

sistemas de informaçao ii

Relatório da 2ª Fase

LI51D  
Docente: Nuno Datia  
Semestre de Inverno 16/17  
Licenciatura em Engenharia Informática e de Computadores  
DEETC

**Grupo 27:**

Ana Carolina Baptista – 41487

Pedro Tavares – 41490

Gonçalo Santos- 40599

**Aplicação .Net**

A aplicação criada para esta fase do trabalho fornece acesso às funcionalidades de (e) a (k) da fase passada usando duas tecnologias diferentes (ADO.NET e EntityFrameworks), a aplicação também exporta uma representação XML dos alugueres entre duas datas para a pasta da solução (.sln). Para poder aceder a estas funcionalidades é necessária a criação da BD, em anexo estão todos os ficheiros necessários para a criação da base de dados, estes deveram ser executados pela seguinte ordem **Creates.sql -> CreateViews.sql -> CreateTriggers.sql -> (todos os ficheiros na pasta StoredProcedure) -> Inserts.sql .**

Ao iniciar a aplicação é perguntado qual a tecnologia a utilizar até ao final da execução do programa, não podendo ser mudada a meio. Após esta escolha terá acesso ás funcionalidades descritas que poderão ser executadas o numero de vezes desejadas.

Nesta fase foram alterados alguns procedimentos armazenados que já existiam na fase anterior e criados outros novos para auxiliar no acesso a dados em ambas as funcionalidades.

Foi adicionado um trigger ás tabelas Promoção e Aluguer para não deixar inserir se as datas de inicio forem anteriores ao dia corrente.

**Comparação entre a tecnologia EF e ADO.NET quanto à facilidade de programação e desempenho**

Para fazer as medições de tempos foi chamado um procedimento armazenado que lista todos os equipamentos que não tiveram alugueres na ultima semana utilizando as duas tecnologias, as chamadas ao procedimento são sequenciais para cada tecnologia (sem parar o programa) e os resultados obtidos estão presentes na Tabela 1.

Como é possível observar na Tabela 1 o ADO.NET é substancialmente mais rápida que o EF na primeira execução já nas seguintes execuções sequenciais são mais semelhantes, isto é esperado uma vez que o EF é montado em cima do ADO.NET logo nunca seria mais rápido. Quanto á facilidade de programação o EF fornece uma maneira mais intuitiva e fácil de programar abstraindo todo o código de acesso a dados, o modelo de dados no programa pode ser atualizado a partir do modelo presente na BD sempre que pedido portanto não haverá problemas de sincronização entre os modelos.

|  |  |
| --- | --- |
| Ado.Net | Entity Frameworks |
| 413 922 Ticks | 10 722 357 Ticks |
| 21 932 Ticks | 28 092 Ticks |
| 26 664 Ticks | 16 563 Ticks |
| 18 576 Ticks | 15 359 Ticks |
| 26 227 Ticks | 46 514 Ticks |
| Media: 101 464Ticks | Media:2 165 777 Ticks |

Tabela 1

**Garantia da consistência dos dados**

Uma vez que o utilizamos o ADO.NET de forma conectada a probabilidade de inconsistência é inferior pois não fazemos operações (selects, deletes, inserts, updates) sobre um clone da base de dados que é o que acontece no EF.