

# Análisis de riesgos para el diseño de arquitectura multilingaje de un módulo mini-PFAL (Plant Factory with Artificial Lighting) con enfoque de Ingeniería de Software

Risk analysis for the multilanguage architecture design of a mini-PFAL (Plant Factory with Artificial Lighting) module with a Software Engineering approach

Recepción del artículo: 23/03/2023 • Aceptación para publicación: 07/06/2023 • Publicación: 30/06/2023

● <https://doi.org/10.32870/ecucba.vi20.296>

Luis Tonatiuh Castellanos Serrano\*

María Victoria Gómez Águila

José Alfredo Castellanos Suárez

Universidad Autónoma Chapingo. Centro de Investigaciones Interdisciplinarias y de Servicio en Ciencia, Naturaleza, Sociedad y Cultura (CIISCINACyC). México-Texcoco Chapingo, Edo. México. México.

\*Autor para correspondencia: procesoslce@hotmail.com

## Resumen

Las *Plant Factory* son tecnologías que han empezado a detonarse desde 1980 y con el pasar de los años han evolucionado de forma asombrosa dotándose de herramientas tecnológicas que impulsan sus capacidades como son la Big Data, Internet de las Cosas (IoT), cómputo ubicuo, entre otras. El software que opera los algoritmos de control para el funcionamiento de las estaciones de microambientes controlados, es una parte medular imprescindible para su óptimo funcionamiento, de este modo las herramientas para la administración oportuna del diseño de software son agentes claves para garantizar un desarrollo oportuno de las interfaces. El presente trabajo expone el estudio de caso del análisis de riesgos para un módulo mini PFAL, donde se establecen el análisis de requerimientos, exposición de los resultados técnicos del sistema PFAL alcanzados, identificación de la problemática del proyecto, proceso de administración del riesgo, proceso de proyección del riesgo; y finalmente se concluye con ayuda de gráficas de dispersión y superposición, los riesgos latentes del sistema que se proponen ser valorados para el éxito del proyecto con miras a formar parte de un plan de mitigación, monitoreo y manejo de riesgo (MMMR) en fases futuras.

**Palabras clave:** Programación, identificación, agricultura, fábrica de plantas, administración de riesgos.

## Abstract

The Plant Factory are technologies that have begun to be detonated since 1980 and over the years have evolved in an amazing way, providing technological tools that boost their capabilities such as Big Data, IoT, ubiquitous computing, among others. The software that operates the control algorithms for the operation of the controlled microenvironment stations, is an essential core part for its optimal operation, thus the tools for the timely administration of the software design are key agents to guarantee a timely development of the interfaces. The present work exposes the case study of the risk analysis for a mini PFAL module, where the requirements analysis is established, exposition of the technical results of the PFAL system achieved, identification of the project problem, risk management process, process risk projection; and finally, with the help of scatter and superimposition graphs, the latent risks of the system that are proposed to be assessed for the success of the project with a view to being part of a risk mitigation, monitoring and management plan (MMMR) in future phases are concluded.

**Keywords:** Programming, identification, agriculture, plant factory, risk management.

## Introducción

El termino PFAL alude a sus siglas en ingles *Plant Factory with Artificial Lighting* (Fabricas de Plantas con Luz Artificial), estos dispositivos empiezan a datar sus primeras experiencias en los años de 1980 (Kozair *et al.*, 2015) y hasta la fecha siguen presentando una evolución constante en las técnicas para mejorar su operatividad.

Los PFAL son ambientes controlados que pueden ser micro hasta macrosistemas, se caracterizan por que intervienen cuatro factores principales: Uso de luz artificial (hoy en día popularmente luces Led); Verticalidad; Sistemas de riego eficientes como pueden ser hidropónico, acuapónico y aeropónico; Controles para la administración de variables de instrumentación de sensores y actuadores (Costa *et al.*, 2020)

Los PFAL han mutado desde sus inicios como sistemas altamente tecnificados, las tecnologías del del IoT, bigdata, data lake y data warehouse, etcétera, están formando parte de la industria de los PFAL hoy en día, creando redes de hosting más complejas que permiten monitorear los procesos en tiempo real con ayuda de App vinculadas con el Smartphone, transformando así la industria de la agricultura digital en una nueva era (Goto, 2012).

Por otro lado, la infraestructura de las Plant Factory no solo se delimita a la construcción de los soportes estáticos, conexión hidráulicas, actuadores, sensores, instrumentación y sistemas electrónicos inteligentes sobre todo actualmente el trabajo de programación que en su desarrollo requiere de herramientas de análisis, gestión y administración como es la metodología de la ingeniería de software

El presente artículo tiene como objetivo realizar el análisis de los requisitos de programación necesarios para el diseño de las arquitecturas de programación, la problemática y los resultados técnicos obtenidos hasta el momento, para implementar con la teoría de administración de riesgo en software (Mohapatra, 2010) y poder así concluir cuáles son los riesgos prioritarios que se deben de atender, para ello se expondrá la metodología para la cuantificación de los datos, con procesos de interpolación y superposición de la información del análisis de riesgos, para concluir el orden en función del impacto que los riesgos deben

atenderse para tener éxito en la construcción del software orientado al diseño de una mini Plant Factory IoT.

## Materiales y Métodos

### Análisis de requerimientos del software

El software en su conjunto cuenta con tres etapas de arquitectura:

- 1. Sistema embebido:** Microsistema operativo que realiza la gestión autónoma de la máquina, se dedica al administración y gestión de los datos, instrumentación, actuadores y sistemas de comunicación
- 2. Aplicación de escritorio:** Su función principal es ser un servicio de host local para el almacenamiento y gestionar la conexión con la base de datos operativa de las variables recolectadas, con la conectividad WIFI poder acceder a los servicios de WebService para conectar los datos a la nube
- 3. Aplicación Móvil:** Dashbord gráfico e interactivo que permite la administración remota de la máquina a través de conectividad bluetooth o WIFI, de este modo el usuario cuenta con aplicación pocket que puede interactuar como interfaz de herramientas de internet de las cosas (Castellanos, 2022)

Cuadro 1. Descripción de caso de uso programación de sistema embebido.

Caso uso	Especificaciones
Actor	Usuario y sistema automatizado
Descripción	Firmware principal de control para los procesos de riego, sensado de variables y horarios de iluminación, que funja como sistema operativo de control principal de la máquina. Despliegue de información en pantalla Touchpad tipo TFT de 3.5" como interfaz gráfica de salida de información con gráficos RGB y dashboard flexible y de fácil manejo, la interfaz touch deberá funcionar como interfaz de entrada para los ajustes operativos de la máquina. Interfases de comunicación: 1. USB 2. Bluetooth 3. WIFI
Precondiciones	Firmware basado en programación C++ susceptible a ser grabado en un microcontrolador Microchip Atmega 2560 (Microchip, 2022) 1. Despliegue de información de sensores (°C/%RH) 2. Configuraciones o Opciones avanzadas: Testing para evaluación de actuadores e instrumentación o Riego: Ajustes de ciclos de riego
Actividades	o Iluminación: Ajustes de horarios de encendido y apagado de luminarias Led's o Hora: Ajuste de hora de módulo reloj interno o Sensores: Testing de matriz de sensores 3. Graficación de información (Menú que permite desplegar la información de forma gráfica)

Para realizar la propuesta de diseño, se emplearon las tablas de caso de uso para poder realizar la descripción exacta y oportuna de los requerimientos del software (para la Plant Factory), y con ello poder realizar la trazabilidad de los requerimientos necesarios para su diseño. En los Cuadros 1, 2 y 3 se exponen los detalles de cada arquitectura.

Cuadro 2. Descripción de caso de uso de la aplicación de escritorio general.

Caso uso	Especificaciones
Actor	Usuario
Descripción	Interfaz Gráfica de Usuario (GUI) con soporte en sistema operativo Windows basado en lenguaje Visual Basic (preferentemente) o JAVA, entregado en formato de instalador EXE que contempla instalación de complementos como son los sockets de conexión a Webservice y complementos de base de datos local
Precondiciones	Compatible con sistema operativo Windows 10 (mínimo) con .NET Framework 4.8, HDD de 500 GB y memoria RAM de 8 GB, procesador no se delimita a ninguna marca o modelo específico
Actividades	<ol style="list-style-type: none"> <li>1. Formulario principal con despliegue de información relevante (datos de sensores, historiales, gráficas de información, configuraciones del sistema, etcétera)</li> <li>2. Conexión con base de datos local SQL (preferentemente) o Access para resguardo de historial de variables</li> <li>3. Conectividad con Webservice para el alojamiento y vinculación de datos en la nube</li> </ol>

Bajo el análisis exhaustivo de los Cuadros 1, 2 y 3 de análisis de requerimientos de cada arquitectura respectivamente, en la Figura 1 se realiza el trazado del flujograma de los requerimientos de las diferentes arquitecturas de software que debe soportar el sistema mini PFAL para su íntegra operatividad:

Cuadro 3. Descripción de caso de uso de aplicación móvil de Smartphone.

Caso uso	Especificaciones
Actor	Usuario
Descripción	Dashboard de usuario para Smartphone visualmente atractiva, despliegue de datos dinámicos, gráficas interactivas e interfaz de fácil manejo, con opción a futuro de poder descargar desde la Play Store®
Precondiciones	Sistema operativo Android 6.0 Marshmallow o superior, preferentemente con 8 GB de RAM y 16 GB de almacenamiento interno, con procesador de soporte de funciones básicas
Actividades	<ol style="list-style-type: none"> <li>1. Menú Principal: Opciones para realizar una navegación de las dashboard para el despliegue de información y configuración del sistema embebido</li> <li>2. Graficación: Dashboard visualmente atractiva y dinámica para el despliegue de datos en formato de gráficos interactivos</li> <li>3. Configuraciones: Envío de datos de configuración para el sistema microcontrolado para el testing y ajuste de parámetros operativos de la maquina por conectividad bluetooth o WIFI</li> <li>4. Administración de BD: Gestionar la interconectividad WIFI para conexión con el Webservice para la consulta remota de la BD local y en nube</li> </ol>

## Resultados y Discusión

### Resultados técnicos del sistema PFAL

Los avances alcanzados hasta el punto actual en los sistemas mecánicos, electrónicos y programación congregan un módulo mini PFAL con las siguientes características:

1. Módulo ortométrico con dimensiones 81.5x76.5x92 cm
  - a. Estructura metálica
  - b. Maderas tratadas para humedad
  - c. Acrílicos

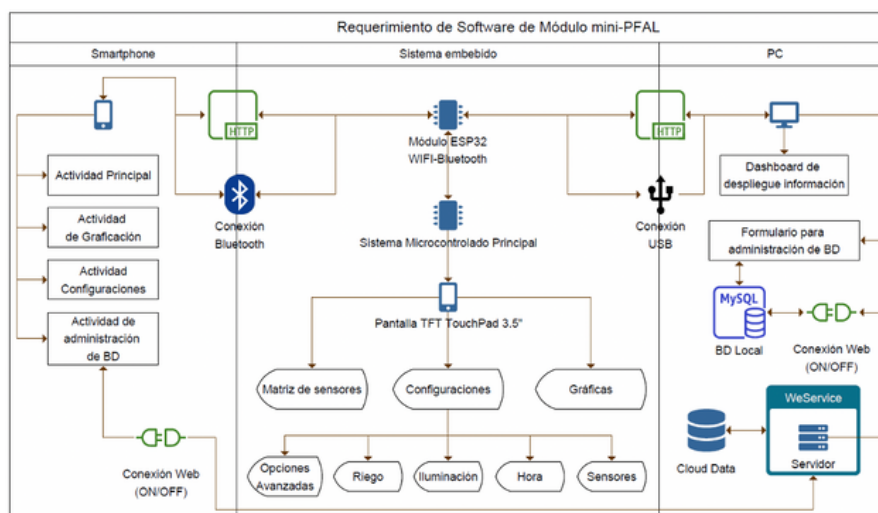


Figura 1. Flujograma de información de las etapas de software solicitadas para el diseño operativo del mini-PFAL (Autoría Propia con apoyo del software y ED live©)

2. Matriz de 5 sensores para recolección de temperatura, humedad y luminiscencia
  - a. Diseño de PCB
  - b. Montaje
  - c. Programación orientada hardware
3. Sistema hidropónico
  - a. Canastilla de siembra
  - b. Protector de raíces
  - c. Boquilla de nebulización
  - d. Contenedor de sustancia nutritiva
  - e. Circuito de alimentación hidráulica
  - f. Circuito de retroalimentación hidráulica
4. Sistema electrónico de control
  - a. Programación orientada hardware
    - i. Firmware del Sistema Operativo (etapa en constante cambio y ajustes)
    - ii. Controladores PID (etapa en experimentación para ajuste variables  $K_p$ ,  $K_i$  y  $K_d$ )
    - iii. Interfaz gráfica de usuario (etapa en constante cambio y ajustes)
    - iv. Protocolos de comunicación (etapa en afinación)
  - b. Interfaz de usuario con pantalla Touch Pad TFT 320x480
  - c. Sistema de multiplexación para sensores (etapa en diseño)
  - d. Sistema de potencia (etapa en ajustes y correcciones)
    - i. Control de riego (en proceso de rediseño)
    - ii. Control de temperatura (en proceso de rediseño)
    - iii. Control de ventilación (en proceso de rediseño)
  - e. Sistema de alimentación (etapa en ajustes)

Algunos materiales como las maderas tratadas con barnices y pinturas especiales para la humedad, sistemas electrónicos montados en protoboard, cableado, etcétera, serán sustituidos hasta ir superando las etapas de experimentación y pre-diseño.

Los avances hasta este punto han definido componentes claves para el diseño de las primeras fases experimentales, pero no son absolutas, la experiencia de armado conduce a conclusiones técnicas y empíricas para realizar cambios y modificaciones en ciertas secciones, durante todo

el proceso de ensamble, en la metodología se trazaron 3 fases, y en la práctica se vuelve un bucle entre la fase de planificación y la fase de ejecución, esto conlleva a constantes modificaciones para regular el óptimo control de las variables dependientes e independientes de todo el micro ambiente controlado los resultados se puede observar en la Figura 2 a continuación:

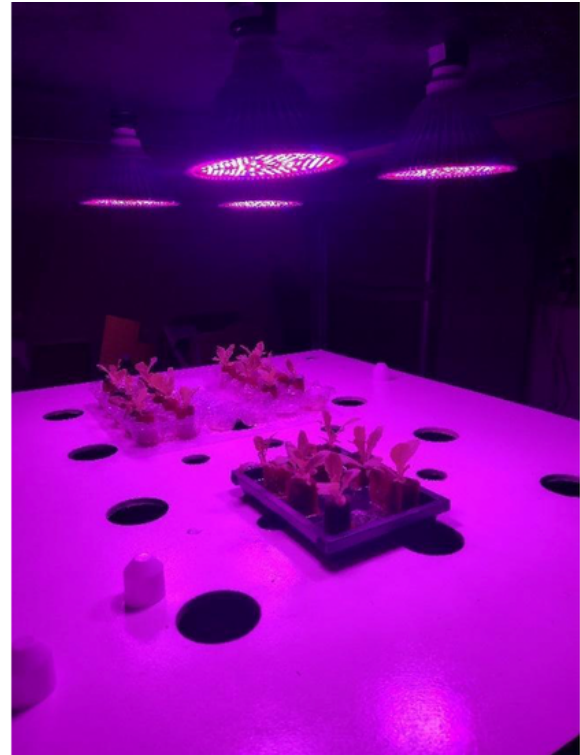


Figura 2. Evidencia fotográfica de microambiente artificial controlado.

En la Figura 2 se puede apreciar algunos círculos negros que representan los contenedores de crecimiento donde la raíz será expuesta al riego aeropónico automatizado, también las luminarias Led's para crecimiento fotosintético de la planta y, finalmente, los contenedores de plántulas, el experimento que se evidencia evalúa la longitud de onda y luminosidad de las luminarias, por otro lado el sistema de control (software-hardware) se encuentran aún en etapa de diseño y mejora continua.

Como parte de la continuación del proyecto es importante concluir el módulo mini PFAL con los resultados alcanzado hasta el momento y replicar otro módulo para comprobar la verticalidad del proyecto, un punto sumamente importante que caracteriza a estos dispositivos.



### Avance hacia la solución de problemas identificados previamente

Para realizar un análisis minucioso se han establecido los criterios de la Cuadro 4, donde se enlistan las tareas ejecutadas hasta el momento y el porcentaje alcanzado en cada una:

Cuadro 4. Porcentaje de tareas completadas en diseño electrónico y mecánico

Tarea	Porcentaje
Investigación bibliográfica	90%
Cotización de insumos, herramientas y maquinaria especializada	100%
Compras y facturación	90%
Simulación mecánica	50%
Simulación electrónica	60%
Diseño de programación	60%
Diseño de piezas	60%
Impresión 3D y fabricación de piezas	50%
Ensamblado de prototipo	50%
Pre-prototipo	100%
Pruebas, correcciones y ajustes	70%
Media ponderada de alcances:	69%

Dado que el propósito principal del objeto de estudio son las tareas pendientes del proyecto centrado en la temática de software, se propone la siguiente lista de ejes claves para alcanzar los requerimientos del sistema:

Cuadro 5. Porcentaje de tareas completadas en diseño de software

Tarea	Porcentaje
<ul style="list-style-type: none"> <li>Reconfiguración de matriz de lectura de sensores</li> </ul>	65%
<ul style="list-style-type: none"> <li>Graficación de usuario</li> </ul>	
<ul style="list-style-type: none"> <li>Almacenamiento de información en SD</li> </ul>	
Vinculación de puerto COM con interfaz JAVA	100%
Diseño de GUI JAVA y conexión por puerto COM	40%
Diseño de BD Local	50%
Diseño y vinculación de aplicación móvil para administración remota por protocolo WIFI y Bluetooth	55%
Diseño de Dashboard móvil para despliegue de información con atractivo visual	30%
Media ponderada de alcances:	57%

### Problemas o problemáticas potenciales desde el inicio del proyecto

1. Recursos humanos en programación:
  - 1.1. Orientada a hardware (C++)
  - 1.2. JAVA
  - 1.3. SQL
  - 1.4. HTML
  - 1.5. Android Studio
2. Recursos humanos en diseño electrónico
  - 2.1. Diagramas electrónicos
  - 2.2. Simulación
  - 2.3. Diseño de PCB
  - 2.4. Manufactura de PCB's
3. Recursos humanos en diseño mecánico aplicado
  - 3.1. Programas CAM/CAD
4. Presupuesto
5. Logística
6. Administración

### Acciones correctivas planeadas

1. Tiempo de entrega de los avances de módulos de programación
2. Ajuste de horarios de trabajo
3. Cumplimiento del cronograma de actividades
4. Entrega de reportes parciales

### Administración del riesgo

Con el estudio de los puntos anteriormente descritos, se puede trazar un panorama para realizar un análisis de riesgos en el diseño de software de un módulo mini PFAL con características descritas en el apartado de "Análisis de requerimientos del software".

Para la *administración del riesgo* se aplicará la taxonomía de riesgos y observación, la cual está basada en elementos o características de los proyectos de software que permiten ser indicadores para la identificación oportuna de los riesgos en estos ámbitos (Torres, 2022), de este modo Patricia Torres enlista 6 puntos importantes análisis de este ámbito:

- Complejidad tecnológica
- Entorno organizacional
- Equipo de trabajo
- Planificación y control
- Requerimientos
- Usuarios

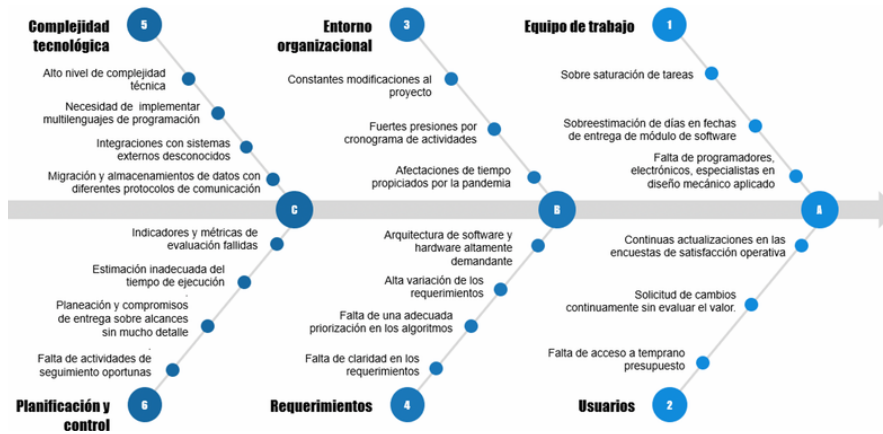


Figura 3. Análisis de riesgos en el software de prototipo mini PAFL.

Aplicando el Diagrama de Ishikawa o también conocido como diagramas de espina de pescado (Kern, 2021), se realiza el análisis empalmándolo bilateralmente con la taxonomía de riesgos y observación, lo que proporciona para propósitos del análisis de riesgo de software la Figura 3.

### Proyección del Riesgo

La proyección del riesgo, también llamada estimación del riesgo intenta calificar cada riesgo en dos formas: 1) la posibilidad o probabilidad de que el riesgo sea real y 2) las consecuencias de los problemas asociados con el riesgo, en caso de que ocurra.

Proceso de proyección del riesgo:

- Establecer una escala que refleje la probabilidad percibida de un riesgo.
- Delinear las consecuencias del riesgo.
- Estimar el impacto del riesgo sobre el proyecto y el producto.
- Valorar la precisión global de la proyección del riesgo de modo que no habrá malos entendidos (Pressman, 2010, p. 644)

Al emplear como instrumento de la matriz de valoración del impacto de B. W. Boehm en 1986 establece como parámetros Categorías (Catastrófico-Crítico-Marginal-Despreciable) versus componentes (Rendimiento-Apoyo-Costos-Calendario), se puede entonces realizar la elaboración de la tabla de riesgos que se presenta a continuación:

Cuadro 6: Riesgos de diseño de software de prototipo de mini PFAL IoT (Pressman, 2010).

#	Riesgos	Categoría	Probabilidad	Impacto	Preocupación administrativa
1	Fecha de entrega temprana	Tecnología por construir	90%	3	2
2	Pérdida de fondos	Impacto empresarial	10%	1	1
3	Tiempo de pagos a proveedores	Entorno de desarrollo	60%	2	2
4	Cambios en requisitos del Cliente	Impacto empresarial	50%	3	3
5	Tecnología no será satisfactoria a las expectativas	Entorno de desarrollo	10%	2	1
6	Personal inexperto	Tamaño y experiencia del personal	15%	3	4
7	Perdida de personal	Tamaño y experiencia del personal	30%	1	3
8	Multilinguaje de paradigma de programación	Tecnología por construir	40%	3	4
9	Interfaces y Dashboard poco atractivas	Entorno de diseño	20%	2	4

Cuadro 7. Cuantificación de los valores del impacto del riesgo.

Valores de impacto y Preocupación			
Valor	Impacto	Preocupación administrativa	Ponderación
1	Catastrófico	Muy alta	1
2	Crítico	Alta	0.75
3	Marginal	Moderado	0.50
4	Menores	Bajo	0.25
5	Despreciable	Muy bajo	0.00

Interpolando los datos de la proyección de riesgos podemos obtener el siguiente gráfico de “riesgo-preocupación-administrativa” que se aprecia en el gráfico 1 en formato de dispersión:

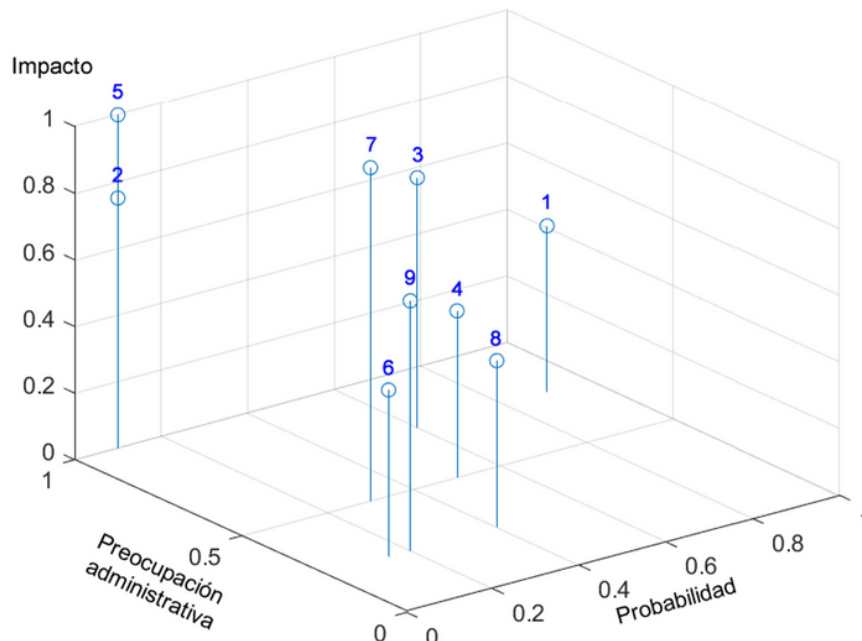


Figura 4. Diagrama de dispersión de datos 3D (MathWorks, 2022).

**Conclusiones**

Realizando la superposición de los datos del análisis de riesgos obtenidos del Cuadro 6 y con ayuda del soporte de un gráfico de superposición de burbujas como se muestra en el Figura 4, podemos observar que los riesgos 1 y 5 se superponen con una denotación de impacto grave casi con áreas circulares equitativas, mientras que existe otra superposición en los riesgos 6, 7 y 9, se puede observar que el riesgo 9 enfatiza una superposición casi absoluta en el riesgo 6 y al ser circundante en la superposición con el riesgo 7

forma un área tripartita que denota cierta atención en el análisis.

De esta forma el acomodo de los resultados nos ofrece los indicadores necesarios para ser considerados en el diseño de software orientado a un módulo mini PFAL, como se puede observar en el análisis de riesgos implementado.

Se puede concluir entonces que los riesgos 1, 5, 6, 7 y 9 son los de mayor prioridad, son aquellos que en el Cuadro 6 se deberán reacomodar como riesgos de alta prioridad para implementar un plan de mitigación, monitoreo y manejo de riesgo (MMMR), por lo tanto, se deberá seccionar en la tabla lo que se conoce como



Figura 5. Superposición de los riesgos latentes en las etapas de desarrollo de software de mini PFAL.

*Línea de Corte* (LC) para superponer el orden de los riesgos, quedando así, de la siguiente forma:

1	5	6	7	9	LC	2	3	4	8
---	---	---	---	---	----	---	---	---	---

Esta propuesta es una metodología validada para el análisis de requerimiento del objeto de estudio del diseño de software de multi arquitectura para un módulo mini PFAL, no obstante (pero no es una propuesta absoluta), sabemos que en la práctica los riesgos son agentes externos que ingresan perturbaciones a los proyectos en cualquier momento, por ello los planes para mitigación de riesgos deben contemplar planes secundarios para tener un éxito práctico.

También evaluar los riesgos entre el impacto y la preocupación administrativa influye mucho en reacomodo y la priorización de los riesgos que se van a atender con antelación. La propuesta de análisis de riesgo cotejada con el análisis de requerimientos del objeto de estudio denota una preocupación en los siguientes temas:

1. Calendarización del proyecto y las tareas que se deben implementar en el cronograma para cumplirlas en tiempo y forma.
2. Actualización interna en temas de hardware y software con el propósito de ofrecer las expectativas de la institución promotora.
3. Reclutamiento de nuevos miembros al equipo de trabajo, y ejecutar programas de capacitación y actividades de actualización (personal de nuevo ingreso y con antigüedad).
4. Crear equipos multidisciplinarios que puedan integrar miembros como diseñadores gráficos para mejorar exponencialmente la presentación de las aplicaciones de software.

### **Agradecimientos**

Los resultados forman parte del programa interno de Proyectos de Transferencia de Tecnológica (DTT), impulsados por la Dirección General de Investigación y Posgrado (DGIP) de la Universidad Autónoma Chapingo (UACH).



**Literatura citada**

- Castellanos, L. T. S., Gómez-Águila, M. V., Paniagua, F. S., Suárez, J. A. C. y Rodríguez, A. M. (2022). Software engineering for a mini-PFAL (plant factory with artificial lighting) with IoT interconnectivity. *Revista ciencias técnicas agropecuarias*, 31(3). <https://rcta.unah.edu.cu/index.php/rcta/article/view/1642>
- Costa, E., Martins, M. B., Vendruscolo, E. P., Silva, A. G. da, Zoz, T., Binotti, F. F. da S., Witt, T. W. y Seron, C. de C. (2020). Greenhouses within the agricultura 4.0 interface. *Ciencia Agronomica*, 51(5). <https://doi.org/10.5935/1806-6690.20200089>
- Goto, E. (2012). Plant production in a closed plant factory with artificial lighting. *Acta Horticulturae*, (956), 37–49. doi:10.17660/actahortic.2012.956.2
- Kern, J. (2021). *Utilizar con éxito los diagramas de causa-efecto: El diagrama de Ishikawa en la teoría y la práctica*. ISBN 979-8505123980.
- Kozai Toyoki, Niu Genhua, Takagaki Michiko (2015). Plant factory: an indoor vertical farming system for efficient farming. *LiderDeProyecto.com / Conceptos Básicos de Riesgos en la Administración de Proyectos*. (s/f). *Liderdeproyecto.com*. Recuperado el 26 de mayo de 2022, de [http://www.liderdeproyecto.com/articulos/conceptos\\_basicos\\_de\\_riesgos](http://www.liderdeproyecto.com/articulos/conceptos_basicos_de_riesgos)
- MatlabWorks. 3-D scatter plot - MATLAB scatter3 - MathWorks España. (s/f). *Mathworks.com*. Recuperado el 2 de julio de 2022, de <https://es.mathworks.com/help/matlab/ref/scatter3.html>
- Microchip (2022). Datasheet ATmega640/V-1280/V-1281/V-2560/V-2561/V. Recuperado el 11 de abril del 2022, de <https://ww1.microchip.com/downloads/aemDocuments/documents/OTH/ProductDocuments/DataSheets/ATmega640-1280-1281-2560-2561-Datasheet-DS40002211A.pdf>
- Mohapatra, P.K.J. (2010). *Software Engineering*. California: New Age International
- Pressman, R. (2010). *Ingeniería del software: un enfoque práctico*. Madrid: McGraw-Hill Interamericana.
- Torres, O. P. (s/f). Gestión de riesgos en proyectos de software. *Piranirisk.com*. Recuperado el 25 de mayo de 2022, de <https://www.piranirisk.com/es/blog/gestion-de-riesgos-proyectos-de-software>
- Zielinski, K. y Szmuc, T. (2005). *Software Engineering: Evolution and Emerging Technologies*. California: IOS Press.