

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

Facultad de Ciencias Químicas e Ingeniería

Maestría y Doctorado en Ciencias e Ingeniería



Modelo para la Gestión de Incertidumbres para el Éxito de los Proyectos de Desarrollo de Software

Presenta:

YVES DAYNEL ST FLEUR

Bajo la dirección de:

DR. J. REYES JUÁREZ RAMÍREZ

Co-Directora:

DRA. SAMANTHA PAULINA JIMÉNEZ CALLEROS

TIJUANA, BAJA CALIFORNIA, MÉXICO

NOVIEMBRE 2022

Universidad Autónoma de Baja California
FACULTAD DE CIENCIAS QUÍMICAS E INGENIERÍA

Folio No.332
Tijuana, B.C., a 27 de septiembre del 2022

C. Yves Daynel St. Fleur
Pasante de: Maestría en Ingeniería
Presente

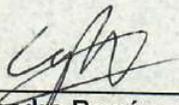
El tema de trabajo y/o tesis para su examen profesional, en la
Opción TESIS

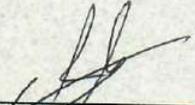
Es propuesto, por los C. Dr. J. Reyes Juárez Ramírez y
Dra. Samantha Paulina Jiménez Calleros

Quienes serán las responsables de la calidad del trabajo que usted presente,
referido al tema "Modelo Para La Gestión De Incertidumbres Para El Éxito De
Los Proyectos De Desarrollo De Software".

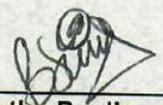
El cual deberá usted desarrollar, de acuerdo con el siguiente orden:

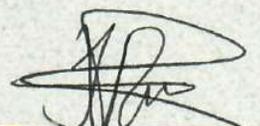
- I. INTRODUCCIÓN
- II. MARCO TEÓRICO
- III. MODELO PROPUESTO PARA EL MANEJO DE INCERTIDUMBRE
- IV. EXPERIMENTACIÓN
- V. RESULTADOS
- VI. CONCLUSIONES, APORTACIONES Y TRABAJO FUTURO


Dra. Ana Alejandra Ramirez Rodriguez
Sub-Directora


M.C. Roberto Alejandro Reyes Martinez
Director




Dra. Samantha Paulina Jimenez Calleros
Co - Directora de Tesis


Dr. J. Reyes Juárez Ramirez
Director de Tesis

Dedicado a mis seres queridos.

Agradecimientos

Primero, me gustaría agradecer a todas las personas que me dieron ánimos a la realización de este proyecto y que día con día han estado motivándome a seguir adelante y concluirlo, sin ustedes esto no sería posible.

En especial quiero agradecer al Doctor J. Reyes Juarez Ramirez y a la Maestra Martha Alicia Rocha Sánchez quienes siempre me han apoyado incondicionalmente. Doctor Reyes, muchas gracias por todas sus enseñanzas. Las conversaciones que hemos tenido me fueron muy útiles, no solo para este trabajo, sino también para la vida. He sido testigo de su gran sentido de responsabilidad y su acompañamiento indefectible en el progreso de sus alumnos. Doctor Reyes, usted es un gran educador. Maestra Marta, nada de lo que pueda escribir es suficiente y quiero agradecerle desde el fondo de mi corazón, nada de esto hubiera sido posible sin su apoyo.

A la Doctora Samantha Paulina Jiménez Calleros muchas gracias por su colaboración, usted es una fuente de inspiración para seguir adelante. A los miembros del Comité Dr. Guillermo Licea Sandoval y Dr. Luis Guillermo Martínez Méndez, muchas gracias por sus valiosas observaciones y comentarios. A mis compañeros y a los grupos de desarrollo de software muchas gracias por el apoyo sobre todo en la fase de experimentación.

A la Universidad Autónoma de Baja California muchas gracias por abrirme sus puertas, al Consejo Nacional de Ciencia y Tecnología (CONACYT), por el financiamiento brindado durante la realización de esta maestría.

¡A TODOS MUCHAS GRACIAS!

Resumen

La incertidumbre es la falta de seguridad, de confianza o de certeza sobre algo, especialmente cuando crea inquietud. En la Ingeniería de Software la incertidumbre es simplemente una expresión de ambigüedad e indeterminación en el proyecto. En la gestión de proyectos de desarrollo de software, el cono de incertidumbre describe la evolución de la presencia de la incertidumbre durante la realización de un proyecto. Investigaciones en la industria del software sobre el cono de incertidumbre han demostrado que, al principio del ciclo de vida del proyecto, antes de la recolección de los requisitos de software, se estima generalmente una incertidumbre de factor 4 tanto en la parte superior como en la inferior del cono. Esto significa que el esfuerzo o el alcance real puede ser hasta 4 veces o $1/4$ de la primera estimación. Las fuentes de la incertidumbre son diversas, entre ellas se encuentran las siguientes: La ambigüedad en los requerimientos, los cambios, las informaciones incompletas, el entorno, la inexperiencia. El nivel de incertidumbre se incrementa cuando los miembros del equipo de desarrollo son novatos. La incertidumbre tiene asociados riesgos, los cuales pueden ser perjudiciales para el éxito del proyecto.

Teniendo en cuenta esta situación, este trabajo de tesis propone estudiar los efectos de la incertidumbre en la estimación de la complejidad de las historias de usuario en las metodologías ágiles para el desarrollo de software, específicamente en Scrum, implementando un modelo de identificación de riesgos para medir el impacto de la incertidumbre en el proyecto de software. Se tiene como hipótesis la siguiente: El nivel de riesgo impacta la estimación de

la complejidad de las historias de usuario.

Se propone un modelo para identificar las diferentes fuentes de incertidumbres, disponer de la capacidad para determinar los posibles riesgos provocados por dichas incertidumbres y sobre todo desarrollar la competencia de evaluación de esos riesgos.

Durante la investigación se realizaron experimentos con equipos de desarrolladores novatos para poder determinar la relación que existe entre las estimaciones de las historias de usuarios y el nivel de incertidumbre en los proyectos de desarrollo de software. Dado que, en promedio, los resultados muestran una fuerte correlación entre estas dos variables, podemos concluir que el nivel de riesgo impacta en la estimación de las historias de usuarios de los proyectos de software, validando la hipótesis de este trabajo.

En términos prácticos, el modelo desarrollado ayuda a los miembros de los equipos de desarrollo de software en la identificación de incertidumbre en los proyectos, les ayuda en la comprensión de la incertidumbre clarificando el problema que representa dicha incertidumbre, permitiendo la evaluación de las probabilidades de ocurrencia de los riesgos y sus impactos, logrando un análisis cuantitativo de los efectos perjudiciales para el proyecto.

Abstract

Uncertainty is the lack of security, confidence, or certainty about something, especially when it creates concern. In Software Engineering, uncertainty is simply an expression of ambiguity and indeterminacy in the project. In the management of software development projects, the uncertainty cone describes the evolution of the presence of uncertainty during the execution of a project. Research in the software industry on the cone of uncertainty has shown that, early in the project life cycle, before the collection of software requirements, an uncertainty of factor 4 is generally estimated at both the top and bottom of the cone. This means that the actual effort or scope can be up to 4 times or $1/4$ of the first estimate. The sources of uncertainty are diverse, among them are the following: Ambiguity in the requirements, changes, incomplete information, the environment, inexperience. The level of uncertainty increases when the members of the development team are newbies. Uncertainty has associated risks, which can be detrimental to the success of the project.

Taking this situation into account, this thesis work proposes to study the effects of uncertainty in estimating the complexity of user stories in agile methodologies for software development, specifically in Scrum, implementing a risk identification model for measure the impact of uncertainty on the software project. The following hypothesis is addressed: The level of risk impacts the estimation of the complexity of user stories.

A model is proposed to identify the different sources of uncertainty, to have the capacity to determine the possible risks caused by said uncertainties and, above all, to develop the

competence to evaluate these risks.

During the investigation, experiments were carried out with teams of novice developers to determine the relationship between the estimates of user stories and the level of uncertainty in software development projects. Given that, on average, the results show a strong correlation between these two variables, we can conclude that the level of risk has an impact on the estimation of user stories of software projects, validating the hypothesis of this work.

In practical terms, the developed model helps members of software development teams to identify uncertainty in projects, helps them to understand uncertainty by clarifying the problem that said uncertainty represents, allowing the evaluation of the probabilities of occurrence of risks and their impacts, achieving a quantitative analysis of the detrimental effects for the project.

Índice general

Resumen	V
Abstract	VII
1. Introducción	1
1.1. Planteamiento del problema	1
1.2. Objetivos	2
1.2.1. General	2
1.2.2. Específicos	2
1.3. Hipótesis	2
1.4. Preguntas de investigación	3
1.5. Metodología	3
2. Marco Teórico	5
2.1. Incertidumbre	5
2.1.1. El cono de la incertidumbre	6

2.2. Riesgos	8
2.2.1. Las fuentes de incertidumbre en los Proyectos de software	8
2.3. Metodologías ágiles	11
2.3.1. Fases de la metodología Scrum	12
2.3.2. Roles en Scrum	14
2.3.3. La estimación de la complejidad en Scrum	15
2.4. Metamodelo de incertidumbre en el desarrollo de software	16
2.4.1. Lenguaje unificado de modelado (UML)	16
2.4.2. Metaclase uncertainty_of	17
2.4.3. Metaclase uncertainty_of_requirements	19
2.4.4. Metaclase uncertainty_of_resources	20
2.4.5. Metaclase uncertainty_of_process	20
3. Modelo propuesto para el manejo de incertidumbre	23
3.1. La identificación de las incertidumbres	24
3.1.1. Guía de identificación de incertidumbre (GII)	26
3.1.2. Identificación de la fuente de incertidumbre	27
3.2. Transformación de incertidumbres en riesgos.	27
3.3. Evaluación de riesgos	29
3.3.1. Estimación de la probabilidad de ocurrencia	29
3.3.2. Estimación del impacto	30
3.4. Herramienta de gestión de incertidumbres (HGI)	31

4. Experimentación **39**

 4.1. Muestra 39

 4.2. Proceso de experimentación 42

5. Resultados **44**

6. Conclusiones, aportaciones y trabajo futuro **51**

 6.1. Conclusiones 51

 6.2. Aportaciones 52

 6.3. Trabajo Futuro 53

Índice de figuras

2.1. Cono de la incertidumbre	7
2.2. Fuentes de incertidumbre en los proyectos	9
2.3. Evaluación del uso de métodos ágiles(23 June 2018, by Chris Kulbacki).	12
2.4. El marco de trabajo Scrum	13
2.5. Roles en Scrum	14
2.6. Metamodelo de incertidumbre en el desarrollo de software	18
3.1. Etapa de identificación de incertidumbres	25
3.2. Etapa de transformación de incertidumbres en riesgos	28
3.3. Etapa de evaluación de riesgos	30
3.4. Matriz de riesgos	32
3.5. Tecnología cliente-servidor	33
3.6. Diagrama de los módulos de la herramienta de gestión de incertidumbre.	34
3.7. Módulo de login de usuarios	35
3.8. Módulo de historias de usuario y de estimaciones	36

3.9. Módulo de registro de riesgo 37

3.10. Módulo de evaluación de riesgos 38

5.1. Las fuentes de incertidumbre de los riesgos identificados 45

5.2. Impacto del nivel de incertidumbre sobre las estimaciones 47

5.3. Impacto del nivel de incertidumbre sobre las estimaciones 48

5.4. Impacto del nivel de incertidumbre sobre las estimaciones 49

5.5. Impacto del nivel de incertidumbre sobre las estimaciones 50

Índice de cuadros

2.1. Comparación entre riesgos e incertidumbres en diferentes áreas de conocimiento	6
3.1. Preguntas de la guía para la identificación de incertidumbres	26
3.2. Formato de declaración de riesgos	29
4.1. Muestra de historias de usuario del proyecto sevilla.	40
4.2. Muestra de historias de usuario del sistema de secas.	41
4.3. Muestra de historias de usuarios del Software Médico	41
4.4. Muestra de historias de usuarios del proyecto BarrioEats	42
5.1. Porcentajes de riesgos identificados por equipos	44
5.2. Interpretación del coeficiente de correlación	46
5.3. Resultados proyecto de Software Médico	46
5.4. Resultados proyecto Sevilla	47
5.5. Resultado proyecto sistema de becas	48
5.6. Resultado proyecto BarrioEats	49

Siglas

CI Cono de la Incertidumbre. 6, 7

ERS Especificación de Requisitos de Software. 1

GII Guía de identificación de incertidumbre. X, 26

HGI Herramienta de gestión de incertidumbres. X, 31–34, 37

HU Historias de Usuario. 2, 15

IG Ingeniería de Software. 5, 8

IRA Impacto del Riesgo sobre el Alcance. 30

IRC Impacto del Riesgo sobre la Calidad. 31

IRM impacto del riesgo sobre el Costo. 31

IRT Impacto del Riesgo sobre el Tiempo. 31

MMI Modelo para el Manejo de Incertidumbre. 23, 24, 35

NIR Nivel de Riesgos. 31

OMG Object Management Group. 16

PDS Proyectos de Desarrollo de Software. 1–3

POR Probabilidad de Ocurrencia del Riesgo. 29

PP Planning Poker. 15

UML Lenguaje unificado de modelado. X, 16

Capítulo 1

Introducción

1.1. Planteamiento del problema

La incertidumbre es la falta de seguridad, de confianza o de certeza sobre algo, especialmente cuando crea inquietud. En la ingeniería de software la incertidumbre es simplemente una expresión de ambigüedad e indeterminación en el proyecto [1]. En la gestión de Proyectos de Desarrollo de Software (PDS), el cono de incertidumbre describe la evolución de la presencia de la incertidumbre durante la realización de un proyecto [2]. Investigaciones en la industria del software sobre el cono de incertidumbre demostraron que al principio del ciclo de vida del proyecto, antes de la recolección de la Especificación de Requisitos de Software (ERS) se estima, generalmente, una incertidumbre de factor 4 tanto en la parte superior como en la inferior del cono [3]. Esto significa que el esfuerzo actual o el alcance puede ser hasta 4 veces o 1/4 de la primera estimación. La tendencia de incertidumbre disminuye a través del curso y desarrollo del proyecto, eso depende del comportamiento de las fuentes de incertidumbre durante el proceso, aunque esta reducción no es siempre garantizada [4]. Eso afectará directamente al ciclo de productividad debido a un alto nivel de incertidumbre. Esto

se convierte en una fuente que puede causar la falla del proyecto. Hacer una estimación con una tasa de incertidumbre elevada tiene como resultado excesos de presupuesto, retraso de las fechas de entrega o falta de tiempo, esto a menudo resulta en software de baja calidad.

Teniendo en cuenta esta situación, este trabajo propone estudiar los efectos de la incertidumbre en la estimación de la complejidad de las Historias de Usuario (HU) en las metodologías ágiles, implementando un modelo de identificación de riesgos.

1.2. Objetivos

1.2.1. General

El objetivo principal de este trabajo es el siguiente: Evaluar el impacto de la incertidumbre en los PDS en los equipos novatos.

1.2.2. Específicos

Los objetivos específicos de este trabajo son los siguientes:

- Identificar cómo se manifiesta la incertidumbre en los PDS en los equipos novatos.
- Determinar cómo los miembros de los equipos perciben la incertidumbre en los PDS.
- Determinar cómo la incertidumbre impacta en las estimaciones de HU en los PDS en los equipos novatos.

1.3. Hipótesis

El nivel de riesgo impacta la estimación de la complejidad de las HU.

1.4. Preguntas de investigación

- ¿Cómo se manifiesta la incertidumbre en los PDS de los equipos novatos?
- ¿Cómo los miembros de los equipos novatos de PDS perciben la incertidumbre en los proyectos?
- ¿Cómo la incertidumbre impacta en las estimaciones de los PDS de los equipos novatos?

1.5. Metodología

▪ Etapa 1 Revisión de literatura

- Paso 1: Como primer paso se realizó una revisión de la literatura.

▪ Etapa 2 Planeación

- Paso 1: Se identificó y analizó cuáles son las variables más importantes que impactan en la estimación de los proyectos.
- Paso 2: Se identificó y analizó cuáles son las variables que causan mayor incertidumbre en la estimación.
- Paso 3: Se determinó la responsabilidad de cada uno los roles y de los requisitos de software en la generación de incertidumbre dentro de la metodología SCRUM.
- Paso 5: Se Implementó el modelo U-SHOT propuesto.
- Paso 7: Se Validó el modelo U-SHOT propuesto.

▪ Etapa 3 Ejecución

- Paso 2: Para validar el modelo, se utilizó un ambiente controlado con desarrolladores novatos, estudiantes de las carreras de ingeniería en computación.

- **Etapa 4: Análisis de resultados**

- Los datos recolectados se almacenaron en una hoja de cálculo, separados por proyectos. Para validar el modelo propuesto se utilizó pruebas de correlación de Spearman. Se utilizó Microsoft Excel. Para normalizar los datos se usó pruebas de normalidad utilizando Microsoft Excel.

Capítulo 2

Marco Teórico

2.1. Incertidumbre

Históricamente el concepto de incertidumbre ha sido ligado a los conceptos de matemáticas, más precisamente a los conceptos de probabilidades. La incertidumbre es vista como un fenómeno en el cual los resultados y sus las probabilidades de variación son impredecibles [5]. Sin embargo, la incertidumbre es mucho más amplio sobre todo en el ámbito de desarrollo de software; una definición tan estrecha nos puede proporcionar una comprensión muy superficial del manejo de la incertidumbre en acciones organizacionales. [5]

La incertidumbre es la falta de seguridad, de confianza o de certeza sobre algo, especialmente cuando crea inquietud. En la Ingeniería de Software (IG) la incertidumbre es simplemente un expresión de ambigüedad e indeterminación en el proyecto [6].

Es importante tener en cuenta que aunque el riesgo y la incertidumbre están relacionadas, no son lo mismo. La incertidumbre es lo desconocido, mientras que el riesgo es lo que puede salir mal. Claramente, gran parte del riesgo del proyecto depende de la incertidumbre, pero hay otros factores que varían al riesgo del proyecto, incluidos los plazos, la falta de recursos

y habilidades inadecuadas [6]. Se maneja incertidumbre y riesgos en diferentes áreas, en el cuadro 2.1 se presenta una comparación de esos términos en diferentes áreas.

Cuadro 2.1: Comparación entre riesgos e incertidumbres en diferentes áreas de conocimiento

Fuente	Riesgos	Incertidumbre
Diccionario	Peligro, probabilidad de falla [6]	Inseguro, dudas, indecisión. [6]
Economía	Eventos sujetos a probabilidades conocidas. [6]	Eventos para los cuales es imposible especificar probabilidades numéricas. [6]
Psicología	La decisión se toma bajo condiciones de probabilidades conocidas. [6]	Un estado mental caracterizado por una falta consciente de un evento. [6]
Gestión de proyectos	El riesgo es lo que puede salir mal en el proyecto. [6]	La incertidumbre es un evento o una situación que no se esperaba que ocurriera, independientemente de si hubiera sido posible considerarlo por adelantado. [6]

2.1.1. El cono de la incertidumbre

En la gestión de proyectos, el cono de incertidumbre describe la evolución de la presencia de la incertidumbre durante la realización de un proyecto. La incertidumbre no solo se reduce conforme pasa el tiempo, sino que también disminuye su impacto, especialmente en la toma de decisiones [2]. El Cono de la Incertidumbre (CI) es limitado y reducido a través de la investigación y la toma de decisiones, lo cual remueve las fuentes de variabilidad del proyecto. Estas decisiones tratan sobre el alcance y lo que está o no incluido en el proyecto. Si éstas cambian durante el desarrollo de este, el CI se hará más extenso. Para el caso de proyectos de software, al inicio de un proyecto no están claros los detalles específicos de la naturaleza del software que se construirá, los detalles de los requerimientos específicos, los detalles de

la solución, el plan del proyecto, la dotación de personal y otras variables del proyecto. La variabilidad en estos factores contribuye a la variabilidad de las estimaciones del proyecto: una estimación precisa de un fenómeno variable debe incluir la variabilidad del fenómeno en sí. A medida que estas fuentes de variabilidad se investigan más a fondo y se determinan, la variabilidad en el proyecto disminuye y, por lo tanto, la variabilidad en las estimaciones del proyecto también puede disminuir. Este fenómeno se conoce como el CI que se ilustra en la Figura 2.1.

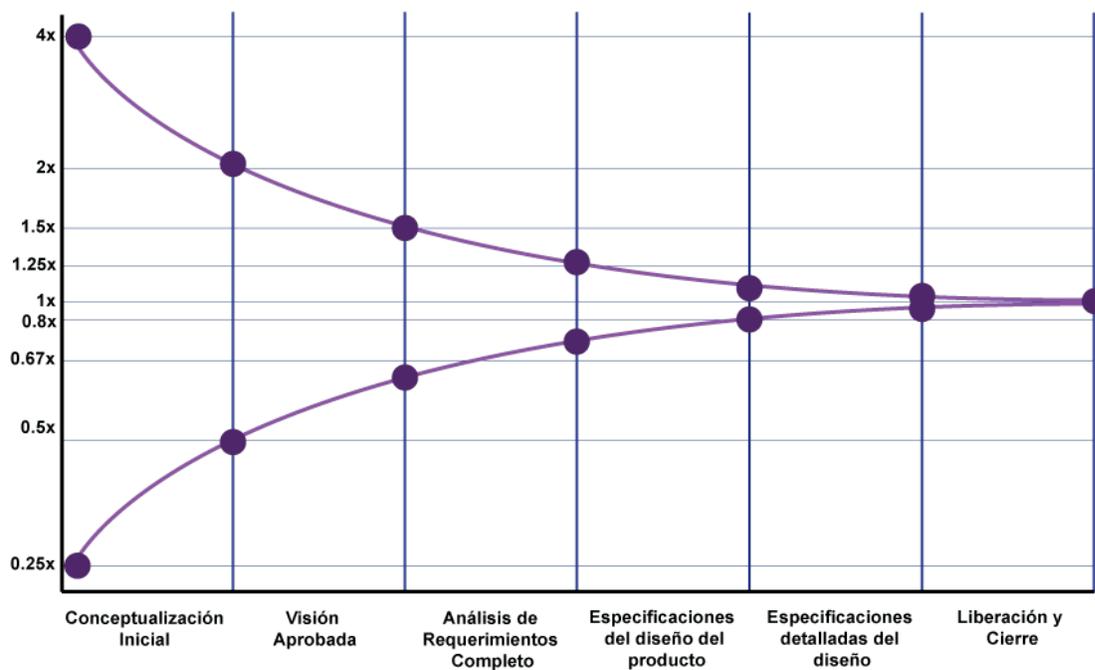


Figura 2.1: Cono de la incertidumbre

En la Figura 2.1 el eje vertical contiene el grado de error que se ha encontrado en las estimaciones creadas por estimadores calificados en varios puntos del proyecto. Las estimaciones podrían ser sobre cuánto costará un conjunto particular de características (funcionalidades del sistema) y cuánto esfuerzo se requerirá para entregar ese conjunto de características, o podría ser sobre cuántas características se pueden entregar para una cantidad particular de

esfuerzo o programa de tiempo. El cono muestra que en un proyecto secuencial o en cascada que está en etapa de factibilidad normalmente daremos una estimación que está lejos de la realidad, en un rango de entre un 25 % y un 400 %. Esto es, por ejemplo, que un proyecto de 40 semanas tomará entre 10 y 160 semanas. Al cono de la incertidumbre también se le conoce como “cono de la muerte”.

2.2. Riesgos

En el mundo de la IG el riesgo es todo lo que puede salir mal en los proyectos. Un riesgo individual es una amenaza específica que se percibe como un obstáculo para el logro de uno o varios requerimientos en un proyecto . En conjunto, los riesgos individuales representan el conjunto identificado de escenarios indeseables que ponen en riesgo el cumplimiento de los requisitos del proyecto.

2.2.1. Las fuentes de incertidumbre en los Proyectos de software

En los proyectos de software la incertidumbre es casi omnipresente, eso es explica ya que sus fuentes son diversas, varios autores hablan de esas fuentes, tratando de agrupar las incertidumbres y crear una etiqueta única para cada una. Según Marinho como se muestra en la Figura 2.2 en los proyectos de software las fuentes de incertidumbre son las siguientes categorías: [6]:

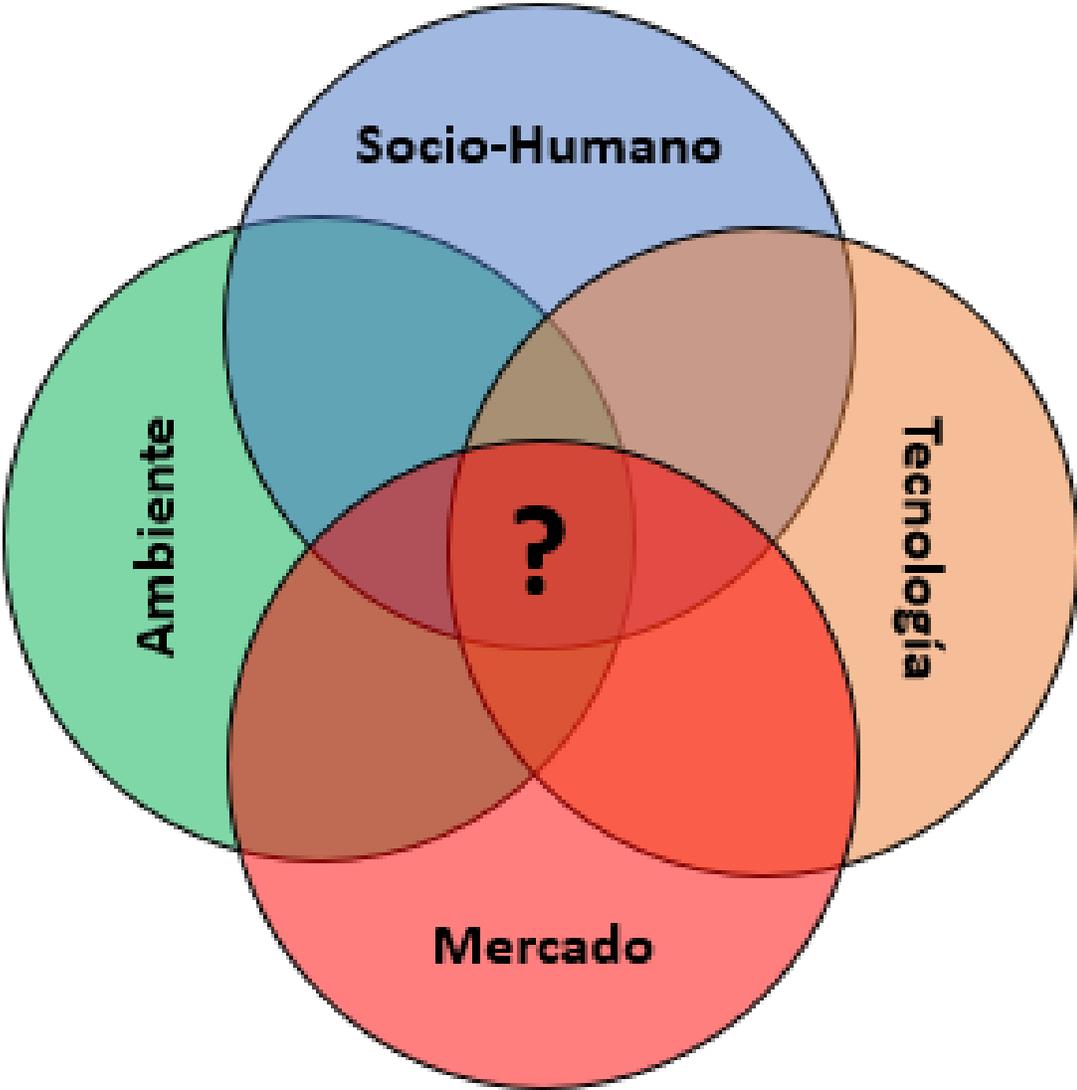


Figura 2.2: Fuentes de incertidumbre en los proyectos

- **TECNOLOGÍA:** La fuente de incertidumbre más citada en los artículos es la incertidumbre tecnológica depende de que si el proyecto usa tecnología nueva o tecnología madura. La incertidumbre del proyecto no es universal, sino subjetiva, porque depende del saber hacer tecnológico que existe o es accesible a la empresa. La incertidumbre tecnológica causa, entre otras cosas, un impacto en el proyecto, en la comunicación, en el tiempo de congelación del plan y el número de ciclos de planificación. También puede afectar la experiencia técnica del gerente de proyecto y los miembros de su equipo.
- **MERCADO:** El mercado de consumo y los diferentes mercados se comportan y piensan de manera diferente. Por lo tanto, los equipos de proyecto deben saber cómo piensan sus clientes, si tienen claro cuáles son sus principales problemas. Los proyectos en varios niveles tienen sus propios elementos. Si las necesidades del mercado ya son bien conocidas, el proyecto probablemente tendrá poca incertidumbre. Por otro lado, si no son bien conocidas el proyecto probablemente tendrá mucha incertidumbre. La incertidumbre del mercado comprende al cliente, proveedores, socios y situación actual del mercado.
- **AMBIENTE:** La incertidumbre ambiental puede involucrar acciones relevantes para las organizaciones o grupos dentro de la organización (proveedores, competidores, consumidores, gobierno, accionistas, capacidad del equipo, tamaño del ciclo de vida del proyecto, los recursos, etc.) pueden afectar el producto, así como las dudas sobre la probabilidad o naturaleza de los cambios en el estado general del medio ambiente.
- **SOCIO-HUMANO:** Las organizaciones cuentan con herramientas tecnológicas modernas que satisfacen las necesidades y las deficiencias estructurales, pero no es suficiente para requerir la adquisición de los conocimientos individual y grupal debido a factores cognitivos intrínsecamente relacionado con la forma en que las personas perciben, aprenden, recuerdan y piensan acerca de la información, eso causa mucha incertidumbre.

2.3. Metodologías ágiles

Las metodologías ágiles son una opción para el desarrollo de proyectos que necesitan rapidez y flexibilidad para adaptarse a las necesidades del cliente siempre está enfocada a los resultados [1]. El manifiesto ágil pretende privilegiar a los individuos y su interacción sobre los procesos, la colaboración con el cliente y la respuesta ante el cambio sobre el seguimiento de un plan. La agilidad nos permite rectificar y cambiar las prioridades y requerimientos según los resultados obtenidos durante el desarrollo del proyecto, al mismo tiempo mejora la experiencia con el cliente al estar en constante contacto con el proyecto [7]. Las principales ventajas de la metodología ágil son [8]:

- **MEJORA LA CALIDAD:** Minimiza los errores en los integrables, mejora la funcionalidad y la experiencia de usuario.
- **MAYOR COMPROMISO:** Mejora la satisfacción del empleado y genera conciencia de equipo.
- **RAPIDEZ:** Acorta los ciclos de producción y minimiza los tiempos de reacción y toma de decisiones.
- **AUMENTO DE LA PRODUCTIVIDAD:** Al asignar mejor los recursos, y de forma dinámica, mejora la producción según las prioridades que tenga la empresa.

En la actualidad Scrum es la metodología ágil más popular como se muestra en la Figura 2.3 y sigue ganando importancia [9]. Scrum es un proceso diseñado para añadir energía, enfoque, claridad y transparencia en las actividades y productos a los miembros del equipo de desarrollo de software. Esta metodología se basa en las teorías actuales de control de proceso y específicamente tiene como objetivo producir el mejor resultado final, con los recursos actuales y el tiempo disponible [10].

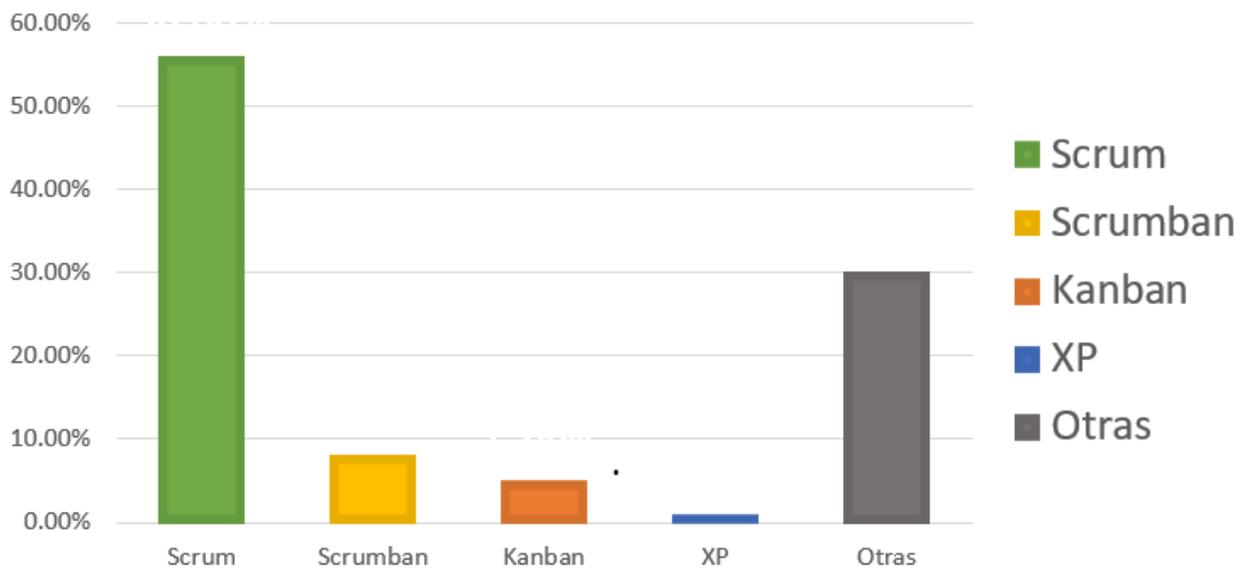


Figura 2.3: Evaluación del uso de métodos ágiles(23 June 2018, by Chris Kulbacki).

2.3.1. Fases de la metodología Scrum

El desarrollo de producto tiene un ciclo de vida en la metodología Scrum. Estas son fases en las que se divide un proceso Scrum [11] como se muestra en la Figura 2.4:

- ¿Qué y quién? El producto que queremos conseguir una vez terminemos el *Sprint*, y los roles de equipo con sus tareas asignadas.
- ¿Dónde y cuándo? El plazo y el contenido del *Sprint*.
- ¿Por qué y cómo? Las distintas herramientas para aplicar esta metodología ágil.

Cada *Sprint* puede tener una serie de eventos o etapas. Los más comunes son:

- Reunión para la planificación del *Sprint*. En ella, se divide el tiempo de duración del *Sprint*, así como el objetivo y entregable del mismo. Además, el equipo de desarrollo deberá saber cómo realizarlo.



Figura 2.4: El marco de trabajo Scrum

- Scrum diario. Se basa en poner en común y sincronizar actividades para elaborar el plan del día.
- Trabajo de desarrollo durante el *Sprint*. Nos aseguramos que los objetivos se están cumpliendo, que no se producen cambios que alteran el objetivo del *Sprint* y se mantiene un *feedback* constante con el cliente o dueño del proyecto.
- Revisión del *Sprint*. Reunión con el cliente o dueño del proyecto, en la que se estudia y revisa el *Product Backlog* del *Sprint*. Se definen los aspectos a cambiar, en caso necesario, de mayor valor o probables para planificarlo en el siguiente *Sprint*.
- Retrospectiva del proyecto. Oportunidad del equipo de desarrollo para mejorar su proceso de trabajo y aplicar los cambios en los siguientes *Sprints*.

2.3.2. Roles en Scrum

La metodología Scrum tiene unos roles y responsabilidades principales como se muestra en la Figura 2.5, asignados a sus procesos de desarrollo. Estos son [12]:

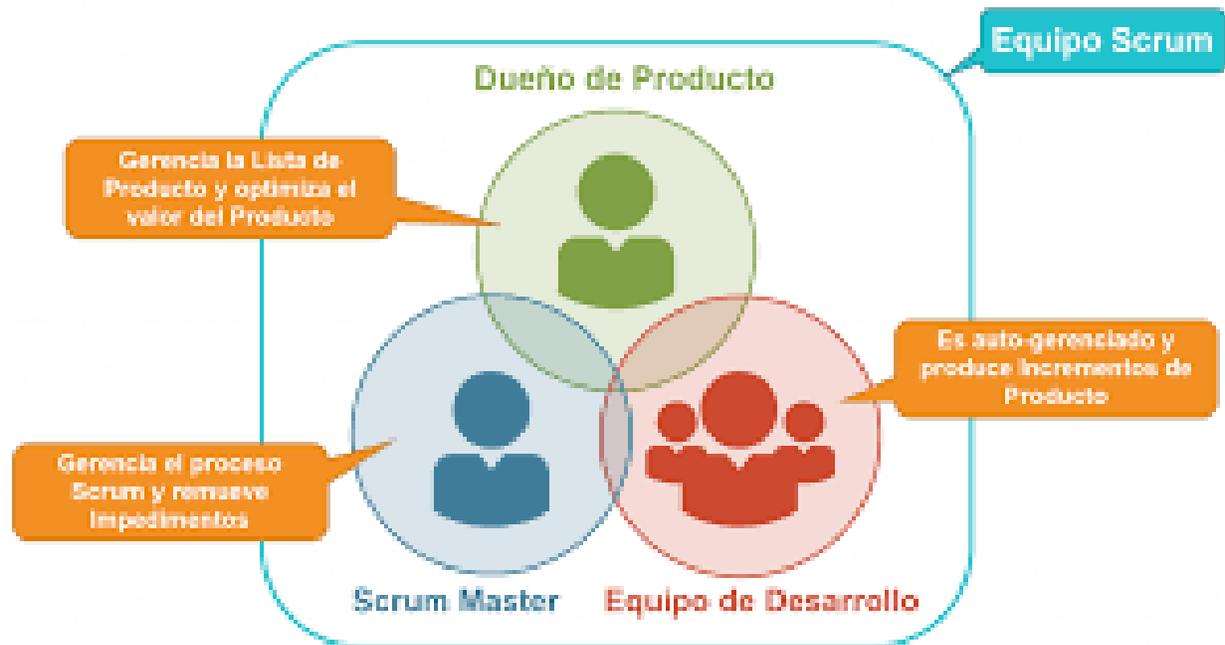


Figura 2.5: Roles en Scrum

- *Project Owner* o responsable del proyecto. Se asegura de que el proyecto se esté desarrollando acorde con la estrategia del negocio. Escribe historias de usuario, las prioriza, y las coloca en el *Product Backlog*.
- *Scrum Master* o facilitador. Elimina los obstáculos que impiden que el equipo cumpla con su objetivo.
- *Development team Member* o el equipo de desarrollo. Los encargados de crear el producto para que pueda estar listo con los requerimientos necesarios. Se recomienda que sea un equipo multidisciplinar, de no más de 10 personas.

2.3.3. La estimación de la complejidad en Scrum

Las estimaciones de esfuerzo o complejidad en condiciones de incertidumbre es un factor esencial en la planificación de proyectos de software para evitar excesos de presupuesto, retraso de las fechas de entrega o falta de tiempo o software de baja calidad [13].

Scrum no establece una técnica específica para realizar la estimación, sin embargo, la técnica más empleada es Planning Poker (PP) [14]. PP es una técnica de estimación basada en el consenso para estimar el tamaño de historias de usuario. Una HU se puede definir como una pieza corta de la funcionalidad que proporciona un cliente o un usuario de un sistema con un valor, representa las necesidades de los usuarios. Para medir estas historias de usuario se utilizan puntos de historia, los cuales representan un valor relativo el cual es usado para comparar el esfuerzo para implementar los requerimientos [15].

PP tiene muchos beneficios, sin embargo, este método no siempre es eficiente debido a que el resultado es siempre basado en la observación de expertos y su experiencia. El valor asignado a las historias de usuario es un valor que no puede ser convertido fácilmente en tiempo [16] además la decisión de los miembros del equipo es poco clara, debido a que toman en cuenta la complejidad y la importancia de la tarea en forma general, sin desglosar sus factores. Por tal razón, es necesario desglosar estas variables en sus elementos para definir claramente cómo un miembro del equipo llega a su decisión de seleccionar el valor de estimación.

Scrum tiene también debilidad, por ejemplo, se maneja mucha incertidumbre, es muy difícil estimar cuánto tiempo llevará un proyecto o cuánto costará [17].

La complejidad, incertidumbre hacen muy difícil desarrollar productos y servicios de forma planificada y secuencial, al tiempo que generan tensiones y frustraciones en los equipos de trabajo.

2.4. Metamodelo de incertidumbre en el desarrollo de software

Vamos a establecer un metamodelo de la incertidumbre en el proceso de desarrollo de software. Para la representación escogemos el UML, para poder describir, integrar la representación de la incertidumbre en el proceso. Es modelo está basado en el modelo propuesto por Xiao en su tesis doctoral [18].

2.4.1. Lenguaje unificado de modelado (UML)

UML, por sus siglas en inglés, (Unified Modeling Language) es un lenguaje de modelación muy utilizado el mundo de desarrollo de software. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados. Está respaldado por el Object Management Group (OMG). Vamos a utilizar este lenguaje por esos razón principalmente. UML permite la modelización de la incertidumbre gracias a sus mecanismo de extensión. la utilización de concepto como:

- Estereotipos
- Valor marcados
- Restricciones

Permite representar descriptivamente la incertidumbre en el ámbito de desarrollo de software.

El metamodelo 'Uncertainty-of' está enfocado en el concepto de 'Incertidumbre' que es representado por la metaclase 'uncertainty_of' que tiene tres bloques principales.

- Incertidumbres
- Los efectos de la incertidumbre
- El Manejo de la incertidumbre

Vamos a describir cada uno de esos bloques, plantar sus definiciones y los atributos de cada metaclase y sus asociaciones.

Incertidumbres Contiene cuatro metaclases principales:

- uncertainty_of
- uncertainty_of_requirement
- uncertainty_of_resources
- uncertainty_of_process

2.4.2. Metaclase uncertainty_of

Definición

La metaclase "uncertainty_of" permite representar las dudas sobre la validez y la veracidad de la información relacionado con el proceso de desarrollo, las dudas sobre la cantidad, la capacidad de los recursos disponibles, también las imprecisiones en la expresión de dicha información.

Atributos:

- `isUncertain`: indica el estado de una incertidumbre sobre un elemento, `isUncertain` es Verdadero si una incertidumbre identificada no ha sido tratado, o su tratamiento todavía está en proceso.
- `descriptionOfuncertainty`: describe la incertidumbre detalladamente.
- `previsibility`: indica el tipo de incertidumbre. “Know Unknown” o “Unknown Unknown”

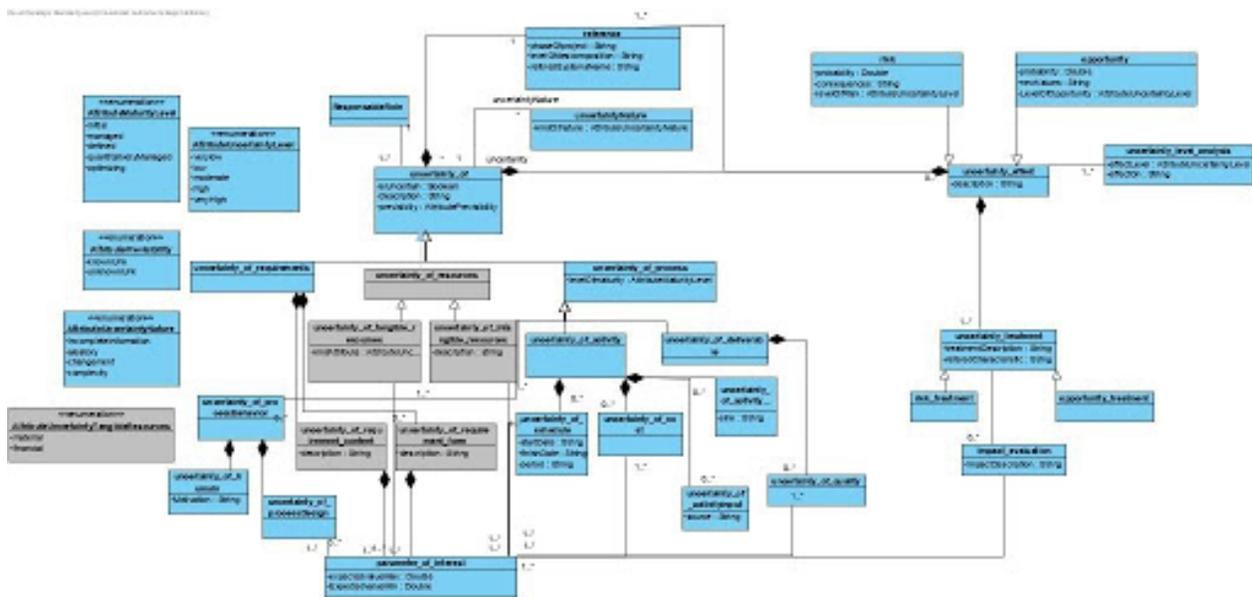


Figura 2.6: Metamodelo de incertidumbre en el desarrollo de software

Asociación:

- `reference`: Una incertidumbre debe ser asociado al menos a una referencia, la metaclassa “reference” permite indicar a qué elemento de la fase del proyecto la incertidumbre está asociado.
- `responsibleRole`: Una incertidumbre puede ser asociado a rol que oficialmente debe ser responsable de la gestión de esa incertidumbre.
- `uncertaintyNature`: Una incertidumbre puede ser asociado al menos a una natura de incertidumbre que es utilizado para indicar que si dicha incertidumbre proviene de una

falta de información (incomplete Information) , del entorno de la situación (aleatory), de un cambio (changement) o de la complejidad (complexity). El valor de este atributo es de tipo enumeración.

Las metaclasses **uncertainty_of_process** , **uncertainty_of_resources**, **uncertainty_of_requirement** son especialización de **uncertainty_of**. Heredan de los atributos y de sus asociaciones como se puede observar en la Figura 2.6

2.4.3. Metaclass **uncertainty_of_requirements**

Definición:

Permite de presentar las incertidumbre relacionado con los requerimientos del proyecto, en Scrum conocido con las Historias de Usuarios.

Atributos:

Hereda de los atributo de la clase de "uncertainty_of".

Asociado:

Está compuesta de la metaclass "uncertainty_of_requirement_content" y "uncertainty_of_requirement_form".

- **uncertainty_of_requirement_content**: Indica las incertidumbres derivadas del fondo. Incertidumbre que existe sobre lo que realmente el software debe de hacer.
- **uncertainty_of_requirement_form**: Indica las incertidumbres derivadas de la forma de presentación de los requerimientos. Incertidumbre que existe sobre la estructura de representar los requerimientos del sistema.

Están compuestas de la metaclass "parameter_of_interest". "parameter_of_interest": los parámetros de interés indican las propiedades cuantificables según el objetivo de proyecto como:

- La calidad.

- El costo
- La duración.

2.4.4. Metaclase `uncertainty_of_resources`

Definición

Permite de presentar las incertidumbre relacionado con los recursos disponibles para la realización del proyecto.

Atributos:

Hereda de los atributo de la clase de `uncertainty_of`.

Asociado:

La metaclase tiene dos sub-metaclasses: `uncertainty_of_tangible_resources` y `uncertainty_of_intangible_resources`

- `uncertainty_of_tangible_resources`: indica las incertidumbre relacionados a los recursos de naturaleza material (dinero, mobiliario)
- `uncertainty_of_intangible_resources`: indica las incertidumbre relacionados a los recursos de naturaleza inmaterial (Relación con los clientes, los procesos organizacionales, tecnología, capacidad, habilidades, motivaciones).

Están compuestas de la metaclase "parameter_of_interest".

2.4.5. Metaclase `uncertainty_of_process`

Definición

Permite representar a las incertidumbres asociadas a los elementos del proceso de desarrollo.

Atributos:

- `uncertainty_of_process` herada de los atributos de `uncertainty_of`.
- `levelOfMaturity`: corresponde al nivel de capacidad del proceso.

Asociación:

La metaclass `uncertainty_of_process` tiene tres submetaclass `uncertainty_of_activity`, `uncertainty_of_deliverable`, `uncertainty_of_processBehavior`.

- La metaclass `uncertainty_of_activity` está compuesta de las metaclasses siguientes:
 - `uncertainty_of_activityInput`: indica las incertidumbre relacionadas con los elementos de entrada de una actividad.
 - `uncertainty_of_activityOutput`: indica las incertidumbres relacionadas con los elementos de salida de una actividad.
 - `uncertainty_of_cost`: indica las incertidumbres relacionados al presupuesto de una actividad.
 - `uncertainty_of_schedule`: indica las incertidumbres relacionadas a la duración y detalla de la realización de una actividad. Una actividad es incierto cuando uno o varios de los elementos que la componen son inciertos.
- La metaclass `uncertainty_of_deliverable` está compuesta de la metaclass:
 - `uncertainty_of_quality`: indica las incertidumbre relacionadas con las propiedades de calidad de un entregable de un proceso. Un entregable es incierto cuando hay dudas sobre lo validez.
- La metaclass `uncertainty_of_processBehavior` está compuesta de las metaclasses:
 - `uncertainty_of_processDesign`: indica un posible solución alternativa para la realización de un proceso.

- `uncertainty_of_human`: indica las incertidumbres sobre las competencias y otras características de las personas durante la ejecución del proceso, sobre las decisiones que van a tomar.

Capítulo 3

Modelo propuesto para el manejo de incertidumbre

Este capítulo tiene como objetivo principal describir el modelo y la herramienta para el manejo la incertidumbre en los proyectos de desarrollo de software. Este modelo también ayudará a contestar a las preguntas de investigación de este trabajo que son:

- ¿Cómo se manifiesta la incertidumbre en los proyectos de desarrollo de software en los equipos novatos?
- ¿Cómo los miembros de los equipos de desarrollo de software novatos perciben la incertidumbre en los proyectos?
- ¿Cómo la incertidumbre impacta en las estimaciones de historias de usuarios en los proyectos de desarrollo de software en los equipos novatos?

Para poder responder a esas preguntas se implementó el Modelo para el Manejo de Incertidumbre MMI basando en la propuesta de Xiao Jing [18].

El MMI tiene como objetivo ayudar a los miembros de los equipos de desarrollo de software identificar las incertidumbres que les podrían impedir llevar a cabo de manera exitosa el proyecto [18]. El MMI tiene tres etapas principales que son:

- La identificación de las incertidumbres.
- La transformación las incertidumbres identificadas en riesgos.
- La evaluación de los riesgos.

En las siguientes secciones se describen estas etapas.

3.1. La identificación de las incertidumbres

La incertidumbre es el mundo desconocido, es la falta de seguridad, de confianza o de certeza sobre algo, especialmente cuando crea inquietud. En los proyectos de desarrollo de software la incertidumbre es expresión de ambigüedad e indeterminación. La incertidumbre tiene dos efectos principales sobre los proyectos. Un efecto positivo y un efecto negativo. En este caso lo más preocupante para los miembros de los equipos de desarrollo de software son las incertidumbres negativas ya que tienen impactos directo sobre los objetivos del proyecto que son en caso de Scrum las historias de usuarios, ya que las historias de usuarios representan las necesidades que deben satisfacer el proyecto. La identificación de las incertidumbres permite determinar el conjunto de escenarios indeseables que ponen en peligro el cumplimiento de los requerimientos del proyecto. Los miembros del equipo deben analizar esas incertidumbres para cada uno de los requerimientos del proyecto.

El objetivo de esta etapa es identificar las incertidumbres negativas que surgen o surgirán durante el proyecto, analizando los requerimientos del proyecto.

En la Figura 3.1 se pueden observar el diagrama de la etapa de identificación de incertidumbres.

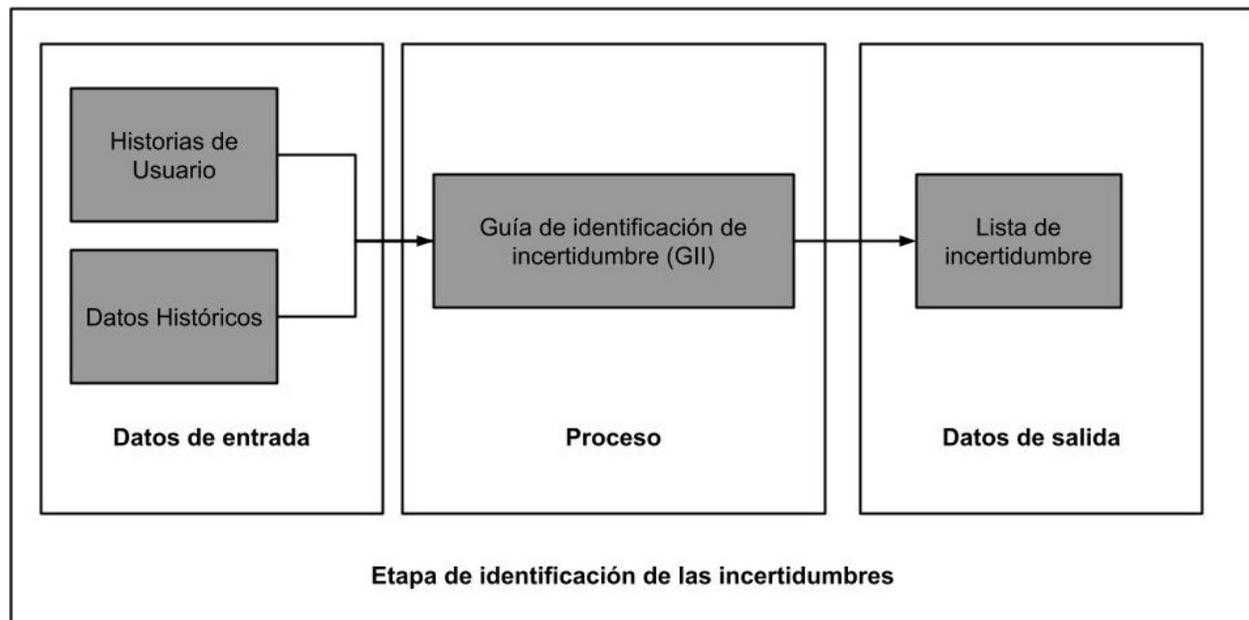


Figura 3.1: Etapa de identificación de incertidumbres

Los elementos de entrada de la etapa de identificación de las incertidumbres son:

- Los requerimientos del sistema.
- Los datos históricos.

A esos elementos de entrada se les aplica los pasos de la Guía de Identificación de Incertidumbre (GII) que se describe a continuación para obtener el elemento de la salida de la etapa de identificación de incertidumbres.

El elemento de salida de la etapa de identificación es: La lista de las incertidumbres identificadas del proyecto.

3.1.1. Guía de identificación de incertidumbre (GII)

En los proyectos de desarrollo de software identificar incertidumbre es una tarea compleja sobre todo para los equipos novatos. Por eso se propone esta guía. La GII consiste en un conjunto de preguntas que ayuda a los miembros de los equipos de desarrollo de software a identificar las incertidumbres relacionadas a cada una de las historias de usuarios de los proyectos. El proceso consiste en que cada miembro del equipo de desarrollo tiene que responder a esas preguntas para cada uno de los requerimientos del proyecto. Cuando el miembro del equipo no puede responder correctamente a una de esas preguntas. Se identifica una incertidumbre.

En el Cuadro 3.1 se presenta las preguntas de la guía para la identificación de incertidumbres.

Cuadro 3.1: Preguntas de la guía para la identificación de incertidumbres

Abreviatura	Nombre	Descripción
What	¿Qué?	¿Qué se debe hacer para cumplir este requerimiento?
Why	¿Por qué?	¿Porque debe cumplir este requerimiento?
How	¿Cómo?	¿Cómo se debe desarrollar este requerimiento cuál o cuáles son los pasos para desarrollar este requerimiento?
Who	¿Quién?	¿Quién o quienes de los miembros del equipo tiene las capacidades para resolver este requerimiento?
When	¿Cuándo?	¿Cuándo se debería empezar y terminar el desarrollo de este requerimiento? ¿Cuándo Se puede considerar el desarrollo de este requerimiento como exitoso?
Where	¿Dónde?	¿Dónde se implementaría este requerimiento en el proyecto? ¿Dónde va a impactar este requerimiento en el proyecto?

3.1.2. Identificación de la fuente de incertidumbre

- Identificación de la fuente de incertidumbre del riesgo.

Las Fuentes de incertidumbre en mundo de desarrollo de software son diversificadas entre ellas podemos citar:

- La Ambigüedad
- Los Cambios
- Las informaciones incompletas
- El entorno
- La inexperiencia

Cada incertidumbre identificada debe ser relacionada con una o más fuentes de incertidumbre.

- Rol identificador de la incertidumbre: Para cada incertidumbre identificada se debe considerar el perfil de la persona que detectó la incertidumbre considerando su rol en el equipo. Los diferentes roles son:
 - Project Owner
 - Scrum Master o facilitador.
 - Development team Member o el equipo de desarrollo.

3.2. Transformación de incertidumbres en riesgos.

Para poder contener la incertidumbre negativa la estrategia es poder aprender más sobre la naturaleza de la incertidumbre formulando el problema que representa dicha incertidumbre para el proyecto. [6]. Una vez que se revela una incertidumbre, se formula en riesgo el problema

que representa dicha incertidumbre para el proyecto, para después aplicar las técnicas de evaluación de probabilidad de ocurrencias y evaluación de los impactos [19], [20]. En la Figura 3.2 se pueden observar el diagrama de la etapa de transformación de incertidumbres en riesgos.

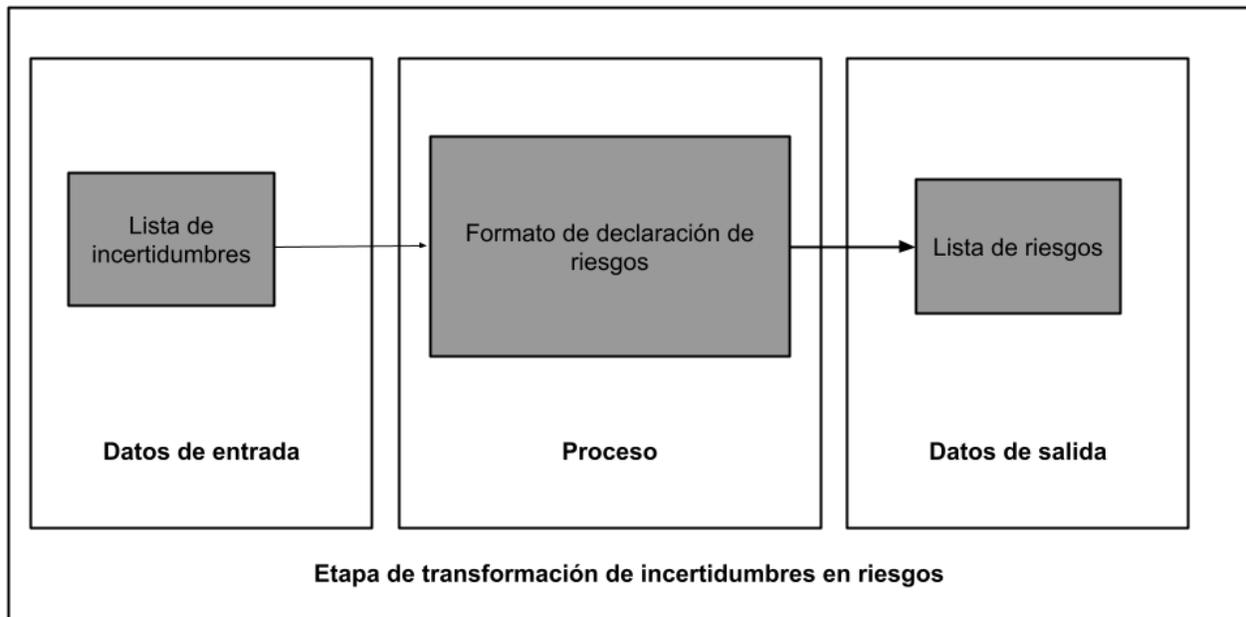


Figura 3.2: Etapa de transformación de incertidumbres en riesgos

Para la transformación de incertidumbres en riesgos se utiliza el formato de la declaración de riesgo. La declaración del riesgo debería una descripción concisa de la amenaza. El formato de la declaración de riesgo tiene la siguiente estructura como se puede ver el Cuadro 3.3.

Para cada riesgo se utiliza el enunciado siguiente: **Dado que** [CONDICIÓN], **existe la posibilidad de que** [SALIDA] **afecte negativamente** [ACTIVO], **lo que puede resultar en** [CONSECUENCIA] [21]. A continuación se describe cada uno de los elementos del enunciado.

Cuadro 3.2: Formato de declaración de riesgos

Abreviatura	Nombre	Descripción
Condition	Condición	Es una sola frase que describe la situación o el entorno clave actual basado en hechos que está causando preocupación, duda, ansiedad o malestar.
Departure	Salida	Describe un posible cambio en el plan del proyecto. Es un evento no deseado que se vuelve posible como resultado de la condición.
Asset	Activo	Representa el recurso principal que se ve afectado por el riesgo.
Consequence	Consecuencia	Es una frase única que describe los impactos negativos previsibles.

3.3. Evaluación de riesgos

El objetivo de este paso es identificar la fuente y analizar los efectos de las riesgos identificados 3.3.

Los elementos de evaluación de riesgos son:

- Estimación de la probabilidad de ocurrencia.
- Estimación del impacto.

3.3.1. Estimación de la probabilidad de ocurrencia

- Estimación de la POR: Es la posibilidad de materialización del riesgo identificado. Cada riesgo identificado debe ser relacionado con una probabilidad de ocurrencia. Las diferentes opciones con sus respectivas valores son:

- *Probabilidad de ocurrencia remota con un valor de 1:* el riesgo puede materializarse en un escenario muy remoto.
- *Probabilidad de ocurrencia inusual con un valor de 2:* el riesgo se podría materializar en escenarios muy reducidos

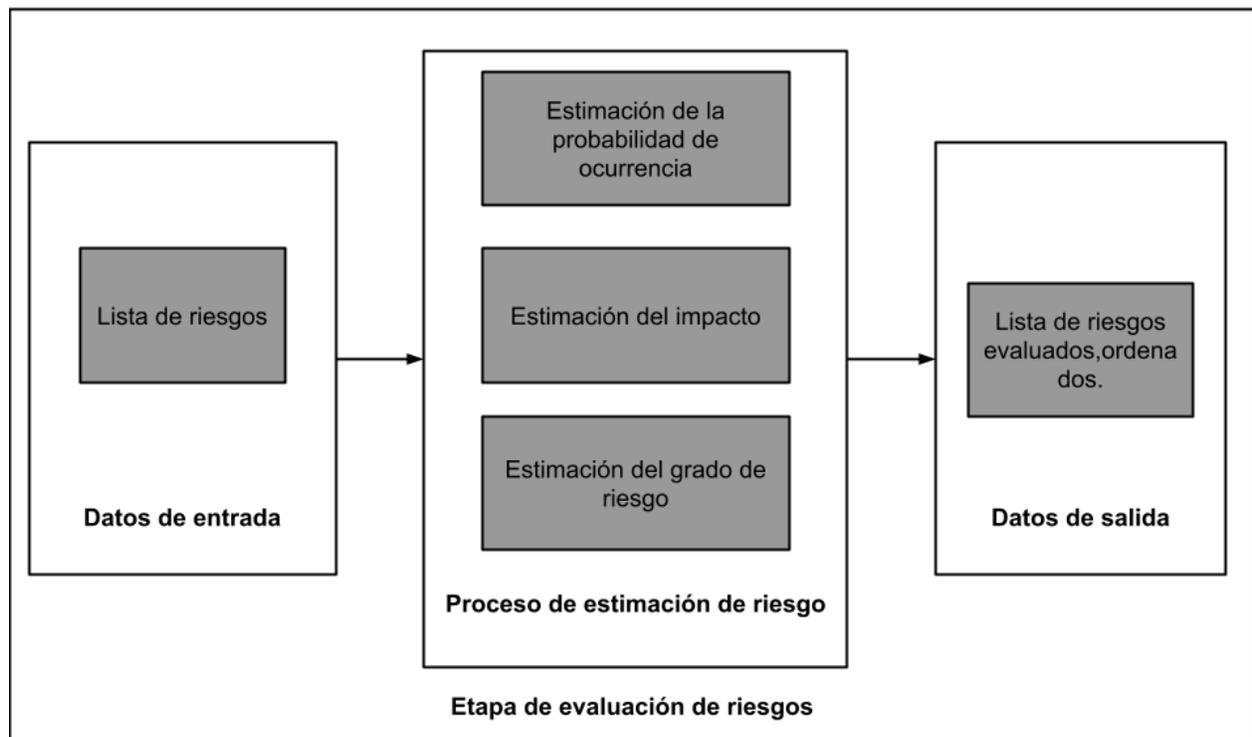


Figura 3.3: Etapa de evaluación de riesgos

- *Probabilidad de ocurrencia posible con un valor de 3:* el riesgo puede ocurrir en cualquier momento
- *Probabilidad de ocurrencia probable con un valor de 4:* es posible que ocurra en varias ocasiones.
- *Probabilidad de ocurrencia recurrente con un valor de 5:* el riesgo se va a materializar, tarde o temprano, en algún momento.

3.3.2. Estimación del impacto

- Estimación del impacto del riesgo: es las consecuencias probables en caso de ocurrencia de dicho riesgo. Se realiza:
 - Estimación del IRA.

- Estimación del IRT.
- Estimación del IRC.
- Estimación del IRM.

Los diferentes grados de impactos son:

- *Impacto insignificante con un valor de 1*
 - *Impacto moderado con un valor de 2*
 - *Impacto serio con un valor de 3*
 - *Impacto grave con un valor de 4*
 - *Impacto catastrófico con un valor de 5*
- Estimación de las fases del proyecto afectada por el riesgo: Consiste en indicar que fases o elementos de proyecto pueden ser afectados por el riesgo.
 - Estimación del NIR: Es asignar un número entre 0 y 100 que representa la gravedad del riesgo. Se calcula la fórmula siguiente:

$$NIR = (POR * IRA) + (POR * IRT) + (POR * IRC) + (POR * IRM)$$

El elemento de salida de este paso es la estimación del nivel de riesgos. Ver Figura 3.4.

3.4. Herramienta de gestión de incertidumbres (HGI)

Se implementó el modelo conceptual descrito anteriormente en una herramienta computacional llamada HGI que archiva materiales experimentales y permite replicaciones y el almacenamiento masivo de datos experimentales sobre manejo de incertidumbres en proyectos de desarrollo de software.

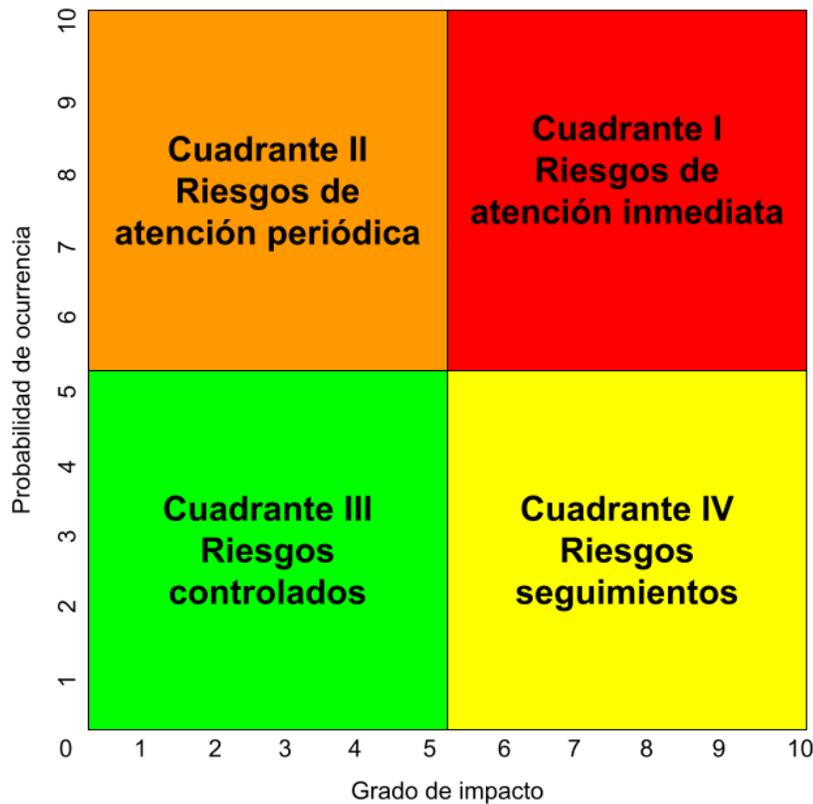


Figura 3.4: Matriz de riesgos

La HGI está desarrollada con las tecnologías PHP y JavaScript, MySQL. La herramienta trabaja mediante la tecnología cliente-servidor que se describe en la Figura 3.5:

En este tipo de tecnología, el servidor es el encargado de manejar todos los accesos a bases de datos, generación de páginas dinámicas y/o estáticas y validación de sesiones, presentándolas en el navegador (cliente). El esquema de la aplicación se describe a continuación:

Se presentan las interfaces donde a través de un navegador (cliente), el servidor procesa todas las peticiones y operaciones que realicen éstas. Dentro de estas interfaces se encontraran los módulos de la herramienta, con privilegios distintos para los usuarios que intenten acceder a ellas. Los módulos podrán ser accedidos mediante un URL, Los módulos están protegidos

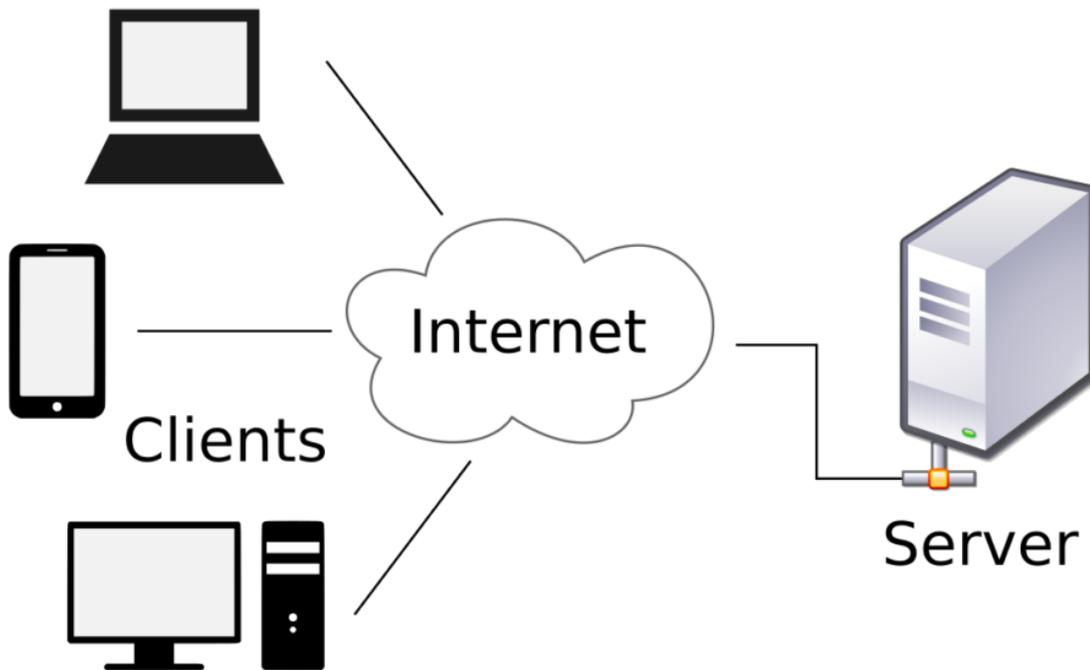


Figura 3.5: Tecnología cliente-servidor

con un login y un password.

HGI está conformada por diferentes módulos con se puede visualizar en el diagrama de la Figura 3.6 cada uno de esos módulos tiene su propósito específico para poder cumplir con los requerimientos del modelo propuesto para el manejo de la incertidumbre.

Módulo de usuarios: Este módulo permite el registro de los usuarios, guardado información como (Nombre completo, Correo, Institución, Contraseña). Dicha información sirve también para la autentificar los usuarios en la herramienta de cuenta con un módulo de login como se ilustra en la Figura 3.7. Los usuarios acceden a las interfaces mediante un login y un contraseña. Éstos son verificados en la base de datos, si son correctos se presenta la interfaz menú principal que permite acceder a todos los otros módulos.

Módulo de historias de usuarios y de estimación: Este módulo permite registrar las historias de usuarios de cada proyecto. Una historia de usuario es un componente clave

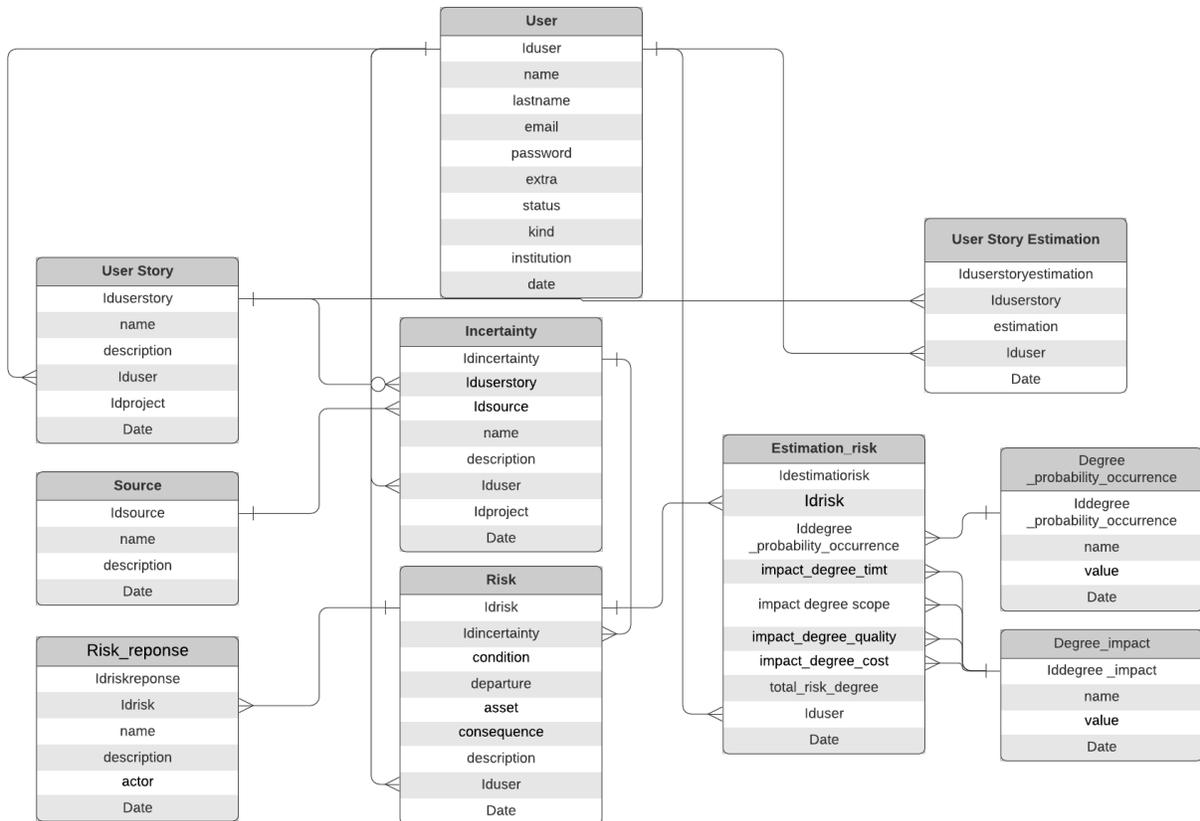


Figura 3.6: Diagrama de los módulos de la herramienta de gestión de incertidumbre.

del desarrollo de software ágil y de la metodología de gestión de incertidumbre ya que es una explicación general de una función de software a desarrollar escrita desde la perspectiva del usuario final y es el principal componente de los datos de entrada de la etapa de identificación de incertidumbres. La HGI permite para cada historia de usuario registrada guardar la estimación de esta dicha historia de usuario como se puede ver en la Figura 3.8 ya que la estimación es capital en Scrum, es una puesta en común de los requisitos a lograr, para definir entre todos una suposición lo más exacta posible, de lo que se puede lograr y en cuanto tiempo. la HGI permite a cada miembro del proyecto registrar para cada una de las historias de usuario un valor de su estimación. Los datos capturados por el usuario en el

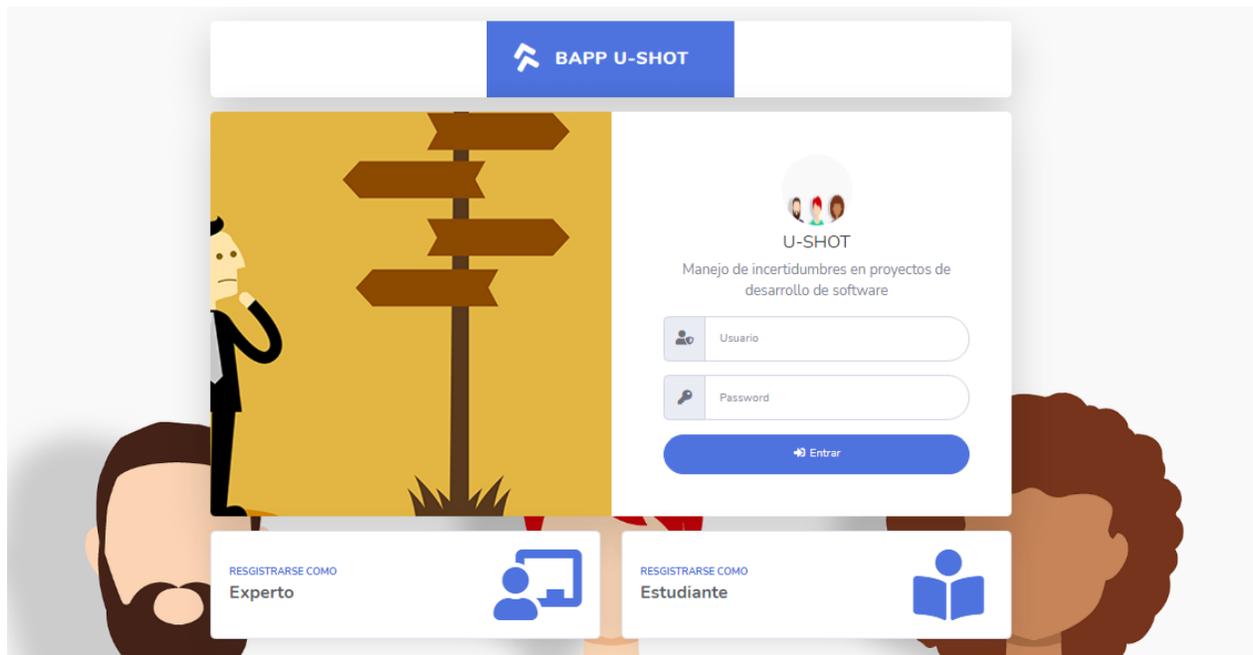


Figura 3.7: Módulo de login de usuarios

módulo de historias de usuarios y de estimación son:

- Título de la historia de usuario.
- El enunciado de la historia de usuario.
- El nombre del proyecto al que pertenece la historia de usuario.
- La estimación de la historia de usuario.

Módulo de registro de riesgo: Este módulo es capital en la herramienta, ya que permite registrar los riesgos identificados con el formato propuesto por la MMI como se puede ver en la Figura 3.9. En la sección historia de usuario se permite seleccionar una historia de usuario dentro del backlog del proyecto como dato de entrada y se registra el riesgo identificado relacionado con la historia de usuario en cuestión. Los datos capturados por el usuario en el módulo de registro de riesgo son:

Proyecto SoftwareMedico

Historia de Usuarios

← Regresar + Agregar

Show 10 entries Search:

Proyecto	Nombre	Historia de Usuario	Estimación	
UberEats/BarrioEats	HU-101	Como usuario final quiero agregar un producto a mi carrito para tener lo que quiero ordenar.	7	Estimar Editar Borrar
UberEats/BarrioEats	HU-102	Como usuario final quiero eliminar un producto de mi carrito para no pedirlo.	2	Estimar Editar Borrar
UberEats/BarrioEats	HU-103	Como usuario final quiero confirmar mi compra para obtener lo que hay en mi carrito.	3	Estimar Editar

Figura 3.8: Módulo de historias de usuario y de estimaciones

- En el campo "Given that" se registra la frase que describe la situación o el entorno clave actual basado en hechos que está causando preocupación, duda, ansiedad o malestar.
- En la campo "there is a possibility of". El posible cambio en el plan del proyecto, el evento no deseado que se vuelve posible como resultado de la condición.
- En el campo "adversely impacting" permite registrar el elemento principal que se ve afectado por el riesgo.
- En el campo "which can result in" se registra la frase que describe los impactos negativos previsibles.
- Fuente del riesgo donde se registra el origen del riesgo identificado que puede ser la incompletitud de la información, la falta de experiencia, la ambigüedad, los cambios, el entorno.
- El perfil del identificador donde se registra el rol del identificador dentro del equipo.

The screenshot shows a web interface for risk registration. At the top, there is a 'Capturar' button with a plus icon and a 'Regresar' button with a left arrow. Below this is a 'Historia de usuario' section with a document icon and a 'seleccionar' dropdown menu. The main section is titled 'Identificación' with a lightbulb icon. It contains two columns: 'Incertidumbre' and 'Identificador'. Each column has an 'Eligir' dropdown menu. Under 'Incertidumbre', there are two text input fields: the first contains 'Given that' and the second contains 'adversely impacting'. Under 'Identificador', there are two text input fields: the first contains 'there is a possibility of' and the second contains 'which can result in'. At the bottom right, there is a 'Siguiete >' button.

Figura 3.9: Módulo de registro de riesgo

Módulo de evaluación de riesgos: Después de de registrar un riesgo, la herramienta HGI permite evaluar este riesgo como se puede ver en la figura 3.10. Los datos capturados por el usuario en el módulo de evaluación de riesgo son:

- La probabilidad de ocurrencia que se registra seleccionando de los cinco niveles de probabilidad de ocurrencia el nivel adecuado según el riesgo. Las cinco opciones son: probabilidad de ocurrencia remota, probabilidad de ocurrencia inusual, probabilidad de ocurrencia posible, probabilidad de ocurrencia probable, probabilidad de ocurrencia recurrente.
- El nivel de impacto del riesgo sobre la meta, la calidad, el costo y el tiempo del proyecto seleccionando de los cinco niveles de probabilidad de impacto el nivel adecuado según el riesgo.
- Las fases del proyecto afectadas por este riesgo.

Evaluación

Probabilidad de Ocurrencia:

Estimación de impacto:

- Alcance:
- Tiempo:
- Calidad:
- Costo:

Nivel:

Fases afectadas:

- Daily scrum meeting
- Sprint retrospective
- Sprint review
- Sprint planning

[Siguiete >](#)

Figura 3.10: Módulo de evaluación de riesgos

Capítulo 4

Experimentación

En este capítulo presenta la metodología que guió la experimentación, donde se describen la muestra que formó parte del experimento y se detallan los pasos seguidos para comprobar la hipótesis planteada en el Capítulo 1.

4.1. Muestra

El experimento se realizó con cuatro proyectos de desarrollo de software en la Universidad Autónoma de Baja California y el Instituto Tecnológico de Tijuana. Los 18 estudiantes (17 hombres y 1 mujer) que integraban los equipos de desarrollo tenían al rededor de 20 años de edad y tenían conocimiento de las metodologías ágiles como Scrum, y conocimientos técnicos de programación debido a que estaban inscritos en el curso de ingeniería de software en dichas universidades.

A pesar de que los proyectos se desarrollaron en un ambiente académico, tenían clientes reales, los cuales exigían entregables y proveían los requisitos de desarrollo. A continuación se describen los cuatro proyectos que formaron parte del experimento:

- **Proyecto sevilla:** Es una aplicación que permite la interacción digital entre maestros y alumnos para facilitar el proceso de aprendizaje mediante el uso los recursos de la tecnología digital. La aplicación permite la organización de clases en línea, con opción para subir materiales académicos, además se pueden realizar evaluaciones y los alumnos pueden contestar dichas evaluaciones en línea. El equipo responsable del desarrollo de este proyecto tenía 6 integrantes. La Tabla 4.1 muestra algunas de las historias de usuario registradas para este proyecto son:

Cuadro 4.1: Muestra de historias de usuario del proyecto sevilla.

Nombre	Descripción
Crear salas.	Como usuario maestro, necesito poder crear salas, para poder brindarle a los alumnos dónde trabajar de manera privada.
Barra de progreso.	Como usuario alumno, Necesito saber mi progreso a lo largo de las actividades, con la finalidad de administrar mis tiempos.
Realizar evaluaciones.	Como usuario maestro, necesito realizar evaluaciones a los alumnos, para poder calificar a los alumnos.

- **Proyecto sistema de becas:** Es una aplicación Web que tiene como objetivo principal agilizar el proceso de solicitud, asignación y monitoreo de las becas a los estudiantes. Permite a los alumnos hacer solicitudes de becas de manera automática, proporcionando toda la información necesaria para dicho proceso. La plataforma otorga a los administradores los privilegios para poder examinar las solicitudes. Los administradores pueden rechazar o aceptar las solicitudes de becas. La aplicación permite a los involucrados visualizar en tiempo real el estado de las solicitudes de becas. El equipo responsable del desarrollo de este proyecto tenía 4 integrantes. La Tabla 4.2 muestra algunas de las historias de usuario registradas para este proyecto son:

Cuadro 4.2: Muestra de historias de usuario del sistema de becas.

Nombre	Descripción
Formulario de solicitud de becas.	Como alumno, necesito poder llenar el formulario de becas, para poder solicitar una beca.
Examinar las solicitudes de becas.	Como administrador, necesito tener control sobre las solicitudes de becas, para examinar las solicitudes de los aspirantes.
Consultar el estado de la solicitud.	Como alumno, necesito ver el estado de mi solicitud , para estar informado sobre el avance de mi solicitud.

- Proyecto Software médico: Es una aplicación web que permite realizar y administrar las citas en las clínicas médicas. Software médico permite mantener organizada la información de los pacientes, el médico puede consultar su agenda en tiempo real y revisar los expedientes de los pacientes fuera del consultorio. El equipo responsable del desarrollo de este proyecto tenía 3 integrantes. La Tabla 4.3 muestra algunas de las historias de usuario registradas para este proyecto son:

Cuadro 4.3: Muestra de historias de usuarios del Software Médico

Nombre	Descripción
Visualizar las citas médicas.	Como médico, necesito poder visualizar las citas médicas, para poder tener información de los paciente.
Registrar los datos de los pacientes.	Como paciente, quiero registrar mis datos, para poder agendar una cita
Consultar información general.	Yo como paciente, quiero consultar información sobre el COVID-19, para estar informado sobre la enfermedad.

- Proyecto BarrioEats: Es una aplicación web de ventas de comida donde se muestra al cliente el menú del restaurante con una descripción e imágenes de los platillos, de esta

manera el usuario final puede visualizar dicho menú, realizar el pedido o incluso modificar los ingredientes, asimismo mostrar el precio de los productos. El equipo responsable del desarrollo de este proyecto tenía 5 integrantes. La Tabla 4.4 muestra unos ejemplos de las historias de usuario de este proyecto.

Cuadro 4.4: Muestra de historias de usuarios del proyecto BarrioEats

Nombre	Descripción
Agregar un platillo al sitio web.	Como administrador del restaurante, quiero agregar platillos al sitio web, para que los clientes vean opciones y ordenen los productos.
Describir el platillo.	Como usuario administrador, quiero describir el platillo, para que el cliente final conozca los ingredientes del producto.
Agregar notas a al pedido.	Como usuario final, quiero agregar notas a mi pedido, para dar detalles sobre mi pedido.

En total formaron parte del experimento 18 estudiantes entre ambas universidades. En la siguiente sección se describe el proceso del experimento.

4.2. Proceso de experimentación

Los miembros de los equipos registraron las historias de usuarios de cada proyectos en uShot. Después, cada equipo realizó las estimaciones de complejidad de las historias de usuarios de su proyecto y de la misma manera se registraron en la plataforma. Los integrantes de los equipos utilizaron planning poker para la estimación de la complejidad.

En el siguiente paso, los integrantes aplicaron las preguntas para la identificación de incertidumbre para cada historia de usuario. Basados en esta incertidumbre identificada, los integrantes redactan un riesgo utilizando el formato de declaración de riesgo para el registro;

posterior a esto, realizaron una la estimación de la probabilidad de ocurrencia y del impacto en el proyecto. Con esta información la herramienta calcula el nivel de riesgo identificado.

La herramienta almacena los riesgos identificados, el nivel de riesgo asignado a cada uno, el rol de la persona en el equipo que identificó el riesgo, el impacto del riesgo, y el nivel de riesgo. Como resultado se obtuvo un dataset con la complejidad de cada historia de usuario y los riesgos identificados con sus niveles.

Capítulo 5

Resultados

En esta sección se presenta los resultados del experimento. El porcentaje de riesgos identificados por equipo es la siguiente como se puede observar en tabla 5.1. Del total de los riesgos considerados para este experimento, el proyecto Sevilla identificó 10 % de los riesgos, el proyecto Sistema de Becas identificó 38 % de los riesgos, el Proyecto Software Médico identificó 22 % de los riesgos y el Proyecto BarrioEats identificó 30 % de los riesgos.

Cuadro 5.1: Porcentajes de riesgos identificados por equipos

Equipos	Porcentajes de riesgos identificados
Proyecto Sevilla	10 %
Proyecto sistema de becas	38 %
Proyecto Software Médico	22 %
Proyecto BarrioEats	30 %

Las fuentes de incertidumbre de los riesgos identificados se repartieron de la siguiente manera. En general como se muestra en la Figura 5.1 la principal fuente de incertidumbre mencionado por los miembros de los equipos es la incompletitud de la información con el

38% de las fuentes identificadas. La segunda fuente de incertidumbre mencionado por los miembros de los equipos es la la falta de experiencia con el 33 % de las fuentes identificadas. La tercera fuente de incertidumbre generador riesgos es la ambigüedad con 16 % de los riesgos identificados. La cuarta fuente de incertidumbre identificadas es el entorno con 8 % de los riesgos, y la quinta son los cambios con 5 % de los riesgos identificados.

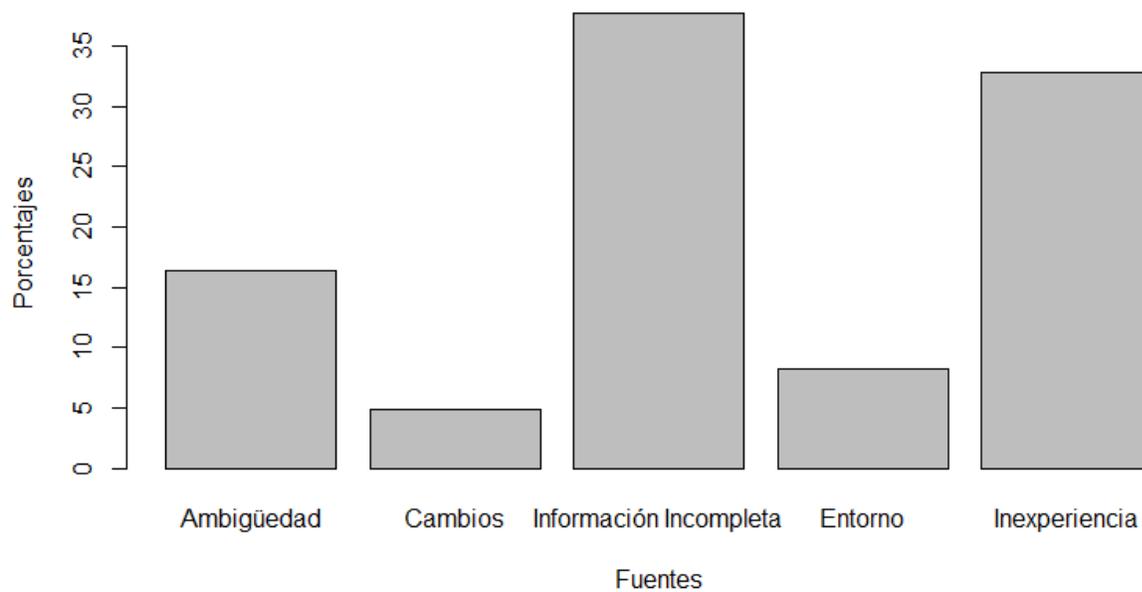


Figura 5.1: Las fuentes de incertidumbre de los riesgos identificados

Se analizó en este trabajo la relación que existe entre el nivel de riesgo y la estimación de las historias de usuarios, para eso se utilizó la correlación. La correlación es una medida estadística que expresa hasta qué punto dos variables están relacionadas linealmente. En el caso de este trabajo, nos permite analizar hasta que punto el nivel de riesgo está relacionado linealmente con la estimación de las historias de usuarios en los proyecto. La correlación no nos informa sobre causas y efectos. El coeficiente de correlación de la muestra, r , cuantifica la intensidad de la relación.

Para la interpretación de la magnitud del coeficiente de correlación se propone el cuadro 5.2. La relación se da entre el nivel de riesgo y la estimación de las historias de usuarios.

Cuadro 5.2: Interpretación del coeficiente de correlación

Rango de valores	Interpretación
$0,00 \leq r < 0,10$	Correlación nula
$0,10 \leq r < 0,30$	Correlación débil
$0,30 \leq r < 0,50$	Correlación moderada
$0,50 \leq r < 1,00$	Correlación fuerte

En el proyecto de Software Médico como resultado del impacto del nivel de incertidumbre sobre las estimaciones de historias de usuarios se registró una correlación de 0.9269 como se puede observar en la Tabla 5.3 y en la Figura 5.2.

Cuadro 5.3: Resultados proyecto de Software Médico

Indicador	Resultado
Correlación (r)	0.9269

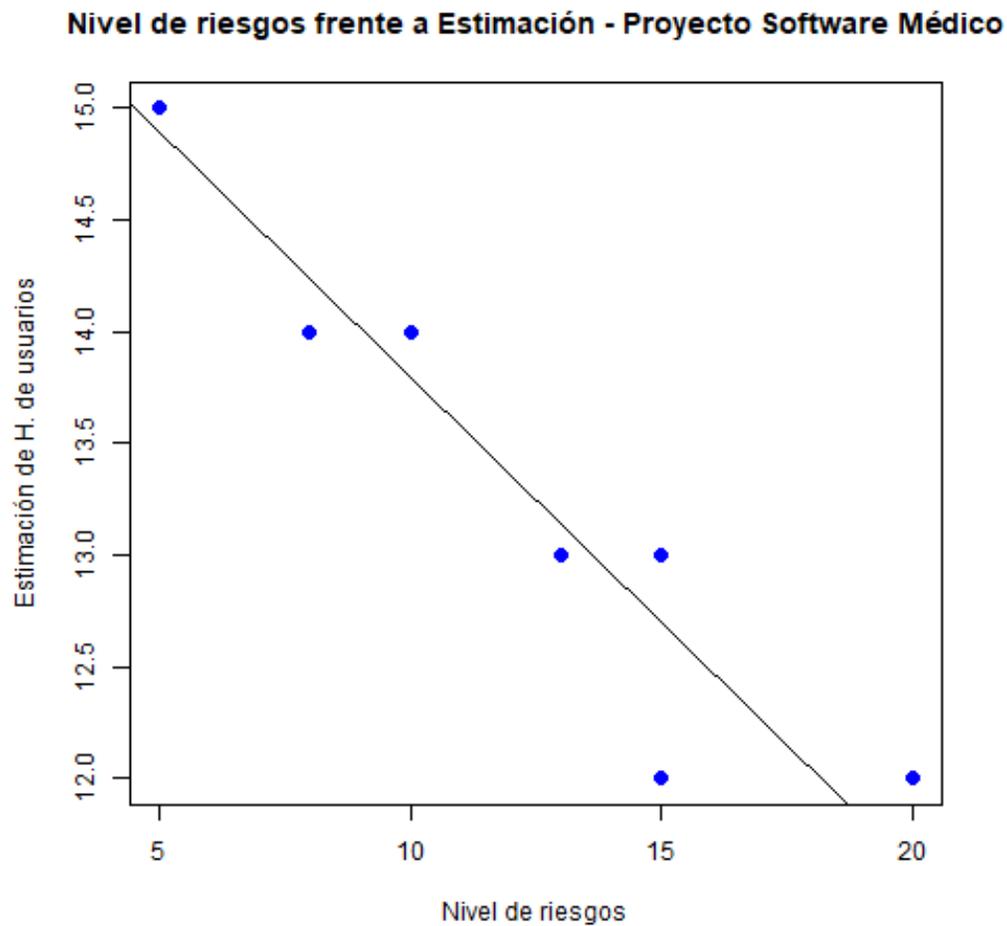


Figura 5.2: Impacto del nivel de incertidumbre sobre las estimaciones

En el Proyecto Sevilla como resultado del impacto del nivel de incertidumbre sobre las estimaciones de historias de usuarios se registró una correlación de 0.7619 como se puede observar en la Tabla 5.4 y en la Figura 5.3

Cuadro 5.4: Resultados proyecto Sevilla

Indicador	Resultados
Correlación (r)	0.7619

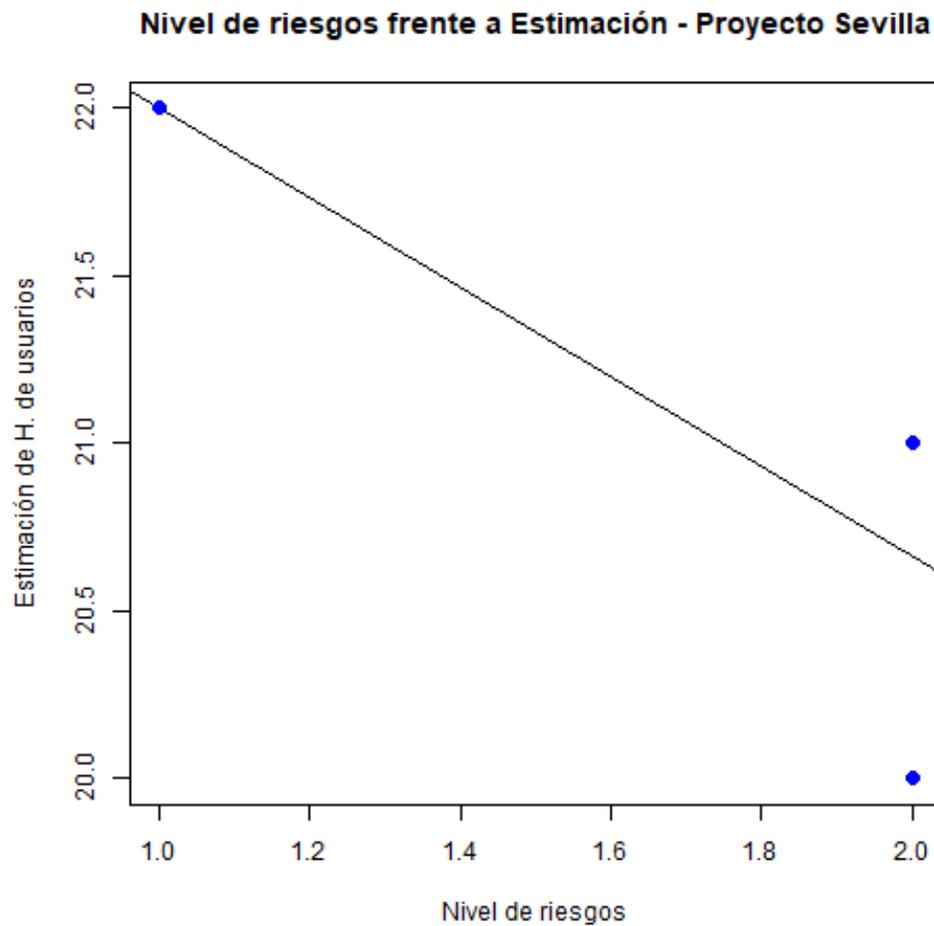


Figura 5.3: Impacto del nivel de incertidumbre sobre las estimaciones

En el Proyecto sistema de becas como resultado del impacto del nivel de incertidumbre sobre las estimaciones de historias de usuarios se registró una correlación de 0.7624 como se puede observar en la Tabla 5.5 y en la Figura 5.4

Cuadro 5.5: Resultado proyecto sistema de becas

Indicador	Resultados
Correlación (r)	0.7624

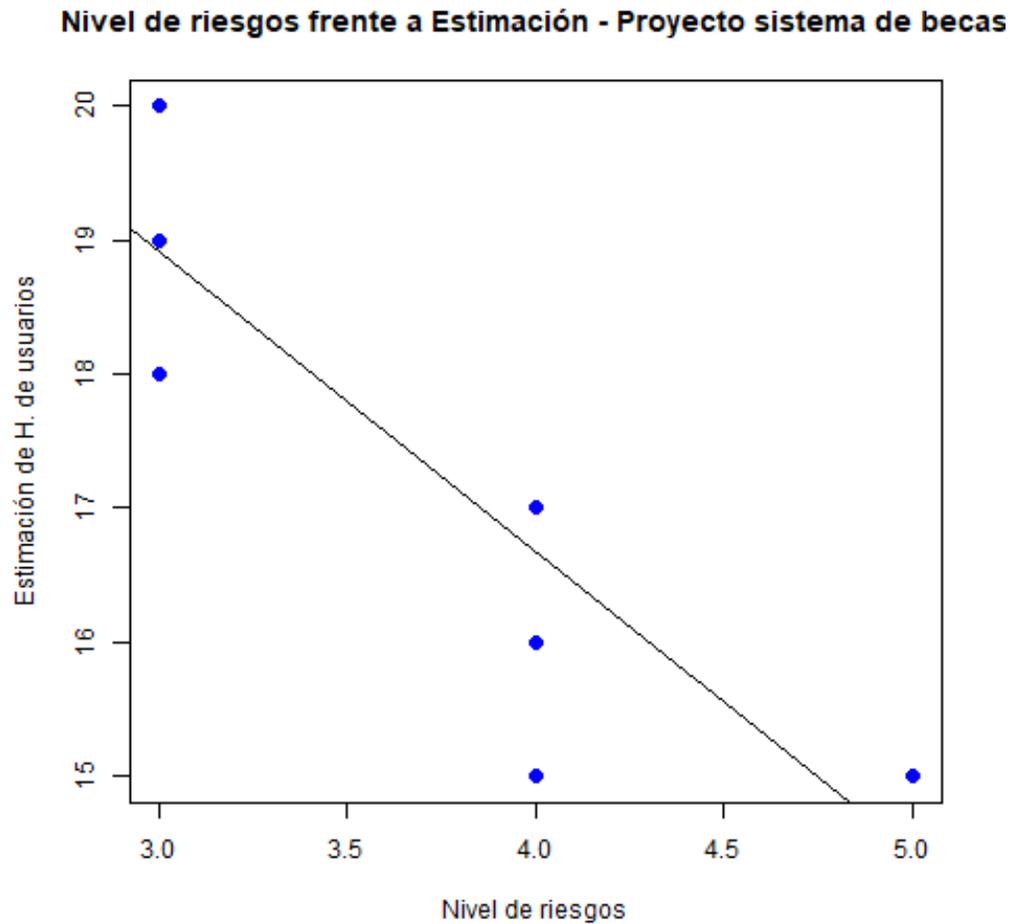


Figura 5.4: Impacto del nivel de incertidumbre sobre las estimaciones

En el Proyecto BarrioEats como resultado del impacto del nivel de incertidumbre sobre las estimaciones de historias de usuarios se registró una correlación de 0.8118 como se puede observar en la Tabla 5.6 y la Figura 5.5

Cuadro 5.6: Resultado proyecto BarrioEats

Indicador	Resultados
Correlación (r)	0.8118

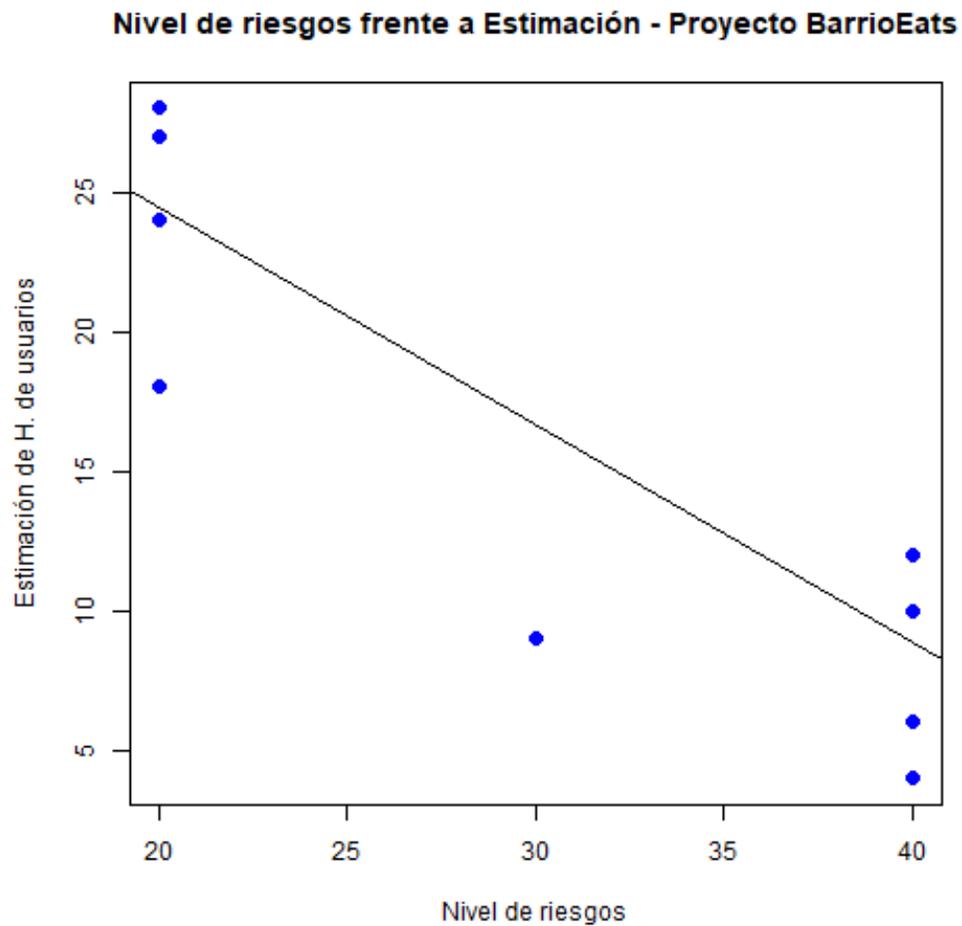


Figura 5.5: Impacto del nivel de incertidumbre sobre las estimaciones

Como se puede observar en los resultados, se obtuvo en promedio una correlación fuerte, lo que demuestra que las estimaciones de las historias de usuarios y el nivel de incertidumbre en los proyectos están relacionados linealmente. Considerando las correlaciones obtenidas, podemos deducir que el nivel de riesgo impacta en la estimación de las historias de usuarios.

Capítulo 6

Conclusiones, aportaciones y trabajo futuro

Este capítulo resume los resultados obtenidos de este estudio y presenta una conclusión. Además, enumera algunas contribuciones principales de este trabajo. Finalmente, establece algunas direcciones para posibles trabajo futuro.

6.1. Conclusiones

Tener en cuenta la presencia de incertidumbre en los proyectos de desarrollo de software es de una importancia capital. Además, lo que es aún más importante, es poder identificar las diferentes fuentes de incertidumbres, disponer de la capacidad para determinar los posibles riesgos provocados por dichas incertidumbres y sobre todo desarrollar la competencia de evaluación de esos riesgos. De hecho en la planificación estratégica en los proyectos de desarrollo de software la evaluación de los riesgos es determinante para el éxito de los proyectos.

Durante la investigación se realizaron experimentos poder poder determinar la relación

que existe entre las estimaciones de las historias de usuarios y el nivel de incertidumbre en los proyectos de desarrollo de software. Dado que, en promedio, los resultados muestran una fuerte correlación entre estas dos variables, podemos concluir que el nivel de riesgo impacta en la estimación de las historias de usuarios de los proyectos de software validando la hipótesis de este trabajo.

En este trabajo se desarrolló un modelo que ayuda a los miembros de los equipos de desarrollo de software en la identificación de incertidumbre en los proyectos, les ayuda en la comprensión de la incertidumbre formulando en riesgo el problema que representa dicha incertidumbre permitiendo así la evaluación de las probabilidades de ocurrencias y los impactos, y el análisis cuantitativo de los efectos perjudiciales para el proyecto.

6.2. Aportaciones

A continuación se describen algunas aportaciones de este trabajo:

- La identificación de incertidumbre es una tarea compleja sobre todo para los equipos novatos. Por eso este trabajo propone una guía de identificación de incertidumbre mediante preguntas.
- Una ampliación del metamodelo de la incertidumbre citado en la literatura. Para la representación abstracta y la comprensión del mundo la incertidumbre.
- En la actualidad, las tecnologías que se emplean en el desarrollo de software carecen de evidencias. Existe una falta de evidencia que apoye la mayoría de las creencias sobre las que se basa la construcción de software. La experimentación realizado este trabajo comprobó que:
 - Fuentes de incertidumbre identificados por los equipos novatos son:

- La incompletitud de la información.
- La falta de experiencia de los desarrolladores novatos.
- La ambigüedad de los requerimientos.
- Los cambios.
- El entorno.

6.3. Trabajo Futuro

Este trabajo logró varias aportaciones, sin embargo, aún queda trabajo por ser completado con el objetivo de mejorar este proyecto:

- Realizar experimentaciones a grandes escalas con más participantes.
- Analizar el efecto de la aplicación de las acciones de contingencia de los riesgos sobre el desempeño de los proyectos.
- Implementar una herramienta capaz de identificar los riesgos a partir de los requerimientos de los proyectos basándose en los datos históricos.

Bibliografía

- [1] A. Mundra, S. Misra, and C. A. Dhawale, “Practical scrum-scrum team: Way to produce successful and quality software,” in *2013 13th International Conference on Computational Science and Its Applications*, June 2013, pp. 119–123.
- [2] V. G. Ricardo Antunes, “A production model for construction: A theoretical framework,” *Department of Civil and Environmental Engineering, University of Auckland, 20 Symonds Street, Auckland 1142, New Zealand; E-Mail: v.gonzalez@auckland.ac.nz*, 2015.
- [3] B. Barry *et al.*, “Software engineering economics,” *New York*, vol. 197, 1981.
- [4] S. McConnell, *Software estimation: demystifying the black art*. Microsoft press, 2006.
- [5] R. B. Duncan, “Characteristics of organizational environments and perceived environmental uncertainty,” *Administrative science quarterly*, pp. 313–327, 1972.
- [6] M. L. Marinho, S. Sampaio, T. Lima, and H. P. Moura, “A guide to deal with uncertainties in software project management,” *arXiv preprint arXiv:1411.1920*, 2014.
- [7] K. N. Rao, G. K. Naidu, and P. Chakka, “A study of the agile software development methods, applicability and implications in industry,” *International Journal of Software Engineering and its applications*, vol. 5, no. 2, pp. 35–45, 2011.
- [8] K. Kaur, A. Jajoo, and Manisha, “Applying agile methodologies in industry projects:

- Benefits and challenges,” in *2015 International Conference on Computing Communication Control and Automation*, Feb 2015, pp. 832–836.
- [9] H. Sverrisdottir, H. Ingason, and H. Jónasson, “The role of the product owner in scrum-comparison between theory and practices,” *Procedia - Social and Behavioral Sciences*, vol. 119, pp. 257–267, 03 2014.
- [10] A. Mundra, S. Misra, and C. A. Dhawale, “Practical scrum-scrum team: Way to produce successful and quality software,” in *2013 13th International Conference on Computational Science and Its Applications*, June 2013, pp. 119–123.
- [11] D. K. Rigby, J. Sutherland, and H. Takeuchi, “Embracing agile,” *Harvard Business Review*, vol. 94, no. 5, pp. 40–50, 2016.
- [12] D. I. Guzmán, U. C. Islas, C. P. Corona, and B. E. Pedroza-Méndez, “Metodología ágil scrumban en el proceso de desarrollo y mantenimiento de software de la norma moprosoft,” *Research in Computing Science*, vol. 79, pp. 97–107, 2014.
- [13] F. Raith, I. Richter, R. Lindermeier, and G. Klinker, “Identification of inaccurate effort estimates in agile software development,” in *2013 20th Asia-Pacific Software Engineering Conference (APSEC)*, vol. 2. IEEE, 2013, pp. 67–72.
- [14] V. Mahnic and I. Rožanc, “Students’ perceptions of scrum practices,” 05 2012.
- [15] C. Torrecilla-Salinas, J. Sedeño, M. Escalona, and M. Mejías, “Estimating, planning and managing agile web development projects under a value-based perspective,” *Information and Software Technology*, vol. 61, pp. 124–144, 2015.
- [16] R. Popli, P. Malhotra, and N. Chauhan, “Management of time uncertainty in agile,” *International Journal of Software Engineering & Applications*, vol. 5, no. 4, p. 122, 2014.

-
- [17] N. Ionel, “Critical analysys of the scrum project management methodology,” 2008.
- [18] J. Xiao, “Gestion des incertitudes dans le processus de développement de systèmes complexes,” Ph.D. dissertation, 2009.
- [19] A. Jaafari, “Management of risks, uncertainties and opportunities on projects: time for a fundamental shift,” *International journal of project management*, vol. 19, no. 2, pp. 89–101, 2001.
- [20] M. V. Tatikonda and S. R. Rosenthal, “Technology novelty, project complexity, and product development project execution success: a deeper look at task uncertainty in product innovation,” *IEEE Transactions on engineering management*, vol. 47, no. 1, pp. 74–87, 2000.
- [21] N. NASA, “Risk management handbook,” NASA/SP-2011-3422, Version 1, Tech. Rep., 2011.