



Solución IoT para robot de exploración ambiental de datos críticos con almacenamiento en blockchain

Autor:

Ing. Gonzalo F. Carreño

Director:

Esp. Ing. Sergio Alberino (UTN.BA)

*Esta planificación fue realizada en el curso de Gestión de proyectos
entre el 1 de julio de 2024 y el 1 de junio de 2025.*

Índice

1. Descripción técnica-conceptual del proyecto a realizar	5
2. Identificación y análisis de los interesados	6
3. Propósito del proyecto	7
4. Alcance del proyecto	7
5. Supuestos del proyecto.	7
6. Requerimientos	8
7. Historias de usuarios (<i>Product backlog</i>).	10
8. Entregables principales del proyecto	11
9. Desglose del trabajo en tareas	11
10. Diagrama de Activity On Node.	12
11. Diagrama de Gantt	12
12. Presupuesto detallado del proyecto	12
13. Gestión de riesgos	12
14. Gestión de la calidad	15
15. Procesos de cierre	15

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento.	1 de julio de 2024

Acta de constitución del proyecto

Buenos Aires, 1 de julio de 2024

Por medio de la presente se acuerda con el Ing. Gonzalo F. Carreño que su Trabajo Final de la Carrera de Maestria en Internet de las Cosas se titulará “Solución IoT para robot de exploración ambiental de datos críticos con almacenamiento en blockchain”, consistirá en la implementación de un sistema embebido de robot de exploracion ambiental conectado a un sistema back-end en la nube publica capaz de persistir los datos en una red blockchain, y tendrá un presupuesto preliminar estimado de \$ 108 dólares estadounidenses y 721 horas de trabajo, con fecha de inicio 1 de julio de 2024 y fecha de presentación pública estimada 20 de octubre de 2025.

Se adjunta a esta acta la planificación inicial.

Dr. Ing. Ariel Lutenberg
Director posgrado FIUBA

Esp. Lic. Mariano Landini
FCE UBA - Cliente

Esp. Ing. Sergio Alberino
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

El presente proyecto es un emprendimiento personal que busca integrar el dispositivo robótico de exploración ambiental controlable a distancia desarrollado en el marco de la carrera de especialización en sistemas embebidos con un backend para el análisis y explotación de datos en la nube pública y una red blockchain para el almacenamiento inmutable de datos críticos.

Estado del arte: Los robots exploradores son los dispositivos robotizados que han sido creados con el fin de reconocer y explorar un lugar o terreno siendo capaces de moverse de forma autónoma o controlados por personas a control remoto. Su objetivo es evitar poner en riesgo la vida de los humanos, ya sea debido a que el lugar es inaccesible o porque se encuentra en una zona contaminada. Algunos de los tipos de robots exploradores más conocidos son los espaciales, de minas, de rescate en catástrofes, de tuberías, acuáticos y/o submarinos, y de suelos.

Estos dispositivos, forman parte de una solución arquitectónica IoT más amplia, cumpliendo el rol de nodos “edge”. Una vez implementados y funcionando envían lecturas del reconocimiento y operaciones realizadas a sistemas de procesamiento y almacenamiento de datos que conforman el backend, generalmente en la nube, desde donde se puede visualizar los resultados obtenidos, sus métricas derivadas, realizar trazabilidad de las operaciones así como almacenar y gobernar los datos generados.

En situaciones en las que el robot es utilizado para explorar y monitorear un área ambientalmente sensible, como una reserva natural o un sitio afectado por un desastre ecológico y su misión es recopilar datos críticos -como niveles de contaminación, temperatura, humedad, y calidad del aire- es de gran importancia almacenar estas mediciones de una forma en la que se pueda asegurar la integridad y transparencia de los datos, como por ejemplo, en una cadena de bloques (blockchain).

Una arquitectura blockchain se basa en el procesamiento y almacenamiento de transacciones agrupadas en bloques encadenados e inmutables de forma distribuida entre los nodos de una red en lo que se conoce como un distributed ledger. De esta manera se puede asegurar integridad de los datos ya que los registros generados no se pueden modificar una vez creados. Además, como la red es accesible entre los actores intervinientes en el caso de uso -pudiendo ser pública o privada y con o sin permisos-, se puede garantizar la transparencia de los datos. La mayoría de las redes blockchain constan de la tecnología para la implementación de código backend ejecutable en la red, que aunque su nombre puede cambiar dependiendo de la red en la cual se implementan, usualmente se los conoce como Smart Contracts. La ejecución de estos componentes es realizada por los nodos de la red en el proceso que se conoce como minería, y como tal es una actividad que requiere el pago de un fee conocido como gas medido en diferentes unidades dependiendo de la red y normalmente pagable desde una cuenta nominada en el token de la red asociada a la aplicación o propietario de los smart contracts. La forma de interactuar con los smart contracts en un caso de uso interactivo desde afuera de la red, se realiza a través de otro componente conocido como dApps (de-centralized applications) que haciendo uso de ciertas librerías de Web3 invocan a estos para almacenar y obtener datos en y desde el ledger.

En el siguiente diagrama se puede apreciar una posible arquitectura del sistema final.

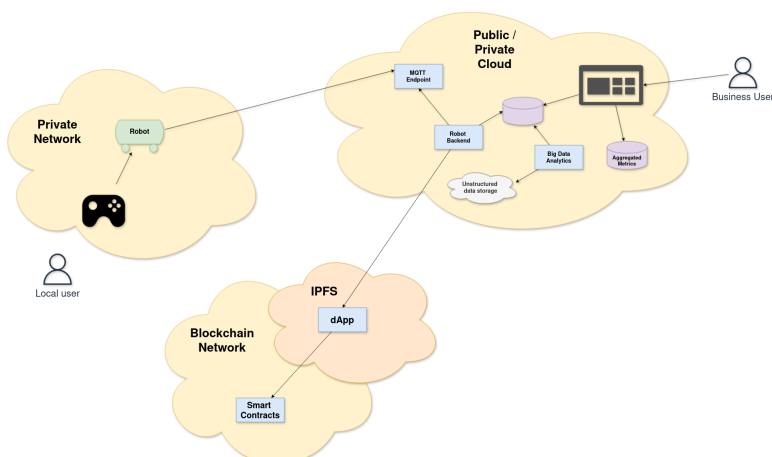


Figura 1. Arquitectura del sistema

El hardware utilizado para la solución de IoT propuesta es un robot de exploración ambiental de control inalámbrico, desarrollado en la Carrera de Especialización en Sistemas Embebidos de la Universidad de Buenos Aires. En la siguiente imagen puede apreciarse una fotografía del mismo.

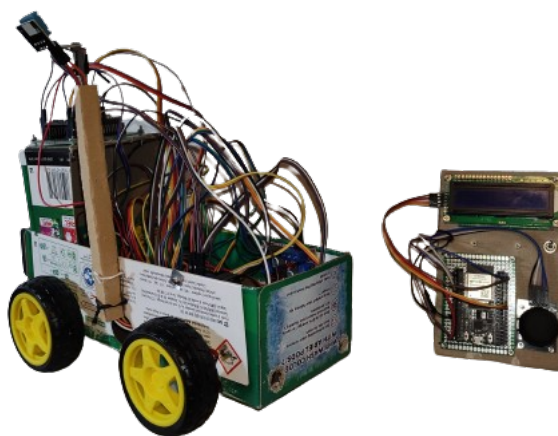


Figura 2. Arquitectura del sistema

2. Identificación y análisis de los interesados

A continuación se enumeran los diferentes roles e individuos que participarán en el proyecto.

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Esp. Lic. Mariano Landini	FCE UBA - Cliente	-
Responsable	Ing. Gonzalo F. Carreño	UTN.BA	Alumno
Orientador	Esp. Ing. Sergio Alberino	UTN.BA	Director Trabajo final

3. Propósito del proyecto

El propósito de este proyecto es desarrollar una solución IoT para un robot de exploración ambiental para casos de uso de datos críticos en los que sea necesario contar con almacenamiento inmutable y transparente para todas las partes en una arquitectura blockchain. Por otra parte, se pretende volcar en un desarrollo concreto y de aplicación industrial los conocimientos adquiridos durante la cursada de la Maestría en Internet de las Cosas.

4. Alcance del proyecto

Contando con el hardware mencionado anteriormente que auspicia de componente edge en una arquitectura IoT, se propone la implementación de la plataforma tecnológica que alcanza:

- La publicación del endpoint MQTT para la recepción de los datos enviados por el robot.
- La adaptación del sistema embebido del robot de exploración ambiental para la conexión segura con el backend vía MQTT.
- La arquitectura e implementación de los sistemas backend y el modelo de datos necesario para el almacenamiento de las mediciones enviadas por el robot.
- La arquitectura, implementación y despliegue de la dApp (de-centralized application) y Smart Contracts necesarios para el almacenamiento de las mediciones en una red Blockchain (a definir).
- La definición de métricas agregadas de valor y posterior arquitectura e implementación de los sistemas analíticos para procesar de forma batch y/o real-time (dependiendo de las métricas a definir) los datos enviados utilizando herramientas de procesamiento paralelo basadas en big data.
- La implementación de la interfaz gráfica para poder visualizar los datos enviados y analíticas calculadas.

5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone que:

- será posible tener acceso a redes de desarrollo y testing de forma gratuita mediante la obtención de Faucets,
- será posible experimentar, desarrollar y hacer testing del backend cloud con el presupuesto estimado,
- arquitectónicamente resultará viable implementar la solución propuesta,
- se dispondrá del conjunto de librerías, drivers y APIs de bajo nivel para el desarrollo de las funcionalidades planteadas en el alcance sin ser necesario el desarrollo de drivers y dichos componentes de bajo nivel. Además, tanto estos componentes de software como los open source de la comunidad de software libre utilizados durante el desarrollo del producto, se encontrará estable para que su integración en el proyecto no resulte en desvíos,

- tanto el prototipado de los componentes de software del sistema embebido como el ensamblado de los componentes de hardware del dispositivo no producirán desvíos considerables en el plan,
- no habrá desvíos no contemplados en el plan que impidan o demoren entregas en el proyecto,
- el comité académico encargado de la corrección tendrá disponibilidad para realizar la evaluación en las fechas planificadas de entrega,
- el director asignado tendrá la disponibilidad de tiempo para darle seguimiento al proyecto,
- el alumno contará con una disponibilidad de entre 3 y 5 horas diarias (incluyendo fines de semana) para el desarrollo del proyecto en el tiempo convenido,
- los materiales, componentes, software de terceros y servicios cloud utilizados funcionaran de forma óptima y de acuerdo a lo esperado,
- el robot desarrollado seguira funcionando de forma estable sin ser necesario su ajuste, reparacion o modificacion a nivel hardware o sistema base (fuera de lo planificado para la integracion con MQTT),
- los recursos no directamente relacionados con el desarrollo del proyecto, pero utilizados durante el mismo, funcionaran adecuadamente y en caso de falta de suministro (por ejemplo el servicio de internet) o averia (por ejemplo en el caso de la computadora utilizada) la resolucion sera expeditiva no suponiendo un desvio en el plan,
- no sucederan nuevos eventos de impacto global (pandemia, guerras, etc) durante el desarrollo del proyecto que impliquen una demora o imposibilidad en la entrega.

6. Requerimientos

A continuación se listan los requerimientos del producto:

1. Requerimientos funcionales

- 1.1. El robot de exploracion ambiental debe poder enviar a la plataforma datos de mediciones de parametros ambientales, incluyendo los datos de fecha, hora, localizacion geografica (puede ser implementada como un mock inicialmente) y la categorizacion si es o no un valor critico.
- 1.2. El robot de exploracion ambiental debe incorporar logica para categorizar los valores medidos de cada parametro ambiental como valores criticos si:
 - 1) representan un maximo o minimo global sensado hasta el momento,
 - 2) representan un maximo o minimo local durante el ultimo dia.
- 1.3. La plataforma debe poder recibir y almacenar las mediciones de parametros ambientales enviadas por el robot.
- 1.4. La plataforma debe poder procesar las mediciones de parametros ambientales enviadas por el robot para generar metricas de valor para el usuario de negocio.
- 1.5. El la plataforma proveer dos front-end con interfaz web:
 - 1) el front-end para el usuario de negocio,

- 2) el front-end para el usuario administrador.
- 1.6. El front-end para el usuario de negocio debe proveer metricas para visualizar:
 - 1) las lecturas historicas almacenadas,
 - 2) agregaciones (maximo, minimo, promedio, etc) de cada parametro ambiental agrupado por frecuencias (ventanas de tiempo) y coordenadas geograficas,
 - 3) las referencias a los datos persistidos en blockchain.
- 1.7. El front-end para el usuario de administracion debe permitir:
 - 1) acceder a los diferentes recursos utilizados por la herramienta (topics MQTT, smart contracts, buckets, etc),
 - 2) resetear valores y estado.
2. Requerimientos no funcionales
 - 2.1. La plataforma debe contar con al menos un backend de procesamiento y acceso a datos operacionales para la logica de negocio.
 - 2.2. La plataforma debe contar con al menos un backend de acceso, procesamiento, almacenamiento de datos analytics para la generacion de metricas.
 - 2.3. El envio de los valores ambientales sensados al backend debe ser mediante MQTT.
 - 2.4. Las lecturas ambientales categorizadas como criticas deben ser almacenadas en blockchain para garantizar fiable, inmutabilidad.
 - 2.5. La gestion de datos almacenados en blockchain debe ser implementada mediante smart contracts desplegados en la red.
 - 2.6. La interaccion con los smart contracts debe realizarse desde una dApp.
 - 2.7. Los sistemas de transferencia y almacenamiento de datos utilizados deben contar con seguridad, permitiendo encriptacion, autenticacion y autorizacion.
3. Requerimientos de documentación
 - 3.1. documentación de arquitectura técnica a alto nivel del diseño del sistema.
 - 3.2. documentación técnica de la implementación del software.
 - 3.3. manual de usuario.
 - 3.4. memoria final.
4. Requerimiento de testing
 - 4.1. se debe incluir tests de unitarios de componentes,
 - 4.2. se debe incluir tests funcionales (smoke test) del producto general.
5. Requerimientos opcionales
 - 5.1. De infraestructura y despliegue:
 - 1) se permite realizar el despliegue de la dApp en un IPFS (preferentemente) o en la nube.
 - 2) se permite agregado hardware al robot para la captura de datos adicionales.
 - 3) se permite agregar automatizacion para la creacion de la infraestructura como codigo.
 - 5.2. De datos:
 - 1) se permite almacenar cualquier otro dato adicional sensado o derivado.
 - 2) se permite agregar cualquier implementacion de gobierno de datos.
 - 3) se permite almacenar cualquier otra metrica o grafico de eplotacion de datos adicional.

7. Historias de usuarios (*Product backlog*)

A continuacion se listan las historias de usuario. La ponderación de *story points* se realiza considerando 1 punto = 1 hora:

1. Metricas agrupadas

- Detalle: Como usuario quiero una metrica que me indique los valores agregados de minimo, maximo y promedio de cada parametro ambiental agrupado por fecha y coordenadas.
- Esfuerzo: 24 puntos
- Criterio de aceptación: funcionalidad verificada (incluyendo prototipado, integración y ensamblado), tests y documentación.

2. Metricas historicas

- Detalle: Como usuario quiero poder ver en series de tiempo los valores medidos para los diferentes parametros ambientales.
- Esfuerzo: 24 puntos
- Criterio de aceptación: funcionalidad verificada (incluyendo prototipado, integración y ensamblado), tests y documentación.

3. Graficos de explotacion

- Detalle: Como usuario quiero contar con metricas graficas (heatmap, pie, etc) para poder visualizar mas facilmente los agregados.
- Esfuerzo: 24 puntos
- Criterio de aceptación: funcionalidad verificada (incluyendo prototipado, integración y ensamblado), tests y documentación.

4. Smart contracts

- Detalle: Como arquitecto quiero que el acceso a los datos en blockchain se realice por medio de la implementacion de smart contracts.
- Esfuerzo: 24 puntos
- Criterio de aceptación: funcionalidad verificada (incluyendo prototipado, integración y ensamblado), tests y documentación.

5. dApp

- Detalle: Como arquitecto quiero que el acceso a los smart contracts se realice desde una dApp utilizando una biblioteca Web3.
- Esfuerzo: 24 puntos
- Criterio de aceptación: funcionalidad verificada (incluyendo prototipado, integración y ensamblado), tests y documentación.

6. Integracion MQTT

- Detalle: Como arquitecto quiero que el envio de los datos desde el robot al backend se realice mediante MQTT.
- Esfuerzo: 24 puntos

- Criterio de aceptación: funcionalidad verificada (incluyendo prototipado, integración y ensamblado), tests y documentación.

7. Arquitectura operacional

- Detalle: Como arquitecto quiero que la arquitectura de tanto los front-ends como el back-end sean web exponiendo APIs via servicios REST. La interfaz grafica debe ser implementada en Angular u otro framework MVC/MVVC de Javascript.
- Esfuerzo: 24 puntos
- Criterio de aceptación: funcionalidad verificada (incluyendo prototipado, integración y ensamblado), tests y documentación.

8. Arquitectura analitica

- Detalle: Como arquitecto quiero que las metricas de valor sean generadas de forma batch mediante un pipeline de datos.
- Esfuerzo: 24 puntos
- Criterio de aceptación: funcionalidad verificada (incluyendo prototipado, integración y ensamblado), tests y documentación.

9. Encriptacion punto a punto

- Detalle: Como ingeniero de seguridad quiero que los datos transmitidos en cada segmento viajen encriptados mediante certificados.
- Esfuerzo: 24 puntos
- Criterio de aceptación: funcionalidad verificada (incluyendo prototipado, integración y ensamblado), tests y documentación.

8. Entregables principales del proyecto

Los entregables del proyecto son:

- Video demostrativo del uso de la plataforma.
- Código fuente del firmware.
- Código fuente del software.
- Documentación:
 1. Manual de usuario,
 2. Memoria final,
 3. Documentación de arquitectura técnica del sistema (hardware y software),

9. Desglose del trabajo en tareas

El conjunto de actividades y tareas que se realizarán durante el proyecto son:

10. Diagrama de Activity On Node

A continuación se detalla la lista de actividades que se realizarán durante el proyecto. Los tiempos están expresados en días, y como se considera una dedicación de 4 horas diarias (incluyendo fines de semana).

Id	Tarea	Duración	Dependencia	Predecesora
1	Gestión del proyecto	-	-	-
2	Investigación y prototipado	-	-	-
3	Set-up integración continua (opcional)	-	-	-
4	Diseño y desarrollo de funcionalidades	-	-	-
5	Testing	-	-	-
6	Documentación	-	-	-

11. Diagrama de Gantt

12. Presupuesto detallado del proyecto

El siguiente cuadro presenta los costos en dólares estadounidenses estimados para el proyecto:

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
Procesamiento cloud	1	\$ 50,00	\$ 50,00
Almacenamiento cloud	1	\$ 20,00	\$ 20,00
Transferencia de datos cloud	1	\$ 50,00	\$ 50,00
Alojamiento IPFS	1	\$ 20,00	\$ 20,00
Varios	1	\$ 20,00	\$ 20,00
TOTAL			\$ 160,00

13. Gestión de riesgos

1. Riesgo de demora

- Severidad (S): 9 - Teniendo en cuenta que solo habrá un recurso (el alumno) asignado al proyecto desarrollando el producto, una demora en cualquier tarea puede implicar demora en la fecha de entrega.
- Ocurrencia (O): 5 - Dado el desafío de innovación planteado por el alcance el la probabilidad de demora es media.

2. Riesgo de no contar con toda la funcionalidad deseada

- Severidad (S): 9 - El no cumplimiento con la funcionalidad deseada pone en riesgo el éxito del proyecto.
- Ocurrencia (O): 5 - Dado el desafío de innovación planteado por el alcance el la probabilidad de no contar con toda la funcionalidad es media.

3. Riesgo de calidad insuficiente

- Severidad (S): 4 - La calidad insuficiente no pone en riesgo el cumplimiento con la funcionalidad pero si compromete la estabilidad y resistencia a fallas del producto, por lo que puede desencadenar en un producto poco o menos confiable.
- Ocurrencia (O): 4 - Se estima que con las técnicas empleadas durante el desarrollo del producto, este riesgo tiene una baja probabilidad de ocurrencia.

4. Riesgo de desvío en costos

- Severidad (S): 5 - La ocurrencia de este riesgo impacta en los costos del proyecto, pero no imposibilita ni demora la entrega del producto.
- Ocurrencia (O): 8 - Teniendo en cuenta que los precios son estimados sin haber cerrado la definición de la arquitectura final y la inflación argentina, es muy probable que exista un desvío en costos.

5. Riesgo de indisponibilidad de recursos

- Severidad (S): 5 - Al momento de realizar el presente plan se identifican ciertos recursos y se asume que será posible disponer de ellos. No obstante, existe el riesgo de que esto no suceda así, y sea más difícil por ejemplo adquirir Faucets para el acceso a redes de desarrollo blockchain; realizar el despliegue de la dApp; acceder a servicios cloud que soporten todas las prestaciones necesarias por la arquitectura; o haya recursos no directamente asociados al proyecto pero cuya ausencia lo afectan, como por ejemplo fallas en el acceso a internet, el mal funcionamiento de la computadora utilizada para su desarrollo, fallas en el hardware del robot que hagan necesario su reparación, etc.
- Ocurrencia (O): 2 - Se espera que la probabilidad de ocurrencia de este riesgo sea realmente baja.

b) Tabla de gestión de riesgos: (El RPN se calcula como $RPN=S \times O$)

Riesgo	S	O	RPN	S*	O*	RPN*
Riesgo de demora en la entrega	9	5	45	9	1	9
Riesgo de no contar con toda la funcionalidad deseada	9	5	45	10	1	10
Riesgo de calidad insuficiente	6	4	24	4	2	8
Riesgo de desvío en costos	5	8	40	5	3	15
Riesgo de indisponibilidad de recursos	5	2	10	-	-	-

Criterio adoptado: Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a 15.

Nota: los valores marcados con (*) en la tabla corresponden luego de haber aplicado la mitigación.

c) Plan de mitigación de los riesgos que originalmente excedían el RPN máximo establecido:

1. Riesgo de demora en la entrega: las posibles causas del evento asociado están vinculadas a situaciones no controlables ni predecibles que impactan de alguna manera en la disponibilidad de tiempo o alguno de los recursos necesarios para la realización del proyecto. Con el fin de cumplir con la entrega de la funcionalidad en la fecha acordada se consideran como posibles acciones de mitigación la eliminación (o no realización) de otras tareas dentro del plan, como por ejemplo, tareas de documentación, de testing, y de ser necesario, de desarrollo de funcionalidad. Se asume que la eliminación de estas tareas ponen en riesgo la calidad del producto y/o contar con toda la funcionalidad esperada.
 - Nueva Severidad (S*): 9 - No cambia.
 - Nueva Ocurrencia (O*): 1.
2. Riesgo de no contar con toda la funcionalidad deseada: las posibles causas del evento asociado están vinculadas a situaciones que impactan de alguna manera en la viabilidad o desarrollo de alguna de las funcionalidades en el tiempo planificado. Por este motivo se considera como herramientas de mitigación: sacrificar algún otro entregable (como por ejemplo la cobertura de testing y/o documentación, lo cual puede implicar sacrificar calidad, mantenibilidad y/o usabilidad respectivamente), o bien redefinir el alcance y funcionalidad en base a lo que es posible desarrollar.
 - Nueva Severidad (S*): 10 - Dado que tras la mitigación se incrementa el impacto por pérdida de calidad.
 - Nueva Ocurrencia (O*): 1 - Se reduce mucho la probabilidad de ocurrencia dado que se agrega tiempo para el desarrollo de funcionalidad eliminando el tiempo empeñado para el desarrollo de tests y/o documentación.
3. Riesgo de calidad insuficiente: las posibles causas del evento asociado están vinculadas a la falta de estabilidad del producto, sea por una arquitectura demasiado compleja o ineficiente, o la presencia de bugs en el desarrollo. Para mitigar este problema se plantea emplear mas tiempo en el desarrollo de pruebas de concepto y definicion de la arquitectura, ademas incrementar las prácticas de testing y CI/CD (como actividad opcional) siempre que esto no dispare el riesgo 1, el cual podría generar una demora una demora en el proyecto.
 - Nueva Severidad (S*): 6 - No cambia.
 - Nueva Ocurrencia (O*): 2 - Se reduce la probabilidad de que esto suceda.
4. Riesgo de desvío en costos: Las posibles causas del evento que dispara este riesgo estan vinculadas a la dificultad de estimar los costos de procesamiento, almacenamiento y transferencia de datos tanto en la nube publica como en la red blockchain a utilizar, debido a que la arquitectura no ha sido definida al momento de la escritura del presente plan y los costos vinculados a su implementacion varian dependiendo del proveedor, red y tecnologias utilizadas. Ademas, indirectamente asociado a esto, es posible olvidar estimar algún componente. Finalmente el impacto de la inflación en Argentina puede ser otro causal del desvio en costos. Para mitigar el primer factor se agrega el item *Varios / Imprevistos* a la tabla de materiales con el fin de proveer holgura en el caso de no contemplar algún componente adicional. Para mitigar el factor inflación se plantean los precios en dólares americanos.
 - Nueva Severidad (S*): 5 - Esto no varía.
 - Nueva Ocurrencia (O*): 3 - Se reduce mucho la probabilidad de ocurrencia dado que la inflación del dólar estadounidense es menor que la del peso argentino.

14. Gestión de la calidad

1. Funcionalidad del front-end negocio: Metricas y graficos de explotacion
 - 1.1. Verificación previo a la entrega: se verificará mediante la ejecución de tests de integración para esta funcionalidad.
 - 1.2. Validación: el cliente validará la funcionalidad en el producto final.
2. Funcionalidad del front-end admin: Control de la plataforma
 - 2.1. Verificación previo a la entrega: se verificará mediante la ejecución de tests de integración para esta funcionalidad.
 - 2.2. Validación: el cliente validará la funcionalidad en el producto final.
3. Funcionalidad de la DApp, almacenamiento en blockchain y ejecucion de smart contracts
 - 3.1. Verificación previo a la entrega: se verificará mediante la ejecución de tests de integración para esta funcionalidad.
 - 3.2. Validación: el cliente validará la funcionalidad en el producto final.
4. Documentación técnica, manual de usuario y memoria final
 - 4.1. Verificación previo a la entrega: se verificará mediante la revisión de los documentos.
 - 4.2. Validación: el cliente validará los documentos.
5. Testing
 - 5.1. Verificación previo a la entrega: Se verificará el cumplimiento con los tests por funcionalidad previo la integración de cada componente en el prototipo final.
 - 5.2. Validación: el cliente validará el reporte de los tests de integración.

15. Procesos de cierre

- Pautas de trabajo que se seguirán para analizar si se respetó el Plan de Proyecto original:
 - Responsable: Ing. Gonzalo F. Carreño:
 - Se evaluarán los requerimientos y los objetivos alcanzados frente a los planteados en el plan.
 - Se pondrá especial interés en verificar si se cumplieron los objetivos de tiempo y funcionalidad propuestos.
- Identificación de las técnicas y procedimientos útiles e inútiles que se emplearon, y los problemas que surgieron y cómo se solucionaron: - Responsable: Ing. Gonzalo F. Carreño:
 - Se evaluará cuál fue la configuración que mejores resultados arrojó para los objetivos planteados en el plan.

- Se identificarán nuevas herramientas o procedimientos, en caso que corresponda.
- Indicar quién organizará el acto de agradecimiento a todos los interesados, y en especial al equipo de trabajo y colaboradores - Responsable: Ing. Gonzalo F. Carreño:
 - Luego de la presentación del proyecto mediante la defensa pública, se procederá a agradecer a todas las personas que participaron del desarrollo del proyecto, al director y a las autoridades de la CEIoT.