

Editora IDEIMusic

Relatório

08-12-2014

Ana Sofia Sá - 1100537
Francisco Santos - 1111315
Rui Silva - 1121296
3DC

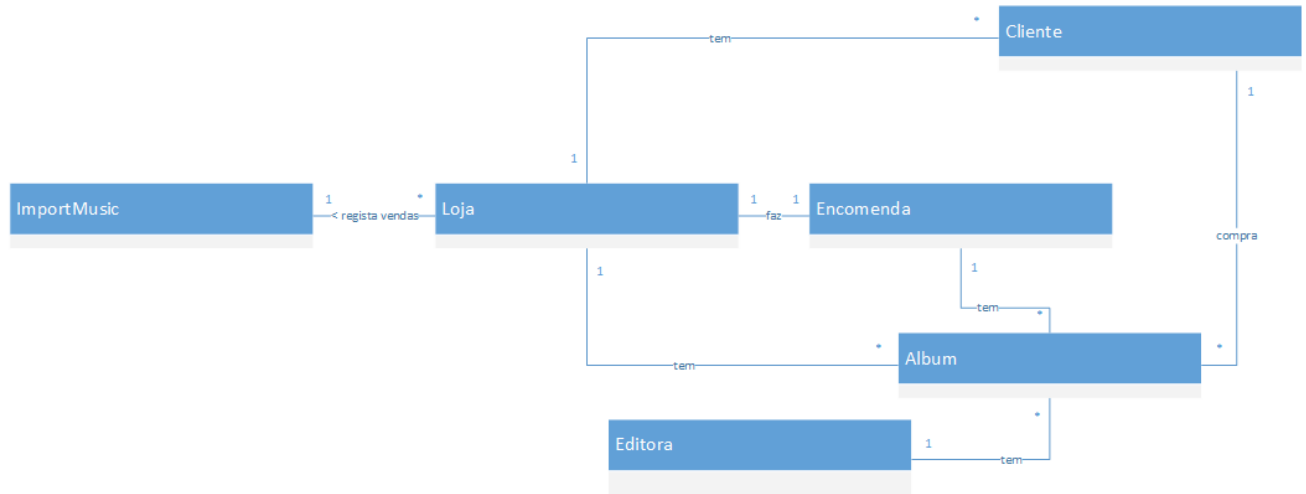
Introdução

Este projecto foi desenvolvido no contexto da disciplina do ISEP de Arquitectura de Sistemas (ARQSI). Consiste na criação de um "ecossistema" de comercio de discos e vinil de musicas. Para este fim implementamos 3 diferentes servidores independentes que comunicam entre si.

- 1 IDEIMusic: Desenvolvida em ASP.Net, é a editora de discos. Responsável pelo fornecimento dos mesmos às lojas de música no geral
- 2 MusicStore: Desenvolvida em PHP do lado do servidor e html, css e javascript do lado do cliente (utilizando comunicação mediada por ajax). Refere-se a uma loja de musica que recebe os discos da editora IDEIMusic para posteriormente os redistribuir para os clientes finais.
- 3 ImportMusic: Um servidor de análise de mercados. As compras registadas são enviadas para este, e posteriormente registadas numa base de dados

Modelo de Dados

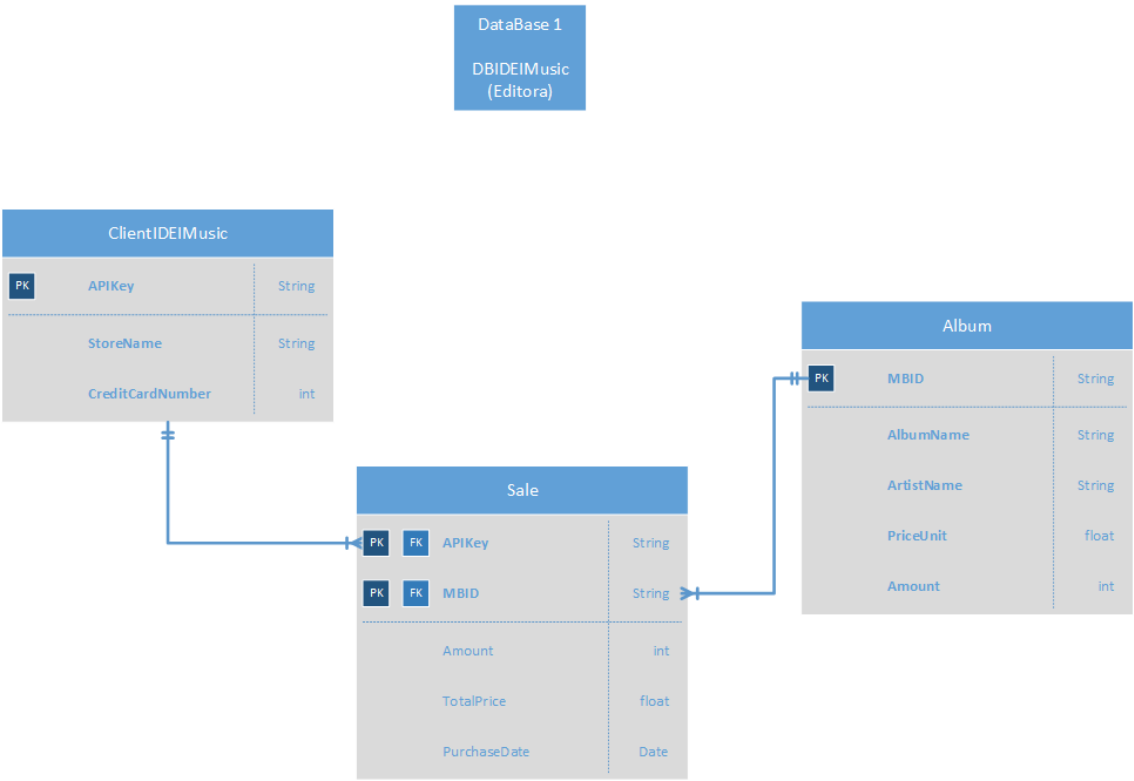
O modelo de dados representa a nossa ideologia da arquitectura e relacionamento dos diferentes conceitos dos varios sistemas.



Neste diagrama identificamos as entidades que fazem parte do projeto, bem como a interação entre si.

Modelo de Dados

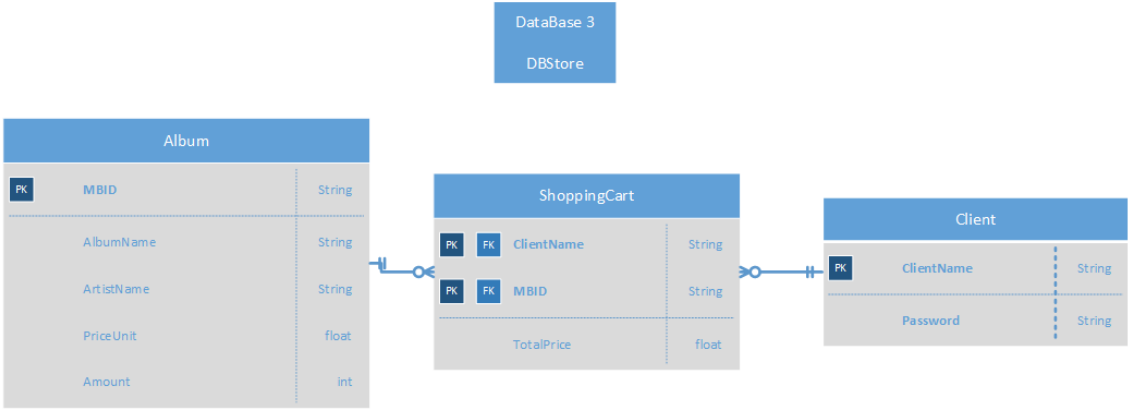
O Modelo de dados ilustra as tabelas e relações entre os diferentes dados guardados em cada um dos servidores. O servidor da editora e da loja de musica são muito semelhantes para permitirem uma mais facil comunicação entre os dois.



Modelo de Dados da IDEIMusic



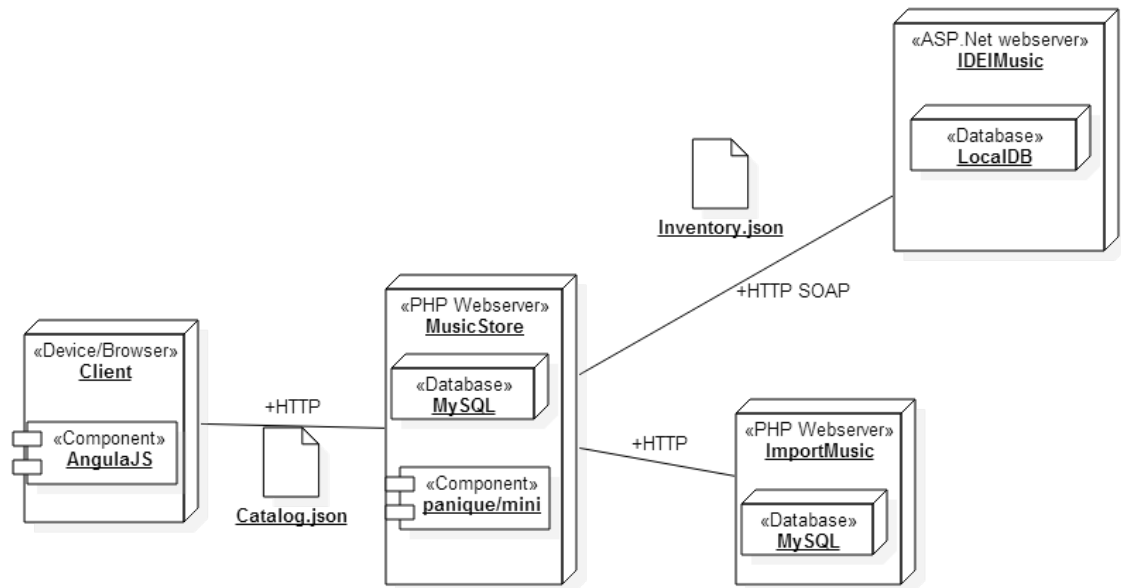
Modelo de Dados ImportMusic



Modelo da Dados MusicStore

Deployment View

A “deployment view” representa a arquitectura dos diferentes servidores e dos protocolos principais que usam para comunicação entre si. Principalmente foram realizadas comunicações pelos protocolos mais comuns de HTTP (GET e POST), sendo ocasionalmente usado o serviço web SOAP para comunicação que envolva um maior conjunto de dados.



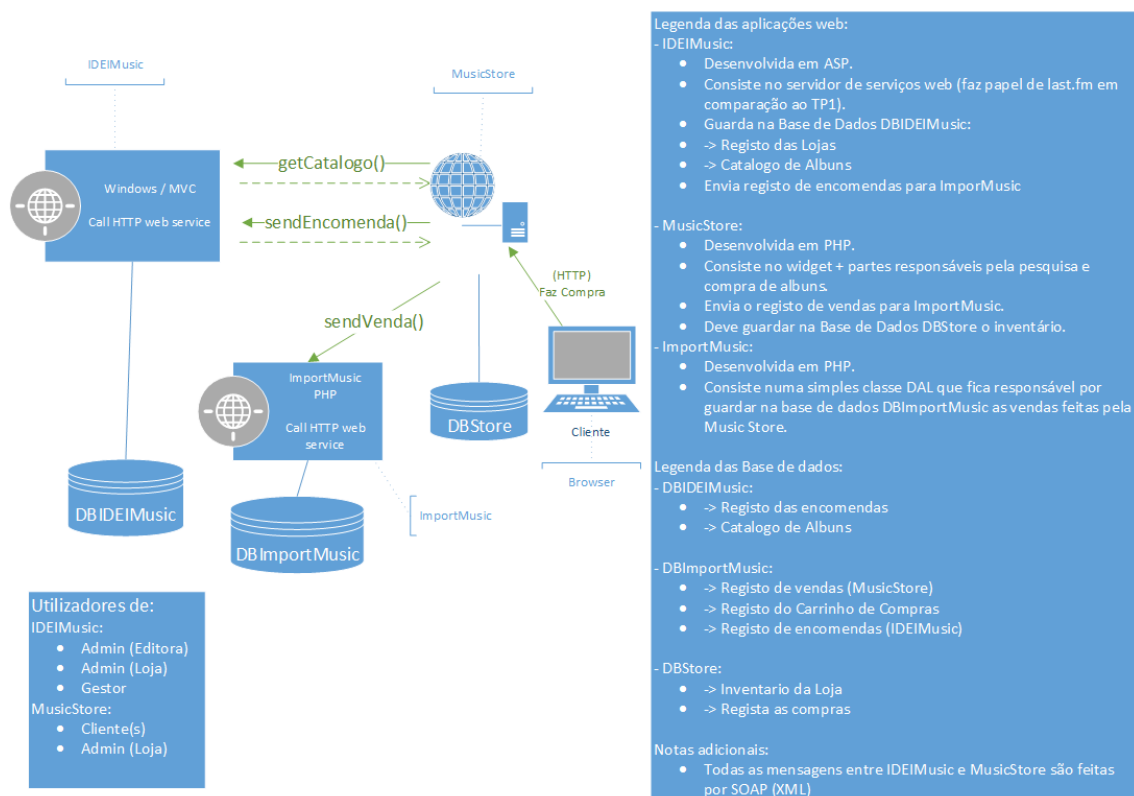
Este diagrama pretende ilustrar quais as máquinas que estão a ser usadas no projeto, a forma como cada nó comunica com outro, bem como quais os serviços instalados em cada um dos mesmos.

Logical View

A “logical View” envolve vários tipos de diagramas, de classe, comunicação e sequência. Foram usadas várias classes (em javascript e php) que procuraram seguir o modelo de dados definido anteriormente. No servidor da editora foi utilizado o método “code first” pelo que as classes desenvolvidas seguem rigorosamente o modelo de dados para uma operação própria da base de dados segundo o nosso modelo.

No caso das linguagens php e javascript, não sendo tão focadas na programação orientada a objectos foram adaptadas para alguns conceitos relacionados a objectos.

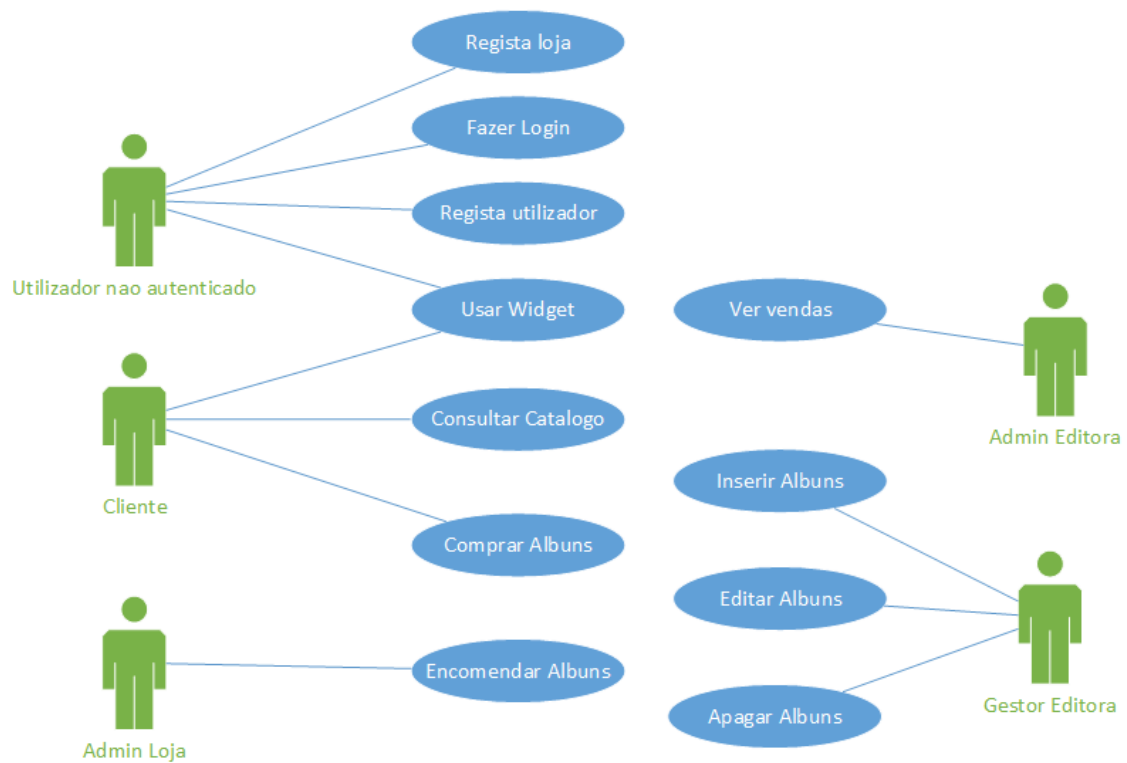
O seguinte diagram ilustra a nossa interpretação do enunciado procurando abarcar o máximo de informação num unico diagrama.



Neste diagrama pretende-se ilustrar a forma como cada empresa comunica com outras e o acesso a bases de dados.

Outros Diagramas

Diagrama de Casos de Uso



Representação das funcionalidades a implementar no projeto.

Diagramas de Sequência

Diagrama de Sequência: Registro da loja na Editora

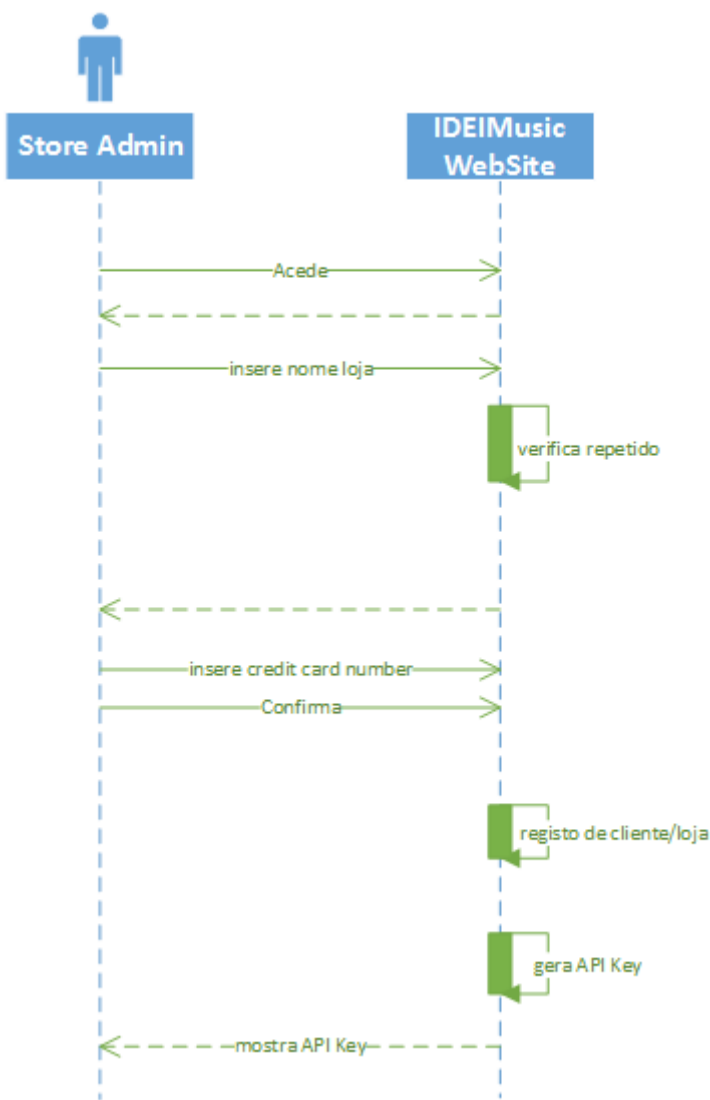
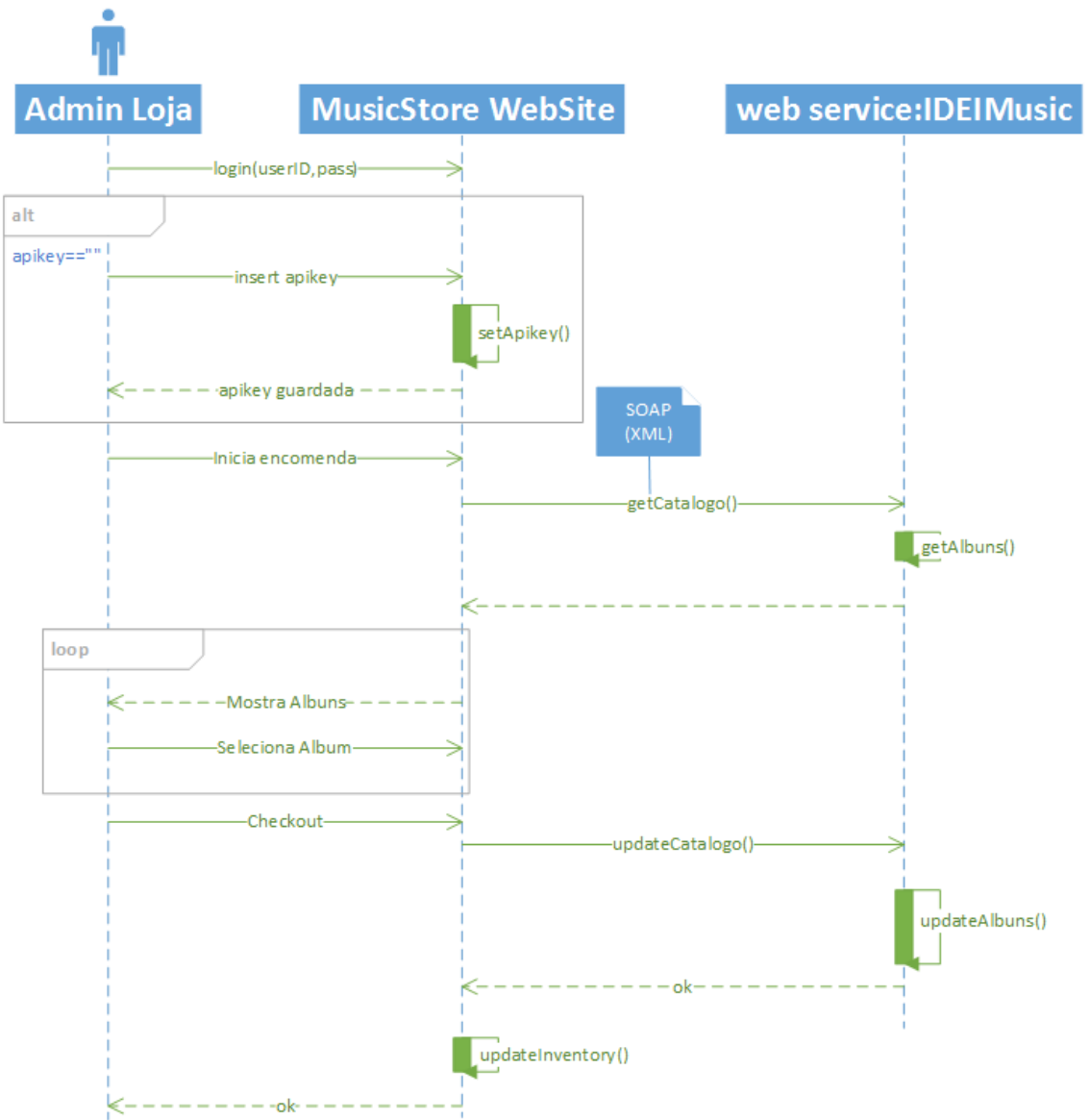


Diagrama de Sequência: Obter Catálogo e Enviar Encomenda à Editora



Comunicação IDEIMusic - MusicStore

A comunicação é realizada usando SOAP (XML). O gestor da MusicStore (para o caso deste exercício, no geral será aplicado para todos os clientes da editora) deve estar registado e ser detentor de uma chave API *Application Programming Interface*, caso não o seja, será fornecida uma assim que este efectue o primeiro pedido de discos após devidamente registado.

Uma encomenda da editora para a loja de música é iniciada pela loja com um pedido inicial HTTP `getAllAlbums()`.

A editora devolve uma lista em formato SOAP (XML) com todos os albuns disponíveis.

O gerente da MusicStore pode então preparar uma resposta com os discos que pretende encomendar (assim como a sua quantidade).

Em caso de sucesso, uma resposta da IDEIMusic confirma a encomenda. Caso contrário, são devolvidos os erros.