

# Proposta de Uma Linha de Produtos de Software para Sistemas de Aprendizagem Ubíquos com Internet das Coisas

Ricardo Fuzeto, Rosana T. V. Braga

<sup>1</sup>Instituto de Ciências Matemáticas e Computação – Universidade de São Paulo (USP)  
São Carlos – SP – Brasil

ricardofuz@usp.br, rtvb@icmc.usp.br

**Resumo.** *Os sistemas de aprendizagem podem incorporar dispositivos da Internet das Coisas, os quais adquirem a ciência de contexto por meio de sensores e permitem que os sistemas de aprendizagem evoluam rumo ao aprendizado ubíquo. No entanto, conforme identificado por um Mapeamento Sistemático realizado anteriormente, desenvolver sistemas para este domínio de aplicação envolve um alto grau de retrabalho, já que técnicas de reúso de software são pouco exploradas. Com base nisto, este trabalho apresenta a UL-SOPL (Ubiquitous Learning Service-Oriented Product Line), uma Linha de Produtos Orientada a Serviços projetada a partir de sistemas de aprendizagem existentes. São apresentados cenários de uso em que a UL-SOPL é instanciada derivando sistemas de aprendizagem específicos.*

**Abstract.** *Learning systems might incorporate Internet of Things devices, which acquire the sensors' context awareness and allow learning systems evolution towards ubiquitous learning. However, as identified in a Systematic Mapping previously done, developing systems in this application domain involves a high degree of rework, since reuse techniques are little explored. Based on this, this work presents UL-SOPL (Ubiquitous Learning Service-Oriented Product Line), a Service Oriented Product Line designed from existing learning systems. Use scenarios where UL-SOPL is instantiated deriving specific learning systems are also presented.*

## 1. Introdução

Os sistemas de aprendizagem podem ser classificados considerando diferentes características de projeto e implementação. Entre essas classes de sistemas, estão os sistemas de aprendizado ubíquo, que utilizam informações sobre os participantes para oferecer objetos de aprendizado e conduzir atividades de forma mais inteligente [Hwang 2006]. Para obter essas informações, é necessário utilizar dispositivos capazes de mensurar diferentes grandezas. Em geral, esses dispositivos capturam dados sobre o ambiente de inserção do usuário, como por exemplo se ele encontra-se em uma biblioteca ou sala de aula, ou informações mais precisas do usuário, como sua posição geográfica, por exemplo.

Estes dispositivos possuem os mesmos objetivos e características de um dispositivo utilizado no paradigma da Internet das Coisas (do inglês *Internet of Things* - IoT) [Atzori et al. 2010], o qual introduz a utilização de pequenos dispositivos computacionais distribuídos ao longo de um ambiente, de forma pervasiva. Cada dispositivo é capaz de coletar informações sobre o ambiente e objetos presentes, e tornar um serviço mais

inteligente e/ou possibilitar novas aplicações [Atzori et al. 2010]. Estas características são desejáveis para sistemas de aprendizado ubíquo, que precisam dos dados sobre os usuários e seus ambientes de inserção para agirem de forma personalizada e inteligente [Möller et al. 2013].

No entanto, após a realização de um Mapeamento Sistemático da literatura, identificou-se que, apesar de haver um grande volume de novos sistemas sendo propostos neste domínio de aplicação, a maioria desses sistemas não consideram a existência dos demais durante seu desenvolvimento, o que implica no retrabalho desnecessário para solucionar os mesmos problemas. Para lidar com tal retrabalho, pode-se utilizar técnicas de reúso de software, tais como Linhas de Produtos de Software [Clements and Northrop 2002, Gomaa 2005], arquiteturas baseadas em componentes [Crnkovic 2001] e orientação a serviços [Rosen et al. 2012].

Neste trabalho, é apresentado um modelo de *features* [Gomaa 2005, Clements and Northrop 2002] para a Linha de Produtos Orientada a Serviços [Lee et al. 2008] nomeada *UL-SOPL (Ubiquitous Learning Service-Oriented Product Line)*, projetada a partir do Mapeamento Sistemático feito anteriormente [Fuzeto and Braga 2016]. Também são apresentados dois cenários de uso, em que diferentes produtos gerados a partir da UL-SOPL são usados, evidenciando as *features* instanciadas.

O restante deste artigo está organizado da seguinte forma: a Seção 2 trata da revisão bibliográfica, trazendo os conceitos pertinentes sobre educação eletrônica, Internet das Coisas, Linhas de Produtos de Software e orientação a serviços, finalizando com os resultados do mapeamento sistemático feito sobre educação eletrônica com IoT. Na Seção 3 são apresentados os detalhes da LPS proposta, a qual permitirá instanciar sistemas de aprendizagem ubíquos utilizando dispositivos IoT. Por fim, na Seção 4 apresentam-se as conclusões deste trabalho e os trabalhos futuros.

## 2. Conceitos Básicos

O domínio de aplicação deste trabalho envolve os conceitos de outros dois domínios distintos: a *educação eletrônica* e a *Internet das Coisas (IoT)*. Nesta seção, ambos domínios são apresentados de forma mais aprofundada, finalizando com os resultados do Mapeamento Sistemático realizado sobre o domínio de aplicação deste trabalho, e conceitos básicos sobre LPS e orientação a serviços.

### 2.1. Ensino Eletrônico

O Ensino Eletrônico (ou *e-learning* em inglês) é a modalidade de ensino onde o processo de ensino/aprendizagem acontece através do uso de tecnologias computacionais, principalmente computadores pessoais. Segundo [Maurer 2003], um sistema *e-learning* deve permitir que os usuários possam se comunicar, gerenciar o aprendizado fornecido e adquirido, e anotar os materiais de ensino com informações relevantes. No mundo contemporâneo, onde a internet é o meio de comunicação mais utilizado devido ao seu alcance global, este aprendizado acontece em ambientes virtuais, nos quais os participantes podem interagir entre si por meio da comunicação via *web*. Dentre os ambientes utilizados no Brasil, um dos mais conhecidos e que exemplifica bem o paradigma do *e-learning* é o Moodle [Moodle 2017].

O *e-learning* deu origem a outros paradigmas educacionais, conforme foram incorporadas novas tecnologias computacionais ao processo de ensino e aprendizagem. São esses paradigmas:

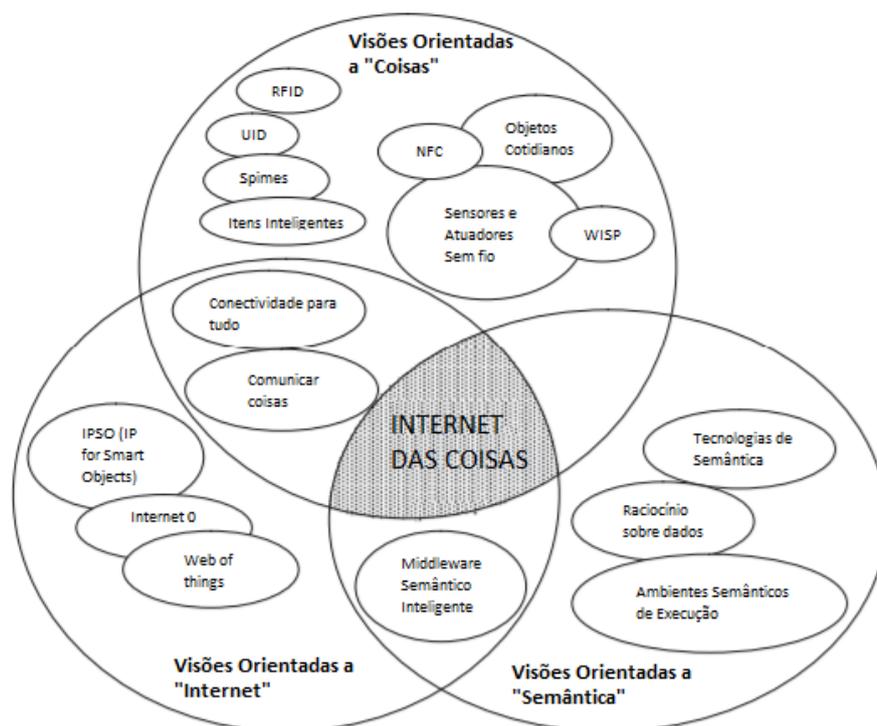
- Ensino Móvel (*m-learning*): sistemas *m-learning* aprimoram o processo de aprendizado por meio do emprego de um dispositivo móvel [Kearney et al. 2012]. Com a utilização desses dispositivos para incorporar a localização do usuário de forma positiva, todo o processo de ensino e aprendizagem é enriquecido, já que os participantes fazem contato mais direto com seus objetos de aprendizado.
- Ensino Pervasivo (*p-learning*): o aprendizado pervasivo é definido como o aprendizado disponível em qualquer lugar, a qualquer momento [Syvanen et al. 2005]. O grande avanço no uso do aprendizado pervasivo é permitir que o aprendizado não seja personalizado apenas em termos de local, mas também de momento em que ocorre.
- Ensino Ubíquo (*u-learning*): o aprendizado ubíquo toma as características dos aprendizados móvel e pervasivo, indo um passo além. Segundo [Hwang 2006], e diferente do que Syvanen *et. al.* afirmam sobre esta modalidade, o aprendizado ubíquo incorpora a *ciência de contexto* no processo de ensino e aprendizagem. Isso significa utilizar as informações sobre cada participante para personalizar o aprendizado, porém de forma mais inteligente. Isto é feito oferecendo conteúdos relacionados aos interesses de um participante ou ajustando os níveis de complexidade dos objetos de aprendizado, por exemplo. Exatamente por estes motivos que sistemas de aprendizado ubíquo frequentemente utilizam, internamente, sistemas de recomendação.

A ciência de contexto, citada anteriormente, é uma característica de sistemas computacionais que são capazes de coletar informações sobre um usuário, e utilizá-las para tornar um dado serviço mais eficiente [Hwang et al. 2009]. Essas informações são reunidas e sumarizadas em uma espécie de arquivo, onde pode-se encontrar todas as informações sobre um dado usuário que já tenham sido coletadas pelo sistema. Este arquivo é chamado de *perfil de usuário*, e a atual situação de um usuário, descrita pelas informações de seu perfil, é chamada de *contexto*.

## 2.2. Internet das Coisas

A Internet das Coisas (do inglês *Internet of Things* - IoT) é um paradigma computacional em que os usuários estão rodeados por dispositivos pervasivamente distribuídos pelo ambiente, que possuem endereçamentos únicos e são capazes de se comunicarem uns com os outros e cooperar para atingir objetivos comuns [Atzori et al. 2010]. Esses dispositivos são chamados de *coisas*, e muitas vezes possuem algum tipo de comunicação *sem fio*, o que facilita sua distribuição de forma pervasiva e potencializa a cooperação. O paradigma da IoT é formado a partir das diferentes visões das tecnologias que a constituem, conforme pode ser visto na Figura 1.

De acordo com cada visão destacada na Figura 1, diferentes tecnologias participam, na visão da IoT, como *possibilitadoras*, isto é, permitindo que a aplicação da Internet das Coisas seja possível. [Atzori et al. 2010] destaca, entre todas as tecnologias incorporadas na Internet das coisas, as seguintes como mais importantes:



**Figura 1. O paradigma da “Internet das Coisas” como resultado da convergência de diferentes visões. Traduzido de [Atzori et al. 2010].**

- *RFID e Wireless Sensor Networks (WSN)*: RFID é considerada a tecnologia fundamental da IoT por ser composta, primordialmente, de alguns leitores e várias etiquetas. Esta tecnologia também consome pouco ou nenhuma energia, já que as etiquetas RFID não utilizam baterias e funcionam com os pulsos gerados, por indução, a partir dos sinais de comunicação com os leitores RFID. As redes de sensores sem fio (WSNs) auxiliam o trabalho das RFIDs, coletando informações sobre o ambiente e fornecendo mais dados sobre o estado das diferentes *coisas* do sistema.
- *Middleware*: um *middleware* é uma camada de aplicação, ou um conjunto de camadas, inserido entre as camadas de aplicação e tecnologias de empacotamento e transporte de dados. Este tipo de aplicação tem sido popular no desenvolvimento de sistemas baseados em IoT por seu *design* ser baseado no uso de SOA (*Service-Oriented Architecture*), um modelo de arquitetura de *software* em que o sistema é construído a partir de serviços.

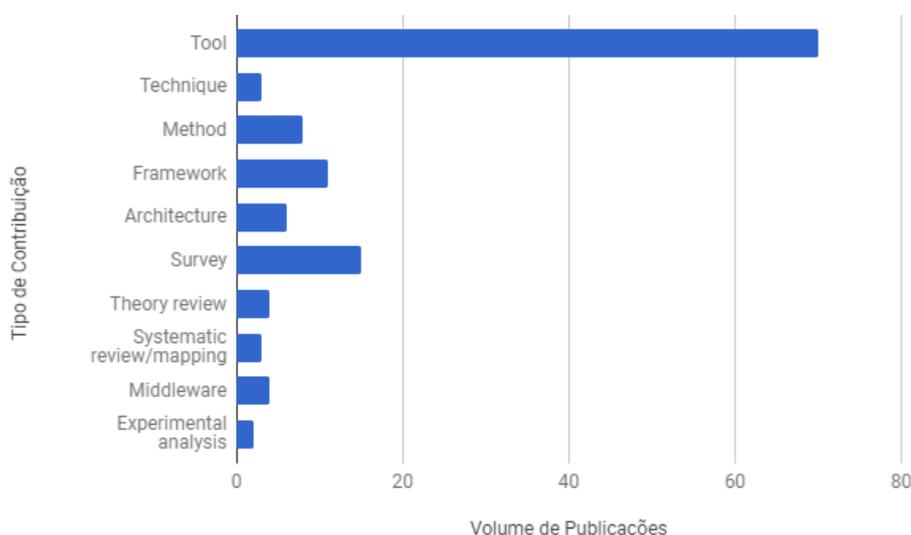
### 2.3. Mapeamento Sistemático

Foi feito um Mapeamento Sistemático [Petersen et al. 2008] no domínio de aplicações apresentado, considerando ambos ensino eletrônico e IoT. O objetivo deste MS é identificar os estudos que relacionam o aprendizado eletrônico com a utilização de dispositivos IoT, com preferência pela utilização desses a fim de incorporar ciência de contexto ao sistema. Uma versão preliminar deste SM foi publicada [Fuzeto and Braga 2016], e posteriormente complementada com mais bases de dados.

Para realizar o MS, foram consultadas cinco bases de dados distintas: ACM Di-

gital Library, IEEE Xplore, ScienceDirect, Scopus e SpringerLink. A pesquisa foi feita procurando pelas modalidades de educação eletrônica conjuntamente com IoT, RFID e WSNs. Esta pesquisa possibilitou encontrar resultados relevantes ao domínio de pesquisa que não possuem referência à IoT no texto, e, portanto, não seriam encontrados.<sup>1</sup>

No total, foram recuperados mais de 4000 resultados. Após as fases de leitura e filtragem, foram selecionados 124 estudos. De acordo com os dados dos estudos selecionados, mais de 55% dos estudos apresentam uma proposta de sistemas com IoT, seguidos por *surveys* em cerca de 12%, e *frameworks* em aproximadamente 9%. Esses dados podem ser visualizados na Figura 2. Em relação às tecnologias IoT utilizadas, os *Personal Handheld Devices* (PDAs) são os mais usados, aparecendo em quase 64% dos estudos selecionados. Em seguida, tem-se as WSNs em pouco mais de 48%, RFIDs em quase 43%, e, por fim, a utilização da nuvem em cerca de 12%.



**Figura 2. Diferentes tipos de contribuições encontrados no MS, e o volume proporcional de cada uma.**

Ao analisar os dados obtidos, as seguintes conclusões foram feitas:

- Há uma tendência crescente em realizar pesquisas neste domínio de aplicação. O volume de publicações é crescente, com grandes picos anteriores a 2010. No entanto, os volumes de publicações anuais mostra uma tendência de crescimento, e é esperado que continue assim devido à popularização cada vez maior da IoT;
- Quando se trata de ensino eletrônico e IoT, frequentemente são desenvolvidos sistemas de aprendizado *ubíquo*. Esta relação tem fundamento, uma vez que é destacada uma das principais características da IoT: a ciência de contexto. Conforme mencionado por [Atzori et al. 2010], o trabalho em conjunto dos sensores de um sistema IoT permite que diversas informações sejam coletadas sobre o ambiente de um objeto, bem como algumas de suas propriedades. Esses objetos podem ser relacionados aos participantes de um curso, quando transporta-se este cenário para sistemas de aprendizagem. Portanto, as informações coletadas pelos sensores compõem o *contexto*, tornando o sistema, assim, *ciente de contexto*. Conforme

<sup>1</sup>O planejamento completo do MS está disponível em [Fuzeto and Braga 2016].

dito anteriormente, a ciência de contexto é característica primordial de sistemas de aprendizado ubíquo.

- Entre os estudos selecionados, não há a reutilização de outros sistemas para evitar retrabalho. Apesar de ser possível encontrar alguns estudos que apliquem técnicas de reuso [Badr and Maamar 2009, Kim et al. 2011, Villanueva et al. 2015], os sistemas propostos anteriormente não são considerados, e, portanto, não há reuso nas soluções propostas.

## 2.4. Abordagens Tecnológicas

Conforme citado anteriormente, o sistema aqui proposto é, na verdade, uma Linha de Produtos de Software com elementos de orientação a serviços incorporados. Para entender melhor o *design* do sistema, é necessário entender os conceitos básicos dessas duas abordagens.

Uma Linha de Produtos de Software (LPS) “consiste em uma família de sistemas de *software* que possuem funcionalidades comuns e algumas funcionalidades variáveis” [Gomaa 2005]. Em uma LPS, os produtos que fazem parte da família compartilham um conjunto de características (do inglês *features*) comuns. Além disso, também existem *features* que aparecem em parte dos produtos, que definem os pontos de *variabilidade* da linha de produtos [Clements and Northrop 2002]. Para desenvolver uma LPS, pode ser adotada uma das duas abordagens definidas por [Bachmann et al. 2012]: reativa (utilizar um conjunto de produtos como base para a análise de *variabilidade*) e proativa (realizar análise de domínio para definir as variabilidades e comunalidades da LPS).

Serviços são “unidades discretas de negócio disponibilizadas por meio de um contrato de serviço” [Rosen et al. 2012]. Em se tratando de sistemas computacionais, serviços podem ser vistos como aplicações acessadas via interfaces bem definidas. Para isso, a abordagem mais utilizada atualmente é a disponibilização de tais aplicações como um *web service*. Assim, o acesso a essas aplicações é fácil, já que o acesso a um *web service* é feito via requisições HTTP. A estruturação de um serviço é apresentada na Figura 3.

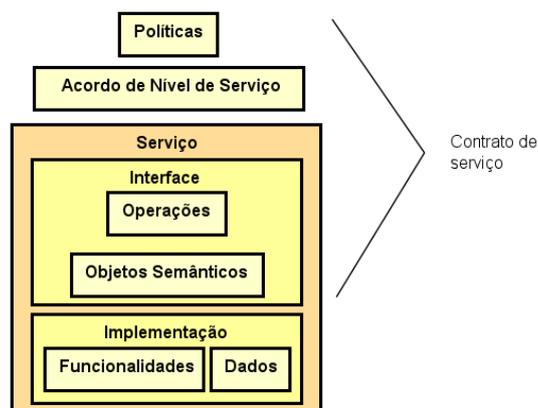


Figura 3. Estrutura de um serviço, adaptada de [Rosen et al. 2012].

A união entre as abordagens de LPS e orientação a serviços foi proposta por [Lee et al. 2008], em que cada *feature* da linha de produtos deve ser visualizada como

um serviço distinto, que pode ser construído separadamente ou através da composição de outros serviços da LPS. A utilização desta visão e a visão de cada *feature* como um serviço distinto promove ainda mais o reúso de *software*, uma vez que a LPS não fica restrita a uma única implementação de uma *feature*, podendo incorporar implementações diferentes, que são disponibilizadas como serviços distintos. Esta visão resulta em uma Linha de Produtos Orientada a Serviços (LPOS).

### 3. UL-SOPL

Segundo os dados do MS apresentado na seção anterior, o grau de reúso de software para sistemas educacionais ubíquos que utilizam dispositivos IoT ainda é baixo. Isto pode ser observado pelo grande volume de novos sistemas propostos, entre os quais raramente um sistema proposto por outro autor é referenciado. Por isso, identifica-se como grande a necessidade de aplicar, neste domínio, uma abordagem de reúso de *software* cujo resultado possa ser utilizado por terceiros. Neste projeto, a abordagem de LPOS foi escolhida para amenizar este problema.

#### 3.1. Projeto da LPOS

Neste artigo propõe-se a criação de uma LPOS, denominada UL-SOPL (*Ubiquitous Learning Service-Oriented Product Line*). A ideia é identificar as funcionalidades de sistemas de aprendizagem que utilizam dispositivos IoT, resultando em um modelo de *features*, que posteriormente poderá ser implementado utilizando a abordagem de serviços.

Como haviam vários exemplos de sistemas disponíveis para realizar a análise de domínio, a abordagem de LPS escolhida foi a *reativa*. Assim, a UL-SOPL incorpora as soluções desenvolvidas em todos os estudos selecionados, criando um *design* de sistema mais abrangente para o domínio de aplicação considerado. Além disso, com o grande volume de novos sistemas propostos, espera-se que a abordagem de LPS seja recebida positivamente pela comunidade de interesse. O modelo de *features* da UL-SOPL, identificando os pontos de comunalidade e varibilidade, pode ser visto na Figura 4.

A notação utilizada no modelo de *features* possui os seguintes significados:

- *Mandatory*: todas as *features* marcadas com um círculo preenchido estão presentes em todos os produtos, e, portanto, devem ser instanciadas toda vez que um novo produto é gerado;
- *Optional*: as *features* marcadas com um círculo vazio são opcionais, isto é, ausentes em pelo menos um dos produtos instanciáveis da linha de produtos;
- *Or*: algumas *features* são organizadas em grupos, sendo associadas a uma *feature* no nível hierárquico imediatamente superior. Os grupos cujas associações estão marcadas com um arco preenchido são grupos de cardinalidade “um ou mais”, isto é, pelo menos uma das *features* de tal grupo deve ser instanciada. Por esse motivo, as *features* deste tipo de grupo não possuem sinal de *mandatory* ou *optional*;
- *Abstract*: as *features* com coloração mais clara são abstratas: representam características e/ou comportamentos importantes para o produto instanciado, no entanto sem gerar impacto significativo na estrutura do sistema [Thum et al. 2011];
- *Concrete*: as *features* com coloração mais escura são concretas: ao contrário das abstratas, as *features* concretas causam impacto significativo no produto gerado por representarem parte da implementação e/ou estrutura da linha de produtos;

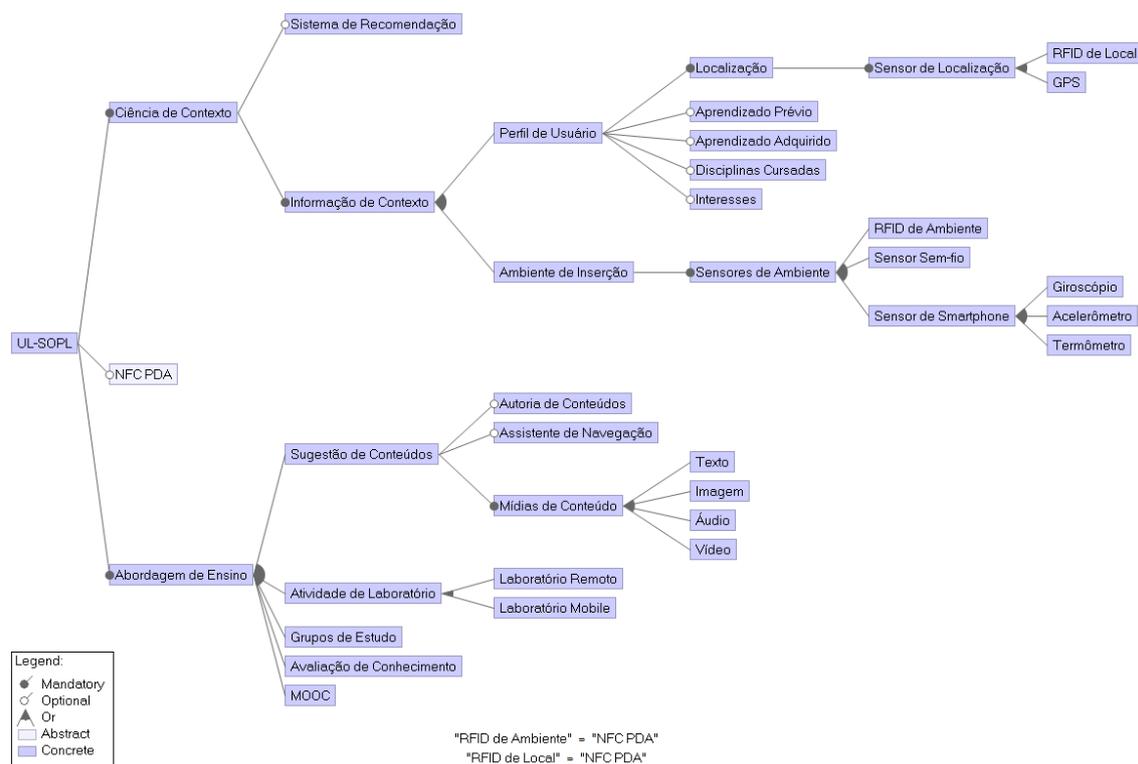


Figura 4. Modelo de *features* da UL-SOPL.

- *Equações*: as equações que aparecem abaixo do modelo são restrições. A princípio, estas fórmulas são escritas utilizando matemática *booleana*, descrevendo relações entre *features* mutuamente inclusivas ou exclusivas. No caso da ferramenta utilizada, a notação é diferente do padrão mas possui a mesma lógica.

As *features* da UL-SOPL foram divididas inicialmente em dois subgrupos: “Ciência de Contexto”, que contém as características referentes à coleta e manipulação das informações de contexto, e “Abordagens de Ensino”, que contém as formas que a UL-SOPL pode apresentar os aprendizados aos usuários. Dentro de “Ciência de Contexto”, tem-se “Sistema de Recomendação”, que trata a existência e utilização de um subsistema para selecionar e recomendar objetos de aprendizado, e “Informações de Contexto”, que contém os dados constantes no contexto de um usuário e quais tipos de dispositivos podem coletá-los. Dentro desse grupo, pode-se encontrar “Perfil de Usuário” e “Ambiente de Inserção”, ambos tratando das informações de contexto do usuário, no entanto diferindo no referencial: enquanto “Perfil de Usuário” trata das informações dependentes unicamente do usuário, “Ambiente de Inserção” trata das informações do ambiente onde o usuário está. “Perfil de Usuário” também é um grupo de possíveis informações que podem descrever o usuário, no entanto mais à frente, para ambos grupos, encontramos os tipos de dispositivos IoT que coletam os dados de contexto.

Para o grupo “Abordagens de Ensino”, tem-se o subgrupo “Sugestão de Conteúdos” responsável por enviar objetos de aprendizado adequados aos usuários. É importante lembrar que “Sugestão de Conteúdos” não é responsável por avaliar a adequabilidade de um objeto de aprendizado para um usuário.

Considerando o domínio de aplicação, que envolve sistemas de aprendizagem e IoT, deve-se considerar a necessidade de usar tecnologias *web* para que a construção dos serviços que irão conter as funcionalidades do sistema seja possível. Para isso, o projeto da LPS incorpora os conceitos de orientação a serviços, resultando na criação de uma LPOS. Desta forma, a LPS terá o suporte às tecnologias de comunicação *web* necessárias para a IoT, bem como um *design* que permita o escalamento do número de sensores e dispositivos utilizados.

O objetivo da UL-SOPL é melhorar o grau de reuso de *software* no domínio de sistemas considerado. A melhora no grau de reuso é confirmada após realizar uma análise automatizada, que levanta mais de 150 mil combinações diferentes possíveis de serem feitas a partir do modelo de *features* apresentado. Além disso, a linha de produtos, após implementada, permitirá instanciar novos sistemas mais rapidamente, o que facilitará o trabalho de quem precisar de sistemas diferentes para um mesmo nicho de aplicação. Por fim, o terceiro objetivo da LPOS é incentivar o uso e experimentação de sistemas de aprendizado ubíquo em diversas atividades pedagógicas, principalmente nas situações e ambientes onde sistemas computacionais não são utilizados por alguma limitação tecnológica, permitindo que os esforços em desenvolver os produtos desejados sejam focados nessas barreiras.

Além da análise de domínio conduzida a partir dos estudos selecionados no MS descrito na Seção 2.3, o modelo de features da UL-SOPL considerou em seu projeto a LPS proposta por [Junior 2015], denominada *M-SPLearning*. A linha de produtos em questão incorpora sistemas de aprendizagem *móveis*, não considere *features* referentes à ciência de contexto e integração com IoT. Por outro lado, sistemas de aprendizagem ubíquos incorporam as características de sistemas *móveis*, portanto é plausível que os modelos da *M-SPLearning* sejam utilizados de forma a descrever parte das características da UL-SOPL. Assim, o modelo de features da *M-SPLearning* foi adaptado para o domínio de sistemas de aprendizagem ubíquos, além de incorporar padrões que facilitem o uso de dispositivos IoT.

É importante mencionar que, atualmente, tem-se apenas os artefatos que descrevem a UL-SOPL, e que a implementação da linha de produtos ainda está sendo feita. A seguir, são descritos dois cenários de uso que ilustram possíveis produtos instanciáveis a partir da UL-SOPL.

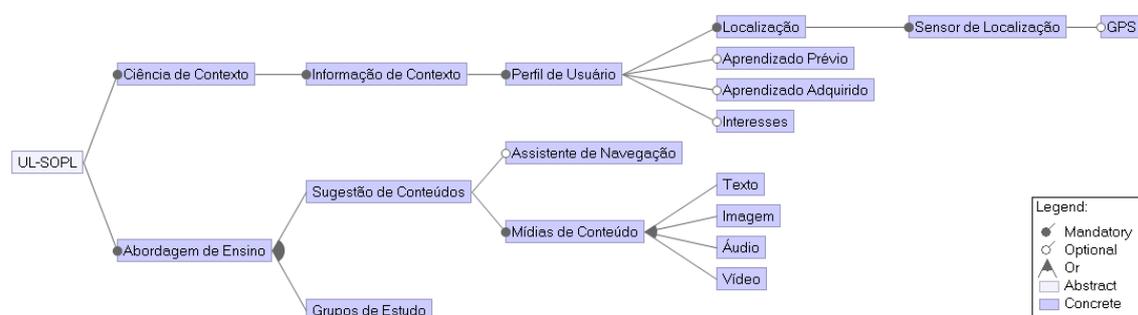
### 3.2. Cenários de Uso

Para entender como cada *feature* impacta no comportamento final de cada produto, e como os produtos utilizam os diferentes serviços instanciados e os dispositivos IoT, a seguir são apresentados dois possíveis cenários de uso da UL-SOPL. Os modelos de features correspondentes aos produtos gerados para esses dois cenários, já tendo sido selecionadas as features opcionais e alternativas, são mostrados nas Figuras 5 e 6, respectivamente.

*Cenário de uso 1*: um usuário caminha por um dos ambientes monitorados pelo sistema, com o aplicativo aberto em seu *smartphone*. Aqui, a *feature* “Assistente de Navegação” é instanciada, já que o sistema precisa auxiliar o usuário a navegar até o objeto de aprendizado. Então, o aplicativo informa a nova posição do usuário, e o servidor, através do serviço de ciência de contexto, pode identificar que há um objeto de aprendizado novo e próximo do usuário. Este cenário indica que as *features* “GPS” e “Mídias de

Conteúdo” são instanciadas, pois o sistema precisa da posição do usuário, que a princípio é coletada via GPS, e objetos de aprendizado para serem enviados ao usuário em questão. O servidor recupera esse objeto e o envia para o usuário, que poderá localizá-lo através do aplicativo e adquirir um novo aprendizado.

Também é possível que o sistema identifique outros usuários nas proximidades, e com os mesmos interesses. O servidor, então, solicita ao serviço de análise de contexto se os perfis possuem interesses em comum. Assim, o aplicativo informaria quais usuários próximos possuem interesses em comum, incentivando a formação de grupos de estudo e colaboração. A *feature* “Interesses” deve ser instanciada, para que o sistema armazene esse tipo de informação e possa comparar os perfis dos usuários para montar os grupos de estudo. Além disso, a *feature* “Grupos de Estudo” também é instanciada, para que o sistema tenha essa funcionalidade.



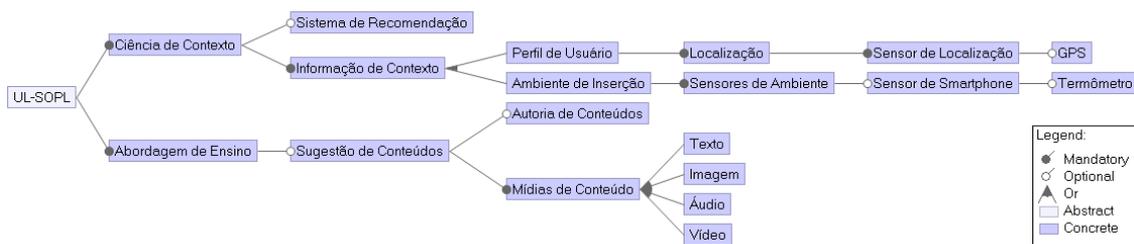
**Figura 5. Modelo de *features* do produto presente no cenário de uso 1.**

*Cenário de uso 2*: um usuário aprende algo novo, porém sem o auxílio do sistema. Através das ferramentas de criação (caso tenham sido implementadas neste produto), ele pode registrar esse novo aprendizado e enviar para o servidor, que armazena o novo objeto na base de dados. Para que seja possível a autoria de novos conteúdos por parte dos usuários, a *feature* “Autoria de Conteúdos” é instanciada. Com isso, o sistema pode classificar este novo objeto e relacioná-lo com objetos já existentes no sistema, através do serviço de análise de contexto. Caso o objeto de aprendizado criado tenha uma localização e a temperatura do ambiente associados, as *features* “GPS” e “Termômetro” devem ser instanciadas para que o sistema possa ser capaz de coletar esses dados.

Com o envio do novo objeto de aprendizado para o sistema, duas coisas podem acontecer: os objetos de aprendizado relacionados podem ser sugeridos ao usuários (apenas os que ele não aprendeu ainda), e outros objetos próximos podem ser sugeridos também (a relação neste caso é feita meramente pelo local). Como há a sugestão de conteúdos para os usuários, a *feature* “Sistema de Recomendação” deve ser instanciada para que esse tipo de subsistema seja utilizado. É importante lembrar que o aplicativo deve adaptar o objeto de aprendizado produzido, já que, por exemplo, uma câmera de 13 mega pixels de resolução pode tirar fotos com mais de 1 *Megabyte* de tamanho.

#### 4. Conclusões

Neste trabalho, foi apresentada a UL-SOPL, uma LPOS que visa facilitar a implementação de sistemas de aprendizagem ubíquos utilizando dispositivos IoT. O modelo de *features* que a descreve foi apresentado, restando a implementação de cada *feature*



**Figura 6. Modelo de *features* do produto presente no cenário de uso 2.**

como um serviço distinto, assim aumentando o grau de reúso de *software* para sistemas de aprendizagem com dispositivos IoT integrados. É feita a discussão sobre as características essenciais dos elementos de um produto instanciado a partir da linha de produtos (servidor e aplicativo móvel).

Como trabalhos futuros, pretende-se implementar uma versão da UL-SOPL e instanciar diferentes produtos a partir dela. Para avaliar a adequabilidade dos produtos e a validade da LPOS, é necessário realizar experimentos considerando a facilidade de instanciação de diferentes produtos, e avaliar a eficiência da linha de produtos em termos de aprendizado adquirido pelos sujeitos, comparando cada produto com os demais e com nenhum produto (sem a utilização do sistema).

## 5. Agradecimentos

Agradecemos ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), ao Procad-CAPES e à Universidade de São Paulo (USP) pelo apoio financeiro dado para a realização deste projeto.

## Referências

- Atzori, L., Iera, A., and Morabito, G. (2010). The internet of things: A survey. *Computer networks*, 54(15):2787–2805.
- Bachmann, F., Cohen, S., Donohee, P., Little, R., and Northrop, L. (2012). A framework for software product line practice, version 5.0.
- Badr, Y. and Maamar, Z. (2009). Autonomic service-oriented grid to enhance e-learning experiences. In *Digital Ecosystems and Technologies, 2009. DEST'09. 3rd IEEE International Conference on*, pages 649–654. IEEE.
- Clements, P. and Northrop, L. (2002). *Software product lines*. Addison-Wesley,.
- Crnkovic, I. (2001). Component-based software engineering—new challenges in software development. *Software Focus*, 2(4):127–133.
- Fuzeto, R. and Braga, R. (2016). Um mapeamento sistemático em progresso sobre internet das coisas e educação à distância. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, volume 5, page 1334.
- Gomaa, H. (2005). *Designing software product lines with UML*. IEEE.
- Hwang, G.-J. (2006). Criteria and strategies of ubiquitous learning. In *Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006. IEEE International Conference on*, volume 2, pages 72–77. IEEE.

- Hwang, G.-J., Yang, T.-C., Tsai, C.-C., and Yang, S. J. (2009). A context-aware ubiquitous learning environment for conducting complex science experiments. *Computers & Education*, 53(2):402–413.
- Junior, V. F. (2015). *Estudo e definição de uma linha de produtos de software para o desenvolvimento de aplicações educacionais móveis*. PhD thesis, Universidade de São Paulo.
- Kearney, M., Schuck, S., Burden, K., and Aubusson, P. (2012). Viewing mobile learning from a pedagogical perspective. *Research in learning technology*, 20(1):14406.
- Kim, S., Song, S.-M., and Yoon, Y.-I. (2011). Smart learning services based on smart cloud computing. *Sensors*, 11(8):7835–7850.
- Lee, J., Muthig, D., and Naab, M. (2008). An approach for developing service oriented product lines. In *Software Product Line Conference, 2008. SPLC'08. 12th International*, pages 275–284. IEEE.
- Maurer, H. A. (2003). Necessary aspects of quality in e learning systems. In *Quality education@ a distance*, pages 29–37. Springer.
- Möller, D. P. F., Haas, R., and Vakilzadian, H. (2013). Ubiquitous learning: Teaching modeling and simulation with technology. In *Proceedings of the 2013 Grand Challenges on Modeling and Simulation Conference(24)*. Society for Modeling & Simulation International.
- Moodle (2017). Moodle: Open-source learnign platform. [https://moodle.org/?lang=pt\\_br](https://moodle.org/?lang=pt_br).
- Petersen, K., Feldt, R., Mujtaba, S., and Mattson, M. (2008). Systematic mapping studies in software engineering. In *Proceedings of the 12th international conference on evaluation and assessment in software engineering*, volume 17(1).
- Rosen, M., Lublinsky, B., Smith, K. T., and Balcer, M. J. (2012). *Applied SOA: service-oriented architecture and design strategies*. John Wiley & Sons.
- Syvanen, A., Beale, R., Sharples, M., Ahonen, M., and Lonsdale, P. (2005). Supporting pervasive learning environments: Adaptability and context awareness in mobile learning. In *Wireless and Mobile Technologies in Education, 2005. WMTE 2005. IEEE International Workshop on*, pages 3–pp. IEEE.
- Thum, T., Kastner, C., Erdweg, S., and Siegmund, N. (2011). Abstract features in feature modeling. In *Software Product Line Conference (SPLC), 2011 15th International*, pages 191–200. IEEE.
- Villanueva, V. J. A., Marquez, F. J. D., Solarte, A. Z. M., and Dávalos, A. G. (2015). Architecture for integrating real objects with virtual academic communities. In *e-Learning (econf), 2015 Fifth International Conference on*, pages 385–391. IEEE.