de asegurar que Scrum sea entendido y adaptado, ayudando a todos en cada una de las interacciones con el fin maximizar el valor creado por el Equipo Scrum. Además, se caracteriza por las siguientes características:
 - 50% desarrollador, no sólo perfil gerencial.
 - Escucha el progreso y remueve los impedimentos.
 - Es el moderador de los eventos.
- **Dueño del producto (*Product Owner*):** es el responsable de maximizar el valor del producto y del trabajo del Equipo de Desarrollo. El cómo se lleva a cabo esto puede variar entre distintas

organizaciones, Equipos Scrum e individuos. El Dueño del producto es una única persona, no un comité y es responsable de gestionar la Pila del producto (*Product Backlog*). La gestión de la Pila del producto incluye:

- Definir los elementos de la Pila del producto.
- Priorizar los elementos en la Pila del producto para alcanzar los objetivos y misiones de la mejor manera posible.
- Revisar junto con otros involucrados el producto al final de cada *sprint*.
- Asegurar que la Pila del producto es visible, transparente y clara para todos, y que muestra aquello en lo que el equipo trabajará a continuación.
- Asegurar que el Equipo de Desarrollo entiende los elementos de la Pila del producto al nivel necesario.
- Para que el Dueño del producto pueda hacer bien su trabajo, toda la organización debe respetar sus decisiones. Las decisiones del Dueño de producto se reflejan en el contenido y en la priorización de la Pila del producto.

3.1.3 Artefactos

Los artefactos definidos por Scrum están diseñados para maximizar la transparencia de la información clave, que es necesaria para asegurar que todos tengan el mismo entendimiento del artefacto.

- **Pila del producto (*Product Backlog*):** es una lista ordenada de todo lo que podría ser necesario para construir el producto, y es la única fuente de requisitos para cualquier cambio a realizarse en el producto. Es priorizada por el Dueño del producto, esto puede cambiar en la planificación de cada *sprint*. La Pila del producto enumera todas las características, funcionalidades, requisitos, mejoras y correcciones que constituyen cambios a ser hechos sobre el producto para entregas futuras. Los elementos de la Pila del producto tienen como atributos la descripción, la ordenación, la estimación y el valor.
- **Pila de pendientes del *sprint* (*Sprint Backlog*):** es el conjunto de elementos de la Pila de producto seleccionados para el *sprint*, más un plan para entregar el Incremento de producto y conseguir el objetivo del *sprint*. La Lista de Pendientes del *sprint* es una predicción hecha por el Equipo de Desarrollo acerca de qué funcionalidad formará parte del próximo Incremento y del trabajo necesario para entregar esa funcionalidad en un Incremento “Terminado”.
- **Incremento:** El Incremento es la suma de todos los elementos de la Pila del producto completados durante un *sprint* y el valor de los incrementos de todos los *sprints* anteriores. Al final de un *sprint*, el nuevo Incremento debe cumplir la Definición de “Terminado” del Equipo Scrum.

3.1.4 Eventos

- **Reunión de planificación del *sprint* (*Sprint Planning Meeting*):** El trabajo a realizar durante el *sprint* se planifica en la Reunión de planificación del *sprint*. Este plan se crea mediante el trabajo colaborativo del Equipo Scrum completo y tiene un máximo de duración de ocho horas para un *sprint* de un mes. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito.
- **Scrum diario (*Daily Scrum*):** Es una reunión con un bloque de tiempo de 15 minutos para que el Equipo de Desarrollo sincronice sus actividades y cree un plan para las siguientes 24 horas. Esto se lleva a cabo inspeccionando el trabajo avanzado desde el último Scrum Diario y haciendo una proyección acerca del trabajo que podría completarse antes del siguiente. El Scrum Diario se realiza a la misma hora

y en el mismo lugar todos los días para reducir la complejidad. Durante la reunión, cada miembro del Equipo de Desarrollo explica: ¿Qué hice ayer?, ¿Qué voy a hacer hoy?, ¿Tengo impedimentos?

- **Revisión del *sprint* (*Sprint Review*):** Se basa en una reunión informal, no una reunión de seguimiento, para inspeccionar el Incremento y adaptar la Pila del producto si fuese necesario. Durante la Revisión del *sprint*, el Equipo Scrum y los interesados colaboran acerca de lo que se hizo durante el *sprint*. Basándose en esto, y en cualquier cambio a la Pila del producto durante el *sprint*, los asistentes colaboran para determinar las siguientes cosas que podrían hacerse para optimizar el valor. El resultado de la Revisión del *sprint* es una Pila del producto revisada, que define los elementos de la Pila del producto posibles para el siguiente *sprint*.
- **Retrospectiva del *sprint* (*Sprint Retrospective*):** Es una oportunidad para el Equipo Scrum de inspeccionarse a sí mismo y crear un plan de mejoras que sean abordadas durante el siguiente *sprint*. La Retrospectiva del *sprint* tiene lugar después de la Revisión del *sprint* y antes de la siguiente Reunión de planificación del *sprint*. El Scrum Master alienta al equipo para que mejore su proceso de desarrollo y sus prácticas para hacerlos más efectivos y amenos para el siguiente *sprint*. Para el final de este evento, el Equipo Scrum debería haber identificado mejoras que implementará en el próximo *sprint*.

(Schwaber, 2013)

3.2 Metodología Kanban

En la actualidad existen muchas y variadas formas de manejar la gestión de proyectos o metodologías de desarrollo. Todo depende del tipo de trabajo que se esté realizando: repetir una receta o una serie de patrones, en este caso conocidos en la manufactura, o crear un nuevo producto.

Por ende, dentro de la gestión de proyectos, cualquiera sea la que se elija, una de las realidades que menos se considera es: **¿Cómo lograr que un equipo haga un buen trabajo y a la vez “lo pase bien”?**

Existe una propuesta llamada *Lean Thinking* (“Pensamiento Esbelto”) [*Lean Thinking*] (Womack & Jones, 2003), inspirada en el “Conocimiento Profundo” y en los “14 Puntos” de W. Edwards Deming (1900-1993) [W. Edwards Deming], adoptada por Taichi Ohno en Toyota [The Toyota Way] (Liker, 2004) en los dos tipos de trabajo que se pueden realizar:

- Manufactura: *Toyota Production System* (TPS).
- Desarrollo de Nuevos Productos: *Toyota Product Development System*.

Estos puntos se oponen a la rigidez de los procesos, y a que los procesos son más importantes que las personas. Al automatizar un proceso, se olvida que hay gente que los maneja, los critica y los mejora. Si en algún momento se requiere mejorar los procesos, lo harán personas que no entienden cuál es el real problema de negocio. Si existe alguien que puede hacer un trabajo, es la persona más capaz de realizarlo, considerando todos los derechos y responsabilidades que ello significa. Para la organización de las responsabilidades de los obreros de la compañía Toyota, no se ocupan complejos sistemas como los ERP, sino que se realiza en simples tableros con “**Tarjetas-Señal**” (Kanban en japonés) como el que se muestra en la Figura N° 7 (Tablero Kanban Básico), que gatilla las acciones que se deben realizar. Estos Tableros Kanban logran una efectiva auto-organización del personal, asignando las responsabilidades respectivas.



Figura N° 7. Tablero de Kanban Básico.

Fuente: <https://kanbantool.com/es/metodologia-kanban>

3.2.1 Ciclo de Vida

El Kanban es una tarjeta física que se utiliza en el Sistema de Producción de Toyota (TPS - *Toyota Production System*) para soportar un control productivo descentralizado por demanda (“pull”).

Así mismo, Kanban es una herramienta proveniente de la filosofía Lean mencionada anteriormente, de tipo “pull”, lo que significa que los recursos deciden cuándo y cuánto trabajo se comprometen a hacer. Los recursos toman (“pull”) el trabajo cuando están listos, en lugar de tener que recibirlo (“push”) desde el exterior. Al igual que una impresora tira en la página siguiente sólo cuando está lista para imprimir sobre ella. Kanban se basa en la optimización de procesos continuos y empíricos, que se corresponde con el principio de *Kaizen Lean*.

Dicho principio enfatiza la respuesta al cambio por sobre seguir un plan (generalmente Kanban permite una respuesta más rápida que Scrum). La mayoría de los métodos ágiles como Scrum y XP ya están bien alineados con los principios de Lean.

En 2004, sin embargo, David Anderson fue pionero en una aplicación más directa de *Lean Thinking* y Teoría de las Restricciones para el desarrollo de software. Bajo la guía de expertos como Don Reinertsen, desarrollaron un "sistema Kanban para el desarrollo de software". Así, mientras Kanban aplicado al desarrollo de software es algo relativamente nuevo, Kanban tal como se utiliza en *Lean Production* lleva más de un siglo y medio. Kanban comparte con otras metodologías como FDD o Scrum la idea de crear un *Backlog* del producto que tenga una serie de *items* (*user stories, features, etc.*) priorizados.

Aunque la principal diferencia con otras aproximaciones ágiles, es que en Kanban no existen las iteraciones tipo *timebox, sprints, etc.* En su lugar, Kanban se centra en controlar el WIP (*Work In Progress*). Esto es, cuando hay poco WIP, se añade el *item* más prioritario del *Product Backlog*, y se controla que nunca se supere una cierta cantidad de WIP. Dadas sus características, no se adapta a un desarrollo basado en entregas, y actualmente se utiliza especialmente en entornos de mantenimiento (corrección de *bugs, etc.*). Al igual que Scrum o XP, Kanban podría formar parte de las herramientas de proceso de desarrollo ágil de cualquier compañía.

De hecho, es común que Kanban pueda combinarse con los actuales procesos que usted tenga para el desarrollo de software, y existen muchos casos que lo utilizan para complementar (o reemplazar) a Scrum.

3.2.2 Tableros Kanban

El trabajo con Tableros Kanban se basa en un marco teórico, convenciones que han sido acordadas por los grupos que han propuesto este enfoque de cambio de administración y evoluciones propias que se han observado de las organizaciones que lo han implementado (ver Figura N° 8).

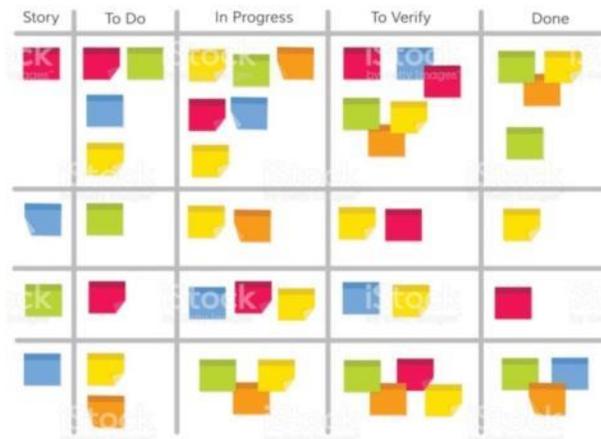


Figura N° 8. Tablero de Kanban para desarrollo de software.

Fuente: <https://www.istockphoto.com/mx/vector/pizarra-con-publicar-notas-para-desarrollo-%C3%A1gil-de-software-tablero-con-notas-gm692728328-127847853>

Es un enfoque para cambiar la administración, un *framework* para catalizar el cambio en una organización. Kanban usa la limitación del “Trabajo en Curso” o “*Work In Progress*” (WIP) como mecanismo de control para incentivar las discusiones de cambio. Si no se respetan los límites del WIP y no se conversan los problemas, el trabajo se estancará y no se podrá mejorar.

Por ende, las discusiones sobre mejoras son objetivas, gracias a que la visualización, mediciones y claridad de las políticas y los modelos de *Lean*, *Theory of Constraints* (Teoría de Restricciones), y las enseñanzas de W. Edwards Deming, le permite al equipo analizar científicamente sus problemas y proponer soluciones.

Esta metodología tiene un enfoque más evolutivo que revolucionario de lo que estamos acostumbrados para asumir el cambio. Con Kanban, se comienza con un proceso que ya existe. Se “Kanbaniza” mapeando el flujo, visualizándolo, y limitando el WIP para crear un *pull system*. Se mantienen intactos los roles existentes, responsabilidades, cargos, y prácticas. Los únicos cambios se realizan en la interfaz con quienes nos relacionamos en el trabajo, por ejemplo, jefaturas y operaciones. Cualquier cambio que se haga, será específicamente elegido para evitar agitar la jerarquía social, o invocar una respuesta emocional defensiva de la gente afectada.

Esto provoca un cambio evolutivo. Inicialmente, esto significa Optimización de Procesos (*Kaizen*), pero gradualmente, mientras la organización madura en sus capacidades, se transforma en grandes cambios administrados (*Kaikaku*). Se ha observado que eventos *Kaizen* progresivos llevan a una madurez organizacional mejorada, y a provocar niveles de cambio de *Kaikaku* más dramáticos.

Está diseñado como un enfoque que permitirá personalizar y evolucionar un proceso existente, sin importar donde se encuentre el proceso. Por lo tanto, es imposible definir un test de conformidad para Kanban.

Se utilizan cinco propiedades principales para catalizar el comportamiento emergente de la evolución. Estas cinco propiedades son:

- Visualizar el *Workflow*.
- Limitar el *Work-In-Progress*.
- Medir el flujo.
- Explicitar las políticas de los procesos.
- Usar modelos para evaluar oportunidades de mejoras.

Estas propiedades representan las cinco prácticas que deben estar presentes para que el enfoque Kanban funcione. El proceso del equipo para el desarrollo de software y la administración de proyectos, siempre será única, y con el tiempo será adaptada a medida, optimizada para darle valor al flujo de trabajo, medir el riesgo, capacidades y habilidades del equipo, demanda del cliente, cuellos de botella y variaciones de distintos índices que pueden afectar al equipo.

En este orden de ideas, no puede haber un test de conformidad. Cualquier medida que se pueda aplicar. Impulsa mecanismos para simplificar la coordinación de los elementos de todo un sistema. Por ejemplo, la combinación de la visualización y de la limitación del WIP (*pull system*) permite una interfaz simplificada con las jefaturas. Como resultado, la mayoría de las organizaciones que adoptan Kanban no necesitan el concepto de “*Product Owner*” (Dueño del Producto), y pueden fácilmente enfrentar múltiples canales de entrada, encolando requerimientos

Las “*Stand-up Meetings*” (Reuniones de Pie) diarias se ha demostrado que son efectivas incluso con 50 o más personas. La razón es porque el equipo confía implícitamente en que el trabajo que se muestra en la visualización se está realizando.

Los equipos iterarán sobre los tickets de trabajo, en vez de los miembros del equipo. Las tres preguntas se obvian. Equipos Kanban más maduros reducen la discusión sólo al trabajo que está bloqueado o defectuoso, enfocándose sólo en las excepciones en vez del trabajo que está avanzando con normalidad.

3.2.3 Revisión de Tableros Kanban

Actualmente los tableros Kanban tienen un gran número de aplicaciones en distintas áreas. Sin embargo, a continuación, desglosaremos las implementaciones más recurrentes basadas en los tableros virtuales.

3.2.4 Tableros Virtuales

Los Tableros Kanban Virtuales nacen de la necesidad de modernizar un simple sistema de administración para, por ejemplo, coordinar equipos que no trabajan en conjunto, poder hacer mediciones sobre la información contenida, poder acceder a la información desde cualquier lugar, obtener la historia de un proyecto, etc.

3.2.5 Ventajas y Desventajas de los Tableros Kanban

Los tableros Kanban presentan las siguientes ventajas:

- Facilidad de uso.
- Flexibilidad.
- Visibilidad.
- Concurrencia de múltiples usuarios (incluso un equipo entero).

- Facilidad para generar reportes.
- Persistencia de la información.
- Cálculo de métricas (tiempo promedio de una tarea en comenzar, tiempo de ciclo).
- Portabilidad.

Por otro lado, tenemos las siguientes desventajas:

- Dificultad de realizar las entregas a tiempo en grandes proyectos (dado que no hay un control específico del tiempo empleado en cada actividad, en grandes proyectos pueden acumularse un gran número de pequeños retrasos que provocarían la demora en la entrega del producto final).
- Falta de reglas (aunque la existencia de pocas reglas es una ventaja, puede convertirse en un problema cuando el desarrollador es inexperto y necesita una guía para realizar su trabajo. Por ello, se aconseja hacer uso de Kanban tras haber ganado experiencia con otras metodologías, ya que de este modo habrá reglas que se habrán interiorizado).
- Dificultad a la hora de prever posibles problemas (aunque la localización y solución de problemas es sencilla en Kanban, su previsión antes de que ocurran no lo es).

3.3 Razones que justifican la selección de la metodología

La elección de una metodología al momento de generar un producto de software final siempre es complicada debido a la gran variedad que existen. Sin embargo, el enfoque utilizado en el desarrollo según las metodologías antes mencionadas permitió no solo genera la documentación necesaria sino también un proceso acelerado de desarrollo enfocándose en el producto final.

Es por ello que se utilizó una composición de dos metodologías como lo son Scrum y Kanban donde se utilizara correspondiendo a cada tipo de interacción. A continuación, se desglosa la forma de trabajo unificada para ambas metodologías.

Primero, de Scrum se tomó los siguientes aspectos referentes al flujo de trabajo, modelo de negocio e interacción con el cliente:

- Permite involucrar más al cliente con el desarrollo del producto, lo que genera mayor satisfacción.
- Permite mayor integración entre todas las personas involucradas en el proceso de desarrollo del proyecto.
- Busca la mejora continua durante todo el proceso de desarrollo en todos los niveles y aspectos del proyecto.
- Está abierto al cambio, acepta los cambios sin ningún problema buscando siempre la forma de maximizar beneficios y/o minimizar impactos negativos.
- Permite tener una buena organización a nivel interno y externo.
- Permite establecer tiempos fijos de entrega, lo cual impide que se atrase cualquier entregable al cliente, siempre y cuando las condiciones para el desarrollo estén disponibles al cien por ciento.
- Permite reconocer los errores y fortalezas de los integrantes del equipo, con el fin de buscar la mejora constante a medida que avanza el proyecto.

Segundo, de Kanban se utilizó el desarrollo controlado de la aplicación a través de un tablero virtual Kanban enfocado en la entrega y avance incremental del producto final donde se podrá observar el continuo avance. Todo esto teniendo en cuenta las ventajas antes mencionadas que nos ofrece esta versátil pero sencilla metodología.

3.4 Resumen del Capítulo

Al tomar ciertos elementos de distintas metodologías mencionadas se buscó explotar lo mejor de cada una de ellas. La metodología Kanban nos permitió mantener a la vista todo el trabajo en progreso (WIP) de forma más sencilla, al mismo tiempo se logró segmentar las actividades de forma atómica con gran facilidad, detectar los cuellos de botella con las actividades que no permitían el avance efectivo.

En cuanto a Scrum, utilizando los *Daily Scrum* o reuniones diarias en plazo más largo nos facilitó generar metas cortas para cada *sprint*, al mismo tiempo recibir retroalimentación a través de correcciones e incluso iterar en la corrección de errores o gestión de mejoras en los siguientes *sprints*.

Al combinar los *sprint reviews* con la ventaja visual del flujo del tablero que otorga Kanban fuimos capaces de diferenciar en qué tareas estábamos fallando, cuáles eran las tareas atrasadas y donde existían cuellos de botella, esto fue una gran ventaja ya que nos permitió ir solventando las actividades pautadas en el desarrollo de forma más rápida.

Al no tener definidos un *product owner* las revisiones se hicieron con plazos mayores, por lo cual en ciertos momentos se perdió ciertos avances en la velocidad de desarrollo en las funcionalidades. Así mismo, al centrar en los objetivos por *sprint* resultó un software más enfocado en el producto final, genero una herramienta enfocada en aprovechar mejor los tiempos para el proceso de arbitraje y en mejorar todos los procesos internos que eso conlleva dentro de AsoVAC.

En cuanto a generación de documento se refiere, el panel Kanban nos permitió tener un conteo de todas las actividades realizadas para llevar a cabo el desarrollo, y así poder tener porcentajes de avances en cada *sprint*, esto junto a la creación de tareas atómicas nos facilitó el proceso de desarrollo.

En conclusión, el uso de estas dos metodologías combinadas nos permitió aprovechar bondades que existen en ambas para ser capaces de desarrollar de una forma más rápida e incremental, se aprovechan las características de cada metodología según el enfoque del desarrollo por ende es provechoso el utilizar Scrumban.

CAPÍTULO IV. Marco Aplicativo

El objetivo de este capítulo es dar a conocer las funcionalidades desarrolladas para el Sistema de Gestión de Arbitraje de Artículos de Investigación, se describen los requerimientos, la forma en la cual fue aplicada la metodología de desarrollo Scrumban, los modelos creados, los diferentes módulos que componen el sistema, las principales interfaces y las pruebas realizadas sobre dicho sistema.

4.1 Requerimientos del Sistema de Gestión de Arbitraje de Artículos Académicos

4.1.1 Requerimientos Funcionales

- Desarrollar funcionalidad para el registro de:
 - Trabajos, registro de los trabajos, autores y co-autores donde se incluya el pago de la convención.
 - Árbitros, asignación de trabajos, evaluaciones de los trabajos, envío de la información al autor y co-autores de las correcciones a realizar.
 - Cronograma del evento, áreas, subáreas y coordinadores de área.
- Generación de certificados mediante la exportación de la información registrada por los usuarios a formato .pdf.
- Desarrollar interfaz de configuración para administrar y respaldar la información de cada convención.

4.1.2 Requerimientos no Funcionales

- Disponibilidad.
- Mantenibilidad.
- Seguridad.
- Usabilidad.

4.2 Arquitectura del Sistema

La arquitectura de la solución planteada para el sistema de arbitrajes para la Asociación Venezolana para el Avance de la Ciencia es la que se muestra a continuación:

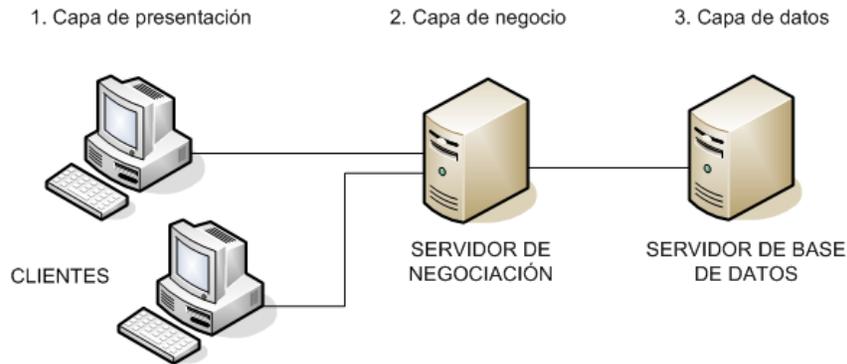


Figura N° 9. Arquitectura General.

Fuente: Creación propia

Así mismo, la arquitectura en la web que usaremos será la indicada en la Figura 9 Como SMBD utilizaremos PostgreSQL en su versión 9.6, ya que es una de las bases de datos más potentes de software libre, con un alto rendimiento, seguridad y a su vez está disponible para todas las versiones de los sistemas operativos Unix y también Windows. Su versatilidad y robustez la hace el candidato perfecto para el sistema que se busca implementar.

En este orden de ideas, se propone utilizar el *framework* de Django (versión 1.11.3) basado en Python (versión 2.7) para el desarrollo del sistema. Siendo uno de los muchos *frameworks* que están establecidos sobre la base del patrón MVC en sus beneficios se encuentra la separación de responsabilidad y organización del código.

Este *framework* incorpora un patrón llamado Modelo-Plantilla-Vista (*Model Template View*). Los *templates* son el componente donde se implementan todas las interfaces de los usuarios, que serán desplegadas por las vistas en el lado del servidor y los modelos corresponde al diseño de la base de datos donde se almacenara la información. Además, posee un sistema jerárquico de plantilla que proporciona la reutilización de código y la extensibilidad de las aplicaciones. Cabe destacar que Django posee un buen soporte para PostgreSQL, la base de datos que se piensa utilizar para el sistema.

4.3 Roles de los actores

Según el proceso de revisión por pares durante un proceso de arbitraje desarrollado en capítulos anteriores se definen un conjunto de roles en este sistema. A continuación, describiremos los roles según las funcionalidades desarrolladas:

- **Administrador:** Es el responsable de la carga inicial de la información necesaria para la entrada en funcionamiento del sistema, puede crear coordinadores de áreas, árbitros y asignar trabajos a estos últimos.
- **Coordinador de general:** Es el responsable de administrar todo el proceso de arbitraje basado en la revisión por pares, sus atribuciones son similares a las del administrador con la principal diferencia que utiliza las funcionales de la aplicación en los estados correspondientes.
- **Coordinador de área:** Es el responsable de crear los árbitros y asignarles trabajos inscritos.
- **Árbitros de subárea:** Es seleccionado por el coordinador de área para realizar la revisión de un artículo determinado, solo tiene acceso a los artículos a los cuales ha sido asignado para revisar.
- **Autor:** Son aquellos que están habilitados para enviar archivos y documentos dentro del sistema.

4.4 Aplicación de la Metodología combinada Scrumban

4.4.1 Cronograma tentativo de actividades (Tablero Kanban)

En la Tabla N° 1 se muestran las actividades que se van a realizar durante el desarrollo del sistema y el tiempo tentativo para la realización de cada una de ellas a través de la metodología Scrumban.

Tabla N° 1. Cronograma tentativo de actividades

Fuente: creación propia

Sprint	Actividad	días
1	Instalación y configuración del ambiente de desarrollo.	1
	Diseño de la estructura de las tablas para la base de datos.	12
	Modelado del proceso de arbitraje a nivel de aplicación	2
2	Creación, diseño y corrección de las interfaces.	10
2	Creación de los modelos	2
2	Revisión y corrección de los modelos	2
2	Definir estados del proceso de arbitraje	1
3	Iniciar sesión, recuperar contraseña, registro de usuario, control de acceso por roles y estado del arbitraje	3
3	Crear Inicio, navegación a través del <i>navbar</i> (barra de navegación) y <i>sidebar</i> (barra lateral de navegación)	2
3	Envío de correos electrónicos, creación de plantillas de correo electrónico	2
3	Generación de plantillas para certificados digitales	2
3	Navegación del administrador	2
3	Navegación del coordinador general	2
3	Realización de pruebas funcionales del inicio de sesión, recuperar contraseña, acceso por roles y envío de correos electrónicos con certificado.	2
4	Navegación de coordinador de área	2
4	Navegación del árbitro de subárea	2
4	Navegación del autor	1
4	Generación de certificados digitales	4
4	Crear navegación para los roles de administrador, coordinador general y coordinador de área según el estado del proceso de arbitraje	4
4	Realización de pruebas funcionales sobre la navegación de los distintos roles sobre la plataforma según el estado del proceso y la generación de los certificados digitales	2
5	Crear una página 404	1
5	Validar flujo de administrador, coordinador general y coordinador de área según el estado del proceso	1
5	Generación de claves según el nivel de accesos	2
5	Asignación de roles	3
5	Agregar coordinador de área, árbitros mediante carga de datos masivos	2
5	Crear navegación para autores y árbitros de subárea	2
5	Desarrollo de interfaces de Organizadores, Locación y Eventos	2

5	Realización de pruebas funcionales teniendo en cuenta rol del usuario, solicitud de clave de accesos según el mismo, cargar masiva por importación de datos con formatos de pruebas definidos, y comprobación de las plantillas de las interfaces para organizadores, locación y eventos	2
6	Mejoras Generales de las interfaces – Estandarización de estilos	7
6	Mejoras de seguridad y accesos a los URL de la aplicación	8
7	Crear, editar, eliminar y actualizar trabajos por parte del autor	5
7	Flujo de Postular trabajos por parte del autor	6
7	Realización de pruebas funcionales para las operaciones de crear, editar, eliminar, actualizar y postular un trabajo por parte del autor	4
8	Flujo del autor para todos los estados del proceso de arbitraje	4
8	Crear, editar y actualizar los autores del proceso de arbitraje	2
8	Desarrollar un filtro de área y subáreas para segmentar la información	3
8	Relacionar las áreas y subáreas con el coordinador general, coordinador de área y árbitro de subárea	2
8	Pruebas funcionales de la creación, edición, actualización de los autores.	2
8	Pruebas funcionales de la utilización del filtro del área y subárea	2
9	Agregar áreas y subáreas en el registro de usuarios	3
9	Confirmación y verificación de acceso al arbitraje según su rol y clave maestra	2
9	Agregar opciones de acceso hacia el arbitraje y los eventos de forma independiente	3
9	Operaciones de crear, editar, consultar, actualizar y eliminar para locaciones	3
9	Pruebas funcionales para agregar las áreas y subáreas. La confirmación y verificación del acceso al arbitraje según su rol y las operaciones de crear, agregar y eliminar para las locaciones	4
10	Operaciones de crear, editar, consultar, actualizar y eliminar para organizadores	3
10	Operaciones de crear, editar, consultar, actualizar y eliminar para los eventos.	3
10	Mejora de usabilidad en todas las tablas del arbitraje AsoVAC utilizando <i>Bootstrap tables</i> (tablas basadas en el <i>framework</i> Bootstrap).	6
10	Pruebas funcionales de las operaciones de creación, modificación, actualización y eliminación (conocido por sus siglas en inglés como operaciones CRUD) de organizador y eventos. Así como la revisión del funcionamiento de las tablas utilizando la implementación <i>Bootstrap table</i> (tablas basadas en el <i>framework</i> Bootstrap).	3
11	Configuración inicial de Bases datos al iniciar la aplicación	1
11	Agregar opción para cambiar de subáreas de un árbitro de subárea	1
11	Agregar y editar información a los autores	3
11	Descarga y carga de Autores utilizando archivos .xlsx	2
11	Carga de trabajos utilizando archivos .xlsx	2
11	Ingresar contenido de postular trabajo	2
11	Mostrar campo de observaciones en los trabajos	2
11	Permitir comentarios de trabajos	1
11	Pruebas funcionales de la configuración inicial, cambio de subáreas de los árbitros, editar información de autores, descarga y carga de autores y trabajos.	1

12	Agregar opción de verificación de pago del trabajo	2
12	Carga de datos utilizando archivos .xlsx para áreas y subáreas	2
12	Carga y asignación de coordinador de área, y árbitros de subárea utilizando archivos .xlsx	1
12	Descarga y carga de archivos .xlsx de coordinadores de área, y árbitros de subárea	3
12	CRUD de árbitros de subárea	2
12	Validaciones en el registro de autores	1
12	Agregar <i>tokens</i> de confirmación al asignar trabajos	2
12	Mejoras de usabilidad en todos los modales de la aplicación	1
12	CRUD de trabajos asignados a cada árbitro de subárea	1
12	Pruebas funcionales de la verificación del pago de los trabajos, CRUD de árbitros de subárea y trabajos asignados a los árbitros de áreas junto a las verificaciones de carga, asignación y descarga de coordinadores de área y árbitros de subárea utilizando archivos .xlsx con un formato definido.	1
13	CRUD del proceso de arbitraje	3
13	Agregar código a áreas y subáreas	2
13	CRUD de Asignación de Sesiones	3
13	Agregar observaciones al asignar una nueva sesión	2
13	Agregar funcionalidad de editar datos básicos	2
13	Parametrización de imagen de cabecera y <i>footer</i> (pie de página) para envíos de correos de forma dinámica	2
13	Pruebas funcionales sobre el CRUD del proceso de arbitraje, asignación de sesiones junto a la agregación de código de áreas y subáreas. Así como la validar las observaciones en la asignación de sesiones. Parametrización de funciones de administración del arbitraje.	1
14	Generación dinámica para certificado de trabajo aprobados enviados vía correo electrónico	3
14	Generación dinámica para certificado de trabajo aprobados con observaciones enviados vía correo electrónico	3
14	Generación dinámica para certificado de trabajo rechazado enviados vía correo electrónico	3
14	Correcciones y mejoras de usabilidad y experiencia de usuario en un conjunto de vistas	3
14	Generación dinámica de los gráficos que reflejan los resultados utilizando Chartjs	3
15	Configuración base para todos los certificados en el sistema de arbitraje	2
15	Configuración, y actualización de IVA, montos de arbitraje en datos básicos	2
15	Generación de porta nombres para asistentes, autores y conferencistas	2
15	Generación de certificados para comisión organizadora	2
15	Generación de certificados para coordinadores de sesiones	2
15	Generación de certificados masivos utilizando archivos .xlsx	2
15	Validación de generación de certificados individuales y masivos usando archivos .xlsx	3

4.4.2 Descripción de las iteraciones y listas de tareas en cada iteración

A continuación, se describirá cuáles fueron las actividades realizadas durante cada *sprint* presentado anteriormente en la Tabla N°1.

Sprint 1: Se comenzó con la instalación de la aplicación, las cuales fueron Python 2.7.9, PostgreSQL 9.6 y Django 1.11.

Así mismo, se realizó el modelado de la base de datos en conjunto con los diagramas asociados al sistema (Modelo relación, Modelo entidad-relación). Además, se modeló el proceso de arbitraje y revisión de trabajos usados en AsoVAC los cuales están basados en la revisión por pares descrita en capítulos anteriores. En otro orden de ideas, se instalaron todas las dependencias necesarias en un ambiente virtual para la realización del sistema.

Sprint 2: Durante este *sprint* se realizaron el diseño y la maquetación de las principales interfaces, se definieron los estilos bases para la aplicación, así como también la configuración de los archivos estáticos utilizando Bootstrap 3.3.7.

De igual manera se crearon los modelos con las correcciones pertinentes en el modelo de datos tras aplicar la tercera forma normal (3FN) en el modelo de base de datos utilizando el ORM que posee Django.

Así mismo, para darle continuidad al modelado de la lógica de negocio se establecieron nueve estados bases en el proceso de arbitraje bien identificados y delimitados (identificados de forma numérica de uno al nueve a nivel de implementación). Todo esto para gestionar los procesos internos de los actores durante el proceso de arbitraje.

Sprint 3: Se realizó todo el proceso de autenticación contra la aplicación utilizando ciertas ventajas a nivel de seguridad que nos ofrece Django como *framework*. Análogamente el manejo de sesiones, recuperación de contraseña, reinicio de contraseña, acceso por roles y estados del proceso.

De manera consecutiva se generaron muchas plantillas y/o formatos para reutilizar en la aplicación. En primer lugar, se generaron las plantillas de los componentes principales de la barra de navegación (*navbar*) y de barra lateral de navegación (*sidebar*). En segundo, todas las plantillas relacionadas al envío de correos que manejara toda la aplicación. En tercer lugar, las plantillas para la generación de los distintos certificados digitales que podrán ser emitidos en ciertos estados del proceso.

Así mismo, se realizó toda la navegación en la aplicación para el administrador y el coordinador general para definir cuales funcionalidades utilizan según estado del proceso y el flujo de cada uno.

Para finalizar, se realizaron pruebas funcionales y pruebas unitarias de registro, inicio de sesión, recuperación de contraseña, envío de correo para funciones básicas y generación de plantillas base para certificados digitales.

Sprint 4: Continuando con el proceso de desarrollo de la navegación para el resto de los roles de usuario se hicieron el flujo del coordinador de área, árbitro de subárea y autor los cuales tiene intervenciones bien definidas según su rol y el estado del proceso de arbitraje.

Aunado a esto, se desarrollaron todos los modelos de certificados de forma dinámica, dichos certificados se descargan y se envían adjuntos en los correos electrónicos que genera la aplicación, por ejemplo, cabe mencionar los certificados de aceptación de trabajo, certificados de rechazo, certificados de participación, entre otros.

De igual manera, el desarrollo de las navegaciones para los roles de administrador, coordinador general y coordinador de área tomando en cuentas todos los estados existentes en el proceso del arbitraje.

Para concluir, los componentes y funcionalidades desarrollados pasaron por un conjunto de pruebas funcionales sobre la navegación de los distintos roles sobre la plataforma según el estado del proceso y la generación de los certificados digitales.

Sprint 5: Durante este momento del desarrollo se creó una vista para las peticiones o indicar fallos en la aplicación en caso de que este se llegase a caer en algún momento. Así mismo, se validó el flujo de accesos del administrador, coordinador general y coordinador de área desarrollado en el *sprint* anterior.

De igual manera, se creó un nivel adicional de seguridad tras realizar la generación de claves maestras según el rol de cada usuario con permisos en la plataforma junto a esto la asignación de los roles y la carga masiva de datos mediante importación de archivos con extensión .xlsx para los coordinadores de área y árbitros de subárea.

Análogamente, se implementó el flujo de navegación para los árbitros de subáreas y los autores para todos los estados del proceso de arbitraje. Aunado a la generación de las interfaces sobre los eventos, organizadores y locaciones para los eventos durante el proceso de arbitraje. Junto con las funcionalidades de crear, editar, eliminar y actualizar los organizadores, locaciones y los eventos del proceso de arbitraje de AsoVAC. Así mismo, se realizó la validación de accesos según el rol del usuario.

Sprint 6: Se implementaron mejoras en todas las interfaces referentes al diseño adaptativo, a la utilización de buenas prácticas de programación, aplicación y mejoras en interacciones y mejor utilización del sistema de plantillas que ofrece Django.

De igual manera, se desarrollaron mejoras en el ámbito de control de acceso a la aplicación, seguridad en la navegación en la misma y restricciones de acceso según los distintos roles existentes en la aplicación.

En este orden de ideas, se realizaron pruebas funcionales para el control de acceso basados en el rol y en el estado del proceso de arbitraje.

Sprint 7: Se desarrolló la lógica de negocios correspondiente a la creación, edición, eliminación y actualización de los trabajos por parte de los autores. Así como también, el postular trabajo para el proceso de arbitraje incluyendo el detalle de los pagos, facturas y todo lo concerniente al gasto monetario asociado.

Así mismo, se culmina el flujo de postular trabajos por parte del autor en el cual los autores agregan su trabajo de investigación asociado a alguna área y especialidad para poder ser arbitrado en el proceso de arbitraje.

Para culminar, durante este *sprint* se realizó de pruebas funcionales para las operaciones de crear, editar, eliminar, actualizar y postular un trabajo por parte del autor y otros roles en los distintos estados del sistema de arbitraje.

Sprint 8: Se implementaron y revisaron conjuntamente el flujo del autor para cada uno de los estados del proceso del arbitraje, cuales acciones puede realizar dicho rol de usuario al utilizar la aplicación.

Análogamente, se desarrollaron las operaciones de creación, edición y actualización de los autores, estos son agregados por un administrador, por un coordinador general o por un coordinador de área.

Así mismo, se le agrega al registro de usuarios la relación con las áreas y subáreas. Acompañado del desarrollo de un filtro de áreas que permitirá ver toda la información relacionada con el área o subárea específica que se desea consultar, esto mejora y segmenta el proceso del arbitraje.

Para finalizar, se realizaron pruebas funcionales del filtro de área y subárea, la creación, edición y actualización de los autores que participan del proceso de arbitraje. Así como la relación de las áreas y subáreas con los roles antes mencionados.

Sprint 9: Se incorporaron los campos de áreas y subáreas en el registro de todos los usuarios para cualquier proceso de arbitraje. De igual manera, la confirmación y verificación de acceso al arbitraje según el rol de cada usuario a través de la clave maestra o clave de seguridad.

Así mismo, se realizó un flujo de creación, edición, eliminación y actualización de locaciones, independiente del sistema de arbitraje para facilitar y aumentar la versatilidad de los eventos internos de AsoVAC.

Para finalizar, se realizaron pruebas funcionales de la creación de usuarios con áreas y subáreas. Adicionando la verificación del correcto funcionamiento de la confirmación de la clave maestra según el rol de dicho usuario (coordinador general, coordinador de área, y coordinador de subárea). Culminando con las pruebas del flujo del creación, edición, modificación y eliminación sobre locaciones, organizador y eventos de forma independiente al proceso del arbitraje.

Sprint 10: En este *sprint* se desarrollaron las de operaciones de crear, editar, consultar, actualizar y eliminar para organizadores y eventos.

Así mismo, se realizó una nueva implementación de las tablas para mejorar la usabilidad, experiencia de usuario y diseño de interfaces existentes en la aplicación utilizando las tablas de Bootstrap.

Para finalizar, se validó el correcto funcionamiento de todas las tablas de la aplicación tras la mejora a través de *bootstrap table* (tablas basadas en el *framework* Bootstrap), todo esto seguido de la prueba funcional del flujo creación, lectura, actualización y eliminación de organizadores y eventos.

Sprint 11: Durante este *sprint* se realizaron desarrollos sobre la configuración inicial de la base de datos, cambios en la asignación de las subáreas para los árbitros de subárea. Así como también, carga de trabajos a través de archivos .xlsx con un formato definido.

De igual manera, se verifico la funcionalidad de descarga y carga de autores utilizando archivos .xlsx, y la carga del trabajo en formato pdf al momento de postular el trabajo para ser arbitrado posteriormente.

Finalmente, se verificó la acción de mostrar, y agregar comentarios en los trabajos siendo coordinador de área, árbitro de subárea y que estos fueran visibles para los autores.

Sprint 12: Continuando con el flujo del desarrollo enfocado en los trabajos se agrega la verificación del pago del trabajo en sus diferentes modalidades. Así como también la carga, descarga y asignación de los coordinadores de área, árbitros de subárea utilizando archivos .xlsx para ambos propósitos.

De igual manera, se verificó el funcionamiento del CRUD de árbitros de subárea; las validaciones en el registro de los autores, la agregación de forma correcta de un token de confirmación al realizar la asignación del trabajo.

Para finalizar, se agregaron mejoras de usabilidad en los modales, y se verificaron los cambios realizados junto al CRUD de los trabajos asignados a cada árbitro de subárea para dar la posterior asignación de sesiones.

Sprint 13: En este *sprint* el desarrollo se centró en la asignación de sesiones y en las acciones relacionadas, se realizó el CRUD del proceso de arbitraje de un trabajo, acompañado de la asignación de códigos para las áreas y subáreas.

Así mismo, se validó el correcto desarrollo del CRUD sobre la asignación de sesiones, y el proceso de agregar observaciones sobre una sesión existente por parte de un coordinador de área o un coordinador general.

Finalmente, se mejoró la generación de las plantillas ya que se parametrizó un encabezado que es posible cambiar dinámicamente y un conjunto de colores dentro del formato del pie de página el cual afecta a todos los correos existentes en la plataforma.

Sprint 14: En este *sprint* se desarrolló la generación de los certificados de aprobación, aprobación con observaciones y rechazo de los trabajos para ser enviados vía correo electrónico.

En este orden de ideas, se hicieron correcciones en la gran mayoría de las vistas en cuanto a usabilidad y experiencia de usuario se refiere.

Para finalizar, se definieron los gráficos base para el apartado de resultados los cuales son construidos dinámicamente con la información totalizada del sistema de arbitraje en curso.

Sprint 15: En este *sprint* se desarrolló la generación de ciertos certificados para usuarios que no interactúan de lleno utilizando la aplicación del sistema de arbitraje, pero son parte importante del proceso, esto consistió en generar certificados para el comité organizador, los coordinadores de sesiones, porta nombres para asistentes, autores y conferencista.

Así mismo, se realizó la generación de varios de estos certificados utilizando archivos .xlsx con sus respectivas validaciones.

Para finalizar, se generó una configuración por defecto para la generación base de los certificados y al mismo tiempo la actualización del proceso de postulación de los trabajos para configurar el IVA, y el monto necesario para el proceso de arbitraje.

4.5 Descripción de Flujos Principales

A continuación, se definen los flujos o funcionalidades principales según el rol de usuario y el estado del proceso en el que se encuentra el sistema de arbitraje.

4.5.1 Flujo coordinador de general

El rol de coordinador general posee todas las responsabilidades, este rol es asignado por el administrador el cual tiene control completo de todas las funcionalidades.

- **Estado Desactivado:** En este estado no tiene acceso a la aplicación, ya que es aquí donde es asignado por el administrador, no es posible avanzar hacia el estado iniciado sino se ha elegido al coordinador.
- **Estado Iniciado:** Una vez asignado por el administrador el coordinador general podrá hacer los siguientes flujos.

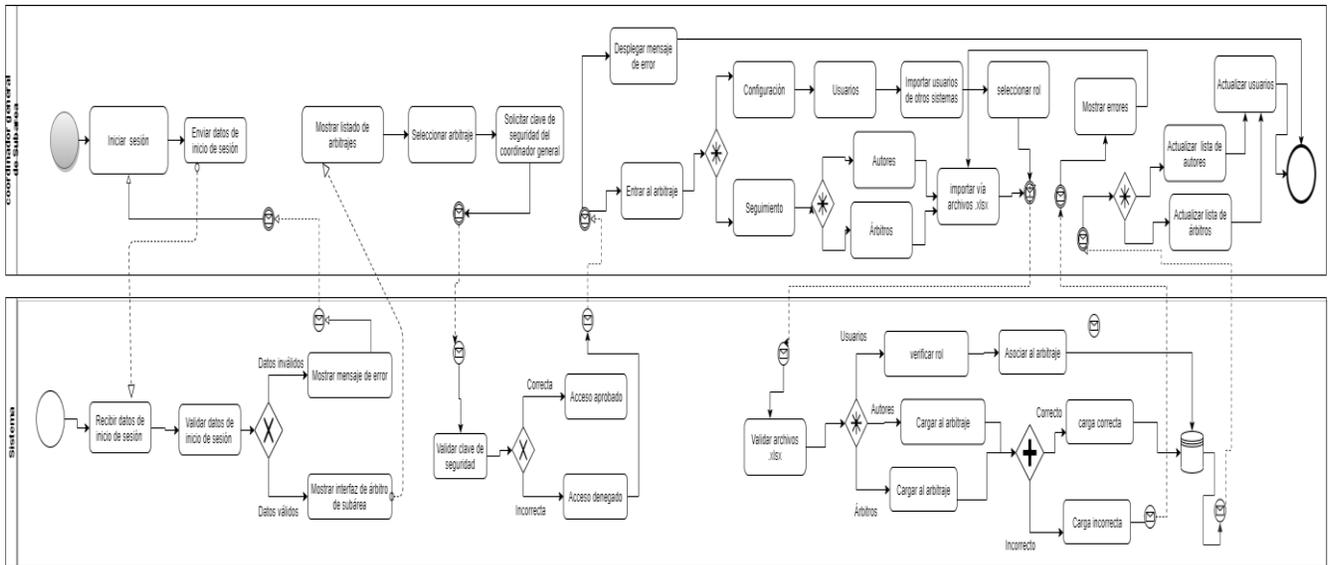


Figura N° 10. Flujo del coordinador general - estado iniciado

Fuente: Elaboración propia

- **Estado En selección y asignación de coordinadores de área:** En este estado la asignación de los coordinadores de área es la actividad principal en cuanto al avance del sistema de arbitraje se refiere.

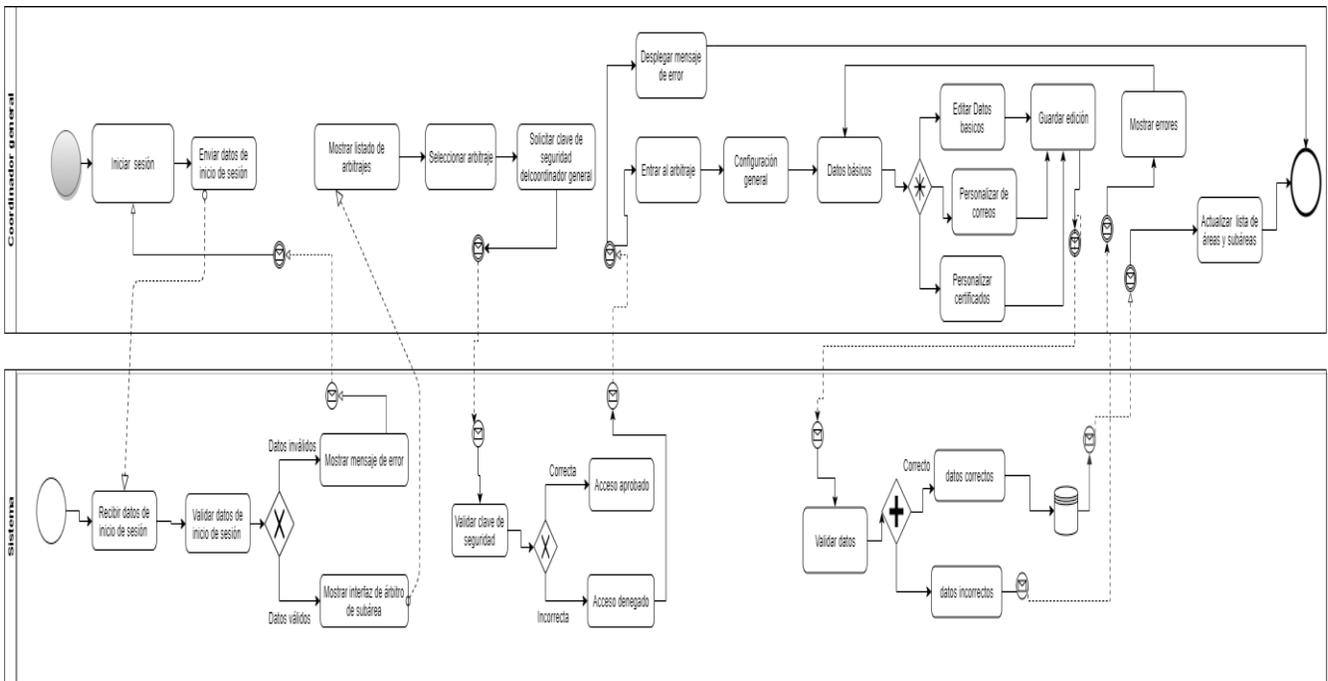


Figura N° 11. Flujo del coordinador General - estado selección y asignación de coordinadores de área – Editar datos básicos del arbitraje

Fuente: Elaboración propia

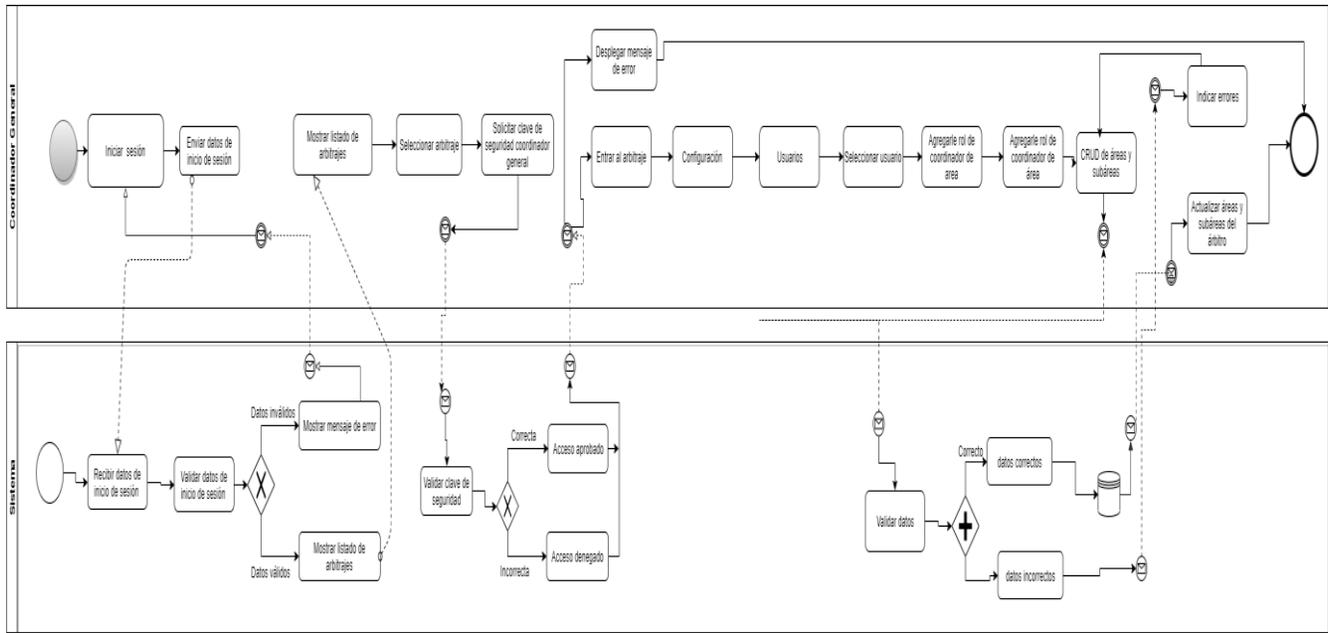


Figura N° 12. Flujo del coordinador general - estado en selección y asignación de coordinadores de área – asignar árbitros de subárea

Fuente: Elaboración propia

- **Estado En carga de trabajos:** En este estado se realiza la verificación de los pagos de los trabajos para finalizar su proceso de postulación. Al mismo tiempo, se puede ver información del mismo, y revisar si este posee asignado árbitros.
- **Estado En asignación de árbitros a trabajos:** En este estado del arbitraje puede hacer las mismas funcionalidades del coordinador de área y realizar procesos administrativos del sistema de arbitraje.
- **Estado En arbitraje:** En este estado se realiza el arbitraje de las publicaciones científicas por partes de los árbitros de subárea los cuales ven a detalle el trabajo y posteriormente emite un arbitraje a dicho artículo. Este proceso es el núcleo de la revisión por pares.

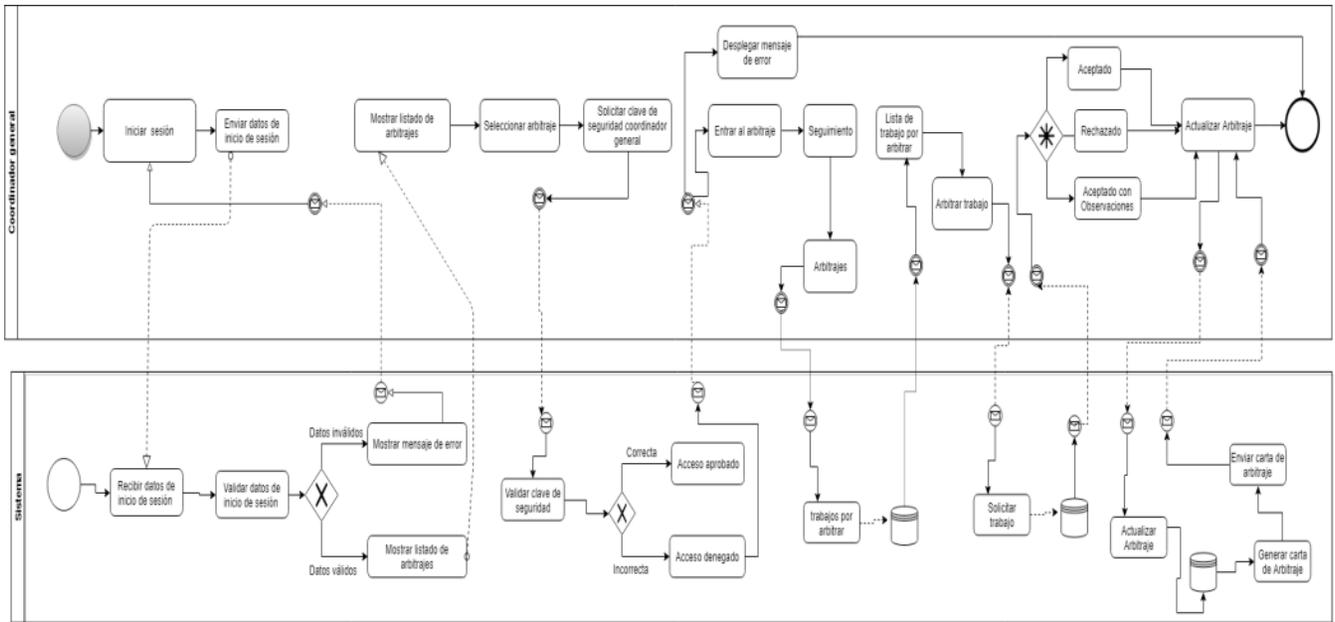


Figura N° 13. Flujo del coordinador general - estado en arbitraje – Arbitrar trabajo

Fuente: Elaboración propia

Adicionalmente, el coordinador general o coordinador de área tiene la decisión final sobre la aceptación o rechazo del arbitraje teniendo en cuenta el veredicto de los árbitros de subárea asignados, es aquí donde se le indica al autor cual fue el resultado de su arbitraje.

- **Estado En cierre de arbitraje:** En este estado es posible comenzar a ver los resultados del proceso, así como darle seguimiento al proceso de arbitraje, y finalizar el proceso del arbitraje para los coordinadores generales o coordinadores de área.

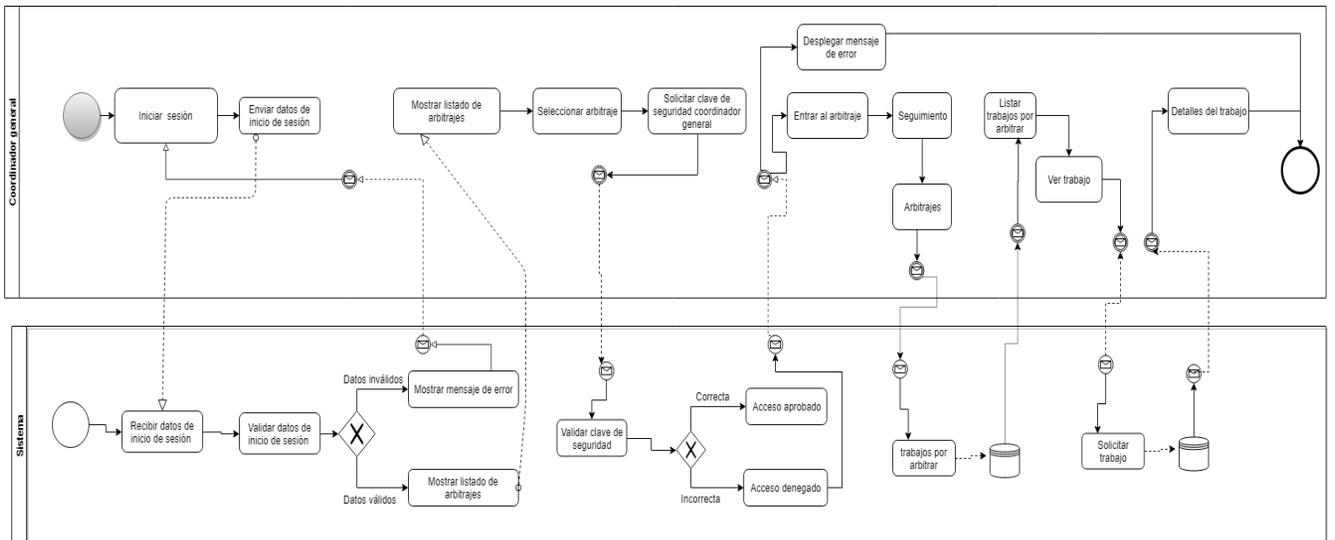


Figura N° 14. Flujo del coordinador general - estado en arbitraje – Mostrar arbitrajes

Fuente: Elaboración propia

- **Estado En asignación de sesiones:** En este estado se realiza la creación de las sesiones, la asignación de coordinador de sesión, el cambio de modalidad de presentación, entre otras funcionalidades relacionadas con las sesiones del arbitraje.
- **Estado En resumen:** En este estado es posible la generación de certificados, la visualización de los resultados del proceso de arbitraje y exportar dichos resultados.

4.5.2 Flujo de coordinador de área

- **Estado Desactivado:** En este estado no tiene acceso a la aplicación.
- **Estado Iniciado:** En este estado el coordinador de área tiene funciones muy puntuales las cuales van relacionadas con la administración de áreas y de árbitros de subárea.

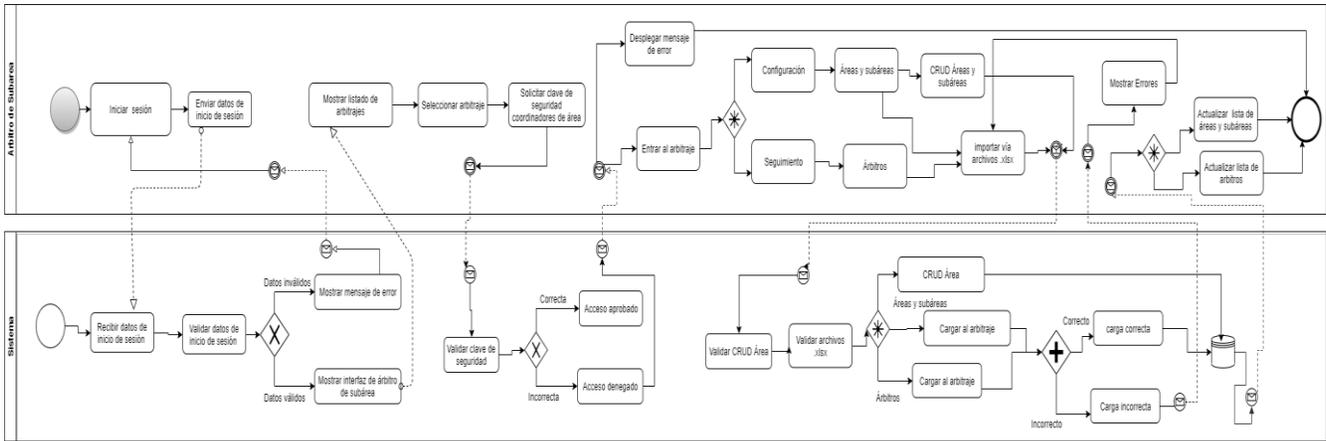


Figura N° 15. Flujo del coordinador de área - estado iniciado

Fuente: Elaboración propia

Cabe destacar que el coordinador general tiene estas funcionalidades análogamente, ya que es el quien realiza la asignación de los coordinadores de área.

- **Estado En selección y asignación de coordinadores de área:** En este estado la funcionalidad más relevante es la asignación de los árbitros de subárea por parte de los coordinadores de área. A continuación, se podrá ver el flujo en la Figura N°16.

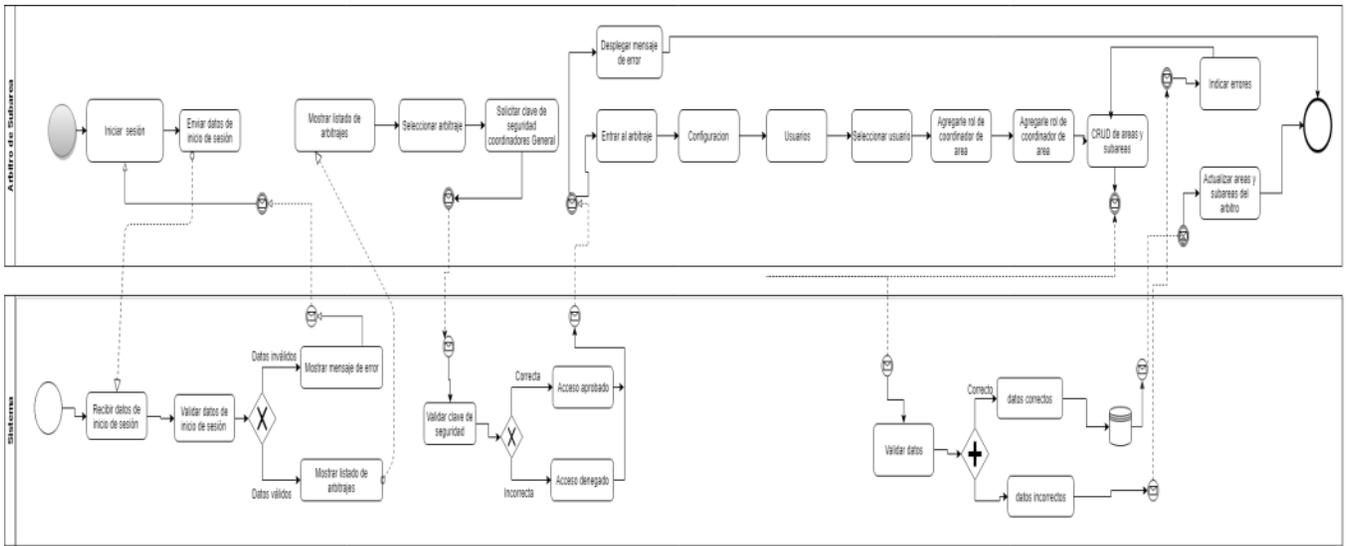


Figura N° 16. Flujo del coordinador de área - estado iniciado

Fuente: Elaboración propia

- **Estado En carga de trabajos:** En este estado se realiza la verificación de los pagos de los trabajos para finalizar su proceso de postulación. Al mismo tiempo, se puede ver información del mismo, y revisar si este posee asignado árbitros. A continuación, encontrara estos flujos en las siguientes figuras.

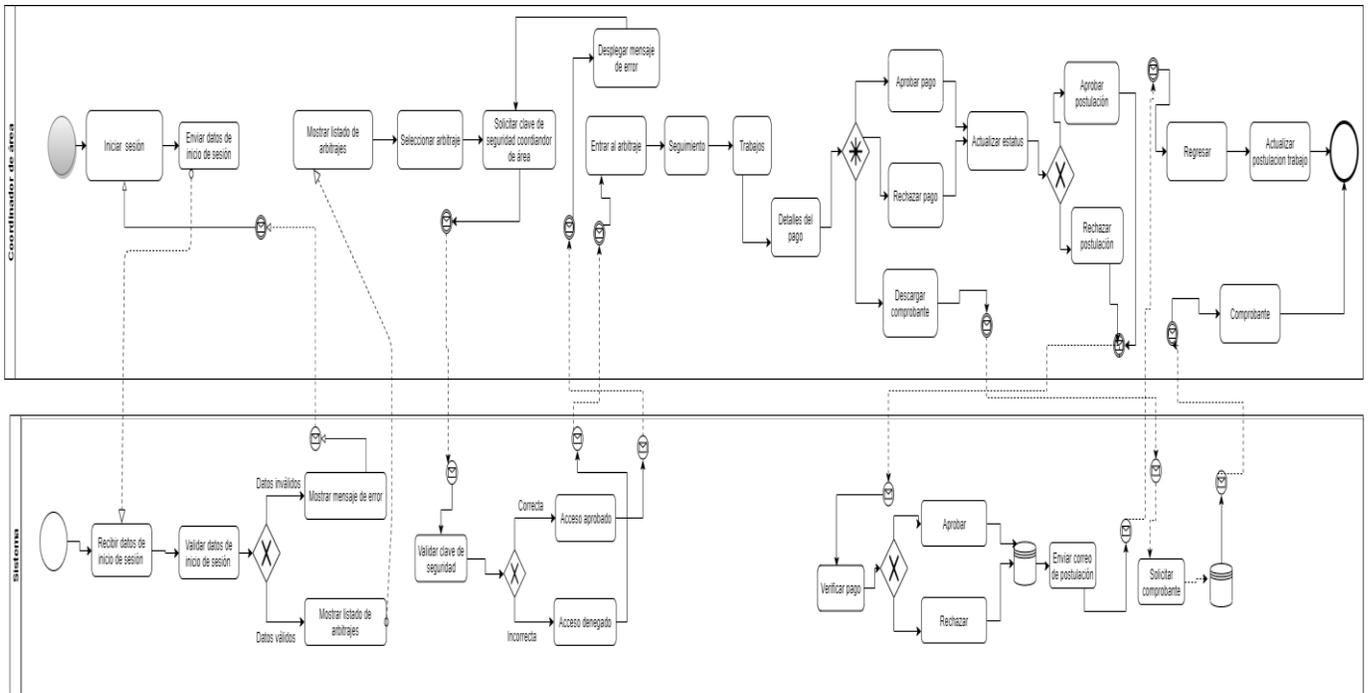


Figura N° 17. Flujo del coordinador de área - estado en carga de trabajos – verificar pago

Fuente: Elaboración propia

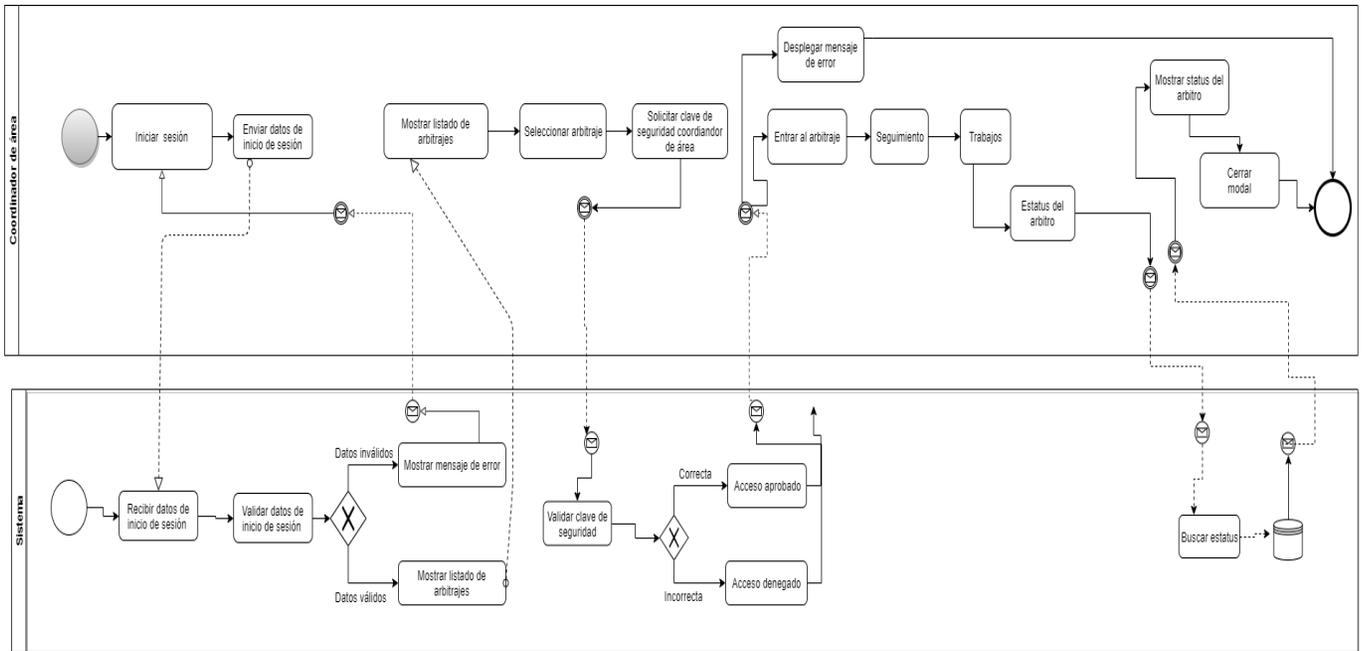


Figura N° 18. Flujo del coordinador de área - estado en carga de trabajos – status árbitro

Fuente: Elaboración propia

- **Estado En asignación de árbitros a trabajos:** En este estado del proceso se realiza las operaciones del estado anterior agregando la asignación de los árbitros a los trabajos. A continuación, en la siguiente figura se ilustra dicho flujo.

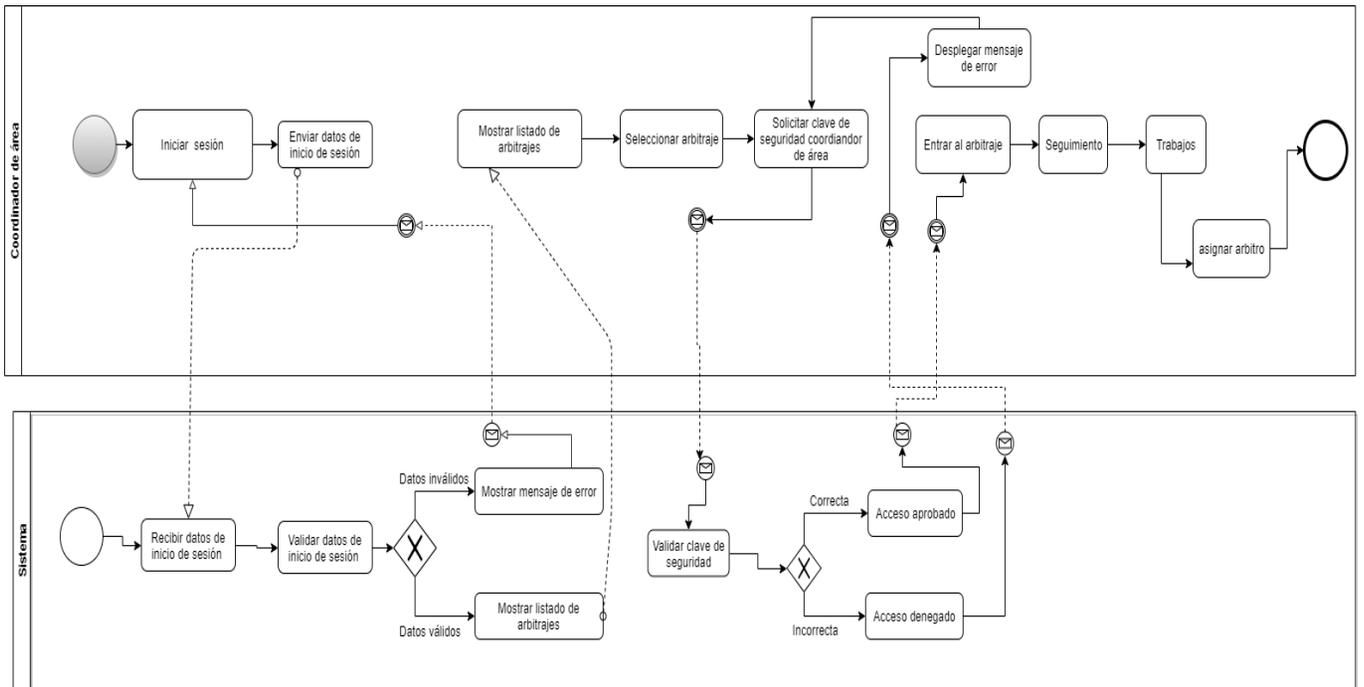


Figura N° 19. Flujo del coordinador de área - estado en carga de trabajos – Asignar árbitros

Fuente: Elaboración propia

- **Estado En arbitraje:** En este estado se realiza el arbitraje de las publicaciones científicas por parte de los árbitros de subárea los cuales ven a detalle el trabajo y posteriormente emite un arbitraje a dicho artículo. Este proceso es el núcleo de la revisión por pares.

Adicionalmente, el coordinador general o coordinador de área tiene la decisión final sobre la aceptación o rechazo del arbitraje teniendo en cuenta el veredicto de los árbitros de subárea asignados, es aquí donde se le indica al autor cual fue el resultado de su arbitraje.

- **Estado En cierre de arbitraje:** En este estado es posible comenzar a ver los resultados del proceso, así como darle seguimiento al proceso de arbitraje, y finalizar el proceso del arbitraje para los coordinadores generales o coordinadores de área.

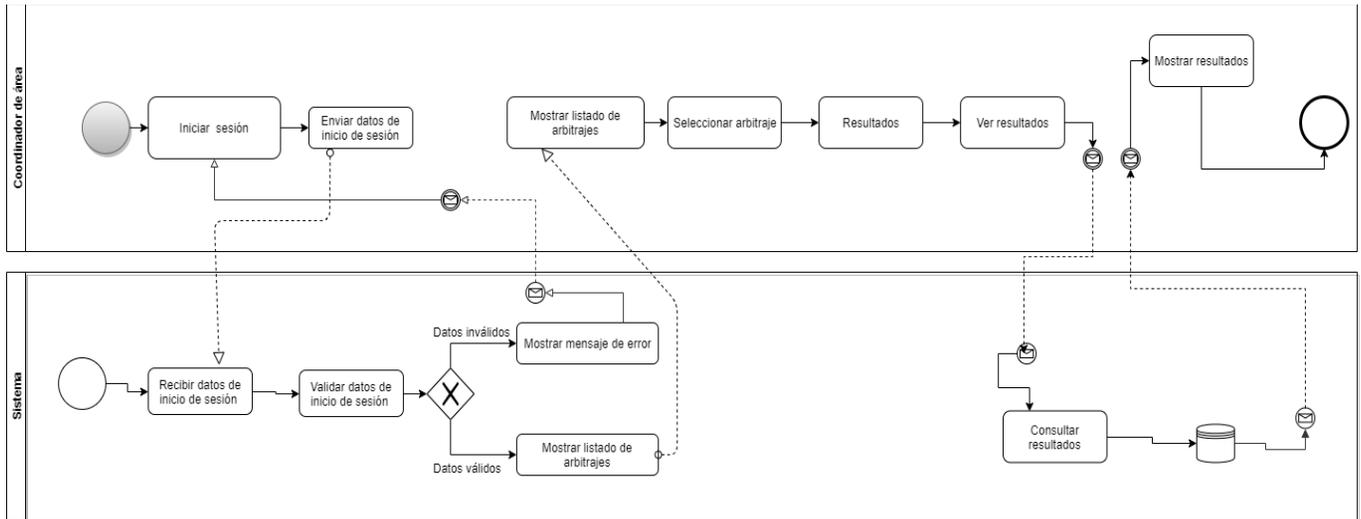


Figura N° 20. Flujo del coordinador de área - estado en cierre de arbitraje – mostrar resultados

Fuente: Elaboración propia

- **Estado En asignación de sesiones:** En este estado se realiza la creación de las sesiones, la asignación de coordinador de sesión, el cambio de modalidad de presentación, entre otras funcionalidades relacionadas con las sesiones del arbitraje.

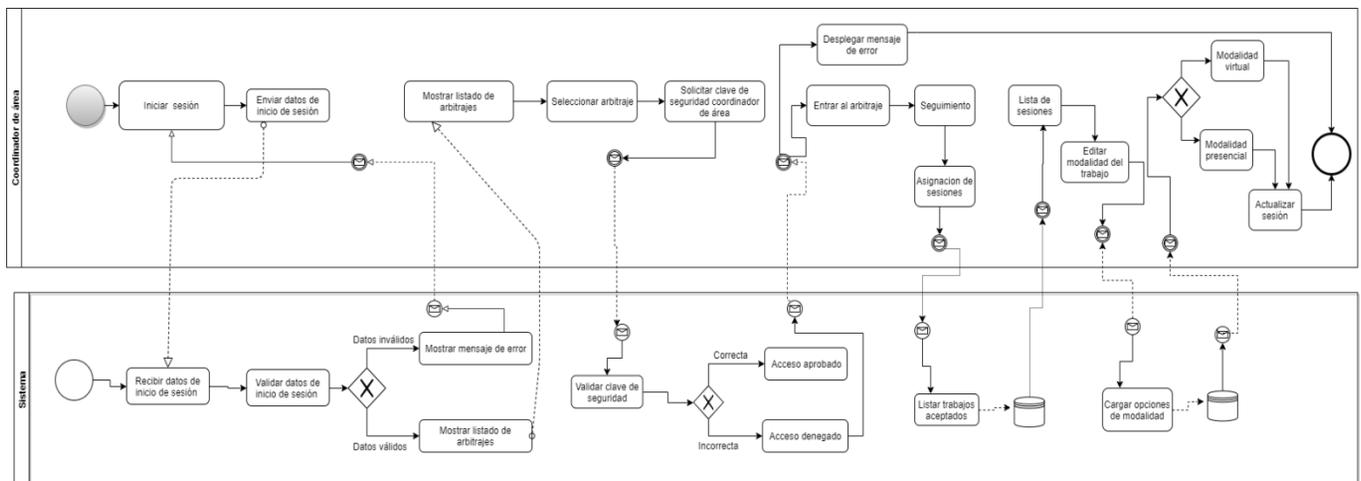


Figura N° 21. Flujo del coordinador de área - estado en asignación de sesión – cambiar modalidad de sesión

Fuente: Elaboración propia

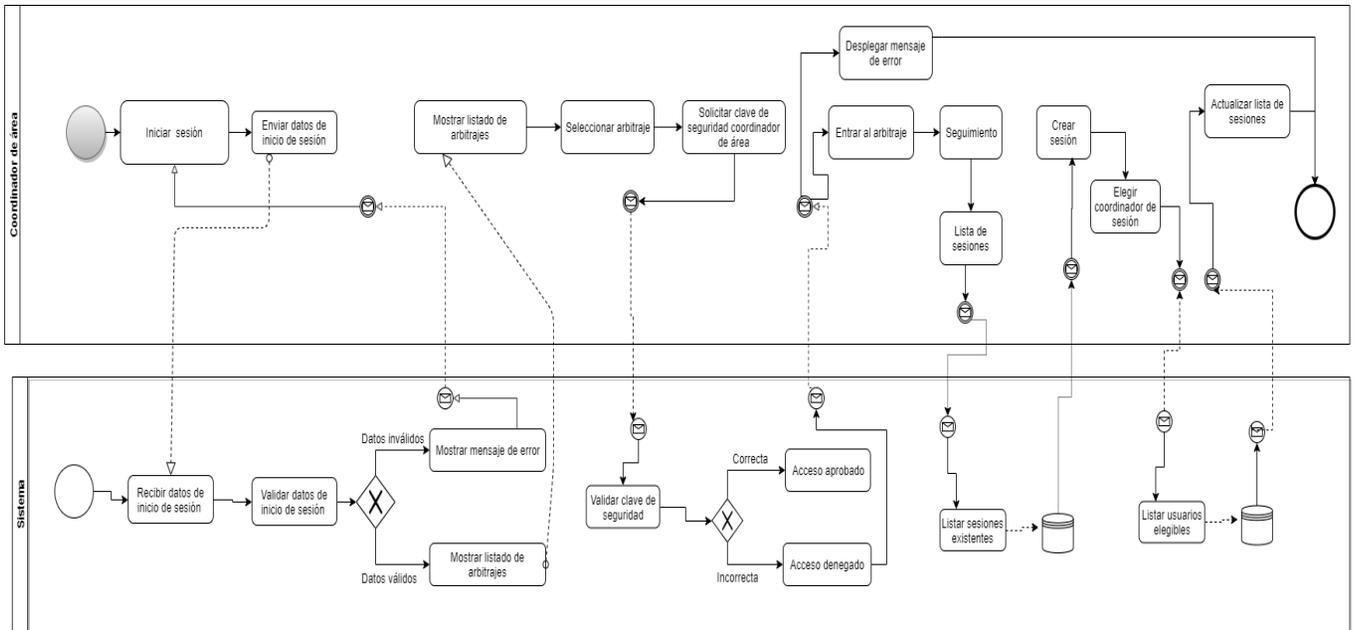


Figura N° 22. Flujo del coordinador de área - estado en asignación de sesión – crear sesión

Fuente: Elaboración propia

- **Estado En resumen:** En este estado se realiza la generación de certificados, la visualización de los resultados del proceso de arbitraje y exportar dichos resultados.

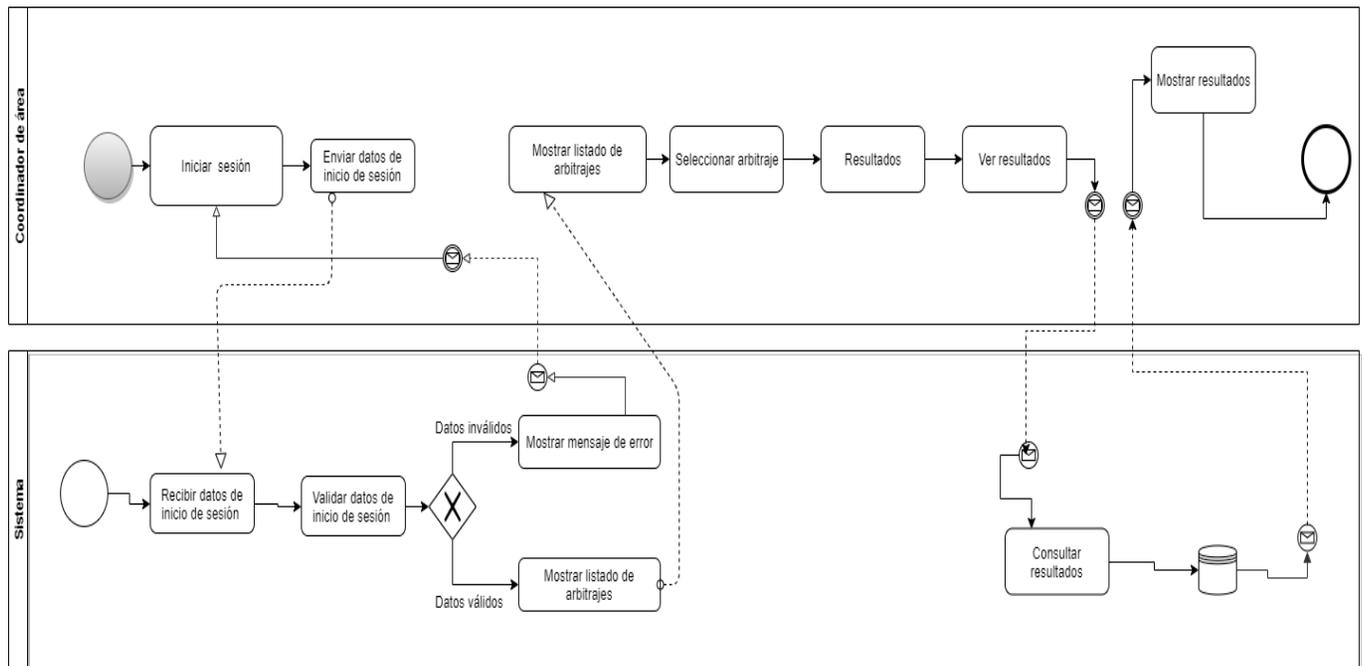


Figura N° 23. Flujo del coordinador de área - estado en resumen – mostrar resultados

Fuente: Elaboración propia

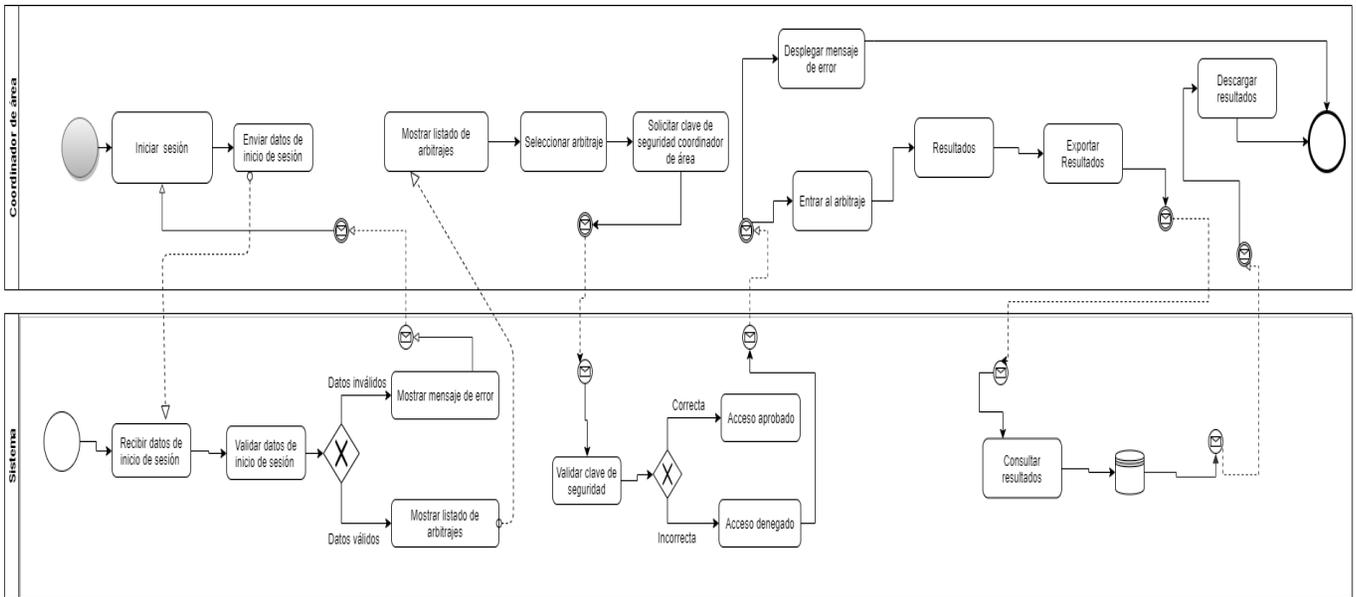


Figura N° 24. Flujo del coordinador de área - estado en resumen – exportar resultados

Fuente: Elaboración propia

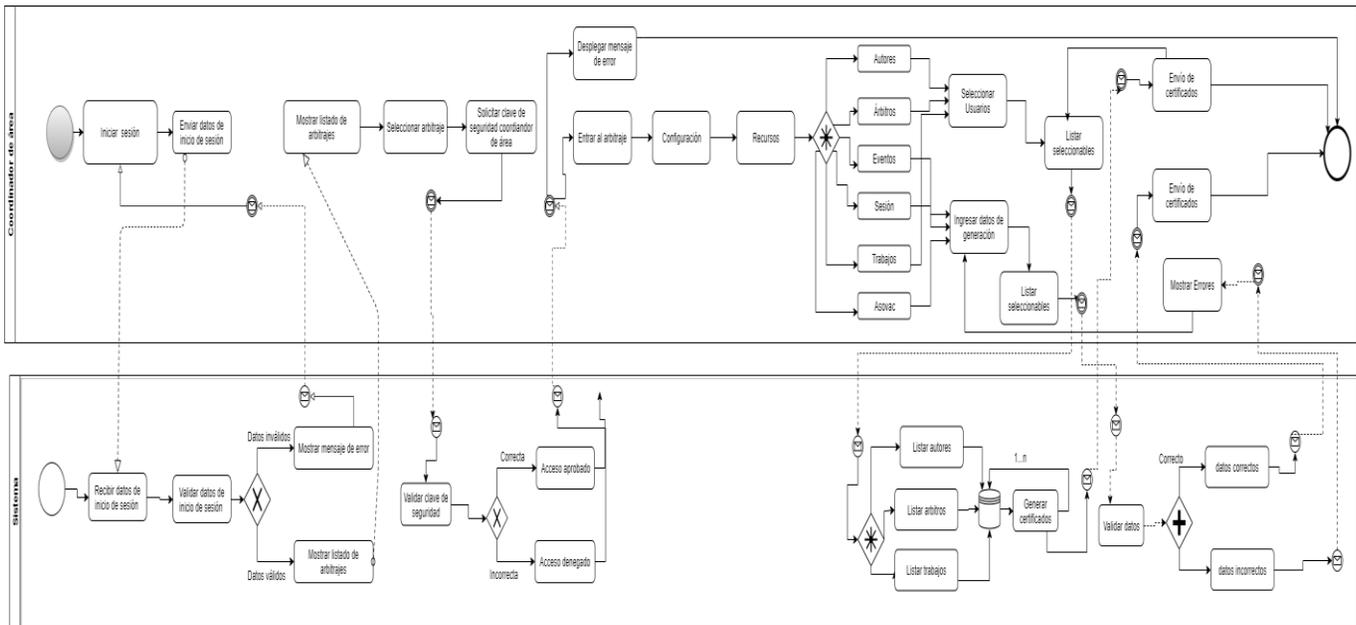


Figura N° 25. Flujo del coordinador de área - estado en resumen – generar certificados

Fuente: Elaboración propia

4.5.3 Flujo de árbitro de subárea

- **Estado Desactivado:** En este estado no tiene acceso a la aplicación.
- **Estado Iniciado:** En este estado no tiene acceso a la aplicación.
- **Estado En selección y asignación de coordinadores de área:** En este estado no tiene acceso a la aplicación.

- **Estado En carga de trabajos:** En este estado el árbitro realiza actividades menores, ya que su flujo iniciara en el siguiente estado. Sin embargo, este tiene acceso hacia información general de arbitraje.
- **Estado En asignación de árbitros a trabajos:** En este estado no tiene acceso a la aplicación. Sin embargo, cada árbitro debe aceptar la asignación para ser árbitro del trabajo al cual fue asignado.
- **Estado En arbitraje:** En este estado se realiza el arbitraje de las publicaciones científicas por partes de los árbitros de subárea los cuales ven a detalle el trabajo y posteriormente emite un arbitraje a dicho artículo. Este proceso es el núcleo de la revisión por pares.

Adicionalmente, el coordinador general o coordinador de área tiene la decisión final sobre la aceptación o rechazo del arbitraje teniendo en cuenta el veredicto de los árbitros de subárea asignados, es aquí donde se le indica al autor cual fue el resultado de su arbitraje.

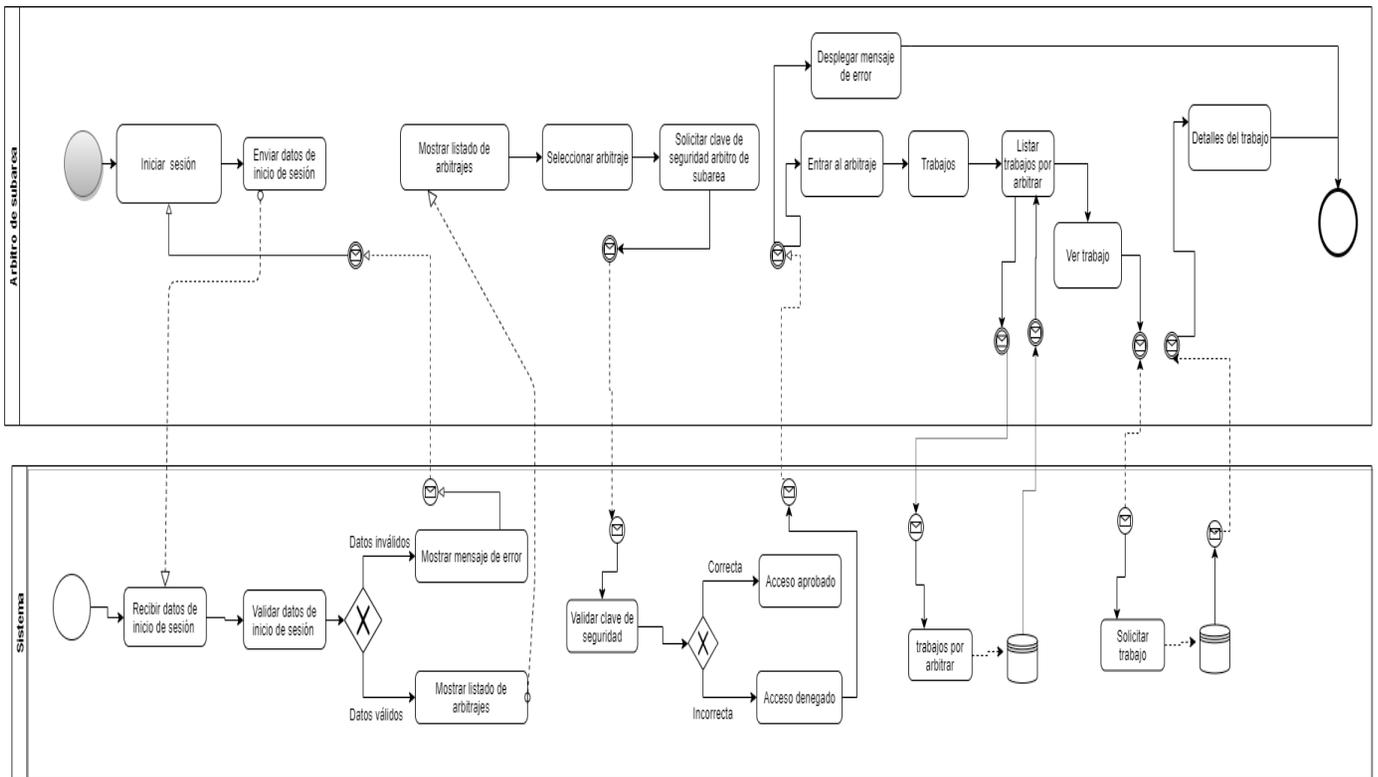


Figura N° 26. Flujo del árbitro de subárea - estado en arbitraje – ver trabajo por arbitrar

Fuente: Elaboración propia

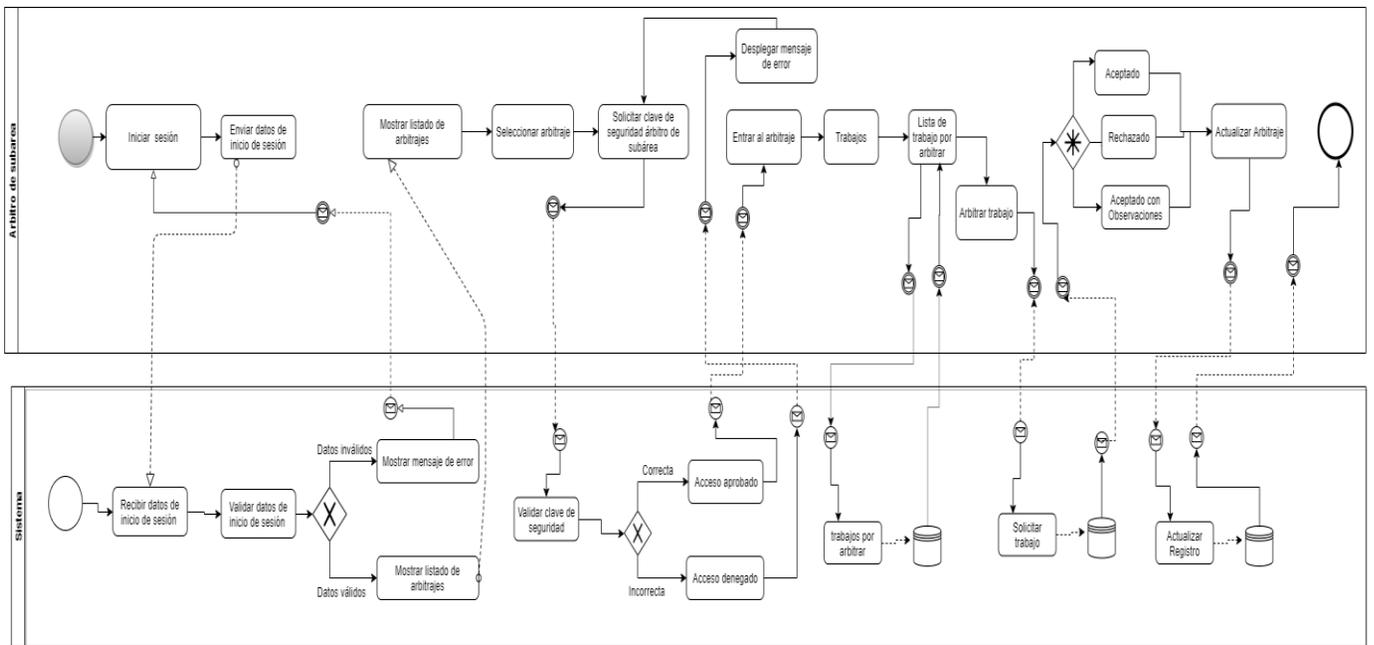


Figura N° 27. Flujo del árbitro de subárea - estado en arbitraje – arbitrar trabajo

Fuente: Elaboración propia

- **Estado En cierre de arbitraje:** En este estado es posible comenzar a ver los resultados del proceso, así como darle seguimiento al proceso de arbitraje.
- **Estado En asignación de sesiones:** En este estado no tiene acceso a la aplicación.
- **Estado En resumen:** En este estado será posible la generación de certificados como la visualización de los resultados del proceso de arbitraje.

4.5.4 Flujo de autor

Este rol tiene funcionalidades muy concretas, y es parte fundamental del proceso de la revisión por pares, ya que los trabajos que postula avanzan a través de todo este proceso de revisión. Los usuarios pueden registrarse, y esperar indicaciones del proceso de arbitraje para realizar la postulación de los trabajos.

- **Estado Desactivado:** En este estado no tiene acceso a la aplicación.
- **Estado Iniciado:** En este estado no tiene acceso a la aplicación.
- **Estado En selección y asignación de coordinadores de área:** En este estado no tiene acceso a la aplicación.
- **Estado En carga de trabajos:** En este estado el autor realiza la creación de su trabajo, y la postulación, así como otras funcionalidades orientadas a los trabajos. A continuación, se presenta el CRUD de los trabajos.

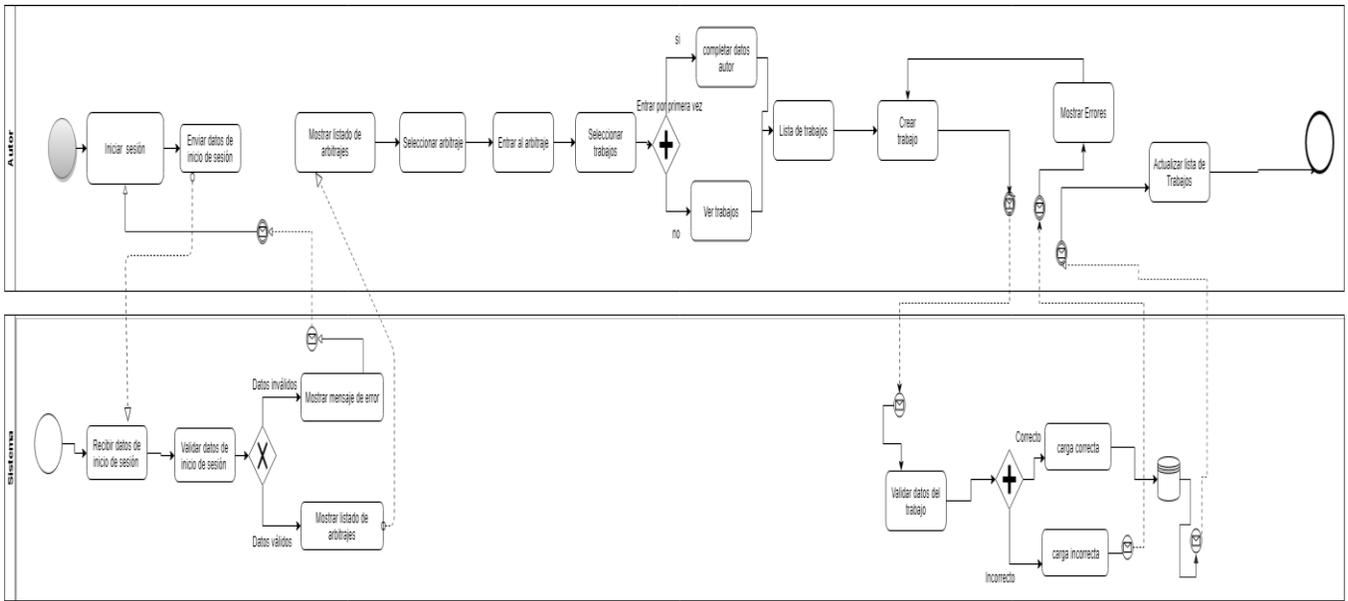


Figura N° 28. Flujo del autor - estado en carga de trabajos – crear un trabajo

Fuente: Elaboración propia

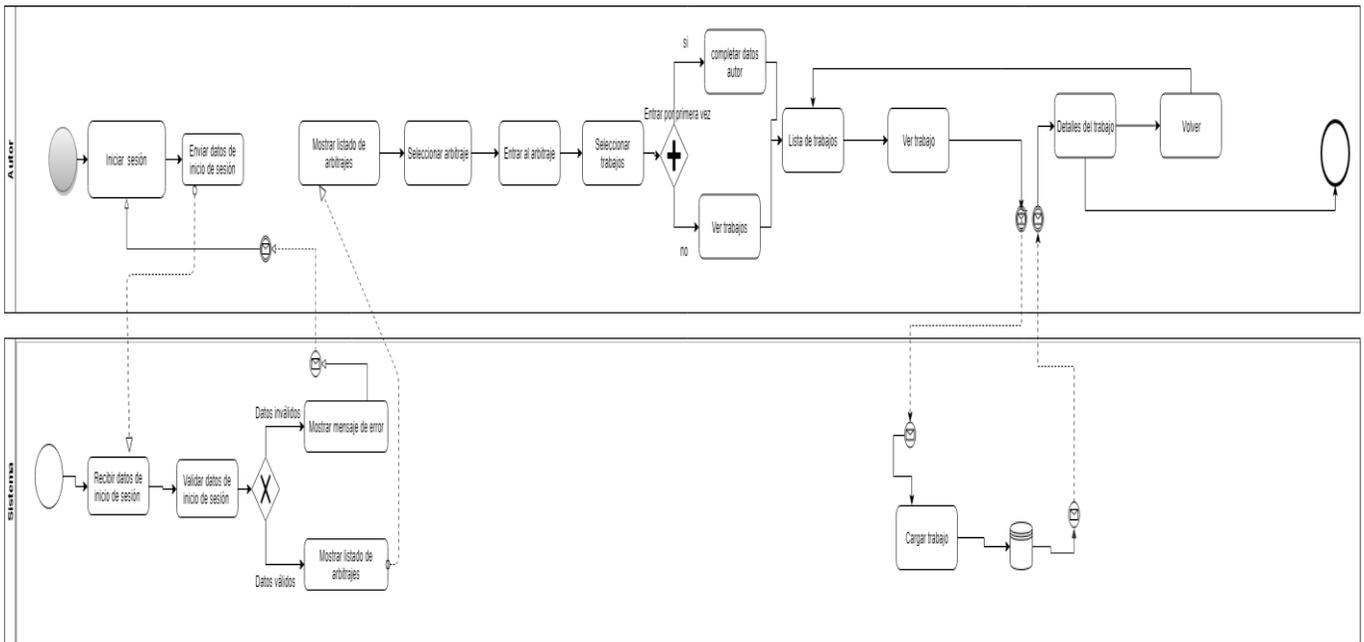


Figura N° 29. Flujo del autor - estado en carga de trabajos – ver trabajo

Fuente: Elaboración propia

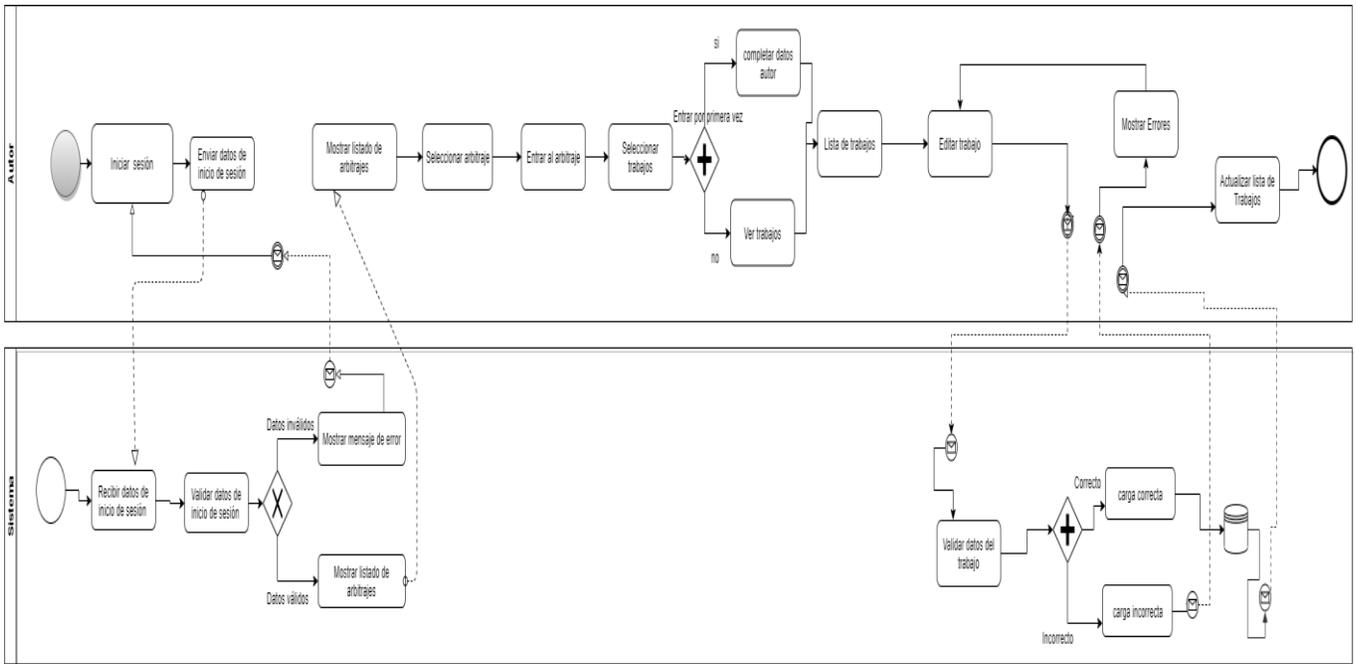


Figura N° 30. Flujo del autor - estado en carga de trabajos – editar trabajo

Fuente: Elaboración propia

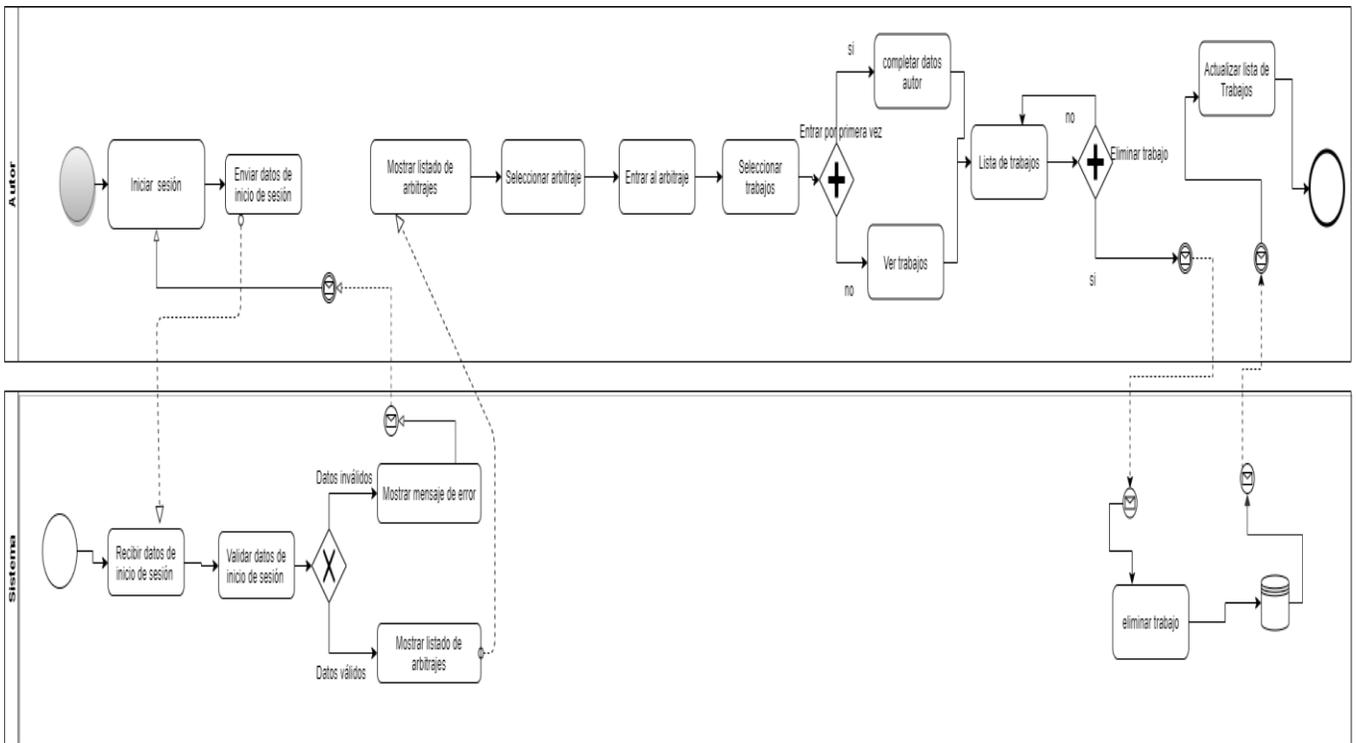


Figura N° 31. Flujo del autor - estado en carga de trabajos – eliminar trabajo

Fuente: Elaboración propia

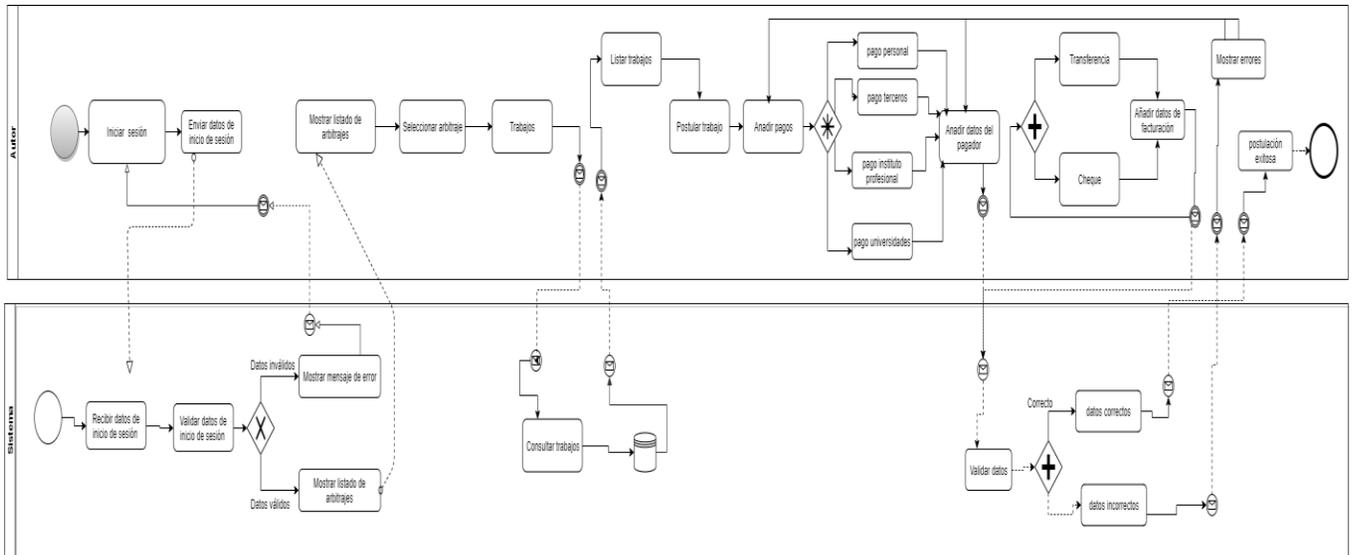


Figura N° 32. Flujo del autor - estado en carga de trabajos – Postular trabajo

Fuente: Elaboración propia

- **Estado En asignación de árbitros a trabajos:** En este estado no tiene acceso a la aplicación.
- **Estado En arbitraje:** En este estado no tiene acceso a la aplicación.
- **Estado En cierre de arbitraje:** En este estado no tiene acceso a la aplicación.
- **Estado En asignación de sesiones:** En este estado no tiene acceso a la aplicación.
- **Estado En resumen:** En este estado el autor podrá visualizar resultados generales y sus propios resultados

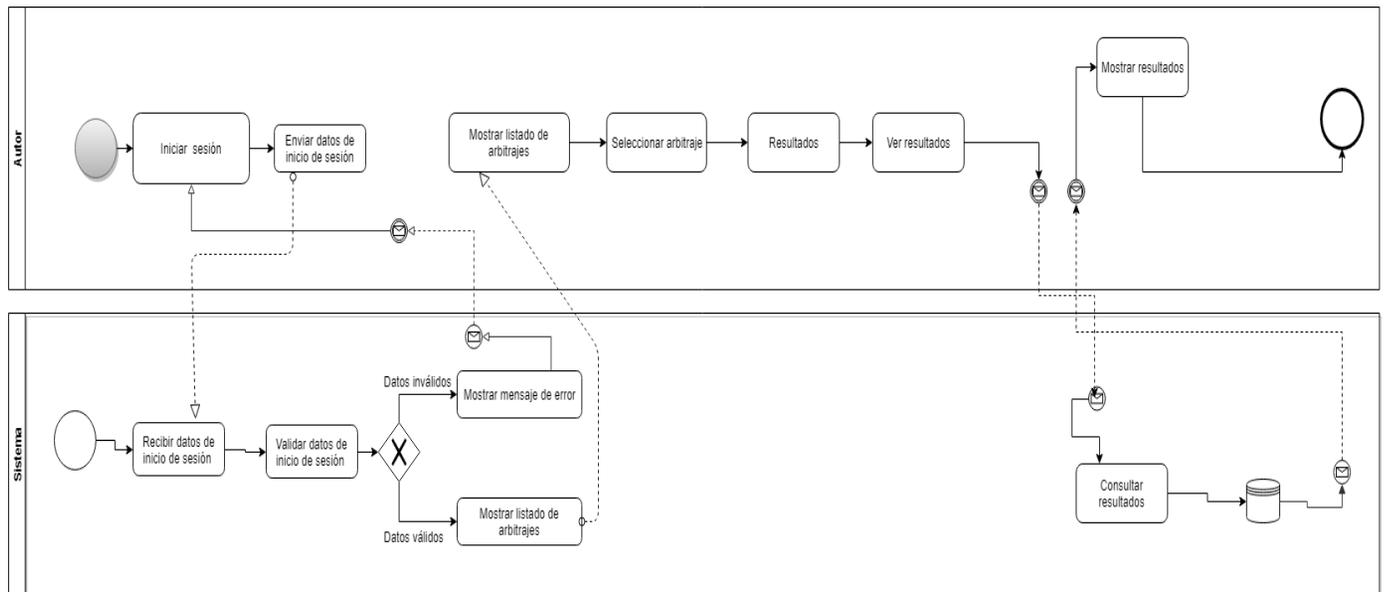


Figura N° 33. Flujo del autor - estado en resumen – ver mis resultados

Fuente: Elaboración propia

4.6 Análisis del modelado de datos y definición

Para el proceso de análisis y modelado de datos se examinaron, revisaron y analizaron siete archivos en formato Excel (.xlsx) de los cuales a través de la captación de requerimiento realizado con el personal de AsoVAC se desprendieron cuarenta y dos (42) tablas para la aplicación.

4.6.1 Listado de Tablas

A continuación, se presenta un listado con todas las tablas de la base de datos pertenecientes al sistema desarrollado, junto con una breve explicación para cada una

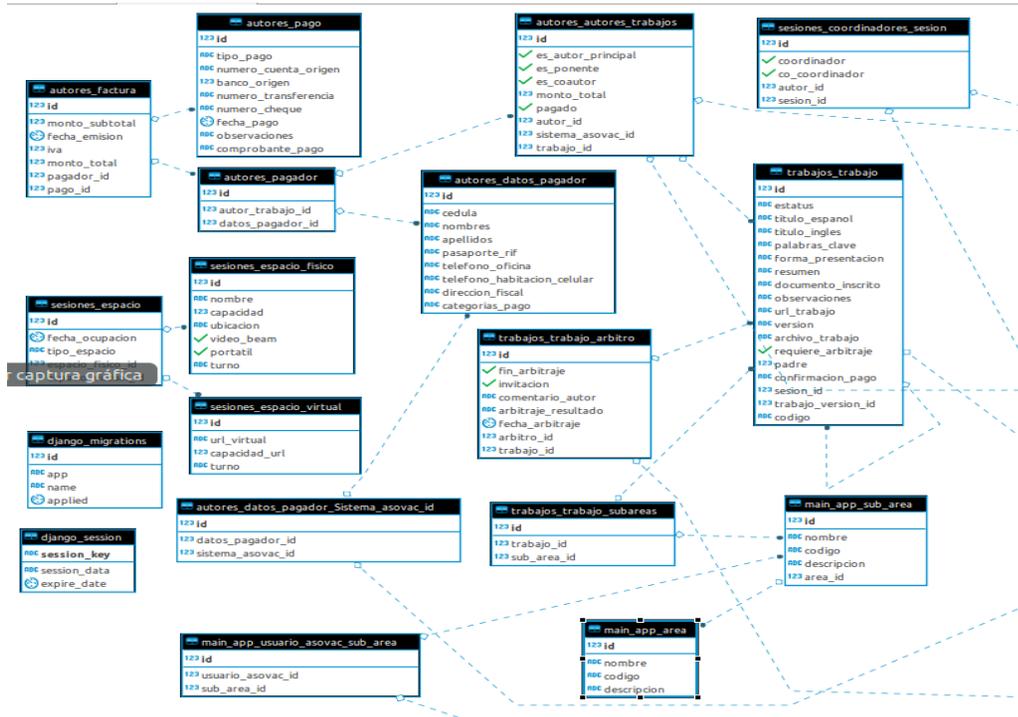


Figura N° 34 Modelo de Base de Datos, parte I

Fuente: Elaboración propia

- autores_autores_Trabajos: Permite llevar la relación entre el autor(res) y el trabajo(s) postulado(s) en el proceso de arbitraje.
- autores_datos_pagador: Permite almacenar y controlar la información referente a los datos de quien financia la postulación del trabajo.
- autores_datos_pagador_Sistema_asovac_id: Permite relacionar los datos del pagador para con el trabajo y proceso de arbitraje en el que participa el trabajo a ser evaluado.
- autores_factura: Permite relacionar las facturas a uno o más trabajos postulados para el proceso de arbitraje.
- autores_pagador: Permite almacenar la información de los financiadores de los trabajos.
- autores_pago: Permite guardar la información del pago de los trabajos postulados.
- main_app_sub_area: Permite almacenar todas las subáreas bajo un código de identificación.

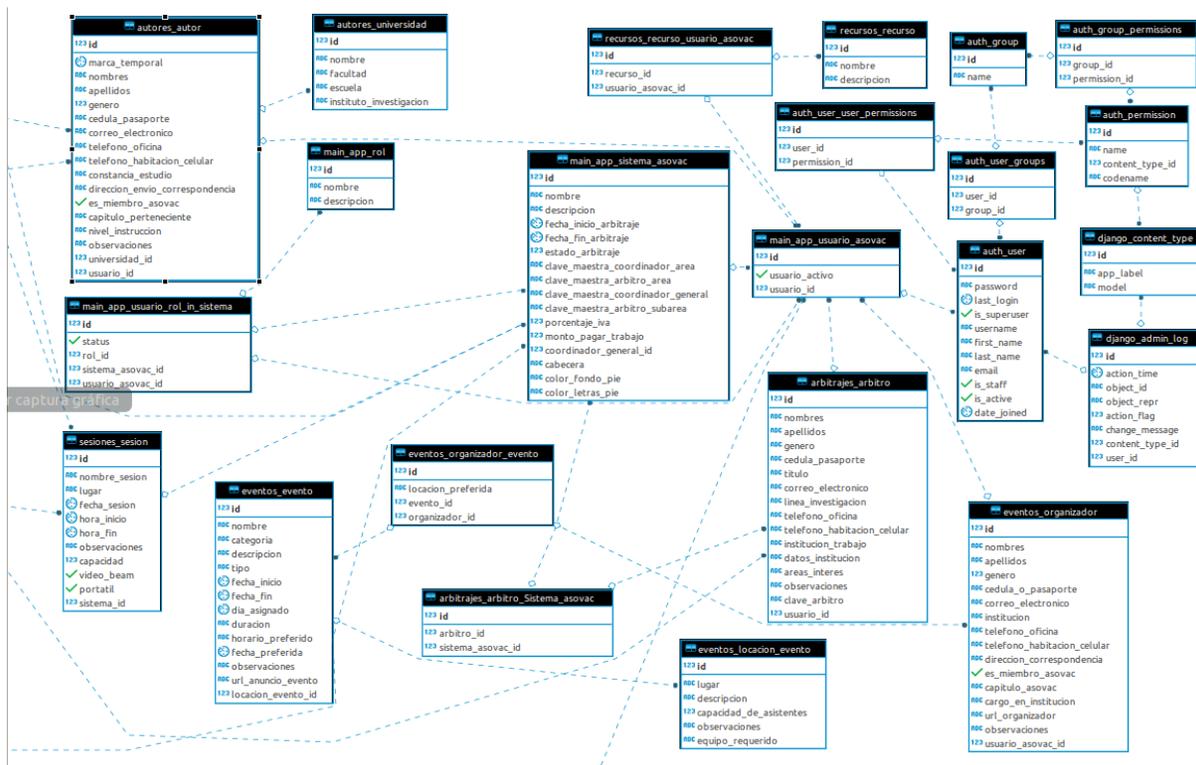


Figura N° 35. Modelo de Base de Datos parte II

Fuente: Elaboración propia

- main_app_area: Permite almacenar todas las áreas bajo un código de identificación.
- sesiones_coordinadores_sesion: Permite saber quién es el coordinador y el co-coordinador de cada sesión y los autores asignados a dicha sesión.
- sesiones_espacio: Permite almacenar la información del espacio dispuesto para el proceso de arbitraje.
- sesiones_espacio_fisico: Permite almacenar la información de espacio físico dispuesto para las distintas ponencias en formato cartel u oral sobre los trabajos arbitrados.
- sesiones_espacio_virtual: Permite almacenar la información de espacio virtual dispuesto para los trabajos arbitrados.
- trabajos_trabajo: Permite almacenar la información de los trabajos registrados en el proceso de arbitraje.
- trabajos_trabajos_subarea: Permite saber a cuales áreas están relacionados los trabajos de investigación.
- trabajos_trabajos_arbitro: Permite Almacenar el status del proceso de arbitraje de cada trabajo y se incorpora para el proceso de la asignación de sesiones.
- django_migrations: Permite almacenar y llevar un control de las migraciones aplicadas para cambiar la base de datos a lo largo del proceso de desarrollo.
- django_sessions: Permite almacenar todos los valores usados en la sesión de los usuarios dentro de la aplicación.
- main_app_usuario_asovac_sub_area: Permite asociar el usuario a las subáreas a las que pertenece.

- autores_universidad: Permite almacenar la información sobre la casa de estudio de los autores.
- eventos_evento: Permite almacenar toda la información referente a los distintos eventos.
- eventos_locacion_evento: Permite almacenar y relacionar donde se llevará a cabo el evento.
- eventos_organizador: Permite almacenar los datos del organizador del evento.
- eventos_organizador_eventos: Permite relacionar, y llevar el control sobre que organizador(es) está encargado coordinar cierto(s) evento(s).
- main_app_rol: Permite almacenar y controlar los roles de usuarios de la plataforma (administrador, coordinador general, coordinador de área, árbitro de subárea y autor).
- main_app_Sistema_asovac: Permite guardar la información de cada proceso de arbitraje.
- main_app_usuario_asovac: Permite almacenar la información de todos los usuarios registrados en el sistema.
- main_app_usuario_asovac_rol: Permite guardar el rol y los usuarios que pertenecen a cada proceso de arbitraje.
- recursos_recurso: Permite almacenar toda la información de los certificados digitales.
- arbitrajes_arbitro: Permite almacenar los datos de cada árbitro.
- arbitrajes_arbitro_Sistema_asovac: Permite relacionar que árbitros participan en cada proceso de arbitraje.
- autores_autor: Permite almacenar toda la información de los autores.
- recursos_recurso_usuario_asovac: Permite conocer y relacionar que recursos tiene o puede solicitar un usuario del proceso de arbitraje.
- sesiones_sesion: Permite guardar la información de las sesiones de cada proceso de arbitraje.
- main_app_usuario_asovac_Sistema_asovac_id: Permite relacionar a los usuarios que participan en cada proceso de arbitraje.
- auth_user_permissions: Permite manejar la autenticación y autorización sobre usuarios y grupos.
- auth_group: Permite asociar cuales usuarios pertenecen a cuales grupos de roles.
- auth_user_groups: Permite almacenar y administrar los grupos de usuarios según su acceso.
- auth_user: Permite verificar los datos de autenticación de los usuarios.
- auth_group_permissions: Permite almacenar y manejar que roles tienen cada grupo.
- auth_permission: Permite verificar los datos de permisos de los usuarios.
- django_content_type: Permite almacenar sobre los modelos existentes en la aplicación.
- django_admin_log: Permite almacenar todo el registro de cambio en el *admin* de Django.

4.6.2 Modelado de datos

El modelo de datos utilizando en la aplicación de arbitrajes para la Asociación Venezolana para el Avance de la Ciencia es el siguiente:

4.7 Descripción de las principales interfaces de la aplicación

El sistema web se encuentra dividido en varios módulos. Los cuales están representados en la siguiente figura.

- **Módulo de Administración:** Se maneja toda la configuración del sitio para el administrador, desde configuración básica hasta eliminación, activación y archivado de procesos anteriores.
- **Módulo de Autores:** Se almacena y manipula toda la información de los postulantes de los trabajos.
- **Módulo de Sesiones:** Se utiliza para controlar la asignación de lugares físicos donde serán expuestos y/o defendidos los trabajos. Consiste en llevar el control de todas las actividades físicas o virtuales según sea el caso.
- **Módulo de Trabajo/ Artículos científicos:** Se almacena, controla y manipula toda la información referente a los trabajos científicos.
- **Módulo de Seguimiento:** Facilita manejar los procesos de revisión por pares del arbitraje involucrados en los estados de asignación de trabajos a áreas, arbitraje y cierre de arbitraje.
- **Módulo de Arbitraje:** Permite llevar el control de todos los estados del proceso de arbitraje, accesos flujos de todos los procesos más importantes de la aplicación. Manejo, control y generación de resultados parciales y totales. Incluso administrar las memorias de AsoVAC.
- **Módulo de Recursos:** Se encarga de gestionar todo lo referente a materiales sobre la convención como los certificados de asistencias, de postulantes, etc.

4.7.1 Interfaces de la aplicación

- **Interfaz de inicio de sesión:** La figura a continuación muestra el formulario de inicio de sesión para aquellos usuarios registrados en el sistema.

Figura N° 36. Interfaz para ingresar al sistema

Fuente: Elaboración propia

A partir de esta interfaz los usuarios registrados en el sistema podrán ingresar con su usuario y contraseña, en caso de no estar registrados se presenta la opción para registrarse, además se muestra la opción de recuperar contraseña en caso de que el usuario la haya olvidado.

- **Interfaz de recuperar contraseña:** El formulario de recuperación de contraseña para aquellos usuarios registrados en el sistema. A partir de esta interfaz los usuarios registrados en el sistema podrán recuperar su contraseña mediante el formulario utilizando su correo electrónico. Se generará una contraseña aleatoria la cual será enviada a su correo para acceder al sistema, y posteriormente cambiar su contraseña.
- **Interfaz registro de usuarios:** La figura a continuación muestra el formulario de registro de usuarios para aquellos usuarios no registrados en el sistema.

Figura N° 37. Interfaz de registro

Fuente: Elaboración propia

En esta interfaz los usuarios pueden registrarse ingresando su información personal. Por defecto tendrán el rol de autor.

- **Interfaz de aplicaciones:** La figura a continuación podrá elegir entre acceder hacia el sistema de arbitraje de AsoVAC o hacia la aplicación de eventos AsoVAC.

Figura N° 38. Interfaz de selección de aplicación

Fuente: Elaboración propia

En dicha interfaz podrá diferenciar dos funcionalidades centradas en dos procesos relacionados en la primera selección llevara a cabo procesos de arbitraje donde podrá revisar todo tipo de postulaciones científicas las cuales son evaluadas siguiendo el método de *peer review* mientras que en la segunda opción podrá ser capaz de agendar, organizar y ajustar eventos dispuesto para distintas actividades que AsoVAC requiera llevar a cabo.

- **Interfaz de accesos a los sistemas de arbitrajes:** En la figura que se muestra a continuación podrá elegir entre los sistemas de arbitrajes creados, hacia cual necesita acceder.

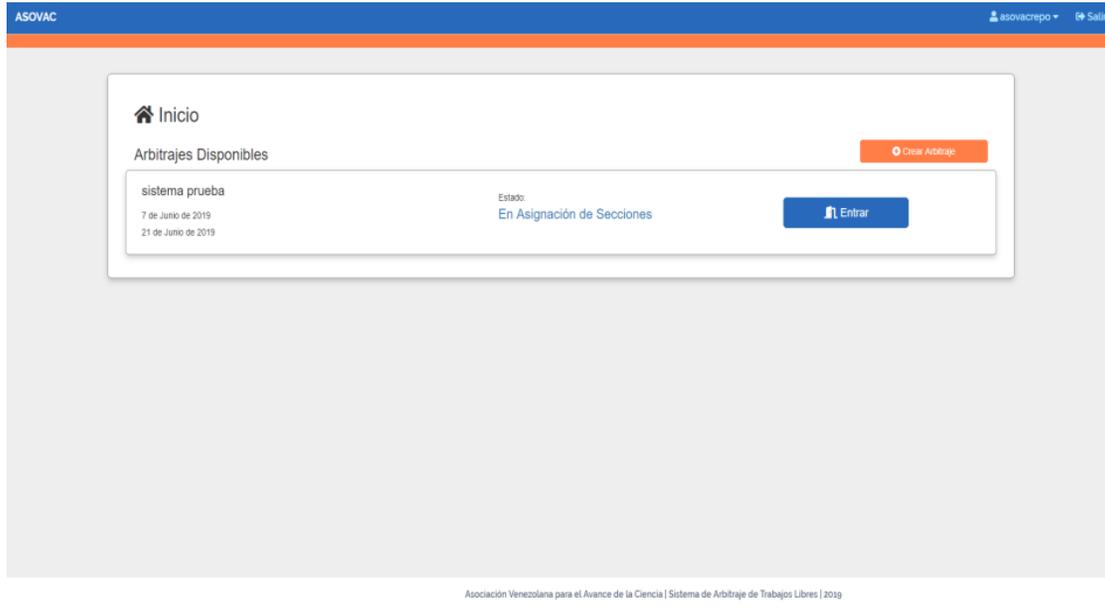


Figura N° 39. Interfaz de selección de arbitraje

Fuente: Elaboración propia

En esta interfaz podrá acceder a los sistemas de arbitrajes existentes donde podrá ver el detalle de cada uno dependiendo del rol que tenga en cada uno de ellos.

- **Interfaz de creación de un arbitraje:** En la figura que se muestra a continuación podrá crear un nuevo sistema de arbitraje.

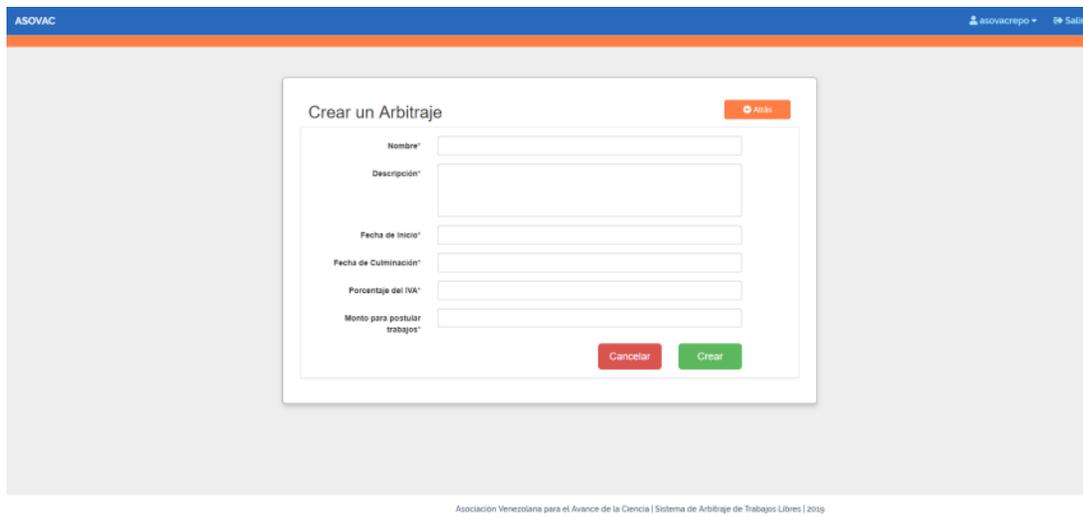


Figura N° 40. Interfaz para crear arbitrajes

Fuente: Elaboración propia

En dicha interfaz se puede crear un nuevo sistema de arbitraje donde se llevará a cabo el proceso de revisión por pares.

- **Interfaz de configuración General:** En la figura que se muestra a continuación podrá cambiar los datos básicos del sistema de arbitraje seleccionado.



Figura N° 41. Interfaz para configuración general parte 1

Fuente: Elaboración propia

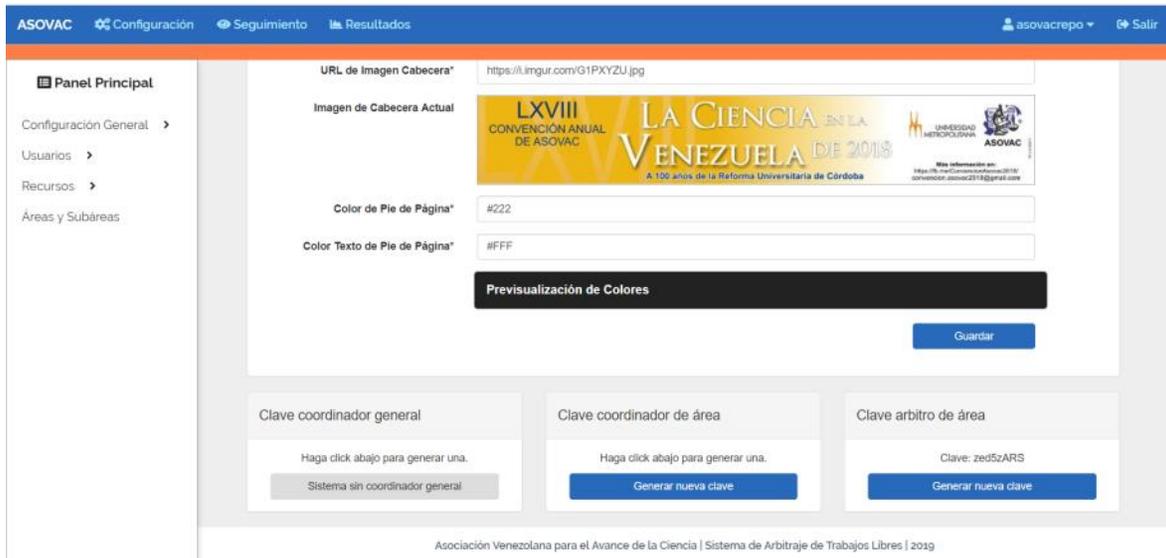


Figura N° 42. Interfaz para configuración general parte 2

Fuente: Elaboración propia

Así como también será capaz de cambiar la fecha de duración del sistema de arbitraje, cargar imagen de cabecera para la generación de correos, certificados para los participantes y la creación de las claves de seguridad para el coordinador general, coordinadores de área y árbitros.

- **Interfaz para cambiar el estado del sistema de arbitraje:** En la figura que se muestra a continuación se puede actualizar el estado del sistema del arbitraje seleccionado.

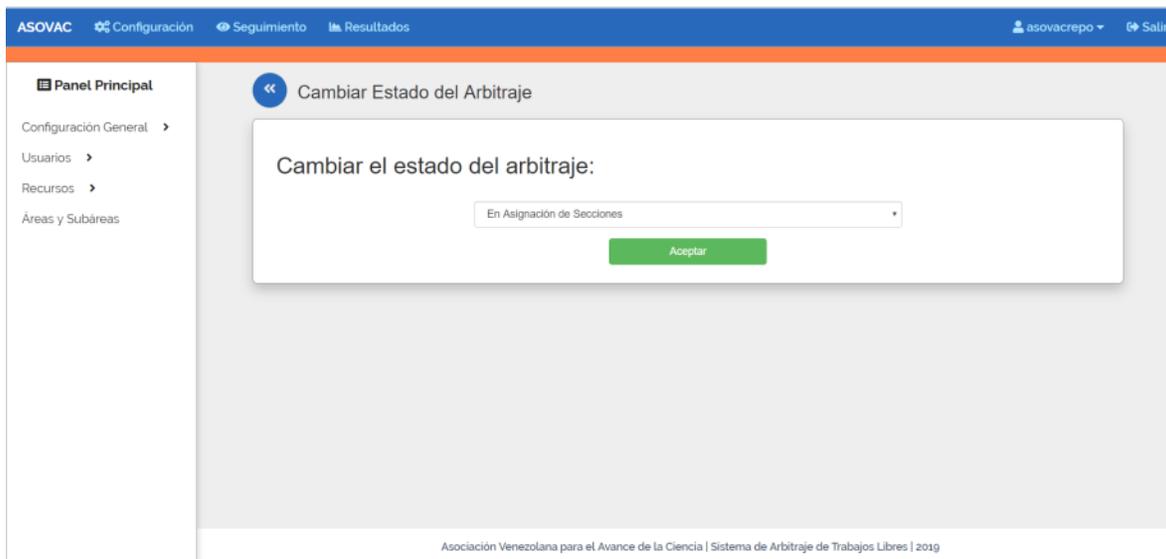


Figura N° 43. Interfaz para cambio de estado del arbitraje

Fuente: Elaboración propia

En dicha interfaz el coordinador general y el administrador son capaces de cambiar el estado del sistema de arbitraje ya que cada estado está destinado para actividades específicas.

- **Interfaz de lista de usuarios:** En la figura que se muestra a continuación se visualizan todos los usuarios existentes en la aplicación así como realizar las operaciones edición, ver detalles, y asignar rol al usuario.

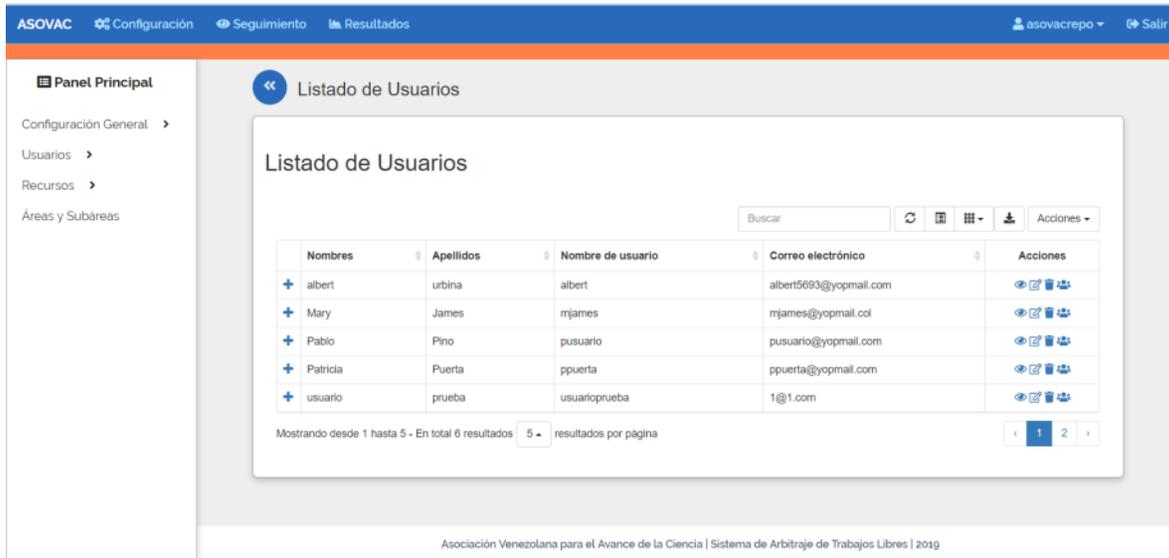


Figura N° 44. Interfaz para mostrar lista de usuarios

Fuente: Elaboración propia

- **Interfaz de detalles de Usuario:** La vista de detalle de usuario aparece luego de darle clic en el icono de visualizar, el cual permite ver los detalles de dicho usuario. Solo los coordinadores de área, el coordinador general y el administrador pueden entrar en esta vista.
- **Interfaz de editar usuario:** La figura a continuación aparece luego de darle clic en el icono de editar, el cual permite cambiar los datos del usuario en caso de ser requerido. Solo los coordinadores de área, el coordinador general y el administrador pueden realizar esta acción.

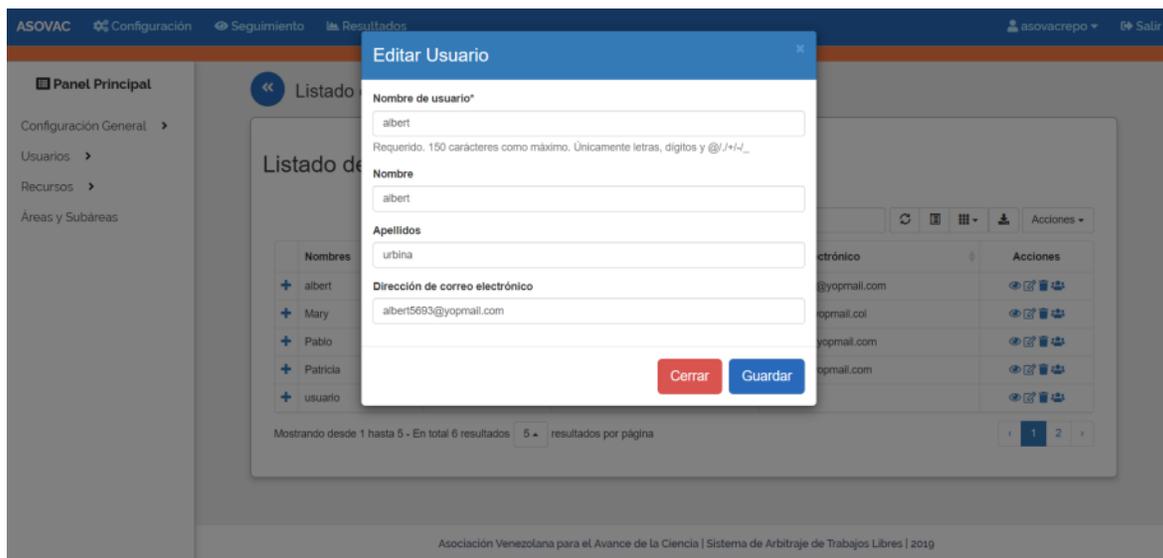


Figura N° 45. Interfaz para editar usuarios

Fuente: Elaboración propia

- **Interfaz de asignar rol para el usuario:** La vista de asignación de rol aparece luego de darle clic en el icono de los usuarios, el cual permite asignarle un rol al usuario. Dependiendo de dicha asignación el usuario tendrá accesos y permisos definidos para realizar ciertos procesos dentro de la aplicación del sistema de arbitraje. Solo los coordinadores de área, el coordinador general y el administrador pueden realizar esta acción.
- **Interfaz de carga usuarios:** la figura a continuación aparece luego de darle clic en el botón de acciones en la opción de carga de usuarios, después debe adjuntar un archivo de Excel con los usuarios que desea cargar. Solo los coordinadores de área, el coordinador general y el administrador pueden realizar esta acción.

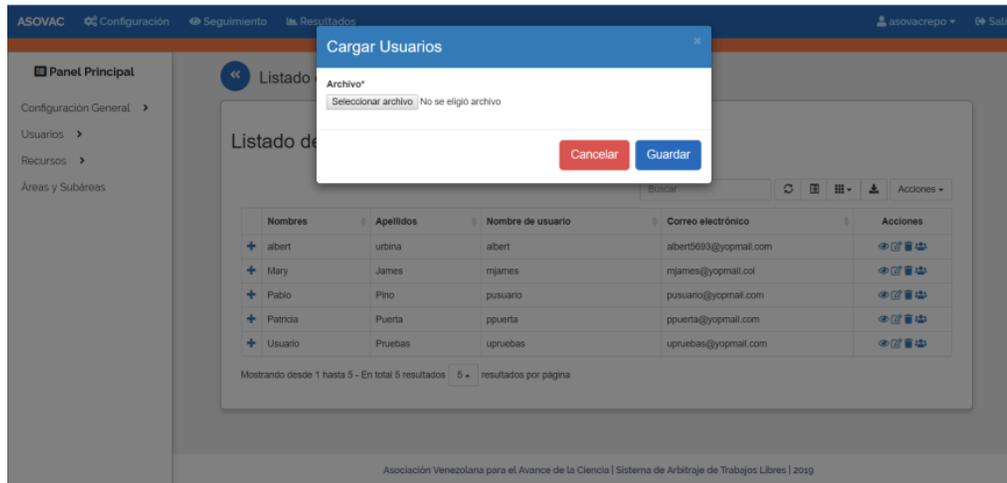


Figura N° 46. Interfaz de carga de usuarios

Fuente: Elaboración propia

- **Interfaz de asignación de coordinador general:** La figura a continuación se accede al darle clic en el menú lateral en la opción de configuración general y en selección de coordinador general. Esta funcionalidad solo es utilizada por el administrador.

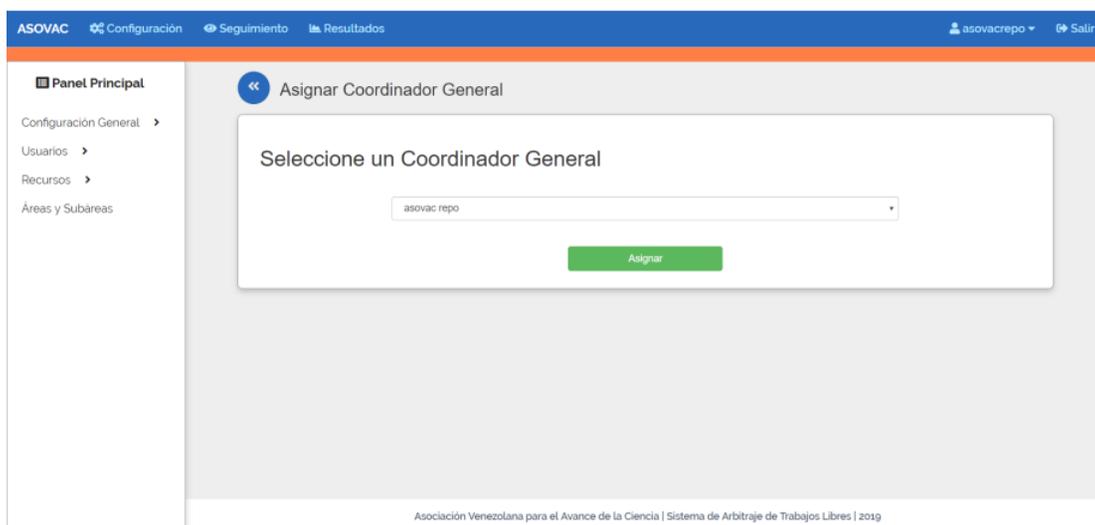


Figura N° 47. Interfaz para asignar coordinador general

Fuente: Elaboración propia

- **Interfaz de áreas y subáreas:** La figura a continuación se accede al darle clic en el menú lateral en la opción de áreas y subáreas, en la cual podrán observar todas las áreas y subáreas existentes. Así como también cargar nuevas áreas y realizar acciones sobre las mismas, estas operaciones son eliminar, editar, ver los detalles de las áreas y subáreas.

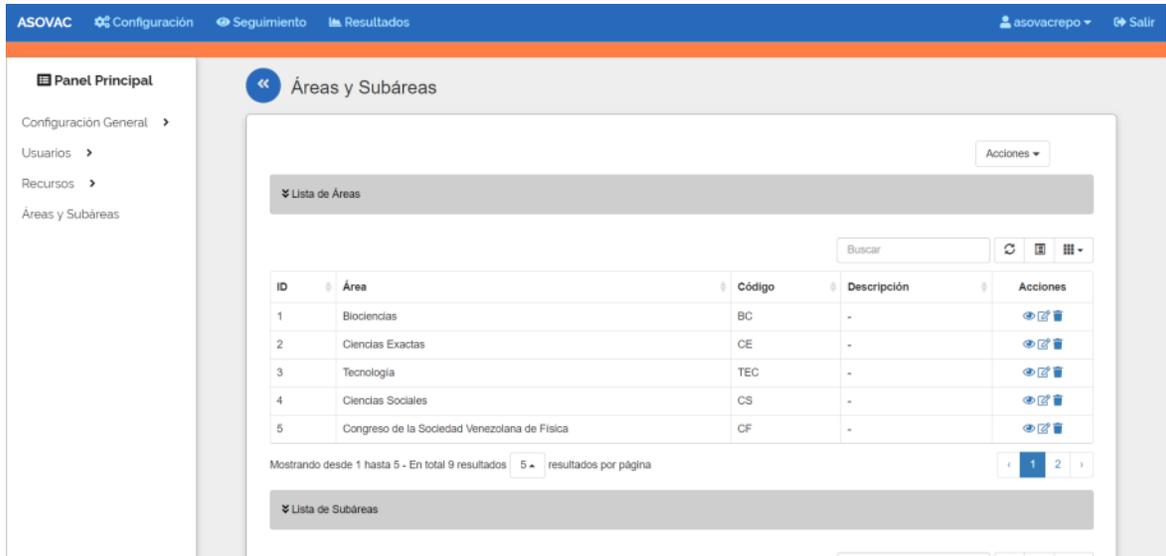


Figura N° 48. Interfaz de carga de áreas y subáreas parte 1

Fuente: Elaboración propia

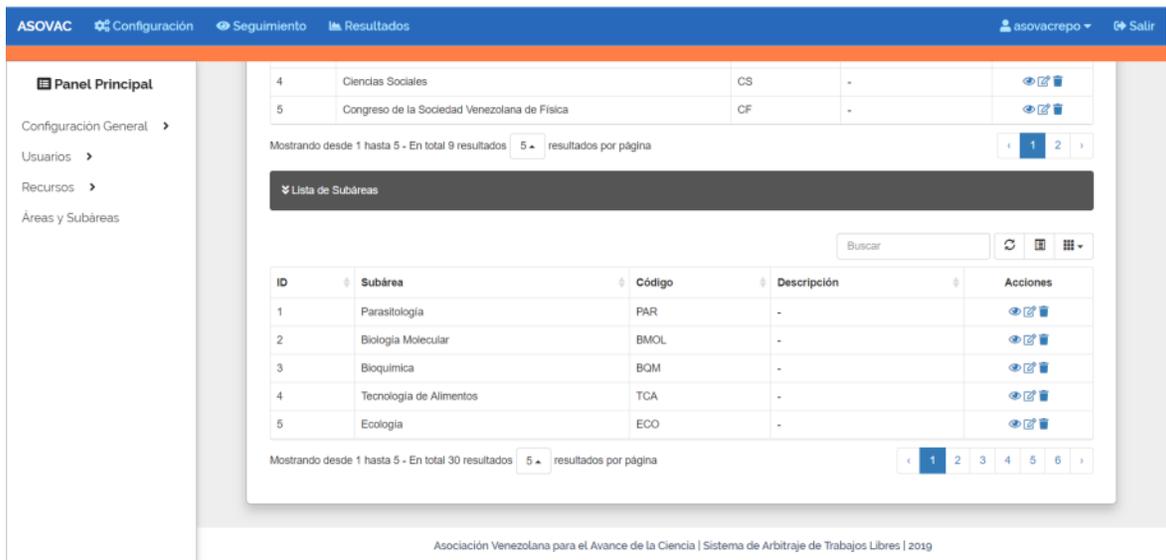


Figura N° 49. Interfaz de carga de áreas y subáreas parte 2

Fuente: Elaboración propia

- **Interfaz de cargar áreas:** La vista para cargas áreas surge al darle clic en el botón de acciones el cual despliega un modal donde debe adjuntarse un archivo en formato .xlsx para cargar nuevas áreas.
- **Interfaz de cargar subáreas:** La figura a continuación surge al darle clic en el botón de acciones el cual despliega un modal donde debe adjuntarse un archivo en formato .xlsx para cargar nuevas subáreas.

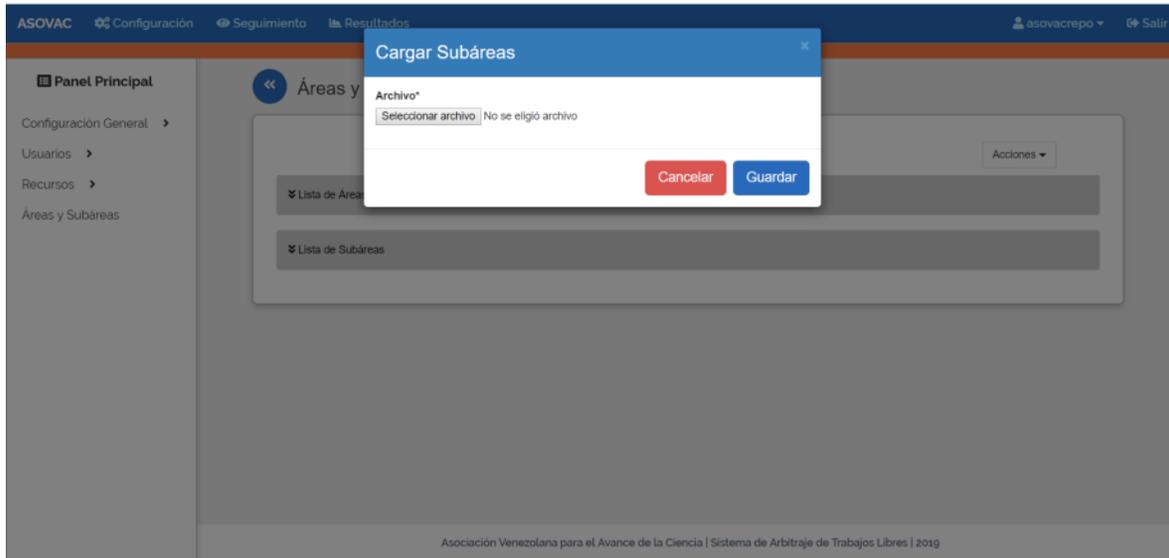


Figura N° 50. Interfaz de carga de subáreas

Fuente: Elaboración propia

- **Interfaz de lista de autores:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de seguimiento y posteriormente en autores. En dicha vista es posible administrar información sobre los autores y otras acciones. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.

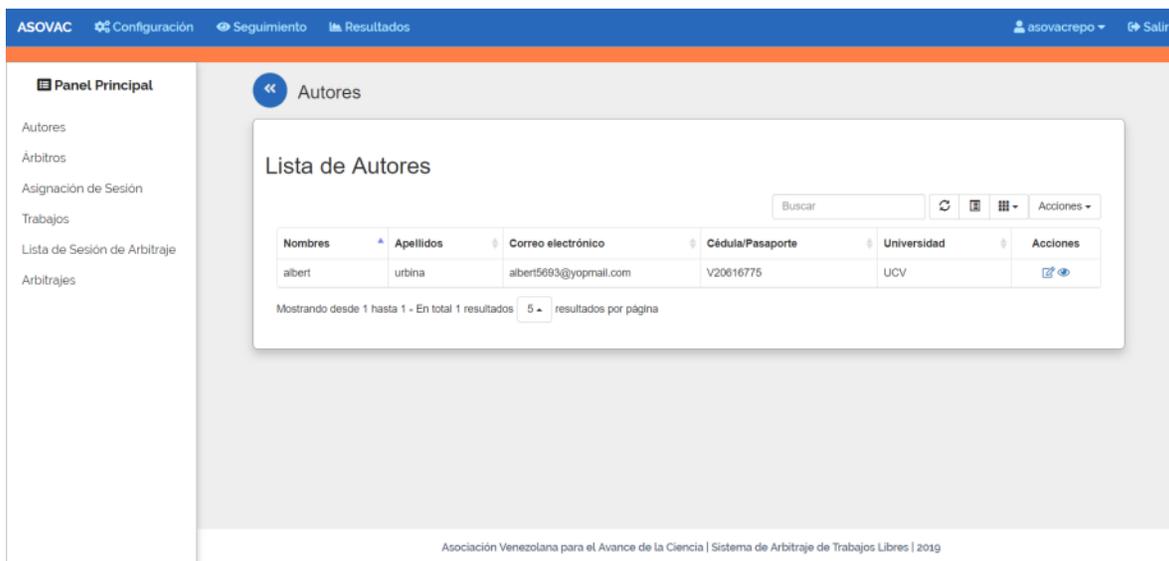


Figura N° 51. Interfaz para listar autores

Fuente: Elaboración propia

- **Interfaz de editar datos del autor:** La figura a continuación aparece al darle clic en el icono de editar dentro de la tabla. En dicha vista es posible actualizar información del autor en caso de ser necesario. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.

The screenshot shows the 'Editar autores' (Edit authors) interface. The top navigation bar includes 'ASOVAC', 'Configuración', 'Seguimiento', and 'Resultados'. The user is logged in as 'asovacrepo'. The left sidebar shows a 'Panel Principal' with options: 'Autores', 'Árbitros', 'Asignación de Sesión', 'Trabajos', 'Lista de Sesión de Arbitraje', and 'Arbitrajes'. The main content area is titled 'Editar autores' and contains a form for 'Datos del autor' (Author Data). The form fields are as follows:

Universidad*	UCV
Nombres*	albert
Apellidos*	urbina
Género*	Masculino
Cédula/Pasaporte*	V20616775
Correo Electrónico*	albert5693@yopmail.com
Teléfono de oficina*	04168966542
Teléfono de habitación/celular*	02124759863
Constancia estudio	Seleccionar archivo No se eligió archivo
Dirección de envío de correspondencia	Ejemplo: Sabana grande, CC El Recreo

Figura N° 52. Interfaz de edición de datos del autor parte 1

Fuente: Elaboración propia

The screenshot shows the bottom part of the 'Editar autores' interface. The form fields are as follows:

Cédula/Pasaporte*	V20616775
Correo Electrónico*	albert5693@yopmail.com
Teléfono de oficina*	04168966542
Teléfono de habitación/celular*	02124759863
Constancia estudio	Seleccionar archivo No se eligió archivo
Dirección de envío de correspondencia	Ejemplo: Sabana grande, CC El Recreo
Es miembro AsoVAC	<input type="checkbox"/>
Capítulo perteneciente	Ejemplo: Caracas
Nivel de instrucción*	Educación Técnico/Profesional
Observaciones	Introduzca sus observaciones aquí

At the bottom right of the form, there are two buttons: 'Cancelar' (red) and 'Editar autor' (green). The footer of the page reads: 'Asociación Venezolana para el Avance de la Ciencia | Sistema de Arbitraje de Trabajos Libres | 2019'.

Figura N° 53. Interfaz de edición de datos del autor parte 2

Fuente: Elaboración propia

- **Interfaz de ver detalles del autor:** Los detalles del autor aparecen al darle clic en el icono de ver detalles dentro de la tabla. En dicha vista es posible visualizar información del autor. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.
- **Interfaz de cargar autores:** La figura a continuación aparece al darle clic en el botón de acciones y posterior a la opción cargar autores. En dicha vista es posible cargar los autores mediante un archivo .xlsx con un formato definido. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.

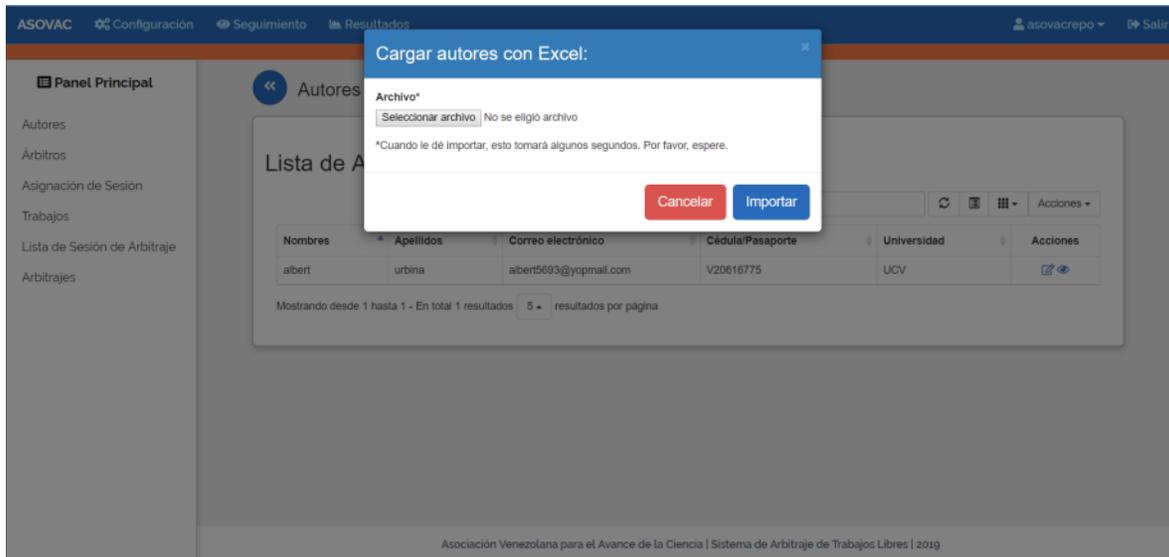


Figura N° 54. Interfaz para cargar autores

Fuente: Elaboración propia

- **Interfaz de lista de árbitros:** La figura a continuación se muestra al darle clic en el menú de navegación en la opción de seguimiento y posteriormente en árbitros. En dicha vista es posible administrar información sobre los árbitros y otras acciones. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.

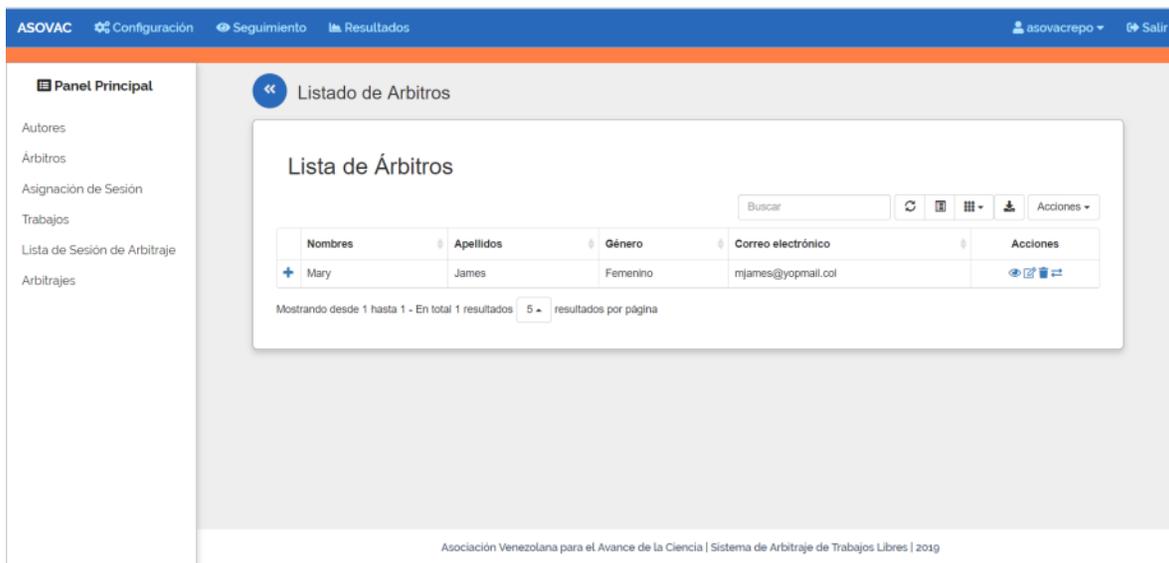


Figura N° 55. Interfaz para listar árbitros

Fuente: Elaboración propia

- **Interfaz de ver detalles del árbitro:** Los detalles de los árbitros se muestra al darle clic en el icono de ver detalles dentro de la tabla. En dicha vista es posible visualizar información del árbitro. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.
- **Interfaz de ver áreas del árbitro:** Las áreas del árbitro aparece al darle clic en el icono de áreas del árbitro dentro de la tabla. En dicha vista es posible visualizar información sobre las áreas a las cuales pertenece dicho

árbitro. Así mismo, es posible agregar y eliminar subáreas según sea la necesidad. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.

- **Interfaz de asignación de sesiones:** La figura a continuación se muestra al darle clic en el menú de navegación en seguimiento y posteriormente en asignación de sesiones. En dicha vista es posible visualizar información sobre los trabajos y su estatus justo al entrar en el proceso de asignación de sesión. Así mismo, es posible descargar la información disponible en dicha tabla. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.

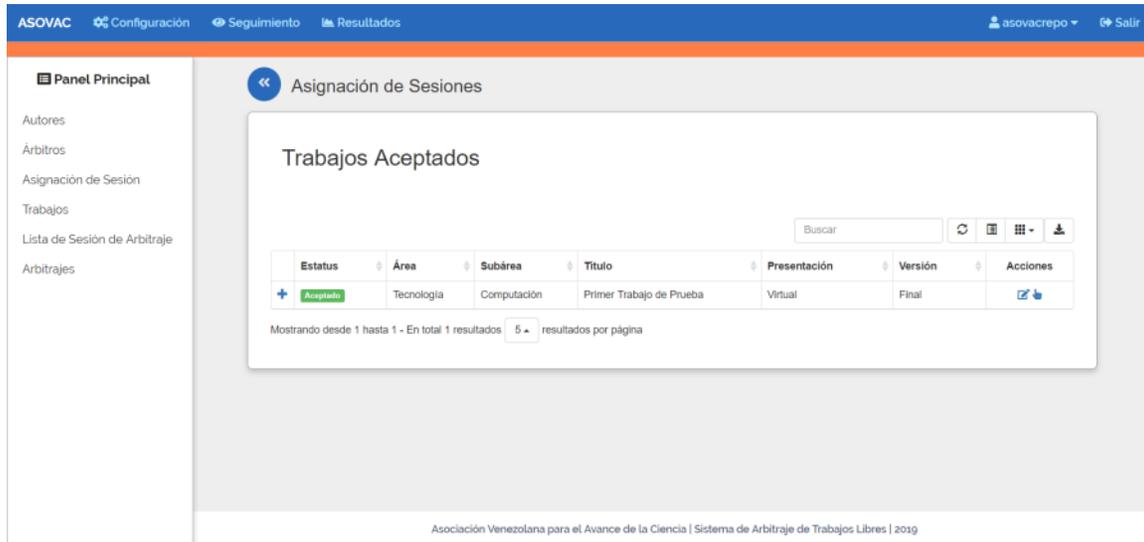


Figura N° 56. Interfaz de asignación de sesiones

Fuente: Elaboración propia

- **Interfaz de modalidad de presentación:** La vista de cambio de modalidad aparece al darle clic en el icono de editar dentro de las acciones de la tabla. En dicha vista es posible elegir el tipo de presentación para el artículo científico evaluado, este puede ser virtual o físico. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.
- **Interfaz de asignación de sesión:** La figura a continuación aparece al darle clic en el icono de modalidad dentro de las acciones de la tabla se despliega el modal para realizar la asignación de la sesión correspondiente. Así mismo, es posible descargar la información disponible en dicha tabla. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.

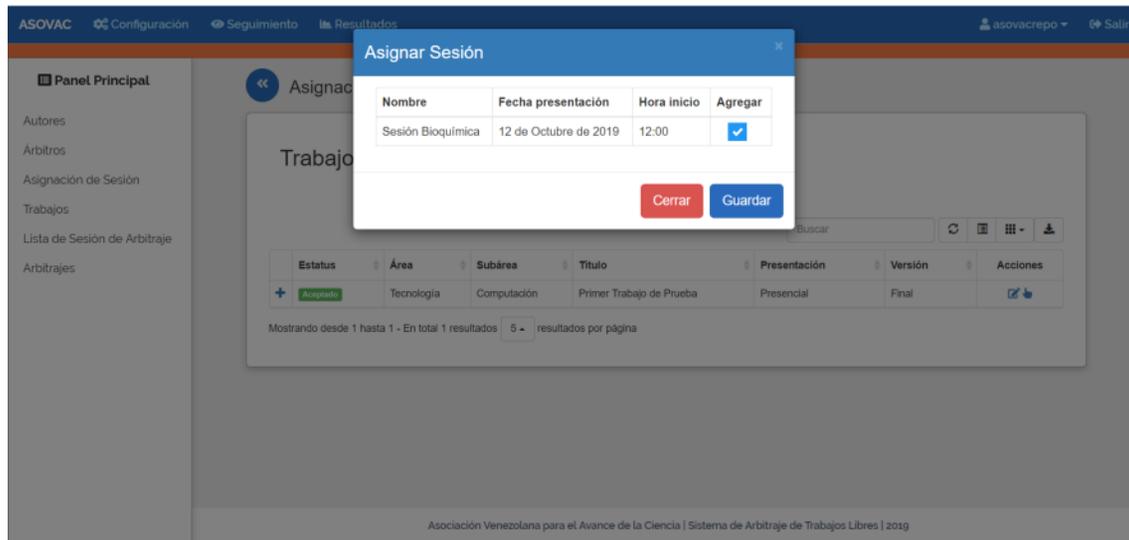


Figura N° 57. Interfaz de asignación de sesiones

Fuente: Elaboración propia

- **Interfaz de lista de trabajos:** La lista de trabajos aparece al darle clic en el menú de navegación en la opción de seguimiento y posteriormente en el menú lateral clic en trabajos. En dicha vista es posible administrar información sobre los trabajos y otras acciones. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.
- **Interfaz de asignación de árbitro:** La figura a continuación aparece al darle clic en el icono de asignar árbitro dentro de las acciones de la tabla se despliega el modal para poder agregar a los árbitros correspondientes según sea el caso. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.

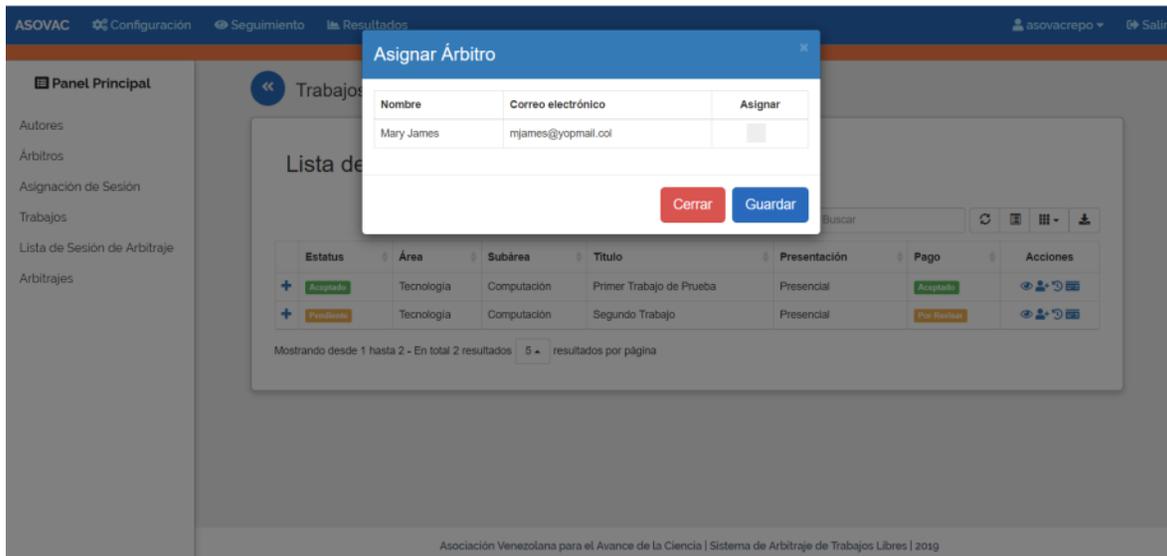


Figura N° 58. Interfaz de asignación de árbitros

Fuente: Elaboración propia

- **Interfaz de detalles de pago:** La figura a continuación aparece al darle clic en el icono de detalles del pago dentro de las acciones de la tabla realiza una redirección hacia la vista donde se podrá rechazar o aceptar un trabajo para iniciar el proceso de revisión de dicho artículo científico, así como también ver mayores detalles sobre los pagos correspondientes.

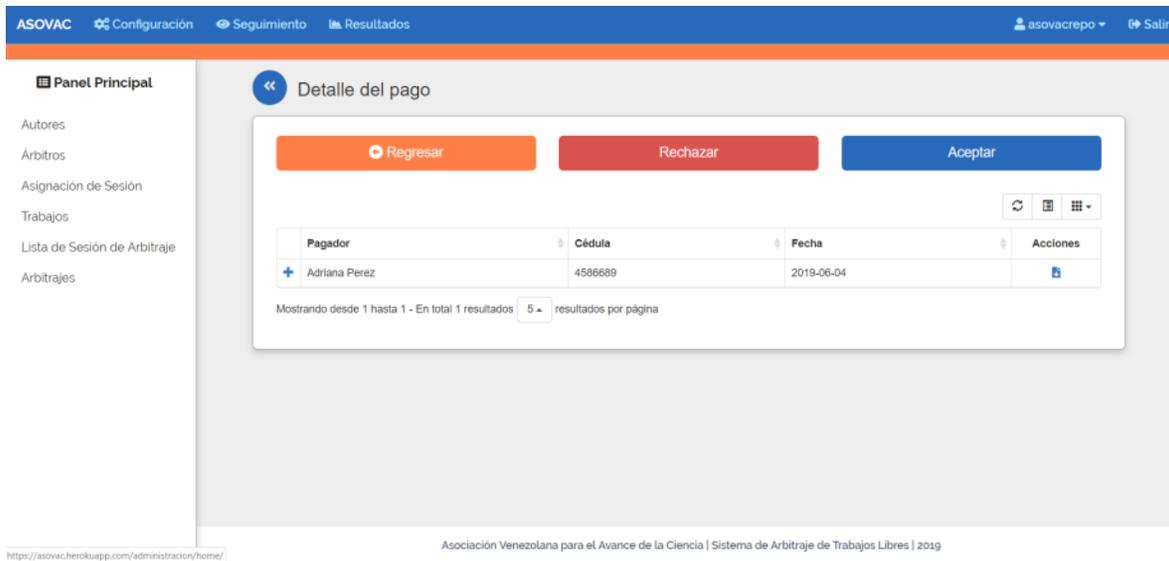


Figura N° 59. Interfaz de detalles del pago

Fuente: Elaboración propia

- **Interfaz de lista de sesiones de arbitraje:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de seguimiento y posteriormente en el menú lateral clic en lista de sesiones de arbitraje. En dicha vista es posible administrar información sobre las sesiones y otras acciones. Solo el coordinador general, el coordinador de área y el administrador tienen acceso a dicha interfaz.

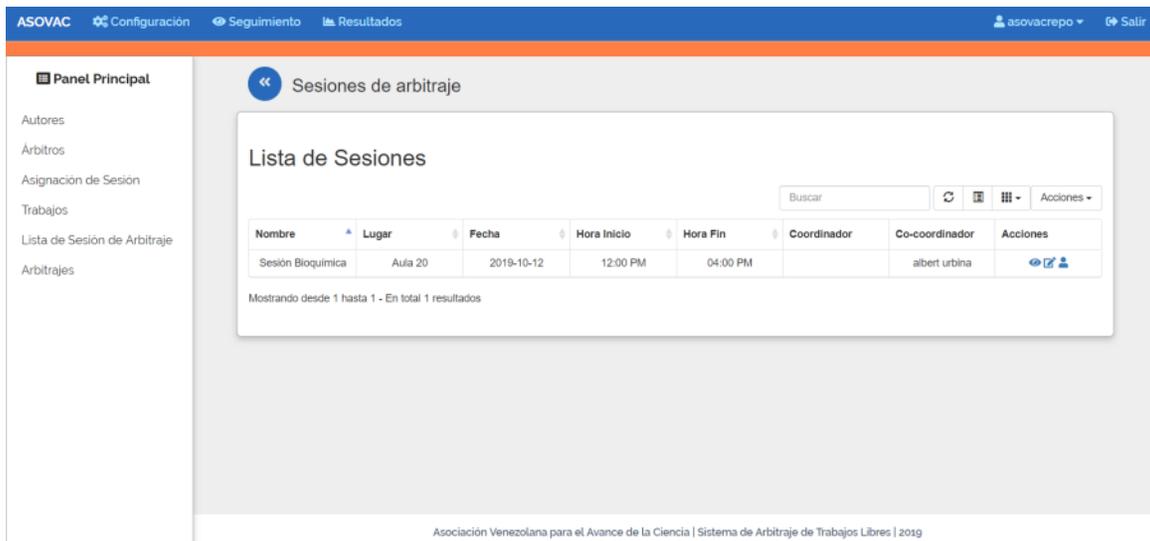


Figura N° 60. Interfaz para listar las sesiones

Fuente: Elaboración propia

- **Interfaz de crear sesión:** La vista de creación de sesión aparece al darle clic en el botón de acciones se despliegan varias opciones y se selecciona la del crear una sesión la cual redirige hacia otra vista donde es posible crear una nueva sesión donde los trabajos que se encuentran en estatus aprobado pueden ser asignados para su defensa en algún lugar dispuestos para la asistencia de otros participantes, árbitros, coordinadores, etc.
- **Interfaz de detalles de sesión:** Los detalles de la sesión aparece al darle clic en el icono de ver detalles dentro de las acciones de la tabla, se despliega el modal para poder visualizar la información correspondiente a dicha sesión.
- **Interfaz de editar sesión:** La figura a continuación aparece al darle clic en el botón de editar dentro de la columna de acciones redirige hacia otra vista donde es posible modificar la información de dicha sesión.

The screenshot shows a web interface for editing a session. The form is titled 'Editar sesión' and contains the following fields:

- Nombre de sesión***: Sesión Bioquímica
- Lugar***: Aula 20
- Fecha de sesión***: 12/10/2019
- Hora de inicio***: 12:00 PM
- Hora de fin***: 04:00 PM
- Observaciones**: Introdúzca sus observaciones
- Capacidad***: (empty field)

Figura N° 61. Interfaz de edición de sesiones

Fuente: Elaboración propia

- **Interfaz de asignar coordinadores de sesión:** La figura a continuación aparece al darle clic en el botón de asignación de sesión dentro de la columna de acciones, la cual redirige hacia otra vista donde al darle clic en la vista de acciones despliega el modal de para realizar la asignación correspondiente.

The screenshot shows a web interface for assigning coordinators to a session. The page is titled 'Sesiones de arbitraje' and 'Lista de trabajos: Sesión Bioquímica'. It features a search bar and a table view selector. The table contains the following data:

Título del trabajo	Cédula/Pasaporte	Correo Electrónico	Nivel de Instrucción	Acciones
Primer Trabajo de Prueba	V20616775	albert5693@yopmail.com	Educación Técnico/Profesional	

Mostrando desde 1 hasta 1 - En total 1 resultados

Figura N° 62. Interfaz de asignación de coordinadores de sesión

Fuente: Elaboración propia

- **Interfaz de Resultados:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de resultados. En dicha vista es posible ver información en forma de gráficos sobre métricas importantes relacionadas al proceso de arbitraje, así como también existe una acción de re direccionamiento dándole clic a la categoría de los gráficos que se deseen ver.

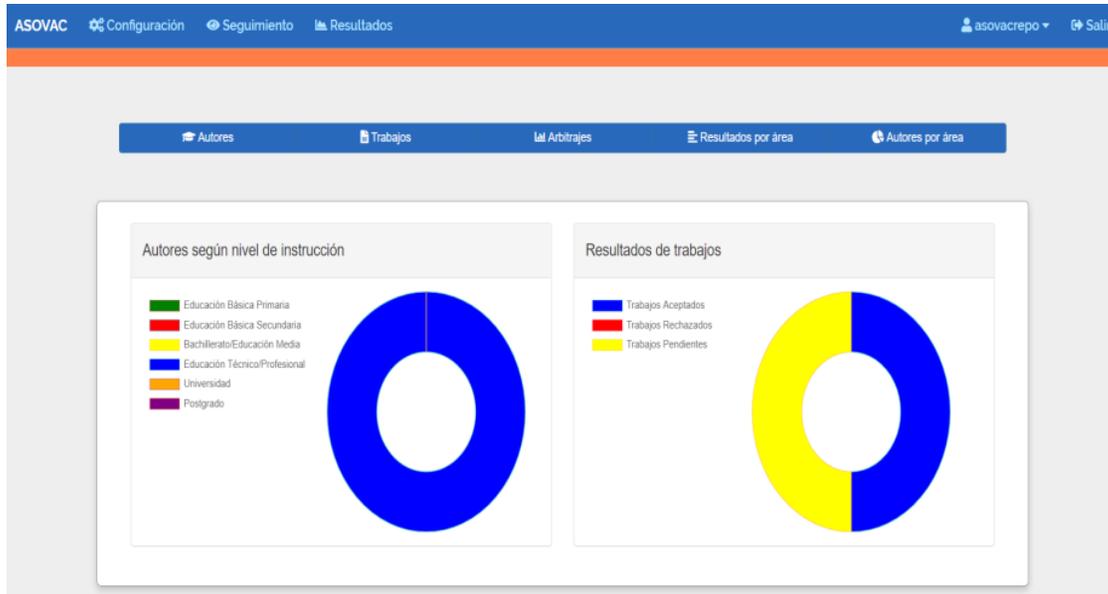


Figura N° 63. Interfaz para mostrar resultados parte 1

Fuente: Elaboración propia



Figura N° 64. Interfaz para mostrar resultados parte 2

Fuente: Elaboración propia

- **Interfaz de generación de certificados de autores:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de configuración, clic en el *sidebar* (barra lateral de navegación) en la opción de recursos y posteriormente en autores. En dicha vista se seleccionan los trabajos arbitrados aprobados o aprobados con observaciones para los cuales es posible generar un certificado con el papel de autor.

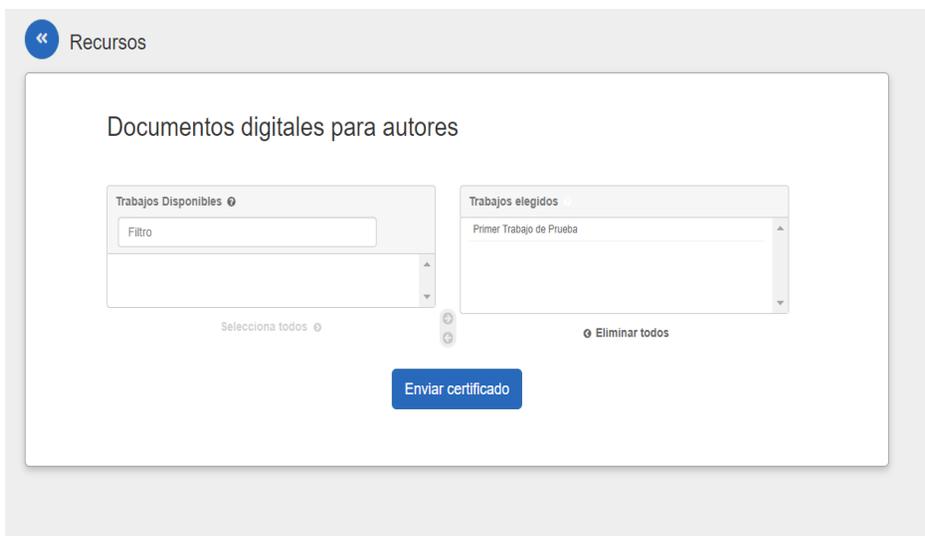


Figura N° 65. Interfaz para certificados de autor

Fuente: Elaboración propia

Así mismo, en esta interfaz al momento de realizar el envío se solicitará una confirmación para la generación de los certificados en formato pdf basado en las selecciones en la sección de árbitros disponibles, es posible realizar selecciones múltiples de ser necesario. A continuación, se presenta el formato base de dichos certificados.



Figura N° 66. Modelo base de Certificado de autores

Fuente: Elaboración propia

- **Interfaz de generación de certificados para árbitros:** La vista de generación de certificados aparece al darle clic en el menú de navegación en la opción de configuración, clic en el sidebar (barra lateral de

navegación) en la opción de recursos y posteriormente en opción de árbitros. En dicha vista se seleccionan los árbitros para los cuales es posible generar un certificado con el papel de autor.

De igual manera, en esta interfaz al momento de realizar el envío se solicitará una confirmación para la generación de los certificados en formato pdf basado en las selecciones en la sección de árbitros disponibles, es posible realizar selecciones múltiples de ser necesario.

- **Interfaz de selección de certificados relacionados a eventos:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de configuración, clic en la barra lateral de navegación (*sidebar*) en la opción de recursos y posteriormente en opción de eventos. En dicha vista se seleccionan diferentes tipos de certificados relacionados a eventos y roles desarrollado en los mismos.

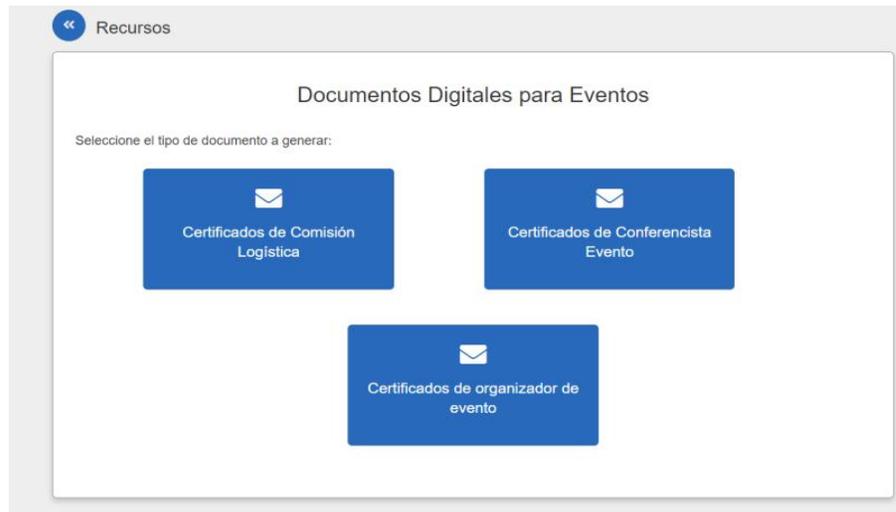


Figura N° 67. Interfaz para certificados relacionados a eventos

Fuente: Elaboración propia

- **Interfaz de generación de certificados para comisión logística:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de configuración, clic en la barra lateral de navegación (*sidebar*) en la opción de recursos, luego en eventos y finalmente en elegir la opción de certificados de comisión logística. En dicha vista se introducen los nombres de las personas que participan en la comisión de logística y se realiza la generación de sus certificados que serán enviados por correo electrónico.

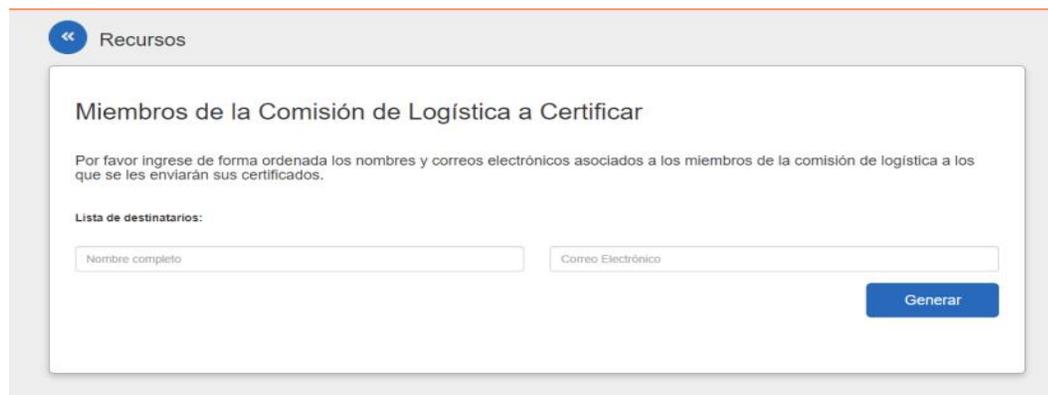


Figura N° 68. Interfaz para certificados comisión logística

Fuente: Elaboración propia

Así mismo, al momento de realizar la generación de los certificados se solicita una confirmación para el envío de los certificados en formato pdf.

- **Interfaz de generación de certificados para organizadores:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de configuración, clic en la barra lateral de navegación (*sidebar*) en la opción de recursos, luego en eventos y finalmente en elegir la opción de certificados de organizadores. En dicha vista se introducen los nombres de las personas que participan como organizadores y se realiza la generación de sus certificados que serán enviados por correo.

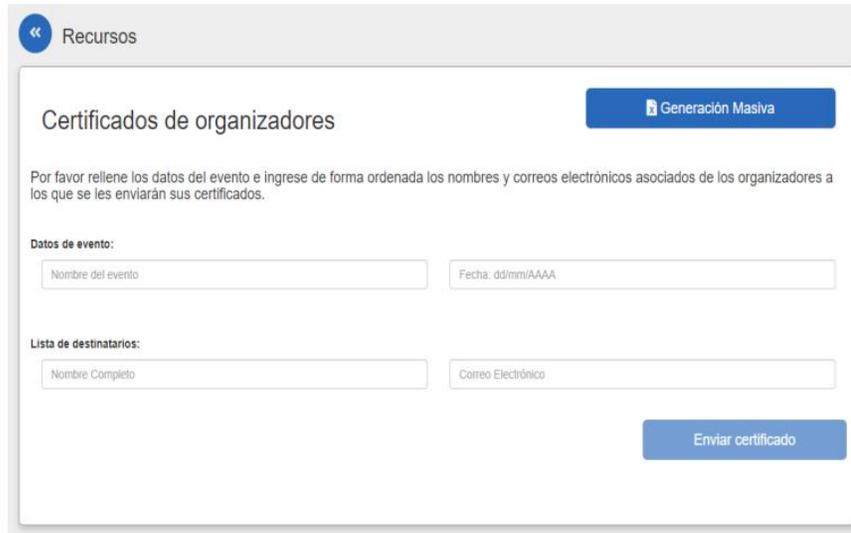


Figura N° 69. Interfaz para certificados de organizadores

Fuente: Elaboración propia

- **Interfaz de generación de certificados para conferencistas:** Los certificados para conferencistas aparecen al darle clic en el menú de navegación en la opción de configuración, clic en la barra lateral de navegación (*sidebar*) en la opción de recursos, luego en eventos y finalmente en elegir la opción de certificados de conferencistas. En dicha vista se introducen los nombres de las personas que participan como conferencistas y se realiza la generación de sus certificados pdf, ya que serán enviados por correo.
- **Interfaz de generación de certificados para sesión:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de configuración, clic en la barra lateral de navegación (*sidebar*) en la opción de recursos y posteriormente en opción de sesión.

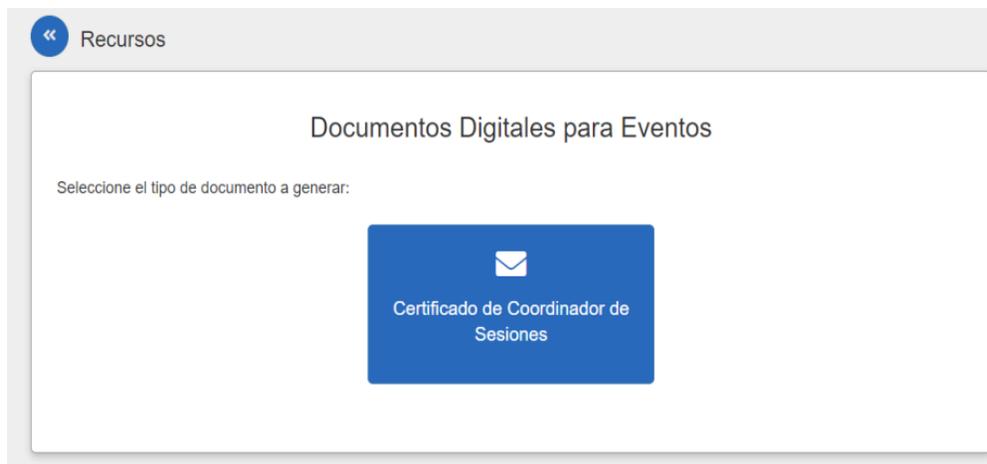


Figura N° 70. Interfaz para certificados de sesiones

Fuente: Elaboración propia

- **Interfaz de generación de certificados para coordinadores de sesión:** Los certificados para coordinadores de sesión aparecen al darle clic en el menú de navegación en la opción de configuración, clic en la barra lateral de navegación (*sidebar*), en la opción de recursos, opción de sesiones y posteriormente seleccionar certificado de coordinador de Sesiones. En dicha vista se introducen los nombres de las personas que participan en la comisión de logística y se realiza la generación de sus certificados que serán enviados por correo.
- **Interfaz de generación de certificados para autores por trabajos:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de configuración, clic en la barra lateral de navegación (*sidebar*), en la opción de recursos y posteriormente en opción de trabajos. En dicha vista se seleccionan los trabajos aprobados u aprobados con observaciones que participan en el proceso de arbitraje y se realiza la generación de sus certificados que serán enviados por correo.

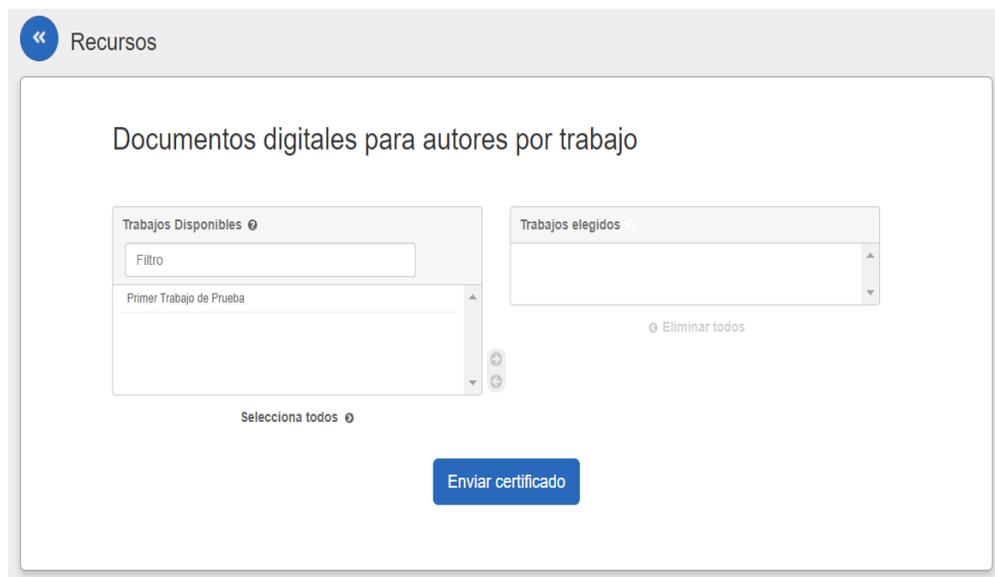


Figura N° 71. Interfaz para certificados de autores por trabajo

Fuente: Elaboración propia

De igual manera, en esta interfaz al momento de realizar el envío se solicitará una confirmación para la generación de los certificados en formato pdf basado en las selecciones en la sección de trabajos disponibles, es posible realizar selecciones múltiples de ser necesario.

- **Interfaz de selección de certificados AsoVAC:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de configuración, clic en barra lateral de navegación (*sidebar*), en la opción de recursos y posteriormente en opción de AsoVAC. En dicha vista selecciona alguna de las opciones que se observan a continuación.

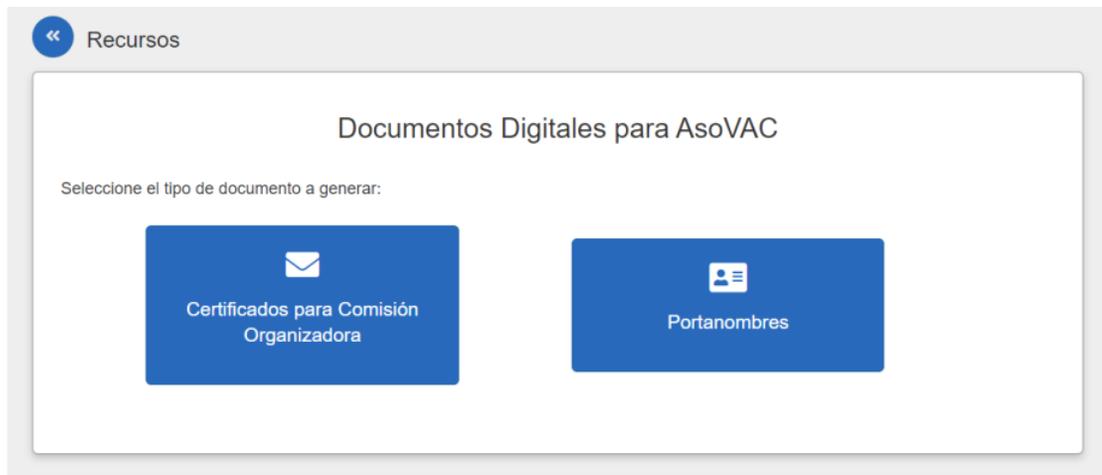


Figura N° 72. Interfaz para certificados digitales de AsoVAC

Fuente: Elaboración propia

- **Interfaz de generación de certificados para comisión organizadora:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de configuración, clic en la barra lateral de navegación (*sidebar*) en la opción de recursos, opción de AsoVAC, y posteriormente certificado de comisión organizadora. En dicha vista se introducen los nombres de las personas que participan en la comisión organizadora y se realiza la generación de sus certificados que serán enviados por correo electrónico.

Figura N° 73. Interfaz para porta nombre de comisión organizadora

Fuente: Elaboración propia

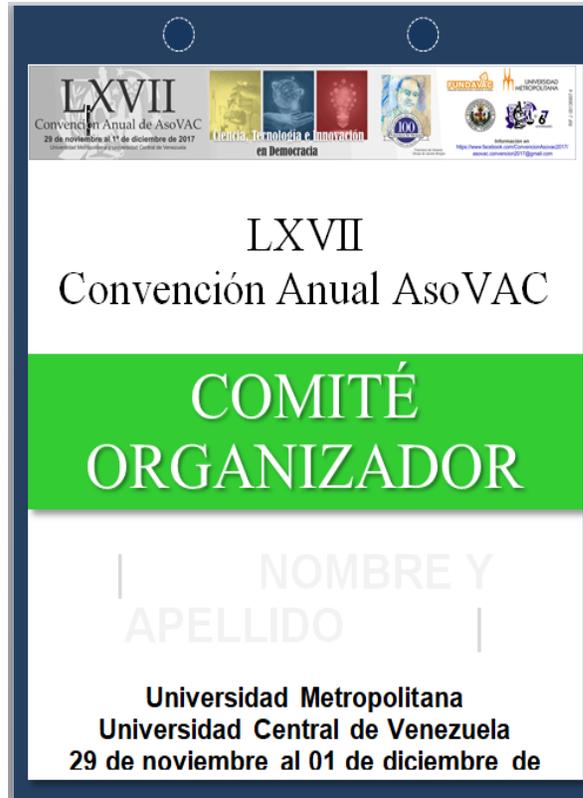


Figura N° 74. Modelo de porta nombre de comisión organizador
Fuente: Elaboración propia

- **Interfaz de generación de porta nombres:** La figura a continuación aparece al darle clic en el menú de navegación en la opción de configuración, clic en la barra lateral de navegación (*sidebar*) en la opción de recursos, opción de AsoVAC y posteriormente selección de porta nombres. En dicha vista se selecciona un rol para los porta nombres, el cual puede ser asistente, conferencista o autor y posteriormente se agrega los datos de las personas para ser enviado por correo electrónico.

Adicionalmente, en esta vista es posible realizar generación de porta nombres de forma masiva a través de la carga de un archivo .xlsx. Todo eso al darle clic en el botón de generación masiva el cual muestra un modal donde aparecen las opciones de descargar el formato de ejemplo, cancelar e importar. Cabe destacar que la generación por esta vía indicara si el archivo .xlsx posee errores y en caso de tenerlo se lo indicara al usuario.

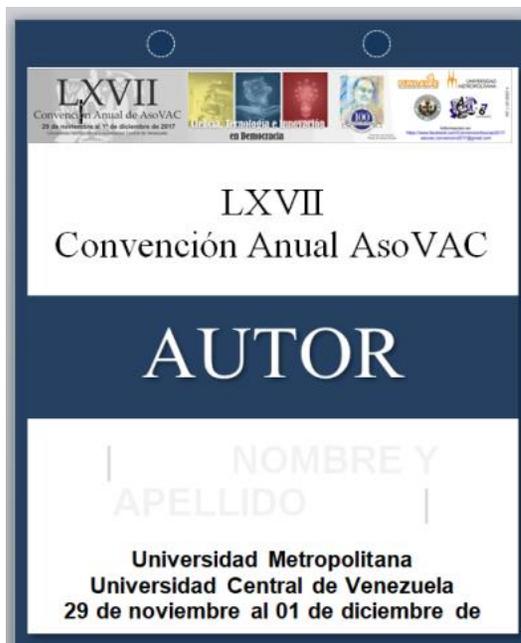


Figura N° 75. Modelo de porta nombre de autor

Fuente: Elaboración propia

4.8 Fase de Pruebas

Las pruebas realizadas para verificar el correcto funcionamiento del sistema y la aceptación de los usuarios por la aplicación se dividieron en dos partes. La primera, basada en el correcto funcionamiento de la misma a través de pruebas funcionales: La segunda, pruebas de aceptación realizadas para los usuarios de la aplicación con especial atención a criterios de usabilidad.

4.8.1 Pruebas funcionales

Para registrar los resultados de las pruebas se realizaron varios *sprint*, especificando los módulos pertenecientes al sistema y las funcionalidades que debería tener cada uno de ellos, como se observa en la Tabla N° 2.

Tabla N° 2. Cuadro de pruebas funcionales realizadas por *sprint*

Fuente: creación propia

Sprint	Funcionalidades probadas	Resultados
1	Revisión del modelo de la Base de datos utilizando datos de pruebas ingresados en PostgreSQL.	El modelo facilita la consultas y el almacenaje de la información necesaria
2	Revisión de componentes para diseño adaptativo para las resoluciones de 1024*768, 1280*720 en los navegadores de Firefox y Google Chrome	Comportamiento adaptativo según las resoluciones mencionadas
3	Registro de usuario	Registro de usuarios de forma exitosa
	Envío de correo electrónico	Envío de correo de forma exitosa a través de un correo Gmail
	Ingreso a la aplicación	Ingreso a la aplicación de forma exitosa

Sprint	Funcionalidades probadas	Resultados
	Recuperación de contraseña	Recepción de correo electrónico para la recuperación de contraseña de forma exitosa
	Reinicio de contraseña	Se generaron contraseñas aleatorias de forma satisfactoria enviadas por correo electrónico
	Acceso por roles	Según el rol que posee el usuario accede correctamente a la aplicación
4	Revisión de acceso por roles según el estado del sistema de arbitraje	Cada usuario con su rol asignado fue capaz de ingresar a la aplicación en el estado del arbitraje que debía acceder. Así mismo el acceso denegado cuando no debía entrar con dicho rol.
	Prueba de envío de certificados adjuntos vía correo electrónico	Recepción de certificados adjuntos en el correo electrónico
	Generación la plantilla base para los certificados digitales	Generación correcta de la base para los certificados
5	Revisión de vista correcto funcionamiento de vista 404	Vista de acceso perdido, o sin información
	Validación de acceso por roles para el administrador, coordinador general y coordinador de área	Correcto bloqueo de vistas para roles que no poseen acceso a dichas vistas
	Generación de claves maestras según rol enviadas por correo electrónico	Recepción de claves maestras vía correo electrónico según el rol del usuario
	Carga masiva para coordinadores de área y árbitros de subárea	Carga a través de archivos .xlsx con cabeceras definidas e información correcta
	Manejo de errores para la carga masiva de coordinadores de área y árbitros de subárea	Generación de errores en archivo .xlsx indicando los errores en la subida de información
	CRUD de Organizadores	Operaciones de creación, actualización, lectura y eliminación exitosa de eventos
6	Mejoras del diseño adaptativo para toda la aplicación y sus múltiples módulos	Validación de mejoras de diseño adaptativo para los navegadores Firefox y Chrome
7	CRUD de trabajos por parte de los autores	Validación de la correcta creación, eliminación, actualización y lectura de los trabajos cargados de parte de los autores
	Postulación del trabajo	Postulación exitosa del trabajo para uno o múltiples autores
	Flujo de pagador del trabajo	Verificación del proceso de pago para diferentes tipos, cantidad y combinaciones de pago de moneda para el arbitraje
	Revisión de accesos del autor según el estado del sistema del arbitraje	Ratificación de accesos correctos en los estados donde interviene el autor, y bloqueo de acceso para los estados en los que no debe ingresar
8	CRUD de autores	Verificación de la correcta creación, eliminación, actualización y lectura de los autores
	Registro de usuarios relacionado a áreas y subáreas	Agregación correcta de áreas y subáreas a usuarios creados

Sprint	Funcionalidades probadas	Resultados
	Filtro para búsqueda de usuarios por área y subáreas	Correcto funcionamiento del filtro de búsqueda aplicado para áreas y subáreas
9	Verificación de acceso de rol usando clave de seguridad	Correcto acceso de usuarios según su rol, y verificación de identidad
	CRUD de locaciones	Validación de operaciones de creación, actualización, lectura y eliminación exitosa para las locaciones
	CRUD de eventos	Validación de operaciones de creación, actualización, lectura y eliminación exitosa para los eventos
10	Mejoras de las tablas utilizando <i>bootstrap table</i> en todas las interfaces del usuario	Verificación del correcto funcionamiento con la implementación de <i>bootstrap table</i>
	Descargar archivo de cabecera .xlsx desde <i>bootstrap table</i>	Descarga exitosa de los archivos de tipo formato de cabeceras para realizar la subida
	Crear botón de acciones junto a las tablas de <i>bootstrap table</i>	Validación de ejecución de todas las funcionalidades dentro de botón de acciones para las tablas
	Definir mínimo de campos en una tabla dinámica de <i>bootstrap table</i>	Validación de mínimo de campos para todas las tablas basadas en <i>bootstrap table</i>
11	Configuración inicial de la base de datos de forma automática	Verificación de la carga exitosa de los roles de usuario base.
	Carga de trabajos vía archivos .xlsx	Carga exitosa de los trabajos utilizando archivos de extensión .xlsx
	Descarga y carga de autores mediante archivos .xlsx	Descarga exitosa con la información correcta de los autores mediante archivos de extensión .xlsx
	Carga de trabajos en formato pdf	Subida exitosa de los trabajos en formato pdf incluyendo la previsualización de los mismos
	Mostrar y agregar comentarios en los trabajos siendo coordinador de área, árbitro de subárea	Validación exitosa de agregar y mostrar comentarios en los trabajos por parte de los coordinadores de área y árbitros de subárea
12	CRUD de árbitros de subárea	Validación de la correcta creación, eliminación, actualización y lectura de los árbitros de subárea
	Agregar validaciones en el CRUD de autores	Validaciones funcionando correctamente
	Agregar validaciones en el CRUD de trabajos	Validaciones funcionando correctamente
	Agregar validaciones en el CRUD de árbitros de subárea	Validaciones funcionando correctamente
	Agregar validaciones en el CRUD de locaciones	Validaciones funcionando correctamente
	Agregar validaciones en el CRUD de organizador	Validaciones funcionando correctamente
	Agregar validaciones en el CRUD de eventos	Validaciones funcionando correctamente
13	CRUD de Asignación de sesiones	Validación de la correcta creación, eliminación, actualización y lectura de los trabajos cargados de parte de los autores
	Asignación de códigos para áreas y subáreas	Verificación de asignaciones para los

Sprint	Funcionalidades probadas	Resultados
		códigos de áreas y subáreas
	Arbitraje de trabajos	Comprobación del proceso de arbitraje por los árbitros de subárea
14	Generación dinámica para certificados de trabajo aprobado enviados vía correo electrónico	Validación de la correcta generación del certificado de trabajo aprobado
	Generación certificados de trabajo aprobado con observaciones enviados vía correo electrónico	Validación de la correcta generación del certificado de trabajo aprobado con observaciones
	Generación certificados de trabajo rechazado enviados vía correo electrónico	Validación de la correcta generación del certificado de trabajo rechazado
	Correcciones de usabilidad y experiencia de usuario en un conjunto de vistas.	Verificación de mejoras y correcciones en el diseño de la aplicación
	Generación dinámica de resultados a través de gráficos utilizando Chartjs	Verificación de gráficos generaciones a través de consultas hacia la base de datos
15	Configuración base para todos los certificados en el sistema de arbitraje	Verificación en la creación de nuevos procesos de arbitraje con su información por defecto para la generación de los certificados
15	Configuración, y actualización de IVA, montos de arbitraje en datos básicos	Verificación de la actualización sobre el IVA y el monto de postulación para los trabajos
15	Generación de porta nombres para asistentes, autores y conferencistas	Validación de generación de porta nombres para asistentes, autores y conferencistas
15	Generación de certificados para comisión organizadora	Validación de generación de los certificados para la comisión organizadora
15	Generación de certificados para coordinadores de sesiones	Validación de generación de los certificados para coordinadores de sesiones
15	Generación de certificados masivos utilizando archivos .xlsx	Validación de generación de certificados masivos utilizando archivos .xlsx para aquellos usuarios que no participan directamente en el proceso de arbitraje
15	Validación de generación de certificados individuales y masivos usando archivos .xlsx	Verificación de generación de certificados masivos e individuales para usuarios de los procesos de arbitraje y usuarios externos.

4.8.2 Pruebas de integración entre módulos

A continuación, se van a desglosar las pruebas de integración y los resultados obtenidos para cada uno de los estados del proceso de arbitraje tomando en cuenta el rol de usuario que realiza las acciones. Los resultados serán basados en la funcionalidad creada y el nivel de automatización logrado con respecto a las convenciones anteriores realizadas por AsoVAC.

-Estado Desactivado: En este estado el único usuario que posee acceso es el administrador. En este estado se realiza la configuración de las áreas y subáreas dentro de la aplicación para los distintos sistemas de arbitraje, este proceso se hace automáticamente y consiste en una migración de datos.

Adicionalmente, Se debe agregar y elegir a un usuario como coordinador general, ya que para pasar al estado Iniciado este es fundamental. Es posible migrar usuarios existentes en otros sistemas de arbitrajes y agregarles un rol dentro del actual sistema.

Así mismo, es posible editar los datos básicos del arbitraje el cual afecta a otros estados y módulos existentes en la aplicación. Actualizar el IVA, el monto de los arbitrajes, cargar firmas e imagen de cabecera para la generación de los certificados son varias de las funcionalidades que son posibles.

Para finalizar, los resultados tras la interacción de los módulos de administración, recursos, áreas y subáreas permiten hacer este proceso con mayor facilidad como también poder hacerlo desde cualquier navegador con acceso a internet.

-Estado Iniciado: En este estado ya existe un usuario asignado para ser el coordinador general. Se pueden llevar a cabo cambios en las áreas, subáreas si es necesario.

Así mismo, es posible agregar usuarios como autores, como árbitros ya que estos posteriormente tendrán una participación en el proceso de arbitraje. Análogamente, los usuarios pueden registrarse de forma independiente e igualmente tanto el administrador como el coordinador general podrá asignarles un rol, o actualizar su información en caso de ser necesario.

Cabe destacar que es posible cambiar la asignación del coordinador general, o utilizar algunas de las funcionalidades existentes. Por otra parte, este punto permite ir preparando el proceso de revisión por pares a través de la aplicación. Los módulos que intervienen son áreas y subáreas, administración, autores, y seguimiento.

-Estado En selección y asignación de coordinadores de área: En este estado se realizan las asignaciones de los coordinadores de área, los cuales permiten el control sobre el proceso. Así mismo, se administran las áreas y subáreas, carga de autores a través de archivos .xlsx para preparar el proceso de revisión por pares que se llevara a cabo. Los módulos que influyen en estas funcionalidades son: administración y autores.

-Estado En carga de trabajos: En este estado se realiza la asignación de árbitros de subárea, la postulación de los trabajos, la carga masiva de autores, cada una de estas operaciones puede ser realizada por diferentes roles.

Así mismo, muchos de estos procesos pueden ser realizados por los coordinadores de área, coordinador general utilizando las opciones de masiva. Es un punto clave de la aplicación puesto que la postulación de los trabajos es lo que hará posible la revisión por pares.

En este estado intervienen los módulos de autores, administración, trabajos, arbitraje, áreas y subáreas, por ende, muchas de las funcionalidades desarrolladas dependen y coexisten entre los módulos de forma correcta.

-Estado En asignación de árbitros a trabajos: En este estado se realiza la asignación de los árbitros a los diferentes trabajos en todas sus áreas y subáreas existentes. Esto es posible ya que se han cargado y postulado todos los artículos que van a ser arbitrados. Según las reglas de negocios definidas es posible asignar varios árbitros de subárea a los trabajos según el criterio del coordinador de área o el coordinador general.

Esta asignación se lleva a cabo según el proceso de revisión por pares donde intervienen los módulos de arbitraje, trabajos, autores, áreas y subáreas, administración, los cuales se integran correctamente generando replicando las reglas de negocio desarrolladas.

-Estado En arbitraje: En este estado del proceso se realizan las revisiones de los artículos postulados y asignados por los árbitros especializados en las áreas, consiste en validar el formato, calidad de trabajo, y si este aporta o no a la comunidad científica. El arbitraje de los artículos científicos puede tener tres estados, pendiente

cuando aún no ha sido tomado por los árbitros asignados, rechazado cuando los árbitros asignados han dispuesto que este no cumple con los requisitos para aprobarse y aprobado cuando agrega valor a la comunidad científica.

Así mismo, el módulo de arbitraje, sesiones, trabajos, autores son fundamentales en este punto ya que la conjunción de todos hace posible este punto tan crítico en la aplicación. Junto con el módulo de administración permiten el correcto funcionamiento de corrección de dichos artículos.

-Estado En cierre de arbitraje: En este punto el proceso de arbitraje para todos los trabajos fue realizado en su gran mayoría, pero es posible que aún no esté finalizado. Así mismo, se lleva a cabo el monitoreo y seguimientos a todos los arbitrajes en cada uno de las áreas y subáreas existentes. La generación de métricas de resultados comienza a llevarse a lugar e incluso la generación de los recursos es posible hacerla en dicho estado.

Los módulos que intervienen exitosamente son: administración, arbitraje, seguimiento, trabajos y recursos, los cuales funciona según lo definido en capítulos anteriores.

-Estado En asignación de sesiones: En este punto del proceso ya se poseen resultados de todos los trabajos arbitrados, es decir, el proceso de revisión por pares tiene un resultado. El coordinador general, coordinadores de área serán capaces de administrar las sesiones para los trabajos aprobados.

Los módulos involucrados en este proceso son: de autores, de sesiones, de trabajos, de administración, y de arbitraje. La integración y relación existente entre ellos permite definir la operación de la asignación de sesión una vez el trabajo allá sido arbitrado por los árbitros de subáreas donde posteriormente la sesión entra en juego para definir el lugar, tipo de presentación del trabajo de investigación científica.

-Estado En resumen: En dicho estado el proceso de arbitraje se dio por culminado, ya se poseen resultados totales, es posible visualizarlos, compararlos y exportarlos en archivos .xlsx para realizar procesos externos a la aplicación.

De igual manera, en dicho estado se generan la gran mayoría de los certificados existentes en la aplicación ya que tanto los usuarios que participaron en el sistema de arbitraje como los usuarios con participación externa al mismo son manejados de forma independiente. Se puede generar certificados individuales o certificados de forma masiva a través de archivos .xlsx dependiendo del propósito del certificado.

En este punto los módulos principales que se encargan de todas estas acciones son: administración, recursos, arbitraje, seguimiento y autores.

Para finalizar, existe un módulo externo al proceso revisión por pares, el cual tiene como finalidad centrarse en la administración de eventos para AsoVAC, en el mismo se registración lugares con características específicas, las cuales son manejadas por organizadores asociados a ciertos eventos, es decir, un usuario es capaz de realizar la creación de eventos de forma automatizada.

Así mismo, esta funcionalidad esta soportada por el módulo de eventos, el cual trabaja en común con el módulo de administración.

4.9 Resultados

El Sistema de Gestión de Arbitraje de Artículos de Investigación fue desarrollado tomando en cuenta los diferentes procesos que se realizan durante la convención anual realizada por la AsoVAC, el funcionamiento de sistemas que realizan el proceso de revisión por pares en la actualidad y adaptando este proceso a las necesidades de esta convención. Por ende se realizaron validaciones en las funcionalidades desarrolladas obteniendo los siguientes resultados.

Tabla N° 3. Resultados parte I

Fuente: creación propia

Resultados		
Funcionalidad: Iniciar sesión		
Primera entrada	Segunda entrada	Tercera entrada
Entrada: vacío	Entrada: credenciales incorrectas	Entrada: credenciales válidas
Resultado esperado: se muestra un mensaje de error notificándole al usuario que sus credenciales son incorrectas	Resultado esperado: se muestra un mensaje de error notificándole al usuario que sus credenciales son incorrectas	Resultado esperado: se muestra la página de bienvenida de la aplicación y se muestra un mensaje notificando que la autenticación fue satisfactoria.
Resultado obtenido: se muestra un mensaje de error notificándole al usuario que sus credenciales son incorrectas	Resultado obtenido: se muestra un mensaje de error notificándole al usuario que sus credenciales son incorrectas	Resultado obtenido: se muestra la página de bienvenida de la aplicación y se muestra un mensaje notificando que la autenticación fue satisfactoria.
Funcionalidad: Recuperación de contraseña		
Entrada: vacío	Entrada: correo electrónico incorrecto	Entrada: correo electrónico válido
Resultado esperado: se muestra un mensaje de error notificándole al usuario que sus credenciales son incorrectas	Resultado esperado: se muestra un mensaje de error notificándole al usuario que sus credenciales son incorrectas	Resultado esperado: se envía el correo con la contraseña de recuperación y se redirecciona la página al inicio de sesión.
Resultado obtenido: se muestra un mensaje de error notificándole al usuario que sus credenciales son incorrectas	Resultado obtenido: se muestra un mensaje de error notificándole al usuario que sus credenciales son incorrectas.	Resultado obtenido: se envía el correo con la contraseña de recuperación y se redirecciona la página al inicio de sesión.
Funcionalidad: Cargar autores vía archivo .xlsx		
Entrada: vacío	Entrada: colocar algunos campos erróneos	Entrada: colocar todos los campos validos
Resultado esperado: se muestra un mensaje indicado que el archivo no tenía información	Resultado esperado: se muestra un mensaje indicando en cual columna se encuentra el primer error en el archivo ,xlsx	Resultado esperado: se muestran un mensaje indicando que se crearon los autores satisfactoriamente.
Resultado obtenido: se muestra un mensaje indicado que el archivo no tenía información.	Resultado obtenido: se muestra un mensaje indicando en cual columna se encuentra el primer error en el archivo ,xlsx	Resultado obtenido: se muestran un mensaje indicando que se crearon los autores satisfactoriamente.

Tabla N° 4. Resultados parte II

Fuente: creación propia

Resultados		
Funcionalidad: Cargar árbitros de subárea vía archivo .xlsx		
Primera entrada	Segunda entrada	Tercera entrada
Entrada: vacío	Entrada: color algunos campos erróneos	Entrada: color todos los campos validos
Resultado esperado: se muestra un mensaje indicado que el archivo no tenía información	Resultado esperado: se muestra un mensaje indicando en cual columna se encuentra el primer error en el archivo .xlsx	Resultado esperado: se muestran un mensaje indicando que se crearon los autores satisfactoriamente.
Resultado obtenido: se muestra un mensaje indicado que el archivo no tenía información	Resultado obtenido: se muestra un mensaje indicando en cual columna se encuentra el primer error en el archivo .xlsx	Resultado obtenido: se muestran un mensaje indicando que se crearon los autores satisfactoriamente.
Funcionalidad: Postular trabajo		
Entrada: sin adjuntar el trabajo	Entrada: datos inválidos	Entrada: datos válidos y formato valido para trabajos
Resultado esperado: se muestra un mensaje de error indicado que el formato del trabajo es el incorrecto.	Resultado esperado: se muestra un mensaje de error indicando los campos que están incorrectos o que poseen datos inválidos.	Resultado esperado: se muestra un mensaje de trabajo postulado exitosamente y se muestra una vista con los detalles del mismo.
Resultado obtenido: se muestra un mensaje de error indicado que el formato del trabajo es el incorrecto.	Resultado obtenido: se muestra un mensaje de error indicando los campos que están incorrectos o que poseen datos inválidos.	Resultado obtenido: se muestra un mensaje de trabajo postulado exitosamente y se muestra una vista con los detalles del mismo.
Funcionalidad: Cargar pagos		
Entrada: campos inválidos	Entrada: cargar pago en formato invalido	Entrada: cargar pago en formato valido
Resultado esperado: se muestra un mensaje indicando que no se cargó el pago correctamente.	Resultado esperado: se muestra un mensaje indicando que no se cargó el pago ya que no se está utilizando un formato valido para guardar el comprobante de pago.	Resultado esperado: se muestra un mensaje de pago cargado exitosamente y este aparece en la tabla de los pagos de la postulación de dicho trabajo.
Resultado obtenido: se muestra un mensaje indicando que no se cargó el pago correctamente.	Resultado obtenido: se muestra un mensaje indicando que no se cargó el pago ya que no se está utilizando un formato valido para guardar el comprobante de pago.	Resultado obtenido: se muestra un mensaje de pago cargado exitosamente y este aparece en la tabla de los pagos de la postulación de dicho trabajo.

Tabla N° 5. Resultados parte III

Fuente: creación propia

Resultados		
Funcionalidad: Arbitrar trabajos		
Entrada: trabajo no verificado	Entrada: rechazar trabajo.	Entrada: aprobar trabajo
Resultado esperado: este(os) trabajo(s) no aparece en la lista de trabajos por arbitrar, ya que no se verificaron sus pagos y por ende su postulación es anulada.	Resultado esperado: el trabajo es arbitrado por todos los árbitros de subárea asignados al mismo pero el coordinador de dicha área o el coordinador general deciden rechazado. Por ende, se le envía un correo indicándole el rechazo de la publicación científica postulada.	Resultado esperado: el trabajo es arbitrado por todos los árbitros de subárea asignados al mismo y el coordinador de dicha área o el coordinador general deciden aprobarlo. Por ende, se le envía un correo indicándole la aprobación de la publicación científica postulada.
Resultado obtenido: Este(os) trabajo(s) no aparece en la lista de trabajos por arbitrar, ya que no se verificaron sus pagos y por ende su postulación es anulada.	Resultado obtenido: el trabajo es arbitrado por todos los árbitros de subárea asignados al mismo pero el coordinador de dicha área o el coordinador general deciden rechazado. Por ende, se le envía un correo indicándole el rechazo de la publicación científica postulada.	Resultado obtenido: el trabajo es arbitrado por todos los árbitros de subárea asignados al mismo y el coordinador de dicha área o el coordinador general deciden aprobarlo. Por ende, se le envía un correo indicándole la aprobación de la publicación científica postulada.
Funcionalidad: Crear sesión		
Entrada: falta alguno de los campos obligatorios	Entrada: datos inválidos para la sesión	Entrada: datos válidos para la sesión
Resultado esperado: se muestra un mensaje indicado cual(es) son los campos obligatorios que necesitan ser llenados para crear la sesión.	Resultado esperado: se muestra un mensaje indicado que los datos introducidos son erróneos.	Resultado esperado: se muestra un mensaje indicado la creación exitosa de la sesión y se actualiza la tabla de la sesiones.
Resultado obtenido: se muestra un mensaje indicado cual(es) son los campos obligatorios que necesitan ser llenados para crear la sesión.	Resultado obtenido: se muestra un mensaje indicado que los datos introducidos son erróneos.	Resultado obtenido: se muestra un mensaje indicado la creación exitosa de la sesión y se actualiza la tabla de la sesiones.

Tabla N° 6. Resultados parte IV

Fuente: creación propia

Resultados		
Funcionalidad: Generar certificados a autores		
Primera entrada	Segunda entrada	Tercera entrada
Entrada: vacío	Entrada: se selecciona un trabajo	Entrada: Seleccionar todos los trabajos
Resultado esperado: se muestra un mensaje para indicar que el campo es requerido.	Resultado esperado: se genera el certificado y es enviado vía correo electrónico a los autores y coautores correspondientes.	Resultado esperado: se generan los certificados y son enviados a todos los autores y coautores de dichos trabajos.
Resultado obtenido: se muestra un mensaje para indicar que el campo es requerido.	Resultado obtenido: se genera el certificado y es enviado vía correo electrónico a los autores y coautores correspondientes.	Resultado obtenido: se generan los certificados y son enviados a todos los autores y coautores de dichos trabajos.
Funcionalidad: Generar certificados de trabajos		
Entrada: vacío	Entrada: se selecciona un trabajo	Entrada: Seleccionar todos trabajos
Resultado esperado: se muestra un mensaje para indicar que el campo es requerido.	Resultado esperado: se genera el certificado y es enviado vía correo electrónico a los involucrados en dicho trabajo	Resultado esperado: se generan los certificados y son enviados a todos los autores y coautores de dichos trabajos.
Resultado obtenido: se muestra un mensaje para indicar que el campo es requerido.	Resultado obtenido: se genera el certificado y es enviado vía correo electrónico a los involucrados en dicho trabajo	Resultado obtenido: se generan los certificados y son enviados a todos los autores y coautores de dichos trabajos.
Funcionalidad: Generar certificados de comisión logística		
Entrada: sin colocar nombres y correos electrónicos.	Entrada: se coloca únicamente el nombre del usuario y no el correo electrónico.	Entrada: se introducen nombres y correos electrónicos validos
Resultado esperado: se muestra un mensaje para ingresar al menos un nombre y correo electrónico.	Resultado esperado: se muestra un mensaje para ingresar al menos un nombre y correo electrónico.	Resultado esperado: se generan los certificados y son enviados vía correo electrónico con los datos correctos.
Resultado obtenido: se muestra un mensaje para ingresar al menos un nombre y correo electrónico.	Resultado obtenido: se muestra un mensaje para ingresar al menos un nombre y correo electrónico.	Resultado obtenido: se generan los certificados y son enviados vía correo electrónico con los datos correctos.

En otro orden de ideas, después de la realización de este proyecto se pudo notar la mejora que significa utilización del Sistema de Gestión de Arbitraje de Artículos de Investigación para AsoVAC. Esta aplicación cambia el enfoque de los actores sobre la convención anual de AsoVAC y el proceso de revisión y postulación de resúmenes de trabajos libres, debido a que se realiza de manera más sencilla, se ahorran recursos, y se organiza el flujo de las actividades a realizar dentro del marco del evento.

Por otra parte, este sistema mejora la comunicación necesaria para el cumplimiento del objetivo, y esta se realiza netamente a través de la aplicación con el fin de que los actores se involucren en la revisión, y arbitraje de los resúmenes de trabajos libres. Otro punto importante y de mejora en el desempeño del proceso es que se mantiene informado a los autores y coautores de las actividades que se van realizando para que estos tengan conocimiento de las etapas por las que va pasando el proceso.

Además es importante tomar en cuenta que con la implementación de este sistema aumentan las probabilidades de administrar de forma más eficiente todos los foros, mesas de trabajo, simposios o eventos que pueda darse cita en la convención de AsoVAC.

Finalmente se puede decir que se logró cumplir el objetivo principal de este proyecto, que era desarrollar una aplicación web que ayude a mejorar, automatizar, y administrar el proceso de revisión por pares para publicaciones científicas y gestión de los eventos de la convención anual de AsoVAC. Esperamos que una vez esté en producción y sea utilizado, se pueda conocer, y aprovechar el verdadero potencial de dicha aplicación, apoyando así al crecimiento de eventos de caracteres científico dentro de las fronteras de Venezuela.

CONCLUSIONES

En la actualidad, muchos gremios de científicos, investigadores como la Asociación Venezolana para el Avance de la Ciencia (AsoVAC) no cuenta con un proceso automatizado o una aplicación web que permite la gestión para la realización del proceso de revisión de artículos científicos. Por esta razón, dicho proceso se realiza mediante la utilización de diferentes herramientas externas a una aplicación centralizada lo cual implica un costo mayor en tiempo y esfuerzo.

Por ende, el proponer y crear una solución enfocada en automatizar el proceso de *peer review*, requirió de grandes conocimientos técnicos, de entender el proceso existente y los puntos clave para ser capaces de realizar esta mejora. Todo esto para agilizar y mejorar los procesos de administración de la Convención Anual de AsoVAC centralizados en un software que cubriera todas sus necesidades y fuera extensible.

Así mismo, fue esencial tomar en cuenta criterios especiales a la hora de implementar las funcionalidades relacionadas con el *peer review*. Es imprescindible seguir estándares existentes y buenas prácticas para asegurar la calidad, la mantenibilidad, usabilidad y utilidad de una aplicación web desarrollada para apoyar este proceso editorial y de transformación digital.

En este orden de ideas, fue importante utilizar tecnologías de desarrollo que han sido probadas y son convenientes por incorporar diversas características deseables para la elaboración de software, ya que estas permiten y facilitan tener un gran soporte a la hora de dudas o problemas. Así como el uso de marcos de trabajo o *frameworks* que permiten tener una buena organización del proyecto, facilitar el proceso de desarrollo mediante un conjunto de primitivas o funciones que facilitan y agilizan el proceso de desarrollo.

De igual manera, lograr cumplir con la creación de este producto de calidad, mediante el uso de las metodologías de desarrollo es primordial. En el caso actual se propuso una combinación de Scrum y Kanban, donde el enfoque fue aprovechar la planificación de un conjunto de entregas donde cada entrega es incremental, y las actividades desarrolladas en cada una de ellas se van integrando con las fases anteriores, siempre asegurando que cada entrega cuente con los requerimientos necesarios a través del monitoreo de tableros de Kanban.

Todo eso para que el producto en cada entrega esté funcional y sea de valor para cliente. Además, nos permitió ir mejorando en muchos aspectos técnicos como implementaciones, pruebas de integración o unitarias, aseguramiento de la calidad del software y tiempos de desarrollo, los cuales a medida que avanza el desarrollo se van perfeccionando.

REFERENCIAS BIBLIOGRÁFICAS

- Acosta, A. (2011). *AgilUs: un método ágil de desarrollo de software que incorpora la usabilidad*. Centro de Ingeniería de Software y Sistemas. Facultad de Ciencias, Universidad Central de Venezuela, Caracas, Venezuela: http://www.ciens.ucv.ve:8080/genasig/sites/interaccion-humanocomp/archivos/243_CLEI_Acosta_Paper.pdf.
- Aprende en línea. (30 de Abril de 2016). Obtenido de <http://aprendeenlinea.udea.edu.co/lms/moodle/mod/page/view.php?id=132758>
- AsoVAC. (2017). *AsoVAC – Caracas*. Obtenido de http://www.asovac.org/wp-content/uploads/2017/12/LXVII_AsoVAC_LibrodeResumenes_verFinal.pdf
- AsoVAC. (2017). *AsoVAC CAPÍTULO ARAGUA*. Obtenido de <http://asovacaragua.org.ve/wp/wp-content/uploads/2017/10/normativa-lxvii.pdf>
- asovacaragua. (2017). *LXVII Convención Anual de la AsoVAC – Asociación Venezolana para el Avance de la Ciencia (AsoVAC) – Capítulo Aragua*. Obtenido de <http://asovacaragua.org.ve/wp/wp-content/uploads/2017/10/normativa-lxvii.pdf>
- asovacaragua.org. (s.f.). *Programas de la AsoVAC – Asociación Venezolana para el Avance de la Ciencia (AsoVAC) – Capítulo Aragua*. Obtenido de http://asovacaragua.org.ve/wp/?page_id=25
- Caivano, R. M. (2009). *Web 2.0 (1ª edición)*. Eduvim.
- Carmen Prado Arreaza, S. V. (s.f.). *Universidad de Granada*. Obtenido de https://www.ugr.es/~setchift/docs/peer_review.pdf
- Consortium, W. W. (2018a). *CCS 2017 Features*. Obtenido de <https://www.w3.org/TR/css-2017/>
- Consortium, W. W. (2018b). *CSS Current Work*. Obtenido de <https://www.w3.org/Style/CSS/current-work>
- Consortium, W. W. (2018c). *W3*. Obtenido de *CSS - Overview*: <https://www.w3.org/Style/CSS/Overview.en.html>
- Consortium, W. W. (2018d). *w3*. Obtenido de https://www.w3.org/standards/techs/css#w3c_all
- Dan Harkey, J. E. (2002). *Cliente/Servidor y Objetos: Guía de supervivencia. 3ra edición*. Universidad Iberoamericana.
- Group, Z. (2004-2017). *Sistema de software de OpenConf Peer-Review, Conference y Abstract Management*. Obtenido de <https://www.openconf.com/>
- Group., T. P. (Julio de (1996-2015)). *PostgreSQL*. Obtenido de <http://www.postgresql.org/about/>
- Holovaty, A. y.-M. (2009). *The definitive guide to django: Web development*. Obtenido de Apress.
- jQuery. (2015). *Write less, do more*. Obtenido de jquery.com: <http://jquery.com/>
- Juan José Martín Frechilla, Y. T. (2005). *UN ARCHIVO PARA LA HISTORIA: ACTA CIENTÍFICA VENEZOLANA 1950-2000*. Caracas.
- Ladrón de Guevara Cervera, M., Hincapié, J., Jackman, J., & Caballero Uribe, C. (2008). *Universidad Autónoma del Estado de México Sistema de Información Científica Redalyc*. Obtenido de <http://www.redalyc.org/pdf/817/81722411.pdf>
- Lario, R. (2013). *Características y ventajas de CSS3*. Obtenido de [dis.um.es](http://dis.um.es/~lopezquesada/documentos/IES_1213/LMSGI/curso/css/css10/Ventajas.html): http://dis.um.es/~lopezquesada/documentos/IES_1213/LMSGI/curso/css/css10/Ventajas.html
- Library, C. ©. (s.f.).

- Mendoza M, L. E. (2015). *Sistema de informacion II*. Obtenido de Universidad Simon Bolivar. Departamento de Procesos y Sistemas: <http://prof.usb.ve/lmendoza/Documentos/PS-6116/Teor%EDa%20PS6116%20Sists.%20Distrib.%20y%20C-S.pdf>
- Mora, S. L. (2002). *Programacion de Aplicaciones Web: Historia, Principios basicos y Clientes Web*. Caracas: Club Universitario.
- Newemage. (2011). *www.newemage.com.mx*. Obtenido de Flash vs JavaScript, ventajas y desventajas.: <http://www.newemage.com.mx/blog/disenio-grafico/flash-vs-javascript-ventajas-y-desventajas>
- Ortega, A. (2015). *AsoVac Carabobo*. Obtenido de <http://asovac.uc.edu.ve/site/index.php/asovac-juvenil>
- Otto, M. T. (2015). *Getbootstrap.com*.
- Pablo, J. (s.f.). *El Conspirador*. Obtenido de <http://www.elconspirador.com/2015/06/08/open-journal-system/>
- Palacio, J. (2008). *Flexibilidad con Scrum*. Obtenido de Safe Creative.
- Presidencia AsoVAC. (2018a). Obtenido de <http://presidencia.asovac.org/acerca-de/programas/convencion-anual/>
- Presidencia AsoVAC. (2018b). Obtenido de <http://presidencia.asovac.org/acerca-de/programas/festival-juvenil-de-la-ciencia/>
- Presidencia Asovac. (2018c). Obtenido de <http://presidencia.asovac.org/acerca-de/programas/festival-juvenil-de-la-ciencia/>
- Prisco, M. C. (1992). *Historia de la AsoVAC*. Obtenido de http://presidencia.asovac.org/acerca-de/historia/?doing_wp_cron=1513637671.9883649349212646484375
- SBPC. (2017). *Quiénes Somos*. Obtenido de <http://www.sbpcnet.org.br/site/es/a-sbpc/quienes-somos.php>
- Schwaber, K. y. (2013). *Scrumguides.org*. Obtenido de La Guía Definitiva de Scrum: Las Reglas del Juego (2a ed.). Recuperado el 20 de junio: <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf>
- Silberschatz A., K. H. (2002). *Fundamentos de Bases de Datos. 4ta ed.* Madrid, España: Mc Graw Hill.
- Simon Fraser University, L. (2014). *Open Journal Systems | Proyecto de conocimiento público*. Obtenido de <https://pkp.sfu.ca/ojs/>
- UCI, E. (s.f.). *E.V.A. UCI Conferencia#5 Modelo Cliente-Servidor. Teleinformatica II*. Obtenido de ecured: http://www.ecured.cu/index.php/Arquitectura_Cliente_Servidor
- Uribe, C. (2008). *Revisión por pares: ¿Qué es y para qué sirve? | Caballero-Uribe | Revista Científica Salud Uninorte*. Obtenido de <http://rcientificas.uninorte.edu.co/index.php/salud/article/viewArticle/1847/6165>
- Valéncia, U. P. (2012). *poli Papers*. Obtenido de <https://polipapers.upv.es/index.php/REDU/help/view/site/topic/000001>
- Veterinaria.org. (1996-2014). *Veterinaria*. Obtenido de <http://veterinaria.org/arbitraje.html#Funcionamiento>
- Wikipedia. (s.f.). Obtenido de https://es.wikipedia.org/wiki/Revisi%C3%B3n_por_pares
- Womack, J. P. (2003). *Lean Thinking: Banish Waste and Create Wealth*. Obtenido de Free Press.

