



UNIVERSIDAD AUTÓNOMA METROPOLITANA
UNIDAD AZCAPOTZALCO

DIVISIÓN DE CIENCIAS BÁSICAS E INGENIERÍA
MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN

DETECCIÓN Y REPRESENTACIÓN DE EVENTOS EN
UN AMBIENTE ACADÉMICO INTELIGENTE

Idónea Comunicación de Resultados

para obtener el grado de:

Maestro en Ciencias de la Computación

PRESENTA:

Ing. Josué Padilla Cuevas

DIRECTORES:

Dr. José Alejandro Reyes Ortiz

Dra. Maricela Claudia Bravo Contreras



MAESTRÍA EN CIENCIAS
DE LA COMPUTACIÓN

Ciudad de México, 2019

A las personas que me apoyaron en todo momento, por confiar y creer en mi, han sido mi ejemplo a seguir a lo largo de mi vida, gracias por su amor, trabajo y sacrificio. Por ustedes he llegado hasta aquí y he cumplido una meta más. Es un orgullo y privilegio ser su hijo.

A mis padres

A las personas que me han enseñado a seguir adelante a pesar de las adversidades, gracias por enseñarme que jamás hay que rendirse.

A mis hermanos

A la persona que fue fundamental en esta etapa de mi vida, simplemente sin ti nada de esto hubiera sido posible. Gracias por estar conmigo en cada momento, solo tu sabes los momentos que pasé para alcanzar esta meta y siempre me motivaste a seguir. Contigo compartí risas y tristezas, pero ahora quiero compartir este logro. Gracias por todo tu amor y cariño.

A Gaby

Reconocimientos

Agradezco a mis directores el Dr. Alejandro y a la Dra. Maricela por su ayuda y conocimiento transmitido a lo largo de esta etapa, ustedes me han guiado en el complicado proceso. Sencillo no ha sido, sin embargo, gracias a sus consejos fue menos complicado. Gran parte del desarrollo de este trabajo se los debo a ustedes. Espero volverme a cruzar en su camino ya que es un honor trabajar a su lado.

Quiero agradecer a todo el personal administrativo y compañeros del departamento de sistemas, pero en especial a mis jefes la Ing. Isabel Cervantes y al Dr. Jesús Isidro por haberme permitido aprender de ustedes y facilitarme el equipo necesario para poder realizar mi proyecto. Siempre estaré agradecido.

Agradezco a la Maestría en Ciencias de la Computación y al coordinador el Dr. Luis Fernando Hoyos por darme la oportunidad de cursar este posgrado y siempre apoyarme en los momentos requeridos.

Deseo expresar mi agradecimiento a mis revisores Dra. Belém Priego Sánchez, Dra. Alma Delia Cuevas, Dr. Leonardo Daniel Sánchez Martínez y Dr. Gabriel González Serna. Gracias por tomarse el tiempo de ayudarme a mejorar este trabajo.

Al Consejo Mexiquense de Ciencia y Tecnología (COMECYT) por el apoyo económico brindado.

Por último, quiero reconocer a mi alma máter la Universidad Autónoma Metropolitana por la formación que me dio a lo largo de estos años. Es gracias a ella que es posible el presente trabajo.

Resumen

Un *Ambiente Inteligente* es un espacio físico o virtual que es capaz de responder a las necesidades de los usuarios según el contexto. Un *Evento* es aquello que ocurre dentro de un ambiente, el cual se caracteriza por variables de tiempo (en qué momento sucede), espacio (en qué lugar) y persona (quién participa). En particular, en este trabajo se aborda la detección y representación de eventos que ocurren en un ambiente *Académico* inteligente; el tipo de eventos académicos que se consideran son: cursos, asesorías, seminarios, presentaciones, entre otros.

En este documento de resultados se presenta un diseño ontológico para el modelado de contexto basado en eventos como núcleo del modelo, que es extensible y adaptable al dominio académico. El modelo ontológico es utilizado para la identificación de eventos académicos a partir de los datos adquiridos del entorno mediante el uso de reglas de decisión. Además, el modelo ontológico se usa para razonar con la información obtenida de los eventos identificados. El modelo de ontologías basado en eventos considera cinco aspectos contextuales con una perspectiva modular: persona, temporalidad (tiempo), espacialidad (ubicación), red (recursos para adquirir datos del ambiente) y evento (eventos académicos). Para la evaluación del modelo de ontologías se llevó a cabo un proceso enfocado en: a) la extensibilidad y adaptación de escenarios de casos de uso en eventos académicos; b) el nivel de razonamiento mediante el uso de preguntas de competencia relacionadas con eventos académicos identificados; c) la consistencia y la coherencia del modelo propuesto. La evaluación del proceso muestra resultados prometedores para el modelo ontológico basado en eventos.

Índice general

| | |
|---|-----------|
| Índice de figuras | IX |
| Índice de tablas | XI |
| 1. Introducción | 1 |
| 1.1. Presentación | 1 |
| 1.2. Planteamiento del problema | 2 |
| 1.3. Justificación | 3 |
| 1.4. Objetivos | 4 |
| 1.4.1. Objetivo general | 4 |
| 1.4.2. Objetivos específicos | 4 |
| 1.5. Contribuciones | 4 |
| 1.6. Estructura de la tesis | 5 |
| 2. Marco teórico | 7 |
| 2.1. Ontologías | 7 |
| 2.2. AMI (Ambiente inteligente) | 9 |
| 2.3. Eventos | 11 |
| 2.4. Agentes inteligentes | 12 |
| 3. Estado del arte | 15 |
| 3.1. Descripción de los artículos | 15 |
| 4. Metodología de solución | 21 |
| 4.1. Metodología implementada | 21 |
| 4.1.1. Fase 1 Modelo de eventos | 22 |
| 4.1.2. Fase 2 Detección de eventos | 22 |
| 4.1.3. Fase 3 Identificación de componentes de los eventos | 22 |
| 4.1.4. Fase 4 Representación de información en un sistema de ontologías | 22 |
| 4.2. Modelo ontológico | 22 |
| 4.2.1. Ontología de espacio físico | 23 |
| 4.2.2. Ontología de red de sensores | 27 |
| 4.2.3. Ontología de tiempo | 29 |

ÍNDICE GENERAL

| | |
|---|-----------|
| 4.2.4. Ontología de persona | 30 |
| 4.2.5. Ontología de evento | 32 |
| 4.2.6. Sistema ontológico: integración de los módulos | 36 |
| 4.3. Detección de eventos | 39 |
| 4.3.1. Detección de eventos ambientales | 39 |
| 4.3.1.1. Obtención y generación de datos | 40 |
| 4.3.1.2. Módulo de detección de eventos ambientales | 45 |
| 4.3.2. Detección de eventos académicos | 50 |
| 4.3.2.1. Caracterización de los eventos académicos | 51 |
| 4.3.2.2. Árbol de decisión para la detección de eventos | 52 |
| 4.3.2.3. Módulo de detección de eventos académicos | 53 |
| 4.3.3. Unificación del sistema | 57 |
| 5. Evaluación | 59 |
| 5.1. Evaluación de la ontología | 59 |
| 5.1.1. Escenarios de casos de uso | 59 |
| 5.1.2. Razonamiento por preguntas de competencia | 64 |
| 5.1.3. Consistencia, coherencia y eficiencia | 65 |
| 5.2. Evaluación del detector de eventos | 65 |
| 5.2.1. Agentes utilizados | 65 |
| 5.2.2. Servidor | 66 |
| 5.2.3. Servicios web | 67 |
| 5.2.4. Resultados del detector: precisión, exhaustividad y medida-f | 71 |
| 6. Conclusiones y trabajo futuro | 77 |
| A. Código/Manuales/Publicaciones | 79 |
| A.1. Apéndice: conjunto de datos para el análisis de resultados | 79 |
| Bibliografía | 87 |

Índice de figuras

| | |
|---|----|
| 2.1. Eventos y sus dimensiones. | 12 |
| 2.2. Ciclo de vida de los agentes. | 13 |
| 4.1. Arquitectura de solución. | 21 |
| 4.2. Taxonomía de espacio físico | 23 |
| 4.3. Fragmento de código OWL 2.0 | 24 |
| 4.4. Ontología de espacio físico | 25 |
| 4.5. Taxonomía red de sensores | 27 |
| 4.6. Individuo de la ontología red de sensores | 29 |
| 4.7. Ontología de tiempo | 30 |
| 4.8. Jerarquía de clases del modelo persona | 31 |
| 4.9. Ontología de evento | 34 |
| 4.10. Representación de un ambiente inteligente | 36 |
| 4.11. Sistema de Ontologías implementado en Protégé (OntroGraf) | 38 |
| 4.12. Paquete generado del nodo sensor | 40 |
| 4.13. Ejemplo de una lectura del nodo sensor | 41 |
| 4.14. Agentes inteligentes ambientales | 41 |
| 4.15. Paquete generado del nodo identificador | 42 |
| 4.16. Ejemplo de una lectura del nodo identificador | 42 |
| 4.17. Diagrama de clases del módulo de generación de datos | 44 |
| 4.18. Diagrama de clases del módulo de detección de eventos ambientales | 46 |
| 4.19. Árbol de decisión | 54 |
| 4.20. Diagrama de clases del módulo de detección de eventos académicos | 56 |
| 4.21. Diagrama de comunicación del detector ambiental | 57 |
| 4.22. Diagrama de comunicación del detector académico | 58 |
| 5.1. Escenario de casos de uso evento de entrada “eventEnters” | 60 |
| 5.2. Escenario de casos de uso asesoría “academicAdvising” | 60 |
| 5.3. Escenario de casos de uso curso de posgrado “postgraduateCourse” | 61 |
| 5.4. Escenario de casos de uso incremento de temperatura “eventTempInc” | 62 |
| 5.5. Escenario de casos de uso taller “workshop” | 63 |
| 5.6. Ejemplo de preguntas de competencia | 64 |
| 5.7. Simulación de paquetes provenientes de nodos | 66 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| 5.8. Detección de eventos ambientales | 67 |
| 5.9. Evento detectado curso de actualización | 68 |
| 5.10. Detección de evento académico taller | 69 |
| 5.11. Detección de evento congreso | 69 |
| 5.12. Evento detectado asesoría | 70 |
| 5.13. Detección de evento curso de licenciatura | 70 |

Índice de tablas

| | |
|--|----|
| 3.1. Comparación de artículos analizados con respecto al proyecto desarrollado | 19 |
| 4.1. Clases y conceptos de la ontología de espacio físico | 26 |
| 4.2. Elementos ontológicos del modelo red de sensores | 28 |
| 4.3. Elementos ontológicos del modelo tiempo | 30 |
| 4.4. Propiedades de datos y objetos del modelo persona | 32 |
| 4.5. Clases y conceptos de la ontología evento | 35 |
| 4.6. Propiedades de objeto entre módulos | 36 |
| 4.7. Niveles de Iluminación en un ambiente académico | 39 |
| 5.1. Número de eventos académicos detectados por día | 71 |
| 5.2. Número de eventos ambientales detectados por día | 71 |
| 5.3. Número de eventos totales detectados por día | 72 |
| 5.4. Datos obtenidos del patrón de referencia | 73 |
| 5.5. Resultados de precisión | 73 |
| 5.6. Datos con Gold Standard | 74 |
| 5.7. Grado de error del conjunto de sensores | 75 |
| 5.8. Resultados de exhaustividad | 75 |
| 5.9. Resultados de Medida-F | 75 |

Introducción

1.1. Presentación

En cualquier ambiente inteligente suceden eventos que están conectados entre sí, de ellos se obtiene información relevante como la hora, ubicación, actores involucrados, además, de conocer el tipo de evento que ocurre. Por supuesto, un ambiente académico no está exento de esta situación, en este contexto, los eventos son relacionados con las actividades académicas. Los usuarios del ambiente se benefician por el procesamiento y representación de eventos de manera automática, porque si se hiciera de forma manual el análisis sería lento y tedioso; mucho menos se podría realizar razonamiento e inferencia. Es muy importante disponer de un modelo para representar eventos académicos con su información contextual: ubicación (¿dónde?), hora (¿cuándo?), persona (¿quién?) y el tipo de evento (¿qué evento?).

En la actualidad existen pocos modelos extensibles y reutilizables para representar eventos académicos, así como razonamiento para obtener conocimiento sobre ellos. Por lo anterior, en esta idónea comunicación de resultados se describen los resultados de un proyecto de investigación cuyo objetivo principal es la detección y representación de eventos en un espacio académico. Los eventos corresponden a actividades de docencia, investigación y difusión de la cultura. Los datos del ambiente provienen de una red de sensores y éstos son usados para la detección de eventos que, posteriormente, son almacenados y representados en un modelo ontológico.

El uso de ontologías permite la comunicación estandarizada entre aplicaciones para el espacio académico. Además, las ontologías proporcionan las herramientas y tecnologías a estas aplicaciones para inferir y descubrir nuevo conocimiento. El resultado de este proyecto es un espacio académico dotado de servicios para ofrecer al usuario de manera automática, es decir, un espacio académico capaz de reaccionar a las solicitudes de manera inteligente apoyándose en variables contextuales.

1.2. Planteamiento del problema

En un ambiente académico suceden diariamente actividades de docencia con alta frecuencia, tales como: impartir asesorías, impartir clases, dirigir prácticas de laboratorio. También existen actividades de investigación: redactar artículos, lectura de artículos, dirigir experimentos científicos. Asimismo, suceden diversas actividades de difusión en un espacio académico, como: seminarios, conferencias, reuniones de proyectos de investigación.

Dentro de un ambiente académico los usuarios tienen la necesidad de contar con servicios de información relacionados con estas actividades o eventos que suceden, por ejemplo, ¿dónde se imparte la conferencia X? ¿a qué hora comienza la clase de la profesora Y?

Este trabajo de investigación considera la existencia de un espacio físico dotado de una red de sensores de presencia y ambientales, de una agenda de actividades programadas, y de un registro de personas (considerando profesores, alumnos, visitantes). El problema es diseñar y construir un modelo integral para la representación de todas las variables (o dimensiones) que intervienen en la ocurrencia de eventos en un ambiente académico. Asimismo, se requiere diseñar e implementar un sistema de detección de eventos.

Por lo tanto, se requiere implementar mecanismos para la detección de eventos **ambientales**, como son: la entrada y salida de personas, y los cambios en las variables ambientales. Aunado a esto, se requiere identificar el lugar y el tiempo en el que estos eventos suceden, por ejemplo: dónde y a qué hora entró o salió una persona; dónde y a qué hora cambió la temperatura, etc

Tomando como base toda la información sobre los eventos **simples** (ambientales), la agenda de actividades programadas, y los datos de las personas en el ambiente, se requiere diseñar e implementar un mecanismo (inteligente) a través del cual se determine el tipo de evento **académico** complejo que está sucediendo.

El beneficio que se obtiene es lograr convertir un espacio académico en un ambiente inteligente, es decir, un espacio donde se tomen decisiones automáticas para beneficiar a los usuarios finales.

Por otra parte, la detección de eventos permite que el sistema sea capaz de reaccionar de manera automática a los sucesos y condiciones climáticas que ocurren en un espacio físico, por ejemplo: adecuar las condiciones climáticas de un salón de acuerdo a las personas presentes y las actividades que están ocurriendo.

Por su parte, la representación de eventos consiste en almacenar información sobre lo

que sucede o los hechos acaecidos en el ambiente académico inteligente, mientras que las ontologías son herramientas que sirven para estructurar conceptualmente determinados ámbitos del conocimiento por medio de vocabularios controlados, para proporcionar la información de la construcción del sistema todo el fin de aportar consistencia, fiabilidad y veracidad al momento de recuperar la información.

1.3. Justificación

Los eventos suceden en los espacios inteligentes con gran frecuencia, por ejemplo, la presencia que indica la identificación de un individuo en un espacio. En un espacio académico inteligente es relevante identificar los tipos de eventos que ocurren en éste, porque expresan las actividades o acciones en relación al tiempo, espacio y personas que intervienen en ellos.

Además, se necesita contar con un registro o representación de estos eventos que sea procesable por computadoras y con una estructura semántica. Esto se logra con el uso de ontologías, las cuales ayudan a llevar a cabo inferencia con la información y con ello, detectar la ocurrencia de eventos complejos y deducir acciones en el contexto. Esto permite el trabajo colaborativo y funciona como soporte común de conocimiento entre aplicaciones y comunidades científicas. Entre dichas aplicaciones se encuentran sistemas de pregunta-respuesta, predicción de eventos y sistemas de búsqueda de información sobre el espacio académico inteligente.

El uso de ontologías, algoritmos de detección de eventos y razonamiento del contexto es relevante dentro de la investigación en las ciencias de la computación, además, la inteligencia artificial se ha convertido en un campo de estudio que ofrece tecnologías novedosas que mejoran la vida de las personas; por ejemplo, en la automatización de oficinas, industrias o ambientes inteligentes.

Con este proyecto, al identificar los eventos, se beneficia a los usuarios, al minimizar el tiempo y esfuerzo dentro del ambiente académico porque se realiza un procesamiento inteligente para conocer las preferencias del usuario, en relación a su registro, la temperatura del lugar, la humedad, la intensidad de la luz, entre otros. Esto sirve para que la comunidad estudiantil y la docente puedan realizar sus actividades de una manera más eficiente. Por ejemplo, al tener la iluminación adecuada dentro de un salón de clases, se optimiza la utilización de los recursos eléctricos mediante la regulación del encendido de lámparas y proyectores automáticamente cuando sea necesario.

1.4. Objetivos

1.4.1. Objetivo general

Diseñar, implementar y evaluar un sistema para la detección y representación semántica de eventos en un ambiente académico usando ontologías, con la finalidad de ofrecer servicios a los usuarios de manera inteligente.

1.4.2. Objetivos específicos

- Diseñar e implementar modelos basados en ontologías para la representación de espacios físicos, tiempo, redes de sensores y personas en un ambiente académico..
- Integrar un sistema de ontologías modulares para la representación de eventos en un ambiente académico con la finalidad de que sea capaz de responder a servicios solicitados por los usuarios.
- Detectar eventos en el ambiente académico (docencia, investigación y difusión) almacenándolos en los modelos semánticos, a partir de datos provenientes de sensores.
- Diseñar e implementar una capa de representación de eventos académicos, su tiempo, espacio físico y los participantes utilizando modelos ontológicos.

1.5. Contribuciones

Las principales contribuciones de este documento son:

- Un sistema de ontologías donde se representa un ambiente académico y los eventos que ocurren dentro de este.
- Un conjunto de ontologías modulares que sirven para representar los elementos puntuales que están involucrados en los eventos ambientales y académicos, como: el tiempo, personas, espacio físico y red de sensores.
- De igual forma un sistema de detección de eventos académicos, programado con base en reglas utilizando un árbol de decisión, que tiene como finalidad proporcionar de cierta inteligencia a un espacio del dominio académico.
- Por último, un sistema de detección de eventos ambientales, que tiene como objetivo ofrecer un espacio sensible al contexto, con ayuda de las variables climáticas; temperatura, humedad y luminosidad.

1.6. Estructura de la tesis

El resto del documento se organiza de la siguiente manera:

- El capítulo 2 presenta las bases teóricas que fundamentan este trabajo, se exponen las definiciones de ontología, ambiente inteligente, eventos y agentes.
- El capítulo 3 expone un estado del arte donde se muestran los avances más importantes que se han logrado con respecto al conocimiento en el área de investigación de este trabajo.
- El capítulo 4 expone el eje fundamental de este trabajo, el cual consta del desarrollo y la metodología de solución implementada. Aquí se muestran los métodos utilizados y sus descripciones para la creación de los modelos ontológicos y la arquitectura del sistema de información encargado de la detección de eventos.
- El capítulo 5 discute la evaluación del modelo ontológico obtenido por medio de casos de uso. Además, se utilizan las métricas conocidas de *precisión*, *exhaustividad* y *valor-f* para la evaluación del sistema de detección de eventos.
- El capítulo 6 contiene las conclusiones generales de este trabajo y presenta las propuestas para trabajos a futuro.

Capítulo 2

Marco teórico

En este capítulo se presentan las bases teóricas que sustentan el proyecto de investigación. A continuación, se describen a detalle los conceptos de ontología, ambiente inteligente, evento y agentes inteligentes.

2.1. Ontologías

El primer concepto que se analiza es “ontología”, etimológicamente significa el conocimiento del ser. Según Gruber [1], “una ontología es una descripción formal de los conceptos y de las relaciones entre ellos”. Conjuntamente, una ontología es la especificación de una conceptualización.

Con ello es posible establecer que una ontología o modelo del dominio, es la representación de la conceptualización explícita compartida de un dominio en particular de acuerdo con Duran [2].

Las ontologías se utilizan para representar conceptos de una manera formal, entendible para las personas y procesable por computadoras. Además, precisa con diversos componentes que sirven para representar el conocimiento, éstos son: conceptos los cuales pueden ser clases de objetos, relaciones que se pueden especializar y formar funciones, instancias o individuos de los conceptos. Finalmente, los axiomas que sirven para realizar inferencias.

La clasificación de ontologías según Guarino [3] se pueden dividir de acuerdo con su nivel de dependencia con una tarea en particular. Los cuatro tipos fundamentales de ontologías son:

- **Ontologías de un dominio**, representan el conocimiento especializado de un dominio o subdominio y son independientes de la aplicación.

- **Ontologías de nivel superior**, describen conceptos generales y fundacionales del conocimiento como las estructuras parte-todo, los procesos o los tipos de objetos.
- **Ontologías de aplicación**, contienen todas las definiciones necesarias para modelar el conocimiento de una determinada aplicación, y generalmente son una especialización de ontologías de dominio.
- **Ontologías de la tarea**, proporcionan un vocabulario sistemático de los términos utilizados para resolver problemas relacionados con las tareas que pueden o no pertenecer al mismo dominio.

El proyecto presentado se basa en el concepto y los elementos de las ontologías definidos por Gruber y se utilizan de tipo aplicación que involucra personas, espacios físicos, eventos y dispositivos o redes de sensores.

Los elementos que conforman a una ontología son los siguientes:

- **Conceptos**: son las ideas básicas que se intentan formalizar. Los conceptos pueden ser clases de objetos, métodos, planes, estrategias, procesos de razonamiento, etc.
- **Clase**: Es un objeto que define una categoría. Describe conceptos en el dominio del discurso.
- **Jerarquía de clases**: compuesta por una colección de clases conectadas por relaciones “es un tipo de” (*class hierarchy*). Pueden ser del tipo *Top-down* (se crean las clases para definir los conceptos generales del dominio, y posteriormente se crean las especializaciones de esto) o *bottom-up* (se crean las clases más específicas, que serán las hojas de la jerarquía, y posteriormente las clases se agrupan en conceptos más generales).
- **Tipo**: Define el tipo de valor (como cadena de caracteres, número, booleano, etc.)
- **Individuos**: se utilizan para representar objetos determinados de un concepto.
- **Relaciones**: representan la interacción y enlace entre los conceptos del dominio. Suelen formar la taxonomía del dominio. Ej. sub clase de, parte de, parte exhaustiva de, conectado a.
- **Propiedades de objeto**: son propiedades que se utilizan para crear relaciones entre individuos (o instancias).
- **Propiedades de dato**: son los atributos que tiene un individuo.
- **Dominio y rango**: se denomina rango a las clases que definen los posibles valores que puede tomar una propiedad. Se denomina dominio de una propiedad al conjunto de clases que tienen asociada aquella propiedad.

- **Axiomas:** son teoremas que se declaran sobre relaciones que deben cumplir los elementos de la ontología. Por ejemplo: Si A y B son de la clase C, entonces A no es subclase de B.

El lenguaje ontológico utilizado en este proyecto es el *OWL Web Ontology Language* y según el sitio web [4], es un lenguaje de etiquetado semántico para publicar y compartir ontologías en la Web. Permite definir el significado de términos en vocabularios y las relaciones entre aquellos términos (ontologías). Es una extensión del lenguaje RDF y emplea las tripletas de RDF, se trata de un lenguaje diseñado para usarse cuando la información contenida en los documentos necesita ser procesada por programas o aplicaciones, en oposición a situaciones donde el contenido solamente necesita ser presentado a los seres humanos. Se estructura en capas que difieren en la complejidad y puede ser adaptado a las necesidades de cada usuario, al nivel de expresividad que se precise y a los distintos tipos de aplicaciones existentes (motores de búsqueda, agentes entre otros).

2.2. AMI (Ambiente inteligente)

La Inteligencia Ambiental (Ambient Intelligence, AMI), consiste en la interacción de las personas dentro de un entorno con múltiples dispositivos tecnológicos. Con el fin de hacer más sencilla la vida de los individuos de una manera transparente, con base en la definición del cómputo ubicuo.

El objetivo final de un ambiente inteligente es que los usuarios mejoren su experiencia dentro de él, con uso de variables del contexto y con un mínimo esfuerzo con base en servicios de personalización, adaptabilidad y predicción, proporcionados por interfaces inteligentes fundamentada en la toma de decisiones racionales y autónomas.

Según Barro [5], un espacio inteligente se define como “un entorno que contiene multitud de dispositivos que trabajan con el fin de proporcionar al usuario acceso a la información y a los servicios”. De acuerdo con Hernández [6], se define un AMI como “un área de la computación que se dirige a tener espacios, tecnológicamente enriquecidos, que proactivamente apoyen a las personas en su vida diaria”.

Los Ambientes inteligentes se dividen en varias categorías según su aplicación.

Una casa inteligente: es un espacio donde por medio de tecnología se puede controlar, monitorear y automatizar diversos servicios, como el encendido de luces, el cierre y apertura de puertas y ventanas, activación de alarmas y aparatos electrodomésticos, con el fin de obtener un mayor confort y seguridad para las personas que la habitan.

Los espacios de trabajo: son considerados oficinas o edificios inteligentes y son lugares donde por medio de la integración de tecnologías se desarrollan actividades la-

2. MARCO TEÓRICO

borales en condiciones óptimas para mejorar la productividad de los empleados y hacer uso eficiente de los recursos disponibles.

Un hospital inteligente: es un lugar donde se ofrecen servicios de salud de forma eficiente mediante el uso de tecnologías y sistemas de información clínicos. Cuenta con tecnología inteligente en todas las áreas, incluyendo sala de emergencias, quirófanos, unidades de cuidados intensivos, áreas de tratamiento tales como radioterapia, suministro central, farmacia e ingeniería biomédica.

Un espacio académico inteligente: es aquel que utiliza la tecnología para ofrecer servicios a una comunidad estudiantil en la investigación o prácticas con el objetivo de mejorar el aprovechamiento de los alumnos y su seguridad, por ejemplo, tener la temperatura adecuada, si se utilizan materiales inflamables dentro de un laboratorio o el monitoreo de los alumnos para brindar un informe actualizado al docente de las actividades realizadas y así poder elegir una mejor estrategia de enseñanza.

Un componente importante dentro de un ambiente inteligente son las redes de sensores, en un artículo realizado por Gascón [7] para la empresa Libelium, las define como una serie de pequeños dispositivos electrónicos que tienen acceso al mundo exterior por medio de sensores. Estos pueden medir las condiciones fisiológicas del individuo para detectar su estado físico como su temperatura corporal, señal cardíaca, movimiento ocular y presión arterial. Para ello, existen varios tipos de sensores:

- Los sensores de tipo biométrico permiten identificar a los individuos por medio de su huella dactilar, retina, RFID (identificación por radiofrecuencia) o reconocimiento facial.
- Los sensores de localización permiten conocer la posición actual de un individuo en un espacio cerrado por medio de proximidad (infrarrojos) o triangulación de radiofrecuencia (RFID y Bluetooth) y en un espacio abierto (GPS, Galileo, GLONASS).
- Los sensores de movilidad permiten detectar los movimientos realizados en una actividad física como el acelerómetro, podómetro y giróscopo.
- Los sensores ambientales permiten conocer las condiciones del entorno al evaluar la humedad, temperatura, nivel de luz, tiempo y contaminación acústica.

El tipo de redes descrito permiten monitorizar cualquier entorno por inaccesible que parezca. Fei Hu [8] describen una serie de aplicaciones de las redes de sensores, por ejemplo, la detección de contaminación, incendios forestales, monitoreo de terremotos, detección de inundaciones, entre otras, aplicaciones importantes que facilitan el entorno en que vivimos.

El proyecto presentado se centra en un espacio académico, donde la idea principal es dotarlo de servicios basados en eventos de actividades académicas y de investigación para ofrecer a los usuarios . Los eventos son detectados y representados en ontologías, y al ofrecer servicios a los usuarios lo convierte en un espacio académico inteligente.

2.3. Eventos

Según Barranco [9], los eventos son definidos como la ocurrencia dentro de un sistema o dominio particular; algo que ocurre o que sucederá en dicho dominio. La detección y procesamiento de estos eventos es algo que se utilizará para poder visualizarlos en tiempo real.

De acuerdo con Reyes-Ortiz [10], los eventos ayudan a organizar la información y otorgan un orden espacio-temporal. Los eventos han sido definidos, como:

- En Miller [11], un evento se define como un suceso que involucra un cambio de estado, donde se involucran aspectos locativos, temporales y causales.
- Allen et al. [12], exponen que el mundo contiene eventos que son el camino por el cual los agentes clasifican ciertos patrones de cambio. Los eventos tienen propiedades esenciales como el tiempo, los efectos y las causas.
- Galton y Augusto [13], presentan una definición desde el punto de vista del campo de la representación de conocimiento, en la cual afirman que todos los eventos están dados de acuerdo a intervalos e instantes de tiempo y que involucran una causalidad.
- Sowa [14], afirma que un evento es una entidad que puede involucrarse en la causalidad y que puede ser identificado por su ubicación en una región del espacio tiempo.
- Finalmente, Zwaan [15], expone que los eventos se organizan en cinco índices: espacialidad, temporalidad, protagonistas, causalidad e intencionalidad.

Con estos índices, los eventos son organizados sobre un marco temporal en el cual ocurren, una región espacial donde suceden, el protagonista involucrado, su estado causal con respecto a eventos anteriores y su relación con el objetivo de los protagonistas.

Las definiciones de eventos son diversas desde diferentes perspectivas [9], sin embargo, todas comparten un significado común, el cual se considera el concepto de evento como un suceso, hecho, o acción que se caracteriza por dimensiones, como: protagonistas, espacialidad, temporalidad y causalidad e intencionalidad. Los eventos consideran

las dimensiones de espacio, tiempo, causalidad y protagonistas que son parte de eventos como se muestra en la Figura (2.1).



Figura 2.1: Eventos y sus dimensiones.

En este trabajo se consideran eventos relacionados con la investigación, docencia y difusión de la cultura, los cuales serán detectados con el apoyo de redes de sensores en el espacio académico. Las ontologías serán utilizadas para la representación de eventos académicos junto con su espacio y tiempo, debido a las ventajas de interoperabilidad, comunicación e inferencia que ellas ofrecen.

2.4. Agentes inteligentes

Se puede definir a un agente como un software desarrollado que utiliza ciertas técnicas o herramientas de inteligencia artificial, tiene como propósito percibir el entorno y actuar de una manera autónoma, es decir, con un grado de racionalidad e independencia. Para lograr esto el agente debe ser capaz de comprender el contexto, basarse en su propio conocimiento y tener mecanismos para realizar acciones; con el fin de brindar servicios a los usuarios o a otro programa, esto es satisfacer las necesidades de éstos.

Varios agentes pueden trabajar conjuntamente para resolver de manera eficaz una tarea o un propósito específico que el usuario final requiere. Las principales propiedades de un agente son: autonomía, sociabilidad, capacidad de reacción e iniciativa, las cuales según Vargas-Quesada [16] se enlistan a continuación.

- **Autonomía:** actuar sin ningún tipo de intervención humana directa y tener control sobre sus propios actos.
- **Sociabilidad:** comunicarse por medio de un lenguaje común con otros agentes, e incluso con los humanos.
- **Capacidad de reacción:** percibir su entorno y reaccionar para adaptarse a él.
- **Iniciativa:** emprender las acciones para resolver un problema.

Para poder conseguir una adecuada comunicación entre agentes es indispensable que “hablen” el mismo idioma. Es decir, tener un protocolo de comunicación en común para poder intercambiar mensajes y así poder desarrollar las tareas que les fueron encomendadas ya sea de manera individual o en conjunto.

Como se muestra en la Figura (2.2) los agentes tienen un ciclo de vida y una vez eliminado no es posible volver a contar con dicho agente. A lo largo del proyecto fueron utilizados agentes desarrollados en *JADE* para hacer la simulación de datos de entrada provenientes de una red de sensores.

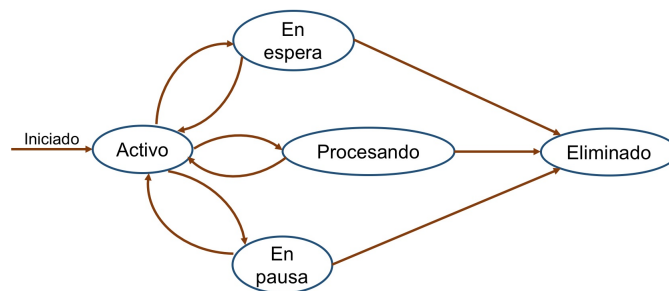


Figura 2.2: Ciclo de vida de los agentes.

Capítulo 3

Estado del arte

En este capítulo se presenta una revisión del estado del arte en el tema de la detección de eventos en distintos ámbitos o dominios, así como las herramientas utilizadas para su representación. A continuación, se describen los trabajos analizados y se presenta una comparativa con respecto al tema principal que se aborda este trabajo.

3.1. Descripción de los artículos

Skocir, P. et al. [17] **Activity detection in smart home environment**. Con este trabajo del 2016, los autores proponen el análisis de datos en un ambiente inteligente, donde se pueda controlar la iluminación o el sistema de calefacción, pero se centra particularmente en la detección de entradas y salidas desde una habitación.

Con esta información, se pretende controlar el bienestar de los usuarios, (por ejemplo, si el usuario permanece demasiado tiempo en una cama en su dormitorio podría implicar un problema con la salud del usuario). Infiriéndose mediante la captura y análisis de datos con ayuda de sensores inteligentes de movimiento. Utiliza algoritmos heurísticos de segmentación *MaxGap* y *MaxGain* y redes neuronales para el procesamiento de los datos.

Shinmoto, R. et al. [18]. **A hierarchical model for recognizing alarming states in a batteryless sensor alarm intervention for preventing falls in older people**. En este trabajo de abril de 2017 se utiliza un modelo jerárquico de campos aleatorios condicionales para predecir estados alarmantes (estar fuera de la cama o silla). Se utiliza un dispositivo sensor portátil llamado *Wearable* para proporcionar una intervención para reducir las caídas.

Los autores proponen un algoritmo de clasificación para aprender a reconocer en tiempo real actividades, correspondientes a los estados alarmantes de un paciente que sale de una cama y que está fuera de cama, si está en una silla o si sale de la misma.

Este algoritmo está basado en un método que calcula las probabilidades marginales para cada sensor de datos recibidos CRF (HCRF), este algoritmo de aprendizaje utiliza información basada en el dominio del tiempo a partir del acelerómetro y los datos RFID obtenidos de la Infraestructura RFID.

Meditkos, G. & Kompatsiaris, I. [19] **iKnow: Ontology-driven situational awareness for the recognition of activities of daily living**. Es un modelo basado en una ontología para la comprensión del contexto en ambientes con múltiples sensores que son utilizados para el reconocimiento de la actividad humana. IKnow utiliza el uso del conocimiento ontológico OWL para capturar relaciones de dominio entre Observaciones y actividades.

Chen, L. & Nugent, C. [20]. **Ontology-based Activity Recognition in Intelligent Pervasive Environments**. Describen un modelo basado en ontologías para el reconocimiento de actividades. El enfoque propuesto adopta ontologías para el modelado de sensores, objetos y actividades, además explota el razonamiento lógico para los propósitos de la actividad de reconocimiento realizadas en un ambiente inteligente para ser más precisos en una *Smart Home*.

Para cumplir este propósito se utilizan algoritmos de reconocimiento de actividad, que pueden dividirse en técnicas de aprendizaje, se incluyen métodos de aprendizaje supervisados y no supervisados, que utilizan principalmente razonamiento probabilístico y estadístico. El proyecto de investigación desarrollado conlleva una cierta similitud con este a diferencia el segundo se delimita en un ambiente académico.

Hromic, H. et al. [21] **Real Time Analysis of Sensor Data for the Internet of Things by means of Clustering and Event Processing**. Los autores ofrecen la propuesta de una aplicación móvil para la solución de la calidad del aire mediante el análisis e interpretación en tiempo real de los datos obtenidos mediante sensores móviles de temperatura, humedad, presión atmosférica y niveles de contaminantes atmosféricos todo esto bajo el concepto del internet de las cosas.

Stocker, M. et al. [22]. **Abstractions from Sensor Data with Complex Event Processing and Machine Learning**. Presentada en junio de 2014 es un modelo de software que representa una máquina de aprendizaje para el monitoreo de un ambiente, utiliza sensores y representa el conocimiento de los eventos adquiridos, o extraídos, de los datos del sensor, con ayuda de módulos de aprendizaje.

Por lo tanto, su funcionalidad principal es para la evaluación de la situación, es decir para ganar conciencia de la situación. **Wavellite** también se basa en la representación del conocimiento y el razonamiento, con el uso de ontologías. A diferencia del trabajo presentado en esta tesis Wavellite es utilizado para el aprendizaje del movimiento de los vehículos que pasan sobre una carretera y utiliza Weka para el aprendizaje automático.

Calbimonte, J. P. et al. [23] **Semantic representation and processing of hypoglycemic events derived from wearable sensor data.** Presentan un framework para inferir eventos semánticamente de glucosa obtenidos de un paciente, mediante datos de sensores móviles desplegados en un cinturón deportivo. El trabajo es parte del proyecto **D1namo** para monitoreo de la diabetes, se centra en la representación y el procesamiento de consultas de datos producidos por los sensores portátiles, con uso de tecnologías semánticas. Además, utiliza un motor de procesamiento de flujo RDF.

Gu, T. et al. [24] **An Ontology-based Context Model in Intelligent Environments. SOCAM,** una arquitectura de *middleware* orientada a servicios para apoyar la construcción de los servicios sensibles al contexto en entornos inteligentes. Apoyado por un modelo semántico basado en una ontología para abordar cuestiones como la representación del contexto, razonamiento y el intercambio de conocimientos.

El principal beneficio de este modelo es la capacidad de razonar sobre diversos contextos para abordar cuestiones como la representación del contexto semántico, el razonamiento del contexto. El trabajo se desarrolla en un escenario de contexto familiar, esto es un hogar que está equipado con varios sensores / actuadores en red, entre los que destacan dispositivos como cámaras, micrófonos, RFID (Radio Detección de la frecuencia), y sensores de cortina.

Hernández-Calderón, J. G. et al. [6] proponen en su artículo. **Un modelo de ambientes educativos inteligentes.** Una arquitectura conceptual para sistemas que soporten modelos académicos y tecnológicamente enriquecidos, que proactivamente apoyen a las personas en su vida diaria. Dada la riqueza de información y conocimientos existentes en ámbitos educativos.

Su arquitectura se divide en cuatro capas que son: capa de elementos físicos y lógicos propios del entorno, capa de reconocimiento y acción, capa de análisis y capa de datos. El modelo propuesto considera los elementos fundamentales que se encuentran en este tipo de sistemas (actores, tales como maestros y alumnos, sus actividades, así como los objetos que manipulan). La arquitectura conceptual propone componentes abstractos que soportan los elementos del modelo.

Camacho, D. & Novais, P. [25], **Innovations and practical applications of intelligent systems in ambient intelligence and humanized computing.** En este artículo presentan una serie de recopilaciones de trabajos recientes de las innovaciones y aplicaciones de hoy en día en sistemas inteligentes ambientales y computación humanizada. Esto para conocer el contexto de lo que está desarrollado hoy en día. Algunos de los que enlista son: el documento titulado “Hacia la mejora de la extracción de características y la clasificación para el reconocimiento de actividades en la transmisión de datos” escrito por Nawel Yala.

Propone por medio de sensores para realizar reconocimiento de actividad en una configuración. Línea / transmisión; para reconocer las actividades cuando se registra un nuevo evento del sensor. Otro es el trabajo titulado "La adquisición del contexto en los estudios de reconocimiento emocional auditivo", escrito por Carneiro, D. et al. En el que describe un entorno para acceder al reconocimiento emocional auditivo basado en un dispositivo móvil.

Integration Of Internet-Of-Things Facilities And Ubiquitous Learning For Still Smarter Learning Environment [26]. Este artículo del 2016, explora cómo el internet de las cosas puede contribuir a entornos de aprendizaje y enseñanza inteligentes y sostenibles. Se fundamenta en un entorno de aprendizaje en línea, un área aún emergente, que hace que las personas accedan a las computadoras sin límite de tiempo y en cualquier lugar, y proporciona servicios de aprendizaje personalizados para el individuo.

El trabajo tiene como objetivo tomar conciencia del desarrollo de sistemas de inteligencia ambiental centrados en el ser humano, para los modelos de ingeniería y de computación. El resultado del documento dará un paso adelante hacia un objetivo común comprensión de cómo los campos interdisciplinarios pueden contribuir al IoT e investigar todo con él, esto es tener siempre a la mano un dispositivo que permita acceder a la red de aprendizaje de un tema en específico. Un entorno de aprendizaje ubicuo permite el aprendizaje sin interrupciones en cualquier lugar y en cualquier momento

Durães, D. et al. [27] en **Modelling an Intelligent Interaction System for Increasing the Level of Attention**, proponen un sistema inteligente ambiental, que se ha diseñado con base en comportamientos biométricos para detectar la falta de atención del alumno. La atención de aprendizaje de un estudiante se puede determinar con precisión y el profesor tiene acceso a estos resultados y con ello puede mejorar las estrategias de enseñanza, los datos se adquieren de rutinas de trabajo normales, a través del mouse y el teclado, que actúan como sensores. Para describir los patrones de comportamiento de los estudiantes y recibir los datos de los eventos.

La Tabla 3.1 muestra una comparativa entre los artículos antes descritos y el tema de este proyecto de investigación, la tabla contiene los conceptos clave utilizados por cada autor y así como las herramientas utilizadas para llevar a cabo la investigación.

Tabla 3.1: Comparación de artículos analizados con respecto al proyecto desarrollado

| | Reconocimiento de eventos | Enfoque | Tipo de ambiente | Fuente de Datos | Manejo de Ontologías |
|---|---------------------------|---|----------------------|---|----------------------|
| Pavle Skocir, Krvic, Matea Tomejnak, Mario Kusek, Gordan Jezic[17] | Sí | Algoritmos heurísticos | Edificio Inteligente | Sensores, temperatura, luminosidad e identificación | No |
| Roberto Luis Shinmoto Torres a, Qinfeng Shi, Anton van den Henge[18] | Sí | Algoritmos probabilísticos | Edificio inteligente | Wearable | No |
| G. Meditskos and I. Kompatsiaris [19] | Sí | Minería y Reglas | Smart Home | Sensores de actividad | Sí |
| Liming Chen, Chris Nugent [20] | Sí | Algoritmos de reconocimiento | Smart Home | Sensores de actividad | Sí |
| H. Hromic, D. Le Phuoc, M. Serrano, A. Antonic, I. Zarko, C. Hayes and S Decker, [21] | Sí | Análisis e interpretación de datos | N/A | Celular | No |
| Markus Stocker, Mauno Rönkkö, and Mikko | Sí | Weka, aprendizaje automático | N/A | Sensores de ambiente | Sí |
| J. Calbimonte, J. Rauviera, F. Dubosson and K. Aberer, [23] | Sí | Minería y reglas | Hospital inteligente | Wearable | Sí |
| T. Gu, X. Hang, H. Keng and D. Qing [24] | Sí | Minería y reglas | Casa inteligente | Sensores, temperatura, luminosidad e identificación | Sí |
| José Guillermo Hernández Calderón, Edgard Benítez Guerrero, Carmen Mezura Godoy [6] | No | Minería y reglas | Espacio educativo | Sensores | No |
| David Camacho, Paulo Novais [25] | Sí | Minería y reglas | Varios | Diferentes tipos de sensores | No |
| Kanagarajan, Sivakumar Ramakrishnan [26] | Sí | Minería y reglas | Varios | Computo Ubicuo | Sí |
| Dalila Duraes, David Castro, Javier Bajo, and Paulo Novais [27] | Sí | Algoritmos estadísticos | Ambiente académico | Mouse y Teclado | No |
| Este Proyecto | Sí | Reglas para detectar eventos académicos | Espacio académico | identificación, agendas de actividades y perfiles | Sí |

3. ESTADO DEL ARTE

Los puntos relevantes que hacen una aportación en el campo de la detección de eventos y la representación de conocimiento, es la creación de un modelo ontológico genérico que abarca el ámbito académico y que además puede ser adaptable a otro tipo de ambiente. Como se puede observar en los trabajos relacionados, las investigaciones actualmente se centran en otro tipo de dominio.

Asimismo, el uso de reglas e inferencia para la detección de eventos es un enfoque que ha sido probado por distintas investigaciones lo que demuestra que es una característica clave en el área de interés.

Metodología de solución

En este capítulo se expone el trabajo elaborado para alcanzar el objetivo de este proyecto. Se presenta como se llevó a cabo el modelo ontológico, su poblado y además, las fases para conseguir la detección de eventos dentro de un ambiente académico inteligente.

4.1. Metodología implementada

Para lograr el objetivo principal se implementó la arquitectura de solución que se muestra en la Figura (4.1) posteriormente se exponen las etapas de dicha metodología.

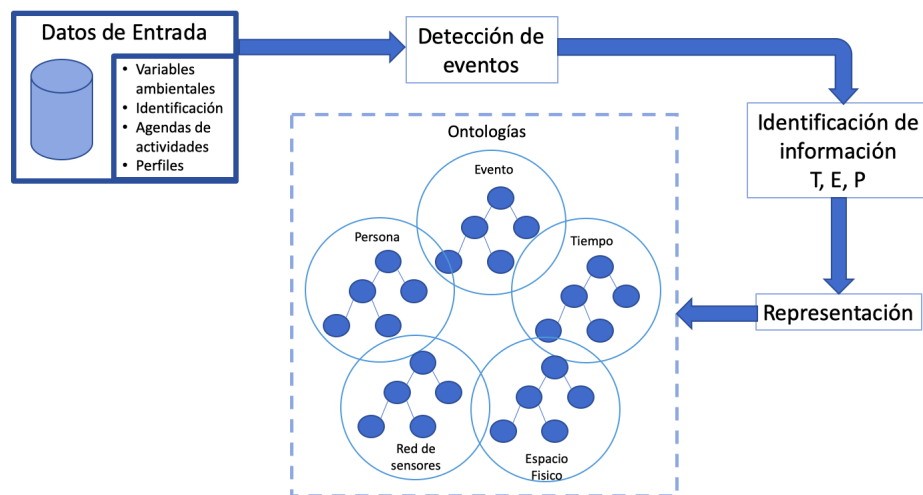


Figura 4.1: Arquitectura de solución.

La metodología implementada se compone de cuatro fases, las cuales se describen a continuación.

4.1.1. Fase 1 Modelo de eventos

En la primera fase se desarrolla un modelo ontológico capaz de almacenar los eventos ocurridos en un espacio académico inteligente, el modelo ontológico consta de cinco ontologías, que son tiempo, espacio, persona, red de sensores y evento, después de desarrollarlas e implementarlas se unificaron en un solo modelo semántico. Esta fase incluye el diseño y codificación de los componentes del modelo ontológico para la representación de eventos.

4.1.2. Fase 2 Detección de eventos

En esta fase se detectan los eventos que ocurren en un espacio académico, tales como: docencia (impartir asesorías, impartir clases, dirigir práctica de laboratorio); investigación (redactar artículos, lectura de artículos, dirigir experimentos científicos); y difusión (seminarios, conferencias, reuniones de proyectos de investigación). Para identificar y clasificar estos eventos en el espacio académico, se parte de datos sobre temperatura, identificación, humedad, luminosidad, perfiles de usuario y agendas de actividades. Estas tareas conforman la detección de eventos que está basada en un conjunto de reglas de decisión.

4.1.3. Fase 3 Identificación de componentes de los eventos

Esta fase tiene como punto de partida el tipo de evento detectado y la información de los sensores. El objetivo de esta fase es complementar el evento con los datos de dónde ocurrieron, cuándo sucedió el evento y quienes están involucrados. La salida de esta fase es un conjunto de eventos con sus componentes ocurridos en el espacio académico.

4.1.4. Fase 4 Representación de información en un sistema de ontologías

En esta fase se utilizan los datos de los eventos y sus complementos para representar la información por medio de un modelo semántico haciendo uso de ontologías. El objetivo es crear instancias ontológicas en la clase de evento correspondiente, esta etapa genera el modelo ontológico con la representación de los eventos acaecidos en el espacio académico.

4.2. Modelo ontológico

En esta sección se presenta el proceso de diseño del sistema ontológico llamado ambiente inteligente que tiene como propósito la identificación y representación de

eventos en un dominio académico, el sistema está compuesto de una serie de ontologías modulares, las cuales dependen de un dominio específico: persona, tiempo, ubicación (espacio físico), red de sensores y evento.

4.2.1. Ontología de espacio físico

La ontología de espacio físico se refiere al modelo donde diariamente ocurren actividades académicas, se relacionan con lugares donde existen estructuras e instalaciones necesarias para el funcionamiento de una organización en este caso de tipo académica, con este modelo se puede encontrar la ubicación de los actores que están inmersos dentro de un entorno académico. En tal ubicación hay dos tipos de espacios como se muestra en la taxonomía de la Figura 4.2; los espacios cerrados que se encuentran delimitados por divisiones físicas en estos se enlistan los cubículos, laboratorios, aulas, auditorios, oficinas administrativas, auditorios, edificios, salones de clase, comedores y baños.

Los espacios abiertos son aquellos que carecen de una estructura física y que son afectados por el medio ambiente. Están divididos en: corredores o pasillos, áreas verdes, estacionamientos y plazas.

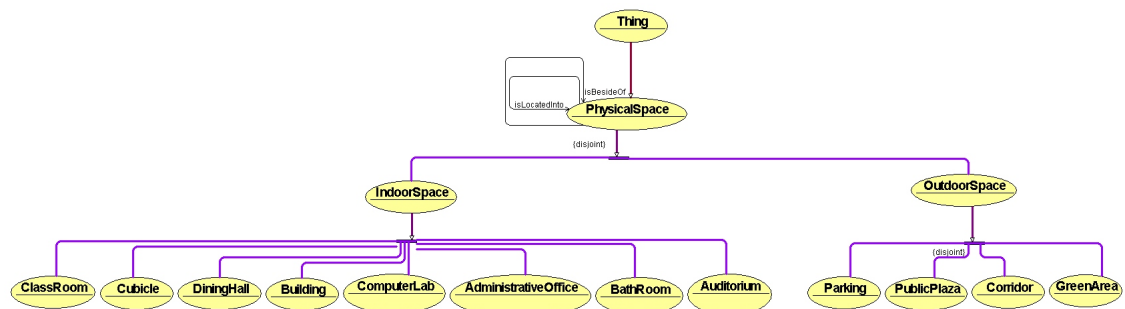


Figura 4.2: Taxonomía de espacio físico

Una vez adquirida la conceptualización de los lugares del ámbito académico a representar se crearon las clases de la ontología espacio físico y sus conceptos están descritos en la Tabla 4.1. Se pueden ubicar dos espacios físicos juntos es decir que uno este muy cercano al otro, es aquí donde se produce una propiedad de objeto llamada (*isBesideOf*), además es posibles que un espacio sea parte o este contenido dentro de otro, de aquí se obtiene la propiedad de objeto (*isLocatedIn*).

La ontología de espacio físico tiene un total de 13 propiedades de datos: *hasAirConditioner*, *hasArea*, *hasCarCapacity*, *hasDoorState*, *hasFan*, *hasLampsNumber*, *hasLevel*, *hasNamePhysicalSpace*, *hasPeopleCapacity*, *hasProjectionScreen*, *hasProjector*, *hasService* y *hasWindow*. Sirven para tener una mejor representación de los espacios,

4. METODOLOGÍA DE SOLUCIÓN

con estas propiedades se conoce en qué condiciones se encuentra el lugar, es decir, saber si el espacio cuenta con lámparas, proyectores o si tiene algún tipo de ventilación en caso de que sea un espacio cerrado, también se obtienen las dimensiones y tipo de servicio que brinda cada lugar, además de obtener su ubicación precisa, por ejemplo conocer en qué edificio se localiza tal espacio, el nivel en el que se encuentra y si el lugar es conocido con algún nombre en particular, ejemplo “laboratorio de cómputo Babbage”, ubicado en el edificio T segundo nivel dentro de la Universidad Autónoma Metropolitana unidad Azcapotzalco Figura 4.4.

La ontología de espacio fue implementada en el Lenguaje Web Ontológico (*OWL*) 2.0. Para representar los siguientes elementos ontológicos: 14 clases, 11 subclases, 13 relaciones de tipo de dato, 2 relaciones de objeto y 33 instancias. En la Figura 4.3 se observa un fragmento de la implementación en el meta-lenguaje *XML*.

```
<?xml version="1.0"?>
<Ontology xmlns="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.semanticweb.org/usuario/ontologies/2016/0/PhysicalSpace"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  ontologyIRI="http://www.semanticweb.org/usuario/ontologies/2016/0/PhysicalSpace">
  <Prefix name="" IRI="http://www.semanticweb.org/usuario/ontologies/2016/0/PhysicalSpace"/>
  <Prefix name="owl" IRI="http://www.w3.org/2002/07/owl"/>
  <Prefix name="rdf" IRI="http://www.w3.org/1999/02/22-rdf-syntax-ns"/>
  <Prefix name="xml" IRI="http://www.w3.org/XML/1998/namespace"/>
  <Prefix name="xsd" IRI="http://www.w3.org/2001/XMLSchema"/>
  <Prefix name="rdfs" IRI="http://www.w3.org/2000/01/rdf-schema"/>
  <Declaration>
    <DataProperty IRI="#hasPeopleCapacity"/>
  </Declaration>
  <Declaration>
    <NamedIndividual IRI="#auditoriumW003"/>
  </Declaration>
  <Declaration>
    <NamedIndividual IRI="#parking2"/>
  </Declaration>
  <Declaration>
    <NamedIndividual IRI="#buildingT"/>
  </Declaration>
  <Declaration>
    <NamedIndividual IRI="#classRoomE313"/>
  </Declaration>
  <Declaration>
    <DataProperty IRI="#hasArea"/>
  </Declaration>
  <Declaration>
    <NamedIndividual IRI="#corridorHE1"/>
  </Declaration>
  <Declaration>
    <NamedIndividual IRI="#cubicleHP287b"/>
  </Declaration>
  <Declaration>
    <Class IRI="#Building"/>
  </Declaration>
  <Declaration>
    <Class IRI="#BathRoom"/>
  </Declaration>
  <Declaration>
    <NamedIndividual IRI="#diningHall1"/>
  </Declaration>
  <Declaration>
    <DataProperty IRI="#hasProjectionScreen"/>
  </Declaration>
  <Declaration>
    <NamedIndividual IRI="#publicPlazaRoja"/>
  </Declaration>
  <Declaration>
    <Class IRI="#PhysicalSpace"/>
  </Declaration>
</Declaration>
```

Figura 4.3: Fragmento de código OWL 2.0

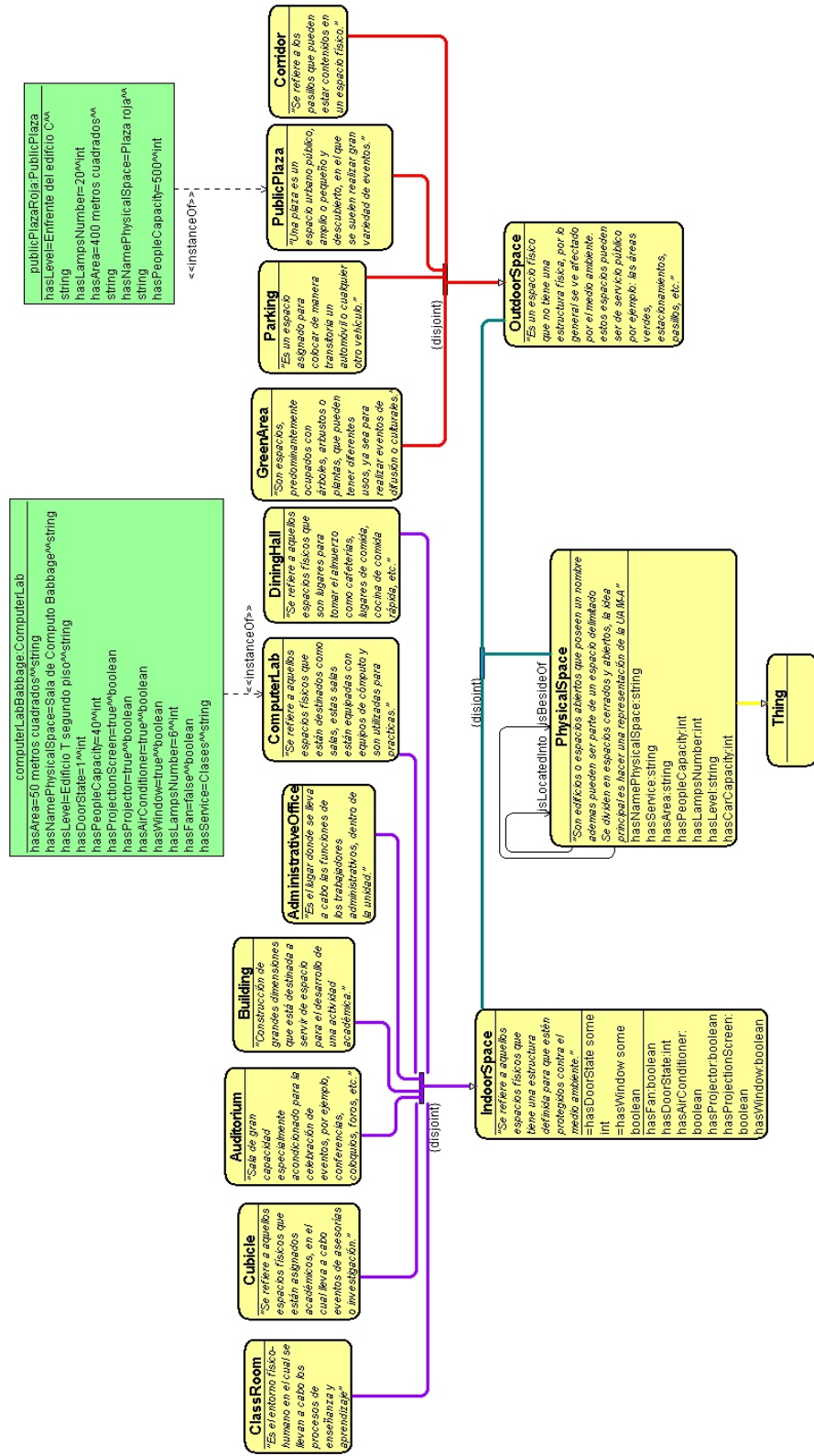


Figura 4.4: Ontología de espacio físico

4. METODOLOGÍA DE SOLUCIÓN

Tabla 4.1: Clases y conceptos de la ontología de espacio físico

| Clase | Descripción |
|----------------------|---|
| PhysicalSpace | Son los lugares dentro de un entorno académico estos se dividen en espacios cerrados y abierto. |
| IndoorSpace | Se refiere a aquellos espacios delimitados por una estructura bien definida, además cuentan con servicios como: iluminación, puertas, ventanas, por mencionar algunos. |
| AdministrativeOffice | Es el lugar donde se llevan a cabo las funciones de los trabajadores administrativos. |
| Auditorium | Espacio de gran capacidad especialmente acondicionado para la celebración de eventos, por ejemplo, conferencias, coloquios y foros. |
| Building | Construcción de extensas dimensiones destinado a servir como un espacio para el desarrollo de una actividad académica. |
| ClassRoom | Es el entorno físico-humano en el cual se llevan a cabo los procesos de enseñanza y aprendizaje. |
| ComputerLab | Son aquellos espacios físicos que están destinados como salas, equipadas como una combinación de equipos de cómputo. |
| Cubicle | Es un espacio físico que está asignado a un profesor o grupo de profesores, en el cual se llevan a cabo eventos de asesorías o investigación. |
| DiningHall | Se refiere a aquellos lugares que están destinados para el consumo de alimentos. |
| OutdoorSpace | Son espacios físicos que no tienen una estructura física, estos espacios pueden ser de servicio público, entre los que destacan las áreas verdes, estacionamientos, pasillos. |
| Corridor | Se refiere a los pasillos que interconectan edificios en el espacio académico. |
| GreenArea | Son espacios predominantemente ocupados con vegetación, pueden tener diferentes usos por ejemplo para realizar eventos de difusión o actividades culturales. |
| Parking | Es un espacio asignado para colocar de manera transitoria un automóvil o cualquier otro vehículo. |
| PublicPLaza | Una plaza es un espacio público, amplio y descubierto, en el que se suelen realizar una gran variedad de eventos académicos. |

4.2.2. Ontología de red de sensores

Entre la información más relevante dentro de un ambiente inteligente se encuentran los datos ambientales relacionados con el clima. Para obtener esta información es necesario contar con datos provenientes de sensores de temperatura, humedad y luminosidad. Otro tipo de información relevante se refiere a la presencia de personas en el ambiente, para obtener la información de la entrada y salida de las personas se utilizan sensores de presencia. Ambos tipos de información son contemplados en el modelo ontológico que representa la de red de sensores.

Esta ontología es una abstracción de una red de sensores real que está compuesta de dispositivos, nodos y tipos de comunicación que existe entre ellos. En la Figura 4.5 se muestra el detalle de la taxonomía de las clases utilizadas en este modelo ontológico y en la Tabla 4.2 se describen a detalle los conceptos utilizados en la ontología.

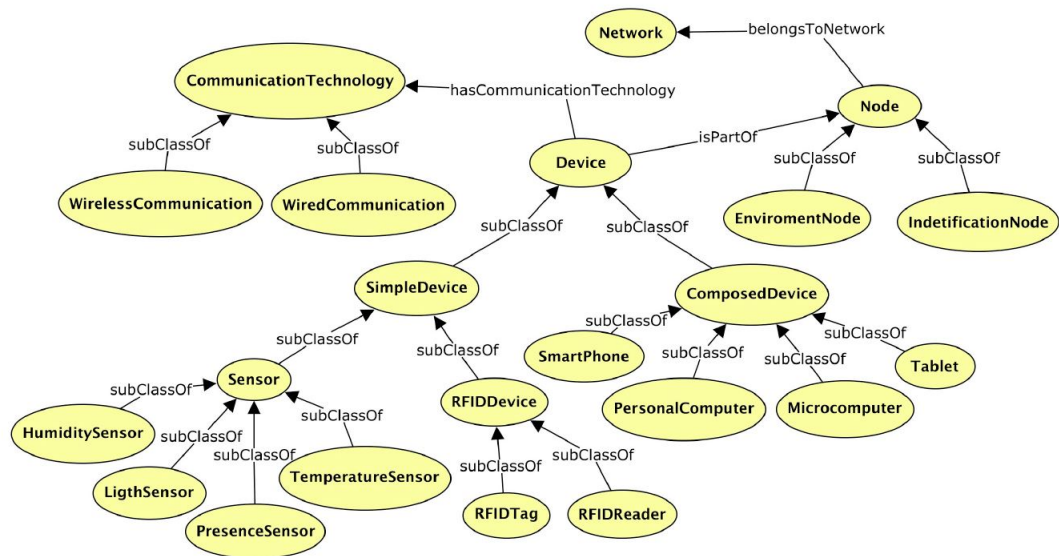


Figura 4.5: Taxonomía red de sensores

La ontología red de sensores tiene dos objetivos principales:

- Representar los conceptos y propiedades de datos que permitan identificar y registrar la presencia de personas (estudiantes, empleados o visitantes) automáticamente, si están entrando o saliendo en un espacio físico específico.
- Representar los conceptos que permitan detectar e inferir sobre eventos simples relacionados con el medio ambiente.

Para lograr estos propósitos, se incluyeron 14 propiedades de datos en el modelo ontológico, los cuales son: *hasTag*, *hasCoverage-Range*, *hasIDCardNumber*, *hasIDCar-*

4. METODOLOGÍA DE SOLUCIÓN

dRead, hasIDNode, hasIDNumber, hasIMEI, hasIPAddress, hasMacAddress, hasModel, hasPhoneNumner, hasProtocolVersion, hasState and isPortable.

Tabla 4.2: Elementos ontológicos del modelo red de sensores.

| Sensor | Descripción |
|-----------------------|---|
| WiredCommunication | Esta clase se refiere a las tecnologías de comunicación en los que la transferencia de datos entre dos o más dispositivos están conectados por un conductor eléctrico. |
| WirelessCommunication | Comunicación inalámbrica se entiende la capacidad que tiene un dispositivo de enviar o recibir datos sin la necesidad de un enlace cableado. |
| Microcomputer | Un microcomputadora es una computadora completa en una escala más pequeña y generalmente el término más común es computadora personal o PC. |
| PersonalComputer | Una PC es una computadora de propósito general utilizada diariamente por una persona a la vez para trabajo o uso personal. PC es una abreviatura de computadora personal. |
| SmartPhone | Un teléfono inteligente es un teléfono móvil que realiza muchas actividades similares a las funciones de una computadora. |
| Tablet | Una tablet es un tipo de computadora que tiene una pantalla con la que se puede interaccionar directamente. |
| Actuator | Genéricamente se conoce con el nombre de actuadores a los elementos finales que permiten modificar variables a controlar en una instalación automatizada. Los actuadores son los elementos del circuito que transforman las señales eléctricas en los efectos deseados. |
| RFIDReader | Un lector RFID es un dispositivo que se utiliza para interrogar una etiqueta RFID. El lector tiene una antena que emite ondas de radio y la etiqueta responde devolviendo sus datos. |
| RFIDTag | Un RFID Tag es un microchip combinado con una antena en un paquete compacto; El empaquetado está estructurado para permitir que la etiqueta RFID sea unida a un objeto a rastrear. RFID" significa identificación por radiofrecuencia. |
| Sensor | Es un dispositivo que permite la recogida de datos del medio en tiempo real. Entre las diferentes variables a medir se pueden encontrar, temperatura, iluminación, humedad y presencia. |
| ActuatorNode | Se refiere a todo aquellos elementos que intervienen en el control y cuya misión es realizar alguna función como respuesta a una necesidad del control (función de salida) como son: regular la cantidad de luz y la temperatura. |
| ConcentratorNode | SupervisorNode es aquel nodo encargado de concentrar toda la información de los diversos sensores conectados a la red (Raspberry). |
| EnviromentNode | EnvironmentNode son aquellos dispositivos con capacidad de leer datos del ambiente como: temperatura, humedad, iluminación (recolector de datos). |

Además, la ontología incluye las propiedades de objeto que permiten indicar a qué red pertenece un nodo: (*belongsTo-Network*), la propiedad (*hasCommunicationTechnology*) relaciona el dispositivo con su tecnología de comunicación correspondiente y finalmente una propiedad que sirve para conocer el nodo al que pertenece un dispositivo (*isPartOf*). Para este modelo fue necesario crear 155 individuos. Un ejemplo de ellos es mostrado en la Figura 4.6, el individuo *NS01* modelado es del tipo nodo ambiental y está compuesto por sensores de humedad, luminosidad y temperatura, su tecnología de comunicación es wifi y su dirección ip privada es la 192.168.1.204.

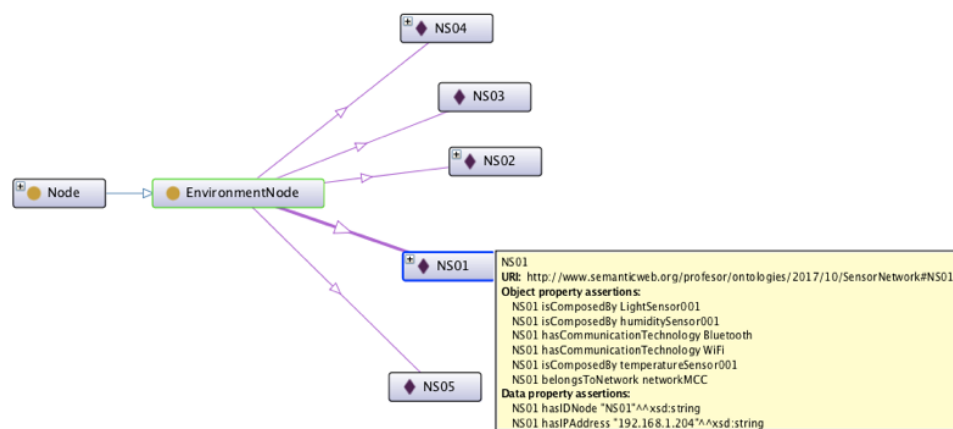


Figura 4.6: Individuo de la ontología red de sensores

4.2.3. Ontología de tiempo

El modelado del tiempo en una ontología es de propósito general, es decir, es una ontología de nivel alto que describe un concepto muy frecuentemente requerido. En este caso el tiempo, permite definir eventos y correlacionarlos con el instante cuando han ocurrido los eventos en el ambiente académico.

La ontología de tiempo está compuesta por dos clases que son, intervalo que a su vez se componen de instantes, uno para indicar cuando inicia el evento, este se representa con la propiedad de objeto *hasBeginning* y el segundo intervalo se utiliza para determinar cuándo termina el evento: *hasEnd*.

Por otro lado, un instante tiene seis datos personalizados para representar un momento completo en el calendario AAAA-MM-DD y HH:MM:SSMM:SS, para esta representación se utilizaron las propiedades de dato: *hasYear*, *hasDay*, *hasMonth*, *hasHour*, *hasMinute*, *hasSecond*. En la Figura 4.7 se muestra la ontología de tiempo.

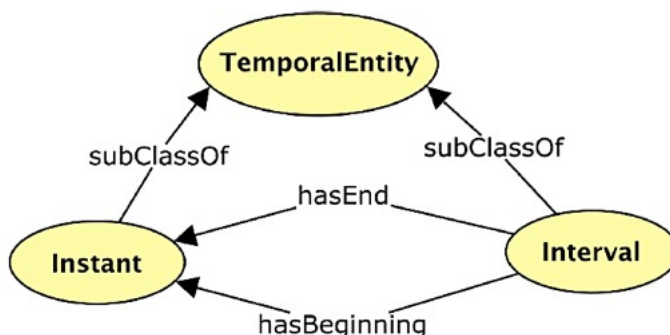


Figura 4.7: Ontología de tiempo

En la Tabla 4.3 se muestra el número de los elementos ontológicos desarrollados para el modelo de tiempo.

Tabla 4.3: Elementos ontológicos del modelo tiempo.

| Número de clases | Número de subclases | Número de relaciones de tipo de datos (dataproperty) | Número de relaciones de objetos | Número de instancias |
|------------------|---------------------|--|---------------------------------|----------------------|
| 3 | 2 | 5 | 2 | 10 |

4.2.4. Ontología de persona

Un elemento fundamental del modelado de eventos es la representación o identificación del actor que participa o al que le sucede un evento. Sin la representación de los actores en el evento no se tendría el conocimiento de quien está involucrado dentro de un acontecimiento. Implementar un modelo completo para la representación de personas puede llegar a ser demasiado extenso, es por eso que en este trabajo se ha acotado a un modelo donde solamente se representan los actores que participan dentro del dominio académico, como son: estudiantes, empleados tanto administrativos como profesores y visitantes que también tienen un rol importante dentro de este ambiente académico. La jerarquía de clases de la ontología de persona se observa en la Figura 4.8.

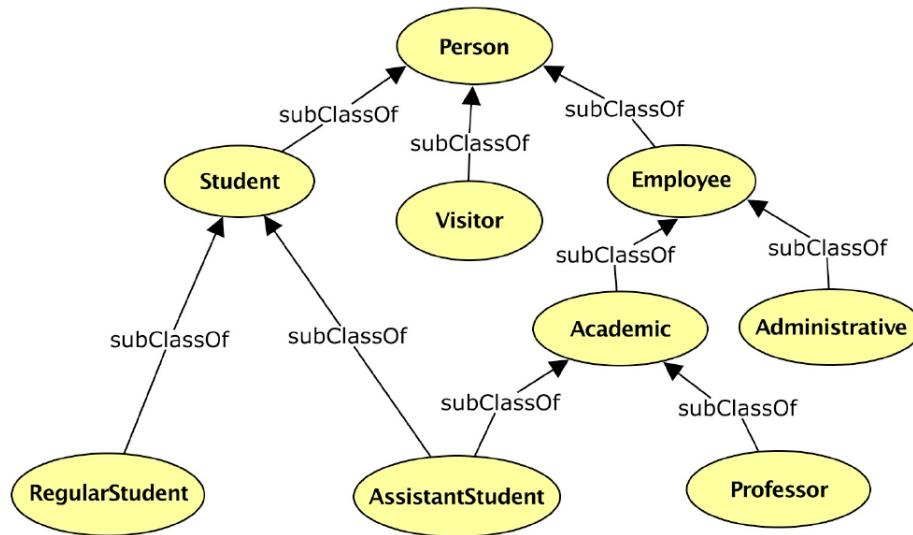


Figura 4.8: Jerarquía de clases del modelo persona

Una de las cualidades más importantes dentro de las ontologías es la reutilización porque se pueden construir ontologías a partir de otras existentes, refinándolas o ampliando sus límites para adaptarla a un dominio o aplicación, es por eso que para este modelo se reutilizó la ontología de persona propuesta en el trabajo de Cruz [28], en el que se consideran las relaciones de datos y propiedades de objetos que se muestran en la Tabla 4.4 relacionadas con la clase de dominio correspondiente y el tipo de dato utilizado.

Tabla 4.4: Propiedades de datos y objetos del modelo persona

| Dato y Propiedad de objeto | Dominio | Rango o Tipo de dato |
|----------------------------|----------|----------------------|
| hasCategory | Academic | String |
| hasEconomicNumber | Employee | String |
| hasEmail | Person | String |
| hasGender | Person | String |
| hasName | Person | String |
| hasPosition | Academic | String |
| hasStudentID | Student | String |
| hasVisitPurpose | Visitor | String |
| hasAcademicTitle | Academic | String |
| hasDepartment | Employee | String |
| providesAdvice | Academic | Student |

4.2.5. Ontología de evento

El modelo de evento Figura 4.9 es el núcleo de este proyecto en él se representan las actividades que se efectúan dentro de un ambiente académico. De todo el universo de eventos que ocurre dentro de este tipo dominio, se diseñó una jerarquización de los eventos en el modelo, los cuales se enlistan a continuación:

- Las asesorías (*AcademicAdvising*) son actividades dónde un profesor imparte una tutoría a uno o dos estudiantes, es una oportunidad para intercambiar información, diseñada para ayudar al alumno a alcanzar sus metas educativas.
- Cursos académicos (*AcademicCourse*), estos son impartidos por uno o dos académicos que pueden ser profesores o ayudantes los cursos están centrados únicamente en los alumnos de licenciatura o posgrado, adicionalmente, se hizo una abstracción de los cursos de actualización para profesores, que se refiere a un grupo de académicos que reciben enseñanza o actualización dentro de la misma aula o espacio físico.
- Los Eventos de difusión (*DifussionEvent*) se utilizan para representar sucesos académicos con la finalidad de difundir la cultura, la investigación y la enseñanza, se dividen en: coloquios, congresos, demostraciones, panel de discusión, presentaciones, seminarios y talleres. En la Tabla 4.5 se describe con más detalle los conceptos de los

eventos descritos en este capítulo. Un evento de difusión puede ser parte de otro evento de este tipo, esto es posible debido a que un panel de discusión, una demostración o una presentación puede ocurrir dentro de un congreso, seminario o taller. Este tipo de propiedades fueron modeladas mediante el uso de *eventIsPartOf* la cual es una relación semántica entre tales clases de eventos de difusión.

- d) Por último, los eventos ambientales (*EnvironmentEvent*) se refieren a los cambios relacionados con el clima: temperatura, humedad, luminosidad. Una propiedad de datos para este tipo de evento es *hasVariation* que indica el valor de la variación de un cambio climático, por ejemplo, la temperatura ha aumentado en 2.5 grados Celsius. Otra variable importante a modelar es la presencia, por ejemplo, detectar si alguien está entrando o saliendo de un espacio físico académico.

Para la implementación del módulo de evento fueron necesarias 16 clases, 11 subclases, 2 propiedades de dato, una propiedad de objeto, las instancias de esta ontología van a ser insertadas automáticamente por el sistema de detección de eventos descrito en las siguientes secciones y están divididas en dos tipos: académicas y ambientales.

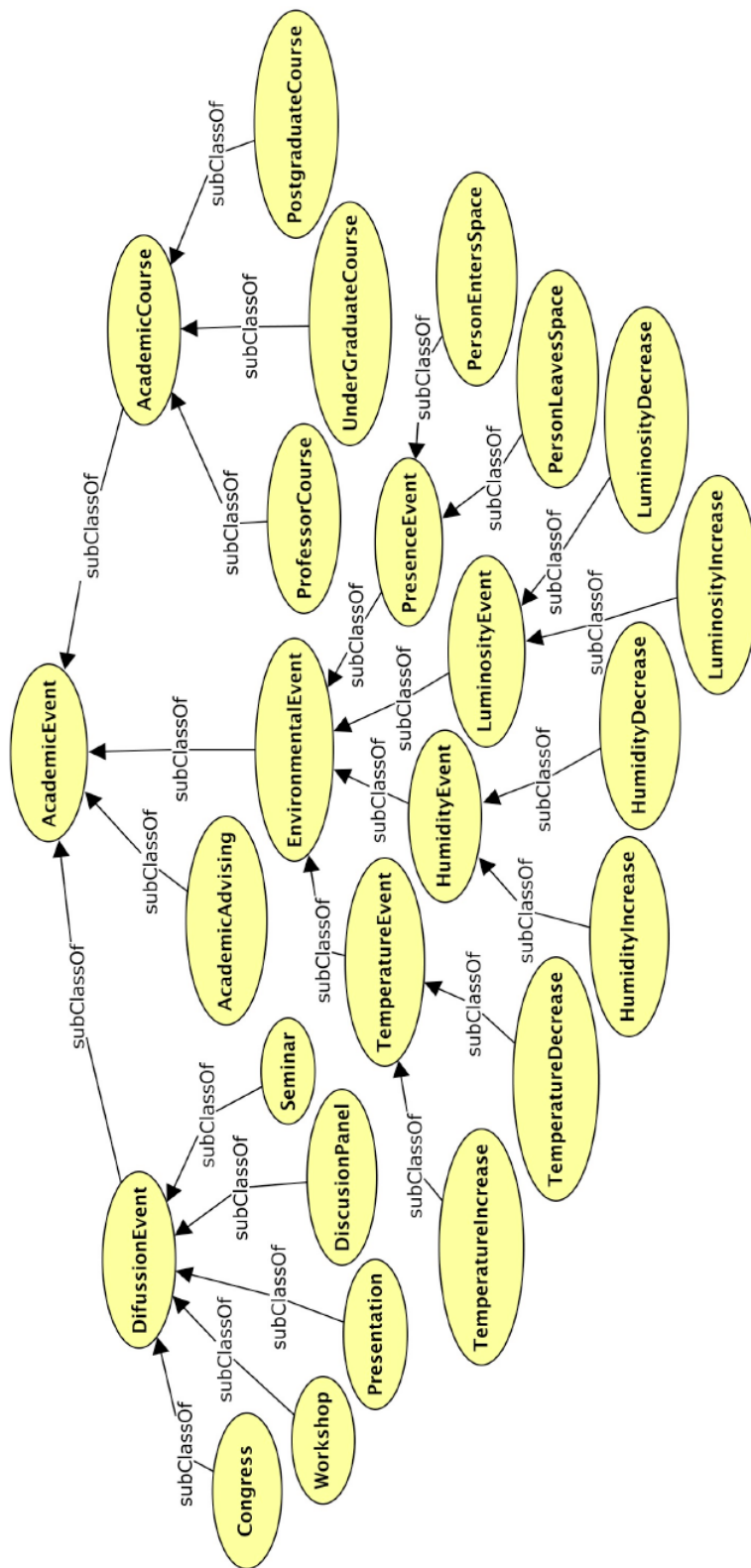


Figura 4.9: Ontología de evento

Tabla 4.5: Clases y conceptos de la ontología evento

| Clase | Descripción |
|---------------------|---|
| Event | Los eventos académicos son el conjunto de actividades de carácter profesional que implica algún tipo de enseñanza o capacitación. |
| AcademicAdvising | La asesoría académica es una oportunidad para intercambiar información entre estudiantes y académicos. |
| DiffusionEvent | Son eventos destinados para la divulgación entre los cuales se encuentran congresos, coloquios, foros, etc. |
| Colloquium | Reunión que convoca a un número limitado de personas para debatir una problemática, sin que necesariamente haya de recaer un acuerdo. |
| Congress | Evento donde se hace un confrontación entre un grupo de personas de la misma especialidad. Realizan un intercambio de experiencias, opiniones, toma de decisiones y resolver problemas. |
| Demo | Las demostraciones brindan a los investigadores y profesionales una oportunidad emocionante e interactiva para presentar sus sistemas, artefactos y/o prototipos de investigación ya sea en una sesión regular o en la exhibición técnica. |
| DiscussionPanel | Evento donde un grupo de personas discute un tema, hecho o problema, conducido por un coordinador. Constituye la parte final de simposios o mesas redondas. |
| Presentation | Una presentación es una forma de ofrecer y mostrar información de datos y resultados de una investigación. |
| Seminar | Reunión especializada que tiene tecnología técnica y académica cuyo objetivo es realizar un estudio profundo de ciertas materias con un tratamiento que requiere una interacción entre los especialistas. En este evento académico se expone lo más reciente de temas generales con conferencistas seleccionados previamente. |
| Workshop | El propósito de los talleres es proporcionar una plataforma interactiva y enfocada para presentar o discutir ideas nuevas. Pueden incluir presentaciones orales, de posters, conferencias magistrales y paneles. Los talleres se pueden programar para 1 día o 2 días. |
| AcademicCourse | Grupo de alumnos o profesores que reciben enseñanza o actualizaciones dentro de una misma aula o espacio físico académico. |
| DiplomaCourse | Es un curso de mediana o larga duración, generalmente ofrecido por una universidad, tiene el propósito de enseñar o actualizar algún conocimiento o habilidad específica. Se diferencia de una especialización, porque no conducen a la obtención de ningún título ni grado académico. |
| PostgraduateCourse | Se llaman cursos de posgrado a los cursos posteriores al título de grado y comprenden cursos de maestría (magister) y doctorado. |
| ProfessorsCourse | Cursos de actualización para profesores que pertenecen a la institución. |
| UndergraduateCourse | Los cursos de licenciatura son utilizados para obtener un título luego de realizar estudios con una duración de 12 trimestres para formar profesionales componentes. |

4.2.6. Sistema ontológico: integración de los módulos

Anteriormente se mencionó que el módulo de eventos es el núcleo del sistema de ontologías desarrollado, pero hay relaciones entre los módulos con un propósito específico. El objetivo principal es integrar cada uno de los módulos y relaciones entre ellos para encontrar una representación total de un ambiente académico (Figura 4.10), que pueda permitir la representación y razonamiento de eventos.

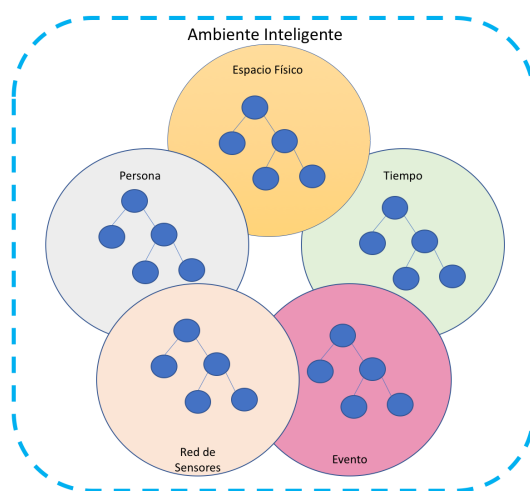


Figura 4.10: Representación de un ambiente inteligente

La información del sistema de ontologías final incluye algunas propiedades de objeto entre módulos, que se indican en la Tabla 4.6 donde se muestra el nombre de la propiedad, clase del dominio y clase del rango.

Tabla 4.6: Propiedades de objeto entre módulos

| Propiedad de Objeto | Dominio | Rango |
|---------------------|---------------|----------------|
| happensIn | AcademicEvent | PhysicalSpace |
| hasPersonInvolved | AcademicEvent | Person |
| hasTemporalEntity | AcademicEvent | TemporalEntity |
| isDetectedBy | AcademicEvent | Node |
| nodeIsLocatedIn | Node | PhysicalSpace |
| hasDeviceAssigned | Person | Device |
| isAssignedTo | Person | PhysicalSpace |
| hasPersonDetected | PhysicalSpace | Person |

Una breve descripción de su propósito se expone a continuación:

Cada propiedad de objeto en el sistema ontológico tiene una intención importante, *happensIn* fue modelado para representar el lugar donde el evento académico ocurrió y es útil para contestar preguntas como ¿Dónde ocurrió el evento X?; El contexto de la persona está relacionado con un evento académico usando *hasPersonInvolved* relación para responder ¿Quién está involucrado en X evento?, las personas involucradas en eventos académicos son empleados, estudiantes o visitantes; el contexto de tiempo es relacionado con los eventos a través de la relación semántica llamada *hasTemporalEntity* adecuado para responder ¿Cuándo X evento ocurrió?

La relación semántica *isDetectedBy* es un complemento de *happensIn*, su finalidad es la detección automática del lugar donde ocurrió el evento, proceso que se llevó a cabo con una regla de inferencia la cual se describirá a detalle más adelante, las relaciones *nodeIsLocatedIn*, *hasDeviceAssigned* e *isAssignedTo* sirven para obtener información estática del ambiente inteligente, es decir esta información nunca tendrá variaciones dentro del contexto, *hasPersonDetected* es una relación dinámica porque su información puede ir variando con el tiempo, porque depende del espacio físico actual en donde se encuentra la persona en un momento determinado.

El modelo ontológico fue implementado usando el lenguaje de ontologías web 2.0 (OWL por sus siglas en inglés). En total el modelo ontológico desarrollado llamado “*intelligentEnvironment*” consiste de 6487 axiomas, 91 clases, 34 propiedades de objeto, 50 propiedades de dato, 719 individuos, 13 clases equivalentes y 9 clases disjuntas para la caracterización de un ambiente académico inteligente y se puede observar en la Figura 4.11.

4. METODOLOGÍA DE SOLUCIÓN

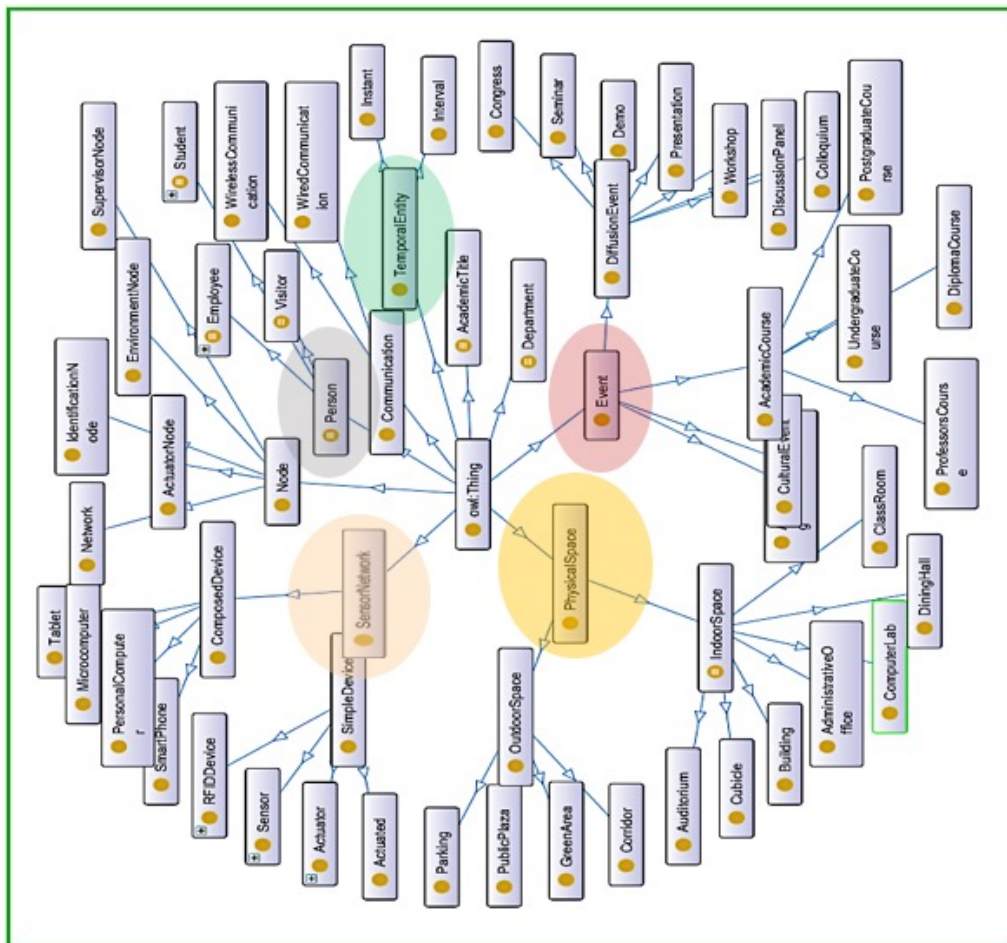


Figura 4.11: Sistema de Ontologías implementado en Protégé (OntroGraf)

4.3. Detección de eventos

En esta sección se presenta el proceso de identificación de eventos dentro de un ambiente académico, como se detectan semiautomáticamente y su representación en el sistema ontológico *intelligentEnvironment*. Para después poder inferir y razonar con ellos. Los eventos están agrupados en dos tipos; académicos y ambientales.

4.3.1. Detección de eventos ambientales

En el dominio académico, la temperatura, humedad y luminosidad son de gran importancia para ofrecer un servicio adecuado al usuario en función del contexto.

La secretaria del trabajo y previsión social en la NORMA Oficial Mexicana NOM-025-STPS-2008, “Condiciones de iluminación en los centros de trabajo”, establece los requerimientos mínimos de iluminación en las áreas de trabajo y centros de estudio, con el fin de contar con la luminosidad requerida para cada actividad visual, con el fin de proveer un ambiente saludable en relación de las actividades a desarrollar. En la Tabla 4.7 se muestran los niveles de iluminación mínimos en un recinto académico.

Tabla 4.7: Niveles de Iluminación en un ambiente académico.

| Tarea visual | Tipo de recinto | Niveles mínimos de iluminación (luxes) |
|--|---|--|
| Atención administrativa | Oficinas administrativas | 300 |
| Actividades relacionadas con la docencia | Salas de clases, educación superior | 300 |
| En interiores: distinguir el área de tránsito, desplazarse caminando, vigilancia, movimiento de vehículos | Interiores generales: almacenes de poco movimiento, pasillos, escaleras, estacionamientos | 50 |
| Distinción clara de detalles: captura y procesamiento de información, manejo de instrumentos y equipo de laboratorio | Talleres de precisión: salas de cómputo, áreas de dibujo, laboratorios | 500 |
| Actividades relacionadas con la investigación | Salas de profesores, cubículos | 400 |
| Lectura e investigación | Bibliotecas | 400 |
| Elaboración de alimentos | Cocinas | 300 |

De acuerdo con la norma Oficial Mexicana NOM-015-STPS-2001, “Condiciones térmicas elevadas la temperatura”, en un ambiente académico para poder desarrollar actividades en condiciones favorables, se requiere un **mínimo de 15°C y máximo de 30°C**.

Según las normas y especificaciones para estudios, proyectos, construcción e instalaciones la humedad relativa debería de estar entre el $50\% \pm 10\%$.

Con base en las normas anteriores, se implementó un módulo de detección de eventos ambientales sustentado en reglas de decisión y se describe a detalle en las siguientes secciones de este capítulo, se mostrará todo el proceso de detección de eventos, desde la generación y obtención de los datos hasta la consumación del módulo.

La finalidad del detector es tener una representación formal de dichos eventos en el modelo para después ajustar automáticamente las variables ambientales con la ayuda de una red de sensores, con el objetivo de proporcionarles las condiciones climáticas ideales a los usuarios de la comunidad académica para obtener un mejor desempeño.

4.3.1.1. Obtención y generación de datos

La obtención de los datos proviene de sensores estratégicamente colocados en espacios físicos dentro del ambiente académico, las mediciones de la temperatura, humedad e iluminación están empaquetados en una estructura de datos que se puede observar en la Figura 4.12 y tiene la siguiente organización:

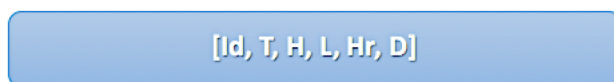


Figura 4.12: Paquete generado del nodo sensor

Donde:

- **Id** es el identificador del nodo sensor.
- **T** corresponde a la lectura de la temperatura.
- **H** corresponde a la lectura de la humedad.
- **L** corresponde a la lectura de la luminosidad.
- **Hr** se refiere a la hora de la toma de las mediciones y tiene el formato (hh:mm:ss).
- **D** se refiere a la fecha de la toma de las mediciones y tiene el formato (DD/MM/AAA).

Por ejemplo, en la siguiente medición el nodo sensor NS01 (Figura 4.13), tiene los valores de la temperatura en $23\text{ }^{\circ}\text{C}$, humedad del 20% y una luminosidad de 395 lúmenes, las lecturas fueron tomadas en el instante del día 28 del mes de junio de 2018 a las 17 horas con 8 minutos y 28 segundos.

[NS01,23.0,0.2,395,17:08:59,28/06/2018]

Figura 4.13: Ejemplo de una lectura del nodo sensor

Para la réplica de los datos se utilizaron agentes inteligentes creados con JADE que es un middleware que facilita el desarrollo de sistemas multi-agente bajo el estándar FIPA. Los agentes fueron creados con el propósito de simular un ambiente donde ocurran cambios frecuentes en la temperatura, humedad, luminosidad, la entrada y salida de personas. Con el objetivo de ver el comportamiento del método desarrollado bajo pruebas de estrés para después adaptarlo a un ambiente académico real.

Los datos simulados se generan con números aleatorios acotados por límites inferiores y superiores para no caer en incongruencias de datos en las variables ambientales. Además, los datos son generados con intervalos de tiempo que van desde 1 hasta 60 segundos. En la figura 4.14 se encuentra una captura del proceso de simulación de los agentes inteligentes desarrollados.

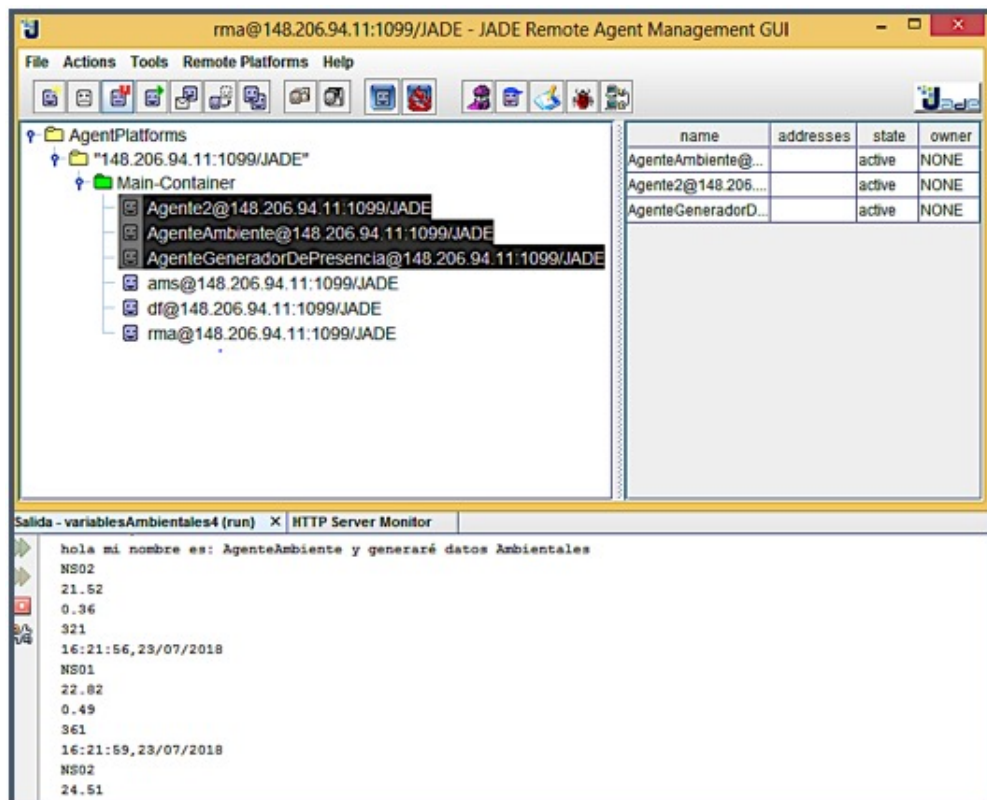


Figura 4.14: Agentes inteligentes ambientales

4. METODOLOGÍA DE SOLUCIÓN

Para los datos de presencia, es decir, conocer si alguien ha ingresado o abandonado un espacio físico. Se genera un paquete similar al anterior (Figura 4.15), este proviene de un *nodo identificador* y tiene la siguiente estructura:

[IdN, E, RFIDTag, Hr, D]

Figura 4.15: Paquete generado del nodo identificador

Donde:

- **IdN** es la referencia única del nodo identificador.
- **E** corresponde al estado del nodo identificador si se encuentra en 0 no hay presencia, si se encuentra en 1 hay detección de presencia..
- **RFIDTag** identificación por radiofrecuencia permite transmitir la identidad de un objeto y es único.
- **Hr** corresponde la hora de la toma de las mediciones y tiene el formato (hh:mm:ss).
- **D** corresponde la fecha de la toma de las mediciones y tiene el formato (DD/M-M/AAA).

En el siguiente ejemplo (Figura 4.16), la lectura proviene del nodo identificador NI03, el estado 1 indica que hay una entrada de un individuo portando un *RFIDTag* y su valor es el 800000002 mostrado en el campo número 3. Y el paquete ha sido enviado el 28 de Julio de 2018 a las 14 horas con 19 minutos y 45 segundos.

[NI03,1,800000002,14:19:45,28/06/2018]

Figura 4.16: Ejemplo de una lectura del nodo identificador

De manera análoga que, con los datos de las variables ambientales, se decidió realizar una generación de datos aleatorios para simular la entrada y salida de personas dentro de un espacio físico utilizando agentes inteligentes, con el fin de someter a pruebas de estrés a los métodos de detección de presencia.

El diagrama de clases del módulo de generación de datos utilizando agentes inteligentes desarrollado en java, se puede analizar en la Figura 4.17. Está compuesto de cuatro clases. Dos heredan de la clase *Agent* encontrada en la biblioteca *core* de *Jade* y dos clases más. La primera llamada "*DatosAmbientales*" donde por medio de números

aleatorios delimitados por umbrales se generan valores para la simulación de la temperatura, humedad, luminosidad y presencia, por último, la clase *GuardarPaquetes* es el productor de la estructura de datos descrita anteriormente.

La salida de este módulo consiste en un repositorio con los datos generados de la simulación, estos serán leídos por otra aplicación desarrollada en *Java* que lleva por nombre Detector de *Eventos Ambientales*.

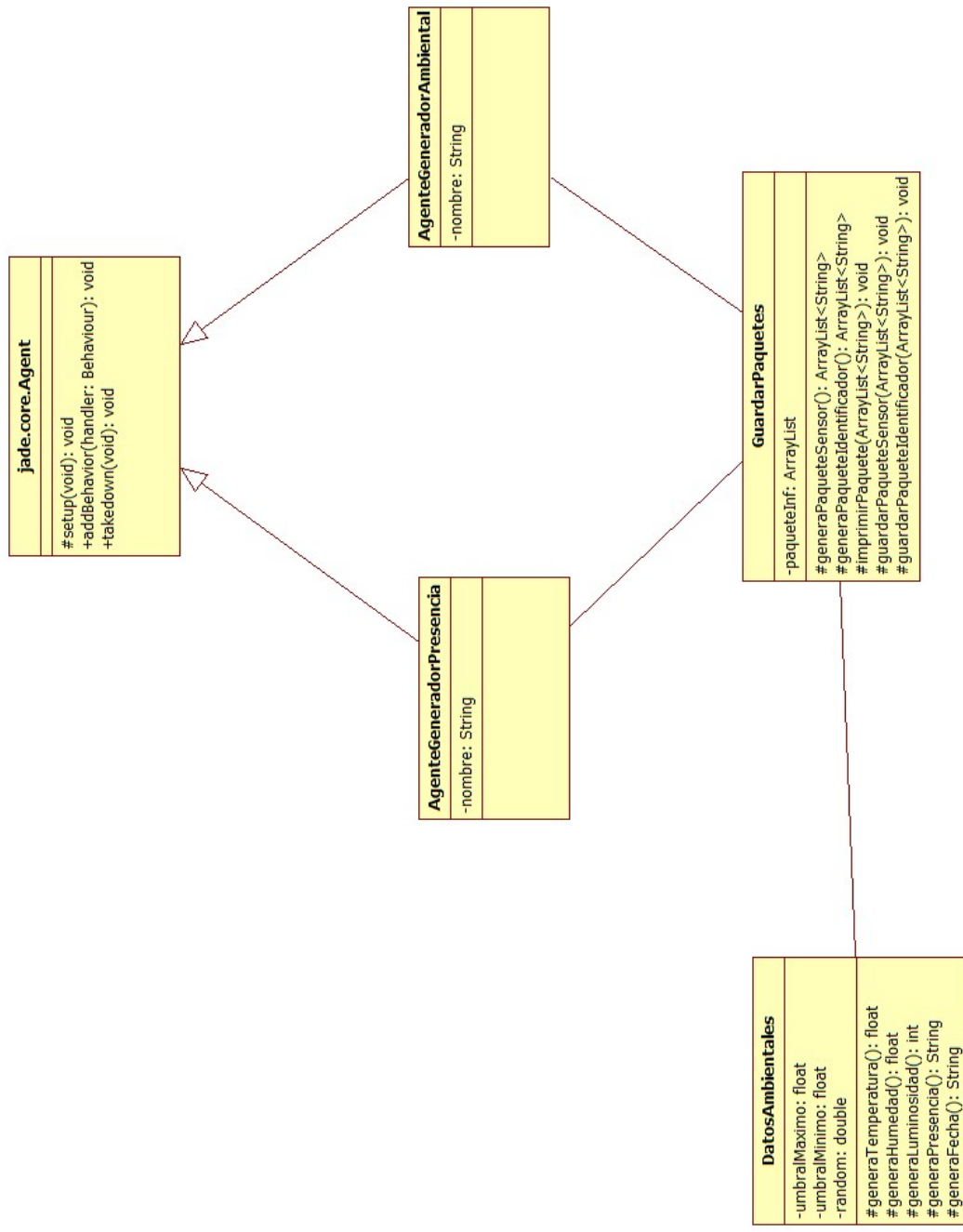


Figura 4.17: Diagrama de clases del módulo de generación de datos

4.3.1.2. Módulo de detección de eventos ambientales

El siguiente módulo tiene como objetivo detectar automáticamente los acontecimientos de tipo ambiental, esto es, mostrar si ocurre un cambio significativo en alguna de las variables ambientales, temperatura, humedad, luminosidad y presencia de individuos. Esta parte del sistema fue desarrollado con base en las once reglas presentadas a continuación:

R1: si existe una variación mayor o igual a dos grados centígrados entre la lectura de la temperatura anterior y la actual, hay un evento de incremento en la temperatura del lugar.

R2: Si existe una variación negativa mayor o igual a dos grados centígrados entre la medición de la temperatura actual y la anterior, hay un evento de decremento en la temperatura.

R3: si ocurre una variación en la humedad mayor o igual al 10 %, hay un evento de incremento en la humedad.

R4: Si ocurre una variación negativa en la humedad mayor o igual al 10 %, hay un evento de decremento en la humedad.

R5: si se detecta un aumento en mayor o igual a 50 luxes en la iluminación de un espacio físico, existe un evento de incremento de la luminosidad.

R6: si se detecta una pérdida en la iluminación mayor o igual a 50 luxes dentro de un espacio físico, hay un evento de decremento en la luminosidad.

R7: la temperatura del espacio físico debe encontrarse entre 15 y 30 grados centígrados, si no es así alertar a la red de sensores para regularla.

R8: la humedad dentro del espacio físico debe estar entre el 40 y 60 %, en caso contrario alertar a la red de sensores para que la regule.

R9: la luminosidad dentro de un espacio académico debe ser entre 300 y 500 luxes, si esta fuera del rango alertar a la red de sensores para que la regule.

R10: si el paquete procedente del nodo identificador contiene un *RFIDTag* (número de identificación único) conocido y es la primera vez que ingresa a la bitácora, detecta un evento entrada e identifica a la persona que arribó al lugar.

R11: si el paquete proveniente del nodo identificador, contiene un *RFIDTag* registrado y la bitácora marca que no es un ingreso, detecta un evento salida y almacénalo en el modelo ontológico.

Con base en las reglas presentadas anteriormente, se implementó el módulo de detección ambiental. Sirve para tener una representación automática de las variaciones climáticas que ocurren dentro de un espacio académico para con esta información crear un espacio sensible al contexto. En la Figura 4.18 se muestra el diagrama de clases y a continuación se proporciona una descripción de cada parte del módulo. El sistema consta de tres paquetes distribuidos en una estructura jerárquica, con la finalidad de agrupar el sistema para cerciorar la correcta funcionalidad y seguridad de acceso.

4. METODOLOGÍA DE SOLUCIÓN

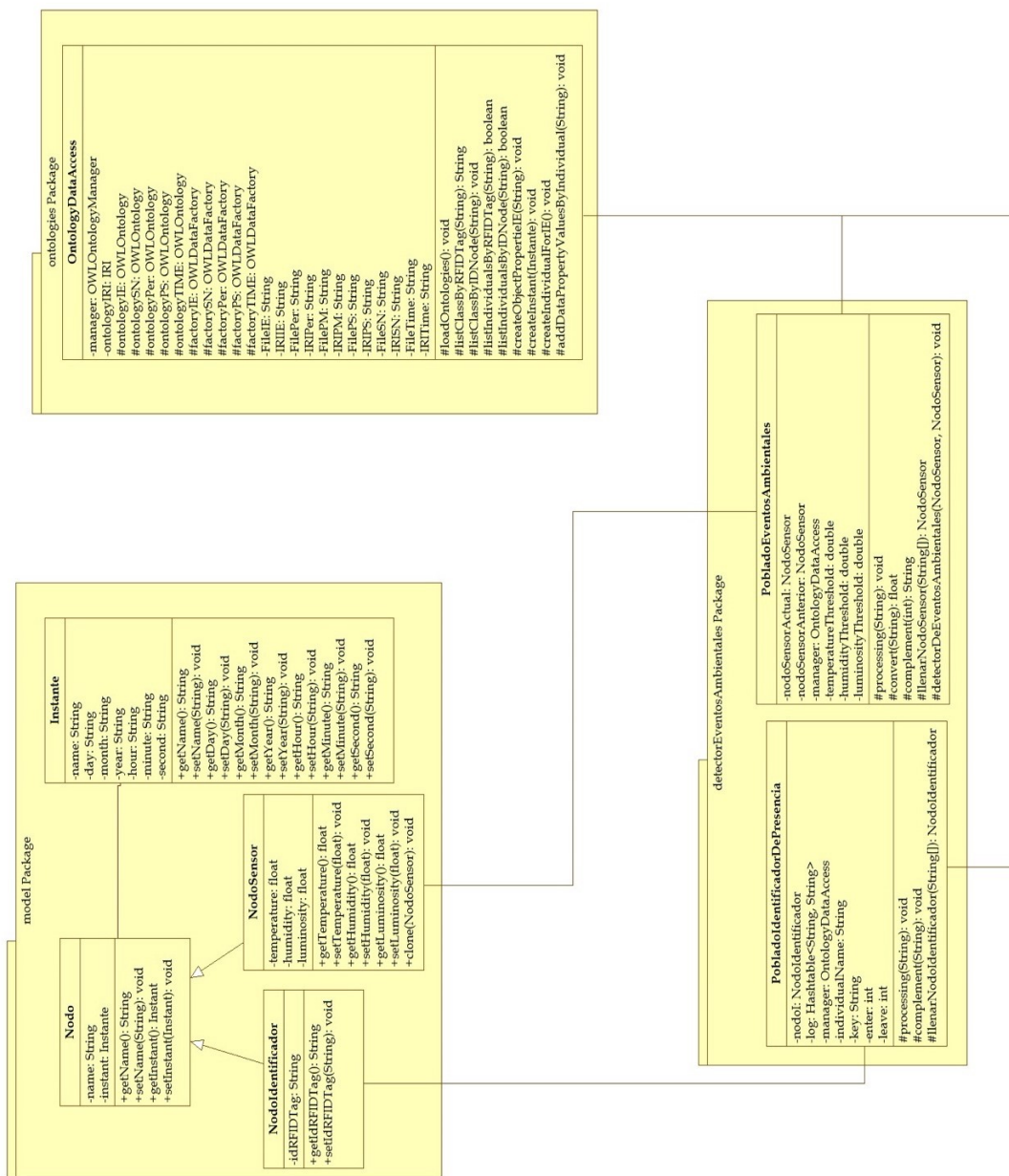


Figura 4.18: Diagrama de clases del módulo de detección de eventos ambientales

El primer paquete tiene por nombre *modelo* y sirve para mapear un objeto con una entidad del mundo real, en este caso se tendrá una correspondencia entre el modelo y los objetos tangibles encontrados en una red de sensores, específicamente en la sección de los nodos.

La clase primaria *Nodo* tiene como atributos el nombre o identificador del nodo, además, de un objeto de la clase *Instante*, sirve para conocer el momento exacto en que un nodo recauda cierta información en un ambiente académico, la clase *Instante* posee los atributos que componen un período de tiempo, estos son, horas, minutos, segundos, día, mes y año. Conjuntamente un identificador único para cada instancia de la clase, estas instancias se vincularán con los individuos de la clase *Instant* del modelo ontológico *TemporalEntity* descrita en la sección 4.1.3.

Las subclases o clases creadas a partir de la clase existente *Nodo* llevan por nombre *NodoSensor* y *NodoIdentificador*. La primera es una abstracción de un mote (nodo en un sensor de red capaz de realizar algún tipo de procesamiento, reúne información sensible proveniente de uno o más sensores). En este caso la información recaudada es la temperatura, humedad y luminosidad existentes en un espacio académico.

Finalmente, *NodoIdentificador* es la última clase del paquete *model*, con ella se obtiene una Identificación por Radiofrecuencia de individuos que ingresan y abandonan un espacio físico por medio de una etiqueta especial que emite ondas de radio. Precisamente esta etiqueta es el único atributo adicional de nuestra subclase y se obtuvo del procedimiento resultante de la lectura de un chip con un dispositivo *RFID Reader*, para luego capturar su información y transmitir la identidad de un objeto.

El segundo paquete se denomina *ontologies* en este se encuentra la lógica que va a interactuar con el modelo de representación de conocimiento *IntelligentEnviroment*. La única clase de este paquete es *OntologyDataAcces* y se comunica con el conjunto de funciones y procedimientos *OWL API*. Que permite trabajar en la creación y gestión de ontologías.

Con los atributos de la clase *OntologyDataAcces* se indica cada identificador de las ontologías modulares (IRI) que componen el sistema ontológico, además de la ruta para referenciar los directorios que contienen el modelo.

Los métodos implementados para esta clase tienen distintas funcionalidades, las cuales se describen a continuación:

- *loadOntologies*: tiene como propósito cargar en memoria el sistema ontológico, para posteriormente hacer procesamiento con éste.
- *listClassByRFIDTag*: es un procedimiento auxiliar en la identificación de individuos dentro del ambiente académico. Por medio de un *RFIDTag* que porta cada

4. METODOLOGÍA DE SOLUCIÓN

usuario, realiza una búsqueda dentro de la ontología de persona para conocer a qué categoría pertenece el individuo detectado (conocer si es un profesor, un ayudante o un alumno).

- *listIndividualsByRFIDTag*: similar al método anterior, pero con la diferencia que realiza una búsqueda en la ontología modular “persona”. Devuelve al individuo al que le pertenece el RFIDTag proporcionado, con este método se conoce el nombre de la persona identificada, así como el género, correo electrónico, clave de identificación (matricula o número económico), grado académico, departamento, preferencia de temperatura y los proyectos de investigación donde contribuye. Responde a la pregunta ¿Quién participa en el evento?
- *listClassByIDNode*: este método tiene la finalidad de saber a qué clase de la ontología “*SensorNetwork*” pertenece un nodo que ha sido identificado, puede ser un nodo concentrador, actuador, ambiental o de identificación.
- *listIndividualsByIDNode*: con la lectura generada por la red de sensores este método obtiene el lugar específico en el que se desarrolla un evento dentro del ambiente académico, por ejemplo, una persona ha ingresado al salón de clases *E313*. El procedimiento es el resultado de una consulta a la ontología *SensorNetwork* en el que se ingresa el identificador del nodo y se obtiene al individuo relacionado. La relación entre ontologías *nodeIsLocatedIn* nos indica en qué espacio físico el nodo está instalado. Responde a la pregunta ¿Dónde ocurre el evento?
- *createObjectPropertieIE*: sirve para crear relaciones entre las ontologías modulares: persona, espacio físico, evento y red de sensores.
- *createInstant*: crea individuos en la clase instantánea de la ontología *TemporalEntity*. Estos son adquiridos por el detector de eventos en el momento exacto en el que ocurre un cambio significativo.
- *createIndividualForIE*: este método sirve para poblar al sistema ontológico *IntelligentEnvironment* con individuos obtenidos del detector de eventos ambientales.
- *addDataPropertyValuesByIndividual*: tiene la finalidad de añadir atributos a las instancias del sistema ontológico, por ejemplo, una vez detectada una variación de temperatura este método ingresa en la propiedad de dato *hasVariation* un número flotante con el dato preciso del cambio de temperatura.

El tercer paquete “*detector de eventos ambientales*” está compuesto por dos clases, *PobladoIdentificadorDePresencia* y *PobladoEventosAmbientales*. La primera clase tiene como objetivo la identificación de las personas que interactúan en un espacio académico, con ello se consigue conocer los perfiles de los individuos y posteriormente por medio de reglas caracterizar eventos académicos.

Los atributos de la clase *PobladoIdentificadorDePresencia* son: una instancia de la clase *NodoIdentificador*, el objeto manager de la clase *OntologyDataAccess* y un registro

o bitácora que esta implementada con una estructura de datos y que realiza una función hash para identificar a las personas por su *RFIDTag*. Esta clase también contiene el nombre del individuo, además de dos índices para contabilizar las entradas y salidas en un espacio físico.

Uno de los métodos implementados de la clase es *processing*. Recibe como entrada el paquete que proviene de la red de sensores ya descrito, convirtiéndolo en un objeto de tipo *nodo identificador* con uso del método *llenarNodoIdentificador*. En el campo “estado” de este objeto se verifica si existe un evento de presencia dentro del espacio académico, en caso de ser afirmativo se extrae el identificador único *RFIDTag* para crear una llave con sus 4 últimos dígitos, el *key* se introduce a la bitácora para llevar un registro y así conocer si el individuo ingreso o abandonó el espacio físico, al mismo tiempo se crea una instancia de la clase *Instante* con la fecha y hora del acontecimiento.

Con la etiqueta *RFID* que pertenece al sujeto detectado se hace un proceso de identificación, que consiste en hacer una búsqueda de la persona asociada con dicho *RFID Tag* en el sistema ontológico “*ambiente inteligente*”. Una vez adquirido el perfil del individuo, se crea una instancia de evento de entrada o de salida según sea el caso, dicha instancia está compuesta por un nombre, descripción del evento, la persona involucrada, el espacio físico donde acaece y finalmente el instante en el que ocurrió.

La segunda clase del paquete tiene por nombre *EventosAmbientales* su propósito principal es la detección y representación de sucesos abruptos relacionados con el clima dentro de un ambiente académico, específicamente con la temperatura, humedad y la luminosidad.

Sus atributos son: dos objetos de la clase *NodoSensor* uno contiene la información actual de las variables ambientales y el segundo los datos de un momento anterior, el fin de estas variables es hacer una comparación entre ellas para identificar un evento de tipo ambiental.

En este proyecto definimos un cambio significativo como una variación de la temperatura entre 2 °C, la humedad entre el 10% y una variación de 50 luxes en la intensidad luminosa. De ser necesario, si en la literatura o por la opinión de expertos se decide establecer otros umbrales estos valores pueden ser alterados al modificar dentro del código las variables *temperatureThreshold*, *humidityThreshold* y *luminosityThreshold*.

Por último, la instancia manager de la clase *OntologyDataAccess* se utiliza para realizar la representación de los eventos en el modelo ontológico. A continuación, se describen los métodos implementados para la clase *PobladoEventosAmbientales*:

- *llenarNodoSensor*: se utiliza para convertir los datos adquiridos de la red de sensores a una instancia de la clase *NodoSensor*.

4. METODOLOGÍA DE SOLUCIÓN

- *complement*: identifica y complementa los componentes de los eventos para presentarlos dentro del sistema ontológico.
- *detectorDeEventosAmbientales*: con base en los umbrales ambientales establecidos, en este método se realiza el cálculo correspondiente para determinar si sucede un evento de tipo climático. Una vez que se ha detectado el evento se crean individuos de las clases; *HumidityEvent*, *TemperatureEvent* o *LuminosityEvent* para formar la representación de dichos acontecimientos en la ontología modular event, se establece el tipo de evento, también se determina en qué lugar ocurre gracias a la relación entre ontologías *happensIn*, también se detecta el instante en el que ocurrió representado por la relación *hasTemporalEntity* y el resultado de la variación de las condiciones climáticas es almacenado en la propiedad de dato *hasVariation*.

Al final de la detección de un evento ambiental, se compara el valor actual de la medición contra los datos ideales de las normas mexicanas descritos en el capítulo anterior. Si el valor actual está fuera del rango se le enviará al *nodo actuador* una alerta para que se encargue de regular el ambiente de manera automática.

- *processing*: este método se comunica con el procedimiento *detectorDeEventosAmbientales* y le envía las lecturas de las variables ambientales separándolas en dos objetos *nodoSensorActual* y *nodoSensorAnterior*. Donde como su nombre lo indica una instancia contiene los valores de las mediciones actuales y el otro las mediciones de un momento inmediato anterior.

4.3.2. Detección de eventos académicos

La detección de eventos académicos es un procedimiento que sirve para que de manera semiautomática inferir a partir de hechos suscitados lo que ocurre en un espacio académico, dicho de otro modo, conocer qué actividades realizan los actores en un lugar determinado. Esto se logra conjuntado distintas variables, hasta llegar a una conclusión, dichas variables son: personas (en función de su perfil). Puesto que cada individuo tiene un objetivo distinto que depende del lugar que visite. Por ejemplo, si un alumno se encuentra en el cubículo de un profesor hay una alta probabilidad que el evento sea una asesoría académica. Otra variable a considerar es el espacio físico donde se realiza el evento; por ejemplo, si es un auditorio probablemente la actividad que se efectúa sea un congreso.

Los eventos académicos detectados en esta parte del sistema son: coloquios, seminarios, congresos, panel de discusión, talleres, demostraciones, presentaciones, cursos de licenciatura, cursos de posgrado, cursos de actualización y las asesorías académicas.

Para lograr que el módulo pudiera detectar los eventos académicos es necesario dotarlo de cierta inteligencia y se obtuvo del conocimiento de expertos en el ámbito académico.

4.3.2.1. Caracterización de los eventos académicos

En esta sección se presenta el conocimiento adquirido de 3 expertos en el ámbito académico para caracterizar los eventos que ocurren en ese dominio y se enlista a continuación:

- Asesorías: son eventos donde participan uno o dos profesores y uno o dos alumnos, estos se desarrollan en cubículos, tienen una duración promedio de media hora. Los eventos de entrada para asesorías ocurren con una diferencia de 2 minutos.
- Eventos de difusión:
 - a) Congreso: en este evento participan alumnos, profesores y visitantes. El número de participantes oscila: mayor a 50 alumnos, participan entre 5 y 10 profesores y menos de 5 visitantes. Se llevan a cabo en auditorios su duración es de 1 día y por lo regular la entrada es en un lapso de 10 minutos.
 - b) Panel de discusión: en este evento participan alumnos y profesores. El número de participantes oscila entre 15 y 50 de los cuales es 3-4 profesores, el número alumnos es menor a 20. Se llevan a cabo en auditorios su duración varía entre 30 y 90 minutos, pueden ser discusiones en el marco de un congreso o seminario. El tiempo de entrada de los participantes es de 10 minutos.
 - c) Presentación: en este evento participan alumnos y profesores. El número de participantes oscila: entre 15 y 50 participantes de los cuales es 1 profesor, alumnos entre 15 y 49. Se llevan a cabo en auditorios, su duración varía entre 20 y 60 minutos, pueden ser presentaciones aisladas o en el marco de un congreso o seminario. El lapso de entrada de los participantes es de 10 minutos.
 - d) Seminario: en este evento participan alumnos, profesores y visitantes. El número de participantes oscila: entre 25 y 60 participantes de los cuales entre 3 y 5 son profesores, alumnos entre 20 y 50, menos de 5 visitantes. Se llevan a cabo en auditorios su duración es mayor a 3 horas y menor a 6 horas en un día. El lapso de entrada de los participantes es de 10 minutos.
 - e) Taller: participan uno o dos instructores que pueden ser académicos o visitantes; con un mínimo de 10 participantes y máximo 25 participantes (alumnos, profesores y visitantes). Se imparten en laboratorios con duraciones entre 8 y 15 horas. Este evento solo puede suceder en el marco de un congreso. El lapso de entrada de los participantes es de 10 minutos.
- Cursos académicos:
 - a) Cursos de posgrado: cursos donde participan uno, dos o tres académicos, el número de alumnos oscila entre 1 y 5 alumnos se imparten en aulas. Los horarios de estos eventos son de la siguiente manera: lunes, miércoles y

4. METODOLOGÍA DE SOLUCIÓN

viernes con horarios que inician a las 7 a.m. Con intervalos de hora y media. Martes y jueves inician 8:30 con intervalos de 2:15 o 1:30. El lapso de entrada de los participantes es de 15 minutos.

- b) Cursos de Actualización: curso donde la totalidad de los participantes son académicos. Se imparten en los laboratorios de cómputo. El número de participantes oscila entre 10 y 25, generalmente se imparten en el inter trimestre, la duración es de 20 horas dividido en 5 días de la semana. La hora de inicio generalmente oscila entre 9 y 10 a.m. y por la tarde de 2 o 3:00 p.m. El lapso de entrada de los participantes es de 15 minutos.
- c) Cursos de Licenciatura: cursos donde participan uno o dos académicos, el número de alumnos oscila entre 5 y 100 alumnos se imparten en: aulas y auditorios. Los horarios de estos eventos son de la siguiente manera: lunes, miércoles y viernes con horarios que inician a las 7 a.m. con intervalos de hora y media. Martes y jueves inician 8:30 y con intervalos de 2:15 o 1:30 horas.

Con el conocimiento adquirido se han determinado los atributos característicos de cada evento académico de modo que claramente se contrasta un tipo de acontecimiento de otro. Se utilizan reglas para la caracterización de eventos y siguen la estructura del tipo IF-THEN de esta manera el resultado depende de una serie de condiciones a cumplirse. Con estas reglas se modeló un árbol de decisión que sirve para esquematizar de mejor manera los eventos que se detectan dentro del sistema mencionado.

4.3.2.2. Árbol de decisión para la detección de eventos

Como ya se ha mencionado, las descripciones de los acontecimientos son el fundamento del árbol de decisión y su propósito es predecir, representar y categorizar una serie de condiciones que ocurren de manera continua relacionadas con el tiempo, persona y espacio con el fin de clasificar eventos académicos.

Este árbol Figura 4.19 está compuesto por seis niveles y las hojas representan el tipo de evento académico detectado. En el primer nivel está el lapso promedio que se da de tolerancia para que un conjunto de individuos acceda a un espacio físico en el que se realiza una actividad y según esta el periodo puede ser 2 minutos, 10 minutos o 15 minutos.

En el segundo nivel se considera el tipo de espacio donde ocurre un evento, se acota a cubículos, auditorios, salones de clase y laboratorios de cómputo. Entre el tercero y quinto nivel se evalúa el tipo y número de personas involucradas en evento académico, pueden ser profesores, visitantes o estudiantes, esta última categoría también engloba a los ayudantes de profesor.

Los eventos que detecta el árbol de decisión son: asesorías, presentaciones, panel de discusión, seminarios, congresos, cursos de actualización para profesores, talleres,

cursos de licenciatura y cursos de posgrado.

4.3.2.3. Módulo de detección de eventos académicos

El árbol de decisión fue traducido a código en el lenguaje de programación Java y su diagrama de clases se aprecia en la Figura 4.20 y se compone de los segmentos que se describen a continuación:

- Paquete “*modelo*”. En este se enlistan dos clases; la primera lleva por nombre *IndividuoDetectado* sirve para crear instancias de personas que han sido reconocidas por el detector ambiental, específicamente por el módulo de identificación. Los atributos que componen a la clase son: *IndividuoPersona* que contiene el identificador único de cada individuo perteneciente a la comunidad académica (matricula o número de empleado).
- El atributo *tipoPersona* permite establecer el rol del sujeto detectado, por ejemplo, determinar si es un trabajador, un estudiante o un visitante. Por otro lado, los atributos *individuoEspacio* y *tipoEspacio* delimitan el lugar en donde ocurrió la detección de la persona, es decir, donde ingresó o salió el individuo.
- La segunda clase correspondiente al primer paquete es una abstracción de un espacio físico es en donde se encuentra instalado el nodo identificador. Sus atributos contienen la ubicación del nodo y las instancias de los individuos que se detectaron dentro de este espacio, los métodos más importantes son *addInstancias*, que sirve añadir individuos detectados a la colección, *totalIndividuos*, devuelve la cantidad exacta de personas involucradas dentro del espacio físico.
- Paquete *detectorEventosAcademicos*: este paquete contiene las clases más importantes involucradas en la detección de eventos académicos. En la primera “*EventosAcademicos*” está implementado el árbol de decisión descrito en la sección anterior, sus atributos son un objeto de la clase *PobladoEventosAcademicos* y una serie de contadores que sirven para llevar un control de los eventos detectados.
- Los métodos *detectorLapso2min*, *detectorLapso10min* y *detectorLapso15min*, son el núcleo del detector y aquí se encuentran programadas las reglas de decisión obtenidas de los seis niveles del árbol y sirven para la clasificación de los eventos.

4. METODOLOGÍA DE SOLUCIÓN

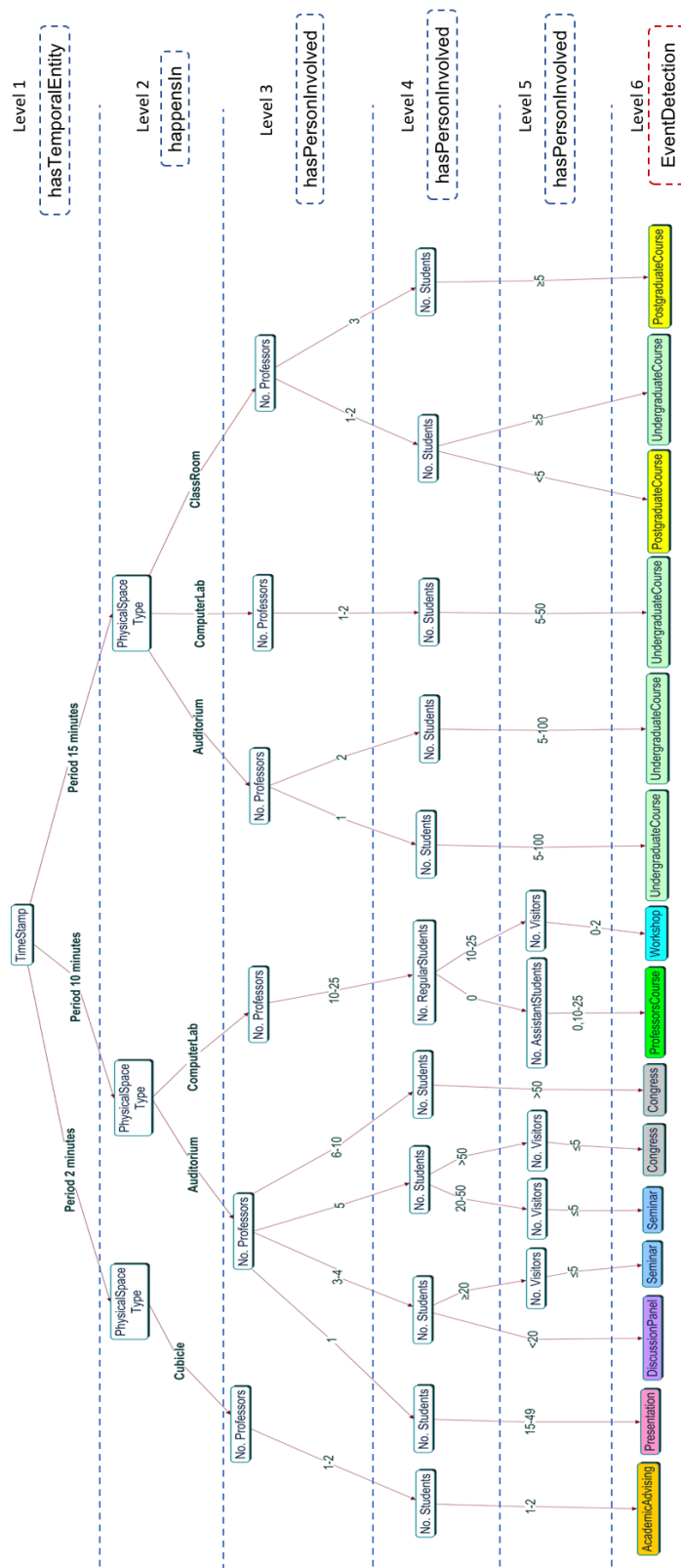


Figura 4.19: Árbol de decisión

- Los métodos *nodosCubiculos*, *nodosAuditorios*, *nodosSalones* y *nodosLaboratorios*, se utilizan para separar a los individuos en el lugar específico en el que se localizan, conjuntamente con el método *quitarRepetidos* que como su nombre lo indica quita de la colección a los objetos de individuos detectados duplicados. Una vez que se obtienen los personajes separados por grupos se hace el conteo de cuantos hay según su rol, el conteo se logra con los métodos, *numeroAlumnos*, *numeroProfesores*, *numeroAyudantes* y *numeroVisitantes*.
- En la clase *PobladoEventosAcademicos*, su único método genera las instancias de eventos académicos con sus componentes (¿Dónde?, ¿Cuándo? y ¿Quién participa en el evento?) y posteriormente hace un poblado en el modelo ontológico “*ambiente inteligente*” insertándolas.
- Por último, en la clase *Servidor* se crea un pool de tres hilos y que a su vez cada uno implementa a la interfaz *runnable* de la clase de *Java* y su función es crear a los hilos mencionados, con ellos se ejecutan periódicamente 3 subprocesos de detección de eventos, el primero se ejecuta cada 2 minutos, el segundo cada 10 minutos y el tercero cada 15, este procedimiento se logra a través del objeto ejecutor y el método *scheduleAtFixedRate* que pertenecen a la clase *ScheduledExecutorService* del framework *Executors* de *Java*.
- En los hilos se encuentra la inicialización del sistema y las subrutinas que cada uno lleva a cabo. En ellas se encuentran en los métodos: *detector2min*, *detector10min* y *detector15min* que realizan su procesamiento con base en una colección de individuos detectados. Dichas colecciones se encuentran almacenadas en los atributos, *individuosLapso2min*, *individuosLapso10min* e *individuosLapso15min*, en estas estructuras de datos se invoca a la clase *EventosAcademicos* para que realice el procedimiento de la rama del árbol de decisiones correspondiente.



Figura 4.20: Diagrama de clases del módulo de detección de eventos académicos

4.3.3. Unificación del sistema

A lo largo de este capítulo se describió individualmente cada módulo del sistema desarrollado. En la Figura 4.21 se puede ver el diagrama que modela la interacción del sistema entre el detector de eventos del tipo ambiental (temperatura, humedad y luminosidad) y el modelo ontológico.

El nodo sensor (*agentes ambientales*) y el servidor se comunican por medio de *sockets*, en el primer mensaje el *cliente* (nodo sensor) le envía una petición de conexión al servidor, una vez que el *servidor* acepta la conexión el nodo sensor le envía un paquete que corresponde a las lecturas de las variables ambientales, la fecha de obtención y el identificador del nodo sensor.

Una vez adquirido el paquete por parte del servidor lo reenvía a los servicios web para procesarlo en busca de un cambio brusco en el clima. Al mismo tiempo el detector ambiental le hace una consulta a la ontología con el identificador del nodo sensor para conocer el espacio específico ocurre la variación ambiental, una vez que el detector ambiental obtiene la respuesta por parte de la ontología, crea un individuo de evento y la manda a insertar en la clase correspondiente del modelo ontológico.

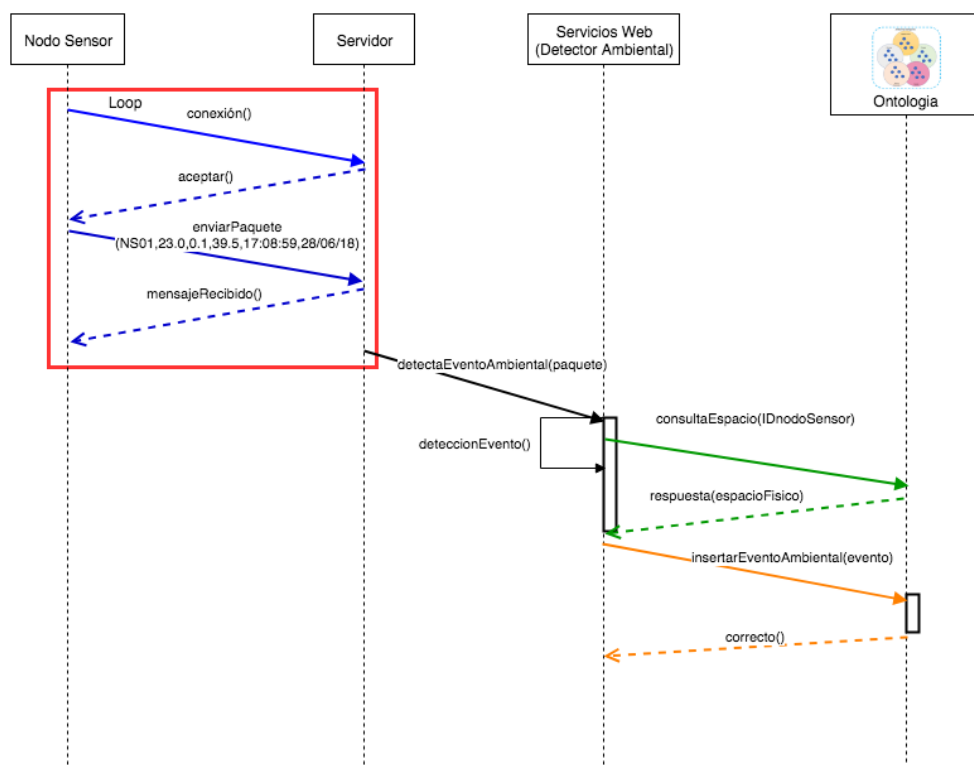


Figura 4.21: Diagrama de comunicación del detector ambiental

4. METODOLOGÍA DE SOLUCIÓN

En la Figura 4.22 se muestra el diagrama de interacción del sistema de detección de eventos académicos. La comunicación comienza por medio de sockets entre el nodo identificador y el servidor, el primero le envía un paquete del tipo presencia detectada compuesto con el RFIDtag del individuo, el identificador de nodo, además de la hora y la fecha de la detección. El servidor le reenvía el paquete al módulo de identificación para que determine ¿Qué persona ingreso al lugar? (perfil y su identificador), ¿Cuál es el espacio físico al que ingresó? Y finalmente el momento exacto en el que ingresó.

Las respuestas las obtiene al realizar una consulta al sistema de ontologías, una vez obtiene la respuesta con los datos necesarios, el módulo identificador crea una instancia de tipo “evento de presencia” y un objeto del tipo “individuo detectado”. La primera la manda a insertar a la ontología y el objeto se lo envía al servidor para que recolecte a todos los individuos detectados y con ellos el detector académico con base en reglas haga una clasificación y determine el tipo de evento para representarlo en el modelo ontológico.

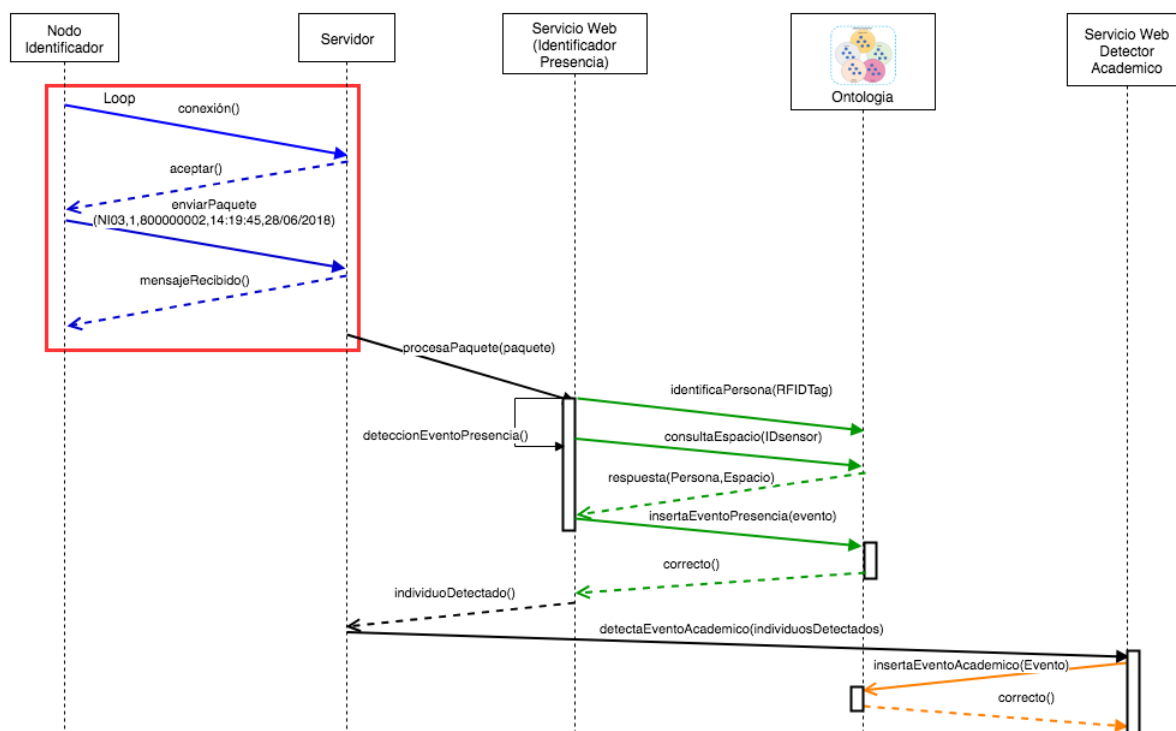


Figura 4.22: Diagrama de comunicación del detector académico

Evaluación

En este capítulo, se muestra el proceso de evaluación del proyecto desarrollado, se encuentra dividido en dos secciones la primera está enfocada a la valoración del modelo ontológico y la segunda en la evaluación de la detección de eventos.

5.1. Evaluación de la ontología

La evaluación del modelo ontológico está centrada en la extensibilidad y adaptación de cinco escenarios de casos de uso para eventos en entornos académicos, el razonamiento e inferencia serán medidos mediante el uso de preguntas de competencia relacionadas con eventos identificados. La consistencia y coherencia también son evaluadas en el modelo ontológico de contexto académico.

5.1.1. Escenarios de casos de uso

Se presentan cinco escenarios de casos para eventos de entorno académico y ambiental, para evaluar la adaptación del modelo ontológico para el dominio académico.

- a) La profesora Maricela Claudia Bravo Contreras ha ingresado al cubículo H289 el día 15 de enero del 2019 a las 10:30:15 a.m. Este escenario se representa en el modelo ontológico de ambiente inteligente como un evento de entrada, el resultado de dicha representación se muestra en la Figura 5.1 solamente con las clases involucradas.
- b) El profesor Alejandro Reyes Ortiz ingresa al cubículo H254 el día 15 de enero del 2019. Casi dos minutos después ingresa el alumno Josué Padilla Cuevas al mismo espacio a las 13:55:48 p.m. Este escenario representa dos eventos de entrada al mismo cubículo, pero además también se detecta un evento académico del tipo asesoría y el resultado de dichos eventos se observan en la Figura 5.2 con las clases involucradas.

5. EVALUACIÓN

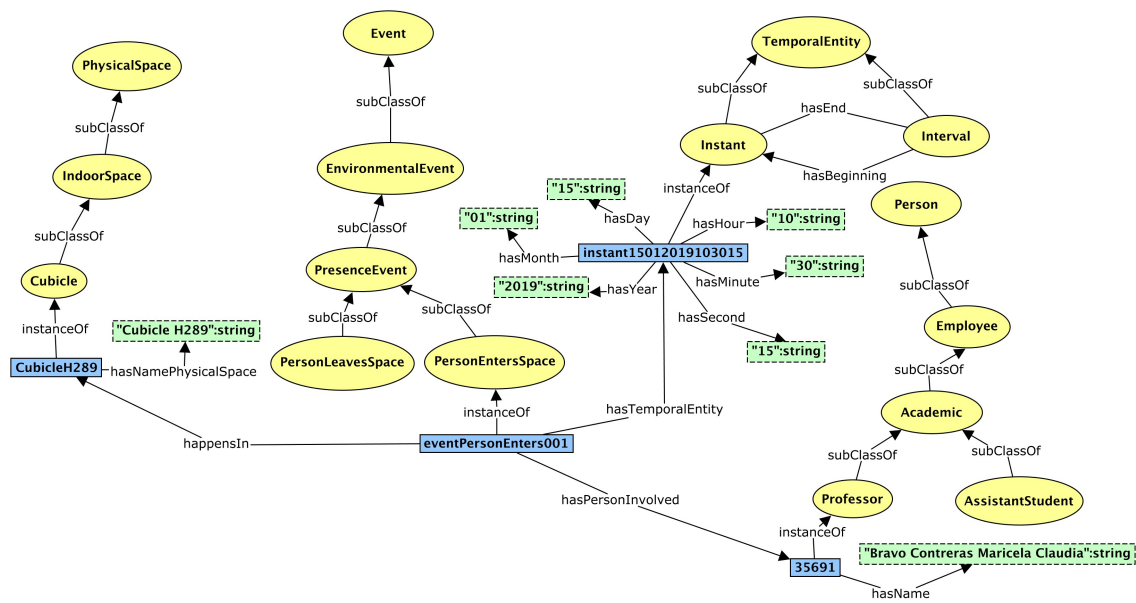


Figura 5.1: Escenario de casos de uso evento de entrada “eventEnters”

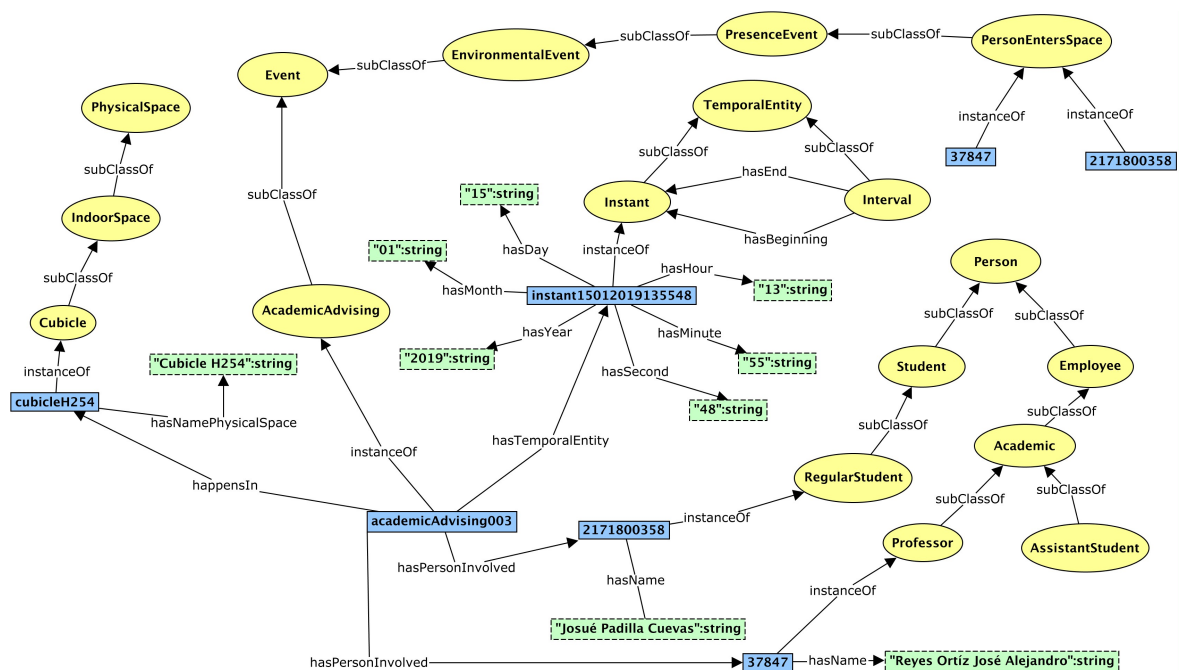


Figura 5.2: Escenario de casos de uso asesoría “academicAdvising”

- c) La profesora Beatriz Adriana González Beltrán llega al aula E313 para comenzar un curso de maestría de la UEA “modelado de sistemas” el 18 de enero de 2019 a

las 19:09:49 a la que asisten tres estudiantes: Jesús Manuel Jacinto Landeros, López Vazquez Aarón y María Fernanda Pineda . Dicho escenario se representa y se adapta al modelo ontológico como se muestra en la Figura 5.3.

- d) En el salón de clases E313 se ha registrado un incremento de temperatura súbito de 2.0 °C debido a las altas temperaturas de la zona o a la llegada masiva de individuos al lugar. El día 18 de enero de 2019 a las 19: 11:20. Figura 5.4.
- e) En la sala de cómputo *Babbage* ubicada en el edificio *T*, el día 18 de enero de 2019 a las 13:27:21 p.m. Se lleva a cabo un “taller intermedio de adobe indesign” donde participan 10 profesores y 10 alumnos, además lo imparte un visitante, en él se explica la manera de utilizar la aplicación para la composición digital de páginas desarrollada por la compañía Adobe *Systems*. Su representación se puede observar en la Figura 5.5.

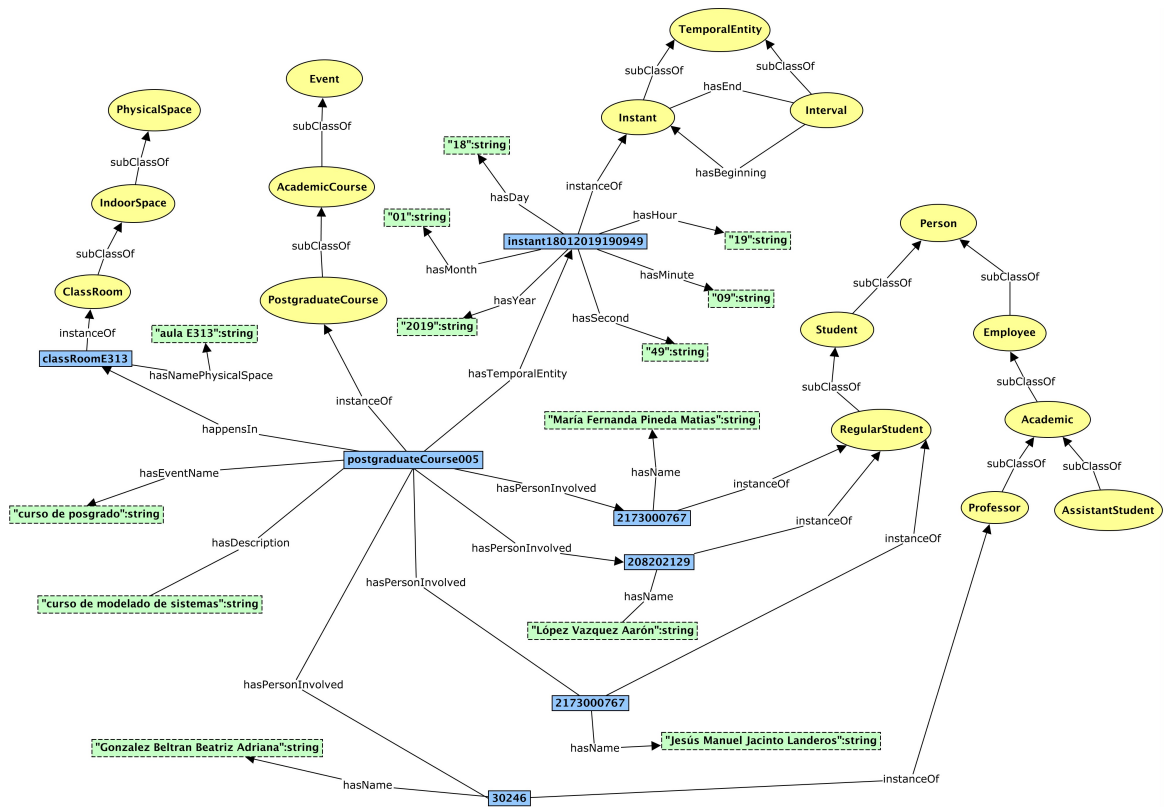


Figura 5.3: Escenario de casos de uso curso de posgrado “postgraduateCourse”

5. EVALUACIÓN

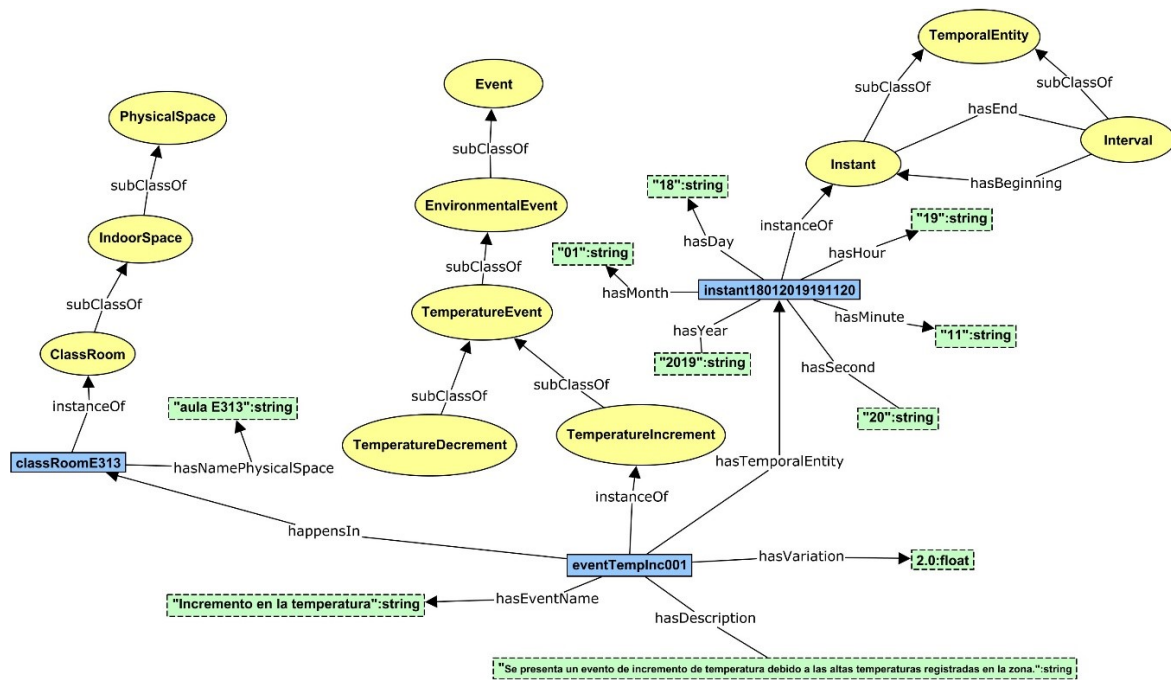


Figura 5.4: Escenario de casos de uso incremento de temperatura “eventTempInc”

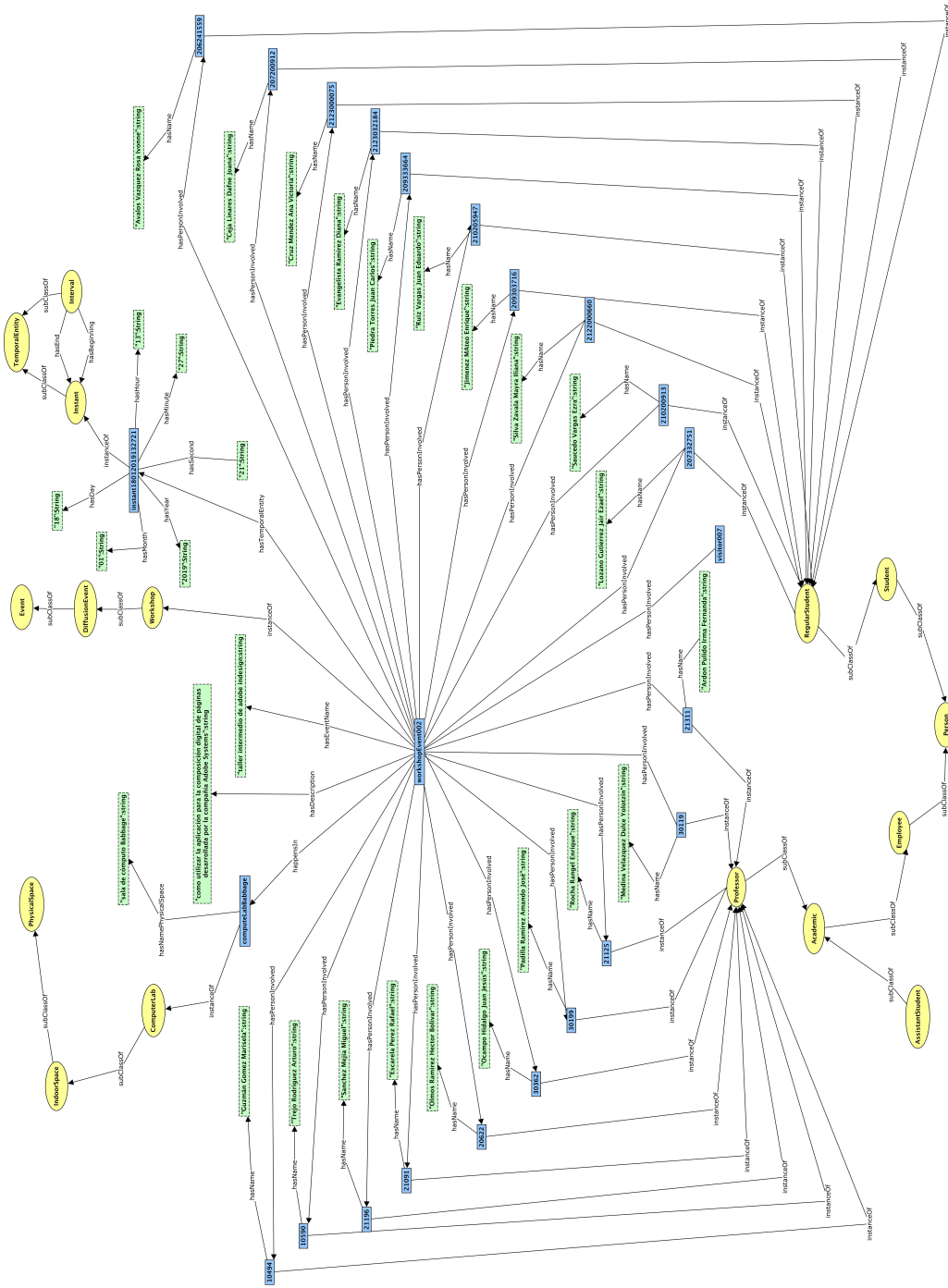


Figura 5.5: Escenario de casos de uso taller “workshop”

5.1.2. Razonamiento por preguntas de competencia

De acuerdo con Gómez [29], la evaluación de la ontología se refiere a la construcción correcta del contenido de la ontología, asegurando que sus definiciones implementadas puedan responder correctamente las preguntas de competencia. Por lo tanto, se evaluó el modelo ontológico utilizando las preguntas para llevar a cabo un razonamiento con la información del contexto académico sobre los diferentes eventos. Se ejecutaron las preguntas de competencia, primero se definieron en lenguaje natural y después se tradujeron a un lenguaje de reglas. Las siguientes preguntas se resolvieron satisfactoriamente:

- ¿Cuáles son los eventos ocurridos el día 19 de enero de 2019?
- ¿Cuántas asesorías se llevaron a cabo el 20 de enero de 2019?
- ¿Quiénes participaron en la primera asesoría detectada?
- ¿Cuál es la temperatura actual del espacio H-254?
- ¿Cuántos congresos se detectaron?
- ¿En dónde se llevaron a cabo los seminarios ocurridos?
- ¿Cuántas personas participaron en el último taller que se llevó a cabo?
- ¿En qué lugar ocurrió un cambio de luminosidad mayor a 60 luxes?
- ¿Qué eventos se realizaron en el salón de clases E313?
- ¿Cuántas personas participaron en el seminario de física?

En la Figura 5.6 se muestran los resultados del razonamiento para las consultas presentadas usando *Protégé*. Las consultas implementadas mostraron resultados correctos y prometedores. Este tipo de consulta podría usarse para desarrollar un sistema de respuesta a preguntas en lenguaje natural para un modelo basado en el contexto académico.

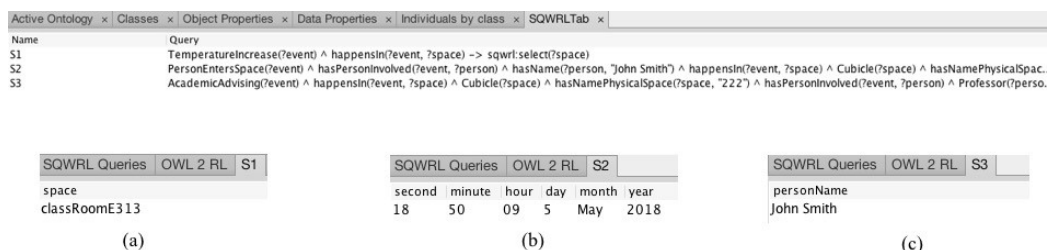


Figura 5.6: Ejemplo de preguntas de competencia

5.1.3. Consistencia, coherencia y eficiencia

El sistema ontológico es correcto si cada módulo que lo integra es correcto y consistente. Para esta evaluación se utiliza el motor de razonamiento Pellet disponible en *Protégé*. Primero, se valida a los módulos por separado revisando la consistencia entre axiomas, instancias y que las clases estén bien definidas. Luego, para evaluar la calidad del sistema de ontología, se consideraron los siguientes principios: a) claridad. Durante la ejecución de consultas y las tareas de razonamiento, en este sentido la ontología no produjo ninguna interpretación errónea; b) coherencia. Después de la ejecución de las tareas de razonamiento, todos los individuos fueron clasificados correctamente. La verificación de la coherencia de la ontología se ejecutó para verificar que ninguna de las definiciones de las clases o axiomas tenga contradicciones lógicas, así como los individuos instanciados en la ontología. Esta actividad final consiste en ejecutar las tareas de razonamiento en la clasificación de taxonomía, calcular los tipos inferidos y la verificación de consistencia; c) eficiencia. El tiempo de ejecución de las 30 preguntas de competencia muestra un promedio de 2523 ms, tal rendimiento está relacionado con el tamaño de los datos de entrada.

5.2. Evaluación del detector de eventos

Para la ejecución y evaluación del detector de eventos se utilizó un equipo de cómputo con las siguientes características: Intel Core I5-2310 a 2.90 GHZ, 6 MB *SmartCache*, 4 núcleos y 4 hilos. Almacenamiento de 512 GB y memoria RAM de 6 GB. *Java 8* con JDK 1.8, *NetBeans* IDE 8.2, *OWL API* y *JAVA Agent DEvelopment Framework* (JADE 4.5).

5.2.1. Agentes utilizados

Para la generación de los datos fue necesario crear dos tipos de agentes especializados los cuales siguen un conjunto de reglas, con el fin de generar paquetes que contengan la información precisa de una lectura del medio ambiente. Para el primer conjunto de agentes el paquete que generan debe contener las mediciones del censado la temperatura, humedad y luminosidad. c

En el caso de la temperatura los agentes generan mediciones aleatorias en un rango de entre 26° / 13 °C puesto que son las temperaturas promedio que se presentan en el mes de junio en la CDMX, fecha en la que se realizó la simulación. En el caso de la humedad se generaron datos aleatoriamente que van desde el 46 % hasta el 71 %. Finalmente, para la luminosidad se crearon variaciones de luxes con un rango de entre 200 y 600.

Para el segundo tipo de agentes los “*generadores de presencia*” se crearon datos que simulan el *RFIDTag* que se encuentra en las tarjetas de identificación de 130 usuarios

5. EVALUACIÓN

pertenecientes al modelo de contexto académico. Con estos datos los agentes realizan el comportamiento de entrada y salida en un espacio físico.

Los agentes encargados en realizar la simulación de los *nodos* ubicados en el espacio académico para la evaluación; Fueron cuatro especializados en generar datos de tipo ambiental (*nodo sensor*) y cinco especializados en generar datos de presencia (*nodo identificador*). Estos fueron programados para enviarle paquetes al servidor cada cuatro segundos en el caso del tipo identificador y cada tres minutos para los del tipo ambiental.

Dentro de la evaluación es importante mencionar que, de aproximadamente 4000 paquetes generados por día, todos fueron creados correctamente y su envío no tuvo retraso en el lapso de tiempo requerido. En la figura 5.7 se observa el proceso de simulación de los agentes inteligentes.

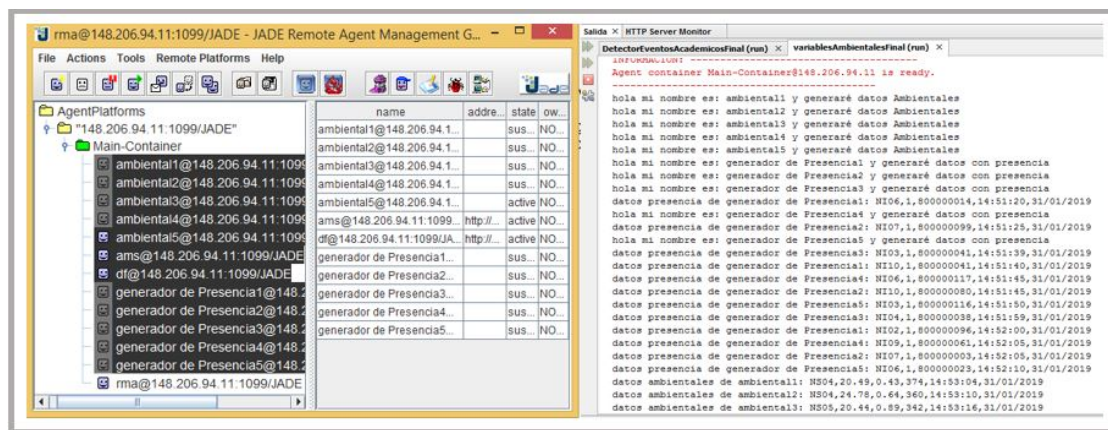


Figura 5.7: Simulación de paquetes provenientes de nodos

5.2.2. Servidor

Para la evaluación, el servidor recibió durante cinco días los paquetes provenientes de la simulación del ambiente académico. Enviándole a los servicios web los datos necesarios para la detección de eventos. Los envíos fueron programados por el servidor cada 2 minutos, 10 minutos y 15 minutos en el caso de los eventos académicos y en los eventos ambientales fueron reenviados inmediatamente al ser adquiridos.

El servidor le envió en promedio al detector de eventos académicos; para el lapso de dos minutos 10 a 15 individuos detectados. Para el lapso de 10 minutos se enviaron en promedio 20 a 75 individuos detectados y en cada iteración para el lapso de 15 minutos se enviaron 50 a 115 individuos.



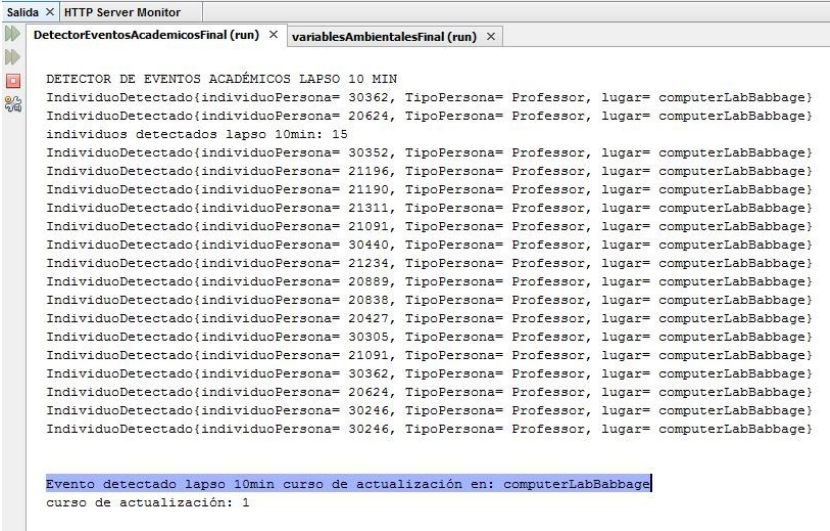
Figura 5.8: Detección de eventos ambientales

5.2.3. Servicios web

Dentro de los servicios web en este proyecto se encuentra el detector de eventos académicos y el detector de eventos ambientales, es importante mencionar que el sistema de detección de eventos está acotado a solo detectar 1000 instancias por clase de evento, con el fundamento de tener una adecuada eficacia en el sistema ontológico al momento de recuperar información. El objetivo principal de esta evaluación es validar el proceso de detección de eventos. En la Figura 5.8 se puede ver el proceso de la detección de eventos ambientales, específicamente se aprecia un decremento de temperatura de 2.76 °C, un incremento de la humedad del 14% y un decremento de 87 lúmenes en iluminación del espacio.

Para la detección de acontecimientos académicos se utilizó un árbol basado en reglas. En la Figura 5.9 se observa el primer ejemplo de una clasificación, para ser más puntuales de un curso de actualización (*ProfessorsCourse*), en este caso el detector recibió un total de 15 individuos en un lapso de 10 minutos (sin contar a los duplicados que pudieron salir e ingresar nuevamente al espacio físico). Todos los individuos recibidos por el servidor ingresaron al mismo sitio; el salón de cómputo Babbage, el rol de todos los individuos detectados es profesor, luego entonces según el clasificador y con la opinión de los expertos en el ámbito académico el evento detectado como curso de actualización es correcto.

5. EVALUACIÓN



```
Salida x HTTP Server Monitor
DetectorEventosAcademicosFinal (run) x variablesAmbientalesFinal (run) x
DETECTOR DE EVENTOS ACADÉMICOS LAPSO 10 MIN
IndividuoDetectado(individuoPersona= 30362, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 20624, TipoPersona= Professor, lugar= computerLabBabbage)
Individuos detectados lapso 10min: 15
IndividuoDetectado(individuoPersona= 30352, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 21196, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 21190, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 21311, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 21091, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 30440, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 21234, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 20889, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 20838, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 20427, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 30305, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 21091, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 30362, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 20624, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 30246, TipoPersona= Professor, lugar= computerLabBabbage)
IndividuoDetectado(individuoPersona= 30246, TipoPersona= Professor, lugar= computerLabBabbage)

Evento detectado lapso 10min curso de actualización en: computerLabBabbage
curso de actualización: 1
```

Figura 5.9: Evento detectado curso de actualización

En el segundo ejemplo mostrado en la Figura 5.10 el servidor le envió al detector de eventos académicos un total de 22 individuos que ingresaron a la sala de cómputo Byron ubicada en el edificio T en un lapso de diez minutos, como se puede observar del número total de individuos 11 son de tipo estudiante y 11 de tipo profesor, por lo que el detector de eventos con base en el conjunto de reglas IF-THEN llegó a la determinación que en ese espacio físico ocurre un evento de difusión del tipo taller (workshop) y al no haber presencia de individuos visitantes lo imparte un profesor de la institución.

En el tercer ejemplo se detecta un evento de difusión de tipo congreso en el auditorio W003. El total de individuos recibidos fue de 72 de los cuales cinco corresponden a profesores y 67 son estudiantes. Con el conocimiento de los expertos en el ámbito académico esta detección es correcta y se puede observar en la Figura 5.11.

```

Salida x HTTP Server Monitor
DetectorEventosAcademicosFinal (run) x variablesAmbientalesFinal (run) x
DETECTOR DE EVENTOS ACADÉMICOS LAPSO 10 MIN
individuos detectados lapso 10min: 22
IndividuoDetectado(individuoPersona= 2122000660, TipoPersona= RegularStudent, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 207332696, TipoPersona= RegularStudent, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 210200913, TipoPersona= RegularStudent, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 21125, TipoPersona= Professor, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 2122000017, TipoPersona= RegularStudent, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 208203963, TipoPersona= RegularStudent, lugar= computerLabByron)

IndividuoDetectado(individuoPersona= 10590, TipoPersona= Professor, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 209201011, TipoPersona= RegularStudent, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 10494, TipoPersona= Professor, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 207330482, TipoPersona= RegularStudent, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 30199, TipoPersona= Professor, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 207330482, TipoPersona= RegularStudent, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 30440, TipoPersona= Professor, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 209303716, TipoPersona= RegularStudent, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 20272, TipoPersona= Professor, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 10590, TipoPersona= Professor, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 20622, TipoPersona= Professor, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 2122000660, TipoPersona= RegularStudent, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 30199, TipoPersona= Professor, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 2123032184, TipoPersona= RegularStudent, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 21190, TipoPersona= Professor, lugar= computerLabByron)
IndividuoDetectado(individuoPersona= 21426, TipoPersona= Professor, lugar= computerLabByron)
evento detectado Taller en: computerLabByron
taller: 1
    
```

Figura 5.10: Detección de evento académico taller

```

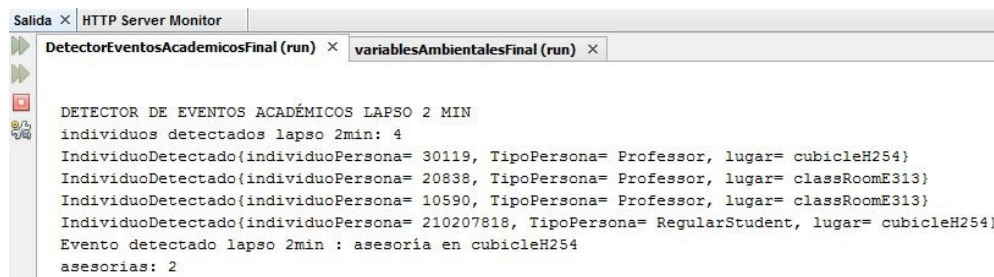
Salida x HTTP Server Monitor
DetectorEventosAcademicosFinal (run) x variablesAmbientalesFinal (run) x
DETECTOR DE EVENTOS ACADÉMICOS LAPSO 10 MIN
IndividuoDetectado(individuoPersona= 20930774, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 212300075, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 204201030, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 207332696, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 212002941, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 212004072, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 212200037, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 212300075, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 213003302, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 20936603, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 207332753, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 210301947, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 204203020, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 204201359, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 209202230, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 209201031, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 210207818, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 209203396, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 204207377, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 212200037, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 212200660, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 209302320, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 212200037, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 212300075, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 209201031, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 212303664, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 21120, TipoPersona= Professor, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 21045, TipoPersona= Professor, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 21311, TipoPersona= Professor, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 10587, TipoPersona= Professor, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 20838, TipoPersona= Professor, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 217307451, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 217303773, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 204203888, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 217303915, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 212200104, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 217307004, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 217303859, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 217304014, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 217300852, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 217300487, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 217304047, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 217300378, TipoPersona= RegularStudent, lugar= auditoriumM003)
IndividuoDetectado(individuoPersona= 217302154, TipoPersona= RegularStudent, lugar= auditoriumM003)
Evento detectado lapso 10min congreso en: auditoriumM003
congreso: 1
    
```

Figura 5.11: Detección de evento congreso

En la Figura 5.12 se ejemplifica la detección de un evento académico del tipo *AcademicAdvising (asesoría)*. Se recibieron 4 individuos por parte del servidor en un lapso de dos minutos, el detector los separa de acuerdo al lugar en el que ingresaron; entonces se tiene que dos profesores se encuentran en el salón de clases *E-313* y los otros dos individuos (un profesor y un alumno) se ubican en el cubículo *H254*, el detector con base en las reglas programadas infiere que se está realizando una asesoría en el cubículo *H254* con el alumno que tiene por matrícula *210207818* y el profesor con número

5. EVALUACIÓN

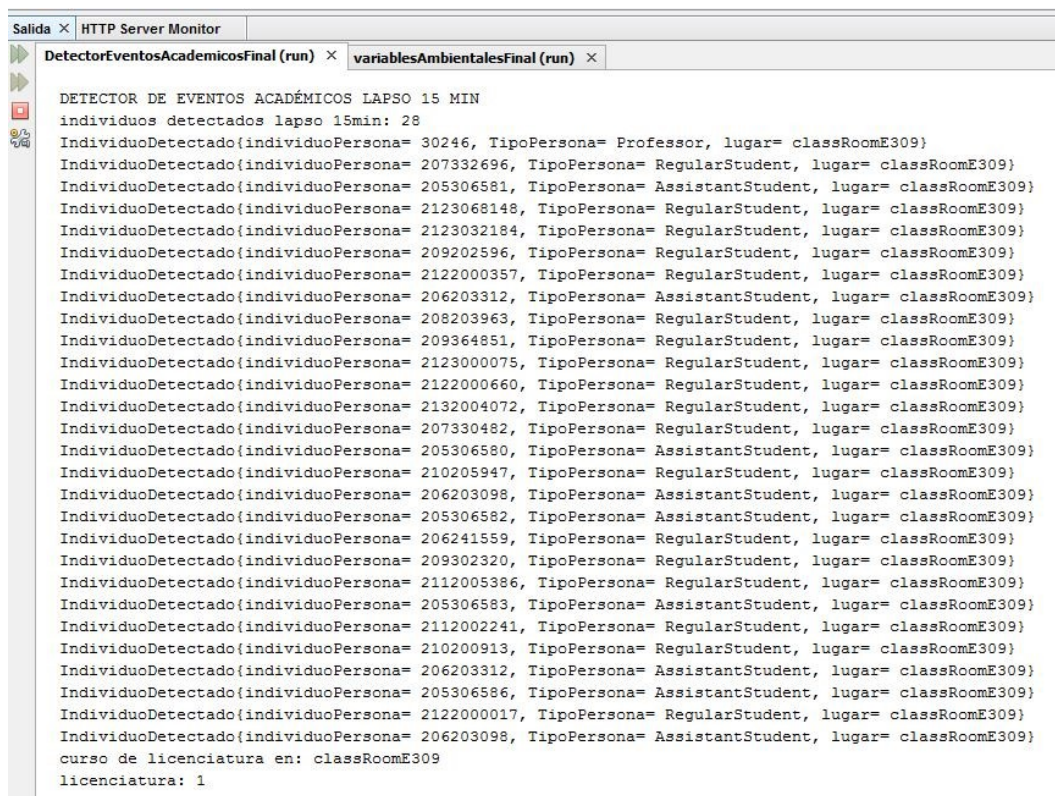
económico 30119.



```
Salida x HTTP Server Monitor
DetectorEventosAcademicosFinal (run) x variablesAmbientalesFinal (run) x
DETECTOR DE EVENTOS ACADÉMICOS LAPSO 2 MIN
individuos detectados lapso 2min: 4
IndividuoDetectado{individuoPersona= 30119, TipoPersona= Professor, lugar= cubicleH254}
IndividuoDetectado{individuoPersona= 20838, TipoPersona= Professor, lugar= classRoomE313}
IndividuoDetectado{individuoPersona= 10590, TipoPersona= Professor, lugar= classRoomE313}
IndividuoDetectado{individuoPersona= 210207818, TipoPersona= RegularStudent, lugar= cubicleH254}
Evento detectado lapso 2min : asesoria en cubicleH254
asesorias: 2
```

Figura 5.12: Evento detectado asesoría

En el último ejemplo el servidor le envía una colección de 28 individuos identificados en el salón de clases *E-309* durante un lapso de 15 minutos, de los cuales uno tiene el rol de profesor y 25 son del tipo estudiante (quitando a 2 duplicados que salieron e ingresar nuevamente al espacio físico). Con estos datos el sistema detecta un curso de licenciatura en el salón E-309, lo cual es correcto y se puede apreciar en la Figura 5.13.



```
Salida x HTTP Server Monitor
DetectorEventosAcademicosFinal (run) x variablesAmbientalesFinal (run) x
DETECTOR DE EVENTOS ACADÉMICOS LAPSO 15 MIN
individuos detectados lapso 15min: 28
IndividuoDetectado{individuoPersona= 30246, TipoPersona= Professor, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 207332696, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 205306581, TipoPersona= AssistantStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 2123068148, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 2123032184, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 209202596, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 2122000357, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 206203312, TipoPersona= AssistantStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 208203963, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 209364851, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 2123000075, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 2122000660, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 2132004072, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 207330482, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 205306580, TipoPersona= AssistantStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 210205947, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 206203098, TipoPersona= AssistantStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 205306582, TipoPersona= AssistantStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 206241559, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 209302320, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 2112005386, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 205306583, TipoPersona= AssistantStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 2112002241, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 210200913, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 206203312, TipoPersona= AssistantStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 205306586, TipoPersona= AssistantStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 2122000017, TipoPersona= RegularStudent, lugar= classRoomE309}
IndividuoDetectado{individuoPersona= 206203098, TipoPersona= AssistantStudent, lugar= classRoomE309}
curso de licenciatura en: classRoomE309
licenciatura: 1
```

Figura 5.13: Detección de evento curso de licenciatura

5.2.4. Resultados del detector: precisión, exhaustividad y medida-f

Los resultados consistieron en evaluar cinco corridas del sistema de detección de eventos una por día, en la Tabla 5.1 se encuentra representado el número de eventos académicos detectados por día, en la Tabla 5.2 el número de eventos ambientales detectados y en la Tabla 5.3 el número de eventos totales que se detectaron en los cinco días de prueba. Por ejemplo, en el día 1 se detectaron 11 asesorías, una presentación, 5 eventos de panel de discusión, un seminario, un congreso, un curso de actualización, un taller, 6 cursos de licenciatura, 7 cursos de posgrado. Además 999 eventos de entrada a un espacio, 999 de salida, 122 decrementos de humedad, 136 incrementos de la misma, de temperatura 50 decrementos y 57 incrementos, finalmente 44 decrementos de luminosidad y 136 incrementos. Esto da un total de 2557 eventos detectados en el primer día.

Para el segundo día se detectaron un total de 2224 eventos divididos de la siguiente manera 27 asesorías, una presentación, 3 paneles de discusión, un seminario, un congreso, dos cursos de actualización, un curso de licenciatura y cinco de posgrado. Los eventos ambientales detectados fueron: 1998 eventos de presencia repartidos de igual manera en entradas y salidas, 41 eventos de incremento de humedad, 48 decrementos, 28 incrementos de temperatura por 25 decrementos y, por último 24 incrementos de luminosidad y 19 decrementos.

Tabla 5.1: Número de eventos académicos detectados por día

| Día | Asesoría | Presentación | Panel de discusión | Seminario | Congreso | Curso de actualización | Taller | Curso de licenciatura | Curso de posgrado |
|-----|----------|--------------|--------------------|-----------|----------|------------------------|--------|-----------------------|-------------------|
| 1 | 11 | 1 | 5 | 1 | 1 | 1 | 1 | 6 | 7 |
| 2 | 27 | 1 | 3 | 1 | 1 | 2 | 0 | 1 | 5 |
| 3 | 9 | 2 | 1 | 1 | 1 | 2 | 0 | 2 | 4 |
| 4 | 5 | 2 | 2 | 2 | 0 | 2 | 3 | 2 | 3 |
| 5 | 11 | 0 | 1 | 1 | 0 | 3 | 1 | 8 | 25 |

Tabla 5.2: Número de eventos ambientales detectados por día

| Día | Entrada | Salida | Humedad | | Temperatura | | Luminosidad | |
|-----|---------|--------|------------|------------|-------------|------------|-------------|------------|
| | | | Decremento | Incremento | Decremento | Incremento | Decremento | Incremento |
| 1 | 999 | 999 | 122 | 136 | 50 | 57 | 44 | 136 |
| 2 | 999 | 999 | 41 | 48 | 28 | 25 | 19 | 24 |
| 3 | 469 | 409 | 8 | 13 | 1 | 4 | 1 | 5 |
| 4 | 512 | 401 | 58 | 55 | 29 | 24 | 15 | 23 |
| 5 | 999 | 999 | 151 | 149 | 61 | 68 | 50 | 39 |

Tabla 5.3: Número de eventos totales detectados por día

| Día | Eventos totales |
|-----|--------------------|
| 1 | 2577 |
| 2 | 2224 |
| 3 | 932 |
| 4 | 1138 |
| 5 | 2566 |

Del total de eventos presentados se utilizó un conjunto de prueba compuesto por 750 eventos, que se evaluaron y anotaron manualmente para generar un patrón de referencia como se muestra en el Apéndice 1. Esta evaluación compara los eventos detectados por el sistema, con los eventos analizados manualmente. Se consideran medidas de precisión, exhaustividad y Valor-F (denominada también F-score o medida-F), métricas bien conocidas para medir la precisión de las pruebas.

La precisión (P) es la proporción de eventos recuperados realmente relevante, del total de los eventos recuperados y el resultado de esta operación está entre 0 y 1. Es el coeficiente del número de eventos que se encuentran en el patrón de referencia entre el número total de eventos detectados, consulte la ecuación 5.1.

$$P = \frac{|\{eventosRelevantes\} \cap \{EventosDetectados\}|}{|\{EventosDetectados\}|} \quad (5.1)$$

Con base en los datos adquiridos del conjunto de prueba mostrados en la Tabla 5.4 se analizan los resultados de los eventos correctos e incorrectos y se dividen en dos categorías: académicos (asesorías, eventos de difusión y cursos académicos) y ambientales (temperatura, humedad, luminosidad y presencia). La evaluación se divide en tres datos fundamentales involucrados en todos los eventos: tiempo, persona y espacio físico.

Tabla 5.4: Datos obtenidos del patrón de referencia

| Tipo | Cantidad evaluada | Tiempo | | | Persona | | | Espacio | | |
|---------------------|----------------------|--------|----|----|---------|----|----|---------|----|----|
| | | si | no | na | si | no | na | si | no | na |
| advising | 63 | 62 | 1 | 0 | 63 | 0 | 0 | 61 | 2 | 0 |
| postgraduateCourse | 45 | 45 | 0 | 0 | 45 | 0 | 0 | 45 | 0 | 0 |
| professorsCourse | 10 | 9 | 1 | 0 | 10 | 0 | 0 | 10 | 0 | 0 |
| undergraduateCourse | 19 | 17 | 2 | 0 | 18 | 1 | 0 | 18 | 1 | 0 |
| congressEvent | 2 | 2 | 0 | 0 | 1 | 1 | 0 | 2 | 0 | 0 |
| discussionPanel | 12 | 10 | 2 | 0 | 11 | 1 | 0 | 11 | 1 | 0 |
| seminarEvent | 5 | 4 | 1 | 0 | 4 | 1 | 0 | 4 | 1 | 0 |
| presentationEvent | 5 | 5 | 0 | 0 | 5 | 0 | 0 | 5 | 0 | 0 |
| workshopEvent | 5 | 5 | 0 | 0 | 5 | 0 | 0 | 5 | 0 | 0 |
| eventHumDec | 64 | 64 | 1 | 0 | 0 | 0 | 64 | 64 | 0 | 0 |
| eventHumInc | 70 | 70 | 0 | 0 | 0 | 0 | 70 | 69 | 1 | 0 |
| eventLumDec | 59 | 58 | 1 | 0 | 0 | 0 | 59 | 59 | 0 | 0 |
| eventLumInc | 63 | 62 | 1 | 0 | 0 | 0 | 63 | 63 | 0 | 0 |
| eventTempDec | 65 | 65 | 0 | 0 | 0 | 0 | 65 | 64 | 1 | 0 |
| eventTempInc | 66 | 66 | 0 | 0 | 0 | 0 | 66 | 65 | 1 | 0 |
| eventEntrance | 94 | 94 | 0 | 0 | 93 | 1 | 0 | 94 | 0 | 0 |
| eventLeaves | 103 | 103 | 0 | 0 | 102 | 1 | 0 | 102 | 1 | 0 |

La Tabla 5.5 muestra los resultados obtenidos de la precisión en la detección de eventos en un entorno académico y se obtiene al aplicar la ecuación 1. La evaluación de enfoque ha indicado resultados prometedores logrando **96 %** de precisión en eventos académicos y **99 %** en eventos ambientales

Tabla 5.5: Resultados de precisión

| Event | Total | Tiempo | | Precisión | Persona | | Precisión | Espacio | | Precisión | Promedio |
|---------------------|-------|-----------|-------------|-----------|-----------|-------------|-----------|-----------|-------------|-----------|----------|
| | | Correctos | Incorrectos | | Correctos | Incorrectos | | Correctos | Incorrectos | | |
| Eventos Académicos | 166 | 159 | 7 | 0.957831 | 162 | 4 | 0.975903 | 160 | 6 | 0.963855 | 0.96 |
| Eventos Ambientales | 584 | 581 | 3 | 0.994863 | 582 | 2 | 0.996575 | 580 | 4 | 0.993150 | 0.99 |

Exhaustividad (E) también denominado “*recall*” expresa la proporción de eventos relevantes recuperados, comparado con el total de eventos que son relevantes existentes

5. EVALUACIÓN

en la ontología, con total independencia de que éstos, se recuperen o no, en este caso se expresa como la ecuación 5.2:

$$E = \frac{|\{eventosRelevantes\} \cap \{EventosDetectados\}|}{|eventosGoldStandard|} \quad (5.2)$$

Para obtener el valor de la exhaustividad se calculó un conjunto de datos de oro obtenidos del producto entre la cantidad de eventos evaluados y el grado de error de cada uno de los sensores (temperatura, humedad y luminosidad), esto es conocido como el conjunto *gold standard* o test de referencia, como se muestra en la Tabla 5.6. El grado de error fue extraído de las hojas de datos que proporcionan los fabricantes y se pueden observar en la Tabla 5.7.

Tabla 5.6: Datos con Gold Standard

| Tipo | Cantidad evaluada | Gold Standard |
|---------------------|-------------------|---------------|
| eventHumDec | 64 | 66 |
| eventHumInc | 70 | 72 |
| eventLumDec | 59 | 70 |
| eventLumInc | 63 | 75 |
| eventTempDec | 65 | 81 |
| eventTempInc | 66 | 82 |
| eventEntrance | 94 | 94 |
| eventLeaves | 103 | 103 |
| advising | 63 | 63 |
| postgraduateCourse | 45 | 45 |
| professorsCourse | 10 | 10 |
| undergraduateCourse | 19 | 19 |
| congressEvent | 2 | 2 |
| discussionPanel | 12 | 12 |
| seminarEvent | 5 | 5 |
| presentationEvent | 5 | 5 |
| workshopEvent | 5 | 5 |

Tabla 5.7: Grado de error del conjunto de sensores

| Tipo de Sensor | Modelo | Error |
|----------------|-----------|---------------|
| Humedad | DHT22 | ± 2 % |
| Temperatura | DHT22 | ± 0.5 Celsius |
| Luminosidad | BH1750FVI | ± 20 % |

La Tabla 5.8 muestra los resultados obtenidos de exhaustividad en la detección de eventos en un entorno académico y se obtienen al aplicar la ecuación 2. La evaluación indica resultados favorables con **90 %** de exhaustividad en eventos ambientales y **96 %** en eventos académicos.

Tabla 5.8: Resultados de exhaustividad

| Event | Gold | Tiempo | | Exhaustividad | Persona | | Exhaustividad | Espacio | | Exhaustividad | Exhaustividad Total |
|---------------------|----------|-----------|-------------|-----------------|-----------|-------------|-----------------|-----------|-------------|-----------------|---------------------|
| | Standard | Correctos | Incorrectos | | Correctos | Incorrectos | | Correctos | Incorrectos | | |
| Eventos Ambientales | 643 | 581 | 3 | 0.903576 | 582 | 2 | 0.905132 | 580 | 4 | 0.902021 | 0.90 |
| Eventos Académicos | 166 | 581159 | 7 | 0.957831 | 162 | 4 | 0.975903 | 160 | 6 | 0.963855 | 0.96 |

Medida-F (F1) es una medida armónica que combina los valores de la precisión y de la exhaustividad. Se expresa como en la ecuación 5.3.

$$F1 = 2 * \frac{(Precision)(Exhaustividad)}{Precision + Exhaustividad} \quad (5.3)$$

En la Tabla 5.9 se muestra el resultado de la medida estadística de precisión obtenida de las evaluaciones de precisión y exhaustividad. La evaluación indica resultados prometedores en la *medida-f* con **94 %** en eventos ambientales y **96 %** en eventos académicos.

Tabla 5.9: Resultados de Medida-F

| Event | Tiempo | | | Persona | | | Espacio | | | Promedio |
|---------------------|-----------|---------------|---------------|-----------|---------------|-----------------|-----------|---------------|---------------|-------------|
| | Precisión | Exhaustividad | Medida-F | Precisión | Exhaustividad | Medida-F | Precisión | Exhaustividad | Medida-F | |
| Eventos Ambientales | 0.994863 | 0.903576 | 0.9462 | 0.996575 | 0.905132 | 0.9482 | 0.993150 | 0.902021 | 0.9452 | 0.94 |
| Eventos Académicos | 0.957831 | 0.957831 | 0.957 | 0.975903 | 0.975 | 0.963855 | 0.963855 | 0.963855 | 0.963 | 0.96 |

Capítulo 6

Conclusiones y trabajo futuro

En este trabajo se ha presentado un sistema de ontologías modulares y después de su integración se obtuvo el modelo de un entorno académico con el fin de identificar automáticamente los eventos con la información sobresaliente de los mismos. El modelo presentado es una ontología personalizada para eventos como: difusión, asesorías, cursos y sucesos ambientales.

Las diversas variables consideradas en el modelo son las personas, el tiempo y la ubicación. Además, se implementó una ontología de red de sensores básicamente para permitir detectar personas en los espacios físicos. Los eventos (docencia, investigación y difusión) se detectan utilizando un árbol de decisiones basado en reglas y que fue caracterizado por tres humanos expertos en el dominio académico. La última etapa exhibió un proceso de razonamiento en varios módulos mediante la ejecución de preguntas de competencia que se transformaron en reglas SQWRL.

En esta tesis se llevó a cabo un proceso de evaluación del modelo ontológico personalizado basado en validar la extensibilidad y adaptación de cinco escenarios de casos de uso para eventos en entornos académicos. Y se demostró que el nivel de razonamiento, la consistencia y coherencia del modelo ontológico personalizado para el contexto académico presenta resultados prometedores.

Así mismo, las principales contribuciones de este documento son (a) el sistema de ontologías modulares que consisten en la abstracción del tiempo, ubicación y persona para que juntas integren una ontología personalizada para eventos académicos; (b) una identificación de eventos basada en reglas utilizando un árbol de decisión y datos sin procesar obtenidos de sensores en un ambiente académico; (c) el razonamiento del contexto utilizando primero, preguntas de competencia en lenguaje natural y luego, implementándolas en las reglas SQWRL; (d) el proceso de evaluación se centró en responder las pregunta de competencia, usar escenarios de casos para validar el grado de adaptación del modelo y los principios de diseño de ontologías como consistencia, coherencia y claridad.

6. CONCLUSIONES Y TRABAJO FUTURO

Se llevó a cabo una evaluación final basada en la precisión, que ha mostrado resultados aceptables.

El resultado de este trabajo que consta del modelo ontológico para eventos en entornos académicos y el sistema para la detección de eventos pueden ayudar a los ingenieros del conocimiento a reutilizar nuestra ontología para modelar y razonar el ambiente académico más a fondo. Además, puede ser útil construir un sistema de pregunta respuesta en lenguaje natural en el dominio académico.

Como continuación de este trabajo, quedaría plantear enriquecer el modelo académico añadiendo y caracterizando eventos adicionales, tales como: eventos culturales, eventos relacionados con la seguridad, eventos sobre la salud de los usuarios.

Otro punto interesante sería mejorar la obtención de los datos por parte de la red de sensores, por ejemplo, se podría identificar una persona a través de su dispositivo móvil a partir de la dirección MAC o con algún wearable, con el propósito de agilizar el procesamiento de detección de un individuo y dejar a un lado la tecnología RFID.

Por otro lado, se puede mejorar este proyecto haciendo un filtrado para reducir la cantidad de información que procesan las ontologías, con la finalidad de reducir el tiempo de cómputo. Además, se puede utilizar un sistema híbrido donde las ontologías trabajen en conjunto con una base de datos para el almacenamiento de la información.

Finalmente, también resultaría interesante evaluar la facilidad de la reutilización y validación del modelo en un dominio de salud, modificando los tipos de eventos. Realizar una investigación para crear un entorno académico inteligente que le proporcione servicios adecuados a los usuarios con base al contexto del lugar. Este proyecto es la base para cumplir el objetivo de crear un ambiente sensible al contexto.

Apéndice A

Código/Manuales/Publicaciones

A.1. Apéndice: conjunto de datos para el análisis de resultados

| |
|----------------|
| sí= correcto |
| no= incorrecto |
| na= no aplica |

A. CÓDIGO/MANUALES/PUBLICACIONES

| eventId | Tiempo | Espacio | Persona | Variación |
|------------------------|--------|---------|---------|-----------|
| advising001 | no | si | si | na |
| advising002 | si | si | si | na |
| advising003 | si | si | si | na |
| advising004 | si | si | si | na |
| advising005 | si | no | si | na |
| advising006 | si | si | si | na |
| advising007 | si | si | si | na |
| advising008 | si | si | si | na |
| advising009 | si | si | si | na |
| advising010 | si | si | si | na |
| advising011 | si | si | si | na |
| postgraduateCourse001 | si | si | si | na |
| postgraduateCourse002 | si | si | si | na |
| postgraduateCourse003 | si | si | si | na |
| postgraduateCourse004 | si | si | si | na |
| postgraduateCourse005 | si | si | si | na |
| postgraduateCourse006 | si | si | si | na |
| postgraduateCourse007 | si | si | si | na |
| postgraduateCourse008 | si | si | si | na |
| professorsCourse001 | si | si | si | na |
| undergraduateCourse001 | si | si | si | na |
| undergraduateCourse002 | si | si | si | na |
| undergraduateCourse003 | si | si | si | na |
| undergraduateCourse004 | si | si | si | na |
| undergraduateCourse005 | si | si | si | na |
| undergraduateCourse006 | no | no | no | na |
| discussionPanel001 | si | si | si | na |
| discussionPanel002 | si | si | si | na |
| discussionPanel003 | si | si | si | na |
| discussionPanel004 | si | si | si | na |
| discussionPanel005 | no | no | no | na |
| seminarEvent001 | si | si | si | na |
| workshopEvent001 | si | si | si | na |
| eventHumDec001 | si | si | na | si |
| eventHumDec002 | si | si | na | si |
| eventHumDec003 | si | si | na | si |
| eventHumDec004 | si | si | na | si |
| eventHumDec005 | si | si | na | si |
| eventHumDec006 | si | si | na | si |
| eventHumDec007 | si | si | na | si |

A.1 Apéndice: conjunto de datos para el análisis de resultados

| eventId | Tiempo | Espacio | Persona | Variación |
|----------------|--------|---------|---------|-----------|
| eventHumDec008 | si | si | na | si |
| eventHumDec009 | si | si | na | si |
| eventHumDec010 | si | si | na | si |
| eventHumDec011 | si | si | na | si |
| eventHumDec012 | si | si | na | si |
| eventHumDec013 | si | si | na | si |
| eventHumDec014 | si | si | na | si |
| eventHumDec015 | si | si | na | si |
| eventHumInc001 | si | si | na | si |
| eventHumInc002 | si | si | na | si |
| eventHumInc003 | si | si | na | si |
| eventHumInc004 | si | si | na | si |
| eventHumInc005 | si | si | na | si |
| eventHumInc006 | si | si | na | si |
| eventHumInc007 | si | si | na | si |
| eventHumInc008 | si | si | na | si |
| eventHumInc009 | si | si | na | si |
| eventHumInc010 | si | si | na | si |
| eventHumInc011 | si | si | na | si |
| eventHumInc012 | si | si | na | si |
| eventHumInc013 | si | si | na | si |
| eventHumInc014 | si | si | na | si |
| eventHumInc015 | si | si | na | si |
| eventLumDec001 | si | si | na | si |
| eventLumDec002 | si | si | na | si |
| eventLumDec003 | si | si | na | si |
| eventLumDec004 | si | si | na | si |
| eventLumDec005 | si | si | na | si |
| eventLumDec006 | si | si | na | si |
| eventLumDec007 | si | si | na | si |
| eventLumDec008 | si | si | na | si |
| eventLumDec009 | si | si | na | si |
| eventLumDec010 | si | si | na | si |
| eventLumDec011 | si | si | na | si |
| eventLumDec012 | si | si | na | si |
| eventLumDec013 | si | si | na | si |
| eventLumDec014 | si | si | na | si |
| eventLumDec015 | si | si | na | si |
| eventHumInc001 | si | si | na | si |

A. CÓDIGO/MANUALES/PUBLICACIONES

| eventId | Tiempo | Espacio | Persona | Variación |
|----------------|--------|---------|---------|-----------|
| eventHumInc002 | si | si | na | si |
| eventHumInc003 | si | si | na | si |
| eventHumInc004 | si | si | na | si |
| eventHumInc005 | si | si | na | si |
| eventHumInc006 | si | si | na | si |
| eventHumInc007 | si | si | na | si |
| eventHumInc008 | si | si | na | si |
| eventHumInc009 | si | si | na | si |
| eventHumInc010 | si | si | na | si |
| eventHumInc011 | si | si | na | si |
| eventHumInc012 | si | si | na | si |
| eventHumInc013 | si | si | na | si |
| eventHumInc014 | si | si | na | si |
| eventHumInc015 | si | si | na | si |
| eventLumDec001 | si | si | na | si |
| eventLumDec002 | si | si | na | si |
| eventLumDec003 | si | si | na | si |
| eventLumDec004 | si | si | na | si |
| eventLumDec005 | si | si | na | si |
| eventLumDec006 | si | si | na | si |
| eventLumDec007 | si | si | na | si |
| eventLumDec008 | si | si | na | si |
| eventLumDec009 | si | si | na | si |
| eventLumDec010 | si | si | na | si |
| eventHumInc002 | si | si | na | si |
| eventHumInc003 | si | si | na | si |
| eventHumInc004 | si | si | na | si |
| eventHumInc005 | si | si | na | si |
| eventHumInc006 | si | si | na | si |
| eventHumInc007 | si | si | na | si |
| eventHumInc008 | si | si | na | si |
| eventHumInc009 | si | si | na | si |
| eventHumInc010 | si | si | na | si |
| eventHumInc011 | si | si | na | si |
| eventHumInc012 | si | si | na | si |
| eventHumInc013 | si | si | na | si |
| eventHumInc014 | si | si | na | si |
| eventHumInc015 | si | si | na | si |
| eventLumDec001 | si | si | na | si |
| eventLumDec002 | si | si | na | si |
| eventLumDec003 | si | si | na | si |
| eventLumDec004 | si | si | na | si |
| eventLumDec005 | si | si | na | si |

A.1 Apéndice: conjunto de datos para el análisis de resultados

| eventId | Tiempo | Espacio | Persona | Variación |
|----------------|--------|---------|---------|-----------|
| eventLumDec005 | si | si | na | si |
| eventLumDec006 | si | si | na | si |
| eventLumDec007 | si | si | na | si |
| eventLumDec008 | si | si | na | si |
| eventLumDec009 | si | si | na | si |
| eventLumDec010 | si | si | na | si |
| eventHumInc002 | si | si | na | si |
| eventHumInc003 | si | si | na | si |
| eventHumInc004 | si | si | na | si |
| eventHumInc005 | si | si | na | si |
| eventHumInc006 | si | si | na | si |
| eventHumInc007 | si | si | na | si |
| eventHumInc008 | si | si | na | si |
| eventHumInc009 | si | si | na | si |
| eventHumInc010 | si | si | na | si |
| eventHumInc011 | si | si | na | si |
| eventHumInc012 | si | si | na | si |
| eventHumInc013 | si | si | na | si |
| eventHumInc014 | si | si | na | si |
| eventHumInc015 | si | si | na | si |
| eventLumDec001 | si | si | na | si |
| eventLumDec002 | si | si | na | si |
| eventLumDec003 | si | si | na | si |
| eventLumDec004 | si | si | na | si |
| eventLumDec005 | si | si | na | si |
| eventLumDec006 | si | si | na | si |
| eventLumDec007 | si | si | na | si |
| eventLumDec008 | si | si | na | si |
| eventLumDec009 | si | si | na | si |
| eventLumDec010 | si | si | na | si |
| eventLumDec011 | si | si | na | si |
| eventLumDec012 | si | si | na | si |
| eventLumDec013 | si | si | na | si |
| eventLumDec014 | si | si | na | si |
| eventLumDec015 | si | si | na | si |
| eventLumInc001 | si | si | na | si |
| eventLumInc002 | si | si | na | si |
| eventLumInc003 | si | si | na | si |
| eventLumInc004 | si | si | na | si |

A. CÓDIGO/MANUALES/PUBLICACIONES

| eventId | Tiempo | Espacio | Persona | Variación |
|-----------------|--------|---------|---------|-----------|
| eventLumInc005 | si | si | na | si |
| eventLumInc006 | si | si | na | si |
| eventLumInc007 | si | si | na | si |
| eventLumInc008 | si | si | na | si |
| eventLumInc009 | si | si | na | si |
| eventLumInc010 | si | si | na | si |
| eventLumInc011 | si | si | na | si |
| eventLumInc012 | si | si | na | si |
| eventLumInc013 | si | si | na | si |
| eventLumInc014 | si | si | na | si |
| eventLumInc015 | si | si | na | si |
| eventTempDec001 | si | si | na | si |
| eventTempDec002 | si | si | na | si |
| eventTempDec003 | si | si | na | si |
| eventTempDec004 | si | si | na | si |
| eventTempDec005 | si | si | na | si |
| eventTempDec006 | si | si | na | si |
| eventTempDec007 | si | si | na | si |
| eventTempDec008 | si | si | na | si |
| eventTempDec009 | si | si | na | si |
| eventTempDec010 | si | si | na | si |
| eventTempDec011 | si | si | na | si |
| eventTempDec012 | si | si | na | si |
| eventTempDec013 | si | si | na | si |
| eventTempDec014 | si | si | na | si |
| eventTempDec015 | si | si | na | si |
| eventTempInc001 | si | si | na | si |
| eventTempInc002 | si | si | na | si |
| eventTempInc003 | si | si | na | si |
| eventTempInc004 | si | si | na | si |
| eventTempInc005 | si | si | na | si |
| eventTempInc006 | si | si | na | si |
| eventTempInc007 | si | si | na | si |
| eventTempInc008 | si | si | na | si |
| eventTempInc009 | si | si | na | si |
| eventTempInc010 | si | si | na | si |
| eventTempInc011 | si | si | na | si |
| eventTempInc012 | si | si | na | si |
| eventTempInc013 | si | si | na | si |

A.1 Apéndice: conjunto de datos para el análisis de resultados

| eventId | Tiempo | Espacio | Persona | Variación |
|------------------|--------|---------|---------|-----------|
| eventTempInc014 | si | si | na | si |
| eventTempInc015 | si | si | na | si |
| eventEntrance001 | si | si | si | na |
| eventEntrance002 | si | si | si | na |
| eventEntrance003 | si | si | si | na |
| eventEntrance004 | si | si | si | na |
| eventEntrance005 | si | si | si | na |
| eventEntrance006 | si | si | si | na |
| eventEntrance007 | si | si | si | na |
| eventEntrance008 | si | si | si | na |
| eventEntrance009 | si | si | si | na |
| eventEntrance010 | si | si | si | na |
| eventEntrance011 | si | si | si | na |
| eventEntrance012 | si | si | si | na |
| eventEntrance013 | si | si | si | na |
| eventEntrance014 | si | si | si | na |
| eventEntrance015 | si | si | si | na |
| eventLeaves001 | si | si | si | na |
| eventLeaves002 | si | si | si | na |
| eventLeaves003 | si | si | si | na |
| eventLeaves004 | si | si | si | na |
| eventLeaves005 | si | si | si | na |
| eventLeaves006 | si | si | si | na |
| eventLeaves007 | si | si | si | na |
| eventLeaves008 | si | si | si | na |
| eventLeaves009 | si | si | si | na |
| eventLeaves010 | si | si | si | na |
| eventLeaves011 | si | si | si | na |
| eventLeaves012 | si | si | si | na |

Bibliografía

- [1] Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing? *International journal of human-computer studies*, 43(5-6), 907-928. Recuperado de <https://www.sciencedirect.com/science/article/pii/S1071581985710816>. 7
- [2] Duran, J., Conesa, J. and Clarisó, R. (s.f.) *Ontologías y web semántica*. Recuperado de <http://www.cartagena99.com/recursos/alumnos/temarios/Ontologias> 7
- [3] Guarino, N. (1998). Formal ontology and information systems En Guarino, N. (Ed.), *Formal ontology in Information Systems* (3-15). 7
- [4] OWL Working Group. (2009). OWL Web Ontology Language Overview. [documentación]. Recuperado de <https://www.w3.org/TR/owl-features/>. 9
- [5] Barro, S. and Bugarín A. (2002). *Fronteras de la computación*. Madrid, España: Ediciones Díaz de Santos. 9
- [6] Hernández-Calderón, J. G., Benítez-Guerrero, E., and Mezura-Godoy, C. (2014). Ambientes inteligentes en contextos educativos: modelo y arquitectura. *Research in Computing Science*, 77, 55-65. 9, 17
- [7] Gascón, D. (2010). Redes de sensores inalámbricos, la tecnología invisible. *Bit*, (180-181), 53-55. Recuperado de <http://www.libelium.com/libelium-downloads/libelium-bit-coit.pdf>. 10
- [8] Hu, F. and Cao, X. (2010), *Wireless sensor networks: principles and practice*. Recuperado de <https://www.taylorfrancis.com/books/9780429111488>. 10
- [9] Barranco, R. (2011). IBM ISII Complex Event Processing: De la detección de eventos a la acción inmediata. Recuperado de <https://www.ibm.com/developerworks/ssa/local/im/ssa/identity-insight-complex-event-processing/index.html>. 11
- [10] Reyes, J. A. (2013). Creación automática de Ontologías a partir de Textos con un Enfoque Lingüístico. (Tesis doctoral). Departamento de Ciencias Computacionales, Morelos, México. 11

BIBLIOGRAFÍA

- [11] Miller, G. A., and Johnson-Laird, P. N. (1976). *Language and perception*. Cambridge, MA, England: Belknap Press. [11](#)
- [12] Allen, J. F., and Ferguson, G. (1994). Actions and events in interval temporal logic. *Journal of logic and computation*, 4(5), 531-579. [11](#)
- [13] Galton A., Augusto J.C. (2002) Two Approaches to Event Definition. En: Hamerlain A., Cicchetti R., Traummüller R. (Eds) *Database and Expert Systems Applications*. DEXA 2002. Lecture Notes in Computer Science, vol 2453. Springer, Berlin, Heidelberg. [11](#)
- [14] Sowa, J. F. (2000). *Knowledge representation: logical, philosophical, and computational foundations*. Pacific Grove: Brooks/Cole. [11](#)
- [15] Zwaan, R. A., Langston, M. C., and Graesser, A. C. (1995). The construction of situation models in narrative comprehension: An event-indexing model. *Psychological science*, 6(5), 292-297. Recuperado de <https://journals.sagepub.com/doi/abs/10.1111/j.1467-9280.1995.tb00513.x>. [11](#)
- [16] Hípola, P. and Vargas-Quesada, B. (1999). Agentes inteligentes: definición y tipología. Los agentes de información. *El profesional de la información*, 8(4), 13-21. Recuperado de <http://eprints.rclis.org/18300/>. [12](#)
- [17] Skocir, P. Krivic, M. Tomelj, M. Kusek, M. and Jezic, G. (2016). Activity detection in smart home environment. *Procedia Computer Science*, (96), 672-681. Recuperado de <https://www.sciencedirect.com/science/article/pii/S1877050916320609>. [15](#)
- [18] Shinmoto, R., Shi, Q., Van den Hengel, A. and Ranasinghe, D. (2017). A hierarchical model for recognizing alarming states in a batteryless sensor alarm intervention for preventing falls in older people. *Pervasive and Mobile Computing*, 40, 1-16. Recuperado de <https://www.sciencedirect.com/science/article/pii/S1574119216302206>. [15](#)
- [19] Meditskos, G. and Kompatsiaris, I. (2017). iKnow: Ontology-driven situational awareness for the recognition of activities of daily living. *Pervasive and Mobile Computing*, 40, 17-41. Recuperado de <https://www.sciencedirect.com/science/article/pii/S157411921630195X>. [16](#)
- [20] Chen, L. and Nugent, C. (2009). Ontology-based activity recognition in intelligent pervasive environments. *International Journal of Web Information Systems*, 5(4), 410-430. Recuperado de <https://www.emeraldinsight.com/doi/pdfplus/10.1108/17440080911006199>. [16](#)

- [21] Hromic, H., Le Phuoc, D., Serrano, M., AntoniĆ, A., Źarko, I. P., Hayes, C. and Decker, S. (junio, 2015). Real time analysis of sensor data for the internet of things by means of clustering and event processing. Trabajo presentado en *IEEE International Conference on Communications*, Londres, Reino Unido. Recuperado de <https://ieeexplore.ieee.org/abstract/document/7248401>. 16
- [22] Stocker, Markus and Rönkkö, Mauno and Kolehmainen, Mikko. (2014). Abstractions from Sensor Data with Complex Event Processing and Machine Learning. 16
- [23] Calbimonte, J. P., Ranvier, J. E., Dubosson, F., and Aberer, K. (2017). Semantic representation and processing of hypoglycemic events derived from wearable sensor data. *Journal of Ambient Intelligence and Smart Environments*, 9(1), 97-109. Recuperado de <https://hesso.tind.io/record/2784>. 17
- [24] Gu, T., Wang, X. H., Pung, H. K. and Zhang, D. Q. (2004). An ontology-based context model in intelligent environments. Trabajo presentado en *Proceedings of communication networks and distributed systems modeling and simulation*. Recuperado. 17
- [25] Camacho, D. and Novais, P. (2017). Innovations and practical applications of intelligent systems in ambient intelligence and humanized computing. *Journal of Ambient Intelligence and Humanized Computing* 8(2), 155-156. Recuperado de <https://link.springer.com/article/10.1007/s12652-017-0454-z>. 17
- [26] Kanagarajan, S. and Ramakrishnan, S. Integration Of Internet-Of-Things Facilities And Ubiquitous Learning For Still Smarter Learning Environment. *Mathematical Sciences International Research Journal*, 5(2), 286-289. 18
- [27] Durães, D., Castro, D., Bajo, J., and Novais, P. (junio, 2017). Modelling an Intelligent Interaction System for Increasing the Level of Attention. Trabajo presentado en *International Symposium on Ambient Intelligence*. 18
- [28] Cruz, I. (2019). Extracción y enriquecimiento de perfiles de investigación usando ontologías (Tesis de maestría). Universidad Autónoma Metropolitana – Azcapotzalco, Ciudad de México. 31
- [29] Gómez-Pérez, A. (1996). Towards a framework to verify knowledge sharing technology. *Expert Systems with Applications*, 11(4), 519-529. 64

Artículo Publicado

Artículo de revista titulado “Ontology for Academic Context Reasoning” en *Procedia Computer Science*, vol. 141, 2018, pp. 175-182. Indizado en *Conference Proceedings Citation Index*, *SCImago Journal and Country Rank* y *Scopus*.



The 9th International Conference on Emerging Ubiquitous Systems and Pervasive Networks
(EUSPN 2018)

Ontology for Academic Context Reasoning

Maricela Bravo*, José A. Reyes-Ortiz, Isabel Cruz-Ruiz, Ariadna Gutiérrez-Rosales, and
Josué Padilla-Cuevas

Autonomous Metropolitan University, San Pablo 180, Azcapotzalco, CDMX, México

Abstract

Ontologies have gained popularity in the scientific community as representational mechanisms to support intelligent reasoning and execute inferences. In this paper we describe an ontology designed specifically to represent academic contexts at a public university. This model consists of a collection of ontologies designed to represent persons, physical space, sensor networks, events, etc. among other entities that exist in the academic environment. In particular, we describe the design requirements that guided the construction of the ontologies. The resulting ontology model is evaluated considering the competency of the ontology, and the concept domain coverage. Results are promising and the set of competency questions are translated to queries showing that the ontology model adheres to the requirements.

© 2018 The Authors. Published by Elsevier Ltd.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>) Selection and peer-review under responsibility of the scientific committee of EUSPN 2018.

Keywords: Context ontology; Ontology design; Ontology competency

1. Introduction

The myriad of daily activities occurring at universities generate a high demand for information services, which causes the requirement to process large volumes of data and offer information to users in a fast, concise, and pertinent manner. These information requirements vary according to the context of the user, the reasons why he / she is in a given building of the institution, and the time in which an event happens. Commonly, users who demand information services at the university are teachers, researchers, students, employees and visitors. The type of information requested ranges from course schedules, location of facilities, location of professors, etc. Therefore, the

* Corresponding author. Tel.: +52 55 53 18 95 32 ext. 149. *E-mail*
address: mcbc@correo.azc.uam.mx

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>) Selection and peer-review under responsibility of the scientific committee of EUSPN 2018.
10.1016/j.procs.2018.10.165

processing of users contexts must consider various situations and characteristics: a) public universities count with wireless network technologies through which they offer Internet access and diverse applications for users at the university; b) most people at the university have mobile devices capable of wireless network connection; and c) universities count with sufficient computing resources to support the user's duties. Given these conditions, it is possible to develop the computational infrastructure to facilitate smart interconnection of devices and data processing, that is, to implement Internet of Things (IoT) technology. However, the implementation of the IoT presents a series of challenges that must be solved so that the interconnection of objects and people in a university can be efficiently exploited. Among the most relevant problems to solve are:

- a) Context management and service personalization. Context management consists of context acquisition, context processing, and context reasoning. Service personalization is the intelligent outcome of context management.
- b) Event detection, management, and activation. Event management requires the representation of events at the academic environment in such a way that it is possible to know *What* happened? *When* happened? *Where*? and *Who* was involved? Handling of events requires the detection, registration and activation of actions in response to events.
- c) The efficient interaction between intelligent information systems and hardware. This interaction requires interoperability between communication protocols, and the dynamic interconnection of mobile devices with wireless networks and sensors/actuators networks.

There are other relevant and derived problems. However, in this article we present an ontology solution approach for the representation and management of contexts and events in the university environment addressing particularly academic issues. Ontologies were selected as the formal representational mechanism as they are very popular in IoT based applications. Ontologies facilitate reusability, knowledge sharing, and execution of formal reasoning tasks such as satisfiability of concepts, consistency checking, classification and inference.

2. Related Work

The revision of the related work was done emphasizing the use of ontologies for context representation, management and reasoning. In 2003 Chen, Finin, and Joshi [1] described CoBrA, a context broker agent architecture that is capable of managing a shared model of the context and reasoning support for context-aware applications. Later in 2004 authors detailed the SOUPA ontology [2] which consists of vocabularies for describing person contact information, beliefs, desires, and intentions of an agent, actions, policies, time, space, and events. Razmerita, Angehrn, and Maedche [3] presented in 2003 OntobUM, a generic ontology-based user modeling architecture. This architecture integrates three ontologies: the user ontology, the domain ontology, and the log ontology. Later in 2007 [4] authors augmented their OntobUM model by representing the behavior of the user, such as: level of activity, type of activity, level of knowledge sharing, etc. Wang et al. [5] described in 2004 CONON, an ontology for modeling context in pervasive computing environments divided into upper ontology and specific ontology. The upper ontology model defines computational entity, location, person and activity as the most important entities of a context model. Later in 2004 [6] authors presented SOCAM, a Service-Oriented Context Aware Middleware architecture to support the construction of context-aware services in intelligent environments. SOCAM architecture incorporates CONON ontology. Preuveneers et al. [7] presented in 2004 CoDAMoS, an extensible context ontology for ambient intelligence, which describes four main concepts: user, environment, platform, and service. Authors described the requirements for ambient intelligence: application adaptability, resource awareness, mobile services, semantic service discovery, code generation, and context-aware user interfaces. In 2007 Ejigu et al. [8] presented an ontologybased context model to facilitate context reasoning by providing structure for contexts, rules and their semantics. This work takes the basic elements of a pervasive computing environment characterized by their dynamicity, heterogeneity, and ubiquity of users, devices and resources, ad hoc connection between devices; and the existence of logical and physical sensors. The sources of context information are implemented as classes: User, Device, Application, Physical Environment, Resource, Network, Location, and Activity; all these classes are subclasses of a general class Context. In 2010 Zainol and Nakata [9] presented a Generic Context Ontology Model to represent context information in general. The aim is to facilitate the common context representation, context matching and context reasoning. This approach is based on the idea that a well-defined context model will minimize the complexity of context-aware systems enhancing their maintainability. This model consists of a formal specification of the semantics of context identifiers; allowing sharing knowledge among different resources. In 2010 PovedaVillalón et al. [10] presented mIO! an ontology network for a mobile environment. mIO! Ontology consists

of eleven modular ontologies: user, role, environment, location, time, service, provider, device, interface, source, and network. This ontology covers a wide range of concepts related with context representation. Skillen et al. [11] presented in 2012 a user profile model for context-aware application personalization; authors concentrated on concepts to model a dynamic context: user time, user location, user activity, and user context. In 2013 Guermah et al. [12] described an architecture for the development of context-aware services based on ontologies. This architecture is composed of three main elements: a meta-model of context, an ontology for the meta-model, and a reasoning engine. The context is modeled based on a meta-model that defines the context and sub-contexts. Context properties are gathered from sensors, each property has a context validity and context specification. In 2014 Nadozeva and Kiritsis [13] presented an ontology-based context model to capture general concepts about users and business. The aim of this work is to propose an ontology-based model and rules to classify the context of users and business. This ontology describes general concepts: space, matter, object, event and action. Domain specific extensions specify the vocabulary and properties related to a generic domain by specializing terms in the upper ontology. This model is composed of three sub-models: user context, business context, and information feature model. In 2015 Kayes, A. Han, J, Colman A. [14] presented Ontology-based Context-Aware Access Control (OntCAAC) a generic framework that models dynamic contexts and access control policies. The aim is to use a policy model for specifying and enforcing context-aware access control. The OntCAAC provides the capability to control access for software services and resources by taking into account the context information. This ontology defines the general context entity classes: User, Role, Source, Owner, Relationships, Place, EnvPerson and Device. It also defines the different types of dynamic context information: ContextInfo, LocationInfo, TemporalInfo. The place ontology identifies different buildings, departments and rooms of the hospital. The Relationship ontology represents the relations between users: Person-Centric or Location-Centric. The former is used for represent relationship between users, the latter represents that the concerned people are collocated. The Status Information Ontology stores the health status, the current personal status, the current location status. Miraoui et al. [15] proposed in 2015 an ontology based on modeling a smart living room environment and their contextual information for enabling a common understanding of context and enhancing its sharing. Authors propose a definition of context for service oriented systems as any information that triggers a service or changes the quality of a service if its values change. Based on this concept modeling a smart room starts by specifying the services that each equipment can provide and the set of information that triggers the service.

None of revised works fully achieve the requirements specification, majority of ontologies include information about person or users, but do not consider IoT-based identification such as RFID tags or MAC address, for instance [2], [3], [5], [6], and [7]. Majority of revised related ontologies consider the geographical localization, but do not correlate localization with physical spaces and persons [3] and [7] not allowing the automatic identification and localization of persons and objects. Another important requirement was the incorporation of networks (computerbased networks and sensor networks), which was not considered. And finally, events or activities management is important, few reported works considered. All these requirements are necessary for context reasoning.

3. Ontology Design Methodology

In this section, the methodology that was defined and executed for design, construction and evaluation of the ontology is described. An initial set of competency questions were used for term elicitation and for final competency evaluation. Ontology design encompasses three stages: specification of ontology requirements, ontology construction, and ontology evaluation.

3.1. Specification of Ontology Requirements

In order to specify a set of initial ontology requirements, we reviewed the concepts of context and events management. Abowd et al. [16] compared and analyzed different definitions of context, and presented their definition as follows: “Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves”. From this definition we consider that the entities (or objects) present at an academic environment are: **Person** (for example professors, students, staff, etc.), **Physical space** (classroom, building, laboratory, bathroom, etc.), and computing **Devices** and **Networks**, among others. Schilit et al. [17] stated that the tree important aspects of context are: where you are, who you are with, and what resources are nearby. From this approach, a context should include information about the geographical localization of objects and persons; together with the accurate identification of persons and objects. Hanzal et al. [18] distinguish between objects and events (both important concepts

related with the concept of context [16] [17]). They clarify the distinction between objects and events from a philosophical perspective stating that objects are *continuants* (they exist and persist through time) and events are *occurrents* (they happen or take place at some point). This clarification of the concept *event* emphasizes the need to represent not only objects or entities, but events occurring at a point, this means that the ontology model should include the concept of *Time*. From this initial analysis we have defined the concept coverage requirements of the ontology and defined the main objective of the ontology, which is to facilitate intelligent context processing in the academic environment.

3.1.1. Concept coverage

Based on the definitions of context and event given in [16] and [17] the ontology model should include the following concepts:

1. Person profile information to represent the user data that is possible to gather from public networks, public Web pages, or public data bases available such as DBLP.
2. Data for the identification of persons such as id credentials, RFID tags, MAC address, and passwords provided.
3. Data for the identification of objects (RFID Tags, MAC address, etc.)
4. Data for geographical localization of persons, objects or physical locations. In order to enable localization services, this data should consist of longitude, latitude and height.
5. Physical space to represent the buildings organization with their precise geographical coordinates.
6. Sensor networks to represent the intranet and private nets organization as they are currently arranged into the institution.
7. Environmental data to represent the physical sensor measures located at the different physical locations.
8. Device to represent any kind of hardware device available including personal computers, microcontrollers, cellular and any device capable of data processing.
9. Events that may occur at an academic environment which result of interest for the users at the university. Events should be correlated with time, physical space, the location, and the person involved.

4. Ontology Construction

The ontology was incrementally implemented using the Protégé ontology editor, and represented using the standard Web Ontology language (OWL). The resulting ontology model consists of a set of modular ontologies among which a set of semantic relations are defined to support intelligent context reasoning.

4.1. Person Ontology

Person ontology represents all possible human users that may be present at the university in a moment or period of time, such as: visitor, professor, student, employee, etc. Additionally, this ontology included the concepts of *Department* and *AcademicTitle* to improve the context descriptions with useful attributes for *Academics*. Figure 1 shows the class hierarchy of the *Person* ontology. An important characteristic of this ontology was to define a unique identifier for every type of person that would be present inside the sensor-enabled context. The concept *Person* is defined as an equivalence through the *hasName* and *hasGender* data properties, indicating that every person should provide his name and gender. The concept *Employee* is defined as a sub class of a *Person* that *hasEconomicNumber* data property. Whereas the concept *Student* is defined as a sub class of *Person* that *hasStudentId*. An important concept is a *Professor* which is an *Academic*, is an *Employee* and is a *Person* that *hasCategory*, *hasDepartment*, and *hasEmail*; and inherits the data property of an *Academic hasProject*. The class hierarchy of the *Person* ontology shows the sub-classification of the class *Student* into *RegularStudent* and *AssistantStudent*. This classification addresses a particular need to represent the two types of students that exist in the university where the *AssistantStudent* is an *Academic*, is an *Employee* and a *Student*.

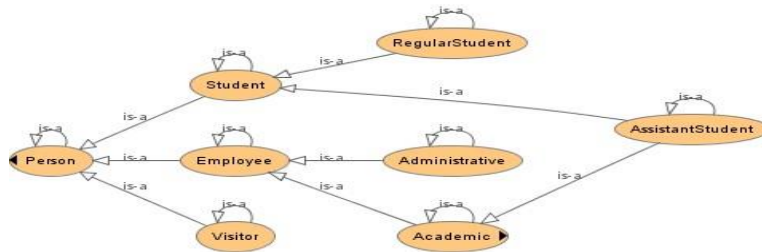


Fig. 1. Class hierarchy of the *Person* Ontology.

4.2. *Physical Space Ontology*

The *PhysicalSpace* ontology represents any kind of physical location such as buildings, cubicles, classrooms, offices, parking lots, plazas, green areas, etc. The class hierarchy of the ontology shown in Figure 2 contains as a main concept the *PhysicalSpace* class to conceptualize any kind of physical space which is divided into two main sub-class definitions: *Internal* and *External*. An internal physical space is used to represent closed rooms; whereas external physical space is used to represent open spaces such as: hallway, green areas, parking, etc. The set of data properties that were defined are: *hasName*, *hasDoorState*, *hasAirConditioner*, *hasLampsNumber*, *hasLevel*, *hasPeopleCapacity*, *hasCarCapacity*, *hasProjector*, *hasService*, *hasWindow*, *isOpen*, *has Area*, among other properties. Two important relationships between concepts (object properties) defined for this class are: *isLocatedInto*, this property is used to specify that a physical space is part of another physical space, enabling physical objects composed of physical objects. For instance a classroom *isLocatedInto* a building; and the property *isBesideOf*, which is used to specify neighboring localities, for instance a cubicle A *isBesideOf* cubicle B and cubicle B *isBesideOf* bathroom 2.

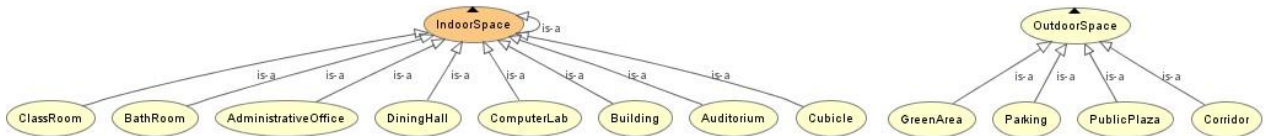
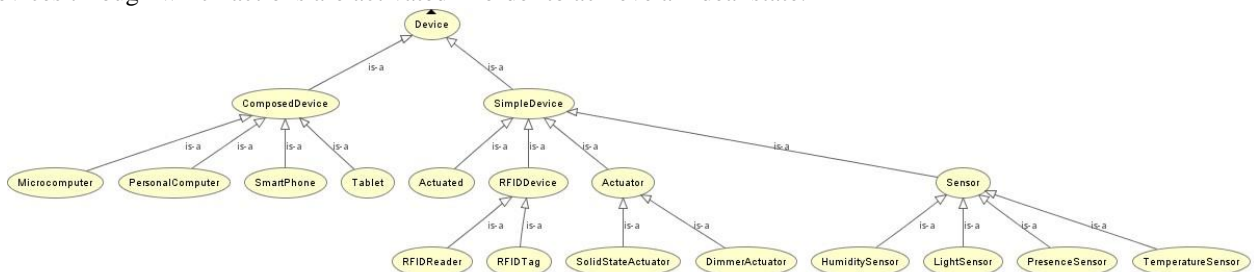


Fig. 2. Class hierarchy of the *PhysicalSpace* class.

4.3. *Sensor Network Ontology*

The *SensorNetwork* ontology represents a collection of physical sensors, mobile sensors and actuators. The objective of the network sensor is to obtain data from the physical context, user context and eventually activate some actuators. This ontology aims at representing environmental data such as temperature, lighting, humidity, and presence of humans into the environment. Another important objective of this model is the possible identification of the users and the data generated by user interaction with the environment. The following types of sensors are considered: *environmental sensors*, which are used to obtain data room temperature, humidity, luminosity, and presence of persons; *mobile and wearable sensors*, such as: accelerometer, gyroscope, magnetometer, proximity sensor, light sensor, barometer, thermometer, pedometer, heart rate monitor, fingerprint sensors, etc.; *identification* sensors carried by the user, which will serve both the user ID to the acquisition of additional information; *actuators* represent the hardware devices through which actions are activated in order to achieve an ideal state.



4.4. Time Ontology

Time representation is an important requirement to manage events in the academic context. Figure 4 shows the structure of this ontology which consists of a class *TemporalEntity* sub divides into the concepts of *Instant* and *Interval* as follows: an *Instant* is defined as an individual that has the data attributes of *hasYear*, *hasMonth*, *hasDay*, *hasHour* and *hasMinute*; whereas *Interval* is defined with the object properties *hasBeginning* with range *Instant* and *hasEnd* with range *Instant*.

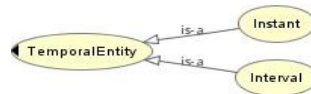


Fig. 4. Time Ontology.

5. Context Reasoning

Ontology reasoning consists of executing a program to infer logical consequences from a set of asserted facts or axioms. In order to realize the intelligent context reasoning the set of ontologies described above were imported and integrated into the *IntelligentEnvironment* ontology (see Fig. 5). This ontology was completed with additional class hierarchies, data properties and object properties. An important class hierarchy that was included into the ontology is the *Event* class to define a variety of events that may occur at the academic environment. A *Presence* event is sub divided into entering and leaving a physical space. These concepts are of great importance for the identification of persons into the academic intelligent environment. A Presence Event is defined through the data and object properties: *hasDescription*, *hasPersonInvolved* (*Person* class), *hasTime* (*TemporalEntity* class), and *happensIn* (*PhysicalSpace* class). All these semantic relationships are useful to answer any question regarding events occurring in the academic environment. What happened? When? Who was involved? At what time occurred the event?

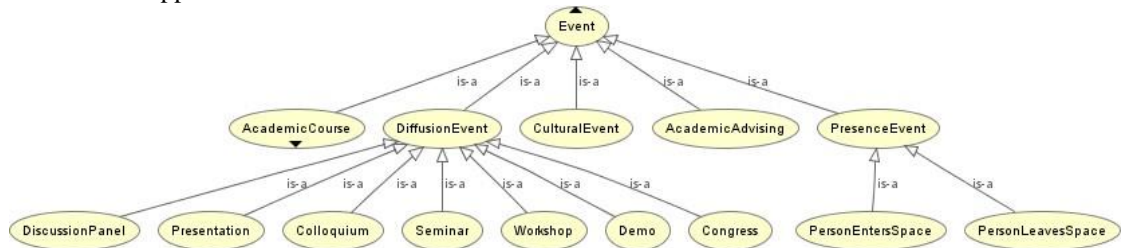


Fig. 5. The Event class hierarchy.

6. Ontology Evaluation

Ontology Evaluation [19] concerns the correct building of the ontology, ensuring that its definitions correctly implement the ontology requirements and competency questions. For evaluation two important aspects are considered: the *competence of the ontology*, that is, if it is able to respond to a set of competency questions; and the verification of *requirements compliance*. The following competency questions were correctly answered:

1. Who is present in the classroom E313?
 Person and (personLocatedIn value classRoomE313)
2. Where is the cubicle of professor Alejandro Reyes?
 Cubicle and (hasEmployeeAssigned some (hasName value "REYES ORTIZ JOSE ALEJANDRO"))
3. Where is professor Maricela?
 PhysicalSpace and (hasPersonDetected some (Professor and hasName value "BRAVO CONTRERAS MARICELA CLAUDIA"))

4. At what time did Professor Alejandro Reyes Ortiz leave the Babbage computing laboratory?
Instant and (instantPersonLeaves some (hasPersonInvolved some (hasName value "REYES ORTIZ JOSE ALEJANDRO")))
5. Who is the professor responsible of Distributed Systems course?
Professor and (isProfessorResponsibleOfCourse some (hasEventName value "Curso Sistemas Distribuidos"))

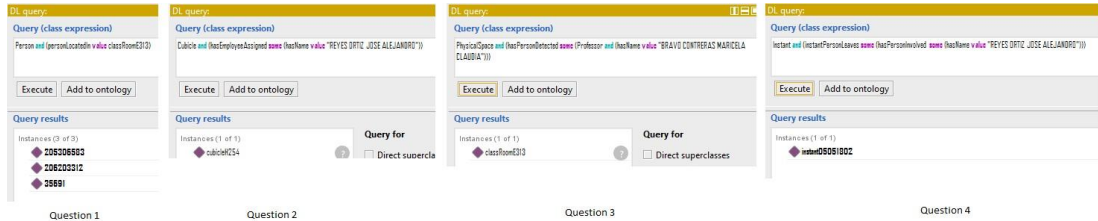


Fig. 6. Answers to competency questions using DL Query tool.

6. How many publications does the professor with economical number “14233” has?
Professor(?prof1) ^ hasPublish(?prof1, ?pub) ^ hasEconomicNumber(?prof1, ?e) ^ swrlb:equal(?e, "14233") -> sqwrl:count(?pub)
7. With whom the professor identified with economical number “14233” collaborates with?
Professor(?p1) ^ collaborateWith(?p1, ?p2) ^ hasEconomicNumber(?p1, ?e) ^ swrlb:equal(?e, "14233") -> sqwrl:select(?p1, ?p2)
8. How many women professors from Systems Department have published at least one article?
Professor(?prof) ^ hasGender(?prof, ?gen) ^ swrlb:equal(?gen, "FEMENINO") ^ hasDepartment(?prof, ?dep) ^ swrlb:equal(?dep, "SISTEMAS") ^ hasPublish(?prof, ?pub) -> sqwrl:count(?prof)

6.1. Evaluation of the Requirements

Ontology requirements were fully attended as follows: Person profile information and data for the automatic identification of persons were included in the *Person* ontology. The particular properties *hasRFIDTag* with domain *Employee* and *Student* and range *RFIDTag*, *hasMACAddress*, and *hasIPAddress* allow the automatic identification of persons and objects at the intelligent environment. Geographical localization of persons, objects or physical locations is included in the *PhysicalSpace* ontology by means of the following properties: *isLocatedInto*, *isBesideOf*, *hasLatitude*, *hasLongitude*, and *hasAltitude*. Network and Device is represented at the *SensorNetwork* ontology. Environmental data is represented at the *PhysicalMeasure* ontology. Events are represented at the *IntelligentEnvironment* ontology, in this ontology events are correlated with time, physical space, and persons.

7. Conclusions

The ontology model reported in this paper is envisioned for a wireless networked environment, where users may be identified by their mobile device mac address or by an RFID card. Such an environment may be an office or laboratory into an academic institution or university, where users enter and leave the environment freely. The ontology was constructed from scratch, because the full list of requirements was not included in any of the reported works. For evaluation, a set of competency questions were translated to queries and executed to show that the ontology is complete and is capable of context reasoning. Results and the evaluation of the ontology model show promising advances towards the construction of an integral IoT platform.

References

- [1] Chen, H., Finin, T., & Joshi, A. An ontology for context-aware pervasive computing environments. The knowledge engineering review, 18 (03), 197-207, (2003).

- [2] Chen, H., Finin, T., & Joshi, A. (2005). The SOUPA ontology for pervasive computing. In *Ontologies for agents: Theory and experiences* (pp. 233-258). Birkhäuser Basel.
- [3] Razmerita, L., Angehrn, A., & Maedche, A. Ontology-based user modeling for knowledge management systems. In *International Conference on User Modeling* (pp. 213-217). Springer Berlin Heidelberg, (2003).
- [4] Razmerita, L. Ontology-based user modeling. In *Ontologies* (pp. 635-664). Springer US, (2007).
- [5] Wang, X. H., Zhang, D. Q., Gu, T., & Pung, H. K. Ontology based context modelling and reasoning using OWL. In *Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on* (pp. 18-22). IEEE, (2004).
- [6] Gu, T., Wang, X. H., Pung, H. K., & Zhang, D. Q. An ontology-based context model in intelligent environments. In *Proceedings of communication networks and distributed systems modeling and simulation conference* (Vol. 2004, pp. 270-275), (2004).
- [7] Preuveneers, D., Van den Bergh, J., Wagelaar, D., Georges, A., Rigole, P., Clerckx, T., ... & De Bosschere, K. Towards an extensible context ontology for ambient intelligence. In *European Symposium on Ambient Intelligence* (pp. 148-159). Springer Berlin Heidelberg, 2004.
- [8] D. Ejigu, M. Scuturici and L. Brunie, An Ontology-Based Approach to Context Modeling and Reasoning in Pervasive Computing, *Pervasive Computing and Communications Workshops, 2007. PerCom Workshops '07. Fifth Annual IEEE International Conference on*, White Plains, NY, pp. 14-19, (2007).
- [9] Zainol, A., Nakata, K., Generic Context Ontology Modelling: A review and Framework, on *2nd International Conference on Computer Technology and Development (ICCTD)*, pp 126-130, (2010).
- [10] Poveda Villalon, M., Suárez-Figueroa, M. C., García-Castro, R., & Gómez-Pérez. A context ontology for mobile environments. (2010).
- [11] Skillen, K. L., Chen, L., Nugent, C. D., Donnelly, M. P., Burns, W., & Solheim, I. (2012, December). Ontological user profile modeling for context-aware application personalization. In *International Conference on Ubiquitous Computing and Ambient Intelligence* (pp. 261-268). Springer Berlin Heidelberg.
- [12] Guermah, H., Fissaa, T., Hafiddi, H., Nassar, M., & Kriouile, A, Context modeling and reasoning for building context aware services, In *Computer Systems and Applications (AICCSA), 2013 ACS International Conference on*, pp. 1-7, (2013).
- [13] Nadoveza A., Kiritsis D., *Ontology-Based Approach for Context Modeling in Enterprises*, *Computer in Industry*, pp. 1218-1231, (2014).
- [14] Kayes, A., Han J., Colman, A. OntCAAS: An Ontology-Based Approach to Context-Aware Access Control for Software Services, *The Computer Journal*, v. 58, No 11, (2015).
- [15] Miraoui, M., El-etriby, S., Tadj, Ch., Zaid Abid, A., *Ontology-Based Context Modeling for a Smart Living Room*, *Proceedings of the World Congress on Engineering and Computer Science 2015 Vol I WCECS 2015*, October 21-23, 2015, San Francisco, USA
- [16] Abowd, G. D., Dey, A. K., Brown, P. J., Davies, N., Smith, M., & Steggle, P. (1999, September). Towards a better understanding of context and context-awareness. In *International Symposium on Handheld and Ubiquitous Computing* (pp. 304-307). Springer, Berlin, Heidelberg.
- [17] Schilit, B., Adams, N., & Want, R. (1994, December). Context-aware computing applications. In *Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on* (pp. 85-90). IEEE.
- [18] Hanzal, T., Svátek, V., & Vacura, M. (2016, July). Event Categories on the Semantic Web and Their Relationship/Object Distinction. In *FOIS* (pp. 183-196).
- [19] Gómez-Pérez, A. Towards a framework to verify knowledge sharing technology. *Expert Systems with Applications*, 11(4), 519-529, (1996).