

Módulo de Identificación de Pasos y Situaciones

MIPS

Rubén Agudo Santos Mikel Villamañe Gironés

1 de julio de 2014

Fe de erratas

- ▶ La palabra “contínuo” varias veces.
- ▶ La palabra “que”. Página 43, párrafo 1, línea 1.

Índice

1. Antecedentes
2. Ejemplo
3. Desarrollo
4. Gestión
5. Conclusiones
6. Líneas futuras

Antecedentes

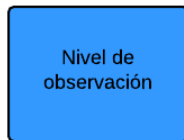
ULISES

- ▶ Enseñar a alumnos habilidades
- ▶ Unir un sistema interactivo a uno educativo
- ▶ En definitiva, evaluar

Antecedentes

Funcionamiento actual: Nivel de observación

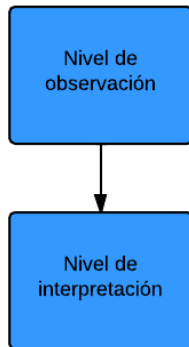
- ▶ Captura de datos
- ▶ Se crean
 - ▶ Propiedades
 - ▶ Observaciones



Antecedentes

Funcionamiento actual: Nivel de interpretación

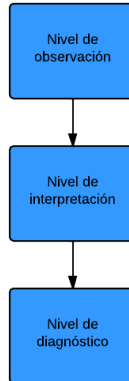
- ▶ Relaciones entre observaciones
 - ▶ Pasos
 - ▶ Situaciones



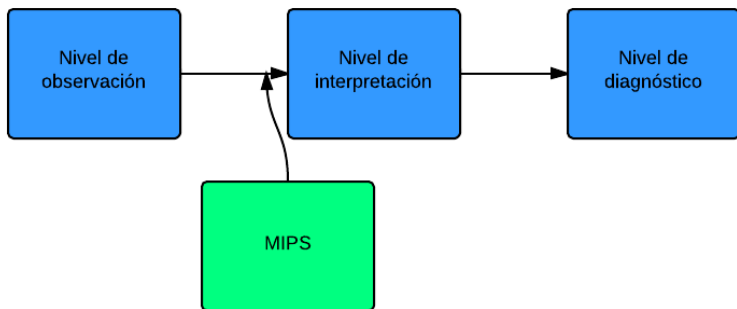
Antecedentes

Funcionamiento actual: Nivel de diagnóstico

- ▶ Usando distintos métodos de diagnóstico
 - ▶ *Clustering*
 - ▶ Clasificación supervisada
 - ▶ ...



Después del proyecto



Ejemplo

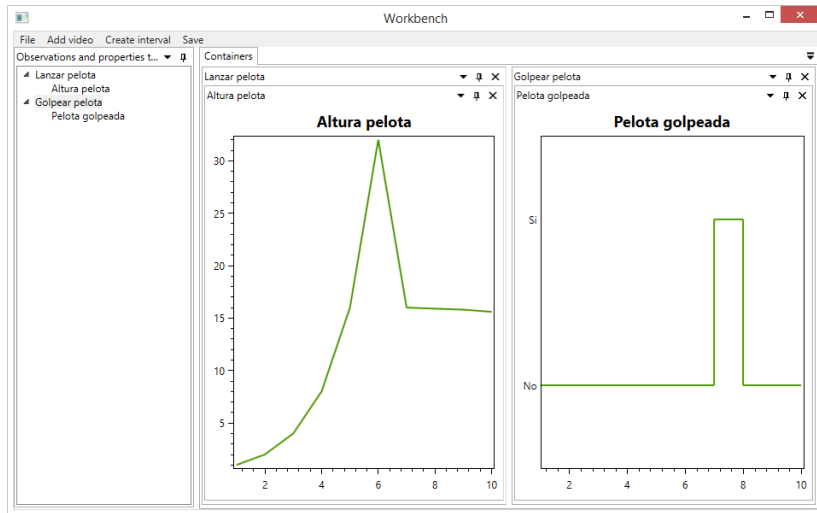
Ejemplo de observaciones y propiedades

```
1 <?xml version="1.0" encoding="utf-8" ?>
2 <data xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3     EspObservation="ObservationModelExample.xml"
4     instantLength="1">
5     <observation name="Lanzar pelota">
6         <property name="Altura pelota" type="1">
7             <instant ins="1" value="1"/>
8             <instant ins="2" value="2"/>
9             <instant ins="3" value="4"/>
10            <instant ins="4" value="8"/>
11            <instant ins="5" value="16"/>
12            <instant ins="6" value="32"/>
13            <instant ins="7" value="16"/>
14            <instant ins="8" value="15.9"/>
15            <instant ins="9" value="15.8"/>
16            <instant ins="10" value="15.6"/>
17        </property>
18    </observation>
```

```
19 <observation name="Golpear pelota">
20     <property name="Pelota golpeada" type="0">
21         <instant ins="1" value="No"/>
22         <instant ins="2" value="No"/>
23         <instant ins="3" value="No"/>
24         <instant ins="4" value="No"/>
25         <instant ins="5" value="No"/>
26         <instant ins="6" value="No"/>
27         <instant ins="7" value="Si"/>
28         <instant ins="8" value="No"/>
29         <instant ins="9" value="No"/>
30         <instant ins="10" value="No"/>
31     </property>
32 </observation>
33 </data>
```

Ejemplo

Ejemplo de visualización en MIPS



Diseño

SOLID y patrones

SOLID

- ▶ **S**ingle responsibility principle
- ▶ **O**pen-closed principle
- ▶ **L**iskov substitution principle
- ▶ **I**nterface segregation principle
- ▶ **D**ependency inversion principle

Patrones

- ▶ Model View View-Model
- ▶ Iterator
- ▶ Singleton

Desarrollo

Qué se ha hecho

Una aplicación que permite:

1. Cargar un XML con las observaciones y propiedades
2. Visualizar datos discretos y continuos
3. Cargar vídeos y visualizarlos
4. Seleccionar rangos
5. Exportar los rangos seleccionados en XML
6. Organizar los elementos como se desee, tipo Eclipse o Visual Studio

Desarrollo

Herramientas utilizadas

- ▶ Visual Studio 2013 Ultimate
- ▶ OxyPlot
- ▶ AvalonDock
- ▶ Git
- ▶ \LaTeX TeXstudio

Gestión

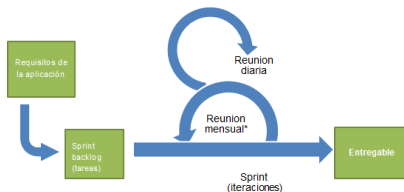
- ▶ Se han utilizado metodologías ágiles de desarrollo
 - ▶ Scrum
 - ▶ Kanban

Gestión

Scrum: ¿Qué es?

¿Qué es?

- ▶ Método de desarrollo iterativo e incremental
- ▶ En cada ciclo de desarrollo (sprint) se genera un entregable.
- ▶ Lo importante es que el diferencial de valor incremente



<https://upload.wikimedia.org/wikipedia/commons/e/e5/Scrumm.PNG>

Autor: Maxie Ayala Licenciado bajo CC BY-SA 3.0

Gestión

Kanban: ¿Qué es?

¿Qué es?

- ▶ Método de gestión del trabajo que se está realizando
- ▶ Típicamente se utiliza un tablero o corcho con 3 columnas
- ▶ Por columnas, mínimo se suele poner:
 1. Qué está pendiente por hacer
 2. Qué se está haciendo actualmente
 3. Qué se ha terminado

Gestión

Scrum: Cómo se ha usado

Ha habido 7 sprints, de un mes de duración

1. Se ha creado un listado de tareas (Scrum backlog)
2. Se han ordenado las tareas por prioridad
3. Por cada sprint
 - 3.1 Se seleccionan máximo 3 tareas por sprint (Sprint backlog)
 - 3.2 Se desarrollan las funcionalidades
 - 3.3 Se documentan
 - 3.4 Se despliega

Gestión

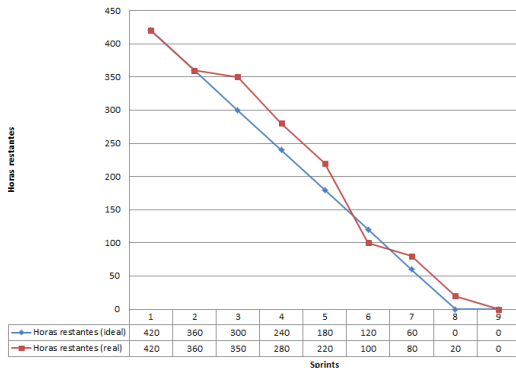
Kanban: Cómo se ha usado

- ▶ Columna “Por hacer”: Scrum backlog
- ▶ Columna “En progreso”: Sprint backlog
- ▶ Columna “Finalizado”: Tareas finalizadas organizadas por sprint

Conclusiones

Sobre la gestión

- Pese a las metodologías ágiles ha habido retraso



Conclusiones

Riesgos que se han cumplido

- ▶ Tener que cambiar de biblioteca
- ▶ Nuevos requisitos
 - ▶ Seleccionar el momento de inicio de sincronización del vídeo
 - ▶ En un mismo XML poder añadir más de un intervalo
 - ▶ Visualizar un árbol lateral con las propiedades y observaciones

Conclusiones

Personales

- ▶ Síndrome del programador
- ▶ Difícil programar sin documentación
- ▶ Estar fuera de la zona de “confort”

Líneas futuras

Mejoras

Ordenadas de más importante a menos importante

1. Que el software sea más abstracto
2. Utilizar Desarrollo Dirigido por Pruebas (Test Driven Development)
3. Mejorar el procesamiento paralelo.
4. Eliminar Singleton por patrones Factory
5. Utilizar los *data bindings* de MVVM

Refactorizar, refactorizar, refactorizar