Thales Cézar Castro
Análise e Desenvolvimento de Sistemas
Redes de Computadores
Atividade 03

O aplicativo criado foi um mini chat, onde é possível mandar mensagens pessoais e em um grupo com todos os usuários, bem como compartilhamento de arquivos. Foi desenvolvido em PHP com o framework CodeIgniter, com suporte de linguagens frontend (HTML, CSS e Javascript).

Em primeiro lugar, foi projetado e construído o banco de dados em mySQL. Nele, há cinco tabelas: usuário (com os dados de quem acessará o chat), mensagem (com todas as mensagens pessoais mandadas através do chat), arquivo (com todos os dados relativos a arquivos compartilhados entre usuários no sistema), geral (contendo mensagens em chat coletivo) e arquivogeral (com os arquivos compartilhados no chat coletivo). As tabelas mensagem e arquivo contém dois ids de dois usuários diferentes por referência (o que manda e o que recebe).

A partir daí, foram as primeiras configurações do framework. Em autoload.php, no diretório config dentro de application, foram inseridas as bibliotecas de banco de dados e sessão, assim como a helper url. Já entre as modificações do arquivo config.php estão a inserção do endereço do site (em localhost) e da definição da index na controller padrão. Em database.php, ficou estabelecido o localhost como nome do host, usuário root para acesso ao banco “chat”, sem senha e usando driver mysqli. Grande parte dos arquivos, inclusive nos que não tiveram mudanças, foi inserida documentação.

Além das views criadas para funcionarem como partes da página (header, footer e nav), foram desenvolvidas várias para mostrar o conteúdo principal. O site inicia em uma tela de login que autentica o usuário que quer acesso ao chat. Uma vez verificado, a tela inicial mostra todos os usuários em que aquele que acessa pode trocar mensagens, bem como link para a sala coletiva. Na barra de navegação, há páginas com perfil da sessão e dados das demais pessoas. As janelas de chat mantem todos as mensagens trocadas, bem como formulário para envio de novas e arquivos.

As controllers foram criadas com o objetivo de mostrar as views com os dados correspondentes de cada uma e para controlar funções específicas do chat, como a inserção de mensagens e arquivos novos. Apesar de um procedimento em Javascript controlar a frequência de atualização de uma janela de conversa, ela chama um procedimento em PHP que recupera os dados do banco novamente e os apresenta na view, atualizando as mensagens e arquivos da conversa constantemente.

Apesar de os sockets terem sido estabelecidos via framework e este já possuir uma documentação em inglês, foi necessário vasculhar os arquivos da pasta system do projeto para que fossem encontrados. Como a aplicação basicamente utiliza inserções e buscas no banco de dados, os sockets estão relacionados à conexão com eles.

Em mysqli_driver.php, há a função db_connect(). Se ela receber um hostname, uma variável $socket guarda-o, caso contrário, o valor é NULL. Este e outros dados, como database, hostname, port, user e password são passados para uma função que retorna TRUE se eles estiverem estabelecidos, possibilitando a conexão com o banco de dados. A função db_connect() é chamada em Session_database_driver.php pela open(), em verificação de conexão com o banco de dados, passando uma tabela. Também é usada em DB_driver.php, em initialize(), que inicia as configurações do banco de dados.

No construtor de cache_redis.php, é determinado o padrão de configuração para redis cache. Aqui estão o tipo de socket tcp, endereço 127.0.0.1, sem senha, porta 6379. Há a possibilidade de estabelecer um caminho para um arquivo de extensão “.sock” caso fosse do tipo unix. Os dados host, port e timeout são verificados quando é tcp.

Há também a chamada a socket no trackback da aplicação. Na função process() em Trackback.php, é feita uma tentativa de conexão com um socket, recebendo um host e a porta 80. Caso haja conexão, os dados de resposta são resgatados e transferidos. A process() é chamada pela send(), que preliminarmente reúne os dados para a tentativa de trackback.