



Universidade do Minho
Escola de Engenharia

Inteligência Ambiente: Tecnologias e Aplicações

Context-Awareness

Mestrado Integrado em Engenharia Informática

1º Semestre

2017-2018

Grupo 8

A75655 - Daniel Camelo Rodrigues

A74219 - Hugo Alves Carvalho

A74702 - José Manuel Gonçalves Leitão da Cunha

A74260 - Luís Miguel da Cunha Lima

9 de Dezembro de 2017

Braga

Resumo

Este documento relata o trabalho prático desenvolvido no âmbito da unidade curricular de **Inteligência Ambiente: Tecnologias e Aplicações**, tendo como tema principal a gestão de incerteza em *Context Awareness*.

Conteúdo

1	Introdução	5
2	Caso de Estudo	6
3	Modelação do Domínio	6
4	Funcionalidades do Sistema	7
5	Sensores	8
6	Requisitos	8
7	Regras de Associação	9
8	Requisitos de Mobilidade	9
9	Vulnerabilidades do Sistema	10
10	Formalização do Domínio	10
11	Conclusão	13

Lista de Figuras

1	Exemplo de arquitetura de um sistema <i>Context-Awareness</i> [2]	5
2	Modelo de Domínio	7
3	Modelo para Regras de Associação	9

1 Introdução

No âmbito da unidade curricular de Inteligência Ambiente: Tecnologias e Aplicações, foi proposto aos alunos a modelação de um sistema *Context Awareness*.

O conceito de contexto está diretamente relacionado com qualquer informação que possa ser adquirida e usada para caracterizar a situação de uma entidade. Uma entidade pode ser uma pessoa, um lugar ou um objeto que seja considerado relevante para a interação entre um utilizador e as suas aplicações [1]. Assim, através de contexto é possível representar o conhecimento sobre o ambiente, auxiliando a forma como este pode ser gerido pelas aplicações.

Um sistema *Context Aware* utiliza o contexto para melhorar a experiência do utilizador com as suas aplicações. Este apresenta a capacidade de reconhecer o meio físico onde se encontra, adaptando-se ao seu comportamento. Assim, o uso de contexto torna-se essencial sobretudo em aplicações interativas, uma vez que o contexto do utilizador está constantemente a ser alterado.

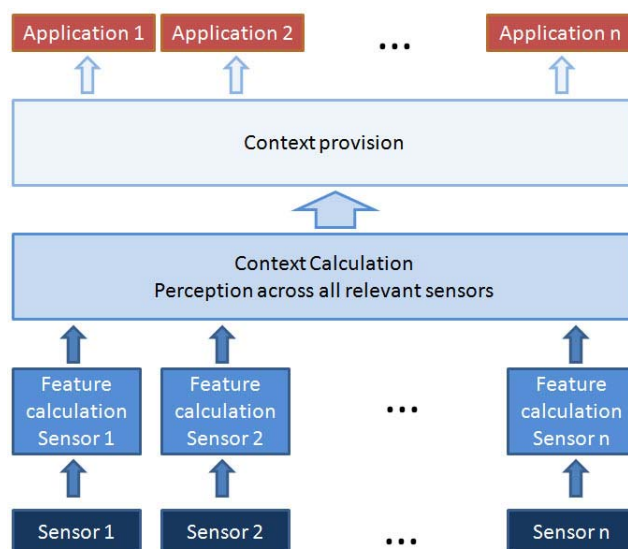


Figura 1: Exemplo de arquitetura de um sistema *Context-Awareness* [2]

Com a realização deste trabalho, pretende-se também aplicar os princípios do uso de técnicas de tratamento de incerteza em cenários com dados incompletos ou incertos, existentes em ambientes inteligentes.

Deste modo, foi modelado um sistema que cumpre estes requisitos, mais concretamente, um *context-aware system* adaptado ao ensino, que permite auxiliar um docente a cumprir os prazos do programa a lecionar.

2 Caso de Estudo

Nos dias de hoje, é recorrente o incumprimento de objetivos estabelecidos aos docentes pelas respectivas universidades e institutos. Um desses objetivos consiste em lecionar, respeitando as restrições temporais, toda a matéria estabelecida para a avaliação a realizar aos alunos.

Para esse fim, as instituições de ensino começaram a adotar um *context-aware system* com diferentes tipos de sensores que monitorizam e enviam informação a um professor, em tempo real. Deste modo, é permitido que os slides sobre a matéria a lecionar pelo professor sejam apresentados no tempo estabelecido para realização da aula. Além disso, o sistema viabiliza a comunicação com aparelhos digitais dos docentes possibilitando a interação com o mesmo. De cada vez que o docente se prepara para lecionar a aula, este deve ser reconhecido pelo sistema. Para tal, o sistema necessita de estar equipado com sensores que possibilitam a recolha de informação, inspecionando, por exemplo, a hora e a data do dia estabelecendo a comparação com a informação guardada na base de conhecimento. Por outro lado, o sistema deverá de ser capaz de perceber quais os dispositivos que estão disponíveis na sala de aula e que tipo de recursos apresentam. Esta característica é importante pois, permite saber quais os dispositivos que são capazes de mostrar os alertas. Ao identificar um dispositivo pertencente a um aluno, é lógico que não apresente a funcionalidade pretendida.

Neste sentido, caso o docente se encontre atrasado na matéria a lecionar, durante o decorrer de uma aula, o sistema consegue deduzir através de regras de inferência que o professor não acabará a aula com tudo ensinado. Nestas situações, um assistente virtual instalado no seu *laptop* apresenta uma notificação, sugerindo-lhe para agilizar o seu modelo de ensino. São considerados dois tipos de alertas: alerta textual visível no computador e alerta sonoro.

O sistema tem de estar apto para se manter atento à atividade do docente. De acordo com esta informação, será possível decidir qual a melhor maneira de lhe comunicar o atraso na matéria a lecionar.

3 Modelação do Domínio

De modo a compreender o funcionamento do sistema descrito, torna-se fundamental elaborar uma correta modelação do domínio. Assim, identificando as diferentes entidades e os seus relacionamentos, é possível interpretar todas as interações existentes no sistema.

Tendo em conta o caso de estudo descrito anteriormente, é possível identificar a **sala de aula** como o local onde decorrerá uma determina **aula**. Por outro lado, o **professor** é o responsável por lecionar a aula para um conjunto de vários **alunos**. Uma vez que o objetivo do sistema passa por controlar o tempo que um professor demora para ensinar a matéria, é necessário que este tenha um *laptop* onde será executada a **apresentação** de slides.

Devido a questões de ambiguidade, é também essencial diferenciar os atributos que cada uma destas entidades deve ter. Deste modo, consideramos as seguintes informações:

- Professor: Identificado pelo seu número (id) e nome;
- Sala de Aula: Identificada pelo seu número;
- Aula: Identificada pelo nome da Unidade Curricular, hora de início e hora de fim;
- Apresentação: Identificada pelo tema e número de slides;
- Laptop: Identificado pelo seu nome;
- Aluno: Identificado pelo seu número e nome;

Seguidamente, apresentamos um esboço de um diagrama de modelo de domínio, que permite uma melhor interpretação de todas as interações descritas.

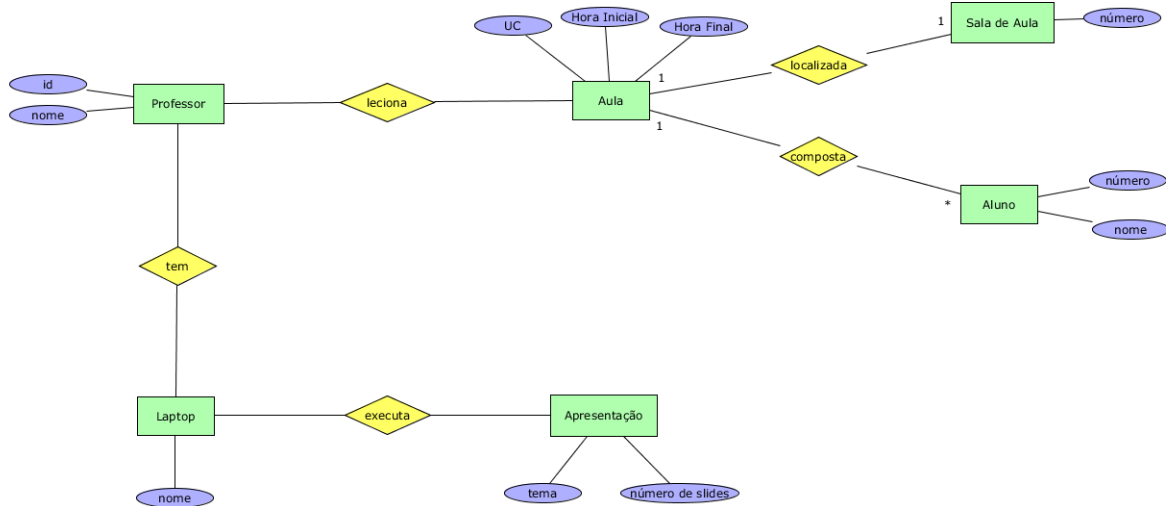


Figura 2: Modelo de Domínio

4 Funcionalidades do Sistema

Após a identificação das entidades do sistema, bem como dos seus respectivos relacionamentos, é necessário elaborar uma lista concreta e precisa, de todas as funcionalidades disponíveis que utilizam dados de *Context Awareness* necessários para a execução do sistema.

Deste modo, as principais funcionalidades do sistema devem corresponder aos seguintes tópicos:

- Identificar corretamente a sala de aula;
- Interpretar quando um professor entra na sala de aula (momento de início da aula);
- Identificar data, hora de início, hora de fim e duração da aula em questão;
- Reconhecer os dispositivos disponíveis na sala de aula, e identificar corretamente o dispositivo do docente, uma vez que apenas este deve receber alertas;
- Localizar o professor dentro da sala de aula;
- Permitir alertar o professor através de um alerta textual, em tempo real, sempre que este se encontra atrasado a lecionar a matéria;
- Permitir alertar o professor através de um alerta sonoro, em tempo real, sempre que este se encontra atrasado a lecionar a matéria para a aula. Por exemplo, quando está a tirar dúvidas a um aluno e se encontra longe do *laptop*;
- Identificar o número de slides total e quantos faltam lecionar para a aula em questão;

5 Sensores

Para a implementação deste sistema, é necessária a identificação de vários tipos de sensores que permitam o seu correto funcionamento. Deste modo, e tendo em conta as funcionalidades descritas anteriormente, foi elaborada uma lista de sensores necessários, bem como as suas respetivas aplicações práticas.

- Sensor que permita localizar a sala de aula;
- Sensor que verifique se o professor já entrou na sala de aula;
- Sensor que reconheça o *Laptop* do docente;
- Sensor que permita localizar o professor dentro da sala de aula, de modo a diferenciar quando usar alertas textuais ou sonoros;
- Sensor que identifique o número de slides a apresentar na aula;
- Sensor para reconhecer data e hora;

6 Requisitos

Na implementação deste sistema, é fundamental a definição dos seus requisitos. Estes dizem respeito às funcionalidades da aplicação relacionadas com o seu uso em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenção e tecnologias envolvidas. Considerando as funcionalidades do sistema, apresentam-se os seguintes requisitos:

- A sala de aula tem que estar equipada com todos os sensores anteriormente referidos;
- É obrigatório que o docente tenha um *laptop* no momento da aula;
- O docente deverá executar corretamente a apresentação no seu *Laptop*;
- O sistema deverá possuir uma base de conhecimento geográfica, permitindo a identificação e localização correta da sala de aula;
- Deverá estar manter a hora atual atualizada, para que seja possível estimar o fim de uma aula e executar os alertas respetivos.
- O sistema deverá ter alta disponibilidade quanto à sua utilização, incluindo o registo de mudanças e respetiva execução;
- O sistema deverá permitir estabelecer a comunicação com uma base de dados, de modo a obter o conhecimento correto para o seu funcionamento;
- O sistema apenas apresenta os alertas a utilizadores que sejam docentes.

7 Regras de Associação

$$P(R | S, M)$$

Figura 3: Modelo para Regras de Associação

Elementos:

- **R** - Risco de não conseguir acabar a matéria da aula;
- **S** - Número de slides que faltam apresentar;
- **M** - Minutos que faltam até acabar a aula;

Níveis de Risco:

- **R = 0**: baixo risco;
- **R = 1**: risco médio;
- **R = 2**: alto risco;

Regras:

- Faltar 8 minutos para acabar de lecionar 5 slides
– $P(R=0 | S = 5, M = 8) = 0.35$;
- Faltar 10 minutos para acabar de lecionar 10 slides
– $P(R=1 | S = 10, M = 10) = 0.60$;
- Faltar 20 minutos para acabar de lecionar 30 slides
– $P(R=2 | S = 30, M = 20) = 0.80$;

8 Requisitos de Mobilidade

- Inteligibilidade: O sistema disponibiliza uma interface simples e intuitiva para o *laptop* do professor, de modo a mantê-lo sempre atualizado sobre o conteúdo a apresentar e sobre o tempo disponível. Caso o professor pretenda alterar qualquer informação sobre as aulas a lecionar, este poderá usar a interface disponibilizada.
- Robustez: O sistema é capaz de se adaptar a algumas situações mais incomuns, como por exemplo a alteração da sala de aula sem que o sistema seja alertado previamente, identificado o professor em outra sala. Deste modo, o sistema verifica com o professor se a sala em que se encontra é mesmo a sala em que ele quer dar a aula, se sim, o sistema encarrega-se de efetuar a alteração.

Outro exemplo é quando por algum motivo o professor não consegue terminar de lecionar a aula, o sistema é capaz de identificar tal situação e depois criar um aviso que será apresentado na aula seguinte, de modo a relembrar o professor que há matéria ainda não lecionada da aula anterior.

- Privacidade: O sistema é concebido para identificar, através de sensores de reconhecimento, a sala e o professor, usando o dispositivo deste para que se encontre conectado ao sistema, enquanto estiver dentro da sala.

Para cada professor existe um perfil que está associado aos seus dados, promovendo assim uma maior segurança e privacidade. No momento de início da aula, é solicitado que o professor prove ser quem o sistema julga ter identificado, pedindo ao professor que introduza a sua palavra pass, que será solicitada na interface do sistema do *laptop*.

Os restantes dispositivos são identificados após a validação da identidade do professor, sendo feita uma verificação de segurança nestes, ao nível da aplicação instalada nos mesmos. Deste modo, evitamos que outras entidades tenham acesso ao sistema sem terem permissão para tal, garantindo uma divisão clara de informações entre os diferentes perfis registados.

- Eficiência: O sistema usa a informação dos diferentes sensores para determinar o comportamento que deve adotar perante cada situação em tempo real. Assim, torna-se capaz de cruzar os diferentes dados obtidos a partir dos sensores, com os dados que estão guardados na base de conhecimento.

9 Vulnerabilidades do Sistema

Na implementação deste sistema, é de extremo interesse reconhecer as vulnerabilidades que este possa apresentar, de forma a tratar e mitigar as falhas encontradas, garantindo maior segurança ao ambiente e estabelecendo uma nova linha de base para futuras análises. Considerando os vários requisitos e funcionalidades do sistema, apresentamos várias incompletudes que este possa apresentar.

- O sistema entender que o professor já se encontra dentro da sala de aula, quando este se encontra fora desta;
- O sistema não conseguir identificar e reconhecer o *laptop* do professor;
- O sistema não reconhecer a sala correta para o funcionamento da aula;
- Possíveis diferenças de tempos entre os diferentes sensores e dispositivos;
- Possibilidade do professor se encontrar dentro da sala de aula, mas num ponto morto do sensor de localização, não conseguindo diferenciar os tipos de alertas;
- Falta de bateria dos vários sensores, ou mesmo do *laptop* do docente;
- Erro de leitura do número de slides da apresentação;
- Informação incompleta ou errada na base de dados;

10 Formalização do Domínio

Com o objetivo de mostrar a possibilidade de computar o sistema descrito, o grupo utilizou o formato JSON para ilustrar um exemplo da modelação do domínio.

```

{
  "professor": {
    "nome": "prof555",
    "id": 555
  },
  "laptop": {
    "nome": "profPC"
  },
  "apresentacao": {
    "tema": "context-awareness",
    "numeroSlides": 25
  },
  "aula": {
    "UC": "IATA",
    "horaInicial": "9:00",
    "horaFinal": "12:00"
  },
  "alunos": [{
    "nome": "Aluno1",
    "numero": 1
  },
  {
    "nome": "Aluno2",
    "numero": 2
  },
  {
    "nome": "Aluno3",
    "numero": 3
  },
  {
    "nome": "Aluno4",
    "numero": 4
  },
  {
    "nome": "Aluno5",
    "numero": 5
  },
  {
    "nome": "Aluno6",
    "numero": 6
  },
  {
    "nome": "Aluno7",
    "numero": 7
  },
  {
    "nome": "Aluno8",
    "numero": 8
  },
  {
    "nome": "Aluno9",
    "numero": 9
  },
  {
    "nome": "Aluno10",
    "numero": 10
  },
  {
    "nome": "Aluno11",
    "numero": 11
  }
],

```

```

        "nome": "Aluno12",
        "numero": 12
    },
    ],
    "sensores": [{
        "id": "S1LS",
        "desc": "localizaSala",
        "estado": true
    },
    {
        "id": "S2VT",
        "desc": "verificaTemperaturas",
        "estado": true
    },
    {
        "id": "S3RL",
        "desc": "reconheceLaptop",
        "status": true
    },
    {
        "id": "S4LP",
        "desc": "localizaProfessor",
        "status": true
    },
    {
        "id": "S5RS",
        "desc": "reconheceSlides",
        "status": true
    },
    {
        "id": "S6ST",
        "desc": "dataHora",
        "status": true
    }
    ]
}

```

11 Conclusão

Terminada a realização do trabalho, é então possível concluir alguns aspetos que foram determinantes para a correta modelação do sistema.

Tendo em conta o caso de estudo descrito, foi fundamental compreender e definir o domínio para a implementação do sistema. Deste modo, foram identificadas todas as entidades que este deve englobar, bem com as suas interações e relacionamentos.

Em seguida, expressou-se, de forma simples e concreta, todas as funcionalidades que o sistema deve incluir no seu funcionamento. Nesse sentido, foi necessário identificar todos os sensores que este necessita, descrevendo o seu respetivo funcionamento. Por outro lado, com o mesmo objetivo de clarificar toda a atividade, foram estabelecidos os principais requisitos da aplicação.

Seguiu-se com a elaboração de um conjunto de regras de associação que representam o processamento de contexto associado a este domínio. Por outro lado, foram descritos e especificados os quatro requisitos de mobilidade em *Context Aware Systems* lecionados: *intelligibility*, *robustness*, *privacy* e *efficiency*. A identificação de presença de incompletude na execução da aplicação foi também de extrema importância, uma vez que permite compreender possíveis falhas que o sistema possa apresentar.

Por último, foi demonstrado através do formato JSON, como este sistema pode ser representado computacionalmente.

Em suma, o grupo ficou bastante satisfeito com o trabalho desenvolvido, uma vez que elaborou e compreendeu as principais etapas da concepção de um sistema *Context Awareness*

Referências

- [1] Anind K. Dey and Gregory D. Abowd. *Towards a better understanding of context and context-awareness*. In International symposium on Handheld and Ubiquitous Computing, pages 304–307. Springer-Verlag, 1999.
- [2] <https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/context-aware-computing-context-awareness-context-aware-user-interfaces-and-implicit-interaction>
- [3] A. Ben Yaghlane, T. Denoeux, and K. Mellouli. *Uncertainty and intelligent information systems, chapter Elicitation of Expert Opinions for Constructing Belief Functions*, pages 75–89. World Scientific, 2008.
- [4] Thomas Buchholz and Michael Schiffers. *Quality of context: What it is and why we need it*. In Proceedings of the 10th Workshop of the OpenView University Association: OVUA'03, 2003.
- [5] <https://www.tutorialspoint.com/json/>