E–ASSESSMENT TOOL WITHIN PBL IN PROGRAMMING SUBJECTS AT ENGINEERING

Departament of Telematics Engineering

University of Seville

Outline

- Introduction
- Tests
- e-Assessment Tool
- Conclusions
NoSQL BBDD in e-Assessment

Introduction
Tests
e-Assessment Tool
Conclusions
NoSQL BBDD in e-Assessment

Introduction

Tests

e-Assessment Tool

Conclusions

Network

BBDD

Server

Student1 Client

Student2 Client
NoSQL BBDD in e-Assessment

Introduction
Tests
e-Assessment Tool
Conclusions

Network

BBDD
Server

Student1 Client
Student2 Client
NoSQL BBDD in e-Assessment

Introduction

3 technologies in BBDD
- Relational BBDD
- NoSQL BBDD
- XML Natives BBDD

Tests

e-Assessment Tool

Conclusions

Data Base

Relacional  XML Native  NoSQL
# NoSQL BBDD in e-Assessment

<table>
<thead>
<tr>
<th>Introduction</th>
<th>NoSQL BBDD:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tests</td>
<td>- Free schema documents</td>
</tr>
<tr>
<td>e-Assessment Tool</td>
<td>- Low restrictions in managed data</td>
</tr>
<tr>
<td>Conclusions</td>
<td>- Easy solutions scaling</td>
</tr>
</tbody>
</table>

![Diagram showing flow of 'Less restrictions' leading to 'More Fast' and 'Less potential problems']
NoSQL BBDD in e-Assessment

Introduction

Tests

e-Assessment Tool

Conclusions

**MongoDB:**
- NoSQL BBDD multiplatform
- Document Oriented
- Each document has a free schema

![Diagram of NoSQL BBDD in e-Assessment]

- Server
- BBDD
- Collection
- Documents
- Fields
NoSQL BBDD in e-Assessment

**MongoDB:**
- Format Data BSON, (Binary Javascript Object Notation)
- For consistency, data replication
- Horizontal scaling

**Introduction**

**Tests**

**e-Assessment Tool**

**Conclusions**

**Clúster Configuration**

**DATA**
- Shard1
- Shard2
- Shard3
- Shard4

**Router**
- mongos
- mongos
NoSQL BBDD in e-Assessment

Tests:
- Programming language C#
- Drivers for each solution
NoSQL BBDD in e-Assessment

Tests:
- Cicles of 500, 2000 and 5000 repetitions
- Insertions
- Selections based on unique Identifier
- Updates

<table>
<thead>
<tr>
<th>Afiliado</th>
<th>Puesto</th>
</tr>
</thead>
<tbody>
<tr>
<td>_ID</td>
<td>AfiliadoID</td>
</tr>
<tr>
<td>Nombre</td>
<td>Centro</td>
</tr>
<tr>
<td>Apellido</td>
<td>Ciudad</td>
</tr>
<tr>
<td>Email</td>
<td>Cargo</td>
</tr>
<tr>
<td>Puesto</td>
<td>Estudios</td>
</tr>
</tbody>
</table>
### NoSQL BBDD in e-Assessment

**Tests:**

- Results cicles 2000 repetitions

<table>
<thead>
<tr>
<th>Tests</th>
<th>MYSQL</th>
<th>SQL Server</th>
<th>Mongo DB</th>
<th>Marklogic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insert</td>
<td>12659</td>
<td>5625</td>
<td>205</td>
<td>1247</td>
</tr>
<tr>
<td>Select</td>
<td>68952</td>
<td>12274</td>
<td>343</td>
<td>14805</td>
</tr>
<tr>
<td>Update</td>
<td>39693</td>
<td>12367</td>
<td>197</td>
<td>44361</td>
</tr>
</tbody>
</table>
NoSQL BBDD in e-Assessment

Tests:

- 2000 Insertions, (In milliseconds)

2000 Insertions

- MYSQL
- SQL Server
- MongoDB
- Marklogic
NoSQL BBDD in e-Assessment

Tests:
- 2000 selections, (in milliseconds)

2000 selections

<table>
<thead>
<tr>
<th>Database</th>
<th>Time (milliseconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MYSQL</td>
<td>80000</td>
</tr>
<tr>
<td>SQL Server</td>
<td>20000</td>
</tr>
<tr>
<td>MongoDB</td>
<td>0</td>
</tr>
<tr>
<td>Marklogic</td>
<td>10000</td>
</tr>
</tbody>
</table>
NoSQL BBDD in e-Assessment

**Tests:**

- 2000 updates, (in miliseconds)

### 2000 updates

<table>
<thead>
<tr>
<th>Database</th>
<th>Time (milliseconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MySQL</td>
<td>36000</td>
</tr>
<tr>
<td>SQL Server</td>
<td>10000</td>
</tr>
<tr>
<td>MongoDB</td>
<td>0</td>
</tr>
<tr>
<td>Marklogic</td>
<td>50000</td>
</tr>
</tbody>
</table>
**NoSQL BBDD in e-Assessment**

**e-Assessment Tool:**
- Written in C programming language
- Under Linux
- Designed for student’s assessment in Object-Oriented Programming (Degree on Telecommunication Technology Engineering)
- Two side
  - Client
  - Server
NoSQL BBDD in e-Assessment

Introductions
Tests
e-Assessment Tool
Conclusions
NoSQL BBDD in e-Assessment

Introductions

Tests

e-Assessment Tool

Conclusions

Network

MongoDB

Server

Student1 Client

Student2 Client
# NoSQL BBDD in e-Assessment

<table>
<thead>
<tr>
<th>Introductions</th>
<th>SERVER</th>
<th>CLIENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tests</td>
<td></td>
<td></td>
</tr>
<tr>
<td>e-Assessment Tool</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conclusions</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
NoSQL BBDD in e-Assessment

Introductions
Tests
e-Assessment Tool
Conclusions

SERVER

CLIENT

./clientepfc login_user
The following information reaches the server through a TCP socket:

**group name, file name**, followed by its contents.

The **group name** is used to connect to MongoDB and specific database for the group is created, so you can work on it.

With the **file name**, and its contents, a file is created on the server side and stored in MongoDB.
NoSQL BBDD in e-Assessment

Introductions

Tests

e-Assessment Tool

Conclusions

Options:
1. Compile.
2. Run.

SERVER

./clientepfc login_user

CLIENT
If option 1: a call for MongoDB is made, the appropriate files are taken and compiled.

¿OK?

Update student’s assessment in MongoDB.
Send OK to student.

¿Error?
Update student’s assessment in MongoDB.
Send OK to student.
If option 1: a call for MongoDB is made, the appropriate files are taken and compiled.

¿OK?

¿Error?

Error message is sent the student.
An error file is sent for review.
If option 2:
Run test.

¿OK?

¿Error?

Update student’s assessment in MongoDB.
Send OK to student.

2 arguments to send:
- Agendaxy
- Principal.java
NoSQL BBDD in e-Assessment

Introductions

Tests

e-Assessment Tool

Conclusions

¿OK?

Update student’s assessment in MongoDB.
Send OK to student.

¿Error?
NoSQL BBDD in e-Assessment

SERVER

CLIENT

2 arguments to send:
• Agendaxy
• Principal.java

If 2:
Run test.

¿OK?

¿Error?

Error message is sent the student.
An error file is sent for review.
NoSQL BBDD in e-Assessment

Introductions

Tests

e-Assessment Tool

Conclusions

MongoDB

Server

Client
## Conclusions

**Conclusions:**

- e-Assessment Tool.
- Programming subject.
- Client/Server.
- MongoDB as efficient, high-performance BBDD
## Future work:

- Implementation in Java programming language.
- Web interface.
- E-mail notification.
- Intelligent random tests.
Thank you.