



#### **Facultad 4**

Centro de Consultoría y Desarrollo de Arquitecturas Empresariales

**Tema:** Digitalización de la recogida de datos en las etapas descongelación, expansión y fermentación del proceso de producción del Ingrediente Farmacéutico Activo en el Centro de Inmunología Molecular.

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas.

**Autores:** Ramón Díaz García

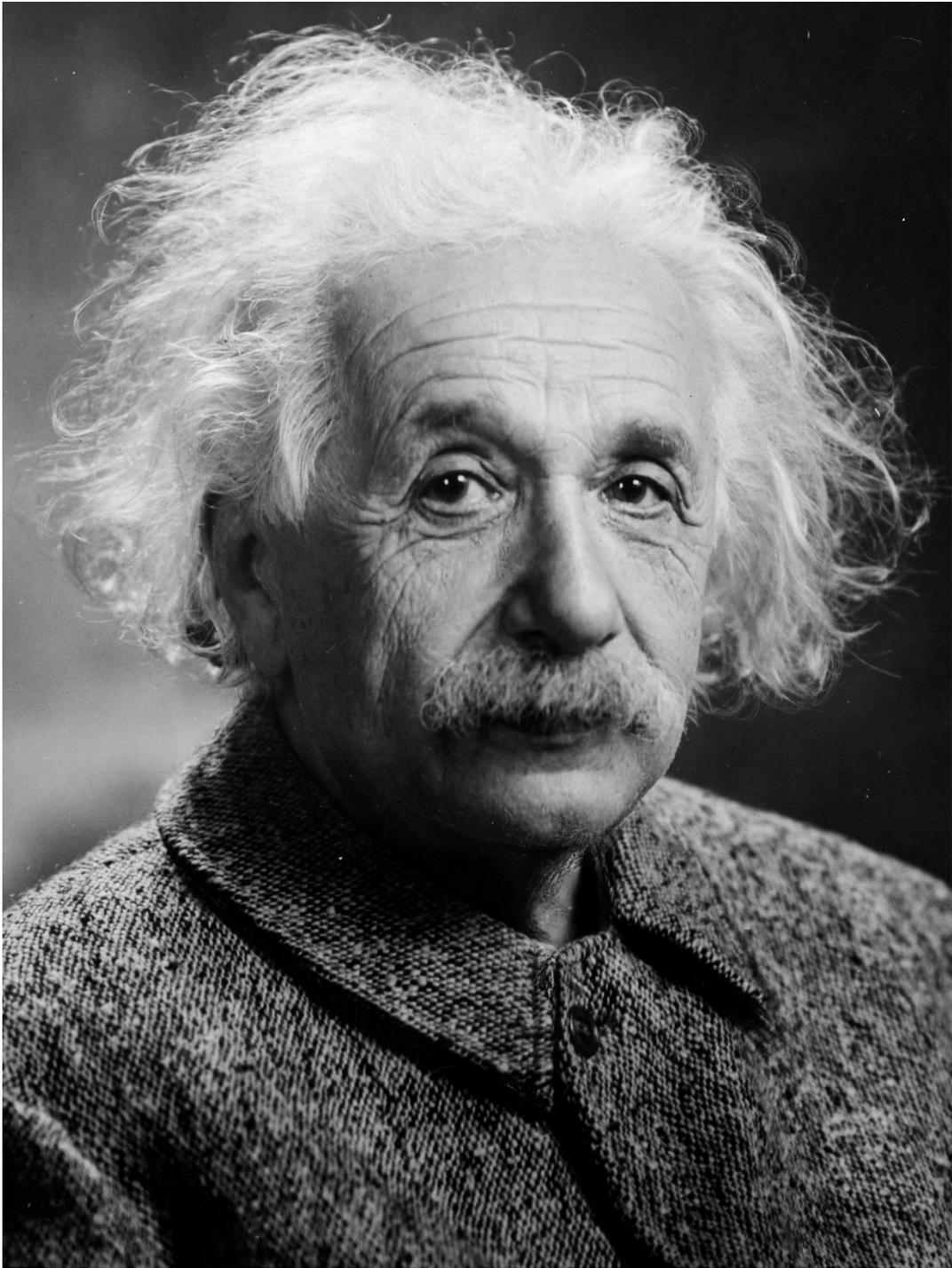
Felix Antonio Sotolongo Barrios

**Tutores:** Ing. Lissette Soto Pelegrín

Ing. Yidier Romero Zaldívar

**La Habana, junio de 2017**

**“Año 59 de la Revolución”**



“Si lo puedes imaginar, lo puedes lograr”

Albert Einstein

### **Dedicatoria de Felix Antonio Sotolongo Barrios:**

El presente Trabajo de Diploma, como símbolo de todo lo que he vivido en estos 5 años, de esfuerzo, dedicación y sacrificio, no solo mío, sino también de mis padres y hermanos que son los que han hecho posible que haya podido hacerme ingeniero, a ellos va dedicado.

## **Dedicatoria de Ramón Díaz García**

Quiero dedicarle este trabajo de diploma a mis padres, mi esposa y toda mi familia que siempre me han ayudado, apoyado, dedicado tiempo y amor en mi vida y sobre todo en mi etapa de estudiante.

## **Agradecimientos de Felix Antonio Sotolongo Barrios**

Quiero agradecer:

.....a esas personas tan especiales en mi vida que son los culpables de todo lo que he logrado, que son los responsables de que haya llegado hasta aquí, que son los que han estado conmigo en cada momento y han confiado en mí. Muchas gracias a mi mamá, mi papá, mi hermana, mi hermano, mis sobrinos, mi cuñado que son los que han estado ahí día a día guiándome hasta llegar a ser lo que soy hoy.

.....a la persona que ha hecho mis días más felices desde que la conocí, que ha estado a mi lado apoyándome cada instante, que me ha dado su amor, su cariño, su amistad. Muchas gracias a mi novia *Ayamzy*.

.....a la persona más insoportable que ojos humanos han visto, que ha sido mi amiga, mi compañera, mi pesadilla. Muchas gracias a *Shirley* por siempre estar ahí.

.....a mis amigos que desde el principio han estado a mi lado en los buenos momentos y en los mejores porque con ustedes siempre me he sentido muy bien. Muchas gracias a *Richard, Flaco, Liosvel, Gustavo, Yasiel*. Gracias por ser de verdad.

.....a todos los que he conocido en mi transecurso por la universidad, mis compañeros de aula, de apartamento, compañeros de todas las fiestas. Muchas gracias.

.....a las personas que han dedicado su tiempo y esfuerzo para que este trabajo de diploma tenga la calidad requerida. Muchas gracias a mis tutores *Lissette* y *Yidier*.

## **Agradecimientos de Ramón Díaz García**

"Las palabras nunca alcanzan cuando lo que hay que decir desborda el alma", tengo la dicha de contar con personas maravillosas que me apoyan sin yo decir una palabra, personas que me brindan su amor, su tiempo, su dedicación y su vida, hoy quiero darles las gracias a todas esas personas que me han ayudado tanto en mi vida personal como profesional especialmente a,

... Mis padres que se han dedicado, sacrificado y esforzado para ayudarme a cumplir esta meta, a convertir un sueño en realidad, dedicando toda su vida a ayudarme y apoyarme en mi vida personal y profesional, por estar pendientes a que cada paso que doy sea seguro y conciso, verdaderamente son los mejores padres del mundo.

... A mi esposa por ayudarme en todo instante, por estar siempre ahí, en los momentos difíciles y en los momentos buenos, por lidiar con mi estado de tensión en esta etapa final de estudiante, por apoyarme y darme aliento cuando más lo necesito, por quererme tanto y brindarme felicidad y amor.

... A mis abuelos por sus bendiciones y ternura, por estar pendiente de mis estudios, mis pruebas y mis calificaciones.

... A mis tíos por sus consejos y cariño, por sus conversaciones y cuentos para motivarme a seguir adelante en lo personal y lo profesional.

... A mis primos por siempre estar presente y pendientes a mis pasos, por sus mensajes de ayuda y el amor que me brindan.

... A mis suegros que los quiero como a unos padres, gracias por su apoyo incondicional.

... A mi amigo Iván, que a pesar de no tener la misma sangre lo considero como un hermano.

...A mis amigos del aula y de la UCI, no voy a especificar nombres para que no se me quede nadie fuera y a los que están hoy aquí conmigo.

... A mi amigo, compañero y hermano Rodolfo.

...A mis amigos del barrio.

...A mis profesores en toda mi etapa de estudiante y en especial a los de la universidad.

...A mi compañero de tesis Félix.

...Al oponente y al tribunal de tesis.

...A mis tutores Yidier y Lissette por luchar contra mi inexperiencia y ayudarme a ser más capaz.

...A toda mi familia en general.... sinceramente gracias por todo.

## Declaración de autoría

Declaramos ser los autores del trabajo de diploma “Digitalización de la recogida de datos en las etapas descongelación, expansión y fermentación del proceso de producción del Ingrediente Farmacéutico Activo en el Centro de Inmunología Molecular”, y otorgamos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo, así como, el derecho de uso en su beneficio.

Para que así conste firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

Firma del autor

Ramón Díaz García

---

Firma del autor

Felix Antonio Sotolongo Barrios

---

Firma del tutor

Ing. Yidier Romero Zaldívar

---

Firma del tutor

Ing. Lissette Soto Pelegrín

## Resumen

La gestión de la información, específicamente el almacenamiento, control y disponibilidad, ha devenido en una ocupación de interés para el desarrollo de la sociedad mundial. Lo cual se ha convertido en un proceso de perfeccionamiento constante, para llevar a cabo la planificación, el seguimiento, la dirección y la toma de decisiones, contribuyendo esto al buen desarrollo y funcionamiento de las entidades en general.

En el Centro de Inmunología Molecular (CIM) se maneja un volumen de información considerable y la información referente a varios procesos es gestionada de forma manual en registros físicos, por lo que se hace necesaria la digitalización de dichos registros para lograr la obtención de resultados eficaces y eficientes a la hora de analizar los datos recogidos.

A partir de esta situación el trabajo de diploma que se propone permite la digitalización de la recogida de información referente a tres etapas del proceso de producción del Ingrediente Farmacéutico Activo (IFA) en el CIM, basando el modelo de datos desarrollado en los estándares que propone la Asociación de Soluciones Empresariales para la Manufactura (MESA), lo que contribuye a la futura implementación de un Sistema de Ejecución de Manufacturas (MES) en el CIM. Dicho sistema debe ser realizado por especialistas asociados a ese centro productivo, debido a que como su equipamiento es heterogéneo los fabricantes no brindan el Sistema de Ejecución de Manufacturas.

**Palabras clave:** digitalización, CIM, IFA, MES, MESA

## **Abstract**

Information management, specifically storage, control and availability, has become an occupation of interest for the development of global society. This has become a process of constant improvement, to carry out the planning, monitoring, management and decision making, contributing this to the good development and operation of the entities in general.

In the Center of Molecular Immunology (CIM), a considerable amount of information is handled and many processes are handled manually in physical registers, so it becomes necessary to digitize these records in order to obtain effective and efficient results when analyzing the collected data.

Based on this situation, the proposed thesis allows the digitization of information on three stages of the Active Pharmaceutical Ingredient production process in the CIM, based on the data model developed in the standards proposed by the Manufacturing Enterprise Solutions Association (MESA), which contributes to the future implementation of a Manufacturing Execution System (MES) in the CIM. This system must be carried out by specialists associated with this production center, because as its equipment is heterogeneous manufacturers do not provide the Manufacturing Execution System.

**Keywords:** digitization, CIM, IFA, MES, MESA

# Índice

Introducción.....	1
Capítulo 1: Fundamentación teórica.....	5
Introducción.....	5
1.1 Sistema de Ejecución de Manufactura.....	5
1.2 Estándares asociados al MES.....	7
1.2.1 MESA.....	8
1.3 Funcionalidad del MES.....	10
1.4 Ejemplos del MES en la biofarmacia.....	12
1.5 Tecnologías.....	13
1.6 Herramientas.....	19
1.7 Metodología de desarrollo de <i>software</i> .....	22
Conclusiones parciales.....	26
Capítulo 2: Propuesta de solución.....	27
Introducción.....	27
2.1 Proceso de producción del IFA en el CIM.....	27
2.2 Propuesta de solución.....	28
2.3 Requisitos del sistema.....	28
2.3.1 Requisitos funcionales.....	28
2.3.2 Requisitos no funcionales.....	28
2.3.3 Historias de Usuario.....	29
2.3.4 Estimación de esfuerzos e iteraciones por Historia de Usuario.....	34
2.4 Descripción de la Arquitectura.....	35
2.4.1 Patrón Arquitectónico.....	35
2.4.2 Patrones de diseño.....	36
2.5 Diagrama de Despliegue.....	37
Conclusiones Parciales.....	37

Capítulo 3: Validación y pruebas ..... 39

    Introducción ..... 39

    3.1 Pruebas a los sistemas de software..... 39

    3.2 Diseño de casos de pruebas al sistema..... 44

    3.3 Resultados de las pruebas..... 50

    Conclusiones parciales ..... 53

Conclusiones generales ..... 54

Recomendaciones..... 55

Bibliografía ..... 56

Anexos ..... 60

# Introducción

Actualmente las empresas de cualquier tamaño en cualquier parte del mundo deben preocuparse por incrementar su competitividad, desarrollar medidas y soluciones que respondan a los cambios del mercado rápida y correctamente, y ayuden a adoptar procesos de producción en tiempo real. Una herramienta crítica para lograrlo, es un Sistema de Ejecución de Manufactura (MES, acrónimo en inglés de *Manufacturing Execution System*) que soporte una ejecución oportuna.

El MES permite el control de la producción en tiempo real, exhaustiva recopilación y reporte de datos. Estos sistemas de ejecución de manufactura cierran la brecha entre los sistemas empresariales (ERP, acrónimo en inglés de *Enterprise Resource Planning*) y el equipo de control (SCADA, acrónimo en inglés de *Supervisory Control And Data Acquisition*). Una solución de un MES efectiva permite la visibilidad y el control de alto nivel en las áreas de producción, mantenimiento, inventario y calidad.[1]

Los MES se basan en definiciones, estándares y consistentes terminologías, modelos y conceptos creados por organismos como la Asociación de Soluciones Empresariales para la Manufactura (MESA, acrónimo en inglés de *Manufacturing Enterprise Solutions Association*) el cual incluye en sus pautas cumplir con los aspectos de las normas ISA-95 e ISA-88.

Las empresas dedicadas a la realización de Sistemas de Ejecución de Manufactura no los producen adaptables a cualquier entidad, sino al equipamiento en específico de un mismo fabricante, razón por la cual para las empresas cubanas se dificulta la obtención de este tipo de herramientas, debido a que, por los efectos del bloqueo, se importan los equipos que estén disponibles para Cuba, independientemente de quien los haya producido.

Una de las instituciones cubanas cuyo equipamiento no es homogéneo es el Centro de Inmunología Molecular (CIM), institución biotecnológica de ciclo cerrado (investigación-desarrollo, producción, comercialización), que a partir del cultivo de células superiores orienta su investigación al desarrollo y fabricación de productos al tratamiento del cáncer y otras enfermedades autoinmunes.[2] Esta entidad es un centro de alta tecnología con importantes resultados tanto científicos como productivos y comerciales. Sin embargo, tiene el reto de crecer para incrementar los ingresos por exportación y aumentar el aprovechamiento de sus capacidades productivas para satisfacer las necesidades de medicamentos del país y garantizar la diversificación de productos y mercados. Este reto requiere la evolución organizativa y tecnológica del CIM, donde el uso extensivo de las tecnologías informáticas se convertirá en un elemento imprescindible para lograr la gestión exitosa de la organización.

En la Universidad de las Ciencias Informáticas (UCI) se encuentra el Centro de Consultoría y Desarrollo de Arquitecturas Empresariales (CDAE), que ha contribuido a la informatización de entidades como el CIM, que, debido a esta colaboración, ha logrado informatizar una parte de sus procesos internos, sin embargo, quedan oportunidades para la informatización como el proceso de producción del Ingrediente Farmacéutico Activo (IFA) compuesto utilizado para crear medicamento.

En la actualidad el control del proceso de producción del IFA en el CIM se basa en gran medida de los registros físicos (anexo 1) que se obtienen de forma manual, lo que provoca que los datos no estén disponibles en tiempo real para ser consultados en el momento que se necesite. Lo expuesto, limita el análisis posterior de la información recogida y se necesita emplear gran cantidad de tiempo y personas para revisar las fallas ocurridas, realizar alertas tempranas o tomar decisiones oportunas, en dependencia del comportamiento del proceso.

Teniendo en cuenta lo antes expuesto se plantea como **problema a resolver** la siguiente interrogante: ¿cómo mejorar la gestión de la información asociada a las etapas descongelación, expansión y fermentación del proceso de producción del IFA en el CIM?

La presente investigación tiene como **objeto de estudio** el proceso de producción del IFA en el CIM.

Para dar solución a la problemática planteada se define como **objetivo general** de la investigación el siguiente: desarrollar un sistema informático que contribuya a mejorar la gestión de la información asociada al proceso de producción del IFA en el CIM.

Como **campo de acción** el registro maestro de lotes para las etapas de descongelación, expansión y fermentación.

Para dar cumplimiento al objetivo propuesto se definen las siguientes **tareas de investigación**:

- Elaboración del marco conceptual para precisar los principales conceptos que se emplean en la investigación.
- Estudio del estado del arte de soluciones empresariales para la manufactura.
- Descripción de las herramientas, tecnologías y metodología de desarrollo.
- Definición de los requisitos funcionales y no funcionales del sistema.
- Descripción de la arquitectura a utilizar en la implementación de las funcionalidades.
- Implementación de la solución propuesta.
- Valoración de los resultados obtenidos.

Los **métodos científicos** son empleados para explicar fenómenos, establecer relaciones entre los hechos, enunciar leyes que expliquen los fenómenos físicos del mundo y permitan obtener,

con estos conocimientos, aplicaciones útiles al hombre[3]. En la presente investigación se hace uso de algunos métodos teóricos y empíricos de investigación, entre los que se encuentran:

### **Métodos teóricos**

- **Análisis y síntesis:** este método se utilizó para comprender el entorno en el que se desarrolla el proceso de producción del IFA en el CIM, para estudiar los contenidos relacionados con los sistemas de ejecución de manufacturas y para diseñar la propuesta de solución desarrollada.
  - o **Análisis bibliográfico:** se utilizó para abordar los aspectos conceptuales y metodológicos de la investigación.
  - o **Análisis gráfico:** se utilizó para apoyar la selección de la biblioteca PrimeFaces utilizadas en el desarrollo.
- **Modelado:** se utilizó para elaborar los diagramas en las fases de análisis y diseño del desarrollo del sistema.

### **Métodos empíricos:**

- **Observación:** se utilizó este método para conocer de forma práctica y verificar el funcionamiento del proceso de producción del IFA en el CIM, se observaron principalmente los subprocesos de descongelación, expansión y fermentación.

En este trabajo se utilizó como **técnica** de recopilación de información la **entrevista** al cliente, realizada a especialistas del departamento de informática de la dirección logística del CIM y a operarios de las plantas de producción del IFA, futuros usuarios del sistema.

### **Resultados esperados:**

- Modelo de datos para la recolección de información relacionada al proceso de producción del IFA en el CIM.
- Sistema informático para la digitalización de la recogida de datos en las etapas descongelación, expansión y fermentación del proceso de producción del IFA en el CIM.

A continuación, se describe de forma general la estructura del documento; la tesis está dividida en tres capítulos:

En el **Capítulo 1: Fundamentación teórica**, se define el marco conceptual donde se especifican los principales conceptos que se emplean en la investigación, se realiza un estudio del estado del arte sobre los estándares e indicadores definidos para la ejecución de manufacturas y sobre los sistemas de ejecución de manufactura y se realiza una descripción de las tecnologías, herramientas y metodología de desarrollo seleccionadas.

El **Capítulo 2: Propuesta de solución**, presenta la descripción del proceso de producción del IFA en el CIM, los requisitos funcionales y no funcionales del sistema desarrollado y la descripción de la arquitectura utilizada en la implementación de las funcionalidades.

Por su parte, en el **Capítulo 3: Validación y pruebas**, se describen las pruebas realizadas a las diferentes funcionalidades del sistema y sus resultados y se evidencian resultados de pruebas funcionales realizadas por el cliente al sistema.

# Capítulo 1: Fundamentación teórica

## Introducción

En el presente capítulo se abordan las principales características y conceptos asociados al Sistema de Ejecución de Manufactura. Para ello se presentan aspectos relevantes tratados por la Asociación de Soluciones Empresariales para la Manufactura, quien tiene una propuesta de estándares para la gestión de información de todo el proceso de manufactura. Se describen las principales funcionalidades de un MES y se evidencian ejemplos del MES en la industria biofarmacéutica. Además, se describen las tecnologías, herramientas y metodologías que se emplean en la construcción de esta solución.

### 1.1 Sistema de Ejecución de Manufactura

Para comprender mejor el significado de los sistemas de ejecución de manufactura se presentan a continuación los significados de cada uno de los términos que lo forman por separado, según la Real Academia de la Lengua Española

- Sistema: m. Conjunto de reglas o principios sobre una materia racionalmente enlazados entre sí. / m. Conjunto de cosas que relacionadas entre sí ordenadamente contribuyen a determinado objeto[4]
- Ejecución: f. Acción y efecto de ejecutar. / loc. verb. Ejecutar, llevar a la práctica. [5]
- Ejecutar: tr. Poner por obra algo. Desempeñar con arte y facilidad algo.[6]
- Manufactura: f. Obra hecha a mano o con auxilio de máquina.[7]

Los Sistemas de Ejecución de Manufactura son un paquete de herramientas que permiten gestionar los diferentes procesos de un sistema productivo.[8] El MES proporciona la información necesaria para optimizar los procesos productivos desde el lanzamiento de la orden de fabricación hasta el producto acabado, aportando beneficios específicos que se focalizan en las actividades de producción. Los servicios del MES ayudan a acortar la distancia entre el ERP y el SCADA. Permite establecer normativas comunes y buenas prácticas en todas las plantas e instalaciones, a la vez que minimiza los costos, los tiempos de entrega e inventarios.[9]

El término MES empezó a principios de 1990. Su punto clave era ofrecer mejor control y visibilidad a través de la recolección y análisis de datos en tiempo real. El núcleo fuerte del MES radica en la interfaz entre el piso de planta y la gestión.[10]

El MES ha transformado las unidades de manufactura convencional en modernos centros de servicios y se enfoca más en mejorar las capacidades del proceso que las capacidades de

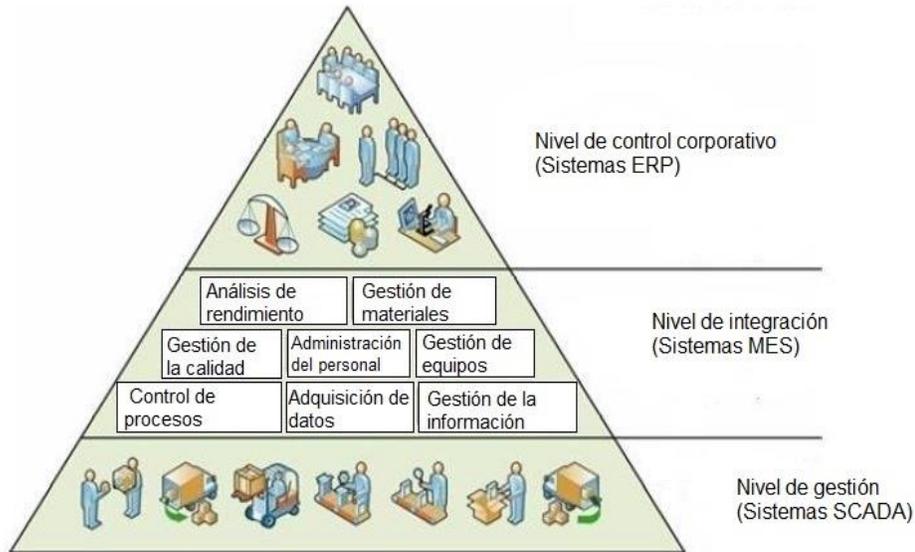
producción. Ha ayudado a las empresas a alcanzar alta productividad y reducir los costos en corto tiempo.[8]

El MES dirige y monitoriza los procesos de producción en la planta, incluyendo el trabajo manual y automático de informes. Un MES puede incluir uno o más enlaces a órdenes de trabajo, recepción de mercancías, transporte, control de calidad, mantenimiento, planificación y otras tareas relacionadas.

Un MES eficaz implica directamente al personal de producción. No se trata solamente de tener a trabajadores que supervisen los procesos de producción y realicen tareas repetitivas, sino de proporcionarles herramientas que hagan hincapié en sus habilidades y rendimiento. El personal de producción puede consultar estos indicadores en tiempo real a través de terminales fáciles de usar que les permiten evaluar la eficacia. Se puede saber al momento si se están cumpliendo los objetivos, y resolver errores o incidencias que paralicen la producción o provoquen fallos de calidad.[11]

El MES se ha diseñado para ser más que un programa informático. Es una solución que perfecciona los procesos de fabricación con un enfoque más general que fomenta la comunicación y la optimización operativa a partir de la información obtenida gracias a la integración de la persona. El MES desempeña un papel fundamental en el centro de la operación de fabricación, conectando el personal del taller, la automatización del equipo, la logística, las ventas y la planificación. Su capacidad para alinear estas diferentes funciones y brindar visibilidad a los principales interesados impulsa el desempeño y el cumplimiento en toda la organización. [8]

La integración de información desde el nivel de gestión hasta el piso de planta cubre una gran cantidad de áreas de conocimiento que varían principalmente del enfoque propuesto, la profundización requerida y las características de aplicación. Existen varios modelos de integración, en la Figura 1.1, se presenta uno básico, en el cual, se pueden identificar claramente tres niveles: Nivel de fabricación, correspondiente a las unidades productivas, estaciones de trabajo o "piso de planta"; Nivel de control de manufactura atribuido a los Sistemas de Ejecución de Manufactura; y Nivel de control corporativo.[12]



**Figura 1.1:** Modelo de integración corporativa. Adaptación de [12].

Los módulos o funciones del MES se pueden implementar de forma independiente con base en las necesidades de la compañía. Las principales motivaciones de la implementación de un MES son:[8]

- Incremento en la confiabilidad y oportunidad de la información en los procesos.
- Medición del rendimiento de las unidades de producción.
- Oportuna gestión de los inventarios.
- Gestión de calidad en tiempo real.
- Rápida respuesta ante dificultades presentadas en el proceso productivo.
- Apoyo en la gestión de mantenimiento.
- Trazabilidad de los productos.

## 1.2 Estándares asociados al MES

Las ventajas de la aplicación de estándares internacionales de integración empresarial en los procesos de producción agroindustriales se verán reflejados en los diferentes niveles de la empresa, comenzando por la fabricación del producto hasta llegar al ámbito del sistema de ejecución de manufactura y de planeación de recursos empresariales. Consiste en el establecimiento de normas a las que debe ajustarse la información, los procesos de intercambio de esta y la interoperación de los sistemas que deben manejarla.

### 1.2.1 MESA

La Asociación de Soluciones Empresariales para la Manufactura es una entidad industrial enfocada en mejorar los procesos comerciales en el sector de la producción a través de la optimización de aplicaciones existentes e introducción de sistemas de información innovadores. La MESA fue la primera organización en consagrarse al tema del MES. [13]

Existen varios modelos de referencia de un sistema MES creados por organizaciones como la Asociación de Soluciones Empresariales para la Manufactura. Uno de los modelos más aceptados es el propuesto por la asociación MESA representado en la Figura 1.2, el cual ha sido substancialmente revisado desde su publicación en 1997 y actualmente tiene un enfoque corporativo que no solo tiene presente las operaciones de producción, sino que también incluye las operaciones de negocio y las iniciativas estratégicas. [14]

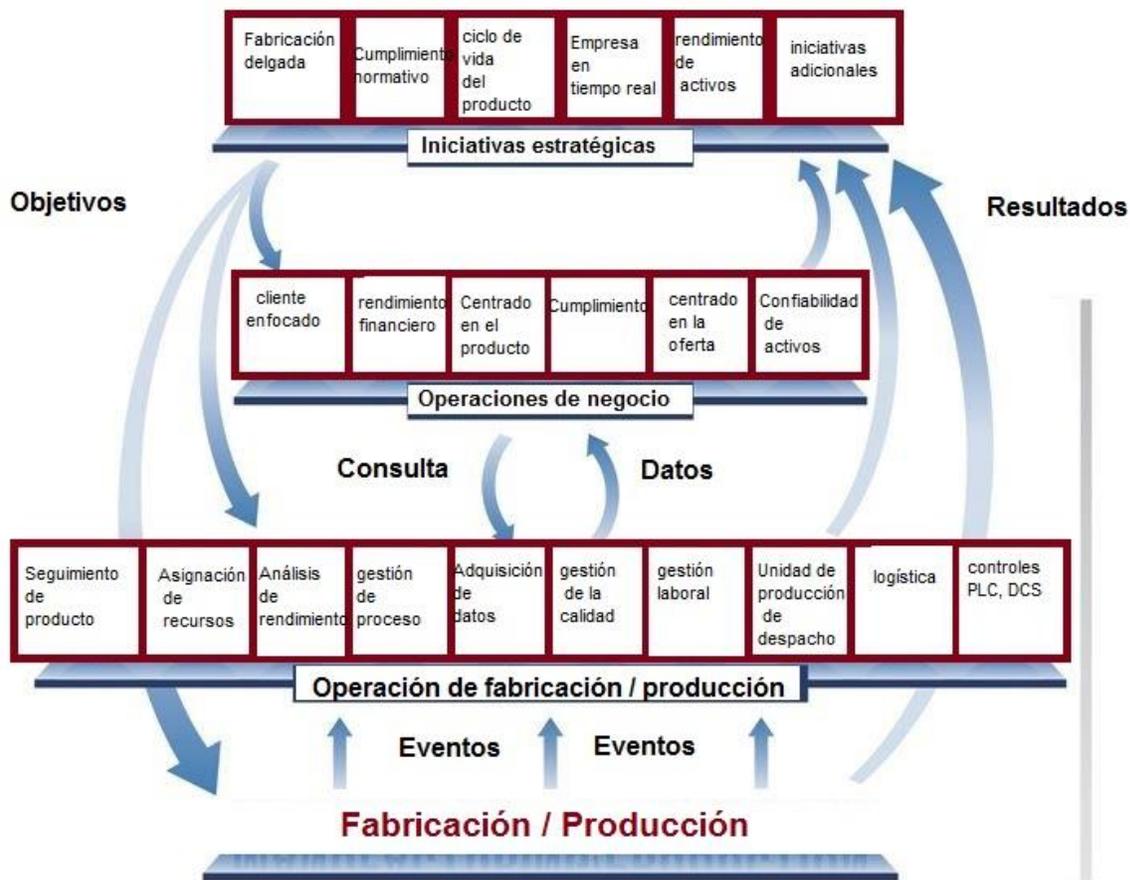


Figura 1.2: Modelo de MES propuesto por MESA. Fuente: tomado de [14]

La implementación de un MES varía según el tipo de industria, aunque también puede variar entre diferentes plantas de una misma organización, esto se debe a que el énfasis en las funciones puede ser diferente de una unidad a otra.

### **Directrices de MESA**

MESA abarca un conjunto de información en las temáticas: gestión de la producción, creación de productos, gestión de la calidad, y optimización de la producción para las plantas de manufactura y proveedores de soluciones. La integración de los sistemas orientados a la producción tiene para MESA particular importancia.[13]

### **Según MESA, los 11 grupos de funcionalidades de un MES son los siguientes:[13]**

1. Buena planificación de flujo de trabajo. Este grupo prevé una óptima secuencia de planificación con respecto a las condiciones básicas pertinentes basado en los recursos disponibles.
2. Dirección de recurso con mantenimiento de estado. La dirección y supervisión de los recursos relevantes (el personal, las máquinas y las herramientas).
3. Control de unidad de producción. Control de flujo de unidades de producción basados en los órdenes y los lotes.
4. Control de la información. Toda la información relevante para el proceso de producción (diseños, prueba de especificaciones, medioambiente, requerimientos de conformidad, instrucciones de seguridad).
5. Registro de datos operativos. Registro automático o manual de todos los datos operativos relacionados con la unidad de producción.
6. Gestión del personal. Registro de horas de trabajo del personal y potencial para cambiar en caso de ausencia y vacaciones.
7. Gestión de la calidad. Análisis de la medición relacionada con la producción, datos en tiempo real para asegurar la calidad del producto y ser capaz de identificar los problemas y los puntos débiles a su debido tiempo.
8. Gestión de procesos. Seguimiento de la producción real incluyendo funciones de gestión de alarmas.
9. Gestión de mantenimiento. Registro del uso de materiales y horas de uso con el fin de iniciar periódicamente tareas de mantenimiento preventivo.
10. Trazabilidad del lote. Registro de todos los datos relacionados con la producción toda la cadena de producción para garantizar que cada producto fabricado es rastreable.
11. Análisis del rendimiento. Seguimiento del tamaño, tiempo, interrupciones, contadores de piezas de la fabricación, esto permite una simple evaluación de la eficiencia de la producción.

## **ISA-88 e ISA-95**

ISA-88 es reconocido mundialmente como el estándar para la industria de procesamiento por lotes. El Lenguaje de Marcado por Lotes (BatchML, acrónimo en inglés de *Batch Markup Language*) proporciona un conjunto de definiciones de tipo y elemento Lenguaje de Marcado Extensible (XML, acrónimo en inglés de *eXtensible Markup Language*) que pueden utilizarse en parte o en su totalidad para el registro de producción y los datos del equipo. Cualquier empresa puede utilizar BatchML libre de impuestos, siempre que se referencie a MESA.

ISA-95 es una norma internacional para el desarrollo de una interfaz automatizada entre la empresa y los sistemas de control. Esta norma ha sido desarrollada para todos los fabricantes de MES. Se desarrolló para ser aplicado en todas las industrias, y en todo tipo de procesos, como procesos discontinuos, procesos continuos y repetitivos.

Los objetivos del ISA-95 son: [12]

- Proporcionar una terminología consistente que sea una base para las comunicaciones de proveedores y fabricantes.
- Proporcionar modelos de información consistentes que sirvan de base para aclarar la funcionalidad de la aplicación y cómo se va a utilizar la información. [15]

B2MML, (acrónimo en inglés de *Business To Manufacturing Markup Language*) es una implementación XML de la familia de estándares ISA-95 que consiste en un conjunto de esquemas XML que implementan los modelos de datos en el estándar ISA-95. Las empresas interesadas en seguir ISA-95 para proyectos de integración pueden utilizar B2MML para integrar sistemas empresariales como ERP y sistemas de gestión de la cadena de suministro con sistemas de fabricación tales como sistemas de control y sistemas de ejecución de fabricación. B2MML es una implementación completa de ISA-95. Cualquier empresa puede utilizar B2MML libre de impuestos, siempre que se referencie a MESA. [15]

## **1.3 Funcionalidad del MES**

Las funcionalidades de un MES totalmente implementado ofrecen una fuerte ventaja y una clara comprensión. El MES proporciona una fácil identificación de sus numerosos beneficios. La MESA Internacional, representa a los profesionales que trabajan alrededor de los sistemas de ejecución de manufactura.[10]

Un MES proporciona la habilidad para controlar máquinas, materiales, documentos y habilidades laborales entre otros recursos necesarios para una operación a realizar como historia de recursos, configuración actual, disponibilidad y otra información crítica simultáneamente disponible para el

técnico en la planta y el gerente. El control de esta información proporciona actualizaciones de estado en tiempo real y control de procesos.[10]

La secuenciación del trabajo basada en prioridad, atributos y requerimientos de recursos busca minimizar el tiempo de preparación y maximizar el flujo a través del sistema de producción. Un cálculo preciso del tiempo empleado se compila a partir de cada operación independiente incluso con la complejidad añadida de las operaciones superpuestas o paralelas. La característica de programación de un MES también prevé la carga de nivel de mano de obra y el equipo. [10]

El MES opera un despacho en tiempo real para todas las operaciones de producción, controlando cuidadosamente la cantidad del producto, reduciendo los trabajos en curso debido al ciclo de retroalimentación en tiempo real. [10]

Una de las fortalezas más auténticas del MES radica en el control de documentos. La implementación del MES elimina el uso de métodos de control de documentos basados en papel. Esta funcionalidad proporciona beneficios de gran importancia en la empresa. Todos los usuarios del MES tienen acceso a los documentos requeridos. El técnico tiene acceso a toda la información relacionada con la unidad de producción incluyendo instrucciones de trabajo, procedimientos operativos, órdenes de cambio de ingeniería, lista de materiales, historia y otra información obligatoria que antes del MES habría sido difícil para obtener y mantener con la recogida de datos en registros físicos. Además, las planillas digitalizadas con información de producción y posible firma por parte de los técnicos son controlados informáticamente por el MES. Esto agrega seguridad y fiabilidad en el circuito de retroalimentación de la producción en el nivel de fabricación. [10]

La recolección de datos establece la significación en el MES. Esta faceta, que se presenta como una función, proporciona todos los datos relacionados con la producción a manos de la dirección en formatos que proporcionan métricas valiosas y una visión de las características de la producción. Estos datos pueden ser recogidos de forma automática desde equipos inteligentes o manualmente a partir de formas de interfaz humana, estos datos son archivados para su fácil recuperación y cálculos históricos.[10]

El MES ofrece un análisis en tiempo real de la calidad. Los datos recolectados de las operaciones de fabricación pueden ser sintetizados y exhibidos en medios fácilmente legibles para identificar cuestiones que requieren medidas de seguimiento. Los sistemas inteligentes de ejecución de manufactura ofrecen la posibilidad de examinar registros históricos identificando la causa raíz de los problemas. Los sistemas completamente implementados incluyen control de proceso estadístico y control de la calidad de proveedores. Bajo la función de recopilación de datos,

el MES, controla la calidad y tiene la capacidad de realizar cualquier inspección y firmar electrónicamente en formularios de entrada manuales. La capacidad de identificar los posibles defectos de calidad y alertar en tiempo real, representa una reducción en el trabajo y un aumento en la satisfacción del cliente debido a un producto más confiable. [10]

La gestión de procesos es un subproducto de la recopilación de datos de recursos, mano de obra y equipo. MES supervisa la producción mientras se ajusta para maximizar las actividades de producción de forma automática o mediante el suministro de información para tomar una decisión determinada. [10]

Proporciona medios de seguimiento para el mantenimiento basado en las operaciones. Identificados por horas de funcionamiento y programación del mantenimiento. Todos los problemas de mantenimiento se registran en la historia como referencia, para ayudar en el diagnóstico de problemas actuales. El seguimiento de productos se manifiesta como otra colaboración de la recogida de datos. A medida que el producto se mueve a través de la producción, se registra información. Estos datos registrados proporcionan rastreabilidad e información histórica sobre la creación del producto final y de todos sus componentes. [10]

El análisis de rendimiento completa el bucle de retroalimentación. La información sobre los patrones de producción que se recogen en la actualidad son comparados con los resultados históricos. Los análisis proporcionados por el MES identifican claramente las fortalezas y debilidades, y ayudan en la búsqueda de la mejora continua. [10]

Los puntos fuertes del MES y la discrepancia entre un sistema accionado por registros físicos y uno controlado por un Sistema de Ejecución de Manufactura son evidentes. Numerosas empresas han implementado este sistema y han comprobado los beneficios. [10]

## **1.4 Ejemplos del MES en la biofarmacia**

A continuación, se presenta algunas de las empresas que desarrollan y comercializan MES en el mundo, vinculadas con la industria biofarmacéutica.

### **Siemens**

Ofrece un sistema, hecho a medida, para acceder a toda su información de procesos en tiempo real. El sistema de ejecución de manufactura, SIMATIC IT, es un MES sofisticado y altamente escalable que cumple con la norma ISA-95. Este sistema de ejecución de manufactura tiene múltiples funciones y permite una utilización óptima con alta calidad de producción. [17]

## **Schneider Electric**

Es una empresa dedicada a la creación de MES, su producto Plantstruxure, tiene como tarea principal el seguimiento de la información de lotes de producción. Cuenta con un MES vinculado a la industria biofarmacéutica el cual tiene acceso en tiempo real a la planta y a la información comercial.[18]

## **Werum IT Solutions**

La empresa líder mundial en cuanto a biotecnología es Werum IT Solutions, la cual es la mayor proveedora de sistemas de ejecución de manufacturas y de fabricación de soluciones para las industrias biofarmacéutica.

El PAS-X es el MES líder en el mercado para las industrias farmacéuticas. Este MES soporta todos los segmentos principales de la industria biofarmacéutica, la cual cubre todo el ciclo de vida del producto desde el desarrollo del proceso hasta la producción y envasado comercial.[19]

## **1.5 Tecnologías**

En este epígrafe se exponen las tecnologías que son empleadas en el desarrollo del proyecto, así como sus principales características y funciones. Estas tecnologías fueron especificadas por el cliente, la plataforma de desarrollo utilizada es *Java EE7*, como *frameworks Java Server Faces* en su versión 2.2, con la biblioteca *PrimeFaces* 5.0.

### **Plataforma:**

#### ***Java EE7***

Una plataforma Java provee un entorno peculiar en el cual podemos encontrar aplicaciones de cualquier tipo programadas en lenguaje Java.

Hoy en día podemos identificar tres plataformas Java: [20]

- 1.Plataforma Java, Edición Estándar (Java SE).
- 2.Plataforma Java, Edición Empresarial (Java EE).
- 3.Plataforma Java, Edición Micro (Java ME).

Cada una de las plataformas Java, contiene una máquina virtual de Java y una Interfaz para la Programación de Aplicaciones (API, acrónimo en inglés de Application Programming Interface), esto permite que las aplicaciones escritas para la plataforma, puedan ser ejecutadas en cualquier sistema compatible, con todas las ventajas del lenguaje de programación Java, independencia de la plataforma, estabilidad, potencia, facilidad de desarrollo y seguridad. La API de la plataforma Java SE define las funcionalidades fundamentales del lenguaje de programación Java y está

formada por una máquina virtual, herramientas de desarrollo, despliegue de tecnologías y otras bibliotecas de clases y *toolkits* usados para el desarrollo de aplicaciones Java. La plataforma Java EE ha sido construida encima de la plataforma Java SE y está formada igualmente por una API y un entorno de ejecución, pero a diferencia de Java SE, está enfocada al desarrollo y ejecución de aplicaciones de gran envergadura, multicapa, escalable, fiable y en una red segura, la plataforma Java EE está especialmente diseñada para desarrollar aplicaciones empresariales. [21]

La plataforma Java EE representa un estándar para la implementación y despliegue de aplicaciones empresariales. Fue diseñada a través de un proceso abierto y se entablaron conversaciones con un amplio abanico de empresas, lo que permitió que se cubriesen la gran mayoría de los requisitos de las aplicaciones empresariales, soporta una amplia variedad de protocolos para interactuar con una aplicación Java EE que se esté ejecutando. Las aplicaciones Java EE permiten el acceso a un subconjunto de los servicios de Java EE disponibles en los contenedores web y *Enterprise Bean* del servidor.[22]

A continuación, se describen las características principales que brinda la plataforma Java EE: [20]

- Modelo Multicapa.

La plataforma Java EE provee un modelo multicapa y distribuido, esto quiere decir que diferentes partes de una aplicación pueden estar corriendo en diferentes dispositivos. La arquitectura define una capa cliente, una capa intermedia y una capa de datos.

- Servidor de aplicaciones Java EE.

Una característica fundamental de la plataforma Java EE es la existencia y la necesidad de utilizar servidores Java EE, este es un servidor de aplicaciones que implementa las APIs de la plataforma Java EE y provee de los servicios estándar de Java EE. Los servidores Java EE son llamados, servidores de aplicaciones, porque permiten obtener datos de las aplicaciones cliente, del mismo modo que un servidor web sirve páginas web a un navegador. Un servidor Java EE contiene diferentes tipos de componentes, los cuales se corresponden con cada una de las capas de una aplicación multicapa.

- Gestión de los componentes mediante contenedores.

Un contenedor de Java EE es una interfaz entre los componentes y las funcionalidades de más bajo nivel que provee la plataforma Java EE para soportar ese componente. La funcionalidad del contenedor está definida por la plataforma Java EE, y es diferente para cada tipo de componente.

Los servidores Java EE permiten que los diferentes tipos de componentes trabajen juntos para proveer las funcionalidades de una aplicación empresarial.

- Soporte para los componentes clientes.

La plataforma Java EE puede soportar un cierto número de aplicaciones sencillas, gracias a la capa Web y sus componentes tecnológicos. En el caso de existir interacciones más complejas con el usuario, es necesario proveer de funcionalidades directamente a la capa del cliente. Estas funcionalidades son comúnmente implementadas mediante componentes de JavaBeans, que interactúan con los servicios de la capa intermedia vía servlets. Los JavaBeans de la capa del cliente se obtienen a través de un applet y se descargan automáticamente al navegador web.

- Soporte para los componentes lógicos de negocio.

La lógica de negocio se suele implementar en la capa intermedia como *Enterprise Java Beans* (EJB), que permiten a los componentes o a los desarrolladores de aplicaciones concentrarse en la lógica de negocio, mientras el contenedor EJB se encarga de entregar un servicio fiable y escalable.

- Soporte de Java EE estándar.

El estándar de Java EE está definido a través de un conjunto de especificaciones. Entre las principales encontramos las especificaciones de Java EE, la especificación de los Enterprise Java Beans, los Java Servlets y las Java Server Pages(JSP).

## **EJB**

Los *Enterprise Java Beans* son una Interfaz de Programación de Aplicaciones que forman parte del estándar de construcción de aplicaciones empresariales JEE. El objetivo de los EJB es dotar al programador de un modelo que le permita abstraerse de los problemas generales de una aplicación empresarial para centrarse en el desarrollo de la lógica de negocio. El hecho de estar basado en componentes permite que éstos sean flexibles y sobre todo reutilizables.

Están diseñados para el desarrollo y despliegue de aplicaciones de negocio basadas en componentes del lado del servidor. Una vez que se desarrolla una aplicación, ésta puede ser desplegada en cualquier servidor que soporte la especificación de EJB.[23]

El contenedor EJB es la parte del entorno Java EE dedicada a ejecutar la parte lógica de la aplicación Java EE. Enterprise JavaBeans son clases Java que contienen y manipulan las estructuras de datos principales de la aplicación Java EE.[23]

El funcionamiento de los componentes EJB se basa fundamentalmente en el trabajo del contenedor EJB. El contenedor EJB es un programa Java que corre en el servidor y que contiene

todas las clases y objetos necesarios para el correcto funcionamiento de los *Enterprise Java Beans*.

La tecnología EJB define tres tipos de beans:[23]

- Los beans de entidad tienen como objetivo encapsular los objetos del lado del servidor que almacena los datos. Estos representan un objeto concreto que tiene existencia en la base de datos.
- Los beans dirigidos por mensajes son los únicos beans con funcionamiento asíncrono, estos usan el *Java Messaging System (JMS)*.
- Los bean de sesión gestionan el flujo de la información en el servidor.

## **JNDI**

La Interfaz de Nombrado y Directorio Java (JNDI, acrónimo en inglés *Java Naming and Directory Interface*), es una Interfaz de Programación de Aplicaciones de Java la cual posibilita a las aplicaciones guardar y recoger datos identificados por un nombre, organizando dichos datos en una estructura. Permite a las aplicaciones basadas en la plataforma Java acceder a múltiples servicios de nombres y directorios. JNDI hace posible que los desarrolladores creen aplicaciones portátiles habilitadas para varios servicios de nombres y directorios distintos. Especifica una Interfaz de Proveedor de Servicio (SPI) que permite que las implementaciones del servicio de directorio sean integradas en el framework. Las implementaciones pueden hacer uso de un servidor, un fichero, o una base de datos.[23]

## **JPA**

Proporciona un modelo de persistencia para mapear bases de datos relacionales. En las últimas versiones de Java, JPA sigue el mismo camino de simplicidad y robustez, y agrega nuevas funcionalidades. Se puede utilizar para acceder y manipular datos relacionales de Enterprise Java Beans, componentes web y aplicaciones Java.[23]

Los principales componentes de JPA son:

- Mapeo objeto-relacional (ORM, acrónimo en inglés de *Object-Relational Mapping*). Es el mecanismo para mapear objetos a los datos almacenados en una base de datos relacional.
- Una interfaz de programación de aplicaciones administradora de entidad para realizar operaciones en la base de datos tales como crear, ver, modificar, eliminar.
- El lenguaje de consulta de persistencia Java (*JPQL, acrónimo en inglés de Java Persistence Query Language*) que permite recuperar datos con un lenguaje de consultas orientado a objetos.

- Las transacciones y mecanismos de bloqueo cuando se accedan a los datos concurrentemente, la API *Java Transaction* (JTA).

### **Framework Java Server Faces (JSF) 2.2**

Un *framework* es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente suelen incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros para ayudar a desarrollar y unir los diferentes componentes de un proyecto. [24]

Java Server Faces (JSF), define un modelo orientado a eventos, basado en componentes para el desarrollo de aplicaciones web,[25] es un marco de interfaz de usuario para la creación de aplicaciones web Java; se desarrolla a través de Java Community Process (JCP).

En términos de las cuatro capas de una herramienta de Desarrollo Rápido de Aplicaciones (RAD), Java Server Faces define tres de ellas, una arquitectura de componentes, un conjunto estándar de widgets de interfaz de usuario y una infraestructura de aplicaciones. La arquitectura de componentes de JSF define una forma común de crear widgets de interfaz de usuario. Esta arquitectura permite widgets de interfaz de usuario JSF estándar, también establece el escenario para componentes de terceros. Los componentes están orientados a eventos, por lo que JSF permite procesar eventos generados por el desarrollador. [26]

JEE incluye los siguientes componentes: [27]

- Componentes clientes.
- Componentes web.
- Componentes empresariales.

JSF tiene una poderosa arquitectura para mostrar componentes de diferentes maneras. También tiene instalaciones extensibles para validar la entrada y la conversión de objetos, también puede mantener sus componentes de interfaz de usuario automáticamente sincronizados con objetos Java que recopilan valores de entrada de usuario y responden a eventos, que se denominan *beans*. Además, cuenta con un potente sistema de navegación y soporte completo para múltiples idiomas. Estas características constituyen los bloques de construcción básicos de la infraestructura de aplicaciones de JSF necesarios para cualquier sistema.

### **Biblioteca PrimeFaces 5.0 para JSF 2.2**

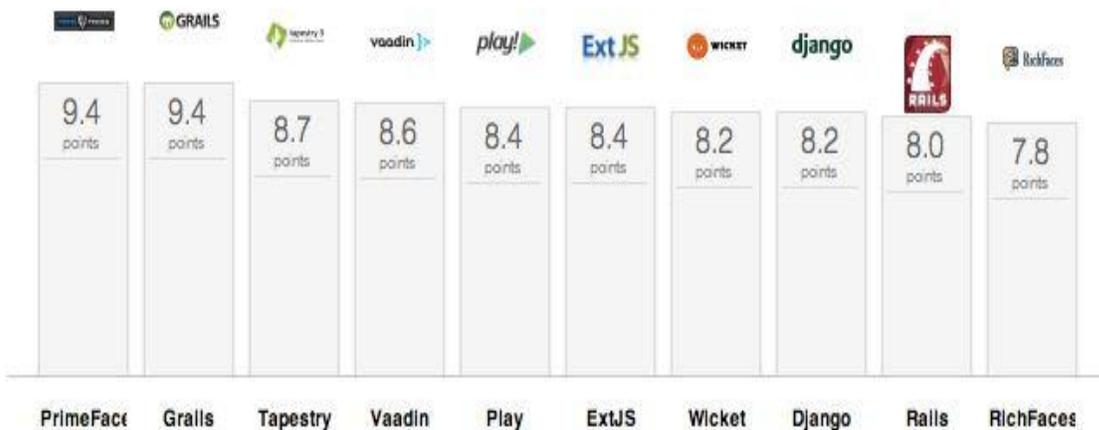
PrimeFaces es una biblioteca de componentes visuales de código abierto para el conjunto Java Server Faces desarrollada y mantenida por Prime Technology. Su objetivo principal es ofrecer un conjunto de componentes para facilitar la creación y diseño de aplicaciones web.[28]

PrimeFaces es una biblioteca ligera, sus componentes se desarrollan con un principio de diseño que establece que "un buen componente de interfaz de usuario debe ocultar la complejidad, pero mantener la flexibilidad". La comunidad PrimeFaces ayuda continuamente al desarrollo de PrimeFaces proporcionando retroalimentación, nuevas ideas, informes de errores y componentes.[28]

Las principales características de PrimeFaces son:[28]

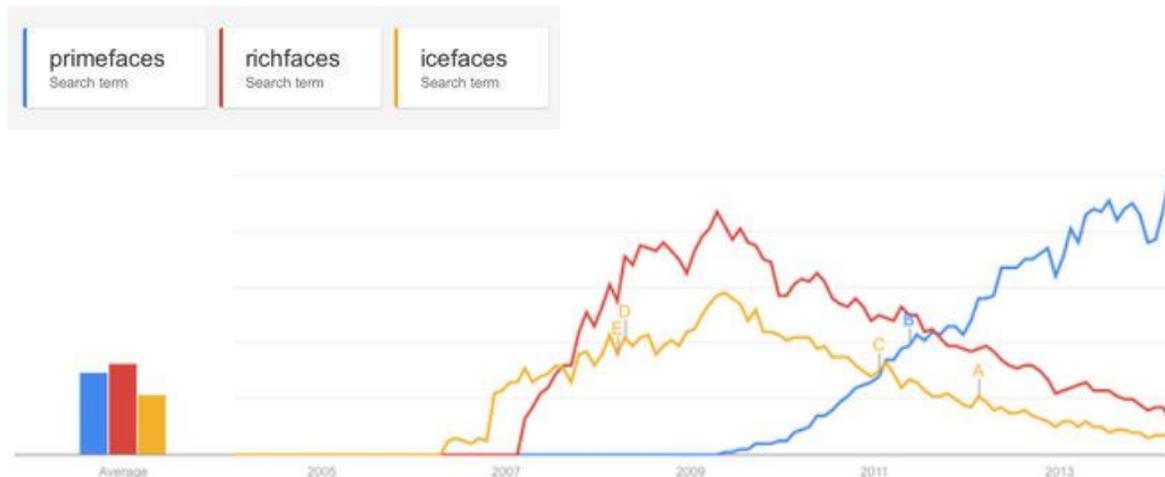
- Soporte nativo de Ajax, incluyendo Push/Coment.
- Kit para crear aplicaciones web móviles.
- Es compatible con otras bibliotecas de componentes como *Jboss RichFaces*.
- Uso de JavaScript no intrusivo.
- Es un proyecto open *source*, activo y estable.

Según DevRates.com, PrimeFaces ha tomado el liderazgo como se puede apreciar en la figura 1.3 y la figura 1.4 con la calificación general 9.4 como el marco favorito de los desarrolladores para crear interfaces de usuario ricas con Java.[29]



**Figura 1.3:** PrimeFaces con Java.

Miles de aplicaciones se han creado con PrimeFaces, incluso los productos de la competencia como *IceFaces* es impulsado por PrimeFaces. A continuación, se muestra un gráfico generado por Google *Trends* comparando la popularidad PrimeFaces con los competidores. Para muchos, PrimeFaces es el estándar de facto de las bibliotecas de componentes JSF. [29]



**Figura 1.4:** PrimeFaces en la actualidad.

El punto fuerte de PrimeFaces es la sencillez de instalación y lo poco pesado que es. El mantenerlo liviano, sin complicaciones a la hora de instalarlo, es decir, sin dependencias ni configuraciones, hace que podamos estar usándolo en unos pocos segundos. [29]

## 1.6 Herramientas

### Ambiente de Desarrollo Integrado

Un entorno de desarrollo integrado, (IDE, acrónimo en inglés de *Integrated Development Environment*) es un programa informático compuesto por un conjunto de herramientas de programación. Puede dedicarse en exclusiva a un solo lenguaje de programación o bien puede utilizarse para varios. Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación; es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. Los IDE proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, PHP, Python, Java, C#, Delphi, Visual Basic. En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución, en donde se permite utilizar el lenguaje de programación en forma interactiva. [30]

### NetBeans versión 8.2

NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE es un producto libre y gratuito sin restricciones de uso. NetBeans IDE 8.2 provee varias mejoras nuevas, tales como mejores características de edición JavaScript, soporte para usar estructuras Spring de soporte web, integración MySQL más ajustada y una mejor manera de compartir bibliotecas entre proyectos dependientes.[31]

Las características de modelos bean y el generador JSF CRUD que se habían eliminado en la versión 6.0 han vuelto, a partir del 6.1, pero ahora en esta versión se encuentran actualizados y con mayores prestaciones. Además de versiones recientes de nuevos módulos, están ahora disponibles como extensiones (plugins). NetBeans IDE 8.2 proporciona analizadores de código y editores para trabajar con las últimas tecnologías Java 8. El IDE también tiene una gama de nuevas herramientas para JavaScript y componentes que mejoran aún más su compatibilidad con Java EE con PrimeFaces.[31]

Esta versión también provee mejoras en el rendimiento, especialmente en el inicio más rápido, menor consumo de memoria y mejores respuestas mientras se trabaja con proyectos muy grandes. En cuanto a base de datos, desde el propio Netbeans se puede conectar a distintos sistemas gestores de bases de datos, como pueden ser Oracle, MySQL y demás, y ver las tablas, realizar consultas y modificaciones, y todo ello integrado en el propio IDE. Se integra con diversos servidores de aplicaciones, de tal manera que se puede gestionar desde el propio IDE: inicio, parada, arranque en modo *debug*, despliegues. Se pueden usar los servidores Apache Tomcat, GlassFish, JBoss, WebLogic, Sailfin, Sun Java System y Application Server.[32]

#### **Servidor de aplicación GlassFish Server versión 4.1.1**

Un servidor es un ordenador dedicado a ejecutar una aplicación servidora y, como aplicación servidora, se conoce al software preparado para aceptar conexiones de otras máquinas con peticiones de servicio y resolver dichas peticiones enviando respuestas de vuelta a las máquinas que originaron las conexiones.[33]

Es aquel software que, además de ser un Servidor WEB, implementa el resto de especificaciones de Java EE como Web Services para Java, Weblogic, Websphere, JBoss, Geronimo y GlassFish. GlassFish es un servidor de aplicaciones que implementa las tecnologías definidas en la plataforma Java EE y permite ejecutar aplicaciones que siguen esta especificación. Es gratuito y de código libre, es la implementación de referencia de Java EE y como tal soporta Enterprise Java Beans, JPA, Java Server Faces, JMS, RMI, Java Server Pages y servlets. Todo lo anterior permite a los desarrolladores crear aplicaciones empresariales que son portables y escalables, además de su integración con las más antiguas tecnologías de este campo. GlassFish permite que componentes opcionales puedan ser instalados para servicios adicionales.[34]

GlassFish es la implementación de referencia de Java EE y tiene como características:

- Es un proyecto Open Source modular que permite incluir sus bibliotecas como parte de otros *frameworks*, *toolkits* y productos.

- Es la base de código de las distribuciones estables, certificadas y con opción de contratar soporte y mantenimiento del Servidor de Aplicaciones de Sun, *Sun Java System Application Server*.
- Es una comunidad que contribuye mediante código, detección de bugs, foros de discusión, *feedback*, documentación, wikis, blogs y otros medios a la difusión y éxito del proyecto. La comunidad GlassFish colabora en la compatibilidad de otros frameworks JAVA Open Source.[33]

### **Servidor de base de datos MySQL Server 5.7**

MySQL es un sistema de administración de bases de datos. Una base de datos es una colección estructurada de tablas que contienen datos. Para agregar, acceder a y procesar datos guardados en un computador, se necesita un administrador como MySQL Server. Dado que los computadores son muy buenos manejando grandes cantidades de información, las bases de datos juegan un papel fundamental en la informatización de procesos.

MySQL es un sistema de administración relacional de bases de datos. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido.[35]

Una base de datos es una colección estructurada de datos. Es usado para añadir, acceder, y procesar los datos almacenados en una base de datos, necesita un sistema de gestión de base de datos como MySQL Server. Al ser los computadores muy buenos en tratar grandes cantidades de datos, los sistemas de gestión de bases de datos juegan un papel central en computación, como aplicaciones autónomas o como parte de otras aplicaciones.

El servidor MySQL también tiene una serie de características prácticas desarrolladas en cooperación con los usuarios. Puede encontrar comparaciones de rendimiento de MySQL Server con otros sistemas de gestión de bases de datos en nuestra página de comparativas de rendimiento.[36]

MySQL Server ofrece hoy en día una gran cantidad de funciones. Su conectividad, velocidad, y seguridad hacen de MySQL Server altamente apropiado para acceder bases de datos en Internet. El software de bases de datos MySQL es un sistema cliente/servidor que consiste en un servidor SQL *multi-threaded* que trabaja con diferentes *backends*, programas y bibliotecas cliente, herramientas administrativas y un amplio abanico de interfaces de programación para aplicaciones.[37]

## **Herramienta de modelado MySQL Workbench 6.3 CE**

MySQL Workbench es una herramienta gráfica para trabajar con servidores y bases de datos MySQL. MySQL Workbench es usada para el modelado visual, desarrollo y administración de SQL.

Funcionalidades de MySQL Workbench:

Desarrollo de SQL, permite crear y administrar conexiones a servidores de bases de datos. Además de permitirle configurar los parámetros de conexión, MySQL Workbench proporciona la capacidad de ejecutar consultas SQL en las conexiones de base de datos utilizando el Editor de SQL incorporado.

Modelado de datos (Diseño), permite crear modelos de esquema de base de datos gráficamente, invertir y enviar ingeniería entre un esquema y una base de datos en directo y editar todos los aspectos de su base de datos utilizando el editor de tablas completo.

Administración del servidor, permite administrar las instancias del servidor MySQL administrando usuarios, realizando copias de seguridad y recuperación, inspeccionando los datos de auditoría, viendo el estado de la base de datos y supervisando el rendimiento del servidor MySQL.

## **Visual Paradigm 8.0**

Es una herramienta CASE que propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación. Constituye una herramienta privada disponible en varias ediciones, cada una destinada a satisfacer diferentes necesidades, aunque también existe una alternativa libre y gratuita de este software. [38]

## **1.7 Metodología de desarrollo de software**

En el proceso de construcción de un software y las metodologías de desarrollo de software constituyen un pilar fundamental. Se definen como un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar un nuevo software. Una metodología puede seguir uno o varios modelos de ciclo de vida, es decir, el ciclo de vida indica qué es lo que hay que obtener a lo largo del desarrollo del proyecto, pero no cómo hacerlo. La metodología indica cómo hay que obtener los distintos productos parciales y finales.

Una metodología define “quién” debe hacer, “qué”, “cuándo” y “cómo” debe hacerlo, durante todo el desarrollo y mantenimiento de un proyecto, con el fin de aumentar la calidad del software que se produce en todas y cada una de sus fases de desarrollo. Establece que procedimientos, reglas y herramientas se deben utilizar durante el desarrollo del software.[39]

## **Metodología Variación AUP-UCI.**

Esta metodología describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en el Proceso Racional Unificado (RUP, acrónimo en inglés de *Rational Unified Process*). Además, aplica técnicas de modelado, gestión de cambios, y refactorización de base de datos para mejorar la productividad.[40]

El Proceso Unificado Ágil (AUP, acrónimo en inglés de *Agile Unified Process*) se preocupa especialmente de la gestión de riesgos. Propone que aquellos elementos con alto riesgo obtengan prioridad en el proceso de desarrollo y sean abordados en etapas tempranas del mismo. Para ello, se crean y mantienen listas identificando los riesgos desde etapas iniciales del proyecto. Especialmente relevante en este sentido es el desarrollo de prototipos ejecutables durante la base de elaboración del producto, donde se demuestre la validez de la arquitectura para los requisitos clave del producto y que determinan los riesgos técnicos.[40]

La metodología AUP presenta diferentes ventajas y desventajas. A continuación, se muestran algunas de ellas:[40]

### **Ventajas**

- El personal sabe lo que está haciendo: no obliga a conocer detalles.
- Simplicidad: Apuntes concisos.
- Agilidad: Procesos simplificados de RUP.
- Herramientas independientes: A disposición del usuario.
- Fácil adaptación de este producto.

### **Desventajas**

- El AUP al ser un proceso simplificado, muchos desarrolladores eligen trabajar con el RUP, por tener a disposición más detalles en el proceso.

Variación AUP-UCI es la metodología de desarrollo que se emplea en los proyectos productivos de la Universidad de las Ciencias Informáticas. Dicha definición se basa en una variación de la metodología AUP en unión con el modelo CMMI-DEV v 1.3. [41]

Se decide hacer una variación de la metodología AUP, de forma tal que se adapte al ciclo de vida definido para la actividad productiva de la UCI. Una metodología de desarrollo de software tiene entre sus objetivos aumentar la calidad del software que se produce, de ahí la importancia de aplicar buenas prácticas, para ello se apoyará en el Modelo CMMI-DEV v1.3. El cual constituye una guía para aplicar las mejores prácticas en una entidad desarrolladora. Estas prácticas se centran en el desarrollo de productos y servicios de calidad. Con la adaptación de AUP que se

propone para la actividad productiva de la UCI se logra estandarizar el proceso de desarrollo de software, dando cumplimiento además a las buenas prácticas que define CMMI-DEV v1.3. Se logra hablar un lenguaje común en cuanto a fases, disciplinas, roles y productos de trabajos. En la variación de AUP para la UCI se definen las siguientes fases, las cuales se aplicarán para guiar el desarrollo del presente trabajo.[41]

**Tabla 1.1** :Fases de AUP adaptada a la actividad productiva de la UCI

Fases	Objetivos
<b>Inicio</b>	Durante el inicio del proyecto se llevan a cabo las actividades relacionadas con la planeación del proyecto. En esta fase se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo y costo y decidir si se ejecuta o no el proyecto.
<b>Ejecución</b>	En esta fase se ejecutan las actividades requeridas para desarrollar el software, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se modela el negocio, obtienen los requisitos, se elaboran la arquitectura y el diseño, se implementa y se libera el producto. Durante esta fase el producto es transferido al ambiente de los usuarios finales o entregado al cliente. Además, en la transición se capacita a los usuarios finales sobre la utilización del software.
<b>Cierre</b>	En esta fase se analizan tanto los resultados del proyecto como su ejecución y se realizan las actividades formales de cierre del proyecto.

Se proponen las siguientes disciplinas:

**Modelado de negocio:** el Modelado del negocio es la disciplina destinada a comprender los procesos de negocio de una organización. Se comprende cómo funciona el negocio que se desea informatizar para tener garantías de que el software desarrollado va a cumplir su propósito.

**Requisitos:** el esfuerzo principal en la disciplina Requisitos es desarrollar un modelo del sistema que se va a construir. Esta disciplina comprende la administración y gestión de los requisitos funcionales y no funcionales del producto.

**Análisis y diseño:** en esta disciplina, si se considera necesario, los requisitos pueden ser refinados y estructurados para conseguir una comprensión más precisa de estos, y una descripción que sea fácil de mantener y ayude a la estructuración del sistema (incluyendo su arquitectura). Además, en esta disciplina se modela el sistema y su forma (incluida su arquitectura) para que

soporte todos los requisitos, incluyendo los requisitos no funcionales. Los modelos desarrollados son más formales y específicos que el de análisis.

Implementación: en la implementación, a partir de los resultados del Análisis y Diseño se construye el sistema.

Pruebas internas: en esta disciplina se verifica el resultado de la implementación probando cada construcción, incluyendo tanto las construcciones internas como intermedias, así como las versiones finales a ser liberadas. Se deben desarrollar artefactos de prueba como: diseños de casos de prueba, listas de chequeo y de ser posibles componentes de prueba ejecutables para automatizar las pruebas.

Pruebas de liberación: pruebas diseñadas y ejecutadas por una entidad certificadora de la calidad externa, a todos los entregables de los proyectos antes de ser entregados al cliente para su aceptación.

Pruebas de aceptación: es la prueba final antes del despliegue del sistema. Su objetivo es verificar que el software está listo y que puede ser usado por usuarios finales para ejecutar aquellas funciones y tareas para las cuales el software fue construido.

Despliegue (opcional): constituye la instalación, configuración, adecuación, puesta en marcha de soluciones informáticas y entrenamiento al personal del cliente.[41]

Los roles de AUP-UCI, que reúnen el conjunto esencial de habilidades para realizar el desarrollo, se describen a continuación: [41]

**Tabla 1.2:** Roles y artefactos de salida

Rol	Función principal	Artefactos de salida
<b>Líder del proyecto</b>  <b>Planificador</b>	Lidera la planeación del proyecto, coordina interacciones con los interesados y conserva el equipo del proyecto enfocado en alcanzar los objetivos del proyecto.	<ul style="list-style-type: none"> <li>○ Registro de proveedores de requisitos.</li> <li>○ Criterios para validar requisitos del cliente.</li> <li>○ Criterios para validar requisitos del producto.</li> </ul>
<b>Analista</b>	Realizar tareas de relevamiento, análisis y diseño de los requerimientos y requisitos en el proyecto.	<ul style="list-style-type: none"> <li>○ Evaluación de requisitos.</li> <li>○ Mapa de procesos.</li> <li>○ Especificación de requisitos de software.</li> <li>○ Manual de usuario.</li> </ul>

<b>Desarrollador</b>	Desarrollar una parte del sistema o completamente, incluyendo su diseño para que se ajuste a la arquitectura, implementación de pruebas unitarias y de integración para los componentes.	<ul style="list-style-type: none"> <li>○ Código fuente.</li> </ul>
<b>Arquitecto</b>	Diseñar la arquitectura del software, la cual incluye tomar las principales decisiones técnicas que condicionan globalmente el diseño y la implementación del proyecto.	<ul style="list-style-type: none"> <li>○ Modelo de diseño.</li> <li>○ Arquitectura de software.</li> </ul>
<b>Probador</b>	Identificar, definir, implementar y dirigir las pruebas necesarias, así como verificar y analizar sus resultados.	<ul style="list-style-type: none"> <li>○ Diseño de casos de prueba.</li> </ul>

## Conclusiones parciales

Una vez expuestos los principales aspectos de este capítulo, se puede concluir lo siguiente:

- Los MES proporcionan la información necesaria para optimizar los procesos productivos en empresas de manufactura, aportando beneficios específicos que se focalizan en las actividades de producción.
- Los servicios de MES ayudan a acortar la distancia entre los ERP y los SCADA, permiten establecer normativas comunes y buenas prácticas en plantas e instalaciones, a la vez que minimizan los costos, los tiempos de entrega e inventarios.
- Como primer paso para el desarrollo de un MES en el CIM es necesaria la digitalización de la recogida de datos en los procesos productivos, lo que permitirá mejorar el proceso de control de la producción del IFA.
- Se escogió como plataforma de desarrollo *Java EE7*, como tecnología de bases de datos *SQL*, como *framework* de desarrollo *Java Server Faces (JSF) 2.2* y la biblioteca *PrimeFaces 5.0*.
- Se seleccionó como entorno de desarrollo integrado *NetBeans* versión 8.2, como servidor de aplicación *GlassFish Server* versión 4.1.1, como servidor de bases de datos *MySQL Server 5.7*, como herramienta de modelado *MySQL Workbench 6.3 CE* y *Visual Paradigm 8* como herramienta case.
- Se propone la utilización de la metodología variación AUP-UCI para el desarrollo de la aplicación, lo que no perjudica la solución propuesta.

# Capítulo 2: Propuesta de solución

## Introducción

En el presente capítulo se da a conocer el proceso de producción del IFA en el CIM y se describen los subprocesos, cuya recogida de datos es objeto de digitalización con la solución propuesta. Se describe la solución propuesta; se identifican los requisitos funcionales y no funcionales, reflejando su descripción en la elaboración de historias de usuario; se realiza la estimación de esfuerzo e iteraciones para el desarrollo; se presenta la arquitectura del sistema, a través del patrón arquitectónico y los patrones de diseño y se especifica el diagrama de despliegue.

### 2.1 Proceso de producción del IFA en el CIM

El proceso de producción del IFA en el CIM cuenta con seis etapas que comienzan con la Descongelación de ampulla, luego la Expansión celular, siguiendo a esta las fermentaciones tanto inóculo como industrial las cuales pasa el proceso por la etapa de Filtración hasta llegar a la de Purificación donde se obtiene como resultado el IFA, como se muestra en la Figura 1. Este proceso se registra en el Libro Maestro de Lotes, que no es más que un histórico del ciclo de vida del proceso de producción del IFA. Hoy en día es totalmente registrado en papel, teniendo almacenado más de 100 registros en dependencia de la vida del producto. El proceso es auditado, se analiza para ver desviaciones, posibles mejoras como gestión y datos productivos que analizarlo en papel es costoso en tiempo y en recursos humanos y en varias ocasiones cuando está lista una información ya no es necesaria.

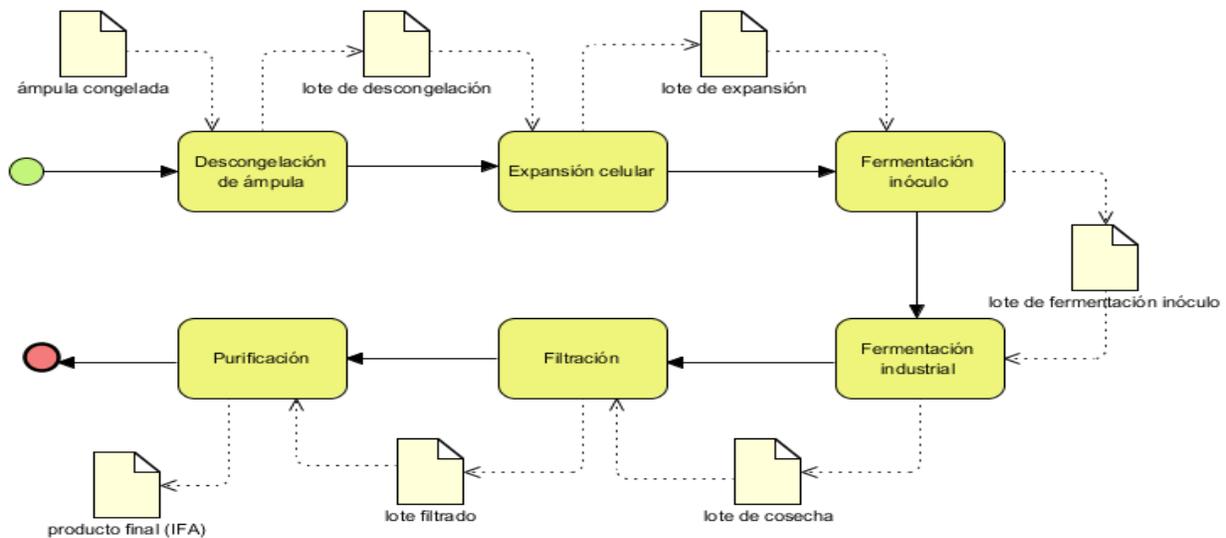


Figura 2.1.- Proceso de producción del IFA

## **2.2 Propuesta de solución**

Se propone desarrollar un sistema que gestione la información referente a la recogida de los datos del proceso de producción del IFA en el CIM, siendo capaz de insertar, modificar, ver y eliminar los datos referentes a dicho proceso, persistiendo los mismos en una base de datos que siga los estándares de la MESA para un correcto proceso de ejecución de manufacturas. Los usuarios contarán con una herramienta sencilla y fácil de usar que estará disponible a través de una aplicación web.

## **2.3 Requisitos del sistema**

Los requisitos para un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas. Estos requerimientos relejan las necesidades de los clientes de un sistema que le ayude a resolver algún determinado problema.[42]

### **2.3.1 Requisitos funcionales**

Los requisitos funcionales de un sistema describen lo que este debe hacer. Estos requerimientos dependen del tipo de software que se desarrolle, de los posibles usuarios del software y del enfoque general tomado por la organización al redactar requerimientos. [42]

Para el desarrollo del sistema propuesto de tomaron como requisitos funcionales los siguientes:

1. Autenticar usuario.
2. Registrar usuario.
3. Registrar los datos referentes al proceso.
4. Modificar los datos referentes al proceso.
5. Ver los datos referentes al proceso.
6. Eliminar los datos referentes al proceso.
7. Buscar los datos referentes al proceso por un criterio determinado.
8. Ordenar los datos referentes al proceso por un criterio determinado.
9. Revisar el registro asociado a cada etapa del proceso.
10. Aprobar el registro asociado a cada etapa del proceso.

### **2.3.2 Requisitos no funcionales**

Los requerimientos no funcionales son restricciones de los servicios o funciones ofrecidas por el sistema. Incluyen restricciones de tiempo sobre el proceso de desarrollo y estándares. Los requisitos no funcionales a menudo se aplican al sistema en su totalidad y en pocas ocasiones se aplican a características o servicios individuales del sistema.[42]

- **Apariencia o interfaz de usuario:** La aplicación debe estar diseñada con una interfaz amigable donde predominen los colores blanco y gris familiarizados con las planillas físicas donde se recogen actualmente los datos, fácil de usar por el usuario de manera que agilice y facilite el trabajo con el software pues el sistema brindará servicios tanto a usuarios familiarizados con ambientes informáticos como a otros no familiarizados.
- **Usabilidad:** La aplicación tiene que ser capaz de ofrecer facilidades de uso para un buen entendimiento y aceptación del producto por los usuarios finales. Debe ser sencilla a la vista teniendo en cuenta que lo pueden utilizar personas que no tienen habilidades con herramientas digitales.
- **Soporte:** Se necesita un servidor que contenga soporte para grandes volúmenes de datos y velocidad media de procesamiento, con menor tiempo de respuesta posible, para la aplicación y la base de datos.
- **Portabilidad:** El sistema será multiplataforma debido a que se ejecutará sobre una máquina virtual que interpreta el código implementado en el lenguaje de programación Java convirtiéndolo a código particular del sistema operativo sin importar sus versiones, y sin necesidad de modificar su código fuente. El sistema se podrá acceder a través de cualquier navegador.
- **Confiabilidad:** El sistema debe ser confiable y preciso en la información que le suministra al usuario para evitar cualquier tipo de error. La información puede ser manejada solamente por usuarios autenticados en la aplicación que le da los permisos de acceso al sistema. Este debe ser estable, fiable y la velocidad de respuesta debe ser rápida durante la utilización del mismo. La información almacenada debe ser confiable en cuanto a su veracidad e integridad desde su recopilación y durante toda su manipulación.
- **Legales:** El sistema estará desarrollado en base a las políticas del software libre, que le fueron ajustadas como: uso del servidor de aplicación GlassFish, servidor de base de datos MySQL Server, lenguaje del lado del servidor Java, que son herramientas libres además del uso de MySQL WorkBench. Como gestor de base de datos.

### 2.3.3 Historias de Usuario

Las historias de usuario corresponden a la técnica utilizada para especificar los requisitos de software. Se trata de formatos en los que el cliente describe brevemente las características que el sistema debe poseer. Cada historia de usuario es lo suficientemente delimitada para que los programadores puedan implementarlas de una a tres semanas para no superar el tamaño de una iteración.[43]

A continuación, se muestran las historias de usuario utilizadas para especificar los requisitos funcionales del software.

**Tabla 2.1:** Historia de usuario #1.

<b>Número:1</b>	<b>Nombre del requisito:</b> Autenticar usuario.	
<b>Programador:</b> Felix Antonio Sotolongo Barrios	<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta	<b>Punto Estimado:</b> 0.6	
<b>Riesgo en Desarrollo:</b> Alta	<b>Referencia -</b>	
<b>Descripción:</b> Para que el usuario pueda entrar al sistema debe autenticarse utilizando su nombre de usuario y contraseña		
<b>Observaciones:</b> El usuario debe poseer un nombre de usuario y una contraseña del sistema		

**Tabla 2.2:** Historia de usuario #2.

<b>Número:2</b>	<b>Nombre del requisito:</b> Registrar usuario.	
<b>Programador:</b> Felix Antonio Sotolongo Barrios	<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta	<b>Punto Estimado:</b> 0.6	
<b>Riesgo en Desarrollo:</b> Alta	<b>Referencia -</b>	
<b>Descripción:</b> El administrador podrá crear la cuenta de cada persona en el sistema.		
<b>Observaciones:</b> El usuario debe haber iniciado sesión con el rol de administrador.		

**Tabla 2.3:** Historia de usuario #3.

<b>Número:3</b>	<b>Nombre del requisito:</b> Registrar los datos referentes al proceso.	
<b>Programador:</b> Felix Antonio Sotolongo Barrios	<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta	<b>Punto Estimado:</b> 0.8	
<b>Riesgo en Desarrollo:</b> Media	<b>Referencia -</b>	
<b>Descripción:</b> El usuario podrá registrar los datos del proceso en la base de datos.		
<b>Observaciones:</b> El usuario debe haber accedido con la responsabilidad de Jefe Etapa Expansión Celular o Jefe de Etapa del Proceso de Fermentación.		

**Tabla 2.4:** Historia de usuario #4.

<b>Número:4</b>	<b>Nombre del requisito:</b> Modificar los datos referentes al proceso.	
<b>Programador:</b> Ramón Díaz García	<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta	<b>Punto Estimado:</b> 0.8	
<b>Riesgo en Desarrollo:</b> Media	<b>Referencia -</b>	
<b>Descripción:</b> El usuario podrá modificar los datos del proceso en la base de datos.		
<b>Observaciones:</b> El usuario debe haber accedido con la responsabilidad de Jefe Etapa Expansión Celular o Jefe de Etapa del Proceso de Fermentación.		

**Tabla 2.5:** Historia de usuario #5.

<b>Número:5</b>	<b>Nombre del requisito:</b> Ver los datos referentes al proceso.	
<b>Programador:</b> Ramón Díaz García	<b>Iteración Asignada:</b> 2	
<b>Prioridad:</b> Media	<b>Punto Estimado:</b> 0.8	
<b>Riesgo en Desarrollo:</b> Media	<b>Referencia -</b>	
<b>Descripción:</b> El usuario podrá ver los datos del proceso en la base de datos.		
<b>Observaciones:</b> No existe restricción de responsabilidad.		

**Tabla 2.6:** Historia de usuario #6.

<b>Número:6</b>	<b>Nombre del requisito:</b> Eliminar los datos referentes al proceso.	
<b>Programador:</b> Ramón Díaz García	<b>Iteración Asignada:</b> 2	
<b>Prioridad:</b> Alta	<b>Punto Estimado:</b> 0.8	
<b>Riesgo en Desarrollo:</b> Media	<b>Referencia -</b>	
<b>Descripción:</b> El usuario podrá eliminar los datos del proceso en la base de datos.		
<b>Observaciones:</b> El usuario debe haber accedido con la responsabilidad de Jefe Etapa Expansión Celular o Jefe de Etapa del Proceso de Fermentación.		

**Tabla 2.7:** Historia de usuario #7.

<b>Número:7</b>	<b>Nombre del requisito:</b> Buscar los datos referentes al proceso por un criterio determinado.	
<b>Programador:</b> Ramón Díaz García	<b>Iteración Asignada:</b> 2	
<b>Prioridad:</b> Alta	<b>Punto Estimado:</b> 0.8	
<b>Riesgo en Desarrollo:</b> Media	<b>Referencia -</b>	
<b>Descripción:</b> El usuario podrá buscar los datos de la etapa del proceso en la que se encuentra por un criterio determinado.		
<b>Observaciones:</b> No existe restricción de responsabilidad.		

**Tabla 2.8:** Historia de usuario #8.

<b>Número:8</b>	<b>Nombre del requisito:</b> Ordenar los datos referentes al proceso por un criterio determinado.	
<b>Programador:</b> Ramón Díaz García	<b>Iteración Asignada:</b> 2	
<b>Prioridad:</b> Media	<b>Punto Estimado:</b> 0.8	
<b>Riesgo en Desarrollo:</b> Media	<b>Referencia -</b>	
<b>Descripción:</b> El usuario podrá ordenar los datos de la etapa del proceso en la que se encuentra por un criterio determinado.		
<b>Observaciones:</b> No existe restricción de responsabilidad.		

**Tabla 2.9:** Historia de usuario #9.

<b>Número:9</b>	<b>Nombre del requisito:</b> Revisar el registro asociado a cada etapa del proceso.	
<b>Programador:</b> Felix Antonio Sotolongo Barrios	<b>Iteración Asignada:</b> 3	
<b>Prioridad:</b> Alta	<b>Punto Estimado:</b> 0.6	
<b>Riesgo en Desarrollo:</b> Alta	<b>Referencia -</b>	
<b>Descripción:</b> El usuario podrá revisar los datos de la etapa del proceso en la que se encuentra para verificar el correcto estado del mismo.		
<b>Observaciones:</b> El usuario debe haber accedido con la responsabilidad de Documentador o Especialista Aseguramiento de la Calidad AntyTer.		

**Tabla 2.10:** Historia de usuario #10.

<b>Número:10</b>	<b>Nombre del requisito:</b> Aprobar el registro asociado a cada etapa del proceso.	
<b>Programador:</b> Felix Antonio Sotolongo Barrios	<b>Iteración Asignada:</b> 3	
<b>Prioridad:</b> Alta	<b>Punto Estimado:</b> 0.6	
<b>Riesgo en Desarrollo:</b> Alta	<b>Referencia -</b>	
<b>Descripción:</b> El usuario podrá aprobar los datos de la etapa del proceso en la que se encuentra para verificar el correcto estado del mismo.		
<b>Observaciones:</b> El usuario debe haber accedido con la responsabilidad de Jefe Operaciones Productivas o Jefe Departamento Aseguramiento de la Calidad IFA Antyter.		

### 2.3.4 Estimación de esfuerzos e iteraciones por Historia de Usuario

**Tabla 2.11.** Estimación de esfuerzos e iteraciones por Historia de Usuario

Historias de Usuario	Puntos de estimación	Iteración	Duración de las iteraciones (Semanas)
Autenticar usuario.	0.6	1	3
Registrar usuario.	0.6		
Registrar los datos referentes al proceso.	0.8		
Modificar los datos referentes al proceso.	0.8		
Ver los datos referentes al proceso.	0.8	2	3
Eliminar los datos referentes al proceso.	0.8		
Buscar los datos referentes al proceso por un criterio determinado.	0.8		
Ordenar los datos referentes al proceso por un criterio determinado.	0.8	3	2
Revisar el registro asociado a cada etapa del proceso.	0.6		
Aprobar el registro asociado a cada etapa del proceso.	0.6		

Luego de definir las historias de usuario y estimar el esfuerzo propuesto para su desarrollo, se realizó el sistema en 3 iteraciones las cuales se describen a continuación:

Iteración 1: Esta iteración tiene como objetivo realizar las HU 1, 2, 3, y 4 las cuales se encargan de la autenticación, registro y restricción de usuarios, así como el registro y modificación de los datos respectivamente.

Iteración 2: Esta iteración tiene como objetivo realizar las HU 5, 6 y 7 las cuales se encargan de mostrar, eliminar y buscar los datos referentes al proceso.

Iteración 3: Esta iteración tiene como objetivo realizar las HU 8, 9 y 10 las cuales se encarga de ordenar los datos, revisar y aprobar el registro asociado a cada etapa.

## **2.4 Descripción de la Arquitectura**

### **2.4.1 Patrón Arquitectónico**

El patrón Modelo-Vista-Controlador (MVC) surge con el objetivo de reducir el esfuerzo de programación, necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos, a partir de estandarizar el diseño de las aplicaciones. El patrón MVC es un paradigma que divide las partes que conforman una aplicación en el Modelo, las Vistas y los Controladores, permitiendo la implementación por separado de cada elemento, garantizando así la actualización y mantenimiento del software de forma sencilla y en un reducido espacio de tiempo. A partir del uso de frameworks basados en el patrón MVC se puede lograr una mejor organización del trabajo y mayor especialización de los desarrolladores y diseñadores. [44]

El modelo es el responsable de:

- Acceder a la capa de almacenamiento de datos. [44]En el sistema se evidencia en las clases de los paquetes "Model" y "EJB"

El controlador es responsable de:

- Recibir los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
- Contiene reglas de gestión de eventos, del tipo "Si Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas.[44] En el sistema de evidencia en las clases del paquete "*Controller*".

Las vistas son responsables de:

- Recibir datos procesados por el controlador o del modelo y mostrarlos al usuario.

- Tienen un registro de su controlador asociado.
- En el sistema se evidencia su uso en las páginas web.[44]

## 2.4.2 Patrones de diseño

En este acápite se hace referencia a los patrones Grasp y Gof utilizados en el diseño de la aplicación

### GRASP

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones.[45]

**Patrón Bajo Acoplamiento y Alta Cohesión:** Este patrón se encarga de que no existan dependencias innecesarias entre las clases y que realicen funciones estrechamente relacionadas con su contenido. El empleo de este patrón se evidencia ya que las clases del proyecto fueron diseñadas bajo estas especificaciones.

**Patrón Controlador:** Este patrón es utilizado en varias clases del proyecto, una de ellas es la clase *OperacionController*, la cual se encarga de recibir los datos del usuario y enviarlos a distintas clases según el método llamado. Este patrón sugiere dividir la lógica de negocios de la capa de presentación, para así aumentar la reutilización de código.

**Patrón Experto:** Este patrón es utilizado para asegurar que cada clase tenga todos los elementos necesarios para realizar sus responsabilidades.

**Patrón Creador:** Este patrón es el encargado de la creación de instancias de objetos o clases ya que posee la información necesaria para este proceso, la utilización de este patrón se evidencia con la clase *OperacionController*.

### GoF

Los patrones GoF son soluciones basadas en la experiencia que corrigen problemas comunes que se presentan en el diseño. Estos describen 23 patrones comúnmente utilizados y de gran aplicabilidad en problemas de diseño usando modelamiento UML. Se agrupan en tres grandes categorías: creacionales, estructurales y de comportamiento. Dentro de esta primera categoría se va a enmarcar el desarrollo de la aplicación utilizando el patrón Singleton. [57]

**Singleton (Instancia única):** El patrón Singleton en la aplicación asegura que sólo exista una instancia de una clase dentro del sistema proporcionando un punto de acceso común, también conocido como “Clase de instancia única” o “Instancia única”. La utilización de este patrón

posibilitará tener solamente un punto de acceso a la clase que se encarga de establecer la conexión a la base de datos. Su uso se evidencia a través del Pool de Conexiones.

## 2.5 Diagrama de Despliegue

Diagrama de Despliegue es un tipo de diagrama del Lenguaje Unificado de Modelado que muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos.[46]

Para el despliegue de la aplicación es necesario una computadora cliente la cual se conectará a un servidor de aplicación a través de un navegador web por el puerto 80 del protocolo HTTP. Este servidor a su vez establecerá la conexión con el servidor de base de datos a través del puerto 3306 del protocolo TCP/IP.

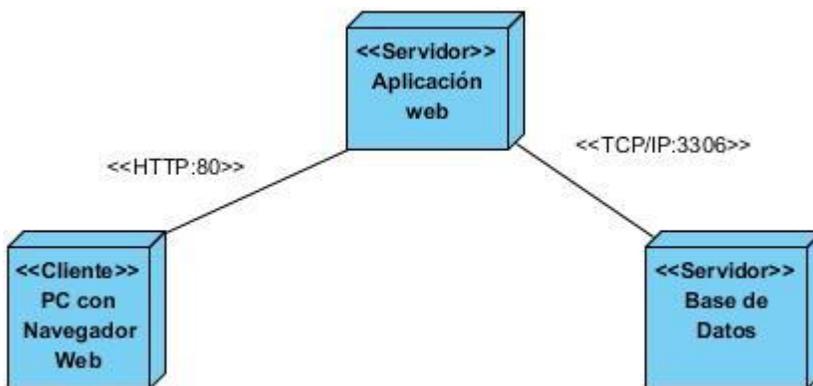


Figura2.2 Diagrama de despliegue.

## Conclusiones Parciales

Durante el desarrollo del capítulo se describió el proceso de producción del IFA en el CIM y los subprocesos de los que se encarga este trabajo a través de la automatización de la recogida de datos; quedó definida la propuesta de solución, los requisitos funcionales y no funcionales, la descripción de las historias de usuario, así como la estimación de esfuerzos e iteraciones para el desarrollo; se definió la arquitectura del sistema a través del patrón arquitectónico y los patrones de diseño y se concibió el diagrama del despliegue del sistema. A través del análisis de los elementos planteados se puede concluir lo siguiente:

- El sistema a desarrollar digitalizará la recogida de datos de los subprocesos o etapas descongelación, expansión y fermentación del proceso de producción del IFA en el CIM.
- La propuesta de solución se basa en un sistema web capaz de insertar, modificar, ver y eliminar los datos referentes a dicho proceso, persistiendo los mismos en una base de datos que siga los estándares de MESA para un correcto proceso de ejecución de manufacturas.

- El sistema cuenta con 10 requisitos funcionales que serán desarrollados en su totalidad y los requisitos no funcionales se basan en rasgos de apariencia, usabilidad, rendimiento, soporte, portabilidad, confiabilidad y aspectos legales.
- A través de las historias de usuario se construirá un sistema que brindará solución a la problemática planteada
- Según los cálculos estimados por los autores de este trabajo, se requieren 3 iteraciones y 8 semanas para el desarrollo del sistema.
- Para definir la arquitectura del sistema se utilizó el patrón arquitectónico Modelo-Vista-Controlador y los patrones de diseño *Grasp* y *Gof*, dentro de este último se utiliza el patrón *Singleton*.
- Para el despliegue del sistema se necesitará instalar un servidor de aplicación o servidor web, un servidor de bases de datos y los usuarios encargados de acceder al sistema necesitarán una computadora (cliente) con un navegador web.

# Capítulo 3: Validación y pruebas

## Introducción

En el presente capítulo, una vez concluida la propuesta de solución, se procede a la implementación de las clases y a la realización de pruebas al sistema, para determinar si la digitalización de la recogida de datos en las etapas descongelación, expansión y fermentación del proceso de producción del IFA en el CIM, cumple las características establecidas y las descripciones de las historias de usuario descritas en el capítulo 2. Como parte del desarrollo de este capítulo se describen las principales pruebas que se pueden realizar a los sistemas de software, se detallan los casos de pruebas a realizar al sistema y, por último, se reflejan los resultados de la ejecución de las pruebas realizadas para comprobar las funcionalidades del sistema.

### 3.1 Pruebas a los sistemas de software

Las pruebas de software consisten en la dinámica de la verificación del comportamiento de un programa en un conjunto finito de casos de prueba. Son una serie de actividades que se realizan con el propósito de encontrar los posibles fallos de implementación, calidad o usabilidad de un programa u ordenador; probando el comportamiento del mismo.[47] Las pruebas de software son un elemento crítico para la garantía de calidad de la aplicación y representan una revisión final de las especificaciones, del diseño y de la codificación.[48] Las pruebas tienen por objetivo asegurar la calidad de la aplicación que se está desarrollando. Existen diferentes tipos de pruebas, ámbitos de aplicación, estructura y estrategias.[49]

Una vez generado el código fuente de cualquier sistema, es necesario probar el *software* para descubrir y corregir la mayor cantidad de errores posibles, contribuyendo de esta manera a elevar la calidad de los productos desarrollados y a la seguridad de los programadores a la hora de introducir cambios o modificaciones.

Existen diferentes tipos de pruebas con diferentes características que se ajustan a condiciones específicas, todas comparten el mismo objetivo: detectar anomalía en el funcionamiento del sistema. Por lo general están orientadas a comprobar la funcionalidad, el sistema debe comportarse a la altura de los requisitos especificados por el cliente y la lógica del negocio.

Las pruebas de *software* son un elemento crítico para la garantía del correcto funcionamiento de la aplicación. Entre sus objetivos están:[50]

- Detectar defectos en el software.
- Verificar la integración adecuada de los componentes.
- Verificar que todos los requisitos se han implementado correctamente.
- Identificar y asegurar que los defectos encontrados se han corregido antes de entregar el software al cliente.
- Diseñar casos de prueba que sistemáticamente saquen a la luz diferentes clases de errores, haciéndolo con la menor cantidad de tiempo y esfuerzo.

Tipos de pruebas de software: [49]

Según el ámbito donde se aplique la prueba:

- Prueba de unidad.
- Prueba de integración.
- Prueba de validación.
- Prueba del sistema.

Según el nivel abstracción de la prueba: [49]

- Caja negra
- Caja blanca

En cada fase del ciclo de vida de desarrollo del software se plantea un conjunto de pruebas que permiten constatar que el software desarrollado satisface las especificaciones de esa fase.

Durante la fase de análisis del sistema se especifican las pruebas de sistema, que comprueban que todo el sistema (parte manual, software desarrollado, bases de datos) funcione correctamente.

Durante la fase de análisis de los requisitos del software se especifican las pruebas de validación que consiste en comprobar que el software desarrollado satisface todos los requisitos funcionales, de comportamiento y de rendimiento especificados durante el análisis.

Durante la fase de diseño se especifican las pruebas de integración que consiste en comprobar que el programa se ha construido correctamente. Su objetivo es coger cada módulo ya probado e integrarlo en el sistema y comprobar que globalmente funciona correctamente.

Al finalizar la codificación de cada módulo se realiza la prueba de unidad, que consiste en comprobar que el módulo funcione acertadamente, todos los caminos de control importantes se ejecuten de acuerdo con lo especificado.[48]

En cuanto al nivel de abstracción, se puede decir que el método de caja blanca se utiliza para el análisis de los distintos caminos de ejecución de los algoritmos y el método de caja negra para el análisis de los procesos externos funcionales.[51]

Las técnicas más comunes aplicadas en los procesos de prueba tienen el objetivo de seleccionar buenos casos de prueba, esto es, casos que tengan una probabilidad alta de descubrir un error. Tradicionalmente se han considerado dos enfoques complementarios para seleccionar casos de prueba, denominados caja blanca y caja negra. Las técnicas de caja blanca, también denominadas pruebas estructurales, utilizan el código fuente de programa, y especialmente su estructura de control, para seleccionar casos de prueba. Por otro lado, las técnicas de caja negra o funcionales, obtienen casos a partir de los requisitos funcionales del programa a probar, por lo que no se tiene en cuenta la forma en que se codifica esa funcionalidad, sino que se consideran únicamente las entradas y las salidas. [47]

En la siguiente tabla se especifican los niveles de pruebas de software, donde se plasma según la prueba, su objetivo, los participantes que intervienen, el ambiente donde se desarrolla y el método que se usa.

**Tabla 3.1:** Niveles de prueba de software. Adaptación de ([52])

<b>Pruebas</b>	<b>Objetivo</b>	<b>Participantes</b>	<b>Ambiente</b>	<b>Método</b>
Unitaria	Detectar errores en los datos, lógica y algoritmo	-Programadores	Desarrollo	Caja Blanca
Integración	Detectar errores de interfaces y relaciones entre componentes	-Programadores	Desarrollo	Caja Blanca
Funcional	Detectar errores en la implementación de los requerimientos	-Programadores -Analistas	Desarrollo	Caja Negra
Sistema	Detectar fallas en el cubrimiento de los requerimientos	-Analistas	Desarrollo	Caja Negra

Aceptación	Detectar fallas en la implementación del sistema	-Analistas -Cliente	Productivo	Caja Negra
------------	--	------------------------	------------	------------

### **Selección de las pruebas a realizar**

Por la importancia y significación que tienen en el ciclo de vida de un producto de software, y teniendo en cuenta los roles que intervienen, se decidió realizar pruebas funcionales y pruebas de aceptación al sistema propuesto, utilizando el método de pruebas de caja negra. A continuación, se especifican algunos detalles de estos tipos de prueba.

### **Pruebas funcionales**

El objetivo de la prueba funcional es validar cuando el comportamiento de la aplicación web, cumple o no con las especificaciones indicadas. La prueba funcional debe tomar el punto de vista del usuario. Las funciones son probadas realizando entradas y examinando las salidas.

Las pruebas funcionales validan partes de las aplicaciones web. Estas pruebas deben simular la navegación del usuario, realizando peticiones y comprobando los elementos de la respuesta, tal y como lo haría manualmente cualquier usuario para validar que una determinada acción hace lo que se supone que tiene que hacer. [53]

Una prueba funcional es una prueba basada en la ejecución, revisión y retroalimentación de las funcionalidades previamente diseñadas para la aplicación web. Las pruebas funcionales se hacen mediante el diseño de modelos de prueba que buscan evaluar cada una de las opciones con las que cuenta el paquete informático. Las pruebas funcionales en la mayoría de los casos son realizadas manualmente por el analista de pruebas, también es posible automatizar este tipo de pruebas utilizando herramientas.[54]

### **Pruebas de aceptación**

Las pruebas de aceptación se refieren a las pruebas llevadas a cabo por parte del cliente, labor fundamental para que valide el sistema, como último paso, previo a la puesta en explotación. Se debe insistir, principalmente, en los criterios de aceptación del sistema que sirven de base para asegurar que satisface los requisitos exigidos. Los criterios de aceptación deben ser definidos de forma clara, prestando especial atención a aspectos como:[55]

- Procesos críticos del sistema.
- Rendimiento del sistema.
- Seguridad.
- Disponibilidad.

El ciclo de vida de las pruebas de aceptación es el siguiente:[47]

- Diseñar casos de prueba de aceptación basados en los requerimientos que presenta el cliente.
- Preparar datos de las pruebas de aceptación.
- Validar datos de los casos de prueba de aceptación.
- Procedimiento de la prueba.
- Ejecutar las pruebas de aceptación para validar el análisis de requerimientos del cliente.
- Comparar los resultados de las pruebas con los casos de prueba iniciales.

### **Método de pruebas de Caja Negra**

El método de caja negra se refiere a las pruebas llevadas a cabo sobre la interfaz del *software*. Es decir, a través de ellas se pretende demostrar que las funciones del *software* son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene. Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un *software*. [56]

Estas pruebas permiten encontrar:[56]

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos o en accesos a las Bases de Datos externas.
- Errores de rendimiento.
- Errores de inicialización y terminación.

Para desarrollar la prueba de caja negra existen técnicas como:[56]

- Técnica de la **Partición de Equivalencia**: esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software.
- Técnica del **Análisis de Valores Límites**: esta Técnica prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.
- Técnica de **Grafos de Causa-Efecto**: es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

Estas son pruebas en las que el sistema a probar es visto como una “caja negra”, es decir, se desconoce su estructura interna y cualquier detalle de implementación. La prueba proporciona un conjunto de datos de entradas al sistema y comprueba que los datos proporcionados en la salida son los esperados. Las pruebas de caja negra tienen como objetivo principal revisar que

se cumplan los requisitos funcionales del sistema, que no hay requisitos incompletos y que la interfaz no contiene errores. [49]

### 3.2 Diseño de casos de pruebas al sistema

El diseño de caso de prueba es una parte de las pruebas de componentes y sistemas en las que se diseñan los casos de pruebas (entradas y salidas esperadas) para probar el sistema.[42]

A continuación, se muestran los casos de prueba diseñados para las **Historias de usuario** descritas en el capítulo anterior. Se diseñó un caso de prueba para cada una de las historias de usuario, donde se tuvieron en cuenta los siguientes aspectos:

- **Código:** identifica la prueba en cuestión, incluye el número de HU y de la prueba.
- **HU:** número de la HU a la cual pertenece.
- **Nombre:** nombre del caso de prueba.
- **Descripción:** acciones que debe realizar el sistema.
- **Condiciones de ejecución:** describe las características y elementos en general que debe tener el sistema para realizar el caso de prueba.
- **Entrada/Pasos de Ejecución:** incluye las entradas necesarias del sistema y los pasos necesarios para realizar el caso de prueba.
- **Salidas/Resultado Esperado:** respuesta que debe dar el sistema luego de aplicar las entradas y los pasos de ejecución descritos.
- **Evaluación de la prueba:** clasificación de la prueba de acuerdo a su resultado (satisfactorio o insatisfactorio).

**Tabla 3.2:** Caso de prueba para la Historia de Usuario #1.

Caso de prueba 1	
<b>Código:</b> HU1_C1	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Autenticar usuario	
<b>Descripción:</b> Para que el usuario pueda entrar al sistema debe autenticarse utilizando su nombre de usuario y contraseña.	
<b>Condiciones de Ejecución:</b> El usuario debe tener una cuenta en el sistema.	
<b>Entradas/Pasos de Ejecución:</b> <ul style="list-style-type: none"><li>- El usuario abre la aplicación web</li><li>- El usuario se autentica, poniendo su usuario y contraseña</li></ul>	
<b>Salidas/Resultado Esperado:</b> El usuario accede al sistema.	
<b>Evaluación de la Prueba:</b> -	

**Tabla 3.3:** Caso de prueba para la Historia de Usuario #2.

Caso de prueba 2	
<b>Código:</b> HU2_C2	<b>Historia de usuario:</b> 2
<b>Nombre:</b> Registrar usuario	
<b>Descripción:</b> El administrador podrá crear la cuenta de cada persona en el sistema.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado en el sistema con el rol de administrador.	
<b>Entradas/Pasos de Ejecución:</b> <ul style="list-style-type: none"><li>- Usando el clic primario selecciona la opción de crear usuario</li><li>- Llena los campos necesarios</li><li>- Presiona el botón guardar</li><li>- Confirma si desea registrar el usuario</li></ul>	
<b>Salidas/Resultado Esperado:</b> Se registra un nuevo usuario en el sistema.	
<b>Evaluación de la Prueba:</b> -	

**Tabla 3.4:** Caso de prueba para la Historia de Usuario #3.

Caso de prueba 3	
<b>Código:</b> HU3_C3	<b>Historia de usuario:</b> 3
<b>Nombre:</b> Registrar los datos referentes al proceso.	
<b>Descripción:</b> El usuario podrá registrar los datos del proceso en la base de datos.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado con la responsabilidad de jefe de etapa de la expansión celular o jefe del proceso de fermentación.	
<b>Entradas/Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>- Selecciona la etapa.</li> <li>- Clic primario a la opción crear.</li> <li>- Registra los datos.</li> <li>- Clic primario en la opción guardar.</li> <li>- Confirma si desea guardar los datos.</li> </ul>	
<b>Salidas/Resultado Esperado:</b> Se notifica que la etapa ha sido creada satisfactoriamente y se inserta la información en la base de datos.	
<b>Evaluación de la Prueba:</b> -	

**Tabla 3.5:** Caso de prueba para la Historia de Usuario #5.

Caso de prueba 4	
<b>Código:</b> HU4_C4	<b>Historia de usuario:</b> 4
<b>Nombre:</b> Modificar los datos referentes al proceso.	
<b>Descripción:</b> El usuario podrá modificar los datos del proceso en la base de datos.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado con la responsabilidad de jefe de etapa de la expansión celular o jefe del proceso de fermentación.	
<b>Entradas/Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>- Selecciona la etapa</li> <li>- Clic secundario sobre el dato que desea modificar</li> <li>- Clic primario en la opción editar</li> <li>- Editar los datos deseados</li> <li>- Clic primario en la opción guardar.</li> <li>- Confirma si desea guardar los datos.</li> </ul>	
<b>Salidas/Resultado Esperado:</b> Se actualiza los valores en la base de datos y no se notifica que la etapa ha sido modificada	
<b>Evaluación de la Prueba:</b> -	

**Tabla 3.6:** Caso de prueba para la Historia de Usuario #5.

Caso de prueba 5	
<b>Código:</b> HU5_C5	<b>Historia de usuario:</b> 5
<b>Nombre:</b> Ver los datos referentes al proceso.	
<b>Descripción:</b> El usuario podrá ver los datos del proceso en la base de datos.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado para ver datos, independientemente de la responsabilidad.	
<b>Entradas/Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>- Selecciona la etapa</li> <li>- Clic secundario sobre el dato que desea ver</li> <li>- Clic primario en la opción ver</li> <li>- Clic primario en la opción cerrar</li> </ul>	
<b>Salidas/Resultado Esperado:</b> Se abre una ventana emergente con los valores	
<b>Evaluación de la Prueba:</b> -	

**Tabla 3.7:** Caso de prueba para la Historia de Usuario #6.

Caso de prueba 6	
<b>Código:</b> HU6_C6	<b>Historia de usuario:</b> 6
<b>Nombre:</b> Eliminar los datos referentes al proceso.	
<b>Descripción:</b> El usuario podrá eliminar los datos del proceso en la base de datos.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado con la responsabilidad de jefe de etapa de la expansión celular o jefe del proceso de fermentación.	
<b>Entradas/Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>- Selecciona la etapa</li> <li>- Clic secundario sobre el dato que desea eliminar</li> <li>- Clic primario en la opción eliminar</li> <li>- Elimina el elemento de la base de datos</li> </ul>	
<b>Salidas/Resultado Esperado:</b> Se actualiza los valores en la base de datos y no se notifica que la etapa ha sido eliminada.	
<b>Evaluación de la Prueba:</b> -	

**Tabla 3.8:** Caso de prueba para la Historia de Usuario #7.

Caso de prueba 7	
<b>Código:</b> HU7_C7	<b>Historia de usuario:</b> 7
<b>Nombre:</b> Buscar los datos referentes al proceso por un criterio determinado.	
<b>Descripción:</b> El usuario podrá buscar los datos de la etapa del proceso en la que se encuentra por un criterio determinado.	
<b>Condiciones de Ejecución:</b> El usuario puede buscar por un criterio determinado.	
<b>Entradas/Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>- Selecciona la etapa</li> <li>- Se posiciona en un campo</li> <li>- Busca según un criterio determinado</li> </ul>	
<b>Salidas/Resultado Esperado:</b> Muestra solamente los resultados que coincidan con la búsqueda	
<b>Evaluación de la Prueba:</b> -	

**Tabla 3.9:** Caso de prueba para la Historia de Usuario #8.

Caso de prueba 8	
<b>Código:</b> HU8_C8	<b>Historia de usuario:</b> 8
<b>Nombre:</b> Ordenar los datos referentes al proceso por un criterio determinado.	
<b>Descripción:</b> El usuario podrá ordenar los datos de la etapa del proceso en la que se encuentra por un criterio determinado.	
<b>Condiciones de Ejecución:</b> El usuario puede ordenar por un criterio determinado.	
<b>Entradas/Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>- Selecciona la etapa</li> <li>- Se posiciona en un campo</li> <li>- Busca según un criterio determinado</li> </ul>	
<b>Salidas/Resultado Esperado:</b> Muestra los datos ordenados dependiendo del criterio ordenado	
<b>Evaluación de la Prueba:</b> -	

**Tabla 3.10:** Caso de prueba para la Historia de Usuario #9.

Caso de prueba 9	
<b>Código:</b> HU9_C9	<b>Historia de usuario:</b> 9
<b>Nombre:</b> Revisar el registro asociado a cada etapa del proceso.	
<b>Descripción:</b> El usuario podrá revisar los datos de la etapa del proceso en la que se encuentra para verificar el correcto estado del mismo.	
<b>Condiciones de Ejecución:</b> El usuario tiene que estar autenticado con responsabilidad de Documentador AntyTer y Especialista Aseguramiento de la Calidad AntyTer	
<b>Entradas/Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>- Selecciona la etapa</li> <li>- Revisa el proceso</li> <li>- Marca el botón revisar</li> </ul>	
<b>Salidas/Resultado Esperado:</b> El sistema muestra el registro llenado con la opción revisar	
<b>Evaluación de la Prueba:</b> -	

**Tabla 3.11:** Caso de prueba para la Historia de Usuario #10.

Caso de prueba 10	
<b>Código:</b> HU10_C10	<b>Historia de usuario:</b> 10
<b>Nombre:</b> Aprobar el registro asociado a cada etapa del proceso.	
<b>Descripción:</b> El usuario podrá aprobar los datos de la etapa del proceso en la que se encuentra para verificar el correcto estado del mismo.	
<b>Condiciones de Ejecución:</b> El usuario tiene que estar autenticado con responsabilidad de Jefe de operaciones productivas AntyTer y Jefe del Departamento de Aseguramiento de la Calidad IFA Antyter.	
<b>Entradas/Pasos de Ejecución:</b> <ul style="list-style-type: none"> <li>- Selecciona la etapa</li> <li>- Revisa el proceso que ya está revisado para aprobar</li> <li>- Marca el botón aprobar</li> </ul>	
<b>Salidas/Resultado Esperado:</b> El sistema muestra el registro revisado con la opción aprobar.	
<b>Evaluación de la Prueba:</b> -	

### 3.3 Resultados de las pruebas

Cómo se definió en el epígrafe 3.1, se realizaron al sistema pruebas funcionales y de aceptación, utilizando el método de caja negra y la técnica de partición de equivalencia. A continuación, se describe el proceso seguido para la realización de las pruebas y los resultados obtenidos.

#### Resultados de las pruebas funcionales

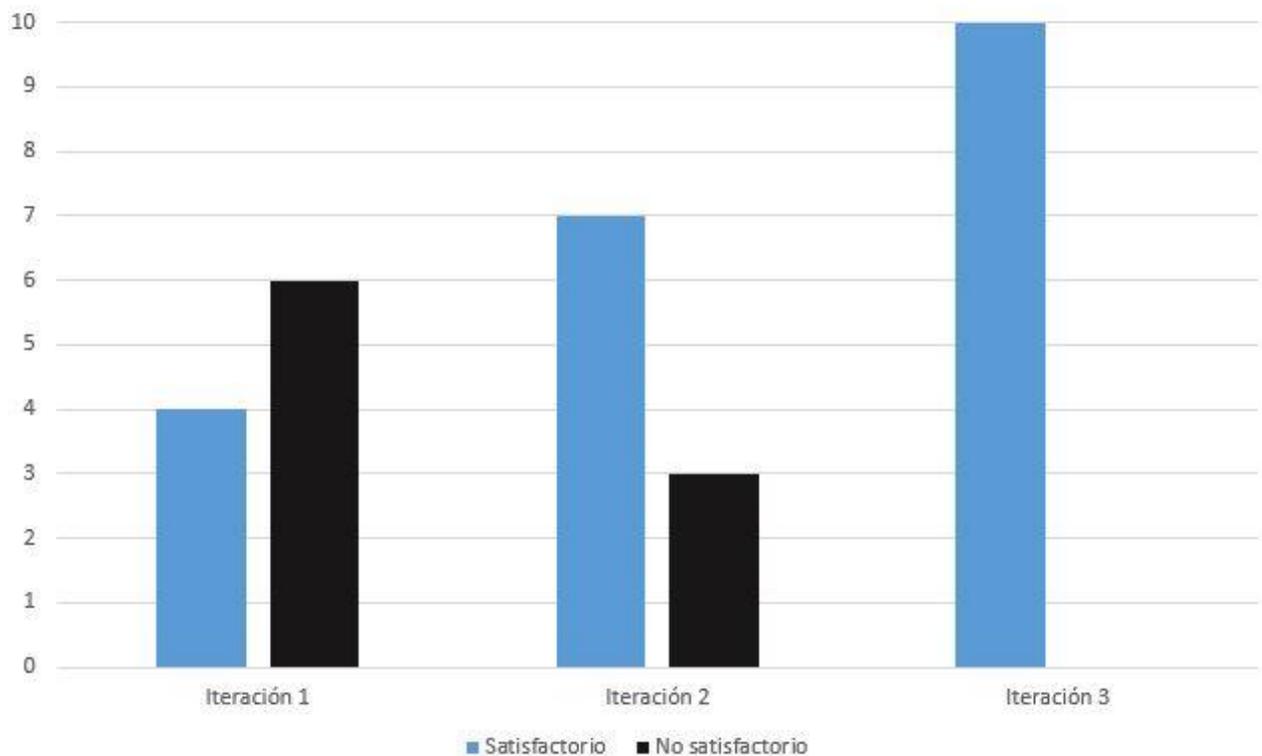
Para la aplicación de las pruebas funcionales se empleó el método de caja negra, estas pruebas se llevaron a cabo por parte de los desarrolladores, se ejecutaron sobre las interfaces gráficas del sistema haciendo uso de los casos de pruebas diseñados en el epígrafe 3.2.

En el caso de las pruebas funcionales fueron necesarias 3 iteraciones, se evaluaron todas las historias de usuario. La tabla 3.1 muestra los resultados obtenidos en cada iteración.

**Tabla 3.12:** Resultados de las pruebas funcionales realizadas al sistema

<b>Resultados</b>	<b>Iteración 1</b>	<b>Iteración 2</b>	<b>Iteración 3</b>
<b>Satisfactorios</b>	4	7	10
<b>No Satisfactorios</b>	6	3	0
<b>Total</b>	<b>10</b>	<b>10</b>	<b>10</b>

Dado que en la primera y segunda iteraciones hubo evaluaciones de casos de prueba que arrojaron resultados no satisfactorios, se hizo necesario realizar en total 3 interacciones para solucionar los errores detectados. Luego de estas 3 iteraciones se alcanzó la conformidad deseada y los resultados se muestran en la figura 3.1



**Figura 3.1:** Resultados de las pruebas funcionales realizadas al sistema.

Fuente: elaboración propia.

En las 3 iteraciones se probaron los 10 casos de pruebas, en la iteración 1 fueron satisfactorios 4 casos de prueba:

- modificar los datos referentes al proceso,
- ver los datos referentes al proceso,
- buscar los datos referentes al proceso por un criterio determinado y
- ordenar los datos referentes al proceso por un criterio determinado,

resultaron no satisfactorios 6 casos de prueba:

- eliminar los datos referentes al proceso,
- autenticar usuario,
- registrar usuario,
- registrar los datos referentes al proceso,
- revisar el registro asociado a cada etapa del proceso y
- aprobar el registro asociado a cada etapa del proceso.

En la iteración 2 fueron satisfactorios 7 casos de pruebas los 4 satisfactorios de la primera iteración y además los siguientes:

- eliminar los datos referentes al proceso,
- autenticar usuario y
- registrar usuario

se mantuvieron como no satisfactorios:

- registrar los datos referentes al proceso,
- revisar el registro asociado a cada etapa del proceso y
- aprobar el registro asociado a cada etapa del proceso.

En la iteración 3 fueron satisfactorios los 10 casos de pruebas y posteriormente se realizaron las pruebas de aceptación con el cliente.

### **Resultados de las pruebas de aceptación**

Las pruebas de aceptación son un tipo de ensayo que se realizan con el fin de verificar si el producto ha sido desarrollado de acuerdo con las normas y criterios establecidos y cumple con los requisitos especificados.

En el caso del sistema desarrollado se realizaron las pruebas de aceptación por parte del cliente, con la presencia del especialista Ing. Luis Alard Lage. Se utilizaron los casos de prueba diseñados en el epígrafe 3.2, los mismos utilizados para las pruebas funcionales. A través de su aplicación se comprobaron los requisitos funcionales. Adicionalmente se verificaron los requisitos no funcionales, descritos en el epígrafe 2.3.2 referentes a

- Apariencia o interfaz de usuario
- Usabilidad
- Soporte
- Portabilidad
- Confiabilidad
- Aspectos legales

El cliente evaluó el funcionamiento global del sistema y verificó el cumplimiento de los requisitos funcionales y no funcionales definidos inicialmente por él. Los problemas detectados durante las pruebas funcionales fueron solucionados antes de realizar las pruebas de aceptación. El sistema demostró ser completamente funcional, cumpliendo con los requisitos planteados. El cliente firmó el Acta de aceptación como constancia de su conformidad con el sistema desarrollado (anexo 2).

## Conclusiones parciales

Durante el desarrollo del capítulo se describieron las principales pruebas que se pueden realizar a los sistemas de software, se detallaron los casos de pruebas a realizar al sistema y, por último, se reflejaron los resultados de la ejecución de las pruebas realizadas para comprobar las funcionalidades del sistema. A través del análisis de los aspectos anteriores se puede concluir lo siguiente:

- Las pruebas que se realizan a los sistemas de software pueden clasificarse atendiendo a su naturaleza en pruebas unitarias, funcionales, de integración, de validación, de sistema, de aceptación con el cliente, entre otras categorías. En el caso de este trabajo se decidió realizar pruebas funcionales al sistema y pruebas de aceptación con el cliente, utilizando el método de pruebas de caja negra, descritas a partir de las historias de usuario.
- Se diseñaron 10 casos de pruebas a partir de las historias de usuario descritas en el capítulo 2 de este trabajo.
- Se realizaron 3 iteraciones de pruebas funcionales al sistema por parte de los desarrolladores y 1 iteración con el cliente final, dando como resultado la aceptación del sistema por parte del cliente.

## Conclusiones generales

- Con el desarrollo del presente trabajo se logró la implementación de un sistema informático para la digitalización de la recogida de datos en las etapas descongelación, expansión y fermentación del proceso de producción del IFA en el CIM, que contribuye a mejorar la gestión de la información asociada a dicho proceso.
- El diseño de un modelo de datos para la recolección de información relacionada al proceso de producción del IFA en el CIM, siguiendo los estándares definidos por MESA, proporciona la garantía del futuro desarrollo de un MES en el CIM.
- Se ratificó la validez de la propuesta a través de la aceptación del sistema por parte del cliente.

## Recomendaciones

- Implantar el sistema informático desarrollado y constatar las mejoras que se obtienen en la gestión de la información asociada al proceso de producción del IFA en el CIM.
- Completar la digitalización de la recogida de datos en los subprocesos de filtración y purificación.
- Implementar un módulo en el sistema que permita el análisis histórico de los datos almacenados y la generación de reportes personalizados.

## Bibliografía

1. Sistema de ejecución de manufactura - Tata Technologies - Mexico. En: [en línea]. Disponible desde: <http://www.tatatechnologies.mx/servicios/manufactura/sistema-de-ejecucion-de-manufactura/>.
2. Centro de Inmunología Molecular - EcuRed. En: [en línea]. Disponible desde: [https://www.ecured.cu/Centro\\_de\\_Inmunologia\\_Molecular](https://www.ecured.cu/Centro_de_Inmunologia_Molecular).
3. Método Científico. En: [en línea]. Disponible desde: [http://newton.cnice.mec.es/newton2/Newton\\_pre/3eso/mcientifico/index.htm](http://newton.cnice.mec.es/newton2/Newton_pre/3eso/mcientifico/index.htm).
4. DLE: sistema - Diccionario de la lengua española - Edición del Tricentenario. En: [en línea]. Disponible desde: <http://dle.rae.es/?id=Y2AFX5s>.
5. DLE: ejecución - Diccionario de la lengua española - Edición del Tricentenario. En: [en línea]. Disponible desde: <http://dle.rae.es/?id=ERtQ2Oa>.
6. DLE: ejecutar - Diccionario de la lengua española - Edición del Tricentenario. En: [en línea]. Disponible desde: <http://dle.rae.es/?id=ERxzMn2>.
7. DLE: manufactura - Diccionario de la lengua española - Edición del Tricentenario. En: [en línea]. Disponible desde: <http://dle.rae.es/?id=OIBold6>.
8. AGUDELO ROSERO, William. *Modelo para la integración de información de manufactura* [en línea]. S.l.: Universidad Nacional de Colombia-Sede Manizales, [sin fecha]. Disponible desde: <http://www.bdigital.unal.edu.co/52554/>.
9. SALAZAR, Veronique. Análisis de la Integración de los Sistemas MES–ERP en industrias de manufactura. En: *Seventh LACCEI Latin American and Caribbean Conference for Engineering and Technology* [en línea]. S.l.: s.n., 2009. [Consultado 16 marzo 2017]. Disponible desde: [http://laccei.org/LACCEI2009-Venezuela/Papers/IE207\\_Salazar.pdf](http://laccei.org/LACCEI2009-Venezuela/Papers/IE207_Salazar.pdf).
10. ELLIOTT, Riley F. Manufacturing Execution System (MES) An Examination of Implementation Strategy. En: [en línea]. 2013, Disponible desde: <http://digitalcommons.calpoly.edu/theses/997/>.
11. industria. En: [en línea]. Disponible desde: <http://www.manufactura.mx/industria>.
12. GRAUER, Manfred, KARADGI, Sachin S., METZ, Daniel y SCHÄFER, Walter. Online Monitoring and Control of Enterprise Processes in Manufacturing Based on an Event-Driven Architecture. En: *Business Process Management Workshops* [en línea]. S.l.: Springer, 2010. pp. 671–682. Disponible desde: <http://link.springer.com/content/pdf/10.1007/978-3-642-20511-8.pdf#page=665>.
13. MEYER, Heiko, FUCHS, Franz y THIEL, Klaus. *Manufacturing execution systems (MES): Optimal design, planning, and deployment*. S.l.: McGraw Hill Professional, 2009.
14. MESA International - MESA Model. En: [en línea]. Disponible desde: <http://www.mesa.org/en/modelstrategicinitiatives/MESAModel.asp>.

15. BOYD, Alan, NOLLER, D., PETERS, P., SALKELD, D., THOMASMA, T., GIFFORD, C., PIKE, S. y SMITH, A. Soa in manufacturing guidebook. En: *MESA International, IBM Corporation and Capgemini co-branded white paper*. 2008,
17. Sistema de Ejecución de Manufactura - MES - Tecnología de automatización - Siemens. En: [en línea]. Disponible desde: <http://www.industry.siemens.com/automation/aan/es/manufacturing-execution-systemes/pages/default.aspx>.
18. plantStruxure.pdf [en línea]. S.l.: s.n. Disponible desde: <http://www.schneider-electric.com.co/documents/local/xperience-efficiency/plantStruxure.pdf>.
19. Home. En: [en línea]. [Consultado 3 junio 2017]. Disponible desde: <http://www.werum.com/en/home/>. Werum IT Solutions is the world's leading supplier of manufacturing execution systems (MES) and manufacturing IT solutions for the pharmaceutical and biopharmaceutical industries. Our out-of-the-box PAS-X software product is run by 17 of the world's top 30 pharmaceutical and biotech companies but also by many mid-sized manufacturers.
20. CUP, Your First. An Introduction to Java Platform, Enterprise Edition. En: *Dostopno na: http://download.oracle.com/javaee/6/firstcup/doc/gkhoy.html*. 2011,
21. JENDROCK, Eric, EVANS, Ian, GOLLAPUDI, Devika, HAASE, Kim y SRIVATHSA, Chinmayee. *The Java EE 6 tutorial: basic concepts*. S.l.: Pearson Education, 2010.
22. COWARD, Danny. *Java EE 7: The Big Picture* [en línea]. S.l.: McGraw-Hill Education Group, 2014. ISBN 978-0-07-183733-0. Disponible desde: <http://dl.acm.org/citation.cfm?id=2829114>.
23. PECH-MAY, Fernando, GOMEZ-RODRIGUEZ, Mario A., LUIS, A. y LARA-JERONIMO, Salvador U. *Desarrollo de Aplicaciones web con JPA, EJB, JSF y PrimeFaces* [en línea]. S.l.: Tabasco, México: Instituto Tecnológico Superior de los Ríos, 2010. Disponible desde: <http://www.tamps.cinvestav.mx/~fpech/sd/files/paper001.pdf>.
24. Framework - EcuRed. En: [en línea]. Disponible desde: <https://www.ecured.cu/Framework>.
25. BERGSTEN, Hans. *JavaServer Faces: Building Web-based User Interfaces*. S.l.: O'Reilly Media, Inc., 2004. ISBN 978-1-4493-7895-0.
26. MANN, Kito D. *Java Server Faces in Action*. S.l.: Dreamtech Press, 2005.
27. FALKNER, Jayson y JONES, Kevin. *Servlets and JavaServer Pages: The J2EE Technology Web Tier* [en línea]. S.l.: Addison-Wesley, 2004. Disponible desde: <http://ic.info.unlp.edu.ar/wp-content/uploads/2016/07/Java-y-aplicaciones-avanzadas-sobre-internet.pdf>.
28. primefaces\_user\_guide\_5\_0.pdf [en línea]. S.l.: s.n.. Disponible desde: [https://www.primefaces.org/docs/guide/primefaces\\_user\\_guide\\_5\\_0.pdf](https://www.primefaces.org/docs/guide/primefaces_user_guide_5_0.pdf).
29. Why PrimeFaces. En: *PrimeFaces* [en línea]. Disponible desde: <https://www.primefaces.org/whyprimefaces/>.

30. NetBeans IDE: Roadmap, Release Notes, System Requirements, Installation Instructions. En: [en línea]. Disponible desde: <https://netbeans.org/community/releases/>.
31. NetBeans IDE 8.2 Release Information. En: [en línea]. Disponible desde: <https://netbeans.org/community/releases/82/>.
30. NetBeans IDE: Roadmap, Release Notes, System Requirements, Installation Instructions. En: [en línea]. . Disponible desde: <https://netbeans.org/community/releases/>.
33. Manual de Glassfish - M-ESPEL-0017.pdf [en línea]. S.l.: s.n. Disponible desde: <https://repositorio.espe.edu.ec/bitstream/21000/4431/1/M-ESPEL-0017.pdf>.
34. GlassFish - EcuRed. En: [en línea]. Disponible desde: <https://www.ecured.cu/GlassFish>.
35. MySQL | Apli Argentina. En: [en línea]. Disponible desde: <https://www.apli.com.ar/node/74>.
36. Capítulo 1. Información general. En: [en línea]. Disponible desde: <http://det.bi.ehu.es/practicassQL/manual5/introduction.html>.
37. Memoria.pdf [en línea]. S.l.: s.n. . Disponible desde: <https://riunet.upv.es/bitstream/handle/10251/17077/Memoria.pdf?sequence=1>.
38. RUIZ, Irene Luque y GOMEZ-NIETO, Miguel Angel. Bases De Datos Desde Chen Has Codd Con Oracle. S.l.: Alfaomega Grupo Editor, 2002. ISBN 978-970-15-0760-5.
39. RAMÍREZ, Frank Jibaja. Metodologías para el desarrollo de software. En: [en línea]. [Consultado 5 junio 2017]. Disponible desde: [http://www.academia.edu/9953322/Metodologias\\_para\\_el\\_desarrollo\\_de\\_software](http://www.academia.edu/9953322/Metodologias_para_el_desarrollo_de_software). Metodologías para el desarrollo de software
40. *METODOLOGIAS AGILES.pdf* [en línea]. S.l.: s.n. [Consultado 3 junio 2017]. Disponible desde: <http://ingenieriadesoftware.mex.tl/images/18149/METODOLOGIAS%20AGILES.pdf>.
41. Metodología UCI. En: [en línea]. Disponible desde: <https://excriba.prod.uci.cu/page/context/shared/document-details?nodeRef=workspace://SpacesStore/a622adab-eac5-4fb3-ba08-a266767fff5f>.
42. SOMMERVILLE, Ian. *Ingeniería del software*. S.l.: Pearson Educación, 2005. ISBN 978-84-7829-074-1.
43. DUARTE, Ailin Orjuela y C, Mauricio Rojas. LAS METODOLOGÍAS DE DESARROLLO ÁGIL COMO UNA OPORTUNIDAD PARA LA INGENIERÍA DEL SOFTWARE EDUCATIVO. En: *Avances en Sistemas e Informática* [en línea]. 1 mayo 2008, Vol. 5, no. 2. Disponible desde: <http://revistas.unal.edu.co/index.php/avances/article/view/10037>.
44. GONZÁLEZ, Yanette Díaz y ROMERO, Yenisleidy Fernández. Patrón Modelo-Vista-Controlador. En: *Revista Telem@tica*. 2012, Vol. 11, no. 1, pp. 47–57.
45. VISCONTI, Marcello y ASTUDILLO, Hernán. Fundamentos de Ingeniería de Software. En: [en línea]. 2012, Disponible desde: <http://roa.ult.edu.cu/bitstream/123456789/401/1/08-Patrones.pdf>.

46. Diagrama de despliegue - EcuRed. En: [en línea]. Disponible desde: [https://www.ecured.cu/Diagrama\\_de\\_despliegue](https://www.ecured.cu/Diagrama_de_despliegue).
47. TUYA, Javier, ROMÁN, Isabel Ramos y COSÍN, José Javier Dolado. *Técnicas cuantitativas para la gestión en la ingeniería del software* [en línea]. S.l.: NetBiblo, 2007. Disponible desde: <https://www.google.com/books?hl=es&lr=&id=PZQoZ9KTNaEC&oi=fnd&pg=PR15&dq=T%C3%A9cnicas+cuantitativas+para+la+gesti%C3%B3n+en+la+ingenier%C3%ADa+del+software&ots=P-1L1HebD-&sig=3Nm5pg2TAD7toqSciK505CQZ5Vs>.
48. AMO, Fernando Alonso, NORMAND, Loïc Martínez y PÉREZ, Francisco Javier Segovia. *Introducción a la ingeniería del software*. S.l.: Delta Publicaciones, 2005. ISBN 978-84-96477-00-1.
49. GÓMEZ, JOSÉ LUIS BERENGUEL. *UF1844 - Desarrollo de aplicaciones web en el entorno servidor*. S.l.: Ediciones Paraninfo, S.A., 2015. ISBN 978-84-283-9717-9. Google-Books-ID: gVGACwAAQBAJ
50. Pruebas de software - EcuRed. En: [en línea]. Disponible desde: [https://www.ecured.cu/Pruebas\\_de\\_software](https://www.ecured.cu/Pruebas_de_software).
51. LÓPEZ, Francisco Javier Moliner. *Informaticos Generalitat Valenciana. Grupos a Y B. Temario Bloque Especifico Volumen I.e-book*. S.l.: MAD-Eduforma, [sin fecha]. ISBN 978-84-665-2077-5.
52. GUILLERMO, PANTALEO y RINAUDO, Ludmila. *Ingeniería de Software*. S.l.: Alfaomega Grupo Editor, 2015. ISBN 978-987-1609-78-9. Google-Books-ID: a8j2DQAAQBAJ
53. SAN MIGUEL, José Talledo. *MF0493\_3-Implantación de aplicaciones web en entorno internet, intranet y extranet* [en línea]. S.l.: Ediciones Paraninfo, SA, 2015. Disponible desde: [https://www.google.com/books?hl=es&lr=&id=RtESCgAAQBAJ&oi=fnd&pg=PA1&dq=MF0493\\_3+-+Implantaci%C3%B3n+de+aplicaciones+web+en+entorno+internet,+intranet+y&ots=2fyIbVcC1q&sig=RIJj0Es7NtJ0OaPnDbuO2g7CeQ](https://www.google.com/books?hl=es&lr=&id=RtESCgAAQBAJ&oi=fnd&pg=PA1&dq=MF0493_3+-+Implantaci%C3%B3n+de+aplicaciones+web+en+entorno+internet,+intranet+y&ots=2fyIbVcC1q&sig=RIJj0Es7NtJ0OaPnDbuO2g7CeQ).
54. FOSSATI, Matias. *Testing: Convertite en un experto probando software*. S.l.: Natsys, 2017. Google-Books-ID: \_kQ1DgAAQBAJ
55. Sesión 5. Implementación del sistema de información. En: *calameo.com* [en línea]. Disponible desde: <http://www.calameo.com/read/000745288d788137e29fc>.
56. Pruebas de caja negra - EcuRed. En: [en línea]. Disponible desde: [https://www.ecured.cu/Pruebas\\_de\\_caja\\_negra#cite\\_ref-1](https://www.ecured.cu/Pruebas_de_caja_negra#cite_ref-1).
57. GUERRERO, Carlos A.; SUÁREZ, Johanna M.; GUTIÉRREZ, Luz E. Patrones de Diseño GOF (The Gang of Four) en el contexto de Procesos de Desarrollo de Aplicaciones Orientadas a la Web. *Información tecnológica*, 2013

# Anexos

## Anexo 1. Muestra de una de las planillas utilizadas actualmente en CIM.

Planilla A1REG-0036. Descongelación y expansión de células productoras de anticuerpos monoclonales.



AIREG-0036  
VERSIÓN 03  
PÁGINAS 1/4  
ANTERIOR A1REG-0036 V01

### *DESCONGELACIÓN Y EXPANSIÓN DE CÉLULAS PRODUCTORAS DE ANTICUERPOS MONOCLONALES*

Tabla#1: Identificación del proceso	
Lote de descongelación	/DC _____ / _____
No. de parte de la línea celular	
Código del ampulla	/ /

Tabla#2: Identificación del personal		
Nombre y apellidos	Responsabilidad	Firma

Tabla#3: Identificación del equipamiento e instrumentos		
Nombre del equipo y/o instrumento	Código	Fecha vencimiento calibración/calificación
Flujo laminar		
Incubadora de CO <sub>2</sub>		
Incubadora de 37°C		
Incubadora de 37°C		
Refrigerador		
Micropipeta		
Centrifuga refrigerada		
Firma:		Fecha:

Tabla#4: Datos del ampulla a descongelar			
Del ampulla a descongelar registre los siguientes datos. Auxiliarse del registro de tanques de nitrógeno.			
Tanque de nitrógeno No.		Rack	Caja
No. del ampulla		Código del ampulla	/ /
Observaciones:			
Firma:		Fecha:	

**DESCONGELACIÓN Y EXPANSIÓN DE CÉLULAS PRODUCTORAS DE ANTICUERPOS MONOCLONALES**

Tabla#5: Descongelación		
		<b>Realizado</b>
Vierta entre 10 y 15 mL de medio de cultivo en un tubo de 50 mL.		<input type="checkbox"/> Sí <input type="checkbox"/> No Vol. medido (mL):
Extraiga del tanque de nitrógeno el ampulita a descongelar, atempérela ligeramente entre las manos y colóquela dentro del beaker con el agua a $37 \pm 1$ °C.		<input type="checkbox"/> Sí <input type="checkbox"/> No Hora:
Agite el vial lentamente hasta que no exista hielo visible.		<input type="checkbox"/> Sí <input type="checkbox"/> No
Bajo el flujo laminar extraiga el contenido del ampulita con una pipeta de 5mL que contenga 3ml de medio de crecimiento.		<input type="checkbox"/> Sí <input type="checkbox"/> No
Deposite la suspensión celular en un tubo de 50 mL que contiene el medio de crecimiento.		<input type="checkbox"/> Sí <input type="checkbox"/> No
Para eliminar el dimetilsulfóxido (DMSO), resuspenda las células en todo el volumen de medio de cultivo y centrifugue durante a 900 rpm y 10 min.		<input type="checkbox"/> Sí <input type="checkbox"/> No
Tome una muestra de 1 mL y realice el conteo en la cámara de Neubauer.		<input type="checkbox"/> Sí <input type="checkbox"/> No Hora:
Tiempo total del proceso de descongelación $\leq 20$ min.		min.
Lote de medio empleado	Fecha de liberación	Fecha de vencimiento
Observaciones:		
Firma:		Fecha:
Firma:		Fecha:

**DESCONGELACIÓN Y EXPANSIÓN DE CÉLULAS PRODUCTORAS DE ANTICUERPOS MONOCLONALES**

Tabla#6: Conteo celular			
<b>Leyenda:</b>			
$X_v$ : Concentración de células vivas. (cél/mL)		vol.: Volumen de la suspensión. (mL)	
$X_d$ : Concentración de células muertas. (cél/mL)		% Viab. : Por ciento de viabilidad en el cultivo analizado. (%)	
$X_t$ : Concentración de células totales. (cél/mL)		Cél muertas : Total de células muertas en cuatro cuadrantes. (cél)	
Dil.: Dilución de trabajo.		No. cél totales : Cantidad total de células en el cultivo (cél)	
No. de cuadrantes	Células vivas	Células muertas	Cálculos
1			$X_v = \text{Cél vivas}/4 * \text{Dil.} * 10^4 =$
2			$X_d = \text{Cél muertas}/4 * \text{Dil.} * 10^4 =$
3			$X_t = X_v + X_d =$
4			No células totales = $X_t * \text{vol} =$
Dilución:			$\% \text{ Viab} = X_v/X_t * 100 =$ Volumen total en que se encuentra la suspensión celular: Vol =
No. de cuadrantes	Células vivas	Células muertas	Cálculos
1			$X_v = \text{Cél vivas}/4 * \text{Dil.} * 10^4 =$
2			$X_d = \text{Cél muertas}/4 * \text{Dil.} * 10^4 =$
3			$X_t = X_v + X_d =$
4			No células totales = $X_t * \text{vol} =$
Dilución:			$\% \text{ Viab} = X_v/X_t * 100 =$ Volumen total en que se encuentra la suspensión celular: Vol =
Realizado por:			Firma:
Supervisado por Aseguramiento de la Calidad:			Firma:



**DESCONGELACIÓN Y EXPANSIÓN DE CÉLULAS PRODUCTORAS DE ANTICUERPOS MONOCLONALES**

ELABORADO POR/ Responsabilidad/	<u>Ing. Analey Segreo Erbite</u> <u>J' Etapa Expansión Celular, Antyter</u>	FIRMA/ _____ FECHA/ _____
REVISADO POR/ Responsabilidad/	<u>MsC. Galia Labrada Milanés</u> <u>Documentadora AntyTer</u>	FIRMA/ _____ FECHA/ _____
REVISADO POR/ Responsabilidad/	<u>Lic. Maria Esther Martinez López</u> <u>Especialista Aseguramiento de la Calidad AntyTer</u>	FIRMA/ _____ FECHA/ _____
APROBADO POR/ Responsabilidad/	<u>MsC. Yailen Rivero Barreiro</u> <u>J' Operaciones Productivas</u>	FIRMA/ _____ FECHA/ _____
APROBADO POR/ Responsabilidad/	<u>MsC. Ma. Ofelia Mollinedo Izquierdo</u> <u>J' Dpto. Aseguramiento de la Calidad IFA Antyter</u>	FIRMA/ _____ FECHA/ _____

**Anexo 2. Acta de aceptación.**