

PhononPlayer

Cover&Lyric Plugin

- Adyson Magalhães Maia

Plugin de visualização da capa e da letra da música que está sendo tocada no PhononPlayer

Ambiente de Desenvolvimento

- Linguagem C++
- Desktop - Linux
- IDE Qt Creator
- Qt 4.5 - Framework UI e Aplicação Multi-plataforma
 - QtGui – interface gráfica
 - Phonon – aplicações multimídias
 - QtNetwork – conexão
 - QtXml – leitura e escrita XML

Cover&Lyric Plugin

- PhononPlayer – tocador de música
- API Last.fm – buscar as capas dos álbuns (XML)
- API Leo Lyrics – buscar as letras das músicas (XML)
- Padrão Model View Controller (MVC)
- Padrão Observer
 - Implementada pelo QT por Signals e Slots

Signals e Slots

- Conceito criado pelo QT
- Implementação generalizada do padrão Observer
 - Encontrada no QT e na Boost
- O conceito é que os controles (widgets) podem enviar sinais contendo informações - Signal
- Estas informações podem ser recebidas por outros controles - Slot
- Utiliza-se chamadas assíncronas
- Ideal para ser usado em Graphical User Interfaces

Exemplo código

```
CoverLyric::CoverLyric(QWidget *parent) : QObject(parent)
{
    controller = new Controller();

    connect(this, SIGNAL(getCover(MediaObject*)), controller, SLOT(getCover(MediaObject*)) );
    connect(controller, SIGNAL(returnCover(QImage) ), this, SLOT(mountCover(QImage)) );

    connect(this, SIGNAL(getLyric(MediaObject*)), controller, SLOT(getLyric(MediaObject*)) );
    connect(controller, SIGNAL(returnLyric(QString) ), this, SLOT(mountLyric(QString)) );

    QTextCodec::setCodecForCStrings(QTextCodec::codecForName("UTF-8"));
    QTextCodec::setCodecForTr(QTextCodec::codecForName("UTF-8"));
    QTextCodec::setCodecForLocale(QTextCodec::codecForName("UTF-8"));
}

void CoverLyric::stateChanged(Phonon::MediaObject *moo)
{
    this->mtResolver = moo;
    emit getCover(this->mtResolver);
    emit getLyric(this->mtResolver);
}

void Controller::getLyric(MediaObject *moo)
{
```

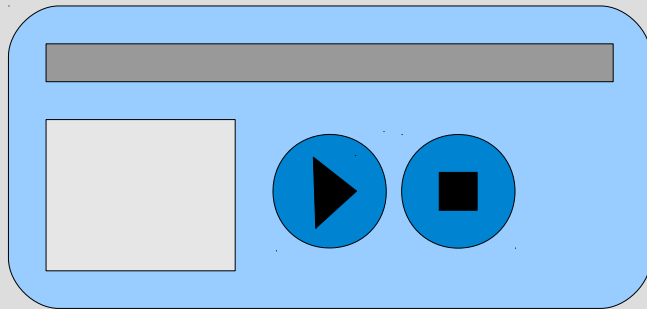
Conexão

Signal

Slot

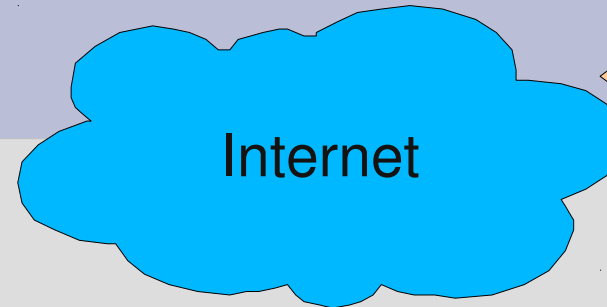
Arquitetura

Tocador



Chamada
Método

Interface



Plugin

Modelo



Fairy lady, who stands
on the walls

Life is short and wait is
long

The stars, away, dim
with the dawn...

Controle

Dispara
Sinais

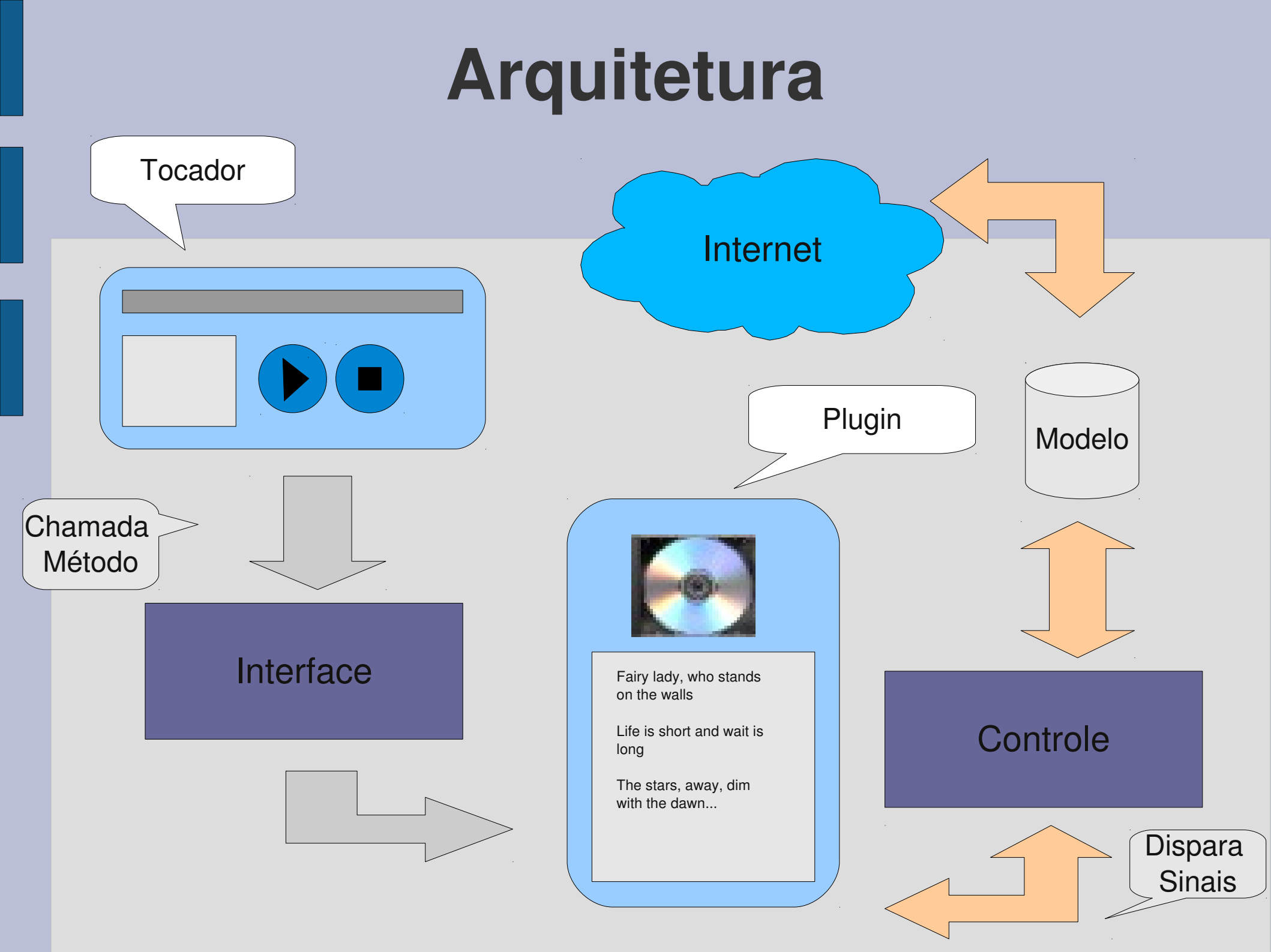
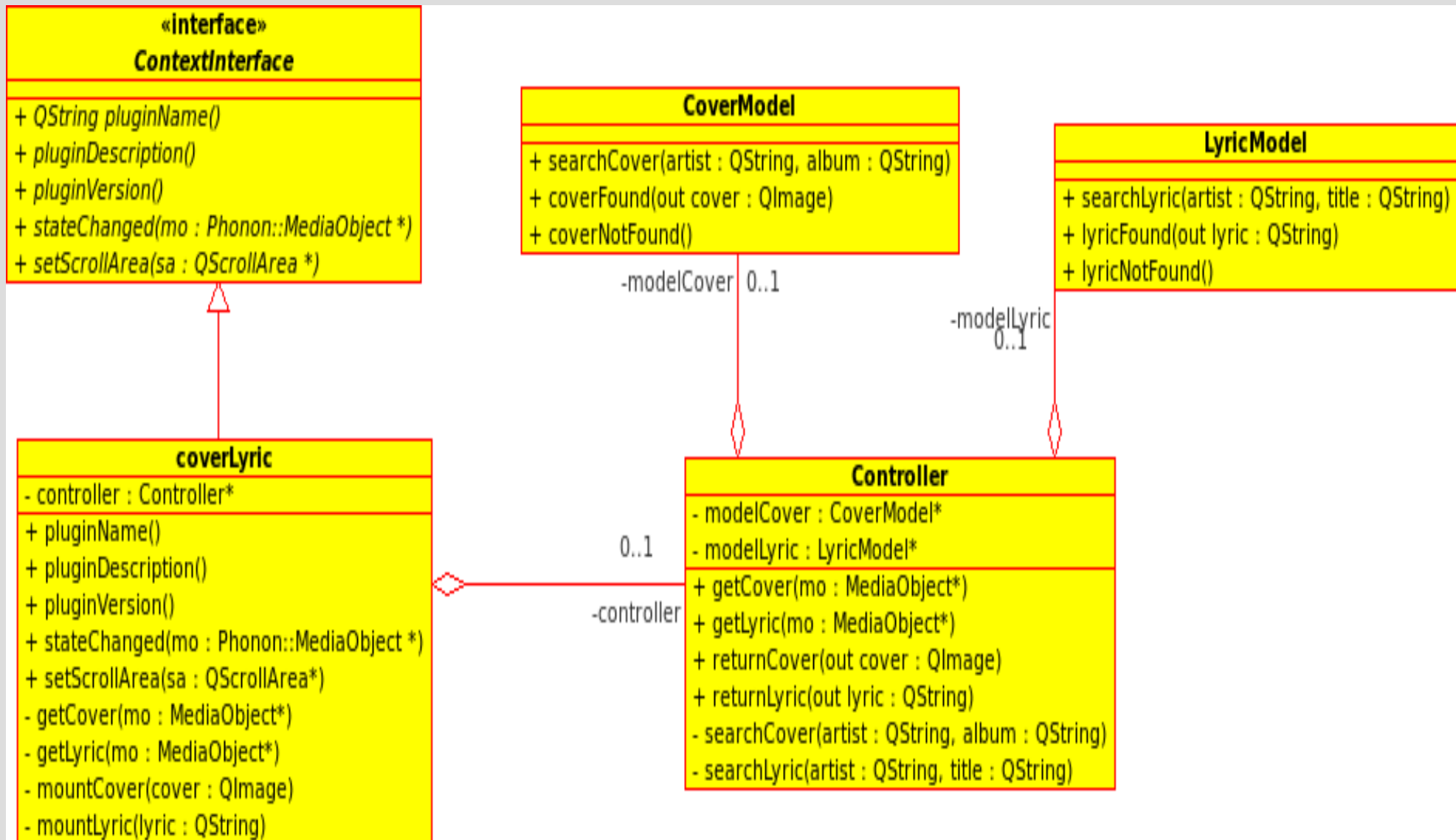
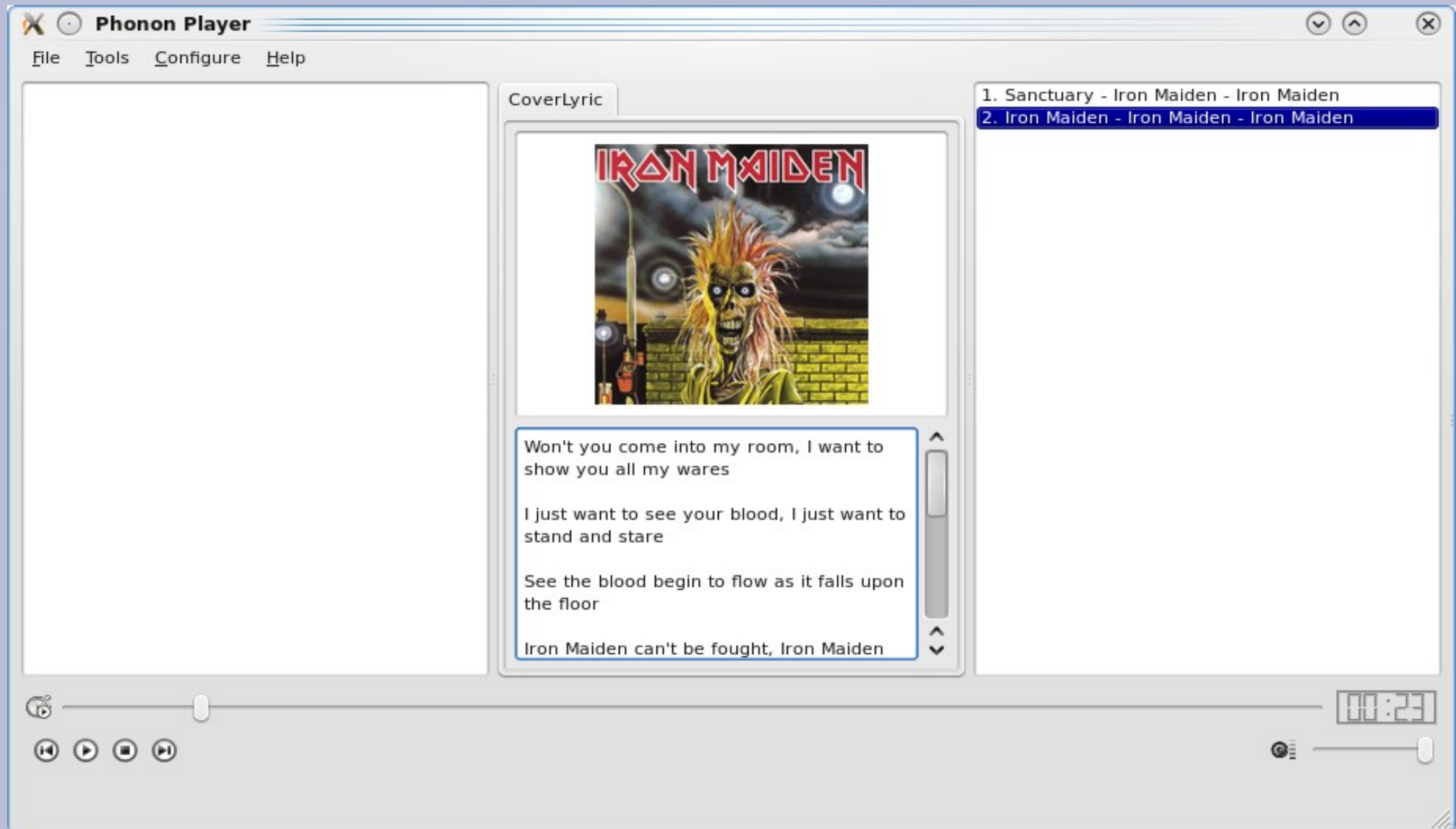


Diagrama de Classe



Screenshots



Trabalhos futuros

- Melhorar a busca pelos meta-dados das músicas
- Buscar em mais fontes de dados
 - internet e localmente
- Oferecer opção de escolha das letras e capas aos usuários
 - Busca retorne vários resultados
- Editar e criar as letras e disponibilizá-las na internet

Dificuldades

- Encontrar serviços na internet que forneçam as informações desejadas de uma maneira fácil de buscar
- Definir uma interface de comunicação entre as duas partes comunicantes
- Criar códigos fontes para ser uma biblioteca

Lições aprendidas

- A criação de aplicações com suporte a extensão por plugins ou desenvolvido em módulos
 - Facilita a evolução e manutenção
 - Mas pode ser uma difícil tarefa para desenvolvê-lo desta forma

Referências

- API Last.fm - <http://www.lastfm.com.br/api>
- API Leo Lyrics - <http://www.leoslyrics.com>
- QT - <http://qt.nokia.com/>
- Boost C++ libraries - <http://www.boost.org/>