

Explorando la influencia de los roles de Belbin en la especificación de requisitos de software

Raúl Aguilar Vera¹, Mirna Muñoz Mata², Julio Díaz Mendoza¹, Juan Uacán Pech¹

avera@correo.uady.mx, mirna.munoz@cimat.mx, julio.diaz@correo.uady.mx,
juan.ucan@correo.uady.mx

¹ Facultad de Matemáticas, Universidad Autónoma de Yucatán, C.P. 97000, Mérida, Yucatán, México.

² Centro de Investigación en Matemáticas, Unidad Zacatecas, C.P. 98068, Zacatecas, Zacatecas, México.

DOI: [10.17013/risti.36.34-49](https://doi.org/10.17013/risti.36.34-49)

Resumen: Se presenta un estudio exploratorio en el que se analiza la influencia de la Teoría de Belbin en la calidad del producto generado con el desarrollo de requisitos software. El estudio se desarrolla mediante un experimento controlado en el que el factor manipulado por el investigador se corresponde con la forma en la que los equipos de desarrollo son integrados con estudiantes. Los resultados indican que existe una diferencia estadísticamente significativa entre la calidad de los productos generados por los equipos con roles compatibles, en comparación con la calidad de los productos generados por equipo integrados en forma aleatoria.

Palabras-clave: Especificación de Requisitos Software; Experimentación en Ingeniería de Software; Roles de Belbin.

Exploring the Influence of Belbin's Roles on Software Requirements Specification.

Abstract: An exploratory study is presented in which the influence of Belbin's Theory on the quality of the product generated with the development of software requirements is analyzed. The study is developed through a controlled experiment in which the factor manipulated by the researcher is the way in which the development teams are integrated with students. The results indicate that there is a statistically significant difference in the quality of the products generated by teams with compatible roles, compared to the quality of the products generated by randomly integrated teams.

Keywords: Belbin Roles; Experimentation in Software Engineering; Software Requirements Specification.

1. Introducción

Transcurrido medio siglo de su concepción, la Ingeniería de Software (IS) ha logrado acumular un cuerpo de conocimientos que es aceptado por profesionistas y académicos

vinculados con la joven disciplina, dicho cuerpo integra un conjunto de áreas relacionadas con procesos de desarrollo —Requisitos, Diseño, Construcción, Pruebas y Mantenimiento— y con procesos de gestión —Calidad, Configuración, Proceso de Software— asociados al primer grupo (Bourque & Fairley, 2014).

A nuestros días, existe una búsqueda constante de nuevos y mejores métodos, técnicas, herramientas, así como de buenas prácticas, que le permitan a la industria del software contar con mejores prácticas para ofrecer un Proceso Software menos complejo y más eficiente (Cuevas, 2002). Por otro lado, es conocido que muchas de las tareas vinculadas con el proceso software, se realizan en el contexto de equipos de desarrollo, y también es bien sabido que los principales problemas o causas de fracaso —en los proyectos software— no son de naturaleza tecnológica, sino que se deben a factores de naturaleza sociológica (De Marco & Lister, 1999); por lo anterior, uno de los factores que ha sido estudiado en los últimos años, es el factor humano, en particular, su incidencia en la mejora del proceso software. En un estudio reciente, Morales y Vega (2018) propusieron un catálogo de factores humanos críticos para el éxito de propuestas de mejora al proceso software, uno de dichos factores se relaciona con el rol que desempeña un Ingeniero de Software al interior del equipo de desarrollo.

El presente estudio, tiene como propósito explorar si el uso de la teoría de roles de Belbin (1993) en la integración de equipos de proyecto, tiene alguna influencia en una de las principales tareas vinculadas con el proceso de desarrollo software, la especificación del software.

2. Trabajos relacionados con este estudio

Diversos investigadores (Belbin, 1981; Briggs-Myers & McCaulley, 1985; Margerison & McCan, 1985; Mumma, 2005) afirman haber identificado roles que definen el comportamiento de los integrantes al interior de un equipo de trabajo, a dichos roles se les conoce como roles de equipo, y aunque no existe evidencia de alguna relación funcional con alguna tarea o proceso particular, su ausencia o presencia, al parecer tiene una influencia significativa en los logros del equipo (Aritzeta, Swailes & Senior, 2005).

Entre los estudios sobre roles de equipo, el trabajo de Belbin (1981, 1993) es posiblemente el más utilizado entre consultores e investigadores; Belbin mantiene que un rol de equipo hace referencia a la forma de comportarse, contribuir y relacionarse con otras personas en el trabajo, y aunque algunos de los roles son naturales, otros roles pudieran ser adoptados por el propio individuo, e incluso algunos pueden llegar a ser descubiertos luego de ser adoptados.

Los roles propuestos por Belbin pueden ser agrupados en torno al tipo de conducta en tres diferentes categorías (ver tabla 1): roles orientados a la acción (A), roles orientados a las personas o roles sociales (S), y roles orientados a los procesos cognitivos o roles mentales (M).

Tipo	Rol	Cracterísticas
Acción	Impulsor (<i>Sharper: S</i>)	Retador, dinámico, trabaja bien bajo presión. Tiene iniciativa y coraje para superar obstáculos.
	Implementador (<i>Implementer: I</i>)	Disciplinado, leal, conservador y eficiente. Transforma las ideas en acciones.
	Finalizador (<i>Completer-Finisher: CF</i>)	Esmerado, concienzudo, ansioso. Busca los errores y las omisiones. Realiza las tareas en el plazo establecido.
Mental	Cerebro (<i>Plant: P</i>)	Creativo, imaginativo, poco ortodoxo. Resuelve problemas difíciles.
	Monitor Evaluador (<i>Monitor-Evaluator: ME</i>)	Serio, perspicaz y estrategia. Percibe todas las opciones. Juzga con exactitud.
	Especialista (<i>Specialist: SP</i>)	Solo le interesa una cosa a un tiempo, cumplidor del deber. Aporta cualidades y conocimientos específicos.
Social	Coordinador (<i>Chairman: CH</i>)	Maduro, seguro de sí mismo. Aclara las metas. Tiene iniciativa y coraje para superar obstáculos.
	Investigador de Recursos (<i>Resouce Investigator: RI</i>)	Extrovertido, entusiasta, comunicativo. Busca nuevas oportunidades. Desarrolla contactos.
	Cohesionador (<i>Teamworker: TW</i>)	Cooperador, apacible, perceptivo y diplomático. Escucha y evita enfrentamientos.

Tabla 1 – Roles de Equipo propuestos por Belbin

El trabajo de Belbin (1993) ofrece también un conjunto de recomendaciones que permiten integrar equipos de trabajo con base en roles compatibles, así como mecanismos para la identificación del rol primario que un individuo puede asumir en un equipo de trabajo, en función de su comportamiento.

Entre los estudios reportados sobre el uso de la teoría de Roles de Belbin en el contexto de la IS, se encuentra el trabajo de Henry y Stevens (1999) quienes exploraron la utilidad de los roles de Belbin en la mejora de la efectividad de los equipos, particularmente, la influencia del liderazgo; dichos autores llevaron a cabo un experimento controlado con estudiantes, y entre sus hallazgos concluyen que la inclusión de un solo líder en el equipo, mejora su rendimiento, en comparación con equipos que tienen varios líderes o que no incluyen dicho rol.

Pollock (2009) exploró la influencia de contar con una diversidad de roles y personalidades, como estrategia para mejorar la efectividad del equipo, dicho autor comenta que aunque no encontró evidencia de que dicha diversidad influya de alguna manera, pudo observar que la presencia de ciertos roles, como son el caso del Impulsor, Coordinador o Finalizador, pueden incrementar la efectividad.

Estrada y Peña (2013) reportan un experimento controlado con estudiantes desempeñando actividades vinculadas con las fases de requisitos, diseño y codificación; las autoras concluyen que algunos roles presentan mayor aportación en ciertas actividades, particularmente señalan al rol Implementador en la tarea de codificación.

En el caso de nuestro grupo de investigación, en un estudio previo comparamos la calidad de la legibilidad del código generado por equipos integrados con roles compatibles —

con base en la Teoría de Belbin— con equipos formados con integrantes asignados aleatoriamente; los resultados obtenidos mediante un experimento controlado con estudiantes, nos permitieron concluir que los equipos que usaron como referencia la Teoría de Belbin, presentan resultados significativamente mejores que aquellos integrados mediante un criterio de aleatoriedad (Aguileta, Ucán y Aguilar, 2017). En un segundo estudio, los autores seleccionaron la tarea de medición software, sin embargo, los resultados obtenidos entre el primer experimento y su réplica experimental —similar interna— no permitieron concluir la existencia de diferencias significativas entre las variables vinculadas ni con el producto (número de puntos de función) ni con el proceso (tiempo utilizado); no obstante, con el estudio reflexionamos sobre la alternativa de explorar la influencia de los diferentes roles de equipo en tareas individuales vinculadas con el proceso software (Aguilar, Díaz y Ucán, 2019).

3. El Proceso de Especificación de Requisitos Software

El propósito de la especificación de requisitos software, es definir el conjunto de funcionalidades, atributos de calidad y restricciones que describen en forma detallada el producto software que se desea desarrollar para un cliente u organización (Bourque y Fairley, 2014). El producto final de dicho proceso se conoce como Documento de Especificación de Requisitos Software; el estándar IEEE Std 830-1998 establece el contenido y las cualidades de un buen documento de especificación de requisitos de software (IEEE CS-SES, 1998).

El proceso de desarrollo de requisitos software consta de cuatro fases (Aguilar, *et al*, 2017). La primera etapa, denominada Elicitación de requisitos, se refiere a la identificación de las necesidades reales y las restricciones operativas expresadas por las partes interesadas de una organización; para obtener dicha información, el Ingeniero de Software diseña una estrategia en la que combina un conjunto de técnicas tradicionales, contextuales, grupales y/o cognitivas (Yousuf & Asger, 2015). La segunda etapa del proceso, el de Análisis, tiene como objetivo garantizar la calidad de los requisitos antes de incorporarlos al documento de especificación; en esta etapa, se definen los límites del sistema de software y su interacción con el entorno. Las solicitudes o requerimientos de los interesados en el nuevo sistema se traducen en requisitos de software. La etapa de Especificación consiste en documentar los requisitos de software acordados, con un nivel de detalle apropiado, y generalmente se usa un modelo de documento, el cual se redacta en términos comprensibles para las partes interesadas. Finalmente, en la última fase del proceso, la de Validación, se realiza una revisión cuidadosa de un conjunto de atributos de calidad vinculados tanto a los requisitos individuales, como al documento de manera integral; el objetivo es identificar problemas en el documento de especificación de requisitos software antes de que se pueda utilizar como base para el diseño del nuevo Sistema. En ocasiones, se utiliza la técnica de casos de uso para documentar requisitos funcionales; dicha técnica permite especificar el comportamiento de un sistema, mediante la secuencia de interacciones entre un sistema y alguien o algo que usa alguno de sus servicios. Es importante destacar que los casos de uso son documentos de texto, no diagramas, y el modelado de casos de uso es principalmente un acto de redactar texto, no de dibujar; no obstante, el Lenguaje de Modelado Unificado, incorpora un tipo de diagrama que permite ilustrar para un sistema, los diferentes casos de uso, actores, así como sus relaciones.

La calidad es difícil de definir, es subjetiva y depende de los diferentes puntos de vista y del contexto sobre el constructo al que se aplica el concepto (Kitchenham & Pfleeger, 1996). En el caso del producto generado con el proceso de especificación de requisitos software, podemos utilizar el estándar IEEE 830-1998 como punto de partida para definir la calidad de los requisitos (ver Tabla 2).

Atributos de Calidad	Definición
Complejidad	Deben considerarse todos los elementos importantes que son relevantes para cumplir con las diferentes tareas del usuario. Esto incluye requisitos funcionales y no funcionales.
Consistencia	Los requisitos establecidos deben ser consistentes con todos los demás requisitos y otras restricciones importantes
Correctitud	Los requisitos implementados deben reflejar el comportamiento esperado de los usuarios y clientes.
Trazabilidad	Debería ser posible referirse al requisito de una manera fácil, tanto para los artefactos derivados del requisito como para aquellos de los que se deriva.
Inambigüedad	Los requisitos solo deben tener una interpretación posible
Verificabilidad	Debe haber un proceso para que una máquina o un ser humano verifique si el requisito se cumple o no

Tabla 2 – Ejemplos de atributos de calidad para el DER

El proceso de verificación llevado a cabo por el ingeniero de software debe realizarse en dos niveles. En un primer nivel, los atributos de calidad deben evaluarse individualmente para los requisitos de software —p.e. trazabilidad, inambigüedad, verificabilidad— y en el segundo nivel, los atributos de calidad se verifican en el documento de especificación de requisitos de manera integral —p.e. completitud, consistencia, correctitud.

4. Planificación del Experimento Controlado

Aunque en la propia definición propuesta para Ingeniería de Software por parte de la IEEE (1990) considera en su segundo inciso, el estudio —investigación— de los enfoques vinculados con el quehacer del proceso software, en el siglo pasado no se le concedió la importancia debida (Juristo y Moreno, 2001); por fortuna, a nuestros días existe una comunidad que utiliza métodos de investigación, fundamentalmente empíricos, que buscan acrecentar el cuerpo de conocimientos disponible para la disciplina (Glass, Vessey, & Ramesh, 2002).

Una de las metodologías empíricas más recurridas en el campo de la IS, es la experimentación, particularmente, la experimentación en entornos controlados (Basili, Selby & Hutchens, 1986); dicha metodología tiene como objetivo identificar las causas por las cuales se producen ciertos resultados, su aplicación nos ayuda a identificar y en su caso, comprender las posibles relaciones entre factores y variables dependientes, ambos parámetros inmersos en quehacer del proceso software. Entre los elementos característicos de los estudios controlados encontrados en la literatura, se destaca el uso de grupos de estudiantes como sujetos experimentales; Genero, Cruz-Lemus y Piattini (2014) indican al respecto, que dicha muestra académica permite al investigador obtener

evidencia preliminar para confirmar o refutar hipótesis que pueden ser contrastadas posteriormente en contextos industriales.

El presente estudio tiene como propósito explorar, mediante la ejecución de un experimento controlado con estudiantes, la influencia del uso de la Teoría de Belbin en la integración de equipos de desarrollo con integrantes que presentan roles compatibles, en la calidad del producto de una de las fases más importantes del proceso de desarrollo, el documento de especificación de requisitos software.

4.1. Factor y Tratamientos

El factor considerado en el experimento, se corresponde con la estrategia utilizada para la integración de los equipos de trabajo; se consideraron dos tratamientos:

- *Equipos Compatibles (EC)*: Equipos de trabajo de tres integrantes, conformados con roles compatibles, de acuerdo con la Teoría de Belbin.
- *Equipos Tradicionales (ET)*: Equipos de trabajo de tres integrantes, seleccionados en forma aleatoria.

4.2. Hipótesis y Variables

Con el objetivo de explorar si los equipos integrados con base en la Teoría de Roles de Belbin —a los que llamaremos Compatibles— generan Documentos de Especificación de Requisitos Software (DERS) significativamente diferentes, en términos de su calidad, de aquellos obtenidos por equipos integrados en forma aleatoria —denominados Tradicionales— se propuso un par de hipótesis:

- H01: La calidad del DERS generado por los equipos tradicionales es igual a la calidad del DERS generado por los equipos compatibles.
- H11: La calidad del DERS generado por los equipos tradicionales difiere de la calidad del DERS generado por los equipos compatibles

La variable dependiente, Calidad del documento ERS, se obtuvo a través de la validación —por parte de un analista experto— de cada uno de los productos generados por los equipos de desarrollo participantes.

4.3. Instrumento para Evaluación de la Calidad del DERS

Para evaluar los productos generados por los equipos de desarrollo, se diseñó un instrumento que consta de 27 ítems que utilizan una escala Likert para obtener la evaluación del DERS por parte de un analista. El instrumento diseñado consideró atributos de calidad vinculados con requisitos individuales (8 ítems) así como con el documento de especificación en su conjunto (16 ítems). Adicionalmente, se incluyen tres ítems con los que se verifica la correspondencia entre los casos de uso y los requisitos funcionales.

4.4. Diseño Experimental

El diseño experimental más apropiado para nuestro estudio, es el diseño factorial con una fuente de variación y dos tratamientos (ver tabla 3), la variable dependiente es una

métrica numérica que se obtiene a través del instrumento de evaluación de los productos generados por los equipos de desarrollo.

	Tratamientos	Equipos
Factor	<i>Equipos Belbin</i>	<i>Métrica sobre la Calidad del DERS</i>
	<i>Equipos Tradicionales</i>	<i>Métrica sobre la Calidad del DERS</i>

Tabla 3 – Diseño factorial con una fuente de variación

4.5. Propuesta del Análisis Estadístico

Para el análisis de datos, se generará una sección de estadística descriptiva con tablas de resumen de los valores obtenidos, así como gráficos de caja y bigotes para analizar visualmente el comportamiento de los mismos; en el caso del análisis inferencial, elegimos el análisis de varianza (ANOVA) de una vía, debido a que nos permite realizar pruebas de hipótesis para determinar si existe o no diferencias significativas entre las medias de los valores recogidos en la variable dependiente, para los diferentes niveles del factor (tratamientos). El ANOVA (Gutierrez y de la Vara, 2012) es una técnica que permite construir con los datos, un modelo estadístico que describe el impacto de un solo factor categórico sobre una variable dependiente.

5. Ejecución del Experimento Controlado

Con base en el plan del experimento, se trabajó con estudiantes de la Licenciatura en Ingeniería de Software que se oferta en la Universidad Autónoma de Yucatán, en particular alumnos inscritos a un curso de la asignatura Desarrollo de Requisitos de Software. A lo largo de período escolar que dura el curso, el profesor —autor principal— impartió al grupo los temas relacionados con el proceso de desarrollo de requisitos de software, y desde la conclusión de la unidad donde se abordan las técnicas de educación de requisitos, los alumnos integrados en equipos de trabajo, formados por el profesor, comenzaron a aplicar el proceso de requisitos para un problema del mundo real seleccionado por los propios estudiantes. El producto final de la asignación consistió en un DERS basado en el IEEE Std 830-1998, e incluye, como sección adicional, los casos de uso que se requieren, de acuerdo con los servicios especificados para el proyecto.

5.1. Participantes

La muestra por conveniencia utilizada para en el experimento estuvo conformada por 34 estudiantes que cursaban la asignatura denominada “Desarrollo de Requisitos de Software” que se encuentra ubicada en el quinto semestre del Plan de Estudios 2009 de la carrera antes citada. Con 32 de dichos estudiantes se integraron 8 equipos de trabajo de cuatro integrantes cada uno; para lo anterior se utilizó la información obtenida de cada estudiante —rol primario— luego de administrar el instrumento de autopercepción de Belbin (instrumento propuesto por Belbin para identificar el papel que puede desempeñar el sujeto que lo responde en un equipo de trabajo). Se integraron cuatro equipos con roles compatibles (EC), es decir, roles que no tienen conflicto entre ellos y

otros cuatro equipos como grupo de control, integrados con estudiantes asignados de forma aleatoria (Equipos tradicionales: ET).

Los dos estudiantes restantes integraron un noveno equipo para efectos de la asignatura, sin embargo, no fueron incluidos en el estudio en virtud del número diferenciado de estudiantes. Por otro lado, por motivos personales, uno de los estudiantes que integraba a uno de los EC (equipo 3) abandonó la asignatura cuando estaban trabajando en la primera fase del proceso de requisitos, por ello fue necesario hacer un ajuste en el equipo, y también fue descartado para el estudio. La tabla 4 ilustra la asignación de los equipos a los tratamientos.

Tratamientos	Equipos
EC	I, II, IV
ET	V, VI, VII, VIII

Tabla 4 – Asignación de sujetos experimentales por tratamiento

5.2. Sujetos Experimentales

Dado que las mediciones se obtendrían de los productos generados por los equipos de desarrollo (DERS), los sujetos experimentales en este caso fueron finalmente los siete equipos de trabajo integrados por el profesor. La conformación de los tres equipos basados en la teoría de Belbin se ilustra en la Figura 1; las letras dentro de los círculos indican el identificador utilizado para los roles de equipo desempeñados por cada participante (consultar tabla 1), y así se puede observar la conformación de los tres equipos compatibles (EC). El número romano al centro los cuatro círculos (que representan a los integrantes) es un identificador para el equipo –sujeto experimental.

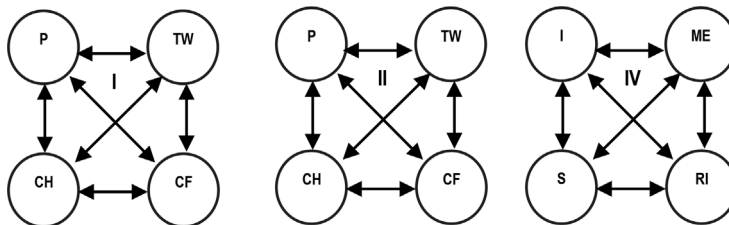


Figura 1 – Equipos integrados con la Teoría de Belbin (Equipos Compatibles: EC)

Durante el desarrollo del curso, los equipos realizaron presentaciones parciales del avance de su proyecto, no obstante, la entrega del producto por evaluar en la asignatura, se programó para el final del curso; dichos DERS generados por los equipos, son los que se usaron posteriormente para evaluar y obtener las métricas de calidad que serían utilizadas como variable respuesta en el experimento controlado; es importante destacar que la evaluación de los productos, para efectos de evaluación en el curso, fue independiente de la evaluación realizada a través del instrumento diseñado para nuestro experimento.

5.3. Proceso de generación de los Objetos Experimentales

El experimento se realizó en cuatro momentos diferentes; en una primera sesión, se administró el instrumento de autopercepción para identificar los roles primarios que podrían desempeñar los participantes; en un segundo momento, el maestro trabajó con los estudiantes a lo largo del curso sobre los temas de la asignatura, y a partir de la tercera unidad los siete equipos desarrollaron las actividades del proceso de requisitos. Posteriormente, los equipos entregaron el DERS —objetos experimentales. Finalmente, con los DERS generados por los equipos, se invitó a dos analistas expertos a evaluar la calidad de los productos generados, utilizando para ello el instrumento citado en la sección 4.3.

La tabla 5 resume las métricas obtenidas con doble evaluación.

Equipo	Evaluación 1	Evaluación 2
I	106	107
II	119	126
IV	121	106
V	99	95
VI	108	89
VII	107	105
VIII	91	89

Tabla 5 – Asignación de sujetos experimentales por tratamiento

6. Análisis de Datos del Experimento

En esta sección se presenta el análisis estadístico de los dos conjuntos de métricas obtenidas por los expertos invitados al experimento.

6.1. Análisis Estadístico de la Evaluación 1

Esta sección presenta tanto el análisis estadístico descriptivo como el análisis estadístico inferencial de las mediciones generadas por el evaluador 1. La Tabla 6 presenta algunas de las medidas más importantes de tendencia central y variabilidad.

Factor	#	μ	Mediana	σ^2	Min	Max	Rango
EC	3	115.333	119.0	66.3333	106.0	121.0	15.0
ET	4	101.25	103.0	62.9167	91.0	108.0	17.0

Tabla 6 – Resumen Estadístico de la Evaluación 1

Podemos observar que tanto la media como la mediana de los EC es mayor que la del ET, y que los EC presentan un menor rango; no obstante, los EC presentan una mayor varianza.

Para comparar visualmente los dos tratamientos, generamos un diagrama de caja y bigotes, dicho gráfico podemos nos permite observar la dispersión y la simetría de ambos conjuntos de datos (ver Fig. 2).

En la Figura 2 podemos ver que presentan ligeras superposiciones en los tratamientos, por lo que es probable que presenten diferencias significativas entre ellos. También podemos observar que no existen grandes diferencias en la dispersión de los tratamientos, aunque los ET tienen mayor simetría. Con el propósito de evaluar si las diferencias observadas en el análisis descriptivo previo, resultaban significativas, se plantearon las siguientes hipótesis estadísticas en torno a la Calidad del DERS.

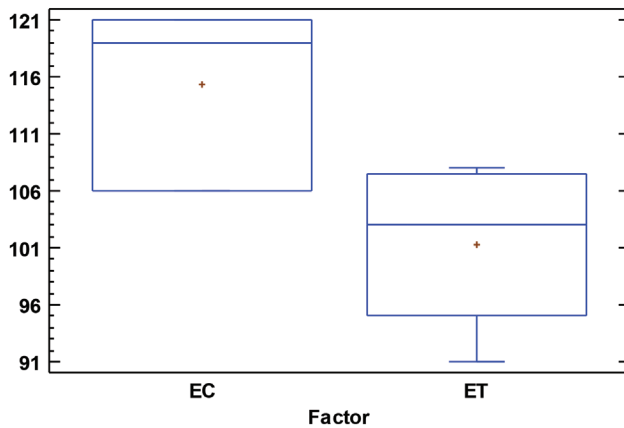


Figura 2 – Diagrama de Caja y bigotes para los tratamientos de la Métrica Calidad del DERS (Evaluación 1)

$$H_{0_{C-DERS}}: \mu_{EC} = \mu_{ET} ; H_{1_{C-DERS}}: \mu_{EC} \neq \mu_{ET} \quad (I)$$

El resultado de evaluar con ANOVA dichas hipótesis estadísticas, se ilustra en la tabla 7.

Métrica	F	p-valor
C-DERS	5.29	0.0698

*p-valor < 0.10

Tabla 7 – ANOVA para la variable Calidad del DERS (Evaluación 1)

La tabla ANOVA descompone la varianza de la variable en estudio en dos componentes: un componente intergrupo y un componente en grupos; dado que el valor p de la prueba F —en la variable Calidad del DERS— es menor que 0.1 (0.0698) la hipótesis de nulidad puede ser rechazada, es decir, la hipótesis alternativa permite afirmar que existe diferencias entre las medias de los dos tratamientos con un 10% de significancia.

Es importante mencionar que el modelo ANOVA tiene asociado tres supuestos que es necesario validar antes de utilizar la información que nos ofrece; los supuestos del modelo son: (1) Los errores experimentales de sus datos se distribuyen normalmente,

(2) existe independencia entre las muestras, y (3) No existe diferencia entre la varianza de los tratamientos (Homocedasticidad)

Para validar el primer supuesto, usaremos el gráfico de probabilidad normal de los residuos. Es una técnica gráfica para evaluar si un conjunto de datos se distribuye de acuerdo con la distribución normal. Como se puede ver en la gráfica que se ilustra en la figura 3a, los puntos no muestran desviaciones de la diagonal, por lo que es posible asumir que los residuos tienen una distribución normal.

En cuanto al supuesto de independencia de los datos, generamos un gráfico de residuales versus secuencia; en este caso en la figura 3b podemos observar que no existe una tendencia en los datos, por lo que es posible suponer que los datos provienen de poblaciones independientes.

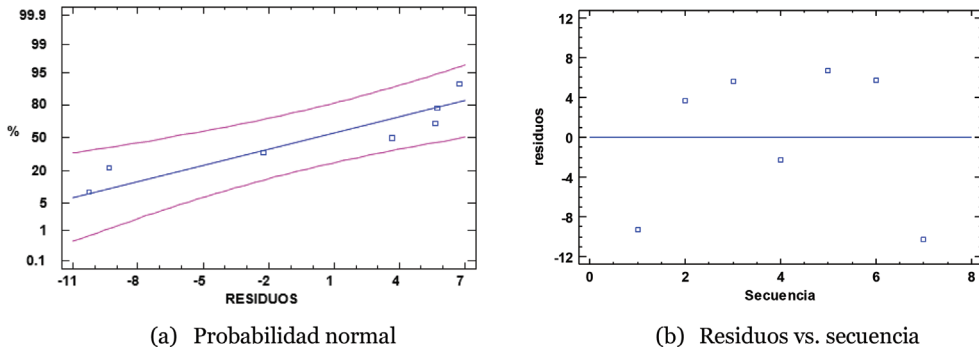


Figura 3 – Gráficos generados con los residuos (Evaluación 1)

Finalmente, la prueba de Levene permite evaluar diferencias significativas entre las varianzas de los dos conjuntos de datos (homocedasticidad) y puesto que el p valor es mayor que 0.05 (ver tabla 8), no existe una diferencia estadísticamente significativa entre las desviaciones estándar, con un nivel del 95.0% de confianza.

Prueba	p-valor
0.000134924	0.9912

Tabla 8 – Prueba de Levene (Evaluación 1)

6.2. Análisis Estadístico de la Evaluación 2

De manera similar a la sección 6.1, en esta sección se presenta el análisis estadístico descriptivo e inferencial de las mediciones generadas por el evaluador 2. La Tabla 9 presenta el resumen estadístico de la evaluación 2, podemos identificar que tanto la media como la mediana de los EC es mayor que la de los ET, aunque el rango en este caso es mayor que el de los ET, así mismo los EC presentan mayor variabilidad en los datos.

Factor	#	μ	Mediana	σ^2	Min	Max	Rango
EC	3	113.0	107.0	127.0	106.0	126.0	20.0
ET	4	94.5	92.0	57.0	89.0	105.0	16.0

Tabla 9 – Resumen Estadístico de la Evaluación 2

En la Figura 4 podemos ver que los conjuntos de datos se encuentran desfasados, por lo que es muy probable que presenten diferencias significativas entre ellos. También podemos observar que tanto los EC como los ET presentan asimetría a su derecha, aunque es ligeramente menor la de los ET.

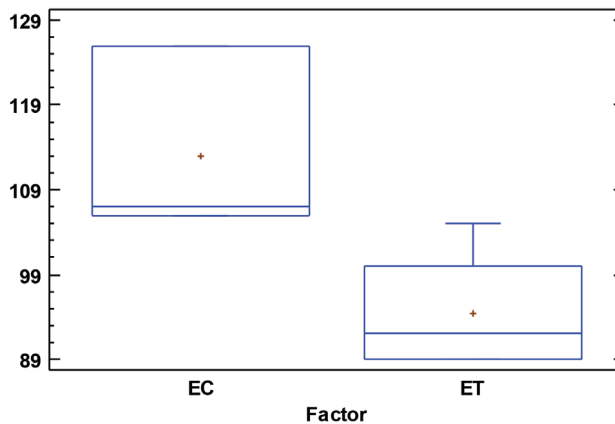


Figura 4 – Diagrama de Caja y bigotes para los tratamientos de la Métrica Calidad del DERS (Evaluación 2)

Para evaluar si las diferencias observadas previamente resultan significativas, se plantearon las siguientes hipótesis estadísticas en torno a la Calidad del DERS.

$$H_{0_{C-DERS}}: \mu_{EC} = \mu_{ET} ; H_{1_{C-DERS}}: \mu_{EC} <> \mu_{ET} \quad (II)$$

El resultado de evaluar con ANOVA dichas hipótesis estadísticas, se ilustra en la tabla 10.

Métrica	F	p-valor
C-DERS	6.90	0.0467

*p-valor < 0.05

Tabla 10 – ANOVA para la variable Calidad del DERS (Evaluación 2)

Dado que el valor p de la prueba F —en la variable Calidad del DERS— es menor que 0.05 (0.0467) la hipótesis de nulidad puede ser rechazada, es decir, la hipótesis alternativa permite afirmar que existe diferencias entre las medias de los dos tratamientos incluso con un 5% de significancia. Al igual que con la evaluación 1, procedimos a evaluar los supuestos del modelo ANOVA en forma similar a la sección previa.

En el caso del primer supuesto (normalidad) el gráfico de probabilidad normal (ver figura 5a) nos indica que los residuos se presentan dentro del intervalo de confianza de la diagonal, por lo que es posible asumir que los residuos presentan una distribución normal.

En cuanto al supuesto de independencia de los datos, en el gráfico de residuales versus secuencia (figura 5b) no se identifica tendencia alguna en los datos, por lo que es posible suponer que los datos provienen de poblaciones independientes.

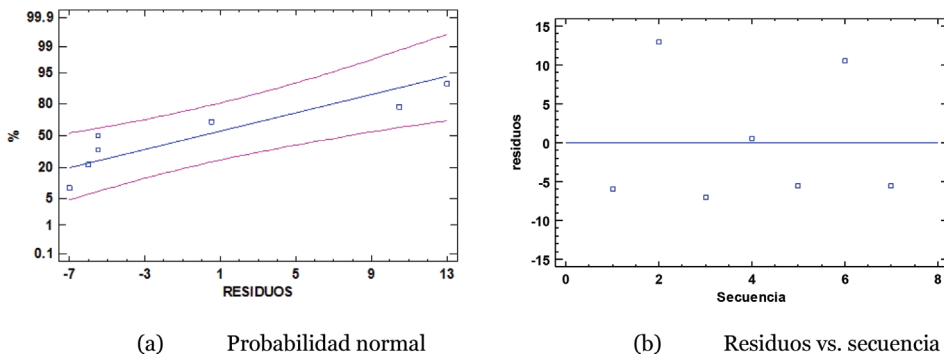


Figura 5 – Gráficos generados con los residuos (Evaluación 2)

Por su parte, con la prueba de Levene podemos concluir que no existe una diferencia estadísticamente significativa entre las desviaciones estándar de ambas poblaciones — con un nivel del 95.0% de confianza— lo anterior debido a que el p valor es mayor que 0.05 (ver tabla 11).

Prueba	p-valor
<i>1.09262</i>	<i>0.3438</i>

Tabla 11 – prueba de Levene (Evaluación 2)

7. Conclusiones y Trabajos en Curso

El proceso de requisitos busca satisfacer las necesidades de las partes interesadas mediante la especificación de las funcionalidades de una aplicación software, para lo cual el Ingeniero de Software interactúa a través de las técnicas de obtención con las partes interesadas de manera individual o grupal, y el proceso de análisis y especificación de requisitos se realiza con varias iteraciones.

En este artículo presentamos un experimento controlado en el que comparamos la calidad de un producto generado por los equipos de trabajo en tareas relacionadas con el desarrollo de software, particularmente, con la tarea del desarrollo de requisitos software. Los tratamientos a comparar se vincularon con la forma de integrar los equipos de trabajo; en primer lugar, la propuesta de integrar equipos utilizando la teoría de roles de Belbin (Equipos compatibles: RC) y en segundo lugar, la forma tradicional de asignar aleatoriamente a sus miembros (equipos tradicionales: ET).

Para llevar a cabo el experimento controlado, utilizamos un diseño de comparación de medias, y para el análisis estadístico de los datos, utilizamos el modelo ANOVA de una vía, el cual sirve para comparar dos medias de dos grupos independientes (no relacionados) que utilizan la distribución F. Con el propósito de obtener una doble evaluación, el experimento incorporó la evaluación de los productos por parte de dos expertos en requisitos y se procedió al análisis independiente de los datos. Los resultados sobre la calidad del DERS mostraron diferencias significativas entre los tratamientos, con mejores resultados para los EC; en el caso de la evaluación 1 las diferencias pueden ser explicadas con un 10% de significancia, y en la evaluación 2 con un 0.05 %.

Los obtenidos en este experimento coinciden con lo encontrado y descrito en (Aguileta, Ucán y Aguilar, 2017), en ambos casos el producto fue desarrollado a lo largo de un período aproximado de tres meses, plazo en el que posiblemente el equipo logro evolucionar de un simple grupo a un equipo efectivo de trabajo; no obstante, el resultado contrasta con lo reportado en (Aguilar, Díaz y Ucán, 2019), y es que posiblemente la tarea de medir el software, no requiere el desarrollo de altos niveles de interacción entre los miembros del equipo, tampoco un período largo de trabajo, pero sobre todo, es probable que deba ser realizada en forma individual.

Es importante mencionar que cuando usamos la experimentación como una metodología empírica, dos de las circunstancias que encontramos son las relacionadas con: (1) el tamaño de muestra no grande, (2) la necesidad de un muestreo conveniente; Sin embargo, lo ideal en las estadísticas para tener una muestra representativa de la población sería: (1) usar un tipo de muestreo apropiado y (2) seleccionar un tamaño de muestra representativo, es prácticamente imposible realizarlo en nuestro campo. Por otro lado, el desarrollo de una serie de réplicas experimentales, o en su caso, la obtención de datos para la variable dependiente por procesos alternos, como fue nuestro caso de utilizar la validación por parte de más de un experto, nos permite corroborar las hipótesis de investigación y así obtener un conocimiento más específico sobre el fenómeno.

Como parte de los trabajos en curso, el grupo de investigación ha realizado experimentos controlados en tareas como el Diseño de Bases de Datos, Desarrollo de Aplicaciones Web, y solo nos resta por realizar la validación de los productos, obtención de las métricas correspondiente y el análisis estadístico de los datos.

Referencias

- Aguilar, R., Oktaba, H., Juarez, R., Aguilar, J., Fernández, C., Rodríguez, O., & Ucán, J. (2017). Ingeniería de Software. En: Pineda, L. (ed.) *La computación en México por especialidades académicas*. Academia Mexicana de Computación, Capítulo V. Mexico: Academia Mexicana de Computación.

- Aguilar, R., Díaz, J. & Ucán, J. (2019). Influencia de la Teoría de Roles de Belbin en la Medición de Software: Un estudio exploratorio. *RISTI - Revista Ibérica de Sistemas y Tecnologías de la Información*, (31), 50-65.
- Aguileta, A. Ucán, J. & Aguilar, R. (2017). Explorando la influencia de los roles de Belbin en la calidad del código generado por estudiantes en un curso de ingeniería de software. *Revista Educación en Ingeniería*, 12(23), 93-100.
- Aritzeta, A., Swailes, S., & Senior, B. (2005). *Team roles: Psychometric evidence, construct validity and team building*. Hull, UK: University Hull.
- Basili, V., Selby, R., & Hutchens, D. (1986). Experimentation in Software Engineering. *IEEE Transactions on Software Engineering*, 12(7), 733-743.
- Belbin, M. (1981). *Management teams: Why they succeed or fail*. New York, USA: John Wiley & Sons.
- Belbin, R.M. (1993). *Team roles at Work*. Oxford: Elsevier Butterworth Heinemann.
- Bourque, P., & Fairley, R. (2014). *Guide to the Software Engineering Body of Knowledge (SWEBOOK V3.0)*. IEEE Computer Society.
- Briggs-Myers, I., & McCaulley, M. (1985). *Manual, A Guide to the Development and Use of the Myers-Briggs Type Indicator*. Palo Alto, USA: Consulting Psychologists Press.
- Cuevas, G. (2002). *Gestión del Proceso Software*. Madrid, España: Centro de Estudios Ramón Areces.
- De Marco, T., & Lister, T. (1999). *Peopleware Productive Projects and Teams*. 2nd Ed. New York, USA: Dorset House Publishing Co.
- Estrada, E., & Peña, A. (2013). Influencia de los roles de equipo en las actividades del desarrollador de software. *Revista Electrónica de Computación, Informática, Biomédica y Electrónica*, 2(1), 1-19.
- Genero, M., Cruz-Lemus, J., & Piattini, M. (2014). *Métodos de investigación en ingeniería de software*. Madrid, España: Ra-Ma.
- Glass, R., Vessey, I., & Ramesh, V. (2002). Research in Software Engineering: An analysis of the literature. *Information and Software Technology*, 44(8), 491-506.
- Gutiérrez, H., & De la Vara, R. (2012). *Análisis y Diseño de Experimentos*. 3ª Ed. Ciudad de México: México. McGraw Hill.
- IEEE (1990). *IEEE Standard Glossary of Software Engineering Terminology* (Std. 610.12-1990). IEE.
- IEEE CS-SES (1998). *IEEE Recommended Practice for Software Requirements Specifications*. IEEE Std 830-1998. IEE.
- Henry, S., & Stevens, K. (1999). Using Belbin's leadership role to improve team effectiveness: An empirical investigation. *The Journal of Systems and Software*, 44, 241-250.

- Juristo, N., & Moreno, A. (2001). *Basics of Software Engineering Experimentation*. Boston, USA: Kluwer Academic Publishers.
- Kitchenham, B., & Pfleeger, S. (1996). *Software quality: the elusive target*. IEEE Software, 13(1), pp. 12-21.
- Margerison, C.J., & McCann, D.J. (1985). Team Management Profiles: Their use in Managerial Development. *Journal of Management Development*, 4(2), 34-37. DOI: 10.1108/eb051580
- Morales, N., & Vega, V. (2018). Factores Humanos y la Mejora de Procesos de Software: Propuesta inicial de un catálogo que guíe su gestión. *RISTI - Revista Ibérica de Sistemas y Tecnologías de Información*, (29), 30-42.
- Mumma, F.S. (2005). *Team-work & team-roles: what makes your team tick?: facilitator guide*. King of Prussia, PA: HRDQ Press.
- Pollock, M. (2009). Investigating the relationship between team role diversity and team performance in information systems teams. *Journal of Information Technology Management*, 20(1), 42-55.
- Senior, B. (1997). Team roles & Team performance: Is there 'really' a link?. *Journal of Occupational and Organizational Psychology*, (70), 85-94.
- Yousuf, M., Asger, M. (2015). Comparison of various requirements elicitation techniques. *Int. J. Comput. Appl*, 116(4): 0.