



navegação

[Página Inicial](#)

[Equipe](#)

[Atividades](#)

[Localização](#)

[Contatos](#)

[Área de Membros](#)

[Pastas das Disciplinas](#)

[EEL170 - Computação I](#)

[EEL270 - Computação II](#)

[EEL670 - Linguagens de Programação](#)

[EEL875 - Internet e Arquitetura TCP/IP](#)

[EEL878 - Redes de Computadores I](#)

[EEL879 - Redes de Computadores II](#)

[EEL480 - Sistemas Digitais](#)

[Tutoriais](#)

[Webmail](#)

[Novidades](#)

[Eventos](#)

acessar

Nome do Usuário

Senha

acessar

Esqueceu sua senha?

Arquivos de Dados

Contém a definição detalhada de cada arquivo de dados utilizado neste sistema.

Os dados necessários para o funcionamento do sistema de gerenciamento de caronas serão armazenados nos seguintes arquivos:

■ Arquivo de usuários ("*users*" - sem extensão)

Este arquivo (tipo texto) deverá conter uma linha por usuário cadastrado no sistema. Cada linha deverá incluir os campos: identificador numérico, apelido, senha codificada, perfil, nome completo e endereço eletrônico do usuário.

O campo identificador numérico é do tipo *mdlUmlUserIdentifierType* (que é igual a *unsigned long long*). Como o arquivo é do tipo texto, o identificador numérico deverá ser gravado como uma *string* com o comprimento mínimo possível.

O apelido (*nickname*) deverá ser gerado automaticamente a partir do nome completo do usuário, e deverá ser composto pelo primeiro nome do usuário seguido por um caractere ponto e se possível (se não houver duplicidade) pelo último sobrenome do usuário em questão. Se já existir um outro usuário com o mesmo apelido, deverá ser utilizado o penúltimo sobrenome.

A senha é do tipo *string* e deverá ser codificada utilizando-se o algoritmo *SHA512* com *salt* aleatório composto por 16 caracteres. Para os usuários que estiverem no estado pendente (convite não aceito/rejeitado ou solicitação não aprovada/reprovada) ou no estado bloqueado, a senha codificada deverá ter comprimento zero.

O campo perfil deverá ser representado neste arquivo por um dígito de acordo com o tipo *mdlUmlUserProfileType* (1 - administrador, 2 - , ...).

O nome completo e o endereço eletrônico são do tipo *string*.

Os campos de cada usuário deverão ser separados pelo caractere dois pontos (":"). Como este arquivo é do tipo texto, o caractere de final de *string* não poderá ser gravado neste arquivo.

Este arquivo deverá estar sempre ordenado (ordem crescente) utilizando o campo identificador numérico do usuário como critério para a ordenação e não poderá conter linhas em branco (nem mesmo no final do arquivo). Os campos identificador numérico, apelido e endereço eletrônico deverão ser únicos por usuário.

Os campos armazenados neste arquivo serão definidos mais tarde neste roteiro (ver [Definição dos Campos](#)).

■ Arquivo de pendências de convites de usuários ("*invited.users*")

Este arquivo (tipo binário) deverá conter três informações por usuário que tenha sido convidado para se cadastrar no sistema, mas que ainda não tenha confirmado, nem rejeitado o convite. As informações armazenadas neste arquivo incluem, por usuário, a validade absoluta deste convite (tipo *time_t*), o identificador numérico do usuário (tipo *mdlUmlUserIdentifierType*) e a sua senha temporária (codificada utilizando o algoritmo SHA512 com *salt* aleatório composto por 16 caracteres - tipo *string*).

Um convite terá validade de 72 horas. Se um usuário não confirmar, nem rejeitar o convite, os dados correspondentes a este usuário (armazenados neste arquivo e no arquivo *users*) deverão ser excluídos pelo programa de limpeza, que deverá ser executado automaticamente, através do agendador de tarefas (*cron*) do *FreeBSD/Linux*, uma vez por dia às 5 horas da manhã, no horário do servidor. O programa de limpeza deverá ser executado pelo *cron* na máquina "*www02.del.ufrj.br*".

Este arquivo deverá estar sempre ordenado (em ordem crescente), utilizando o campo validade como critério para a ordenação.

■ Arquivo de pendências de solicitações de cadastro de usuários ("*requesting.users*")

Este arquivo (tipo binário) deverá conter quatro informações por usuário que tenha solicitado cadastro no sistema e ainda não teve o seu pedido analisado (aprovado ou reprovado). As informações armazenadas neste arquivo incluem, por usuário, a validade absoluta da solicitação (tipo *time_t*), o identificador numérico do usuário (tipo *mdlUmlUserIdentifierType*) que deverá analisar a solicitação, o identificador numérico e a senha criptografada do solicitante.

Um convite terá validade de 7 dias. Se a solicitação for aprovada, a senha (criptografada) deste usuário será incluída no arquivo "*users*" e uma mensagem de boas vindas será enviada para o endereço eletrônico do mesmo. Por outro lado, se a solicitação for reprovada, os dados referentes a este usuário serão apagados (tanto no arquivo "*users*", quanto no arquivo "*requesting.users*"). Se o usuário responsável por analisar a solicitação não aprovar, nem reprovar o pedido dentro do prazo (7 dias), os dados correspondentes a esta solicitação deverão ser excluídos pelo programa de limpeza, que deverá ser executado automaticamente, através do agendador de tarefas (*cron*) do *FreeBSD/Linux*, uma vez por dia às 5 horas da manhã, no horário do servidor. O programa de limpeza deverá ser executado pelo *cron* na máquina "*www02.del.ufrj.br*".

Este arquivo deverá estar sempre ordenado (em ordem crescente), utilizando o campo validade como critério para a ordenação.

■ Arquivo contendo as contas bloqueadas ("*locked.users*")

Este arquivo (tipo binário) deverá conter quatro informações por usuário cuja conta esteja bloqueada, ou seja, o identificador numérico e a senha do usuário em questão, além do instante em que o bloqueio foi realizado e do identificador numérico do usuário que efetuou o bloqueio.

Este arquivo deverá estar sempre ordenado (em ordem crescente), utilizando o campo identificador do usuário como critério para a ordenação.

Para cada usuário do sistema (com exceção daquele identificado pelo número ZERO) este arquivo poderá conter zero ou uma referência de bloqueio. O usuário que representa o administrador principal do sistema nunca poderá ser bloqueado.

■ Arquivo contendo as solicitações de desbloqueio ("*unlocking.users*")

Este arquivo (tipo binário) deverá conter duas informações por usuário que esteja solicitando o desbloqueio, ou seja, a validade do pedido e identificador numérico do usuário em questão.

Um pedido de desbloqueio deve ser atendido em no máximo 24 horas. Os dados correspondentes a solicitações não atendidas dentro do prazo deverão ser excluídos pelo programa de limpeza, que deverá ser executado automaticamente, através do agendador de tarefas, uma vez por dia às 5 horas da manhã, no horário do servidor.

Este arquivo deverá estar sempre ordenado (em ordem crescente), utilizando o campo validade do pedido de desbloqueio como critério para a ordenação.

■ Arquivo de pendências de trocas de senhas ("*password.abeyances*")

Este arquivo (tipo binário) deverá conter três informações por usuário que tenha solicitado a reinicialização da senha e que ainda não tenha nem confirmado, nem rejeitado esta operação. As informações armazenadas neste arquivo incluem, por usuário, a validade absoluta da operação de reinicialização de senha (tipo *time_t*), o identificador numérico do usuário (tipo *xyzUserIdentifierType*) e a sua senha temporária (codificada utilizando o algoritmo SHA512 com *salt* aleatório composto por 16 caracteres - tipo *string*).

Uma senha temporária terá validade de 48 horas. Se um usuário não confirmar, nem rejeitar a troca da senha neste período, os dados correspondentes a este usuário (armazenados neste arquivo) deverão ser excluídos pelo programa de limpeza, que deverá ser executado automaticamente, através do agendador de tarefas, uma vez por dia às 5 horas da manhã, no horário do servidor.

Este arquivo deverá estar sempre ordenado (em ordem crescente), utilizando o campo validade como critério para a ordenação.

■ Arquivo de pendências de trocas de endereços eletrônicos (*email.abeyances*)

Este arquivo (tipo binário) deverá conter quatro informações por usuário que tenha solicitado a troca de endereço eletrônico e que ainda não tenha confirmado (validado) esta operação. As informações armazenadas neste arquivo incluem, por usuário, a validade absoluta desta operação (tipo *time_t*), o identificador numérico do usuário (tipo *xyzUserIdentifierType*), o novo endereço eletrônico (tipo *string*) e a chave de verificação.

A chave de verificação deverá ser uma string com comprimento igual a 64 caracteres e gerada de forma aleatória a partir do conjunto Base64.

Um novo endereço eletrônico poderá ser validado em até 48 horas. Se um usuário não confirmar o novo endereço eletrônico, os dados correspondentes a este usuário (armazenados neste arquivo) deverão ser excluídos pelo programa de limpeza, que deverá ser executado automaticamente, através do agendador de tarefas, uma vez por dia às 5 horas da manhã, no horário do servidor.

Este arquivo deverá estar sempre ordenado (em ordem crescente), utilizando o campo validade como critério para a ordenação.

■ Arquivos de Cookies

Para cada usuário que acesse o sistema via *web* deverá ser criado um arquivo binário contendo os dados relacionados com o *cookie* enviado para o navegador do usuário. Estes dados incluem a validade absoluta (tipo *time_t*) e o valor (tipo *string*) do *cookie*, além do endereço IP da máquina (tipo *string*) a partir da qual o sistema foi acessado.

Este arquivo deverá ser excluído quando o usuário executar a CGI "*mdlUmlLogout.cgi*". Caso isso não aconteça, o arquivo deverá ser excluído pelo programa de limpeza de pendências, como já foi descrito acima.

O nome do arquivo de *cookie* será igual ao identificador numérico do usuário, escrito sempre com 20 dígitos.

◀ Novembro 2019 ▶						
Do	Se	Te	Qu	Qu	Se	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

notícias

Horário 2017/2

31/07/2017

Estágio Supervisionado - Importante

05/03/2012

Mais notícias...