Universidade Católica de Pernambuco

Centro de Ciências e Tecnologia

Ciência da Computação

Introdução à Programação II

Prof. Márcio Bueno

ip2noite@marciobueno.com

Projeto de Introdução à Programação II

Data Entrega: **04/11/2016**

Pontuação: **3,0 pontos** (a prova do 2º GQ vale **8,0 pontos**)

Qtd Participantes: 2a3 pessoas

Enviar e-mail: Deve obedecer ao seguinte padrão:

* Para: **ip2noite@marciobueno.com**
* Assunto: Projeto IP2
* Corpo: Incluir os nomes completos, matrícula, e-mail dos integrantes da equipe (omissões já indicarão pessoas que não fizeram o projeto)e anexe também este arquivo marcando todas as funcionalidades que foram desenvolvidas
* Anexo: Um único arquivo .ZIPcontendo uma pasta com todos os arquivos com os códigos fontes [.c] e dados [arquivos binários com informações de usuários, categorias, produtos e avaliações] (não enviar executáveis e outros arquivos desnecessários)

**Este será um projeto baseado no conceito de uma aplicação do tipo Spotify, Deezer e Apple Music onde usuários poderão criar playlists a partir de músicas existentes na base de dados e compartilhá-las, além de poder fazer busca por playlists para poder favoritá-las. Seguem detalhes das funcionalidades desejadas e no final tem todas as structs que vocês deverão utilizar no projeto.**

1. Você deve fazer DOIS programas, um que será utilizado pelo administrador e outro que será utilizado pelo usuário. No caso do programa que será utilizado pelo administrador será necessário utilizar a seguinte senha sem as aspas: “m@st3r2016” e no caso do programa a ser utilizado pelo usuário, no início deverá ter a opção de login ou cadastro de novo usuário.
2. O programa do Administrador terá as seguintes funcionalidades:

**X** Apresentar a tela de login onde deverá ser digitada a senha para ter acesso a todas as funcionalidades descritas a seguir (a senha não deverá ser visível – apresente \* cada vez que o usuário digitar algum caractere)

**X**  Listar usuários (exibir todos os campos com EXCEÇÃO da senha)

**X**  Consultar usuário por código, login ou parte do nome (exibir todos os campos com EXCEÇÃO da senha)

**X** Alterar a senha de um usuário (solicitar nova senha e confirmação) (buscar a partir do código)

**X** Cadastrar novas músicas

**X** Listar todas as músicas (exibir todos os campos)

**X** Consultar músicas por código, parte do título da música ou parte do nome do artista (exibir todos os campos)

**X** Alterar dados de músicas cadastradas (busca a partir do código da música que NUNCA pode ser alterado)

**X** Excluir músicas cadastradas (busca a partir do código da música – lembre-se que neste caso, esta música também deve ser excluída de TODAS as playlists que já existem)

**X** Listar todas as playlists (para cada playlist exibir o código, título e nome do criador da playlist como cabeçalho e depois exiba todas as músicas desta playlist)

**X** Consultar playlist por código da playlist, parte do título da playlist, por código de música (listar todas as playlists que tenham esta música) (para cada playlist exibir o código, título e nome do criador da playlist como cabeçalho e depois exiba todas as músicas desta playlist)

1. O programa que o usuário vai utilizar deverá ter as seguintes funcionalidades:

**X** Apresentar a tela de login onde deverá ser digitado login e a senha para ter acesso a todas as funcionalidades descritas a seguir (a senha não deverá ser visível – apresente \* cada vez que o usuário digitar algum caractere), no caso do usuário não ser cadastrado, deverá ser permitida a criação deste usuário (onde o usuário informará seu nome, login, senha e confirmação de senha)

**X** Listar seus dados (todos os campos com EXCEÇÃO da senha)

**X** Alterar o nome do usuário (mediante confirmação de senha)

**X** Alterar o login do usuário (mediante confirmação de senha)

**X** Alterar a senha do usuário (solicitar nova senha e confirmação) (mediante confirmação da senha atual)

**X** Listar usuários (exibir apenas o nome usuário)

**X** Consultar usuários por parte do nome (exibir apenas o nome dos usuário)

**X** Listar todas as músicas (exibir todos os campos)

**X** Consultar músicas por código, parte do título da música ou parte do nome do artista (exibir todos os campos)

**X** Listar todas as suas playlists (exibir as playlists criadas e favoritadas) (para cada playlist exibir o código, título e nome do criador da playlist [apenas se for playlist favoritada] como cabeçalho e depois exiba todas as músicas desta playlist)

**X** Listar todas as playlists (com exceção das suas playlists: criadas e favoritadas) (para cada playlist exibir o código, título e nome do criador da playlist como cabeçalho e depois exiba todas as músicas desta playlist)

**X** Consultar playlist por código da playlist, parte do título da playlist, por código de música (listar todas as playlists que tenham esta música – primeiro exibir as playlists deste usuário, depois as favoritadas e finalmente as outras playlists) (para cada playlist exibir o código, título e nome do criador da playlist [apenas se não for deste usuário] como cabeçalho e depois exiba todas as músicas desta playlist)

**X** Cadastrar uma playlist (solicitar o título da playlist) e já deverá ser executada a funcionalidade de inserir músicas na playlist (sem a necessidade de digitação do código da playlist) [observe o limite máxima de playlists que podem ser criadas por usuário]

**X** Inserir música numa playlist sua (deverá ser solicitado o código da playlist e depois ficará em um laço solicitando as músicas que deverão ser incluídas na playlist [observe o limite máximo de músicas dentro de uma playlist]

**X** Excluir música de uma playlist sua (deverá ser solicitado o código da playlist, depois serão exibidos os códigos, títulos e artistas de todas as músicas desta playlist, e então ficará dentro de num laço solicitando os códigos das músicas que deverão ser excluídos)

**X** Excluir uma playlist (deverá ser solitado o código, depois deverão ser exibidos o código, o título e todas as músicas desta playlist, e no final, deverá ser solicitada uma confirmação se realmente deseja ou não excluir esta playlist)

**X** Favoritar uma playlist (a partir do código desta playlist – observe que o usuário não pode favoritar uma playlist sua)

**OBS1: Autoincremento** é um campo que deverá ser gerado automaticamente pelo seu programa e que deverá sempre ser UM a mais que o código do último registro cadastrado. O primeiro código gerado deverá ser o número 1.

**OBS2:** Os seguintes typdefs e structs DEVERÃO ser utilizadas em seu projeto e NÃO poderão conter NENHUM campo extra.

#define TAM\_NOME 50

#define TAM\_LOGIN 15

#define TAM\_SENHA 15

#define TAM\_PLYLST 200

#define QTD\_MUSICAS 100

typedef struct Usuario {

int codigo; // autoincremento

char nome[TAM\_NOME];

char login[15];

char senha[15];

int qtdPlaylists; // quantidade de playlists criadas pelo usuário

int playlists[TAM\_PLYLST]; // vetor com os códigos das playlists criadas

int qtdPlaylistsFav; // IDEM: favoritadas pelo usuário

int playlistsFav[TAM\_PLYLST]; // IDEM: favoritadas pelo usuário

} TUsuario;

typedef struct Musica {

int codigo; // autoincremento

char titulo[TAM\_NOME]; // título da música

char artista[TAM\_NOME]; // nome do artista que está cantando esta música

} TMusica;

typedef playlist {

int codigo; // autoincremento

int codigoUsuario; // o código do usuário que criou a playlist

char titulo[TAM\_NOME]; // título desta playlist

int qtdMusicas; // quantidade atual de músicas incluídas nesta playlist

int musicas[QTD\_MUSICAS]; // vetor contendo os códigos das músicas

} TPlaylist;