

MAA

Módulo de Apresentação de Autenticação: unidade responsável pela execução dos métodos de criação das telas para interação com usuário no quesito autenticação. Tal módulo provê a implementação da interface IAA, descrita abaixo:

IAA

void Executar() : Método responsável pela criação da tela de autenticação, em que o usuário poderá fazer login em sua conta. Não há parâmetros a serem a passados e nem retorno.

MSA

Módulo de Serviço de Autenticação: unidade responsável pela execução dos métodos responsáveis pela autenticação do usuário no sistema. Tal módulo provê a implementação da interface ISA, descrita abaixo:

ISA

bool Autenticar(Email, Senha) : Método responsável pela autenticação de um usuário no sistema, a fim de possibilitá-lo realizar certas ações impossíveis caso o mesmo não seja autenticado. São passados email e senha como parâmetros, de modo a comparar os dados passados com dados existentes no banco de dados no sistema. Caso o email passado não coincida com nenhum do banco de dados, o usuário não possui cadastro no sistema. Caso o email coincida e a senha não, a autenticação falha e o usuário não consegue realizar o login. Retorno booleano, em que 1 representa caso de sucesso, e 0 representa caso de falha.

MAU

Módulo de Apresentação de Usuário: unidade responsável pela execução dos métodos de criação das telas para interação com usuário como por exemplo ver o perfil, descadastrar usuário, entre outros. Tal módulo provê a implementação da interface IAU, descrita abaixo:

IAU

void Executar() : Método responsável pela criação da tela do menu principal do sistema, em que o usuário poderá realizar ações como cadastrar e/ou descadastrar caronas e reservas. Não há parâmetros a serem a passados e nem retorno.

MSU

Módulo de Serviço de Usuário: unidade responsável pela execução dos métodos responsáveis pela escolha do usuário no sistema, como o cadastramento de uma conta e/ou o descadastramento da mesma. Tal módulo provê a implementação da interface ISU, descrita abaixo:

ISU

bool Cadastrar(Usuário, Conta) : Método responsável pelo cadastramento de um novo usuário no sistema a fim de possibilitá-lo a realizar ações apenas possíveis quando logado no sistema. Para a criação do cadastro, é necessária a passagem dos parâmetros Usuário(entidade composta pelos domínios Nome, Telefone, Email, Senha e CPF) e Conta (entidade composta por código de banco, número de agência e número de conta) a fim de se identificar uma ou duas contas correntes para serem efetuados pagamentos). Retorno booleano, em que 1 representa caso de sucesso, e 0 representa caso de falha.

bool Descadastrar(Email) : Método responsável pelo descadastramento de um usuário do sistema, onde todos seus dados serão devidamente excluídos das dependências do sistema. É passado somente um Email como parâmetro ao invés de um Usuário pois como só se pode se descadastrar caso esteja logado no sistema, somente é necessária a passagem de um Email para se identificar o usuário e assim excluir todos os dados correspondentes a esse mesmo usuário. Retorno booleano, em que 1 representa caso de sucesso, e 0 representa caso de falha.

MAC

Módulo de Apresentação de Carona: unidade responsável pela execução dos métodos de criação das telas para interação com usuário como por exemplo escolher se o usuário quer cadastrar ou descadastrar uma carona. Tal módulo provê a implementação da interface IAC, descrita abaixo:

IAC

void Executar() : Método responsável pela criação da tela do menu de caronas do sistema, em que o usuário visualizará as telas de se cadastrar, descadastrar uma carona ou então pesquisar caronas disponíveis. Não há parâmetros a serem passados e nem retorno.

MSC

Módulo de Serviço de Carona: unidade responsável pela execução dos métodos de cadastramento, descadastramento e pesquisa de caronas no sistema. Tal módulo provê a implementação da interface ISC, descrita abaixo:

ISC

bool CadastrarCarona(Carona) : Método responsável pelo cadastro de uma carona por parte de um usuário logado no sistema para que outros possam reservá-la. Deve ser passado uma entidade Carona (composta por Código de Carona, Cidade e Estado de origem, Cidade e Estado de destino, Data, Duração, Número de Vagas e Preço) como parâmetro a fim de se cadastrar no sistema todos os dados referentes a uma nova carona disponível. Retorno booleano, em que 1 representa caso de sucesso, e 0 representa caso de falha.

bool DescadastrarCarona(CodigoDeCarona) : método responsável pelo descadastramento de uma carona no sistema. É passado como parâmetro um Código De Carona a fim de se identificar, por meio desse, o restante dos dados vinculados a tal carona e assim excluí-los permanentemente do sistema. Um usuário só pode descadastrar uma carona se esta pertencer a ele e este estiver logado no sistema. Uma carona só pode ser descadastrada

caso não haja reservas associadas à mesma no sistema. Retorno booleano, em que 1 representa caso de sucesso, e 0 representa caso de falha.

void PesquisarCarona(Cidade, Estado, Cidade, Estado, Data) : método responsável por listar a um usuário todas as caronas disponíveis em que a Cidade e o Estado, tanto de origem quanto de destino, e a Data coincidam com aquelas informadas pelo usuário ao sistema. É passado como parâmetro Cidade e Estado de Origem, Cidade e Estado de Destino, e Data da carona, respectivamente, de modo a informar ao sistema quais são os filtros a serem aplicados em sua pesquisa por todo o sistema para informar ao usuário. Este serviço pode ser realizado tanto por um usuário logado como por um não logado. Não há retorno pois sua responsabilidade é de somente informar ao usuário as caronas disponíveis com dados compatíveis àqueles de sua pesquisa.

MAR

Módulo de Apresentação de Reserva: unidade responsável pela execução dos métodos de criação das telas para interação com usuário como por exemplo escolher se o usuário quer efetuar uma reserva ou cancelá-la, e até mesmo listar as reservas associadas a sua conta. Tal módulo provê a implementação da interface IAR, descrita abaixo:

IAR

void Executar() : Método responsável pela criação da tela do menu de reservas do sistema, em que o usuário visualizará as telas de se realizar uma reserva e de se cancelar uma reserva, ou então de se listar as reservas de caronas associadas a sua conta no sistema. Não há parâmetros a serem passados e nem retorno.

MSR

Módulo de Serviço de Reserva: unidade responsável pela execução dos métodos de efetuar reserva, cancelar reserva e listar as reservas associadas a conta do usuário no sistema. Tal módulo provê a implementação da interface ISR, descrita abaixo:

ISR

bool Reservar(Reserva) : Método responsável pela realização de uma reserva associada a uma carona existente no sistema por parte de um usuário. Deve ser passado uma Reserva(entidade composta por código de reserva, assento e número de bagagens) de modo a informar ao sistema sobre os dados referentes a uma reserva associada a uma carona por parte de um usuário logado no sistema. O usuário não pode realizar reservas com conflitos de data, considerando-se data de partida e duração da carona. Após a confirmação da solicitação de reserva, o sistema informa ao usuário os dados da conta corrente de um usuário para pagamento da carona. Retorno booleano, em que 1 representa caso de sucesso, e 0 representa caso de falha.

bool CancelarReserva(Reserva) : Método responsável pelo cancelamento de uma reserva associada a uma carona por parte de um usuário logado no sistema. Deve ser passado uma Reserva(entidade composta por código de reserva, assento e número de bagagens) de modo a informar ao sistema sobre os dados referentes a uma reserva associada a uma carona por parte de um usuário logado no sistema, onde todos os dados referentes a essa reserva serão permanentemente apagados do sistema. Retorno booleano, em que 1 representa caso de sucesso, e 0 representa caso de falha.

void ListarReserva(CodigoDeCarona) : Método responsável pela listagem de todas as reservas que um usuário logado no sistema possui relacionadas à sua conta. Deve ser passado um código de carona como parâmetro a fim de se informar ao usuário todas as reservas associadas a uma carona oferecida por ele por meio de seu código correspondente. Não há retorno pois sua responsabilidade é de somente informar ao usuário as reservas associadas ao dado compatível ao de sua pesquisa.