

## TRABALHO PRÁTICO

### Metas 1 e 2 - 6 valores

Pretende-se que seja desenvolvido um sistema distribuído que obedeça aos requisitos funcionais e arquiteturais definidos nas próximas secções. A implementação deve ser feita recorrendo à linguagem Java utilizando os conceitos estudados ao longo da unidade curricular. O sistema é composto por quatro tipos de componentes, sendo que três correspondem às aplicações que devem ser desenvolvidas: **Cliente**, através do qual os utilizadores interagem com o sistema; **Servidor**, responsável por toda a lógica de negócio e pelo armazenamento replicado de ficheiros; **Gestor de Registo Dinâmico de Servidores** (GRDS), responsável por manter o registo dos servidores ativos, direccionar as aplicações cliente para os diversos servidores e difundir informação relevante; e **Sistema de Gestão de Base de Dados** (SGBD), responsável pela gestão e persistência da informação de suporte ao sistema (contas de utilizadores, estado de ligação dos utilizadores, histórico de mensagens trocadas e notificações de ficheiros disponibilizados, grupos, contactos estabelecidos, etc.).

#### 1. Requisitos Funcionais

O sistema distribuído pretendido deve oferecer as seguintes funcionalidades:

- **Registo de novos utilizadores**, sendo estes caracterizados por um nome, um *username* e uma *password*;
- **Autenticação de utilizadores** já registados através do respetivo *username* e *password*;
- **Edição dos dados de registo** pelos utilizadores;
- **Listagem e pesquisa de utilizadores**;
- **Estabelecimento de contacto entre utilizadores**. Quando um pedido emitido por um utilizador é aceite, o destinatário passa a integrar a sua **lista de contactos** e vice-versa. Se for recusado, fica sem efeito;
- **Visualização da lista de contactos**, com indicação do estado de cada um (*online* ou *offline*);
- **Eliminação de um contacto** da lista de contactos. Esta operação também elimina o requerente da lista de contactos do visado, bem como os respetivos históricos de troca de mensagens e de disponibilização de ficheiros;
- **Criação de grupos** (o criador de um grupo também é o seu administrador);
- O administrador de um grupo pode **alterar o nome do grupo**, **excluir** membros e **extinguir o grupo**. Quando um grupo é extinto, o histórico e o grupo deixam de ser visíveis para todos os membros e os ficheiros disponibilizados são eliminados nos diversos servidores onde existem;
- **Adesão de um utilizador a um grupo**. Uma adesão torna-se efetiva apenas depois da aceitação por parte do administrador;

- **Saída de um utilizador de um grupo**, sendo as respetivas listas de mensagens trocadas e de ficheiros disponibilizados eliminadas do seu histórico;
- **Visualização da lista de grupos** existentes no sistema e da sua constituição (membros);
- **Envio de mensagens e disponibilização de ficheiros** a qualquer contacto ou grupo. A disponibilização de um ficheiro por um utilizador resulta no carregamento (*upload*) deste no servidor ao qual se encontra ligado e na sua posterior replicação pelos restantes servidores ativos (em *background*). As linhas orientadoras para o processo de replicação são apresentadas mais à frente. Uma notificação de disponibilização de um ficheiro permite que o cliente o obtenha (faça *download*) a partir do servidor ao qual se encontra ligado;
- **Listagem das mensagens trocadas e dos ficheiros disponibilizados com contactos e grupos** (histórico de um utilizador), sendo indicado para cada item o momento de disponibilização e se já foi visto (i.e., apresentado na interface do utilizador) por pelo menos um dos destinatários;
- **Eliminação de mensagens e de notificações de disponibilização de ficheiros no histórico de um utilizador** (relativas tanto a contactos como a grupos). Uma mensagem ou notificação de disponibilização de um ficheiro eliminada pelo seu emissor também é removida do histórico do(s) destinatário(s). Adicionalmente, as réplicas de um ficheiro também são apagadas em todos os servidores onde existem. A eliminação das notificações de disponibilização pelos respetivos emissores é a única forma de apagar ficheiros;
- **Atualização assíncrona (e automática) da informação visualizada pelos utilizadores** quando ocorre uma alteração qualquer (estado dos contactos, histórico de mensagens e de ficheiros disponibilizados, mensagens novas e notificações de ficheiros disponibilizados ainda não visualizadas, pedido de estabelecimento de contacto ou de adesão a um grupo, etc.).

Ao implementar as funcionalidades descritas, tenha em consideração os seguintes aspetos:

- As funcionalidades só estão disponíveis para utilizadores autenticados, o que significa que sem autenticação um utilizador não tem acesso ao sistema;
- Não podem ser criados utilizadores com nomes e *usernames* iguais;
- Podem existir grupos com nomes iguais desde que sejam criados por utilizadores distintos;
- Cada notificação de ficheiro disponibilizado corresponde a um ficheiro distinto mesmo que o seu nome seja igual, independentemente de quem o disponibilizou;
- As operações de carregamento e de obtenção de ficheiros podem ser morosas. Por esta razão, estas devem ser feitas em *background*, tanto do lado dos clientes como dos servidores;
- De um modo genérico, as aplicações cliente e servidor necessitam de suportar a execução simultânea de diversas operações (*downloads*, *uploads*, notificações

assíncronas, etc.). Desta forma, deve recorrer-se aos mecanismos de programação concorrente estudados (e.g., *threads*) sempre que se justificar a sua utilização;

- Existem aspetos relacionados com as funcionalidades descritas que estão omissos neste enunciado. Os grupos de trabalho têm total liberdade para lidar com essas questões e implementarem soluções da forma que melhor entenderem. Em caso de dúvidas, devem contactar um dos docentes.

## 2. Requisitos Arquiteturais e Protocolares

A arquitetura geral do sistema pretendido é apresentada na Figura 1, sendo constituída pelos seguintes elementos:

- Um **sistema de gestão de base de dados** (SGBD) SQL;
- **Múltiplos servidores**, todos com acesso à mesma base de dados no SGBD. Esta inclui a informação necessária ao funcionamento do sistema, com exceção dos ficheiros disponibilizados pelos utilizadores aos seus contactos ou grupos. Estes ficheiros são replicados nos sistemas de ficheiros locais dos vários servidores ativos no momento da disponibilização;
- Um **gestor de registo dinâmico de servidores** (GRDS) responsável por saber quais são os servidores ativos e distribuir os clientes por estes segundo uma política de escalonamento circular (*round-robin*). Também informa, via UDP, os servidores sobre as coordenadas (endereço IP e porto TCP de escuta) do conjunto de servidores ativos e serve de refletor de informação relevante como, por exemplo, pedidos de contato, envio de mensagens e disponibilização de ficheiros;
- **Múltiplos clientes**, sendo que cada um comunica com um único servidor via TCP e com o GRDS via UDP.

As aplicações cliente, servidor e GRDS devem ter o seguinte funcionamento:

- Na fase de arranque, os clientes e os servidores recebem o endereço IP e o porto de escuta UDP do GRDS. Os servidores também recebem o endereço IP do SGBD;
- Um servidor também pode ser lançado sem indicação das coordenadas do GRDS. Neste caso, tenta descobri-lo transmitindo um pedido para o endereço de *multicast* 230.30.30.30 e porto UDP 3030. Se não obtiver qualquer resposta ao fim de três tentativas consecutivas, termina;
- Depois do arranque, os clientes começam por contactar, via UDP, o GRDS para receberem as coordenadas (IP e porto de escuta TCP) de um servidor ativo. Se existir mais do que um servidor ativo, clientes sucessivos são dirigidos para servidores distintos segundo uma política de escalonamento circular. O cliente estabelece uma ligação TCP com o servidor indicado;

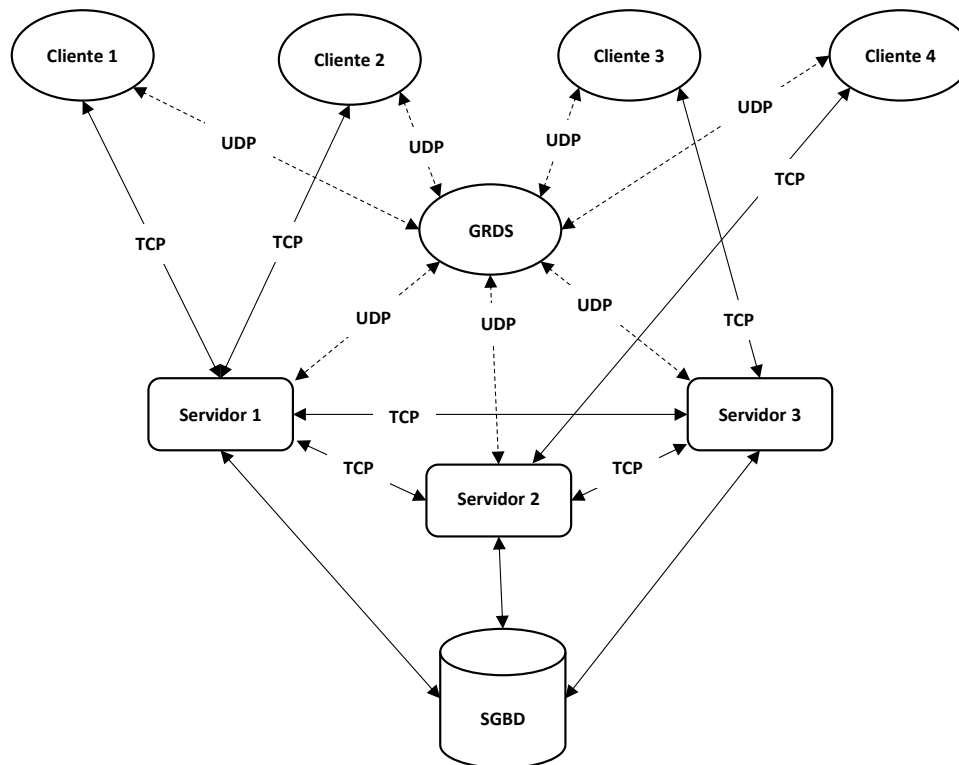


Figura 1 - Arquitetura geral do sistema

- Um cliente que eventualmente perca a sua ligação com um servidor volta automaticamente a contactar o GRDS e a estabelecer uma ligação com o novo servidor indicado. Deve ser minorado o impacto ao nível dos utilizadores (interrupção de serviço impercetível ou mínima);
- Os servidores devem enviar, via UDP e com uma periodicidade de 20 segundos, uma mensagem ao GRDS com indicação de um porto de escuta TCP (automático) em que possam aceitar ligações de clientes. Passados três períodos sem receção de mensagens de um determinado servidor, este é “esquecido” pelo GRDS;
- Os clientes são encaminhados apenas para servidores sem mensagens periódicas em falta;
- Cada servidor acede à mesma base de dados via a API JDBC, sendo recomendada a utilização do sistema MySQL. No entanto, existe liberdade de escolha de outro SGBD relacional;
- Toda a informação do sistema é armazenada na base de dados partilhada, com exceção dos ficheiros disponibilizados pelos utilizadores. Estes são armazenados diretamente nos sistemas de ficheiros locais. Na base de dados apenas são armazenados dados que permitam aos servidores identificá-los/localizá-los;
- Quando um servidor efetua uma alteração na base dados na sequência de uma interação com um cliente (emissão de um pedido de contato ou de adesão a um grupo, aceitação ou recusa de um pedido de contacto ou de adesão a um grupo, envio ou eliminação de uma mensagem, disponibilização ou eliminação de um ficheiro,

registo ou autenticação de um utilizador, saída ou perda de conexão com um cliente, criação ou eliminação de um grupo, eliminação de um contacto, etc.), este envia ao GRDS, via UDP, uma mensagem a informar que houve alterações. Deve ser incluída nesta a identificação dos utilizadores afetados pela alteração. Por sua vez, o GRDS reencaminha a informação recebida, via UDP, para os restantes servidores ativos. Caso tenha sido disponibilizado um ficheiro, também devem ser fornecidos o endereço IP do servidor onde o ficheiro se encontra e o porto de escuta TCP (automático) onde este aceita ligações destinadas exclusivamente à transferência de ficheiros;

- Quando um servidor recebe a informação mencionada no ponto anterior, este informa os clientes afetados aos quais se encontra conectado que houve uma alteração na base de dados e que estes devem atualizar as suas vistas (mensagens/notificações visualizadas, número de mensagens/notificação não visualizadas, pedidos de contacto ou de adesão a grupos, etc.);
- Quando está em causa a disponibilização de um ficheiro, os servidores que recebem a informação vinda do GRDS também obtêm o ficheiro via uma ligação TCP temporária estabelecida com o servidor no endereço IP e porto TCP indicados. A transferência deve ser feita em *background*;
- Quando é solicitada a eliminação de um ficheiro pelo utilizador que o disponibilizou, a mensagem enviada para o GRDS também deve incluir esta indicação para que todos os servidores apaguem o ficheiro nos seus sistemas de ficheiros locais;
- Não é exigido implementar qualquer mecanismo que garanta um nível estrito de consistência ao nível da replicação dos ficheiros em servidores que estiveram inativos. Por exemplo, um servidor que tenha estado inativo durante um período poderá não ter uma cópia local dos ficheiros entretanto disponibilizados, e continuará a ter aqueles que foram totalmente apagados em todos os utilizadores. Estas anomalias devem, de alguma forma, ser indicadas aos respetivos clientes;
- Qualquer servidor que identifique na base de dados um utilizador marcado como estando *online* e que não tenha esse estado revalidado há mais de 30 segundos deve alterá-lo para *offline*;
- Em todos os cenários expostos, a comunicação pode ser feita da forma que os grupos entenderem ser mais apropriada. Ou seja, podem ser utilizadas estratégias baseadas em cadeias de caracteres ASCII ou em objetos serializados;
- Quando um servidor termina de forma ordenada/intencional, este encerra as ligações TCP ativas, o que faz com que os clientes que se encontram ligados a ele também terminem de forma ordenada, informa o GRDS e atualiza a informação na base de dados.

### 3. Extras

A interface do utilizador da aplicação cliente descrita neste enunciado pode ser implementada em modo consola (i.e., texto). Os aspetos fundamentais considerados na avaliação base são se cumpre as funcionalidades pretendidas e se apresenta toda a informação necessária de forma adequada aos utilizadores. No entanto, os grupos que apresentem uma aplicação que cumpra minimamente os requisitos essenciais e que tenha uma interface do utilizador gráfica (GUI) funcional e completa, terão uma bonificação extra que poderá ir até aos 7.5% da nota atribuída. Ou seja, um trabalho avaliado em 80% que tenha a totalidade deste extra passa a valer 86% ( $80\% + 80\% * 7.5\%$ ).

Conforme indicado na secção 2, não é exigido implementar qualquer mecanismo que garanta um nível estrito de consistência ao nível da replicação dos ficheiros. Se for apresentada uma solução funcional, será atribuída uma bonificação extra que poderá ir até aos 7.5% da nota atribuída.

Um trabalho avaliado em 80% que tenha a totalidade dos dois extras referidos passa a valer 92% ( $80\% + 80\% * 2 * 7.5\%$ ). Caso a nota final com bonificação ultrapasse os 100%, esta será truncada neste valor.

### 4. Considerações Gerais

Deve ter-se em consideração os seguintes aspetos:

- O trabalho deve ser realizado preferencialmente por grupos de três alunos, não podendo este valor ser ultrapassado nem inferior a dois;
- A constituição dos grupos deverá ser registada em momento oportuno através da plataforma Moodle;
- A **1ª meta** do trabalho prático consiste na definição do **modelo entidade relacionamento e modelo físico da base de dados**. Para o efeito, deverá ser entregue, até ao dia **21 de novembro de 2021**, através da plataforma InforEstudante, um ficheiro PDF com a designação *PD-2021-F1-TP-Gx.pdf*, sendo x o número do grupo;
- A **2ª meta** consiste no **desenvolvimento do trabalho** e deverá ser entregue até ao dia **2 de janeiro de 2022**, através da plataforma InforEstudante, num ficheiro ZIP com a designação *PD-2021-F2-TP-Gx.zip*, sendo x o número do grupo;
- O ficheiro referido no ponto anterior deve incluir o código fonte (ficheiros “.java”) e a documentação produzida, assim como os ficheiros auxiliares necessários à execução e teste das aplicações sem necessidade de recorrer a qualquer IDE (e.g., o *byte code* gerado e respetivas *batch files* e/ou ficheiros do tipo *jar* executáveis);
- As opções tomadas durante o projeto (e.g., aspetos não especificados no enunciado, variações devidamente justificadas ao nível das funcionalidades implementadas ou do modo de interação entre os diversos componentes, tratamento de anomalias, etc.), os aspetos relevantes do sistema desenvolvido (pormenores de

implementação, diagramas temporais, etc.) e o manual de utilizador devem ser devidamente documentados de um modo sintético num documento do tipo PowerPoint;

- No documento referido na alínea anterior, é aconselhável a utilização de figuras e capturas de ecrã;
- Os resultados das duas metas serão divulgados apenas após a defesa da segunda, sendo a repartição dos 6 valores feita da seguinte forma: 0.7 para a meta 1 e 5.3 para a meta 2;
- Não é expectável que diferentes grupos apresentem soluções iguais. Caso este cenário se verifique, os grupos envolvidos terão de se justificar. **A deteção de situações de plágio leva a uma atribuição direta de 0 valores na nota do trabalho aos alunos de todos os grupos envolvidos;**
- Será valorizado o facto de o código das aplicações desenvolvidas ter sido estruturado de uma forma modular, com uma separação clara entre lógica de funcionamento, lógica de comunicação e interface do utilizador, podendo esta ser em modo texto ou gráfico conforme já mencionado.