

Modelo de características de una línea de productos de software para aplicaciones que usan dispositivos de Reconocimiento Biométrico

Edwin J. Muñoz Riascos*, Luisa Rincón†

Acople Tecnológico, Cali, Colombia.

Email: edwin.munoz@acopletecnologico.com

†Departamento de Electrónica y Ciencias de la Computación, Pontificia Universidad Javeriana, Cali, Colombia.

Email: lfrincon@javerianacali.edu.co

Resumen—Las líneas de productos de software ofrecen beneficios como mejorar los tiempos de desarrollo y aumentar la calidad del producto final. La empresa Acople Tecnológico ha empezado a adoptar este paradigma y utiliza el enfoque extractivo para la definición de una línea de productos de software para las aplicaciones que usan dispositivos de reconocimiento biométrico que ha desarrollado. En este artículo se presenta y analiza el modelo de características que se ha extraído de las aplicaciones ya desarrolladas por la empresa y se discuten las líneas de trabajo futuras.

Keywords: *Líneas de productos de software, modelo de características, biometría*

I. INTRODUCCION

Las Líneas de Productos de Software (SPL, por su traducción en inglés.) son un paradigma de desarrollo de software que busca crear familias de productos en lugar de crear productos individuales. Estas familias agrupan un conjunto de productos que comparten características comunes a la vez que tienen algunas variaciones entre sí. Las líneas de productos de software permiten capitalizar el esfuerzo de desarrollo no sólo en un producto, sino en muchos productos, a la vez que se gestionan eficientemente las variaciones de los mismos. Las SPL prometen beneficios como la reutilización, disminución de los tiempos y costos de desarrollo[1].

Acople Tecnológico SAS, es una empresa especializada en desarrollo de software. La empresa dispone de un portafolio de productos de software que usan dispositivos de reconocimiento biométrico facial y dactilar. Sin embargo, aunque estos productos gozan de buena popularidad en el ámbito empresarial, Acople Tecnológico los desarrolla, prueba, y mantiene por separado, si bien los productos tienen una cantidad significativa de funcionalidades y código en común. Este enfoque, centrado en administrar muchos productos y hacer reutilización no planificada, desperdicia recursos técnicos e implica una inversión alta en tiempo y costo.

Debido a los beneficios que la adopción de las SPL promete, se considera de interés adoptar en la empresa Acople Tecnológico el paradigma de las líneas de productos de software para las aplicaciones que usan dispositivos de reconocimiento biométrico facial. Para ello, como punto de

partida se han analizado los requerimientos de tres de las aplicaciones desarrolladas por la empresa para representar con esta información el modelo de variabilidad que representa una línea de productos de software para la empresa. En este artículo se presenta el modelo de variabilidad obtenido y se discuten los aspectos futuros de la investigación en curso.

La organización de este artículo es la siguiente: en la Sección II se presentan los conceptos preliminares de las SPL y se presentan las aplicaciones de referencia a partir de las cuales se está desarrollando la línea de productos de software. Luego, en la sección III se describe el modelo de características propuesto para la SPL de la empresa Acople. En la sección IV se discuten los resultados obtenidos al analizar el modelo de características. En la sección V se presenta algunos trabajos relacionados con nuestra propuesta. Finalmente, en la sección VI se presentan las conclusiones y el trabajo futuro.

II. CONCEPTOS PRELIMINARES

A. Biometría

El concepto biometría proviene de las palabras Bio (Vida) y Metría (Medida). Esta área del conocimiento permite por medio de métodos automáticos analizar determinadas características humanas con el fin de identificar y autenticar a las personas. Las características pueden ser tanto físicas, como por ejemplo la huella dactilar, el iris, el rostro, la geometría de la mano, etc.; como del comportamiento humano, por ejemplo la voz, la firma, la forma de caminar, etc. por lo tanto, con ello se infiere que todo equipo biométrico mide e identifica alguna característica propia de la persona [2].

Actualmente existen dispositivos de reconocimiento biométrico para detectar todas las características antes mencionadas. Los equipos de reconocimiento biométrico más populares son los que evalúan las huellas dactilares, seguidos de los que llevan a cabo reconocimiento biométrico facial.

B. Modelo de características

Las líneas de productos de software se representan por medio de modelos. Una de las notaciones que existen para expresar estos modelos es la notación de los modelos de características [3]. Los modelos de características especifican el conjunto de productos válidos que pueden obtenerse de una

SPL [4]. Estos modelos se diseñan en la primera fase de la línea de productos y por lo tanto juegan un papel central en todas las fases del desarrollo de la SPL.

Una característica es un rasgo o elemento distintivo percibido por el usuario final que representa aspectos relevantes de un software. Las características se usan en las SPL para especificar y comunicar aspectos comunes y variables de la línea de productos entre las partes interesadas [4]. Las características se relacionan entre sí con dependencias, éstas pueden ser *obligatorias*, *opcionales*, *de cardinalidad grupal*, *de inclusión o de exclusión* [4]. A continuación se explican cada una de ellas:

Obligatoria: se representa con una línea cuyo origen es la característica padre y cuyo destino formado por un punto oscuro es la característica hija. Esta dependencia indica que la característica hija debe ser incluida en los productos que tengan seleccionada su característica padre y viceversa.

Opcional: se representa con una línea cuyo origen es la característica padre y cuyo destino formado por un punto blanco es la característica hija. Esta dependencia indica que la característica hija puede ser o no incluida en los productos que tengan su característica padre. Si la característica hija es incluida entonces su característica padre también debe ser incluida en el mismo producto.

Cardinalidad grupal: [mínimo, máximo] representa el número mínimo y máximo de características que el producto puede tener cuando la característica padre es incluida. Si al menos una característica hija es incluida entonces la característica padre también debe ser incluida en el producto. Una dependencia de cardinalidad grupal une a la característica *Aplicación* con sus características hijas, pues permite escoger una, dos o las tres características entre *TiempoLaborado*, *Acceso* y *Casino*.

De exclusión: se representa con una flecha punteada bidireccional que relaciona dos características. Indica que las características relacionadas no pueden estar juntas en ningún producto derivado de la SPL.

De inclusión: Se representa con una flecha punteada unidireccional. El origen de la flecha es la característica que requiere y el destino es la característica requerida. Esta dependencia indica que la característica requerida debe ser incluida en todos los productos que tengan la característica de la requiere.

Las dependencias de tipo *obligatoria* y *opcional* son conocidas como dependencias estructurales, mientras que las dependencias de tipo *inclusión* y *exclusión* son conocidas como dependencias transversales [5]. La Figura 1 presenta el modelo de características que se está desarrollando para la línea de productos de la empresa Acople Tecnológico. Este modelo de características se explicará en detalle en la siguiente sección.

C. Enfoque para desarrollar Líneas de Productos de Software (SPL)

Existen tres enfoques para adoptar líneas de productos de software: *el enfoque proactivo*, *el enfoque reactivo* y *el enfoque extractivo* [1]. El *enfoque extractivo* permite iniciar una línea de productos a partir de aplicaciones ya desarrolladas por

medio de ajustes sucesivos que adapten los desarrollos previos hacia el desarrollo orientado a SPL. Este enfoque es útil cuando las empresas cuentan con pocos recursos para empezar desde cero el desarrollo de sus líneas de productos y tienen ya desarrolladas de manera independiente aplicaciones que comparten elementos comunes [6]. El *enfoque reactivo* es útil cuando una organización que ya tiene desarrollada una SPL, desea incorporarle a esa SPL nuevas características que le permitan soportar nuevos requerimientos y generar nuevos productos. Entonces se realiza un proceso de reestructuración de esa SPL que permita generar los productos actuales más los nuevos productos de acuerdo a nuevos requerimientos o necesidades [6]. El *enfoque proactivo* es útil cuando la organización no posee los productos ni una SPL que sirva como base para generarlos. Entonces la organización debe destinar recursos importantes que permitan analizar, diseñar e implementar toda SPL deseada. Este enfoque puede ser el de mayor costo en recursos pues su implementación requiere cubrir todas las fases de un proceso normal de desarrollo de software y porque se empieza desde cero [6].

III. SPL DE ACOPLE TECNOLÓGICO

En el desarrollo de la SPL de Acople Tecnológico se ha trabajado con el enfoque extractivo, pues ésta era la alternativa que más se ajustaba a las condiciones de la empresa. En esta sección, se explican las aplicaciones que se analizaron para extraer el modelo de características (Sección A), se presenta el modelo de características resultante (Sección B) y se describen los resultados de verificar si el modelo propuesto tenía defectos sintácticos o semánticos (Sección C).

A. Aplicaciones que usan reconocimiento biométrico analizadas

Las aplicaciones desarrolladas por Acople Tecnológico que usan dispositivos de reconocimiento biométrico y fueron analizadas son: *control de tiempo laborado*, *control de acceso* y *control de casinos*. A continuación se describe cada una de ellas.

- *control de tiempo laborado*: registra para cada persona el momento inicial y final de la jornada diaria de trabajo y calcula el tiempo laborado en cada jornada de trabajo. El tiempo se discrimina en los conceptos de tiempo que se relacionan con el pago de la nómina como: horas ordinarias, recargos nocturnos, extras, festivos, etc.
- *control de acceso*: registra la hora en la que las personas entran y salen de un lugar con acceso restringido.
- *control de casinos*: identifica a las personas que hacen uso de servicios de alimentación de un restaurante empresarial, para luego generar órdenes de cobro por los servicios utilizados.

Todas las aplicaciones involucran dispositivos de reconocimiento biométrico e involucran características como reportes, gestión de usuarios y personas, entre otras.

B. Modelo de características propuesto

La Figura 1 representa el modelo de características propuesto para representar la línea de productos de software de

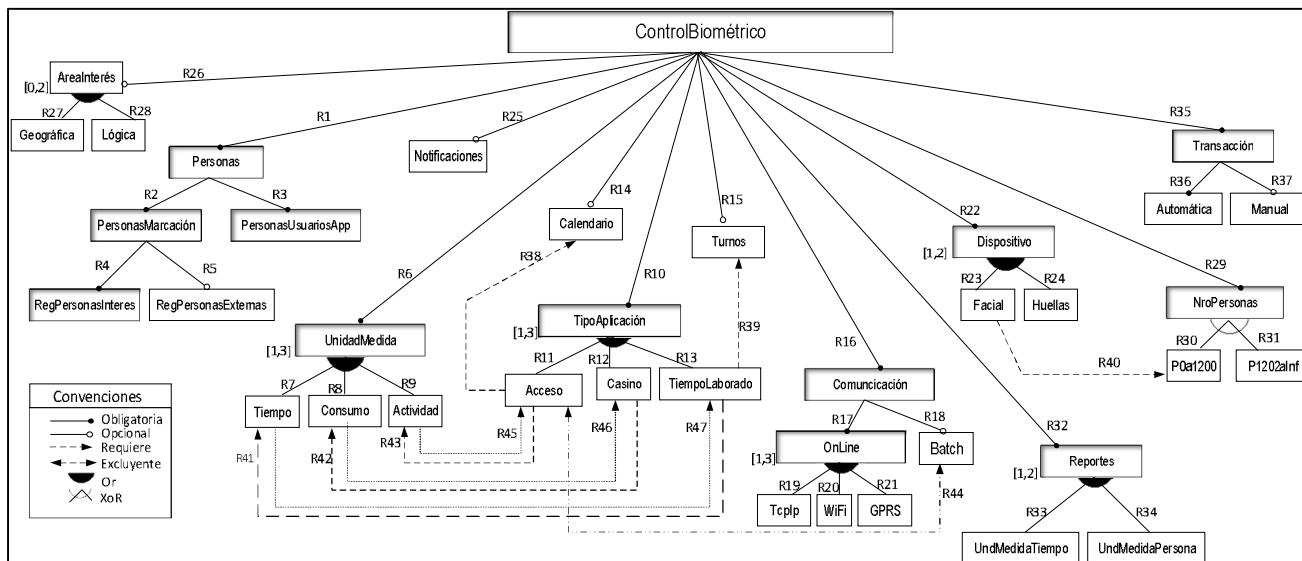


Fig. 1. Modelo de característica propuesto de una línea de productos de software para aplicaciones que usan dispositivos de Reconocimiento Biométrico

aplicaciones de reconocimiento biométrico. Este modelo de características se obtuvo luego de realizar un proceso iterativo de refinamiento. Así, inicialmente fue elaborado un modelo de características para cada una de las aplicaciones. Luego, cada uno de estos modelos fue examinado con el fin de identificar en ellos características y dependencias comunes. Después se identificaron características que pudieran ser agrupadas en características más generales. Así por ejemplo, las características *Tiempo*, *Consumo* y *Actividad* se encontraban cada una en un modelo de características diferente, y fueron agrupadas como hijas de la característica *Unidad de Medida* en el modelo de características propuesto.

Como resultado del análisis se obtuvieron las características obligatorias *TipoAplicación*, *Personas*, *Comunicación*, *Dispositivo*, *Reportes*, *Nro Personas* y *Transacción*. Las características opcionales *ArealInterés*, *Calendario*, *Notificaciones*, *Turnos*, *RegPersonasExternas*, *Batch*, *P1202aInf*. Las características agrupadas por una dependencia de cardinalidad grupal: *ArealInterés*, *UnidadMedida*, *TipoAplicación*, *Dispositivo*, *Reportes*, *OnLine*.

La característica *TipoAplicación* hace referencia a la funcionalidad principal de las aplicaciones analizadas para construir la SPL. En este caso, las operaciones disponibles son *TiempoLaborado*, *Acceso* y *Casino*. Cualquier producto que se configure del modelo de características deberá tener al menos un *Tipo de Transacción* definido. La característica *Personas* se relaciona con la gestión de las personas a las que se les registran marcaciones en los dispositivos biométricos (*PersonasMarcación*), y con la gestión de los usuarios administradores de la aplicación (*PersonasUsuarios.App*). Los usuarios de la aplicación son quienes pueden generar los *Reportes*. La característica *Comunicación*, representa la forma como es transferida la información capturada por el dispositivo biométrico hacia la aplicación. La característica *Batch* hija de *Comunicación* se asocia con el procesamiento de archivos. La característica *Dispositivo* hace referencia a los diferentes tipos de dispositivos biométricos que pueden usar las

aplicaciones. Esos dispositivos por el momento son dos, pero podrían ser de muchos tipos: faciales, de huellas dactilares, o de reconocimiento de iris. La característica padre *NroPersonas* representa el número máximo de personas que el dispositivo de reconocimiento biométrico podría diferenciar. Esta característica tiene a su vez dos características hijas *POa1200* *P1202aInfinito* de las cuales para un producto el usuario sólo podrá seleccionar una. Los dispositivos de reconocimiento biométrico facial que maneja Acople Tecnológico soportan máximo 1200 personas, por lo tanto, existe una relación de inclusión (R40) entre la característica *Facial* y la característica *POa1200*.

La característica *Transacción* indica la forma como se registra de un nuevo evento en la aplicación. Si la transacción es *Automática*, entonces los datos provienen directamente del dispositivo biométrico; de lo contrario proviene de algún proceso manual. La característica *Transacción* y la dependencia R37 indican si existirá alguna opción en la aplicación que permita el registro manual de los datos que debería ser capturados por un dispositivo biométrico. La característica *UnidadMedida* hace referencia a qué mide, calcula o controla la aplicación. Así por ejemplo, la característica *consumo* se podría utilizar en aplicaciones que controlen el consumo o manejo de productos o servicios. Si es *actividad* se puede abstraer para ser utilizada en aplicaciones de tipo transaccional como por ejemplo la compra de un bien o servicio. La característica *Tiempo* tiene una gran variedad de usos como: cálculo de tiempo laborado, tiempos de entregas de productos, etc. El usuario podría escoger entre 1,2 o los 3 tipos de unidades disponibles. La característica *Notificaciones* se encarga de notificar a los usuarios de la aplicación y a las personas interesadas eventos sobre condiciones particulares del sistema. La característica *Calendario* permitirá organizar y gestionar lo referente a calendarios incluyendo *días festivos*. La característica *Calendario* y la dependencia R38 se relacionan para indicar que las aplicaciones con características *Acceso* requieren que esté presente la característica *Calendario*. Existe

además una relación de tipo requerido entre las características de *TiempoLaborado* y *Turnos*. En efecto, si en un producto se selecciona la aplicación de tipo *control de tiempo*, entonces se requiere de la característica que permite gestionar los *Turnos* del personal.

C. Verificación del modelo de características

Verificar el modelo de características implica detectar posibles *defectos sintácticos* y *semánticos* que pueden surgir al modelar la línea de productos de software.

De una parte los *defectos sintácticos* se producen cuando el modelo de características incumple las reglas de la notación, como por ejemplo tener dos características raíz o tener características sin dependencias [8]. Para evitar este tipo de errores, el modelo de características se construyó en la plataforma de SPLOT¹ que sólo permite construir modelos de características que se adoptan correctamente a la notación. El modelo de características fue guardado en el repositorio de SPLOT con el nombre de “Control Biométrico” y se encuentra disponible en Internet². La Figura 2 muestra los detalles del modelo almacenado.

Fig 2. Información modelo de características creado. Portal SPLOT

De otra parte los *defectos semánticos* son imperfecciones que afectan la capacidad del modelo de características para representar de manera correcta el dominio de la línea de productos [9]. Algunos de estos defectos semánticos son: *características muertas, modelos vacíos, falsos modelos de líneas de productos, características falsas opcionales y dependencias redundantes*. Una descripción detallada de cada tipo de defecto se encuentra disponible en el trabajo de Salinesi y Mazo [9].

El modelo propuesto fue analizado con la herramienta VariaMos [10] para detectar si tenía defectos semánticos. Los resultados del análisis indicaban que el modelo propuesto no tiene ninguno de los defectos semánticos considerados en VariaMos.

¹ <http://www.splot-research.org>

² http://gsd.uwaterloo.ca:8088/SPLOT/models/temp_models/model_20141027_1208747069.xml

IV. ANÁLISIS DEL MODELO DE CARACTERÍSTICAS PROPUESTO

El portal de SPLOT permite llevar a cabo operaciones de análisis sobre los modelos de características. Así los análisis del modelo propuesto con SPLOT (ver Figura 3) indican que el modelo tiene 13 dependencias obligatorias, 7 dependencias opcionales, 7 dependencias grupales (or group – xor group) y 13 dependencias transversales. Estas dependencias representan las restricciones que deben cumplir cualquier producto que se derive de la línea de productos de software que se está modelando. Adicionalmente, usando SPLOT es posible obtener el número de configuraciones posibles de la línea de productos, es decir el número de productos diferentes que se podrían obtener a partir de un modelo que representa la línea de productos. En el caso del modelo de características propuesto, podrían crearse 22.848 productos diferentes.

Obtener 22.848 productos diferentes a partir de características que se encontraban en sólo tres productos de software muestra que las líneas de productos de software facilitan la planificación de la reutilización. Sin embargo, es necesario considerar cuáles de estas 22.848 potenciales aplicaciones serían de interés en el mercado. Así de acuerdo a la experiencia de la empresa se considera que la característica *TipoAplicación* podría determinar algunos productos principales dentro de la línea de productos de software. Estos productos serían aquellos que surgen de a combinación de las características de *TipoAplicación*. Las combinaciones posibles son:

1. *Control de tiempo laborado*
2. *Control de acceso*
3. *Control de casino*
4. *Control de tiempo laborado + acceso*
5. *Control de tiempo laborado + casino*
6. *Control de Acceso + Casino*
7. *Control de Tiempo laborado + Acceso + Casino*

Esta clasificación es interesante para establecer un esquema de licenciamiento para Acople Tecnológico, como por ejemplo versiones *básica, pyme, empresarial y corporativa* de acuerdo a las características que tenga cada producto. Así, un producto que tiene sólo *Control de tiempo laborado* sería una aplicación básica y un producto con *Control de Tiempo laborado + Acceso + Casino* sería un producto con licencia corporativa.

Feature Model Statistics	
#Features	38
#Mandatory	13
#Optional	7
#XOR groups	1
#OR groups	6
#Grouped	17
#Cross-Tree Constraints (CTC)	13

Fig. 3. Estadísticas del Modelo de características

V. TRABAJOS RELACIONADOS

Diferentes empresas de la industria han adoptado el paradigma de desarrollo de SPL, prueba de esto es el sitio web destinado a presentar el “Hall de la fama”³ de los mejores casos industriales documentados sobre este paradigma. De hecho, empresas como Nokia [11] y Hewlett Packard [12] han utilizado exitosamente las SPL para la gestión de celulares y el firmware para impresoras respectivamente. Adicionalmente en Colombia la empresa *Heinsohn Business Technology* ha adelantado algunos trabajos en la adopción de este paradigma [13], [14].

VI. CONCLUSIONES Y TRABAJOS FUTUROS

Este artículo presentó el propósito de adoptar una SPL en la empresa Acople Tecnológico usando el método extractivo. Para ello se analizaron 3 aplicaciones ya desarrolladas por la empresa para, a partir de ellas, modelar la línea de productos de software por medio de un modelo de características.

Abstractar los requerimientos funcionales de las aplicaciones fue de gran ayuda pues permitió que el modelo de características propuesto considerará aspectos comunes a varias aplicaciones, a la vez que identificará puntos de variación entre ellas. Este análisis es interesante no sólo a nivel conceptual, sino que también ofrece elementos para pensar en la implementación de la línea de productos.

De hecho, el trabajo de investigación actual está enfocado en la implementación de esta línea de productos de software. Para ello se ha direccionado la investigación a identificar la mejor forma de materializar las características identificadas en componentes de software reutilizables, con el fin de aprovechar hasta donde sea posible, los artefactos de código ya desarrollados por Acople Tecnológico.

REFERENCIAS

- [1] P. Clements and L. M. Northrop, *Software Product Lines: Practices and Patterns*, 1st ed. Addison-Wesley Professional, 2001.
- [2] M. Tapiador, J. A. Sigüenza, and A. García, *Tecnologías biométricas aplicadas a la seguridad*. Ra-Ma, 2005, p. 440.
- [3] K. C. Kang, S. G. Cohen, J. A. Hess, W. E. Novak, and S. P. Peterson, “Feasibility Study Feature-Oriented Domain Analysis (FODA). Technical Report,” 1990.
- [4] S. Apel, D. Batory, C. Kästner, and G. Saake, *Feature-Oriented Software Product Lines*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.
- [5] K. Czarnecki, S. Helsen, and U. Eisenecker, “Formalizing Cardinality-based Feature Models and their Specialization,” *Softw. Process Improv. Pract.*, vol. 10, no. 1, pp. 7–29, 2005.
- [6] C. W. Krueger, “Easing the Transition to Software Mass Customization”, *Software Product-Family Engineering*, Springer, 2002, pp.282-293.
- [7] “SPLOT website.” [Online]. Available: <http://www.splot-research.org/>.
- [8] R. Mazo, R. Lopez-Herrejon, C. Salinesi, D. Diaz, and A. Egyed, “Conformance Checking with Constraint Logic Programming: The Case of Feature Models,” in *Proceedings of the 35th Annual International Computer Software and Applications Conference (COMPSAC)*, 2011, pp. 456–465.
- [9] L. F. Rincón Perez, G. L. Giraldo Gómez, R. Mazo, C. Salinesi, and D. Diaz, “Subconjuntos Mínimos de Corrección para explicar características muertas en Modelos de Líneas de Productos. El caso de los Modelos de Características,” in *Proceedings of the 8th Colombian Computer Conference (CCC)*, 2013.
- [10] R. Mazo, C. Salinesi, and D. Diaz, “VariaMos : a Tool for Product Line Driven Systems,” in *Proceedings of the 24th International Conference on Advanced Information Systems Engineering (CAiSE Forum'12)*, 2012, no. June, pp. 25–29.
- [11] F. J. van der Linden, K. Schmid, and E. Rommes, *Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.
- [12] P. Toft, D. Coleman, and J. Ohta, “A cooperative model for cross-divisional product development for a software product line,” in *Proceedings of the first conference on software product lines: experience and research directions*, 2000, pp. 111–132.
- [13] C. Parra, D. Joya, L. Giral, and A. Infante, “An SOA Approach for Automating Software Product Line Adoption,” in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, 2014, pp. 1231–1238.
- [14] C. Parra, L. Giral, A. Infante, and C. Cortés, “Extractive SPL Adoption Using Multi-level Variability Modeling,” in *Proceedings of the 16th International Software Product Line Conference - Volume 2*, 2012, pp. 99–106.

³ <http://splc.net/fame.html>