

# Aplicando Metodologías Activas en la Enseñanza de Ingeniería del Software en Ingeniería Informática

Víctor Flores, Yessica Gómez

**Title— Applying active methodologies for teaching Software Engineering in Informatics Engineering**

**Abstract — The traditional teaching-learning process used by Chilean universities are changing. This document describes two innovative strategies applied to teach Software Engineering (SE) at two universities in Chile. Experiences are part of our ongoing effort to improve the current teaching-learning strategies of SE in the mentioned universities. In this paper we describe our experiences using both Problem-based learning (PBL) and an agile Software Engineering Methodology (SCRUM), to achieve learning outcomes and to improve students' participation during classes. We present two ways to correlate activities of a software project, these activities are tasks of a software methodology, and an active learning strategy based on PBL. The strategies used to verify the accomplishment of these goals are detailed too.**

**Index Terms—Active Methodology, Project-based learning, Engineering Education, teaching-learning process, Learning systems, Software Engineering.**

## I. INTRODUCCIÓN

Los métodos tradicionales están dando paso a las metodologías activas en Chile, gracias a las reformas educativas que apoyan iniciativas de aprendizaje adecuadas a las habilidades del siglo XXI [30]. La enseñanza de asignaturas troncales en Ingeniería Informática, como es el caso de Ingeniería del Software, deben tener como objetivo alcanzar competencias transversales y específicas en el área de proyectos software, además de cohesionar la teoría con un temprano desarrollo profesional.

Para lograrlo, una buena alternativa es hacer que los estudiantes integren los contenidos aprendidos con la realización de proyectos o prácticas, en un ambiente real y empresarial. Esto también permite que los estudiantes vean los resultados de aprendizaje más rápidamente y validen su utilidad práctica (aprendizaje contextualizado) [1]. Con esta idea en mente se han aplicado metodologías activas en la asignatura Ingeniería del Software II (IS-II) de la carrera Ingeniería Civil Informática, en la Universidad Católica del Norte (UCN) y la Universidad Católica del Maule (UCM).

Una metodología activa de enseñanza-aprendizaje puede entenderse como una forma de enseñar donde el alumno asume un rol protagónico y es responsable de su aprendizaje. Además, en este tipo de aprendizaje el alumno debe descubrir parte del conocimiento necesario para realizar el trabajo [22]. El docente actúa como facilitador del contexto de trabajo para que se dé el aprendizaje y guía también este proceso [7, 9], pero es el alumno quien asume activamente el compromiso, desarrollando además habilidades como trabajo autónomo, responsabilidad, trabajo en equipo o capacidad de síntesis [18].

En el caso de la UCN (sede Antofagasta) y hasta el momento de implementar la metodología activa, los alumnos realizaban prácticas de forma separada de la cátedra, con casos teóricos o preparados por el docente según sus necesidades de enseñanza de ciertos conocimientos, siendo los índices de reprobados relativamente altos. En tanto, en la UCM la enseñanza de la asignatura se desarrollaba inicialmente de manera tradicional: centrada en el docente y donde las estrategias eran expositivas y de ejercicios. Lo anterior impactaba en la poca asistencia y poca motivación de los estudiantes y los resultados de aprendizaje se concentraba sólo en el conocimiento.

En ambas universidades, la carrera Ingeniería Civil Informática está en proceso de rediseño curricular, para pasar de un modelo por objetivo a uno orientado por competencias. Por tanto, en los últimos años (en ambas universidades pero de forma separada) se han producido cambios significativos a la hora de desarrollar las cátedras, lo que ha inspirado cambios de paradigmas en las estrategias didácticas docentes.

En este trabajo se describen las técnicas usadas para integrar la práctica a la teoría, y que esa teoría sea la base para el desarrollo de proyectos software realizado por todos los alumnos del curso, trabajando en equipos con roles definidos. El proceso de enseñanza-aprendizaje es en ambos casos experiencial y sustentado en casos reales, lo que inclina a que el aprendizaje se base significativamente en la vivencia del estudiante [4].

El fin de esta práctica docente es facilitar al alumno, a través de la aplicación de teorías en un ambiente dinámico, la interpretación de conceptos y aplicación de conocimientos teóricos a casos prácticos de proyecto software, que cubra las necesidades académicas pero que a la vez acerque al alumno a lo que será su desarrollo profesional.

El resto de este documento está estructurado de la forma siguiente: el apartado II describe experiencias

Victor Flores is with the Universidad Católica del Norte, Angamos 0610, Antofagasta, Chile (corresponding author to provide phone: +56-55-2355164; e-mail: vflores@ucn.cl). Yessica Gómez is with the Universidad Católica del Maule, San Martín, Talca, Chile (e-mail: [jgomez@cftsanaustin.cl](mailto:jgomez@cftsanaustin.cl)).

previas, generalidades de paradigmas de aprendizaje (convencional y activo) y conceptos como el de PBL, que son importantes en el diseño de experiencias de aprendizaje. En el apartado III se detalla el contexto de trabajo y en el apartado IV se describen las experiencias de aprendizaje activo como forma de alcanzar los objetivos planteados en cada universidad, se describe además los aportes de cada experiencia y los instrumentos usados para medir el desempeño de los alumnos en cada caso. Finalmente se presentan las conclusiones, los agradecimientos y las referencias que sustentan el trabajo de investigación.

## II. APRENDIZAJE ACTIVO, EXPERIENCIAS PREVIAS

Previo a describir el aprendizaje activo es pertinente contextualizar el enfoque educativo actual. Las reformas educativas surgen porque los enfoques tradicionales, como las corrientes pedagógicas por objetivos, no dan respuestas adecuadas en los tiempos actuales. Emerge entonces una visión basada en competencia. La competencia entendida (en un amplio sentido) como las habilidades, aptitudes y conocimientos necesarios para proponer y adaptarse, en un contexto cambiante y orientado al trabajo en equipo [13]. La competencia es en este contexto el elemento organizador de programas de estudios, que tienden a mejorar los resultados en el enfoque educativo actual de Chile.

El aprendizaje activo puede entenderse como el compendio de técnicas docentes cuyo objetivo es desarrollar en el alumno habilidades, capacidades y competencias, demandadas en los profesionales en Chile, siendo el alumno corresponsable de su propio aprendizaje.

### A. Antecedentes Generales de Trabajo

De acuerdo con [5], los programas revisados dentro de una perspectiva situada no sugieren sino posibilidades de acción con un conjunto de recursos para una situación, en lugar de normas de conocimientos para memorizar.

Este enfoque surge como un medio para establecer un diálogo entre la práctica y lo curricular. Esto impone el desafío de crear bancos de situaciones como apoyo a los docentes e impone un cambio de paradigma de enseñanza centrado, como acción educativa, en el aprendizaje de los estudiantes [5].

Así el estudiante es socio activo del proceso de aprendizaje, guiado y acompañado por el profesor. Con ello se consigue acercar al estudiante al medio profesional, que exige adquisición de habilidades cognitivas y metacognitivas [14, 15]. El paradigma de aprendizaje centra su atención en desarrollo de competencias, la capacidad de responder a preguntas complejas y la integración de conocimientos cognitivos. En este contexto el docente diseña actividades de aprendizaje, y es un guía y acompañante del estudiante.

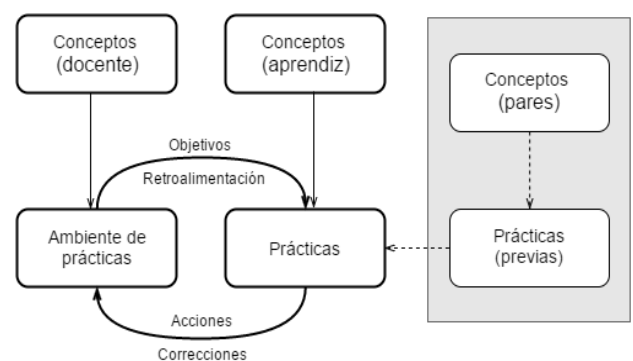
Los métodos activos conforman una tipología de recursos que permiten un aprendizaje a través de la práctica [3, 15]. Ésta toma lugar cuando el profesor ha preparado material o recursos de aprendizaje para que el estudiante aplique su comprensión de los conceptos en función al logro de la tarea. El aprendizaje a través de la

práctica es parte esencial de la experiencia de aprendizaje, porque invita al aprendiz a adaptar su comprensión de conceptos a la tarea dada y luego reflejarla sobre lo que la experiencia significa, además de incluir al proceso la posibilidad de retroalimentación o feedback significativo [3, 7, 9].

La retroalimentación posibilita al estudiante mejorar su acción, modulando su repertorio de práctica de acuerdo a cuán efectiva es su acción. El feedback significativo está presente porque el estudiante debe ser capaz de interpretar el feedback para modular su práctica. Otra de las bondades de estos métodos es que facilitan la comunicación entre pares, a través de un marco conversacional. La fig. 1 ilustra el proceso de aprendizaje a través de la práctica, se indica en la figura que el ambiente de práctica se nutre con conceptos y material preparado por el profesor para que, a partir del logro de los objetivos y la retroalimentación iterativa entre la teoría y la práctica, el alumno sea capaz de aprender, interviniendo además en este proceso los conceptos aportados por los pares y experiencias previas adquiridas igualmente mediante prácticas. Las acciones y/o correcciones hechas en la actividad de práctica general retroalimentación al ambiente de la práctica, en cuyo caso es importante para el docente velar por que dicha retroalimentación se mantenga siempre dentro de los límites de los objetivos de la actividad de práctica.

El marco conversacional considera la unión del aprendizaje experimental y del aprendizaje conversacional, es decir, se adiciona un claro énfasis en la relación iterativa entre teoría y práctica, la necesidad de extenderse para mostrar cómo el ciclo experiencial puede ser mejor motivado a través del enganche con el ciclo de comunicación con los pares [6]. El valor pedagógico de aprender a través de la práctica, con el estudiante recibiendo retroalimentación acerca de su actuación, ha sido fundamental para todas las teorías de aprendizaje [19].

Existen variados factores que influyen en el diseño de actividades de aprendizaje, como ilustra la Fig. 2. Para identificar y seleccionar bien estos factores se debe evaluar su pertinencia, factibilidad, flexibilidad y valor que aportan a la enseñanza [3]. Con ello se podrá adoptar un modelo conversacional de aprendizaje considerando las estrategias adecuadas según objetivos, contemplando el medio y las tecnologías de apoyo, configurando así una



**Fig. 1.** Modelo convencional en el aprendizaje por prácticas (inspirado parcialmente en [3]).



son más accesibles para equipos pequeños de desarrollo. Ejemplos de estos modelos son CMMI (Capability Maturity Model for Integration), TPS (Team Software Process), ISO/IEC 15504 (Information Technology-Process Assessment), ISO/IEC 12207 (Information Technology/Software Life Cycle Processes), PSP (Personal Software Process), MOPROSOFT (Modelo de proceso para la industria del software), ISO 29110, etc.

En este contexto la enseñanza de la disciplina implica que en PBL se mantenga un plan riguroso de entregas, apegado a normas de calidad y de planificación de trabajo; también exige que los estudiantes instalen estas buenas prácticas como desarrolladores para lograr alcanzar el nivel de calidad exigido por las partes interesadas (*stakeholder*).

#### IV. DESCRIPCIÓN DE LAS EXPERIENCIAS

En esta sección se detallan las experiencias docentes para alcanzar los objetivos concretos en cada universidad para la asignatura IS-II, así como los resultados obtenidos en ambas experiencias. La asignatura IS-II se imparte en el IX semestre de Ingeniería Civil en Informática<sup>3</sup> (carrara de 6 años).

##### A. Experiencia en la UCN

La experiencia en la UCN se realiza en IS-II que se imparte en dos clases de dos horas académicas por semana, durante 19 semanas lectivas. Entre los objetivos de esta propuesta se destacan:

- Obj1-UCN: Acercar a los alumnos al ejercicio profesional, enfrentando a los estudiantes a un proyecto real desde el principio hasta lograr una aplicación software, aplicando metodologías de desarrollo software.
- Obj2-UCN: Desarrollar competencias necesarias para el trabajo en equipo y el desarrollo de software, considerando el uso de metodologías.
- Obj3-UCN: Disminuir el abandono en la asignatura, con tareas cercanas al ámbito profesional y motivar el aporte personal al proyecto grupal, involucrando a los alumnos en proyectos reales, supervisando y validando su participación, logro de tareas concretas en las diferentes fases de un producto software y capacidad de retroalimentación activa.

En esta experiencia se usan las fases de análisis de necesidades, diseño, desarrollo y pruebas de una aplicación software, basándose en principios del Manifiesto Ágil<sup>4</sup>. Para desarrollar el trabajo se agruparon los alumnos en grupos libremente elegidos tras realizar la dinámica descrita en la tabla I. Cada grupo de trabajo selecciona un dominio y propone un cliente para desarrollar el proyecto software. El resultado: prototipos de aplicaciones software con manejo de una base de datos para clientes reales.

El docente participa dictando directrices como por ejemplo: (1) limitando la cantidad de alumnos con un

mismo perfil (habilidades) en cada grupo, dimensionando el tamaño del problema, estudiando la factibilidad técnica de los proyectos, ayudando a estimar la cantidad de horas de trabajo, etc., (2) definiendo los formatos de informes (basado parcialmente en [3]) para asegurar retroalimentación activa y (3) definiendo y priorizando la lista de objetivos del proyecto (*product backlog*).

Para estimular el trabajo creativo y eficaz se seleccionan proyectos de clientes con necesidades reales en dependencias como: departamentos, centros de

TABLA I

ASPECTOS PARA LA SELECCIÓN DE ROL Y GRUPO DE TRABAJO.

<b>Descripción:</b> El objetivo de esta dinámica es orientar la selección de los miembros del grupo para asumir roles en el proyecto. Se recomienda buscar compatibilidad horaria y posibilidad de reunirse una vez a la semana (mínimo) para participar en un grupo. En cada rol puede existir más de un alumno o un alumno puede asumir más de un rol, excepto líder de proyecto. Los roles pueden ser itinerantes, es decir: pueden cambiar entre los alumnos del grupo	
Los objetivos de aprendizaje son:	
<ul style="list-style-type: none"> <li>• Desarrollar destrezas en el uso de modelos de desarrollo software orientado a objeto, liderazgo</li> <li>• Aprendizaje autónomo y trabajo en grupo</li> </ul>	
Líder de proyecto	Persona que coordinará las actividades del proyecto, tendrá contacto en el cliente y será responsable del control de versiones (mínimo dos en el proyecto), de cumplir la planificación y de coordinar las entregas a tiempo de las versiones intermedias y versión final, con todo lo comprometido para cada entrega. Esta persona interactuará con todo el equipo de proyecto. Recomendable que el alumno que asuma este rol posea habilidades de liderazgo, trabajo en grupo, comunicación, responsabilidad y negociación
Responsable de base de datos y documentación	Persona encargada de mantener el backlog del producto software y todo lo relacionado con modelos de datos, diseño e implementación de la base de datos. Esta persona interactuará directamente con el programador, responsable de pruebas y plan de calidad. Recomendable poseer habilidades de comunicación, responsabilidad y negociación
Programador	Persona encargada de codificar cada versión del producto software y realizar las primeras validaciones de los programas para asegurar que el funcionamiento cumpla con los requisitos. Esta persona interactuará directamente con el responsable de pruebas y plan de calidad. Recomendable poseer habilidades de comunicación, responsabilidad y trabajo en grupo
Responsable de pruebas y plan de mantenimiento del software	Persona encargada de diseñar las pruebas, asegurar el cumplimiento de los requisitos y calidad del producto software. También será responsable de diseñar e implementar el plan de mantenimiento del producto. Esta persona interactuará directamente con el responsable de base de datos y documentación y con el programador. Recomendable poseer habilidades de comunicación, responsabilidad y trabajo en grupo

**Restricciones:** al final de la sesión los grupos deberán entregar una planificación de revisión semanal de avances con el profesor.

Para facilitar la selección de compañeros de grupo, usar estas preguntas: 1- ¿dispones de tiempo entre clases (mínimo 4 horas a la semana) para hacer trabajo cooperativo con tus futuros compañeros de grupo?, 2- ¿qué conocimientos previos de (1) planificación y gestión de proyectos, (2) bases de datos y (3) programación en Java posees y puedes aportar al grupo? y 3- ¿conoces y has trabajado antes con personas de esta clase?

Esta actividad se desarrollará en 30 minutos

3 <http://malla.ucn.cl/ICCI/index.html>

4 <http://www.agilealliance.org/> (última consulta: nov2016)

investigación o desarrollo adscritos a la UCN, en segunda instancia las empresas de servicio o de explotación de la minería de cobre (principal actividad en la región de Antofagasta), en tercera instancia se seleccionan proyectos relacionados a cualquier otra empresa de la región de Antofagasta.

Esto no determina el aprendizaje activo (como bien se argumenta en [18]), pero la colaboración con empresas ofrece la posibilidad de realizar desarrollos con especificaciones abiertas, lo cual es de mucha utilidad para orientar el aprendizaje sobre casos basados en problemas.

Los alumnos que no asumen roles, se les asigna uno según sus potencialidades (habilidades para el desarrollo, manejo de bases de datos, etc.). El rol del líder del proyecto se deja a elección de los grupos.

Para guiar la construcción del software se usó SCRUM<sup>5</sup>. Las actividades del proyecto estuvieron

TABLA II  
ASPECTOS PARA LA SELECCIÓN DE ROL Y GRUPO DE TRABAJO.

Semana	Actividad/contenido		Peso en evaluación
1	Presentación, cátedra 1	Presentación del plan de evaluación y plan de trabajo, objetivos, planificación	-
2	Cátedras 3 y 4	Clase para proponer temas que cubran los objetivos, metodologías, proceso software	-
3	Cátedras 5 y 6,	Teoría de PBL, teoría de SCRUM	10%
		<b>Sprint Resultado esperado</b>	
4	<b>PBL</b>	1 Selección del tema de proyecto, plan de trabajo, configuración del espacio de trabajo	5%
5			
6			
7		2 Análisis y diseño de primer incremento, primera versión de interfaces, modelo de datos	10%
8			
9			
10	3 Programación, pruebas, documentación, actualización espacio de trabajo, implementación de base de datos. Validación con cliente	15%	
11			
12			
13	4 Análisis resultados incremento anterior, programación, pruebas, documentación, actualización espacio de trabajo. Validación con cliente	20%	
14			
15			
16			
17	5 Análisis resultados incremento anterior, programación, pruebas, documentación, actualización espacio de trabajo. Validación con cliente	15%	
18			
17	6 Preparación versión final, actualización plan mantenimiento, entrega al cliente y validación	10%	
18			
19	Evaluación Metodología proceso software	Presentación de resultados	15%
20		Presentación de resultados -rezagados	-
21	Examen recuperación		100%

guiadas por los objetivos de aprendizaje, orientadas al logro de las competencias de la asignatura, lograr retroalimentación activa y alcanzar lo planteado en el *product backlog*. El plan de trabajo incluyó incrementos de dos o tres semanas y presentación de avances en el plan de trabajo cada tres semanas, con las respectivas actualizaciones de la lista de productos del proyecto o *product backlog*. La ponderación de cada incremento está indicada en la tabla II.

Los grupos de trabajo se establecieron con un máximo de 5 integrantes, dos con rol de programador por grupo. En el curso 2014-2015 el trabajo se desarrolló con la participación de 29 alumnos inscritos en la asignatura: 5 grupos de 5 personas y un grupo de 4, con conocimientos diversos. Para el curso 2015-2016 la matrícula fue de 18 alumnos: 2 grupos de 5 personas y 2 grupos de 4 personas, igualmente con conocimientos diversos.

La planificación del PBL contempló 18 de las 22 semanas lectivas, lo que dio lugar a 5 *sprints* o iteraciones de 3 semanas y 1 de 2 semanas (el último), cada *sprint* o iteración con resultados reflejados en informes parciales. En [29] se indica que al inicio del trabajo en un proyecto, se debe plantear el objetivo deseado. Respecto al *sprint* en [29] (p. 15) se dice que "Cada ciclo de desarrollo o iteración (*sprint*) finaliza con un entregable de una parte operativa del producto (incremento)", siguiendo este postulado se han definido incrementos (*sprints*) y los resultados esperados en cada uno de ellos, como se describe en la tabla II.

Para facilitar el trabajo de cada grupo se usaron herramientas online de gestión de proyectos (como por ejemplo TeamWork Project ® o Trello ®<sup>6</sup>). Cada grupo generó un diagrama de Gantt con las actividades y tareas particularidades de cada proyecto.

En este plan de trabajo la evaluación del proyecto representa el 75% de la nota, el resto corresponde a evaluaciones en clases (de metodologías de desarrollo software, técnicas y teorías que sustentaban el proyecto). Para dar un valor numérico a la apreciación del docente y del cliente se usa la rúbrica detallada en la tabla III. La nota está en una escala del 1 al 7, siendo 4 la mínima de aprobación.

#### B. Aporte del método de trabajo en la UCN

El rendimiento de los alumnos en los dos últimos cursos fue mejor respecto a los dos cursos anteriores, parte de este resultado está basado en la participación activa que se materializó en la entrega de informes y revisión por parte del docente de estos avances (retroalimentación activa) en cada *sprint*, y a las presentaciones orales, a lo largo de los proyectos. Por otra parte el instrumento evaluativo da cuenta del uso de metodología para el proceso de desarrollo abordado por equipos de trabajo asumiendo roles específicos (esto relacionado con el logro del Obj2-UCN).

Al mismo tiempo los resultados de la tabla III dan cuenta de los logros de los aprendizajes traducido en niveles de aprobación. Para facilitar la comprensión de esta tabla, se describe a continuación su formato; la tabla

6 <https://twproject.com/>, <https://trello.com/> (última consulta: nov2016).

contrasta valores de participación de alumnos en dos cursos consecutivos (curso 2014-2015 y curso 2015-2016) en la asignatura IS-II según totales de alumnos inscritos, separados en tres segmentos que se definen como:

- Segmento de alumnos que participaron en el 100% de las actividades programadas, este segmento es identificado con la letra A. En la tabla se detalla para este segmento A la cantidad de alumnos que aprobaron tras participar en el 100% de las actividades.
- Segmento de alumnos con una participación mayor al 75% del total de las actividades programadas en el curso, este segmento es identificado con la letra B. Igualmente en la tabla se detalla para este segmento B la cantidad de alumnos que aprobaron tras participar en más del 75% de las actividades programadas.
- Segmento de alumnos con una participación menor al 75% de las actividades programadas. Este segmento es identificado con la letra C y al igual que los dos anteriores, en la tabla se detalla el total de alumnos aprobados para este segmento.

La columna denominada Ratios de la Tabla III contiene dos valores porcentuales ( $R_1$  y  $R_2$ ) para cada uno de los segmentos de la tabla que corresponden a las filas con las letras A, B y C, para cada curso. El primer valor ( $R_1$ ) se calcula según la fórmula:

$$R_1 = ((T_{\text{alum-part}} * 100) / T_{\text{alum}}) \quad (I)$$

donde:

$T_{\text{alum-part}}$  representa el total de alumnos que participaron en las actividades y  $T_{\text{alum}}$  representa el total de alumnos inscritos en la asignatura IS-II. Este ratio da cuenta del valor porcentual de participación del curso en las actividades programadas para un segmento. Por ejemplo, para el caso del segmento A del curso 2015-2016,  $R_{A1} = ((18 * 100) / 29) = 62,06 \approx 62$ , lo que significa que el ratio de participación es de.

El segundo valor ( $R_2$ ) se calcula según la fórmula:

$$R_2 = ((T_{\text{alum-aprob-part}} * 100) / T_{\text{alum-part}}) \quad (II)$$

donde:

$T_{\text{alum-aprob-part}}$  representa el total de alumnos que aprobaron en ese segmento y  $T_{\text{alum-part}}$  representa el total de alumnos que participaron en las actividades. Este ratio da cuenta del valor porcentual de aprobados para un segmento, es decir según su nivel de participación en las actividades programadas. Por ejemplo, para el caso del segmento A del curso 2015-2016,  $R_{A2} = ((17 * 100) / 18) = 94,43 \approx 94$ .

Los valores calculados con las fórmulas (I) y (II) se detallan en la columna Ratio de la tabla para cada uno de los segmentos, para ambos cursos. A partir de los valores porcentuales de aprobados ( $R_2$ ) de la tabla III, se ha calculado la media de alumnos aprobados en los dos últimos cursos, valor que se ubica en 72,3%; este valor para los dos cursos anteriores se mantenía cercano al

60%. Otro aspecto importante es que más del 63% de los inscritos realizaron todas las actividades planificadas y más del 74% trabajó las tres cuartas partes de lo planificado, es decir disminuyó la deserción que antes se ubicaba en 37%.

Un aspecto interesante de interés a destacar es el siguiente, cuanto mayor ha sido la participación de los alumnos mayor es la tasa de aprobación. Es decir, los porcentajes de aprobación de los alumnos que participaron en todas las actividades son los más altos (fila A-Tabla III) de todos los calculados.

Se fijó en 75% el mínimo de asistencia y participación en las actividades planificadas para no realizar actividades de recuperación (si el alumno no superaba el 75% presentaba el examen de recuperación como detalla la tabla II). El resultado de esto es que el 66% del rendimiento en los cursos se mantuvo en las filas A y B de la tabla III, lo que se relaciona con el logro de los objetivos Obj1-UCN y Obj2-UCN, este resultado también da cuenta del logro del Obj3-UCN, puesto que el 89% de los participantes mantuvo una participación del 75% en el curso. En contraste con lo anterior, los alumnos que mostraron poco interés y poca participación, tuvieron un rendimiento bajo y sólo aprobó el 30% como detalla la fila C (curso 2015-2016) de la tabla III.

De la Tabla III se desprende que los alumnos con mayor participación (ubicada en los segmentos A y B) lograron una aprobación mayor a los del segmento C, este resultado está incidido una relación fuerte sobre la nota de los aspectos: dedicación y participación activa. Es decir, cuando más responsable es la actitud del alumno y más participación activa tiene, mejores resultados de aprobación. Igualmente, se observa en la tabla III que los valores de participación ( $R_1$ ) más altos están en los segmentos A de ambos cursos, esto puede interpretarse como que el tipo de actividades programadas son motivadoras para los alumnos y el dinamismo del proyecto les lleva a mantenerse enganchados hasta el final del curso.

Respecto a los objetivos de la asignatura, se estima que los resultados de aprendizaje se alcanzan de forma más participativa con este método de trabajo, dado que los

TABLA III

INDICADORES DE PARTICIPACIÓN EN ACTIVIDADES DEL CURSO. POR CURSO, FILA A: PARTICIPACIÓN DEL 100% Y CANTIDAD DE ALUMNOS APROBADOS, FILA B: UNA PARTICIPACIÓN ENTRE 75% Y 99% Y CANTIDAD DE ALUMNOS APROBADOS Y LA FILA C CORRESPONDE A VALORES MENORES AL 75%.

Curso 2015-2016. Alumnos inscritos		29	Ratios
A	Participación del 100%	18	62%
	Cantidad de alumnos aprobados	17	94%
B	Participación mayor al 75%	8	27,5%
	Cantidad de alumnos aprobados	4	50%
C	Participación menor al 75%	3	10,3%
	Cantidad de alumnos aprobados	1	33,3%
Curso 2014-2015. Alumnos inscritos		18	
A	Participación del 100%	12	66,6%
	Cantidad de alumnos aprobados	11	91,6%
B	Participación mayor al 75%	4	22,2%
	Cantidad de alumnos aprobados	2	50%
C	Participación menor al 75%	2	11%
	Cantidad de alumnos aprobados	0	0

alumnos "de la era digital" aprenden más trabajando de forma colaborativa y asumiendo roles más activos.

### C. Experiencia en la UCM

La experiencia en la UCM consiste de dos clases de dos horas académicas por semana durante 18 semanas lectivas. La experiencia ha ido evolucionando, el uso de esta metodología activa cuenta con 5 cursos. Los objetivos con esta forma de trabajo se han afinado a medida que se obtiene experiencia; entre los más destacados se pueden mencionar:

- Obj1-UCM: Desarrollar soluciones informáticas de acuerdo a estándares y calidad de software con el fin de dar respuesta a necesidades de empresas regionales.
- Obj2-UCM: Disminuir el abandono en la asignatura con proyectos software reales que impactan en el tejido empresarial de la región.

En esta dinámica de trabajo los estudiantes enfrentan un proyecto pactado con un cliente real, en el cual la temática de contexto de desarrollo es de mediana complejidad. Se enfrentan generalmente problemas asociados a las áreas administrativas como control de bodegas, inventarios, cuentas a pagar o cuentas a cobrar. Aspectos favorables de considerar proyectos reales con empresas en la enseñanza de la ingeniería del software se aprecian en [25],[27],[28]. Sin embargo, plantean desafíos de precisar las competencias específicas de acuerdo al ciclo de vida a considerar o estándar de desarrollo a tener en cuenta.

La forma de seleccionar el proyecto y cliente es la siguiente: el docente posee un banco de empresas con necesidades software. Si el docente considera que los objetivos de aprendizaje se pueden lograr con un cliente (pyme), lo contacta y acuerda tanto el proyecto como el *product backlog*, que implica que el cliente asuma las responsabilidades de reunirse con los grupos de trabajo mínimo tres veces (una al principio, una a mediados del proyecto y una para las pruebas) además de asistir al evento final y participar en la valoración de la calidad de los proyectos. En caso contrario se selecciona una empresa relacionada con la UCM.

Generalmente las empresas son pymes de la región (Talca y alrededores) con el compromiso de implantar la solución al final del proyecto. Luego se inicia el proyecto con la conformación de equipos de hasta 6 integrantes por grupo y se asumen roles de acuerdo a la metodología de desarrollo que guiará el proyecto. Cada grupo es configurado como una pequeña empresa de desarrollo. Generalmente se logra conformar de 6 a 7 equipos.

En las primeras experiencias basadas en esta dinámica, se iniciaba con la elección de la metodología de desarrollo por cada equipo, a aplicar en función de la realidad del contexto del problema y entendimiento de los integrantes.

Sin embargo, posteriormente se optó por establecer que todos los equipos trabajasen con una misma metodología que se adaptara a equipos pequeños de desarrollo y contextos de poca complejidad. Esto permitió organizar de mejor manera las actividades de aprendizaje con miras

al logro de los aprendizajes y objetivos planteados en los proyectos software. La metodología que se ha utilizado de manera más sistemática es el modelo de referencia de MOPROSOFT<sup>7</sup>, que cubre la implementación del software bajo el estándar NMX-I-059-NYCE-2011.

El propósito de desarrollo de software es la realización sistemática de las actividades de análisis, diseño, construcción, integración y pruebas de productos de software nuevos cumpliendo con los requisitos especificados y con las normativas de seguridad de información.

El proceso de construcción del software se compone de uno o más ciclos de desarrollo. Al final de cada fase se debe generar un informe de control. Se dejó a elección de los grupos el uso de herramientas CASE para la gestión y desarrollo de los proyectos. En [26] se destaca la importancia de las herramientas CASE en las distintas fases del desarrollo de proyectos.

La planificación de trabajo contempla la presentación de avances cada 3 semanas. De acuerdo al marco conversacional, el rol docente se configura como gerente de proyectos y facilitador de material como plantillas tipo para registrar los avances o control de calidad de los resultados.

A partir de la semana 15 del cronograma se realizan presentaciones donde cada grupo debe generar evidencia de sus resultados del proyecto. La evaluación del proyecto representa el 60% de la nota que se realiza usando la rúbrica descrita en la tabla IV (en cada presentación de avances), el resto corresponde a evaluación de actividades en clase donde se estudian y discuten las técnicas que sustentan el proyecto.

### A. Aporte del método de trabajo en la UCM

El Obj1-UCM se alcanza cada vez que se involucran empresas en la experiencia docente, y por otra parte el modelo de referencia para abordar el proceso de desarrollo de software dio cuenta de las fases, roles y actividades mínimas que deben ser desarrolladas para alcanzar un estándar mínimo de calidad, lo que se validó con los resultados aprobados por los clientes participantes.

El método de trabajo ha permitido mejorar gradualmente el rendimiento de los alumnos en los últimos 5 cursos (Obj2-UCM). El motivo de implementar esta forma de trabajo fue precisamente el bajo rendimiento experimentado hasta el curso académico 2010-2011 donde fue del 55%. En las últimas versiones de esta forma de trabajo el rendimiento se ha mantenido en torno al 70% de aprobados y no existe una fluctuación significativa entre curso y curso, como sucedía en el pasado. La participación ha sido buena, incluso se ha observado competencia entre los grupos de cuál presenta el mejor poster, etc.

También el ausentismo ha disminuido, manteniéndose bajo el 30% (como valor medio que también aporta al logro del Obj2-UCM) en los últimos 5 cursos, en contraste con los registros anteriores al 2011 que oscilaban en torno al 45%. También es de destacar que muchos alumnos que han trabajado en esta dinámica,

7 <http://www.moprosoft.com.mx/> (última consulta: enero de 2017).

luego han realizado prácticas profesionales, trabajos de grado e incluso han terminado trabajando para empresas que han asumido el rol de cliente en esta asignatura (lo que contribuye con el logro del Obj1-UCM).

#### V.CONCLUSIONES

En este trabajo se reporta el uso de métodos activos para trabajar en combinación con modelos de ciclos de vida software en una asignatura (IS-II) de Ingeniería Civil Informática, en dos universidades de localidades diferentes de Chile. Las experiencias comparten un marco de trabajo similar y objetivos afines, pero que no fueron ideados de forma común ni al mismo momento.

En este documento se evidencia que el desarrollo de proyectos en el área de software y en el marco de metodologías activas, permite que los alumnos potencien conocimientos más profundos y los apliquen de forma práctica según el plan de trabajo de la asignatura IS-II. El elemento clave para que estas experiencias sean metodologías activas está en (1) que los desarrollos software se realizan para clientes reales (empresas de las regiones) bajo la supervisión y (2) que el docente, en un ambiente dinámico y gestionado con PBL mantiene el control de las experiencias y realiza la evaluación de los proyectos.

También permite acercar el mundo universitario (académico) al profesional o empresarial, dado que se trabaja con proyectos reales que responden a necesidades concretas de pequeñas y medianas empresas de dos Regiones de Chile.

En ambas universidades, antes de implementar estas técnicas didácticas, las clases de IS-II se impartían de forma tradicional: el docente realizaba clases magistrales, prácticas con casos propuestos por el docente y exámenes según una planificación concreta. El rol docente en ambas experiencias explicadas en este artículo ha implicado un seguimiento más efectivo de los aprendizajes esperados de los alumnos, a partir de una retroalimentación constante. Esto en concordancia con lo expuesto en el marco conversacional de Laurillard (2012), permitió validar que el uso de PBL contribuye y facilita la validación de las actividades de aprendizaje a través de los instrumentos de evaluación establecidos.

En este tipo de trabajo basado en la combinación de PBL y solución a necesidades de clientes, los alumnos se acercan a un problema real, con un cliente que necesita un producto válido en el mercado. En estas experiencias, tanto en la UCN como en la UCM, se ha comprobado que motiva a los alumnos a aprender y a usar recursos TIC vanguardista en las soluciones aportadas. Esto representa un cambio de paradigma de cómo impartir conocimientos según trabajos como [19], aplicado a la asignatura IS-II de ambas universidades. Además los alumnos ven la posibilidad de continuar trabajando con el cliente, lo que les motiva a mejorar continuamente el producto de sus proyectos más allá de los resultados académicos.

Los instrumentos evaluativos (tabla IV) han permitido dar cuenta del desarrollo de las habilidades transversales. Usar PBL y metodologías de desarrollo de proyectos software en el curso de IS-II en la UCM ha permitido reducir los índices de reprobados en más de un 15%.

TABLA IV

RÚBRICA PARA VALORACIÓN DEL DESEMPEÑO EN EL PROYECTO.

Nota	Informe	Actitud en el proyecto
excelente (6,5 - 7,0)	<b>A)</b> 100% escrito de forma técnica, comprensible, libre de faltas ortográficas, excelente redacción e ideas claras	100% comprometido con el logro del proyecto, aporta resultados según sus responsabilidades
	<b>B)</b> va de lo general al detalle, muestra evidencias formales, acordes con el trabajo de ingeniería realizado, evidencia uso de metodologías	Excelente relación con el cliente y el resto de las personas de su grupo, asume las responsabilidades de su rol (en un 100%)
	<b>C)</b> el trabajo evidencia relevancia para la empresa y para el grupo de trabajo y así queda plasmado en el documento	No aplica
bueno (5,5 - 6,4)	<b>D)</b> se estima que más del 80% del documento está escrito de forma técnica, 100% comprensible, menos de 5 faltas ortográficas, bien redactado cada apartado e ideas claras (generalmente)	100% comprometido con el logro del proyecto, aporta resultados según sus responsabilidades
	<b>E)</b> va de lo general al detalle, se estima que tiene un 75% de evidencias formales, en general está acordes con un trabajo de ingeniería (en un 80%), evidencia un 100% de uso de metodologías	Buena relación con el cliente y el resto de las personas de su grupo, asume las responsabilidades de su rol (en un 80%)
	<b>F)</b> el trabajo posee conclusiones relevantes para la empresa y para el grupo de trabajo y así queda plasmado en el documento	No aplica
regular (4,5 - 5,4)	<b>G)</b> se estima que más del 80% del documento está escrito de forma técnica, 100% comprensible, libre de faltas ortográficas, bien redactado cada apartado e ideas claras (generalmente)	Bien nivel de compromiso con el logro del proyecto (estimable en un 70%), aporta resultados según sus responsabilidades pero algunas veces "no a tiempo"
	<b>H)</b> va de lo general al detalle, muestra la mayor parte de las evidencias formales, en general está acordes con un trabajo de ingeniería, evidencia un uso de metodologías sobre el 60%	En general buena relación con el cliente (estimable en más de un 70%) y el resto de las personas de su grupo, asume las responsabilidades de su rol (en un 60%)
	<b>I)</b> el trabajo posee conclusiones de interés para la empresa y para el grupo y así queda plasmado en el documento	No aplica
deficiente (3,0 - 4,4)	<b>J)</b> Más del 80% del documento está escrito de forma técnica, 100% comprensible, libre de faltas ortográficas, bien redactado cada apartado e ideas claras (generalmente)	Poco nivel de compromiso con el logro del proyecto (menos del 70%), escaso aporte de resultados según sus responsabilidades, no entrega a tiempo lo asignado
	<b>K)</b> de lo general al detalle, muestra la mayor parte de las evidencias formales, en general está acordes con un trabajo de ingeniería, evidencia un uso de metodologías sobre el 60%	Relación con el cliente estimable en menos del 70% y el resto de las personas de su grupo, no asume como se espera las responsabilidades de su rol
	<b>L)</b> el trabajo posee conclusiones de interés para la empresa y para el grupo de trabajo y así queda plasmado en el documento	No aplica



También se logró una participación más activa de los alumnos y mayor asistencia a clases. De hecho la media de ausentismo ha disminuido cerca de un 16%, dato mantenido en los últimos tres años.

En cuanto a la UCN, esta combinación de PBL y metodologías de desarrollo de proyectos software también ha incidido positivamente en los resultados de la asignatura IS-II. En el curso 2014-2015, el 60% de los prototipos funcionaron bien y el cliente seleccionó el mejor para probarlo en el departamento de bienestar estudiantil de la UCN y el curso 2015-2016 el 75% de los proyectos llegaron a término. Estos resultados han motivado a otros docentes para aplicar PBL en sus asignaturas. Particularmente, para el momento de terminar este artículo, el método se implementó en la UCN con un proyecto transversal para las asignaturas Ingeniería del Software I, Ingeniería del Software II y Base de Datos en esta misma carrera. Aún no se cuenta con resultados de esta práctica pero los alumnos de las tres asignaturas están trabajando con mucha motivación.[1], [2].

En cuanto a las mejoras, una destacable es usar los mismos casos de estudio en cursos siguientes, para que los nuevos alumnos puedan tener como entrada los resultados anteriores (prototipos, planes de trabajo, etc.). Esto podría tener un doble beneficio, primero validar la calidad de las situaciones de aprendizaje con más de un curso y segundo verificar la efectividad del caso de estudio y el acompañamiento del estudiante.

Otra mejora es recabar las opiniones de los alumnos con alguna técnica directa como encuestas, para conocer más a fondo lo que quieren los alumnos y cómo perciben la forma de trabajo. Esto ayudaría no sólo en encontrar mejoras en IS-II, sino también adecuar el método a otras asignaturas con objetivos de aprendizaje similares.

Respecto al futuro de la práctica docente, con los resultados descritos se puede pensar que es pertinente seguir aplicando esta práctica. Para ello es necesario considerar aspectos como: mantener una base de datos de clientes potenciales, revisar continuamente los avances en herramientas CASE, metodologías y técnicas de Ingeniería del Software, para poder crear proyectos que se adecúen a los avances y disponibilidad de herramientas TIC.

#### AGRADECIMIENTOS

Los autores agradecen la colaboración de las empresas pymes de la región del Maule, Chile, por la disposición a apoyar el desarrollo de proyectos software y ejercicio profesional de los alumnos de la UCM, durante estos años. También agradecen a las empresas que decidieron dar apoyo a esta iniciativa en la ciudad de Antofagasta, Chile, gracias a este apoyo los alumnos de la UCN lograron desarrollar un proyecto usando PBL y poner en práctica metodologías y técnicas de Ingeniería del Software en un entorno real.

#### REFERENCIAS

[1] M. A. Salido, A. Giret, M. Abril López, "Aplicación de Metodologías Activas en la Enseñanza de Informática en la Licenciatura en Matemáticas". IEEE-RITA, vol.2, pp. 73-78, 2007.

- [2] D. G. Lamar, P. F. Miaja, M. Arias, A. Rodriguez, M. Rodriguez, A. Vazquez, J. Sebastian, "Experiences in the Application of Project-Based Learning in a Switching-Mode Power Supplies Course". IEEE Transactions on Education, vol. 55, pp. 69-77, 2012.
- [3] D. Laurillard, "Teaching as a Design Science: Building Pedagogical Patterns for Learning and Technology", New York: Routledge, Taylor & Francis Group, 2012.
- [4] J. A. González, "La clasificación de los métodos de enseñanza en educación superior", Contextos educativos: Revista de educación, vol. 15, pp. 93-106, 2012.
- [5] P. Jonnaert, J. Masciotra, D. Yaya, "Revisión de la competencia como organizadora de los programas de formación: hacia un desempeño competente", Observatorio de Reformas Educativas. Universidad de Quebec, Montreal, BIE/UNESCO, 2006.
- [6] D.C. Kayes, D.A. Kolb, "Experiential learning in teams", Simulation and Gaming, vol.36, no.3, pp. 330-354, 2005.
- [7] V. Flores, P. Lara, M.C. Gaya, "Project Based Engineering School, Una escuela conectada con un mundo profesional sostenible", X Jornadas Internacionales de Innovación Universitaria", pp. 476-484, Villaviciosa de Odón-España, 2013.
- [8] A. Kolmos, "Changing the curriculum to problem based and Project-based learning", In Yusof, K. M., Azli, N. A., Kosnin, A. M., Yusof, S. K., & Yusof, Y. M. (Eds). Outcome-Based Science, Technology, Engineering, and Mathematics Education: Innovative Practices. Hershey, PA: IGI Global, pp. 50-61, 2012.
- [9] M.C. Gaya, M.J. García, F. Aparicio, V. Flores, P. Lara, P. Enrique, D. Atauri, "Grado en Ingeniería Informática: utilizando ABP", X Jornadas Internacionales de Innovación Universitaria, pp. 476-484, Villaviciosa de Odón-España, 2013.
- [10] B. Barron, "Doing with understanding: Lessons from research on problem- and project-based learning", Journal of the Learning Sciences, vol.7, no. 3, pp. 271-311, 1998.
- [11] S. Boss, J. Krauss, "Reinventing project-based learning: Your field guide to real-world projects in the digital age", Washington DC: International Society for Technology in Education (ISTE), pp. 47-48, 2007.
- [12] M. Armenta, Salinas V., "Aplicación de la técnica educativa aprendizaje basado en problemas para capacitación a distancia (e-learning)", Revista Iberoamericana de Educación a Distancia (RIED), vol. 16, pp. 57-83, 2013.
- [13] X. Mendoza, Bernabeu M. "Aprendizaje Basado en Problemas competencias del profesional de la salud", Revista Innovación Educativa, Instituto Politécnico Nacional de México, vol. 6, no. 35, pp. 1-12, 2006.
- [14] A. Bari, AD Corebima, "The contribution of learning motivation and metacognitive skill on cognitive learning outcome of students within different learning strategies", Journal of Baltic Science Education, vol. 14, no. 4, pp. 487-500, 2015.
- [15] C. Tosun, E. Senocak, "The effects of problem-based learning on metacognitive awareness and attitudes toward chemistry of prospective teachers with different academic backgrounds," Aust. J. Teach. Educ., vol. 38, no. 3, pp. 61-73, 2014.
- [16] Sp. Husamah, "Blended Project Based Learning: Metacognitive Awareness of Biology Education New Students", Journal of Education and Learning, vol. 9, no. 4, pp. 274-281, 2015.
- [17] A.M. Ehuleche, A. De Stefano, "Evaluación de las competencias para la formación de tutores de e-learning", Revista Iberoamericana de Educación a Distancia (RIED), vol. 14, no. 1, pp. 75-86, 2011.
- [18] M. Prince, "Does Active Learning Work?", Journal of Engineering Education, vol. 93, no. 3, pp. 223-231, 2004.
- [19] P.C. Blumenfeld, E. Soloway, R.W. Marx, J.S. Krajcik, M. Guzdial, A. Palincsar, "Motivating project-based learning: Sustaining the doing supporting the learning". Educational Psychologist, vol. 26, no. 3, pp. 369-398, 1991.
- [20] R. Grimes, "Problem-based learning and legal education-a case study in integrated experiential study". Revista de docencia universitaria (REDU), vol. 13, no.1, pp. 361-375, 2015.
- [21] W. He, X.H. Yuan, L. Yang, "Supporting Case-based Learning in Information Security with Web-Based Technology", Journal of Information Systems Education, vol. 24, no. 1, pp. 31-40, 2013.
- [22] E. Exposito, "yPBL methodology: a problem-based learning method applied to Software Engineering", IEEE EDUCON Education Engineering, pp. 1577-1583, 2010.
- [23] S. H. Barrows, R.M. Tamblyn, "Problem-Based Learning: An Approach to Medical Education", Springer Publishing Company. NY, 1980.

- [24] A. Hernández, R. Lacuesta, "Aplicación del Aprendizaje Basado en Problemas (PBL) bajo un enfoque multidisciplinar: una experiencia práctica", En J. C. Ayala (Coord.), Conocimiento, innovación y emprendedores: Camino al futuro, pp. 30-43, Rioja: Universidad de La Rioja. 2007.
- [25] R. Anaya, "Una visión de la enseñanza de la Ingeniería de Software como apoyo al mejoramiento de las empresas de software", Revista Universidad EAFIT, vol. 42, no. 141, pp. 60-76, 2012.
- [26] J. Baik, "The effects of CASE tools on software development effort," ProQuest Diss. Theses, no. December, p. 152, 2000.
- [27] L. E. Sánchez, D. García, C. Blanco, E. Fernández-Medina, M. Piattini. "Papel de las certificaciones profesionales en la enseñanza universitaria de ingeniería de software en España". Revista Española de Innovación, Calidad e Ingeniería del Software, vol. 6, no. 2, pp. 6-24, 2010.
- [28] J.M. Blanco, I. Usandizaga, A. Jaime. "Gestión de Proyectos en el Grado en Ingeniería Informática: del PBL a la espiral de proyectos", ReVisión, vol. 7, no. 3, 2014.
- [29] R. Herranz, "Scrum Manager". Accessed oct 2016. [Online]. Available [http://www.scrummanager.net/files/sm\\_proyecto\\_apuntes\\_12.pdf](http://www.scrummanager.net/files/sm_proyecto_apuntes_12.pdf).
- [30] MINEDUC (Chile), "Cuenta Pública 2015". Accessed oct 2016. [Online]. Available [http://www.gob.cl/cuenta-publica/2015/sectorial/2015\\_sectorial\\_ministerio-educacion.pdf](http://www.gob.cl/cuenta-publica/2015/sectorial/2015_sectorial_ministerio-educacion.pdf).

**Víctor M. Flores Fonseca** es Académico de la Universidad Católica del Norte (UCN) desde 2014. Antes fue Profesor Contratado Doctor en la Universidad Europea de Madrid, desarrollándose como docente en el área de software e investigador en Sistemas Inteligentes. Previamente fue investigador en formación (becario FPI) entre 2004 y 2008 en la Universidad Politécnica de Madrid (UPM). Víctor es Doctor en Informática por la UPM, Master en Ingeniería del Software por la UPM e Ingeniero en Informática por la Universidad Politécnica Santiago Mariño (Venezuela). Dentro de la UCN, sus trabajos de investigación se centran en el área de Ingeniería del Software e Inteligencia Artificial y dentro del área docente, entre sus intereses está el usar metodologías activas de enseñanza-aprendizaje y conectar la empresa con la universidad.

**Yessica Gómez** es Académica de la Universidad Católica del Maule (UCM) desde 2005 en el área de Ingeniería del Software y coordinadora docente en el Centro de Formación Técnica San Agustín Talca desde el 2015. Yessica es Ingeniero Ejecución en Computación de la Universidad de Santiago, Magister en Tecnologías de la Información de Universidad Técnico Federico Santa María y Magister en Educación Basada en Competencias de la Universidad de Talca. Entre sus intereses docentes está el usar metodologías activas de enseñanza-aprendizaje.