

Relatório – EP1

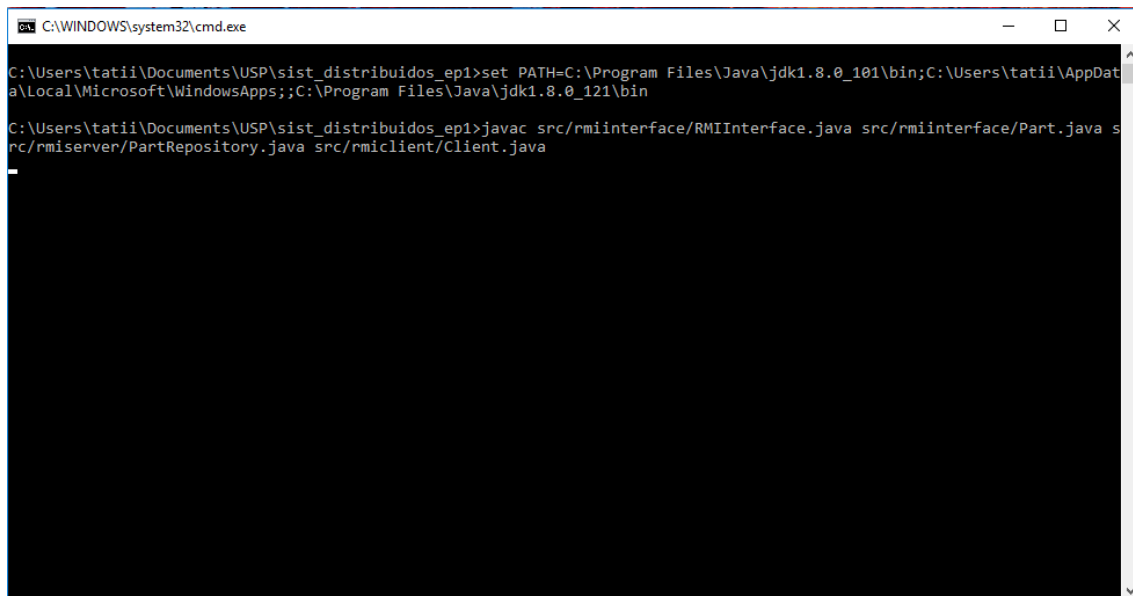
Anna Gabriella Breganholi de Almeida – 8517199

Tatiana Naomi Kuroiva de Siqueira – 8516671

A solução implementada neste exercício foi baseada na utilização da interface Java RMI, que permite a chamada e utilização de métodos que residem em outros servidores virtuais. Para que os exercícios propostos sejam executados, foi elaborada uma interface básica, com botões que carregam o nome das funções que se deseja realizar. A implementação de uma interface deixa o sistema “mais amigável” e intuitivo ao ser utilizado. A seguir, serão apresentados exemplos de funções que o sistema possui e suas respectivas saídas.

Na pasta submetida encontram-se uma série de arquivos que fazem parte da implementação da solução. Na pasta src se encontram os arquivos .java que implementam a solução do programa, subdivididos em pastas com os nomes rmiclient, rmiinterface e rmiserver, que cuidam da implementação do cliente que vai se conectar ao servidor remoto, da interface que o servidor implementa, e a implementação dos servidores e suas subpartes, respectivamente. O código das classes e interfaces implementadas contém mais informações sobre os métodos utilizados na solução (olhar os arquivos .java dentro de cada uma das pastas citadas acima).

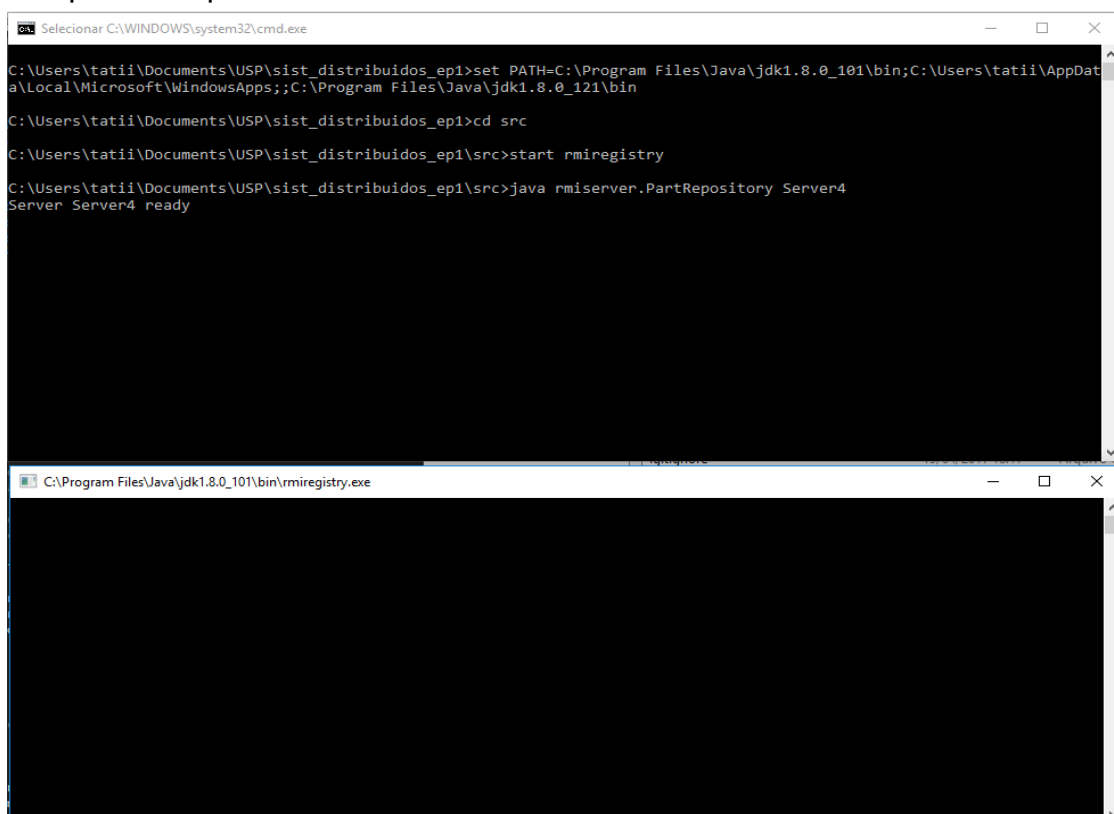
Como citado anteriormente, a solução possui uma interface que permite a conexão de clientes com servidores remotos e demais exercícios. Inicialmente o programa deve ser compilado. O arquivo 1. compileEverything.bat deve ser iniciado (ele é iniciado automaticamente com clique duplo sobre o arquivo). A imagem a seguir mostra o compilador que compilará os dados antes da execução. Após ser compilado, o terminal será fechado automaticamente. A imagem (Imagem 1) a seguir mostra a janela do prompt de comando que abre brevemente e fecha assim que o arquivo é compilado.



```
C:\WINDOWS\system32\cmd.exe
C:\Users\tatii\Documents\USP\sist_distribuidos_ep1>set PATH=C:\Program Files\Java\jdk1.8.0_101\bin;C:\Users\tatii\AppData
a\Local\Microsoft\WindowsApps;;C:\Program Files\Java\jdk1.8.0_121\bin
C:\Users\tatii\Documents\USP\sist_distribuidos_ep1>javac src/rmiinterface/RMIInterface.java src/rmiinterface/Part.java s
rc/rmiserver/PartRepository.java src/rmiclient/Client.java
```

Imagem 1: prompt de comando aberto ao executar o arquivo 1. compileEverything.bat

Uma vez compilado, o próximo passo é executar o arquivo 2. startServer.bat. Como o próprio nome sugere, ele irá iniciar o servidor remoto o qual o cliente irá se conectar mais adiante. Diferente do arquivo anterior, duas janelas do prompt de comando irão abrir, mas não serão fechadas automaticamente (Imagem 2), e devem ser mantidas abertas para a execução dos próximos passos.



```
Selecionar C:\WINDOWS\system32\cmd.exe
C:\Users\tatii\Documents\USP\sist_distribuidos_ep1>set PATH=C:\Program Files\Java\jdk1.8.0_101\bin;C:\Users\tatii\AppData
a\Local\Microsoft\WindowsApps;;C:\Program Files\Java\jdk1.8.0_121\bin
C:\Users\tatii\Documents\USP\sist_distribuidos_ep1>cd src
C:\Users\tatii\Documents\USP\sist_distribuidos_ep1\src>start rmiregistry
C:\Users\tatii\Documents\USP\sist_distribuidos_ep1\src>java rmiserver.PartRepository Server4
Server Server4 ready

C:\Program Files\Java\jdk1.8.0_101\bin\rmiregistry.exe
```

Imagem 2: prompts de comando abertos ao executar o arquivo 2. startServer.bat

O próximo passo é executar o programa 3. runClient.bat, programa responsável por conectar o cliente com o servidor remoto. Com base no arquivo

executado anteriormente, o primeiro prompt da Imagem 2 mostra na ultima linha uma mensagem que diz que o servidor Server4 está pronto. Esse é o servidor que deverá ser escolhido para o cliente se conectar. Esse servidor pode ser alterado no arquivo 2. startServer.bat, para isso o arquivo deve ser aberto em um editor de texto, e a última linha de comando deve ser alterada para “java rmiserver.PartRepository ServerX”, sendo X o servidor que deseja se conectar.

Ao executar o programa, uma janela “pop-up” abrirá para que o cliente digite o nome do server ao qual deseja se conectar. A janela é uma interface simples de execução, ao invés de comandos direto no prompt de comando. No caso do exemplo, como o servidor ativo é Server4, ele deve ser digitado na caixa de texto, como mostra a imagem 3.

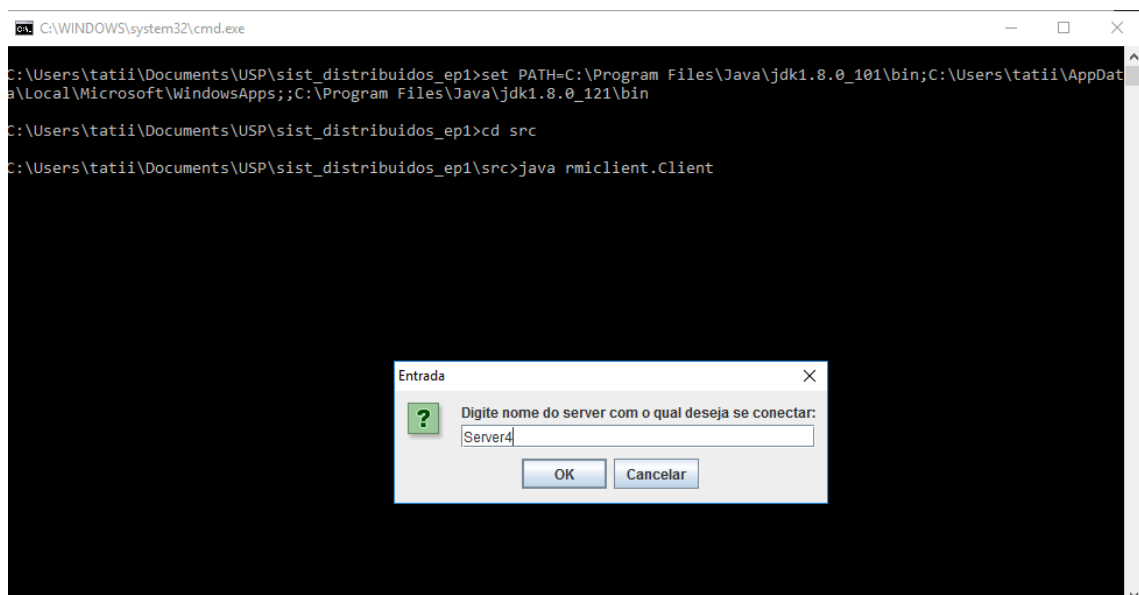


Imagem 3: janela pop-up de conexão ao servidor desejado

Após a conexão do cliente ao servidor ser iniciada, uma nova janela pop-up será aberta, com as opções que o cliente tem dentro do servidor selecionado (Imagem 4).

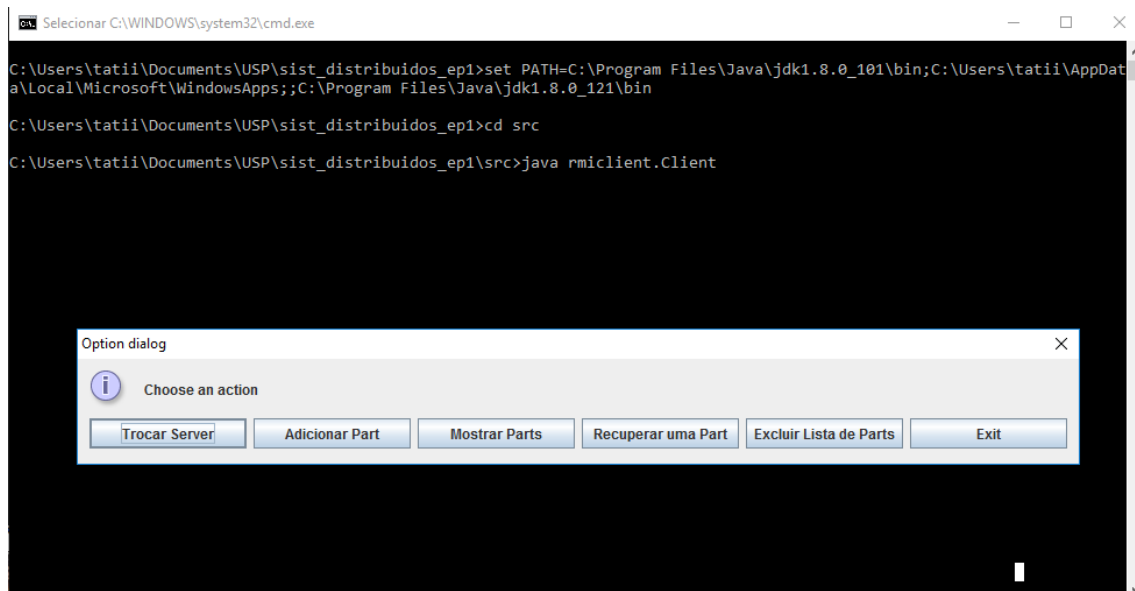


Imagem 4: interface de execução do programa

A primeira opção do menu é a de trocar o servidor ao qual o cliente está conectado. Quando selecionada esta opção, uma nova janela será aberta, listando os servidores ativos que o cliente pode escolher. Para se conectar a outro servidor, o cliente deve digitar apenas o nome entre as duas últimas barras. No exemplo da Imagem 5, o cliente tem apenas o servidor “Server4” listado, logo, ele deve digitar apenas “Server4”.

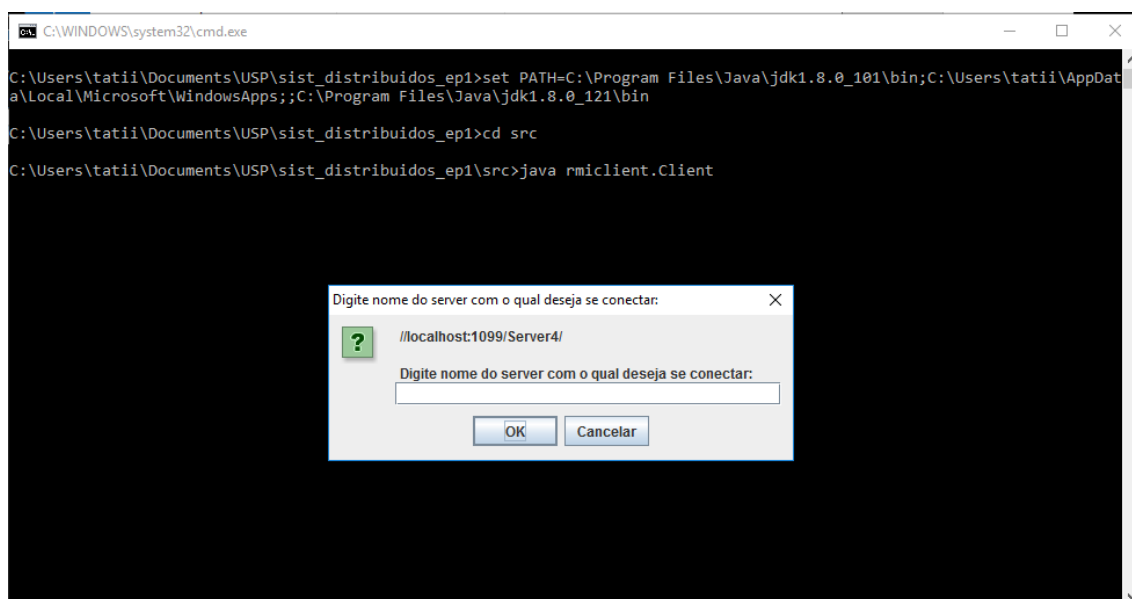


Imagem 5: menu para o cliente trocar servidores

A próxima função do menu adiciona novas peças, podendo estas serem tanto primitivas como agregadas. A imagem 6 mostra que uma nova janela será

aberta com os campos “Nome”, “Descrição”, e “Subparts”, sendo o campo nome o identificador nominal da peça, a descrição um atributo descritivo para quaisquer usos do cliente, e subparts sendo o campo que define se a peça será primitiva ou agregada. No primeiro caso, o campo deve ser deixado em branco, e no segundo, o identificador de uma das partes listadas na parte superior da janela deve ser escolhido para a parte se agregar (ID: XXXXXX), e a quantidade de peças é de escolha do cliente.

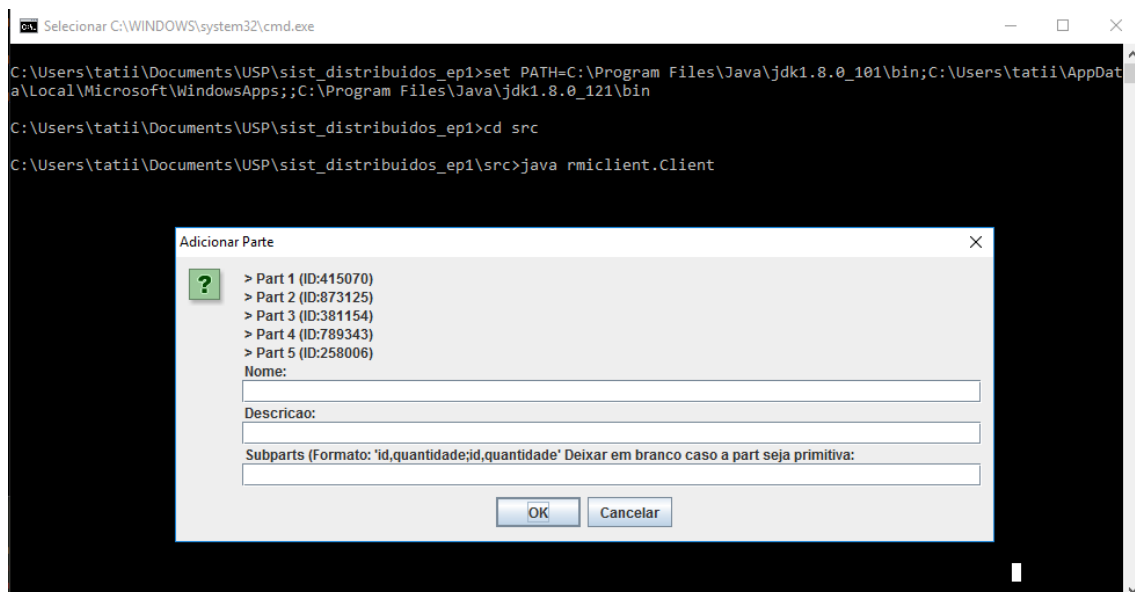


Imagem 6: menu para adicionar uma nova parte (primitiva ou agregada)

O resultado da adição de uma nova parte pode ser verificado ao voltar para o menu inicial, e escolher a opção “Mostrar Parts”. Uma nova janela será aberta com a listagem de todas as partes componentes existentes, independentemente de serem agregadas ou primitivas. Todas as partes listadas trazem também o seu número identificador. A imagem 7 mostra que a inserção de uma nova parte chamada “Part 6” foi realizada com sucesso quando comparada com a listagem da Imagem 6.

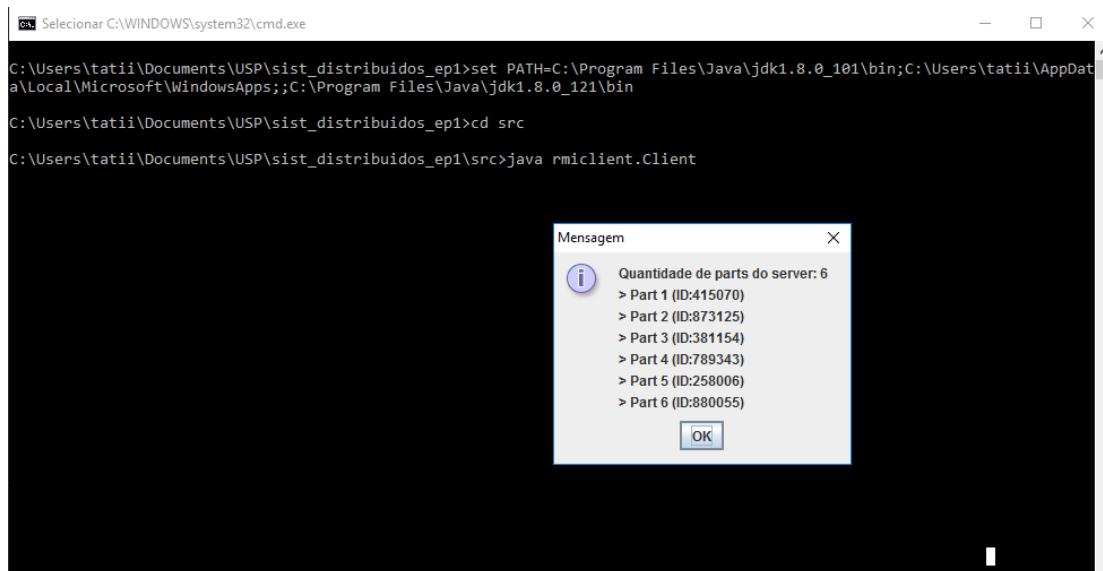


Imagem 7: lista com todas as partes existentes

Antes de seguir para a função “Resgatar Parte”, vamos analisar a função “Excluir Lista de Parts”. Essa função tem como resultado limpar todos os componentes do sistema, e tal ação pode ser confirmada ao verificar a saída da função “Mostrar Parts” novamente, que deve abrir uma janela com a mensagem “Quantidade de Parts do server: 0”.

Os identificadores das partes listados na função de mostrar todas as partes são as entradas para a próxima função do menu inicial, “Recuperar Part”. Essa função abrirá uma caixa de diálogo onde o ID de uma das partes listadas anteriormente deve ser inserido, o que abrirá outra caixa de diálogo com mais funções sobre a parte inserida, como mostra a Imagem 8. Caso seja inserido um identificador não existente na listagem de partes, o sistema dispara uma mensagem avisando que o identificador não existe, e retornará ao menu inicial.

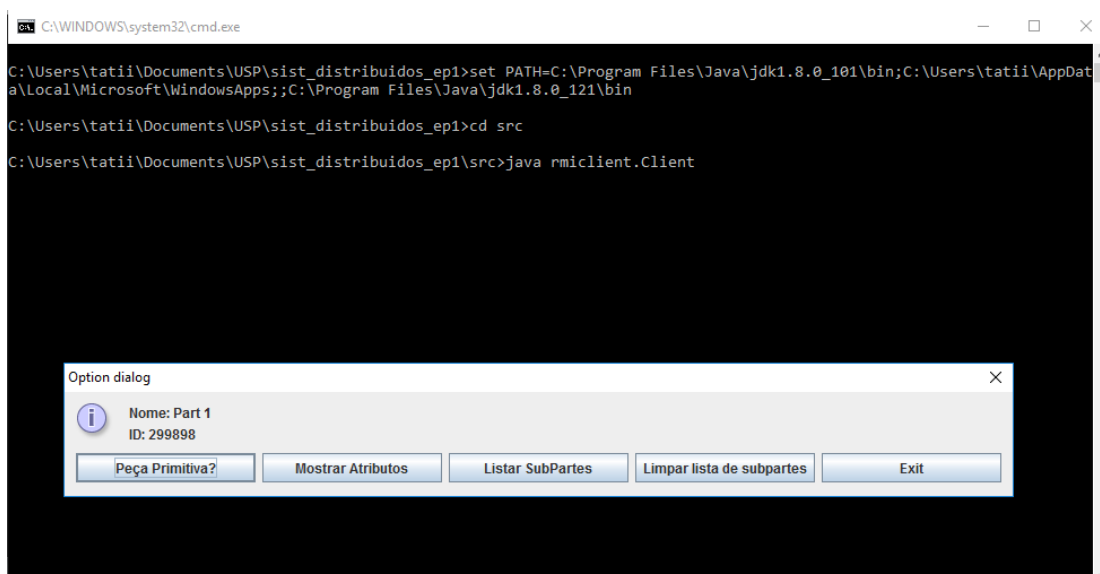


Imagem 8: caixa de diálogo com funções sobre a peça resgatada

A primeira função do novo menu verifica se a parte resgatada é primitiva ou agregada. O retorno da função se dá em uma nova janela com uma mensagem dizendo qual é o tipo da peça. Após a realização de cada função, o cliente tem como opção se manter no menu com operações sobre a parte resgatada ou voltar para o menu inicial onde as operações se aplicam a todas as partes do sistema.

A segunda função do menu, “Mostrar Atributos”, lista as informações sobre a parte resgatada. No exemplo abaixo, a imagem 9 fez o resgate da peça 1, logo, os atributos nome, identificador (ID) e a descrição da parte são listados em uma nova janela. Como a função anterior verifica o tipo da parte (primitiva ou agregada), não há necessidade de repetir a informação na função atual. Caso o cliente deseje ver a lista de outros atributos, ele deverá voltar para o primeiro menu e resgatar uma peça diferente da atual.

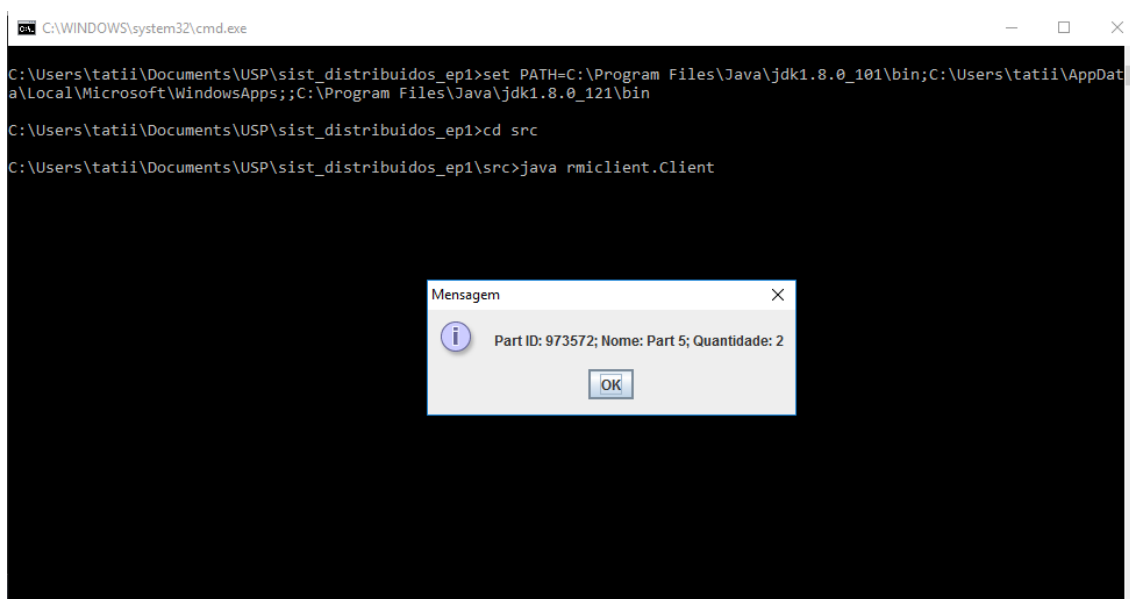


Imagem 9: lista de atributos da peça resgatada

A próxima função do menu, “Listar Subpartes”, tem dois tipos de saídas diferentes. Caso a parte seja primitiva, uma janela será aberta dizendo que a parte é primitiva, logo, não possui subpartes a serem listadas. No caso da parte resgata ser agregada, ao selecionar a função será aberta uma caixa de diálogo listando o identificador da parte a qual a parte atual está agregada, o nome da parte a qual está agregada, e a quantidade de subpartes (Imagem 10).

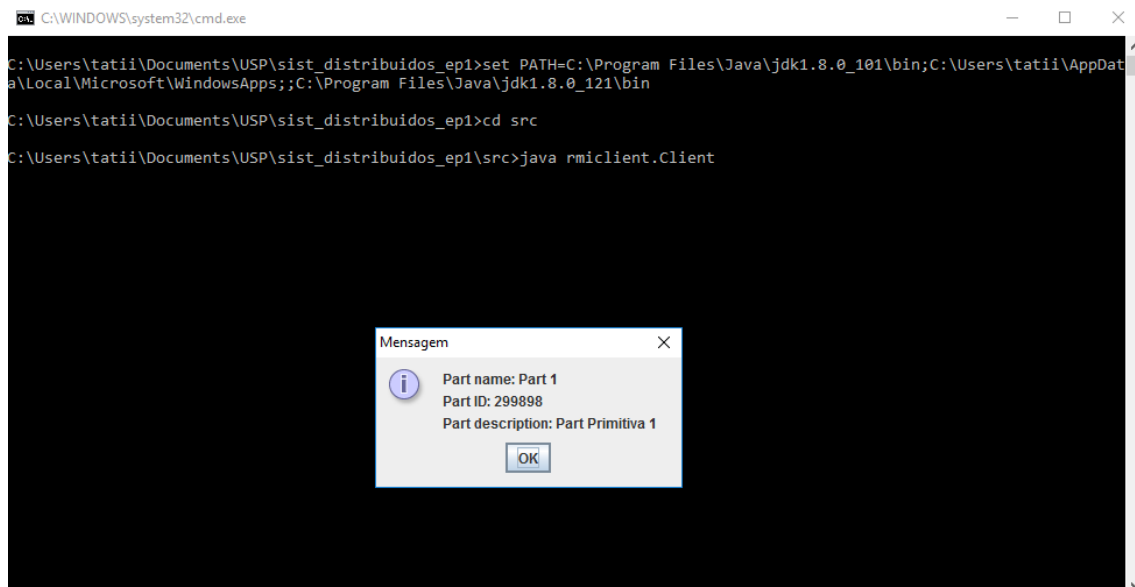


Imagem 10: lista de subpartes (no exemplo acima, a peça é agregada e dispõe de subpartes)

Por fim, as funções “Excluir Lista de Parts” e “Limpar Lista de Subpartes” tem como resultado a limpeza das respectivas listas. Uma caixa de diálogo confirmará que a lista de componentes foi limpa (Imagem 11). Assim como em outras funções, o resultado dessas funções pode ser verificado com a função “Mostrar Parts”. O retorno desta função será uma mensagem dizendo que a quantidade de parts do server é igual a zero.

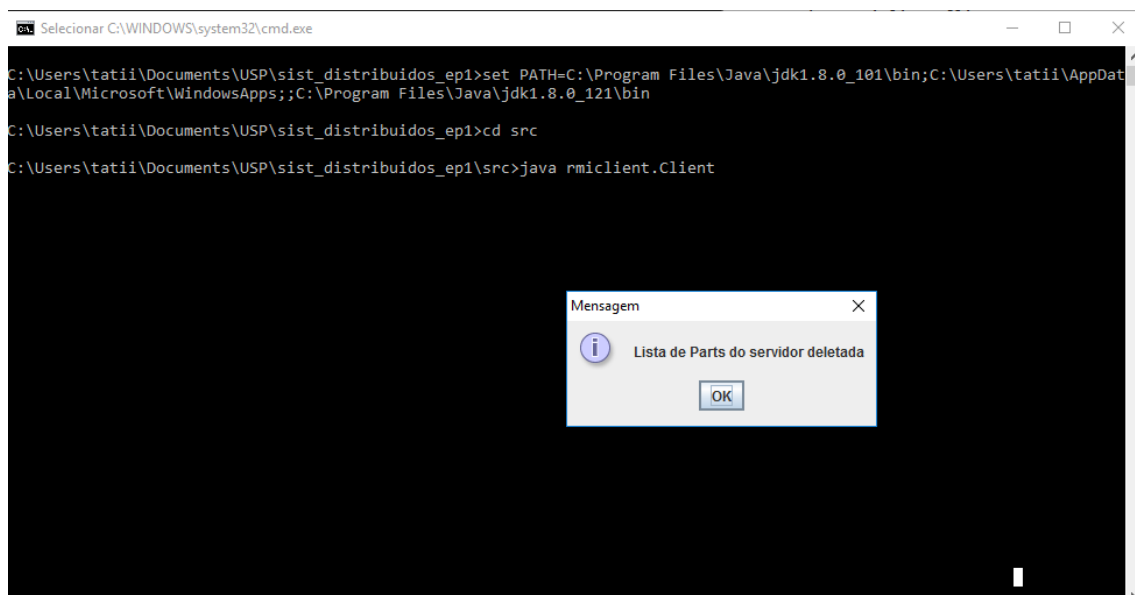


Imagem 11: caixa de diálogo da função “Excluir lista de Parts”