

Características del desarrollo de software en la ciudad de Corrientes

Gladys Dapozo; Cristina Greiner, Emanuel Irrazábal, Yanina Medina, María de los
Ángeles Ferraro, Berenice Lencina
Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura
Universidad Nacional del Nordeste, Av.Libertad 5450, 3400, Corrientes, Corrientes, Argentina
{gndapozo,cgreiner, yanina}@exa.unne.edu.ar, emanuelirrazabal@gmail.com;
mafferraro@hotmail.com

Resumen. En este trabajo se presentan los resultados de una encuesta orientada a determinar el estado de situación del desarrollo de software en la ciudad de Corrientes, Argentina, considerando las empresas específicas de software y las áreas de sistemas de las principales organizaciones, de dependencia estatal o privada, que desarrollan software para contribuir al logro de los objetivos de las mismas. En general, se advierte que las áreas/empresas no aprovechan los métodos y herramientas más actuales para lograr gestionar los proyectos de desarrollo con mayor eficiencia y lograr productos de mayor calidad. Estos resultados permitirán direccionar actividades de divulgación o capacitación sobre estos temas, a través de las universidades o polos tecnológicos de la región.

Palabras clave: Calidad de software. Métodos y herramientas. Gestión de proyectos.

1. Introducción

A partir del año 2002, en la Argentina, las políticas públicas se orientaron a la promoción de la industria del software a lo largo y ancho del territorio.

En particular, en la provincia de Corrientes, en el año 2007 se lanza el Plan Estratégico para la Industria del Software y Servicios Informáticos (PEISSI), cuyo objetivo es fortalecer a este sector de la economía por considerarlo clave para el desarrollo tecnológico del sistema productivo en su conjunto. Se apunta específicamente a proveer asistencia a las empresas del rubro de la provincia a través de las herramientas que dispone el estado para potenciarlas. Con este impulso, nace el Polo IT Corrientes. Las empresas nucleadas en el Polo IT Corrientes, mediante la asociatividad, buscan lograr el crecimiento individual y conjunto, la transferencia de conocimiento, investigación y desarrollo, la búsqueda de alianzas estratégicas con organismos y universidades y la mejora continua. Entre sus logros sobresalientes se encuentra la adhesión a la CESSI, la certificación de calidad bajo la norma ISO 9001:2000 de las empresas integrantes del Polo, mediante el financiamiento obtenido a través del estado provincial [1].

Un primer estudio, realizado en el año 2009, caracterizó las pymes de software de la región NEA con el propósito de definir un marco de referencia para el desarrollo de acciones enfocadas en la mejora de la calidad del proceso y/o de la calidad de los productos de software que las empresas producen. El relevamiento realizado en las provincias de Chaco, Corrientes y Misiones, reflejó que el 96% de las empresas tenían

menos de 10 empleados. De la totalidad considerada ninguna había obtenido una certificación de calidad aunque manifestaban la importancia de contar con ella [2].

Dado el avance de la Ingeniería de Software, en general, así como los métodos y herramientas orientados a mejorar la calidad del software, resulta interesante revisar y analizar cómo trabajan las empresas de software en la ciudad de Corrientes, incluyendo en esta observación a los departamentos o áreas de sistemas cuya función es desarrollar el software que se utiliza en las principales organizaciones gubernamentales y empresas de servicios, que en la zona son las principales demandantes de los recursos humanos formados en las carreras de Informática.

Para este estudio se definieron las siguientes áreas de interés: Conformación de los equipos de trabajo en cuanto a cantidad, roles que desempeñan y formación académica. En lo que respecta a la Ingeniería de software y los aspectos vinculados con la calidad, se decidió observar lo referido a las metodologías de desarrollo, métodos de estimación, gestión cuantitativa, medición de software y testing.

A continuación se brindan los principales conceptos que enmarcan los distintos aspectos analizados.

1.1. Metodologías de desarrollo

A lo largo del tiempo en la ingeniería del software se han sucedido una gran cantidad de metodologías y técnicas de desarrollo software [3]. Las metodologías han pasado de estar directamente relacionadas con la predictibilidad y rigidez, a hacerse cada vez menos prescriptivas. Actualmente las metodologías o técnicas ágiles, como SCRUM, XP, KANBAN, son ampliamente utilizadas en las empresas de desarrollo software y las áreas de sistema. Esto se relaciona con la posibilidad de gestionar los cambios a lo largo del proyecto, con mantenimientos perfectivos en lugar de realizar aplicaciones “desechables” [4]. En este sentido diferentes estudios coinciden en que el uso de las metodologías ágiles está en torno a un 50%, tanto en el hemisferio norte [5] como en Argentina [6].

Por otra parte, el conocimiento de las diferentes metodologías y herramientas no asegura su utilización. Son bien conocidos los estudios donde se relaciona lo que el profesional ha conocido en su vida académica y profesional, con lo que utiliza en el entorno laboral [7], por tanto es necesario preguntar por ambos aspectos: lo que conoce y lo que utiliza.

1.2. Estimación

Según [8] “una estimación es una predicción de cuánto tiempo durará o costará un proyecto”, constituye la base para la planificación de los proyectos. El desarrollo del software requiere de la estimación para controlar y administrar los recursos que se necesitan utilizar antes y durante el proyecto. Son numerosas las variables, relacionadas con los recursos humanos, el contexto y las políticas que intervienen en el proceso de desarrollo, que pueden afectar los resultados finales.

Para las empresas de software una estimación deficiente del esfuerzo y duración que conlleva un proyecto puede ocasionar incumplimiento de plazos, entrega de productos incompletos y pérdida de competitividad. En contextos de desarrollo de proyectos de pequeña o mediana envergadura resulta que a mayor especificidad en cuanto al tipo de proyecto y una mayor disponibilidad y pertinencia de datos

históricos, los valores de esfuerzo y duración estimados se aproximan más a los valores reales [9].

1.3. Gestión cuantitativa

La gestión cuantitativa de procesos [10] proporciona una visión del grado de cumplimiento de metas así como de las causas que explican desviaciones significativas en procesos o productos. El propósito de esta gestión es dirigir un proyecto u organización basado en un conocimiento cuantitativo, es decir medible, determinable, de los aspectos de mayor relevancia, que generalmente son procesos cuyo rendimiento afecta en forma significativa al logro de los objetivos del proyecto y la satisfacción de los clientes [11].

La mejora de procesos basada en medición promueve la gestión cuantitativa de proyectos de software, mediante el seguimiento continuo de procesos y productos, con el fin de predecir su comportamiento y detectar desviaciones durante su ejecución. Las mediciones, cuando son analizadas, constituyen una base importante para una gestión efectiva por parte del equipo de desarrollo [11], [12].

1.4. Medición de atributos de calidad del software

La medición de los atributos de calidad del software es una necesidad para las empresas de software y servicios informáticos (SSI), dado que permite mejorar las tareas menos eficientes y obtener productos de calidad. Esto se traduce en una ventaja competitiva para las organizaciones, ya que el conocimiento adquirido a través de la medición permite modificar aquellos factores que aportan una mayor eficacia en el proceso productivo y contribuyen a la calidad del producto [13].

La calidad del software es una compleja combinación de factores. Diversos autores [14] [15], y estándares tales como el ISO 9126 [16] han tratado de determinar y categorizar los factores que afectan a la calidad del software. Para garantizar la calidad del software, se requiere medir los atributos que la definen. El análisis de estas mediciones durante el proceso de desarrollo proporciona al ingeniero de software una aproximación en tiempo real de la eficacia de los modelos de análisis, diseño y código, y de la calidad general del software que se construye.

La medición del producto se realiza mediante las métricas, para caracterizar numéricamente los distintos aspectos del desarrollo del software. Michael Mah [17] define las métricas de software como “La aplicación continua de mediciones basadas en técnicas para el proceso de desarrollo del software y sus productos para suministrar información relevante a tiempo, así el administrador junto con el empleo de estas técnicas mejorará el proceso y sus productos”. Los datos recogidos durante el proceso de medición se deben conservar en registros históricos, y mantener como un recurso organizacional [18].

1.5. Testing

Los fallos en el software, tanto de funcionalidad como de seguridad, continúan presentándose, y se considera que en parte el problema se debe a que la complejidad del software se incrementa en forma permanente y no se consigue controlarla, a pesar de las nuevas técnicas y metodologías que van surgiendo. La complejidad del software juega un importante papel en la presencia de errores: a mayor complejidad, mayor número de errores. Los desarrolladores dedican un 80 por ciento de los costos

de desarrollo a la identificación y corrección de errores. La planificación y diseño de pruebas en las primeras fases de desarrollo permiten encontrar errores, omisiones, inconsistencias e, incluso, especificaciones redundantes en los requisitos funcionales cuando aún es fácil y económico corregirlas ya que el costo de eliminar defectos aumenta a medida que aumenta el tiempo que transcurre entre la aparición del defecto y su detección [19]. Para que el proceso de prueba del sistema sea eficaz debe estar integrado dentro del propio proceso de desarrollo y, como cualquier otra fase del desarrollo, debe realizarse de manera sistemática [20].

2. Metodología

Para relevar la situación respecto de la actividad de desarrollo de software de las empresas de software y áreas de Sistemas de las organizaciones y empresas de la ciudad de Corrientes, se diseñó un cuestionario para recabar información, enfocado en los puntos de interés de la investigación: Características generales, del desarrollo, estimación, gestión cuantitativa, medición de producto software y testing.

El cuestionario se envió por correo electrónico a los gerentes y responsables de desarrollo de software de las distintas organizaciones preseleccionadas. Una vez recibidas las encuestas, se realizaron entrevistas (presenciales o telefónicas) a fin de verificar las respuestas, aclarar o ampliar los puntos tratados.

A continuación, los datos fueron procesados mediante una planilla de cálculo.

3. Resultados

A continuación se describen los resultados de la encuesta en cuanto a las distintas variables seleccionadas para su análisis:

Características generales:

Se relevaron datos de 21 empresas/áreas de sistemas que desarrollan software en la ciudad de Corrientes, de las cuales 14 (67%) corresponden a áreas de sistemas y 7 (33%) son empresas de desarrollo de software, exclusivamente. Las áreas de sistemas son en su mayoría (79%) de gestión estatal nacional, provincial o municipal. Entre estas se cuentan: Banco de Corrientes, Dirección General de Rentas, Instituto de Viviendas, Dirección Provincial de Energía, Poder Judicial de Corrientes, Ministerio de Salud, Municipalidad de Corrientes, Rectorado UNNE, Obra Social UNNE, Centro de Cómputos de Corrientes y Hospital Escuela de Corrientes.

Son 186 personas que desarrollan software o realizan tareas vinculadas. El 76% corresponde a personas que se desempeñan en áreas de sistemas (ver Tabla 1).

Tabla 1: Cantidad de personas

Áreas/Empresas	Privada	Pública	Emp Soft	Total	%
Área Sistemas	28	113		141	76%
Empresa software			45	45	24%
				186	100%

En cuanto a los roles declarados en la actividad, en la Figura 1 se puede observar una preponderancia del rol de Analista-Programador (37%), seguida por la de

Programador (14%) y Tester (13%). Se destaca que los equipos comienzan a incorporar el rol de Diseñador Gráfico. Cabe aclarar que en la mayoría de los casos los roles no son exclusivos, es decir, habrá más roles que personas.

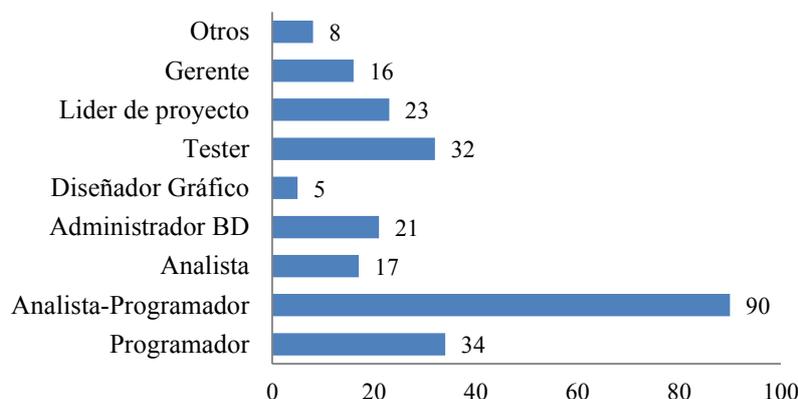


Fig.1: Roles desempeñados en los equipos de trabajo

En cuanto a la formación académica puede observarse en la Tabla 2 que mayoritariamente (63%) los trabajadores poseen título de grado. Se observa también que los estudiantes activos (se encuentran cursando una carrera de informática) constituyen un 18%. Los estudiantes inactivos son aquellos que en la actualidad no se encuentran cursando regularmente una carrera, constituyen el 7%. El más bajo porcentaje corresponde a los Idóneos (5%), justificado en la región porque existen variadas ofertas de carreras de Informática en universidades públicas y privadas. Respecto de la cuestión de género, se puede observar una fuerte presencia masculina en los equipos de trabajo (74%).

Tabla 2: Formación académica

Formación	Varones	Mujeres	Total	%
Título de grado	73	32	105	63%
Título intermedio o superior	8	4	12	7%
Estudiante activo	25	5	30	18%
Estudiante inactivo	10	2	12	7%
Idóneos	9	0	9	5%
Totales	125	43	168	100%
	74%	26%	100%	

Dentro de los títulos académicos habilitantes para el desempeño profesional se observa una mayor presencia de Licenciados en Sistemas de Información (50%), dado que esta carrera se dicta en la ciudad de Corrientes. Continúan los Ingenieros en Sistemas de Información (27%) de UTN Regional Resistencia, oferta distante a 25 km de la ciudad de Corrientes. Ver Figura 2.

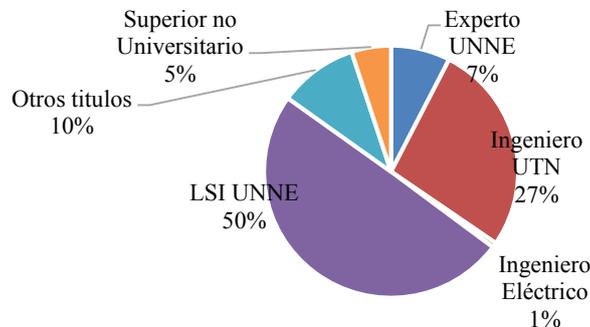


Fig.2: Titulaciones de grado y pregrado

Características de los proyectos de desarrollo software de la región.

En la Fig.3 se puede ver un resumen de los tipos de proyecto llevados a cabo en la región. Por un lado, la mayor cantidad de proyectos es “web” seguido por el tipo de “escritorio”. Los proyectos móviles son los de menor porcentaje con un 26%. Todavía se percibe un sesgo en cuanto a proyectos de tipología exclusiva. Finalmente, como un dato a destacar, es que otro tipo de proyectos, por ejemplo relacionados con firmware, ocupan en promedio un 30% de los desarrollos en las empresas.

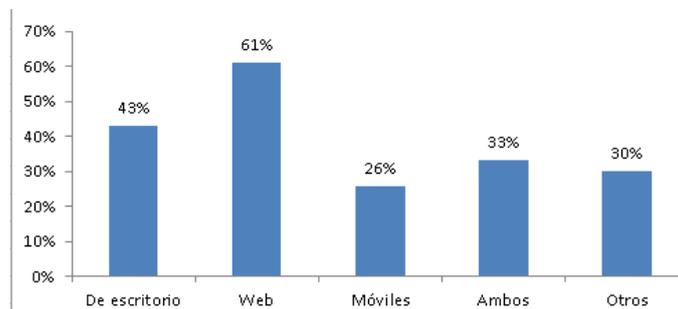


Fig.3. Tipos de desarrollo

Los proyectos que desarrollan un nuevo sistema respecto de los proyectos de mantenimiento, representan un (43%). Esto se da especialmente en la administración pública.

En la Fig. se detallan las metodologías y técnicas de desarrollo, utilizadas por las empresas y áreas de sistema encuestadas. Para cada metodología se preguntó si la conocía (barra de la derecha), si trabajó con ella en los últimos 3 años (barra central) y si trabaja actualmente (barra de la izquierda). Como puede verse en el gráfico, continúa siendo la metodología en cascada la más conocida y utilizada. Las metodologías ágiles están siendo cada vez más conocidas, pero utilizadas en un porcentaje mucho menor. Cerca de un 20% de las empresas que conocen las metodologías ágiles la utilizan. Asimismo, no se utilizan metodologías propias de proyectos web, sino solamente métodos y técnicas genéricas.

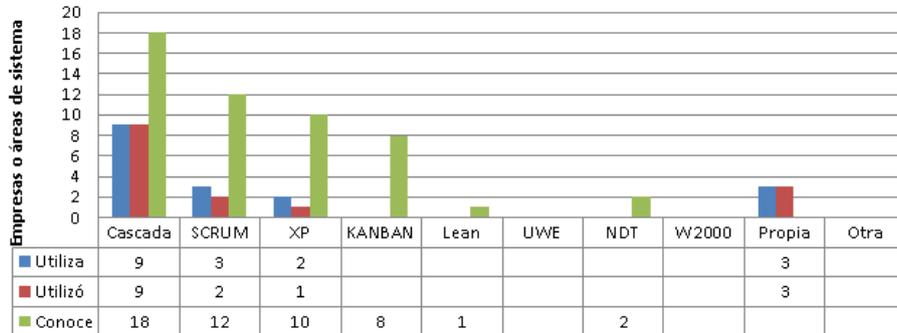


Fig.4. Metodologías de desarrollo

3.1. Estimación

Los resultados indican que el 52% de las empresas/áreas realizan una estimación para el 100% de los proyectos desarrollados. A su vez, cuando se realiza estimación, esta tarea está a cargo de grupo de expertos (40%) o de una sola persona (40%), como se observa en la Figura 5.

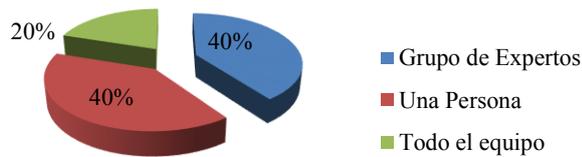


Fig. 5. Responsables de la estimación

El interés que tienen las empresas en la estimación recae, principalmente, en la Duración y Costo, y en el Tamaño-Esfuerzo y Duración, como se advierte en la Figura 6.

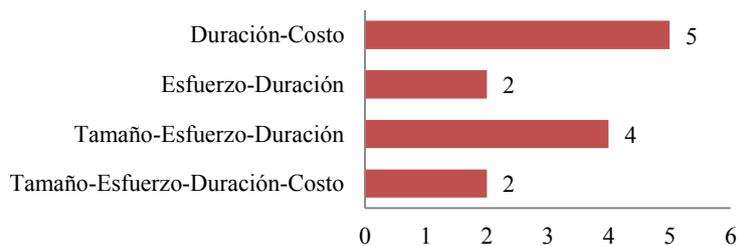


Fig. 6. Atributos que estiman las empresas de los proyectos

Respecto de las técnicas utilizadas para la estimación se puede observar, en la Figura 7, que la mayoría utiliza las técnicas convencionales de juicio de experto, por analogía y por consenso. Se observa que las técnicas paramétricas, como Puntos de Función o COCOMO, son conocidas pero poco utilizadas.

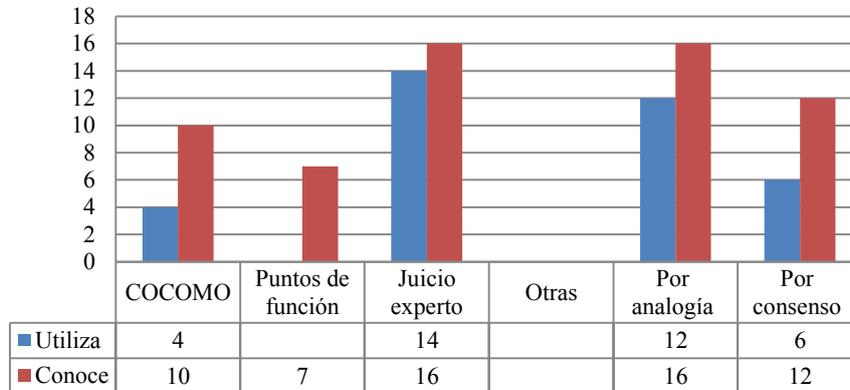


Fig. 7. Técnicas de estimación

La técnica que se utiliza depende de los datos disponibles y de la naturaleza del problema bajo estimación. Respecto al origen de los datos disponibles para el momento de estimar se pudo observar que la gran mayoría (61%) no utiliza datos históricos al momento de realizar la estimación de un proyecto, aspecto que se torna importante para obtener resultados más precisos.

3.2. Gestión cuantitativa

Se sabe que la medición se emplea para establecer una línea base del proceso, a partir de la cuáles evalúan las mejoras. Esta línea base se constituye con datos recopilados en proyectos previos que contienen medidas de proyectos y métricas derivadas de estos. De las empresas encuestadas, un 48 % guardan datos históricos de los proyectos, tales como fecha de inicio, fin, recursos, costos. Sin embargo esta información, como se ha visto en el apartado anterior, no se utiliza para la estimación.

De las herramientas que se utilizan para la gestión de los proyectos puede observarse en la Figura 8 que las más usadas son Project, Excel y Jira, en versión propietaria. En tanto que en versiones libres, se indicó el uso de RedMine y GitHub. Se indicaron otras soluciones, por ejemplo: Planillas de Google Drive, SharePoint, Rational Clear Quest, Team Foundation Server (TFS) y herramientas propias.

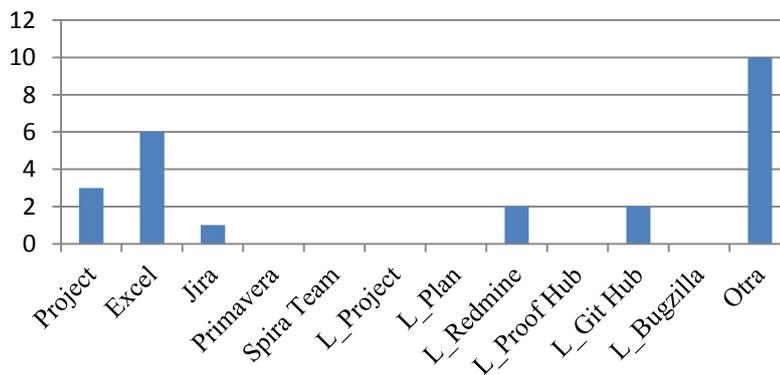


Fig. 8. Herramientas de gestión cuantitativa

Entre los aspectos en los que contribuye la herramienta utilizadas, se señalaron (ver Figura 9): Planificación, Colaboración (comunicación con el equipo) y Estimación. En menor medida contribuyen a la gestión de incidencias, versionado y configuración.

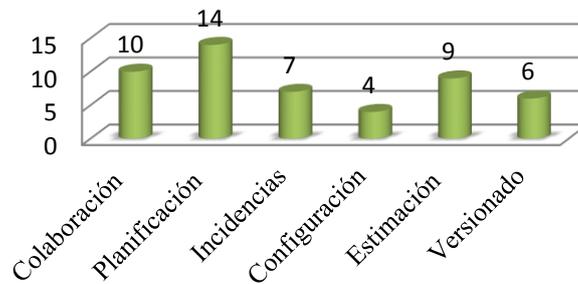


Fig. 9. Aspectos en los que contribuyen las herramientas de gestión

3.3. Medición de atributos de calidad del software

Del total de empresas y áreas de sistemas encuestadas, ninguna de ellas realiza mediciones de atributos de calidad del software.

Entre las razones expuestas por los entrevistados para esta situación, como puede observarse en la Figura 10, destacan el desconocimiento sobre metodologías, herramientas y métricas (38%), seguido por la falta de tiempo y RRHH (28%), y el considerar muy costosa esta actividad (19%). En menor medida, señalaron que la información referida a mediciones no es usada, pudiendo considerarse que a nivel gerencial se le resta valor, y en otro caso señalaron que no resulta útil.

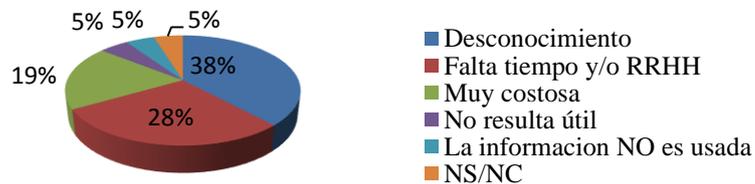


Fig. 10. Razones para no medir atributos de calidad del software

3.4. Testing

Tanto en las empresas de desarrollo como en las áreas de sistemas los recursos humanos en general no tienen un rol exclusivo, sino que desarrollan distintas tareas. Esta característica se repite en lo relacionado a la tarea de testing o prueba de software, ya que sólo en el 10% de los casos cuentan con un equipo exclusivo para testing, siendo 1 caso para empresa y 1 caso para área de sistemas.

De manera similar, solo el 10% de empresas realiza testing en forma automática, siendo en este caso en 2 empresas dedicadas al desarrollo de software.

En cuanto al tipo de pruebas que se realizan, tal como se observa en la Figura 11, la mayoría de los encuestados (58%) señala que realizan prueba de aceptación, siendo la prueba de integración menos utilizada.

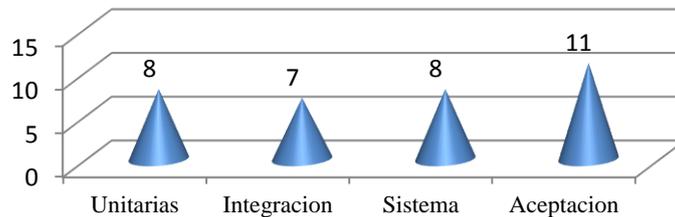


Fig. 11: Tipos de pruebas

Finalmente, sólo el 19% manifiesta utilizar herramientas para realizar test, siendo las nombradas Selenium, JUnit, utPLSQL, Rspec y Capybara.

4. Conclusiones

De los resultados obtenidos, en cuanto a los tipos de proyectos se nota un fuerte avance de los proyectos web pero falta una mayor apuesta por los desarrollos móviles.

Las metodologías ágiles están siendo cada vez más conocidas, pero utilizadas en un porcentaje mucho menor. Sólo un 20% de las empresas que conocen las metodologías ágiles la utilizan.

En la mayoría de las áreas/empresas, las prácticas de estimación más utilizadas son la de juicio de experto y por analogía. Se conocen pero no se utilizan, o se utilizan poco, las técnicas paramétricas que pueden aportar mayor precisión en la estimación. Así también es bajo el porcentaje de las áreas/empresas que utilizan datos históricos para la estimación.

Se destaca también que las organizaciones encuestadas no miden atributos de calidad, y esto se debe en gran medida a que no conocen los métodos y herramientas específicas, y también manifiestan que no pueden dedicarle tiempo ni recursos humanos a esta práctica, desconociendo los beneficios que estas prácticas pueden aportar al desarrollo.

Respecto del testing no existen en general recursos humanos dedicados en forma exclusiva, y en un alto porcentaje se realiza en forma manual, siendo el test de aceptación el más realizado.

Se concluye que las áreas o empresas, en gran medida, desconocen métodos y herramientas que pueden contribuir a mejorar la calidad y la gestión del software que desarrollan, siendo necesario realizar desde las universidades o los polos tecnológicos actividades de divulgación o capacitación sobre estos temas.

5. Referencias

[1] Cuenca Pletsch, L.;Dapozo, G.;Greiner, C.; Estayno, M. “Vinculación Universidad-Empresa orientada a la promoción de la industria del software. Una experiencia de colaboración en la región NEA”. Revista del Núcleo de Estudios e Investigaciones en Educación Superior del MERCOSUR. ISSN 2313-9080. Vol. 1 (2012). Pp. 36-41. http://nemocosur.siu.edu.ar/publicaciones/Numero_1.pdf

[2] Estayno, M.; Dapozo G., Cuenca Pletsch, L.; Greiner C., Pelozo S., “Caracterización de las pymes de software de la región NEA orientada hacia un marco de mejora de la calidad”. Anales del XV Congreso Argentino de Ciencias de la Computación. (CACIC2009). Facultad de Ingeniería. Universidad Nacional de Jujuy. 5 al 9 octubre de 2009. ISBN 978-897-24068-4-1.

- [3] Bohem, Barry. A view of 20th and 21st century software engineering. EnProceedings of the 28th internationalconferenceon Software engineering. ACM, 2006. p. 12-29.
- [4] Fowler, Martin; Highsmith, Jim. The agile manifesto. Software Development, 2001, vol. 9, no 8, p. 28-35.
- [5] VersionOne Inc., State of Agile 9th Survey, USA, 2014.
- [6] Alende, Andrea N., et al. La Utilización de los Métodos Ágiles en las Empresas de Desarrollo de Software de Argentina. 2010.
- [7] Lethbridge, Timothy C. What knowledge is important to a software professional? Computer, 2000, no 5, p. 44-50.
- [8] McConnell, S. (2006) "Software Estimation: Demystifying the Black Art" (Developer Best Practices). Microsoft.
- [9] Gladys N. Dapozo, Yanina Medina, Andrea B. Lencina, Gabriel O. Pedrozo Petrazzini. "Métodos de estimación de esfuerzo y duración en proyectos web pequeños". Revista Electrónica Argentina-Brasil Tecnologías de Información y Comunicación. Editorial Facultad de Trê de Maio (SETREM) Brasil. ISSN 2422-6424. 2015 Volumen 1 Tomo 1 N° 2
- [10]Gou, L., Wang, Q., Yuan, J., Yang, Y., Li, M., & Jiang, N. (2009). Quantitative defects management in iterative development with Bi Defect. Software Process Improvement and Practice, 14(4), 227-241
- [11] Baldassarre, T., Boffoli, N., Caivano, D., &Visaggio, G. (2004). Managing Software Process Improvement (SPI) through Statistical Process Control (SPC).Lecture Notes in Computer Science, 3009, 30-46
- [12] Ardila, C. & Pino, F. (2013). Panorama de gestión cuantitativa de procesos de desarrollo de software en pequeñas organizaciones. Revista S&T, 11(26), 29-46.
- [13] Hernández Ballesteros, J.F., Minguet Melián, J. M. "La Medida de la Calidad del Software como Necesidad y Exigencia en Modelos Internacionales (CMMI, ISO 15504, ISO 9001). www.issi.uned.es/CalidadSoftware/Noticias/PonIng2005.rtf
- [14] Pressman, R. S. Ingeniería de Software. Un enfoque práctico. Editorial MCGRAW-HILL.
- [15] McCall, J.A.; Richards, P.K.; Walters, G.F. - Factors in Software Quality. Vols I, II, III. NTIS AD-AO49-014, 015, 055, Nov. 1977
- [16] International Organisation for Standardisation ISO 9126: Software Engineering – Product quality, Geneva, Switzerland. 2001
- [17] Mah, Michael - High-definition software measurement - Software Development. Vol. 7 N° 5, Pages: 9-14 - ISSN: 1070-8588 - Mayo 1999.
- [18] Greiner, C.; Demchum, D.; Dapozo, G.; Estayno, M. "Una propuesta de solución para automatizar la medición de aplicaciones orientadas a objeto". CACIC 2010.
- [19] Jalote, P., Software Project Management in Practice, Addison Wesley, 2002.
- [20] Ponce, J., Escalona, M.J., Gómez, A., Luque, M., Molina, A. Definición de una política de pruebas en la gestión cultural: aplicación al desarrollo del proyecto Mosaico. REICIS. Revista Española de Innovación, Calidad e Ingeniería del Software. 2010. Vol.6, No. 2. ISSN: 1885-4486.