

# Propuesta tecnológica de Indra para afrontar los retos inmediatos de la Ingeniería del Software

Indra's technological proposal to tackle the immediate challenges of Software Engineering

Gabriel Sánchez Belmonte<sup>1</sup>, Carlos García Moreno<sup>1</sup>, Yolanda Hernández González<sup>1</sup>

<sup>1</sup> INDRA

gsanchez@indra.es, cgarciamo@indra.es, yhernandezg@indra.es

**RESUMEN.** Este artículo: muestra cómo las nuevas tecnologías que se están aplicando en diversos ámbitos lúdicos y productivos se pueden adaptar y aplicar para dar respuesta a los principales retos existentes en el sector de la Ingeniería del Software; expone las propuestas tecnológicas en las que Indra está trabajando para proporcionar el soporte tecnológico necesario para cubrir los distintos retos y oportunidades que se plantean en la Ingeniería de Software; y presenta la Suite MInd y sus principales características, la cual supone el elemento vertebrador de las propuestas tecnológicas planteadas.

Destacando que la Ingeniería del Software es un sector que necesita incorporar el dinamismo y agilidad tanto en la adopción efectiva de nuevas metodologías, como tecnologías y herramientas, para dar respuesta a los retos planteados en la actualidad y en el futuro inmediato, que pueden ser vistos como importantes oportunidades competitivas.

**ABSTRACT.** This paper: shows how new technologies that are being applied in various ludic and productive areas can be adapted and applied to respond to major challenges in the field of Software Engineering; exposes the technological proposals that Indra is working to provide necessary technical support to cover the various challenges and opportunities that arise in Software Engineering; and presents the Suite MInd and its main characteristics, which assumes the backbone of the raised technological proposals.

Stressing that Software Engineering is a sector that needs to incorporate the dynamism and agility in both the effective adoption of new methodologies and technologies and tools, to respond to the challenges today and in the near future, which can be seen as important competitive opportunities.

**PALABRAS CLAVE:** Ingeniería de Software, Retos de la Ingeniería de Software, Propuestas tecnológicas, Suite MInd, Big Data, Gamificación, Tecnologías colaborativas, Serious Games, DevOps.

**KEYWORDS:** Software Engineering, Challenges of Software Engineering, Technological proposals, Suite MInd, Big Data, Gamification, Collaborative Technologies, Serious Games, DevOps.

## 1. Introducción

El término Ingeniería del Software tiene más de sesenta años, una antigüedad más que suficiente para que esta disciplina hubiera alcanzado una madurez y grado de fiabilidad muy alto. Sin embargo, no parece ser así, siendo frecuentes todavía las discusiones sobre las estimaciones o la calidad del software entregado, muchos proyectos siguen teniendo retrasos, sobrecostes, etc.

La tecnología ha evolucionado de forma muy rápida en los últimos años. En la actualidad el cambio es todavía más rápido y en el futuro todo parece indicar que la tecnología seguirá evolucionando de forma cada vez más rápida. Se trata de una evolución exponencial, por lo que reputados expertos predicen que en unos años muchos de nosotros no podremos ni siquiera asumir o entender los cambios que se produzcan. La Ingeniería del Software continúa avanzando de forma más o menos continua, aparecen nuevas metodologías, técnicas y herramientas pero, ¿avanza al mismo ritmo que el entorno tecnológico global? Si comparamos estos avances (tecnológicos y metodológicos) en los últimos veinte años con un caso paradigmático de evolución tecnológica, como es la telefonía móvil, la respuesta parece clara. Se podría decir que, a diferencia del entorno tecnológico global, la Ingeniería del Software evoluciona de forma aritmética. Partiendo de esta situación nos preguntamos si esta evolución es lo suficientemente rápida, y más aún, si las organizaciones están demostrando ser capaces de incorporar los cambios producidos. La respuesta es obvia, no.

Un análisis de esta situación nos induce a hacernos otras preguntas: ¿Por qué el sector de la Ingeniería del Software no está a la cabeza en la adopción de las novedades tecnológicas que sí se incorporan en otros campos? ¿Qué ventajas competitivas supondrían para las empresas incorporar estas tecnologías para dar respuesta a los nuevos (y en algunos casos no tanto) retos en el sector? ¿Qué futuro tienen las grandes organizaciones que no son capaces de adoptar estos cambios?

En este artículo vamos a dar nuestra visión sobre las oportunidades que ofrecen las tecnologías que consideramos deberían guiar el futuro de la Ingeniería del Software en base a los retos a los que se enfrenta, y cómo estamos afrontándolos en Indra a través de la innovación tecnológica. Queremos destacar que el objetivo del artículo no es abordar todas las fases del ciclo de vida de Ingeniería del Software, sino hacer hincapié y profundizar en los aspectos detectados que suponen los principales retos y oportunidades, que son en los que se basa nuestra propuesta tecnológica actual para la evolución de la Ingeniería del Software.

## 2. Retos inmediatos en Ingeniería del Software

Los principales retos a los que se enfrenta la Ingeniería del Software no son nuevos, de hecho podríamos decir que algunos parecen asimilados con resignación por parte de las empresas, e incluso a veces de los clientes, al no encontrarse una respuesta realmente eficaz a los mismos.

Entre los retos “clásicos” encontramos los relacionados con la calidad y la productividad: sobrecostes, errores, funcionalidad no (o incorrectamente) implementada, retrasos, etc. Estos retos se pueden resumir en “insatisfacción del cliente”. Los retos “modernos” a los que se enfrenta el sector desde hace más de una década vienen de la mano del fenómeno de la globalización, desde dos puntos de vista distintos. En primer lugar en lo que respecta a la competencia que suponen los países emergentes y, en segundo, respecto a lo que la globalización tecnológica supone.

Desde nuestro punto de vista todos estos retos suponen en realidad oportunidades. Las empresas que puedan por fin dar respuesta a los retos de siempre y saquen partido de las oportunidades que ofrece el panorama cambiante tecnológico y socioeconómico tendrán una clara ventaja competitiva y, nos atrevemos a aventurar que, serán las únicas que sobrevivan, en un sector que debería ser cada vez más sostenible.

En relación a esto, a continuación analizamos un conjunto de retos concretos que consideramos más destacables, si bien somos conscientes que no son los únicos, y que no reflejan el ciclo completo de la

Ingeniería del Software.

## 2.1. Satisfacción del cliente

En ocasiones parecemos olvidar cuál es el fin del proceso de construcción del software: satisfacer las necesidades del cliente. El software construido debe dar respuesta a los requisitos y a las expectativas depositadas en él. Y aquí es imprescindible poner el foco en la toma de requisitos y la ejecución de las pruebas, sin olvidar el resto de fases del proceso. Parece obvio, pero hoy en día es un aspecto que sigue sin estar resuelto.

Vamos a suponer que realizamos una correcta toma de requisitos y que somos capaces de entender lo que nuestro cliente nos demanda. Uno de los aspectos a tener en cuenta, y olvidado en muchas ocasiones, es que los requisitos se deben trazar directamente con el resto de fases, hasta llegar a las pruebas, debiendo los casos de prueba verificar que el software obtenido cubre el alcance descrito en la toma de requisitos. Un error bastante común es probar el software directamente contra el diseño funcional o contra el diseño técnico, o lo que es peor, teniendo en cuenta únicamente cómo se ha codificado. Además, una insuficiente trazabilidad impacta también en esas fases, produciendo errores que propagan a las siguientes. Otro error grave es reducir el periodo de pruebas ya que, al tratarse de una de las fases que se encuentran al final del ciclo de vida del software, con más frecuencia de lo deseable se trata de reducir tiempos en ella para cubrir posibles desvíos en fases anteriores.

También es habitual olvidarse de que, además de funcionar correctamente, el tiempo de respuesta del sistema debe ser adecuado. Para ello hay que realizar pruebas de rendimiento (conurrencia de usuarios, volumen de datos, etc.), cuyo diseño y ejecución no deben dejarse para el final del proyecto, sino que deben abordarse concurrentemente al desarrollo. También debemos tener en cuenta las pruebas de regresión en un entorno de entrega continua.

¿Qué soluciones deberíamos dar a estos retos? Desde nuestro punto de vista sería necesario un trazado automático de los requisitos de usuario con las pruebas, mecanismos que permitan dimensionar de forma lo más exacta posible el resto de fases del proyecto a partir de estos requisitos, además de mecanismos que permitan una monitorización a alto nivel de un proyecto concreto y de la cartera de proyectos, previendo con la suficiente antelación (y analizando las causas de) desviaciones respecto a los objetivos (funcionales y técnicos) marcados, duración, costes, etc.

Por otro lado, serían especialmente útiles, mecanismos que proporcionen al usuario de las herramientas monitorización sobre la idoneidad del uso que esté haciendo de las mismas, y sobre todo que le motive a la hora de seguir de forma óptima la metodología propuesta al inicio del proyecto, incentivándoles a conseguir la excelencia en todo el proceso. Además se considera necesario proporcionar mecanismos de aprendizaje que eliminen y prevengan posibles malas prácticas, y un incremento de la usabilidad de las herramientas utilizadas, que facilite la correcta aplicación del proceso de Ingeniería del Software. Estos puntos entroncan con el concepto de productividad que tratamos a continuación.

Consideramos que la agilidad en todo el proceso, involucrando al usuario e integrando producción y desarrollo, supone otra respuesta a estos retos, como veremos más adelante.

## 2.2. Productividad / motivación

En cualquier actividad productiva es fundamental la motivación de las personas involucradas, tanto en la calidad como en la eficiencia de los proyectos que dirigen y ejecutan. Más aún en el entorno globalizado, en el que la Ingeniería del Software está cada vez más inmerso, y donde es necesario ser cada vez más competitivos. Además, el objetivo primordial del software que se desarrolla es ofrecer, no sólo respuestas a las necesidades más obvias (las explicitadas), sino también valor añadido a los clientes (y a su vez a los usuarios finales). Para ello es imprescindible el compromiso de los profesionales involucrados con la productividad y con la excelencia, de cara a conseguir dar respuesta a los objetivos reales del cliente de forma proactiva. Se trata, desde

nuestro punto de vista, de una actividad que no debe basarse únicamente en la mecanización, sino que, en cada fase del proceso, cada profesional debe aportar sus capacidades únicas para poder obtener los mejores resultados.

La necesidad de concienciar a los profesionales de la necesidad de su compromiso con la productividad y la excelencia plantea retos aún no resueltos. Para ello, en primer lugar, sería necesario poder contar con mecanismos que permitan monitorizar y valorar de forma objetiva las aportaciones de los profesionales en base a estos criterios, lo cual no es sencillo, teniendo en cuenta que muchos de ellos tradicionalmente se han venido valorando de forma subjetiva (proactividad, creatividad, colaboración, etc.). Además, los criterios supuestamente objetivos, en la práctica hoy en día son prácticamente imposibles de medir. Por ejemplo, los mecanismos utilizados para “medir” la productividad (número de líneas de código, de requisitos implementados, etc.) están afectados por factores que impiden (o dificultan mucho) su objetivización (complejidad y calidad del código, dificultad real de los requisitos, código heredado, etc.). A partir de esta monitorización se podrían seleccionar y aplicar los mecanismos de motivación más adecuados para cada persona. La usabilidad de las herramientas de ingeniería es otro aspecto fundamental para la productividad, facilitando (y optimizando) su correcto uso.

Para dar respuesta a los retos planteados también sería necesario contar con mecanismos para la asignación óptima de profesionales a proyectos, teniendo en cuenta el perfil de los mismos en base a los criterios expuestos. De esta forma se conseguiría optimizar no solo la productividad, sino también la obtención de los resultados que satisfagan a clientes y usuarios finales. Las necesidades de los proyectos en cuanto al dimensionamiento y capacidades del equipo varían con el tiempo, por lo que estos mecanismos deberían ser flexibles, en base a las características cambiantes de los proyectos, y anticipándose a estas variaciones.

### 2.3. Metodologías ágiles / nuevas formas de hacer las cosas

Como hablábamos al principio del artículo el mundo está cambiando a una velocidad que en ocasiones nos impide aceptar (o cuanto menos adaptarnos a) los cambios. En la Ingeniería del Software sucede algo parecido. Desde hace algunos años están apareciendo términos asociados a nuevas formas de trabajar (metodologías) como Extreme Programming, SCRUM, Agile, etc. que, si bien son conocidos, la realidad nos demuestra que en función de la compañía o del proyecto, no han sido adoptados, y cuando se han adoptado muchas veces ha sido de forma parcial, o incorrecta.

Estas metodologías presentan características/objetivos comunes: involucrar más al usuario en el proceso de desarrollo y entrega del producto final, compartir más información entre el equipo (trabajar entre pares, reuniones diarias, etc.) y plazos de entrega más cortos. Estas características ponen el foco en aspectos que las metodologías tradicionales no acababan de resolver, al tener proyectos demasiado largos donde el cliente daba sus requisitos al principio y al cabo de unos meses veía una primera versión del producto, que en muchos casos no tenía mucho que ver con lo que realmente necesitaba.

Si estas metodologías cubren aspectos a mejorar lo normal debería ser que se adoptaran de forma natural en los proyectos, entonces ¿por qué no se ha producido una adopción efectiva de los mismos? En nuestra opinión esto es debido, por un lado, al esfuerzo que supone adaptarse a ese cambio, y por otro a la resistencia al mismo. A muchas personas, en todos los niveles de las organizaciones, les cuesta adaptarse al uso de nuevas herramientas. Incluso actividades realizadas por la mayoría de las personas en la vida cotidiana encuentran oposición a la hora de ser realizadas en el entorno productivo, como puede ser la interacción a través de redes sociales, aspecto que analizaremos más adelante.

Aparte de las metodologías ágiles para las fases de desarrollo, también han surgido en los últimos años intentos metodológicos de trasladar el concepto de agilidad a las operaciones. Este es el caso de DevOps, que además propone la integración de ambas actividades (desarrollo y operación). Si las metodologías ágiles no se encuentran en la práctica implantadas efectivamente en la mayoría de las empresas, DevOps está aún más lejos de serlo, ya que implica la involucración y colaboración continua entre departamentos distintos, hasta el punto

de fusionarse en uno solo.

Estas dificultades a la hora de incorporar nuevas metodologías requieren de la incorporación de mecanismos que motiven a las personas a la hora de la adaptación a los cambios requeridos. Además, para paliar el esfuerzo que supone el cambio, serían necesarios medios que disminuyan la curva de aprendizaje de las nuevas metodologías y de las herramientas correspondientes para su soporte, además de una mejora de su usabilidad.

También consideramos necesarios mecanismos para analizar el comportamiento de las personas dentro de los equipos de Ingeniería del Software, a través de los patrones de uso de las herramientas. De esta forma se podría monitorizar el grado de adaptación a las nuevas herramientas y metodologías, y detectar mejoras a incorporar en las mismas.

## 2.4. Dispersión geográfica / diferencias culturales / idioma

Uno de los mayores retos a los que nos podemos enfrentar cuando queremos implementar un ALM (Application lifecycle management) en una compañía global es la dispersión geográfica. Disponer de equipos ubicados en diferentes geografías trae implícitos una serie de problemas: diferencias horarias, culturales e idioma. En este sentido, si cambiamos el punto de vista, podremos convertir lo que aparentemente parecen dificultades en la oportunidad que supone contar con una fuente capaz de generar continuamente conocimiento e ideas, y de enriquecer el generado por compañeros en otras partes del mundo, aportando puntos de vista distintos. Nuestra visión respecto a este fenómeno, por lo tanto, traduce los aparentes problemas en oportunidades de generación de conocimientos e innovaciones continuas.

Para poder aprovechar estas oportunidades es necesario evitar que los equipos trabajen como elementos estancos, sin compartir información, lo cual, obviamente, impacta en la ejecución y en el éxito de los proyectos. En este sentido es imprescindible que las personas sean capaces de trabajar en equipo y que se produzca una colaboración efectiva, principalmente con los miembros de su mismo equipo, pero también con el resto de la organización. Además, hay que tener en cuenta la dificultad añadida que supone la existencia de diferencias idiomáticas.

¿Cómo podemos articular la cultura de la colaboración y la innovación en un entorno de dispersión geográfica y cultural?, ¿cómo podemos hacer que el conocimiento compartido dentro de un equipo esté disponible para toda la compañía global? Una vez más, pensamos que la respuesta está en proporcionar mecanismos de motivación y de adaptación a nuevas formas de trabajo. En el siguiente punto vamos a tratar nuevos mecanismos para dar respuestas a este reto.

## 2.5. Cultura de la colaboración y la innovación

A parte de las implicaciones socioeconómicas, hay que tener en cuenta que la tecnología también ha contribuido a que vivamos y trabajemos en un entorno globalizado. Actualmente todo el mundo puede llegar a estar conectado en tiempo real gracias a las nuevas tecnologías, lo que supone que personas de distintos países, culturas, idiomas, etc. interactúan a diario y de forma continua en redes sociales, espacios abiertos, mensajería síncrona o asíncrona, etc.

Una comunidad de ingenieros de software de una compañía debe disponer de una metodología, procesos y herramientas que les permitan formar esa red profesional y que reduzca al máximo los retos asociados a la dispersión geográfica: distintos horarios, distintos idiomas, culturas, etc. Actualmente existen este tipo de espacios en muchas empresas, pero ¿se utilizan bien? Y, en los casos en los que el uso que se da a estas redes es el adecuado ¿se aprovecha realmente el conocimiento generado?

Uno de los casos más conocidos de éxito del aprovechamiento y puesta en valor del conocimiento generado en las redes sociales es el de las elecciones de Estados Unidos en 2012, donde a partir del estudio de la interacción social en estas redes se detectaron a los grupos clave, su ubicación, gustos, etc. para diseñar cuándo,

cómo y a través de qué medios hacerles llegar los mensajes de la campaña. Esta información se extrajo a partir de un análisis en gran parte manual, para el cual se precisó una gran inversión económica.

En el sector de la Ingeniería del Software (como en la mayoría) esta información no está siendo explotada de forma que se extraiga conocimiento clave a partir de ella. Es más, en los casos en los que se disponen de mecanismos “sociales” muchas veces se produce el fenómeno conocido como “infoxicación”. Del contenido vertido en las interacciones en redes sociales, debidamente analizado, se podría obtener información relevante sobre el conocimiento, capacidades e intereses reales de los profesionales, detectando a nivel global los expertos en cada campo (lo cual actualmente parece inabarcable en una empresa con miles de profesionales en todo el mundo), y pudiendo evitar que se realicen una y otra vez las mismas consultas a estos expertos. Se podrían emplear estas redes para trabajar colaborativamente, y que todos los equipos de la empresa tuvieran acceso al conocimiento generado en esas interacciones.

En Ingeniería del Software el conocimiento debería extraerse de forma continua, por lo que sería necesario contar con mecanismos automáticos de análisis de información específica vertida en las redes sociales profesionales, que permitiesen tener acceso al conocimiento requerido en cada momento. Estamos hablando, al fin y al cabo, de facilitar a los profesionales su adaptación a la cultura de la colaboración, incrementando al mismo tiempo su productividad.

Por otro lado, vemos especialmente necesario incorporar la cultura de la innovación (y la colaboración en esa innovación) al campo de la Ingeniería del Software. Estamos en un sector que, por la forma tradicional de trabajar (que no ha cambiado mucho en las últimas décadas), no contempla la innovación como herramienta de trabajo. Una empresa de Ingeniería del Software que cuente con cultura de la innovación estará capacitada para afrontar los retos y aprovechar las oportunidades de forma distinta a la tradicional, pudiendo idear dentro de cada proyecto nuevas formas de satisfacer a clientes y usuarios finales, de resolver los problemas que se presenten, e incluso idear cambios en la forma de trabajar que mejoren la productividad. Las redes sociales también nos parecen un mecanismo idóneo para articular esta cultura de la innovación colaborativa. En este caso nos enfrentamos al reto de detectar conexiones y evitar duplicar líneas de innovación, ideas, etc.

Hasta este punto hemos analizado los retos que, desde nuestro punto de vista, se plantean en el sector de la Ingeniería del Software de cara a posibilitar la sostenibilidad del mismo en los próximos años, y a incrementar la competitividad de las empresas. También hemos visto que el sector no está respondiendo de forma ágil a los avances tecnológicos y metodológicos que están produciéndose cada vez a mayor rapidez, fallando en la adopción de los mismos y perdiendo la oportunidad de aprovechar las posibilidades que ofrecen. A continuación vamos a exponer la visión tecnológica de Indra en relación a la aplicación de estas tecnologías, como soporte fundamental para dar respuesta a los retos planteados.

### 3. Propuesta tecnológica para los retos actuales y futuros de la Ingeniería del Software

Indra apuesta fuertemente por la innovación en todos los sectores. En el caso concreto del sector de la Ingeniería del Software, en los últimos años, la actividad innovadora de la compañía se ha traducido en una propuesta tecnológica, cuyos resultados se centran en proporcionar mejoras a la Suite MInd. Esta Suite supone el soporte tecnológico principal para los objetivos de industrialización de la producción de la compañía y proporciona toda la funcionalidad necesaria para soportar el ciclo de vida de aplicaciones de software tanto en el ámbito de Proyectos como de Servicios. Está construida sobre un conjunto de herramientas integradas entre sí (véase figura 1) y soporta distintas metodologías con la posibilidad de disponer de distintos flujos de trabajo y transiciones de estado en función de cada necesidad. MInd aporta a estos procesos un soporte estratégico, táctico y operativo.

Adicionalmente, MInd soporta y facilita la gestión del conocimiento, la automatización de tareas, el control

de parámetros de calidad y la comunicación eficiente y sistematizada entre los miembros de los equipos de trabajo. También proporciona las herramientas y los procedimientos para adaptarse a los estándares internacionales de calidad en procesos de software.

La Suite MInd integra herramientas comerciales, open source así como desarrollos propios. Indra ha ampliado las capacidades de todas estas herramientas para proporcionar mejoras y una estandarización de su uso dentro del ciclo de vida de los proyectos y servicios permitiendo la integración entre ellas para proporcionar un valor añadido a lo que esas herramientas proporcionan por separado. De este modo se logra mantener la trazabilidad entre los diferentes procesos de una organización proveedora o consumidora de servicios de consultoría o desarrollo informático.



Figura 1. Suite MInd de Indra.

Entre las principales ventajas de MInd se encuentra la incorporación de un cuadro de mando que ofrece una visión global de todos los proyectos y servicios, permitiendo conocer el tamaño de las operaciones en curso y su evolución. De esta forma se facilita un seguimiento y control centralizado con independencia de que la

ubicación de la producción esté distribuida en diferentes países.

Por otro lado MInd proporciona mecanismos de automatización en la recogida de métricas y KPIs, y un modelo de estimación competitivo. Aprovechamos nuestra experiencia en los niveles 4 y 5 de CMMI (Gestión Cuantitativa y Alta Madurez) utilizando técnicas estadísticas para el control de los errores software mediante gráficos de control.

Como puede verse, la Suite MInd incorpora ya capacidades relacionadas con los retos y oportunidades analizados. A continuación vamos exponer una serie de propuestas tecnológicas concretas, con las que trataremos de profundizar en la construcción de mecanismos para dar respuesta a dichos retos y oportunidades.

### 3.1. Big Data

Las incorporaciones en el sector de la Ingeniería del Software de las posibilidades que ofrecen las tecnologías relacionadas con el Big Data permitirían analizar la información heterogénea que se genera de forma continua en las distintas herramientas del ciclo de vida de Ingeniería del Software en entornos de desarrollo global multifactoria (con ubicaciones distribuidas geográficamente por todo el mundo) con el fin de poder realizar previsiones que permitan anticiparse a distintas situaciones para optimizar la toma de decisiones relacionadas con todas las fases de los proyectos, mejorando ostensiblemente la productividad y calidad de los productos y servicios desarrollados de acuerdo con el estado real actual y futuro de los proyectos, y con las necesidades de los clientes y usuarios.

Para ello, nuestra aproximación se centra en los siguientes aspectos:

- Extracción de “dark data” en factorías de software, con el propósito de obtener información importante sobre los productos, procesos y proyectos (y su calidad) a partir de la gran cantidad que existe hoy en día de datos no identificados ni aprovechados en el análisis de datos en las factorías de software.
- Construcción de mecanismos de control de la calidad de datos de Ingeniería del Software que permita asegurar la calidad de las colecciones de datos de los proyectos que entran en la fase de preprocesamiento de los algoritmos de análisis de Big Data.
- Adaptación de técnicas y herramientas analíticas, de forma que se posibilite la realización de predicciones y la optimización de la toma de decisiones en entornos de desarrollo global de software. Estas técnicas supondrían nuevas capacidades para la toma de decisiones en la gestión de proyectos software, realizando en base a la información del proyecto (y de proyectos precedentes) predicciones de costes, esfuerzos y calidad, posibles errores (con la correspondiente generación de casos de prueba asociados), tiempo de mantenimiento, desviaciones respecto a los objetivos, etc. También asistiría en la mejora de los procesos (ej. CMMI) mediante la predictibilidad del rendimiento de los mismos, en el enfoque de gestión y en la mejora del rendimiento de la organización. De esta forma también se permitiría dimensionar dinámicamente los proyectos a partir del trazado de los requisitos con la situación real de cada fase del proceso
- Construcción de herramientas analíticas para el análisis de la congruencia sociotécnica en desarrollo global de software, lo cual permitirá optimizar la organización y gestión de proyectos y factorías de software.
- Construcción de cuadros de mando que presenten información analítica en tiempo real.
- Extracción de patrones de comportamiento en el uso de las herramientas y monitorización continua de este uso. De esta forma se permitiría, por un lado monitorizar el grado de adecuación del uso que se hace de las herramientas, detectando desviaciones en el adecuado cumplimiento del proceso de Ingeniería del Software por parte de los equipos de los distintos proyectos. Por otro lado, se podrían detectar mejoras a implantar en las herramientas actuales y el grado de adaptación de los profesionales a las nuevas herramientas y metodologías.



Además, en relación a estas tecnologías, pensamos que la Ingeniería del Software debería dar respuesta a las dificultades que existen hoy en día a la hora de desarrollar aplicaciones Big Data, en las que no se disponen de mecanismos de automatización y los desarrollos son costosos en cuanto a tiempo y nivel de especialización requerida, realizándose de forma ad-hoc para cada caso concreto. Por ello pensamos que dentro de la adopción del Big Data en el sector de la Ingeniería del Software se debería incluir el poder proporcionar entornos de desarrollo para aplicaciones Big Data que automaticen la creación de las mismas. Para ello, nuestra propuesta tecnológica se basa en las Líneas de Producto Software (LPS), facilitando la incorporación de funcionalidades a las mismas mediante la integración automatizada de componentes reutilizables. Dichos componentes, serían soportados por distintas tecnologías y empaquetados como cajas negras parametrizables e integrables en las nuevas aplicaciones Big Data mediante API's bien definidas, formando un auténtico toolkit o librería de componentes que podrían ser usados de modo automatizado mediante el entorno LPS (que los parametrizaría siguiendo las especificaciones del usuario), o de forma independiente por programadores convencionales.

### 3.2. Tecnologías colaborativas

Si bien, la incorporación de herramientas sociales en todos los sectores es un hecho asumido, la Ingeniería del Software en la actualidad no explota de forma automatizada estas nuevas vías de comunicación social para la generación, compartición y sobre todo reúso de conocimiento, desaprovechando numerosas oportunidades presentes en un campo en el que las relaciones entre personas tienen un alto impacto en la generación de conocimiento y en la obtención de resultados. De ahí la necesidad de socializar y potenciar las relaciones entre miembros de un equipo de trabajo, automatizando la utilización del conocimiento generado en las interacciones entre los mismos. Las redes sociales, las plataformas de comunicación y, en general, los nuevos canales de comunicación pueden servir para mejorar la integración de equipos humanos y, por extensión, la calidad y eficacia de los productos software generados. Sería necesario centrarse en la extracción y gestión del conocimiento, ya que este tipo de redes implican un volcado de gran cantidad de conocimiento que no es explotado a día de hoy, necesitando nuevas tecnologías capacitadoras en este campo.

Nuestro enfoque tecnológico se centra, por lo tanto, en el desarrollo de técnicas de Listening Platform para la extracción de conocimiento en el espacio social, identificando temáticas (tecnológicas, metodológicas, etc.), necesidades y opiniones, relaciones entre usuarios, emisión de sugerencias automáticas, creación de vínculos entre usuarios, detección y búsqueda de usuarios expertos en cada tema, etc. Además de tener en cuenta las particularidades de los mensajes en redes profesionales (generalmente largos, con mucha semántica, y muy entrelazados entre sí, y con documentación externa tanto dentro como fuera de la empresa).

Por otro lado, vemos un gran potencial en la aplicación estas técnicas avanzadas de análisis de redes sociales en la innovación aplicada a la Ingeniería del Software, sirviendo como catalizador para la implantación de una cultura innovadora. Nuestra propuesta se centra en proporcionar mecanismos que permitan gestionar el conocimiento generado a través de la interacción social en la generación de ideas. De esta forma ese conocimiento podrá estar disponible, para facilitar la generación y evaluación de las mismas, relacionándolas con contenidos externos (tecnológicos, de mercado, etc.) e internos (productos, proyectos, estrategia). En este caso se haría necesario también incorporar análisis de terminología, polaridad (valoración sobre las ideas o la opinión ante la participación de los usuarios: positiva o negativa), eliminación de ruido, detección de usuarios influyentes, etc.

### 3.3. Gestión flexible de equipos de Ingeniería del Software

En este ámbito, vemos necesario el desarrollo de tecnologías que permitan una asignación óptima de recursos a proyectos, incorporando además una estimación inteligente de los tiempos de desarrollo previstos y una medición precisa del conjunto de dimensiones que afectan a un proyecto, con capacidad predictiva y de adaptación ante evoluciones imprevistas, permitiendo la anticipación a las mismas a la hora de determinar cuáles son los profesionales idóneos entre los disponibles en la organización para llevar a cabo el proyecto, tanto a nivel cualitativo como cuantitativo.

Para ello nuestra propuesta tecnológica se focaliza en el desarrollo de mecanismos que posibiliten cubrir los siguientes aspectos:

- Incorporación e interpretación la información procedente de distintos sistemas corporativos y de las herramientas de Ingeniería del Software para su análisis.
- Representación y medición automatizada del talento de forma integral, usando como base un modelo objetivo de representación del talento, que además tenga en cuenta otros factores, como los rasgos de la personalidad o las relaciones personales entre profesionales.
- Clasificación de profesionales en base a sus competencias adquiridas en la organización, medición de su evolución en el tiempo y predicción de estados futuros mediante modelos de autoaprendizaje. Para llevar a cabo esta categorización se deben tener en cuenta de forma objetiva las aportaciones de cada profesional en base a criterios difícilmente objetivizables, como son la proactividad, creatividad, colaboración, etc.
- Poner especial atención en la reputación como factor determinante, garantizando la calidad de su medición y minimizando los riesgos de fraude.
- Categorización de proyectos, de forma integral e inteligente mediante consultas automáticas sobre datos internos de la compañía, de forma que se obtenga una representación formal de la misma, susceptible de ser procesada por máquinas, permitiendo visualizar los recursos disponibles en la organización, teniendo en cuenta restricciones de cada proyecto: número de personas necesarias para su ejecución, con qué perfiles, durante cuánto tiempo serían necesarias, etc.

Las tecnologías que estamos desarrollando para hacer posible estos objetivos se basan en:

- Extracción, almacenamiento y procesamiento sin supervisión (y en lenguaje natural) de grandes volúmenes de información heterogénea procedente de fuentes dispersas.
- Algoritmos de PLN (procesado de lenguaje natural) que supongan herramientas lingüísticas robustas, orientadas tanto al análisis del texto en dependencias sintácticas como al reconocimiento de entidades, y adaptables tanto a varias lenguas como a fuentes textuales heterogéneas.
- Técnicas de categorización y segmentación, de caracterización dinámica que permitan cuantificaciones evolutivas, de matching, de estimación y recomendación automáticas, técnicas predictivas; etc.

### 3.4. Gamificación

Para dar respuesta a los retos planteados en relación a la motivación de las personas en los equipos de Ingeniería del Software, tanto para su máxima implicación y aportación a los proyectos, como en su adaptación al cambio (tecnológico y metodológico), la aproximación de Indra se centra en el desarrollo de tecnologías que permitan el uso de la mecánica de jugabilidad en Ingeniería del Software, más concretamente en las herramientas del ciclo de vida del desarrollo. Estas tecnologías estarán enfocadas a permitir el máximo aprovechamiento de las ventajas que la gamificación puede aportar al proceso de Ingeniería del Software, y que posibiliten hacerlo desde un punto de vista integral, en lugar de gamificando funcionalidades aisladas.

De esta forma se podrán potenciar distintos aspectos como la estimación de requerimientos, la adquisición de buenas prácticas, el intercambio de conocimientos dentro del equipo, las pruebas, el mantenimiento, la gestión del proyecto, el uso de nuevas herramientas, la aplicación de nuevas metodologías, etc. mediante la motivación y el compromiso de los profesionales, la innovación, la colaboración y la interacción social, etc. Por lo tanto, a través de la gamificación tratamos de ayudar a dar respuesta a la mayoría de los retos planteados (satisfacción del cliente, productividad, agilidad, colaboración, etc.).

Para dar soporte al desarrollo y aplicación del concepto de gamificación en Ingeniería del Software vemos necesario el desarrollo de una serie de tecnologías habilitadoras. En concreto hemos detectado la necesidad de llevar a cabo los siguientes desarrollos tecnológicos, para proporcionar características avanzadas a la gamificación: técnicas multidispositivo; técnicas de Inteligencia artificial y estadística para el análisis del compor-

tamiento de los usuarios y el establecimiento de perfiles (profiling), el análisis de emociones y la personalización; técnicas para la construcción de diálogos flexibles y naturales; técnicas de GIS (Geographic Information Systems); técnicas de colaboración y socialización; técnicas para metodologías ALM informales, ágiles y sociales; etc.

A partir de estos aspectos estamos centrando nuestros esfuerzos en la gamificación de las distintas herramientas del ciclo de vida del desarrollo del software, principalmente la toma de requisitos, la gestión de proyectos y la gestión (definición, ejecución, etc.) de pruebas.

A parte de estos aspectos, consideramos imprescindible desarrollar mecanismos que posibiliten la automatización de la gamificación de forma unificada y coherente de las distintas herramientas y entornos del ciclo de vida, a través de un framework que implemente las funcionalidades comunes y generales en las mecánicas de gamificación, además de la incorporación de características avanzadas, y automatización de las funcionalidades y contenidos incluir para cada usuario.

### 3.5. Serious Games

El concepto de Serious Games se basa en capitalizar el poder de atracción de los juegos para ayudar a las personas a aprender y entrenar, haciendo que se aprenda y disfrute del aprendizaje. En primer lugar, en general, las personas entendemos mejor a través de experiencias prácticas en lugar de teorías, y los juegos ofrecen esta experiencia. Además de eso, los juegos involucran y motivan a los jugadores. Los juegos presentan problemas bien ordenados, en los que la información se presenta al jugador bajo demanda y en el momento requerido, manteniendo un equilibrio correcto entre desafío, el aburrimiento y la frustración. Por otra parte, además de la motivación y la diversión, los juegos permiten repetitividad de las tareas, niveles de dificultad adaptativos, así como la posibilidad de informar de forma automática sobre el comportamiento del alumno, lo que permite extraer de estos resultados datos más amplios sobre el proceso de aprendizaje / formación.

No se debe confundir el concepto de juego serio con el de gamificación. La gamificación consiste en incorporar mecanismos de jugabilidad (competitividad, recompensas, desbloqueo de ítems, etc.) para motivar aspectos concretos a través de incentivos. Los juegos serios permiten el aprendizaje de procesos y tareas de forma transparente al estar en realidad usando un juego, es decir, jugando. Basándonos en esta diferencia de base pensamos que la aplicación de los dos conceptos simultáneamente multiplicaría la efectividad cada uno de ellos por separado para la consecución de los objetivos planteados.

En concreto, desde el punto de vista de los retos planteados, la propuesta tecnológica de Indra se centra en la aplicación del concepto de juegos serios para disminuir la curva de aprendizaje en nuevas metodologías y herramientas, no sólo venciendo la oposición al cambio en las empresas, sino proporcionando además la motivación necesaria para su adopción. Además se podría aplicar a mejorar el uso de las herramientas y procesos actuales, en base al refuerzo del aprendizaje adquirido, eliminando posibles malas prácticas arrastradas en el tiempo.

### 3.6. Interfaces de usuario avanzadas

En los últimos años hemos vivido la irrupción de nuevas formas de interacción con la tecnología, de la mano del campo de los videojuegos. Son muchas las simulaciones y pruebas de concepto que hemos podido ver, sin embargo, la adopción de nuevas formas de interacción con las tecnologías para uso “productivo” parece aún lejana, no sólo en el sector de la Ingeniería del Software. Supone un reto, por lo tanto, trasladar las ventajas que estos nuevos dispositivos proporcionan al desarrollo de proyectos software.

La propuesta tecnológica de Indra en este sentido se centra en incorporar las posibilidades que ofrecen estas nuevas formas de interacción a las aplicaciones de uso ordinario y, especialmente, empresarial. Esto implicaría una nueva forma de concebir la interacción de usuario y una evolución tecnológica en el proceso de producción de software.

Para ello pensamos que supondría ventajas competitivas (permitiendo adelantarse a una necesidad futura evidente), el poder proporcionar un entorno de desarrollo para aplicaciones de interacción natural de forma aislada o combinadas con los mecanismos tradicionales de interacción. Sería un entorno gráfico que permitiría incluir acciones de interacción natural en aplicaciones de ámbito generalista (ej. aplicaciones web, aplicaciones de escritorio, etc.) que no existen en la actualidad de forma generalizada.

Por otro lado, la incorporación de estos desarrollos tecnológicos en las propias herramientas de Ingeniería del Software proporcionaría incrementos en la productividad al facilitar el uso de las mismas, además de la motivación de los profesionales a la hora de desarrollar su trabajo y de seguir correctamente el proceso de Ingeniería del Software, a la vez que se facilitaría la adaptación al uso de nuevas herramientas debido a la reducción del impacto que supondría una interacción natural en esta adaptación, proporcionando así una motivación adicional.

### 3.7. DevOps

Como hemos visto anteriormente, la incorporación efectiva de la agilidad en todo el proceso de Ingeniería del Software supondría una oportunidad para las empresas que consigan llevarlo a cabo de forma efectiva. Nuestra respuesta tecnológica a esta oportunidad se centra en el uso de las tecnologías descritas en los puntos anteriores.

A los retos planteados en este sentido anteriormente se une la dificultad que supone que las metodologías ágiles en realidad no son metodologías propiamente dichas, sino guías, “frameworks” o conjuntos de buenas prácticas que tratan de complementar los principios del “manifiesto ágil”. Esto provoca que cada empresa (incluso cada equipo) implemente la agilidad de forma distinta. Esta dificultad se hace más pronunciada en el caso de DevOps, donde la agilidad va más allá del proceso de desarrollo, añadiendo también la producción, formando una única actividad que se realiza de forma continua.

En este sentido, en el desarrollo de la Suite MInd se ha aplicado la metodología de DevOps, a partir de la incorporación de herramientas habilitadoras para la integración y entrega continuas, y realizando el desarrollo y producción de forma conjunta y continua por parte de un único equipo.

Actualmente los pasos que estamos siguiendo para la asimilación completa de la “metodología” DevOps en el proceso de Ingeniería del Software se centran en la mejora de las herramientas habilitadoras, y en trasladar las lecciones aprendidas y las mejores prácticas detectadas en el desarrollo de la Suite MInd a los proyectos (de tipo producto y servicio) que se desarrollen utilizando dicha suite.

## 4. Conclusiones

En este artículo hemos visto cómo nuevas tecnologías que se están aplicando en diversos ámbitos lúdicos y productivos se pueden adaptar y aplicar para dar respuesta a los principales retos existentes en el sector de la Ingeniería del Software.

Hemos visto cómo los distintos retos y oportunidades planteados están estrechamente relacionados entre sí, y cómo, de la misma forma, la incorporación de cada una de estas tecnologías pueden dar respuesta a uno o varios de forma simultánea. En este sentido hemos expuesto las propuestas tecnológicas en las que Indra está trabajando para proporcionar el soporte tecnológico necesario para cubrir estos aspectos. También hemos presentado la Suite MInd y sus principales características, la cual supone el elemento vertebrador de las propuestas tecnológicas planteadas.

Como conclusión principal queremos destacar que la Ingeniería del Software es un sector que necesita incorporar el dinamismo y agilidad tanto en la adopción efectiva de nuevas metodologías, como tecnologías y herramientas, para dar respuesta a los retos planteados en la actualidad y en el futuro inmediato, que pueden ser vistos como importantes oportunidades competitivas.

Cómo citar este artículo / How to cite this paper

Sánchez Belmonte, G., García Moreno, C., y Hernández González, Y. (2014). Propuesta tecnológica de Indra para afrontar los retos inmediatos de la Ingeniería del Software. *International Journal of Information Systems and Software Engineering for Big Companies (IJISEBC)*, Vol. 1, Num. 1, pp. 24-36. Consultado el [dd/mm/aaaa] en [www.ijisebc.com](http://www.ijisebc.com)