

Trabalho Prático – Parte 1

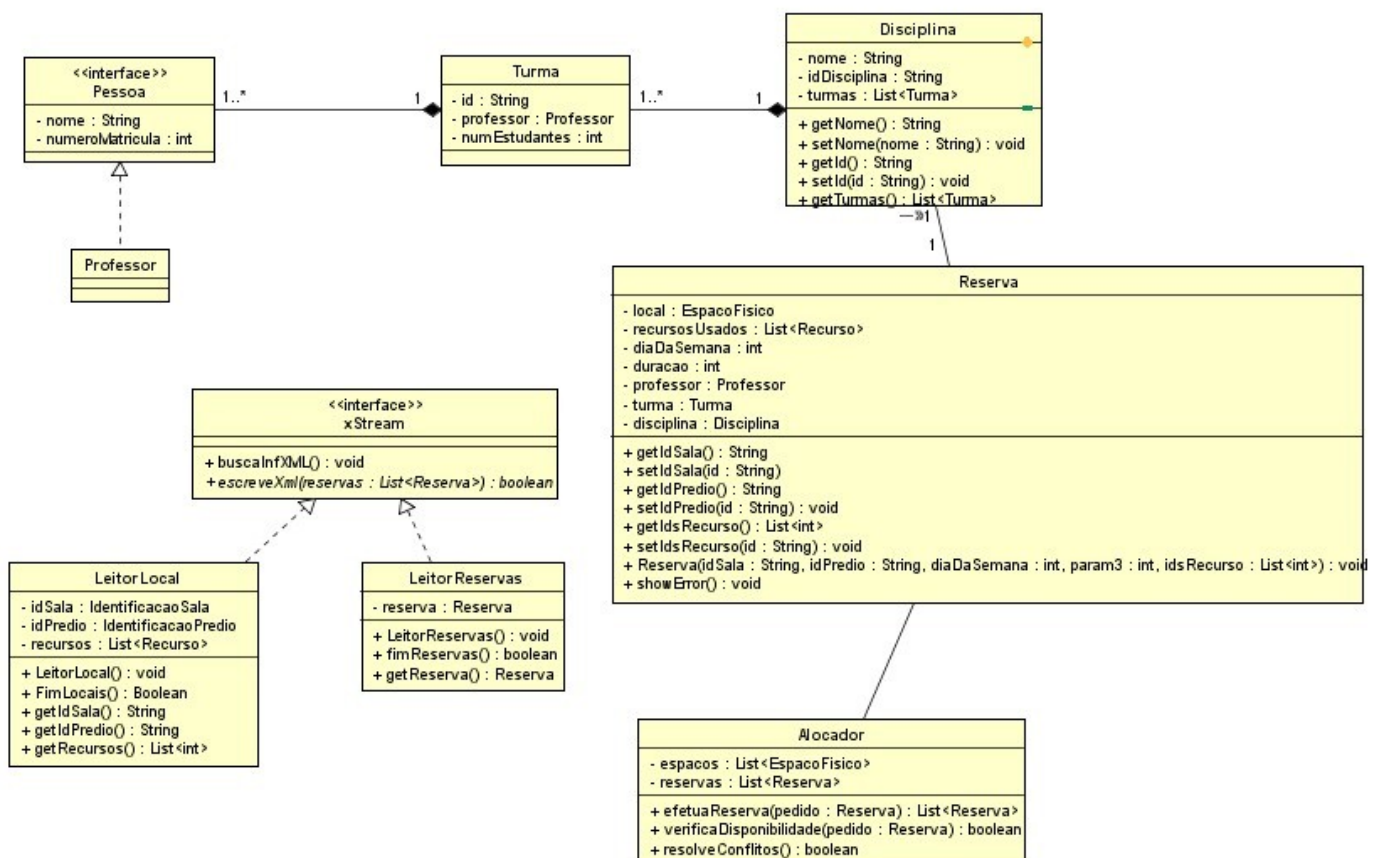
Introdução

Nesta primeira parte do projeto, ficamos a cargo de elaborar o planejamento do projeto através de diagramas de classe e diagramas de sequência. O projeto a ser implementado trata de um problema decorrente a cada semestre no Instituto de Informática: a alocação de salas de aula para as diversas turmas nas diversas cadeiras de graduação e pós-graduação a partir de certas informações, no caso, número de alunos, recursos necessários à aula e horários da cadeira.

A partir de um arquivo de entrada, no nosso caso do tipo XML, devemos, ao final do projeto, ter uma aplicação de alocação dos espaços físicos para essas atividades, otimizando o uso dos recursos disponíveis.

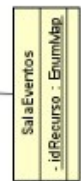
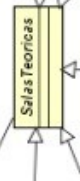
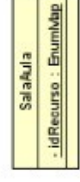
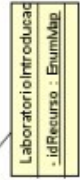
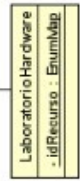
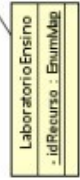
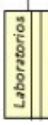
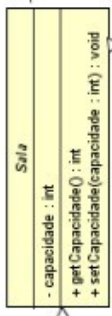
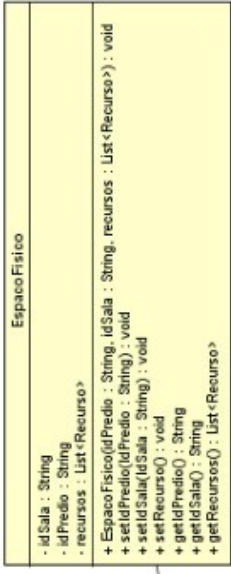
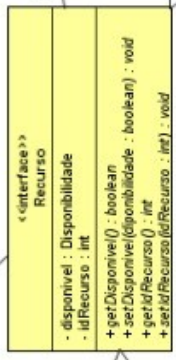
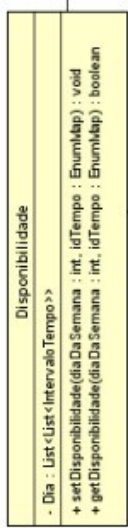
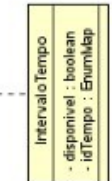
Com isso, o esperado é um arquivo de saída com a solução encontrada para o problema, também no formato XML. No nosso caso, é feita uma “atualização” do arquivo de entrada, completando as informações referentes em cada turma com o local onde a cadeira será dada.

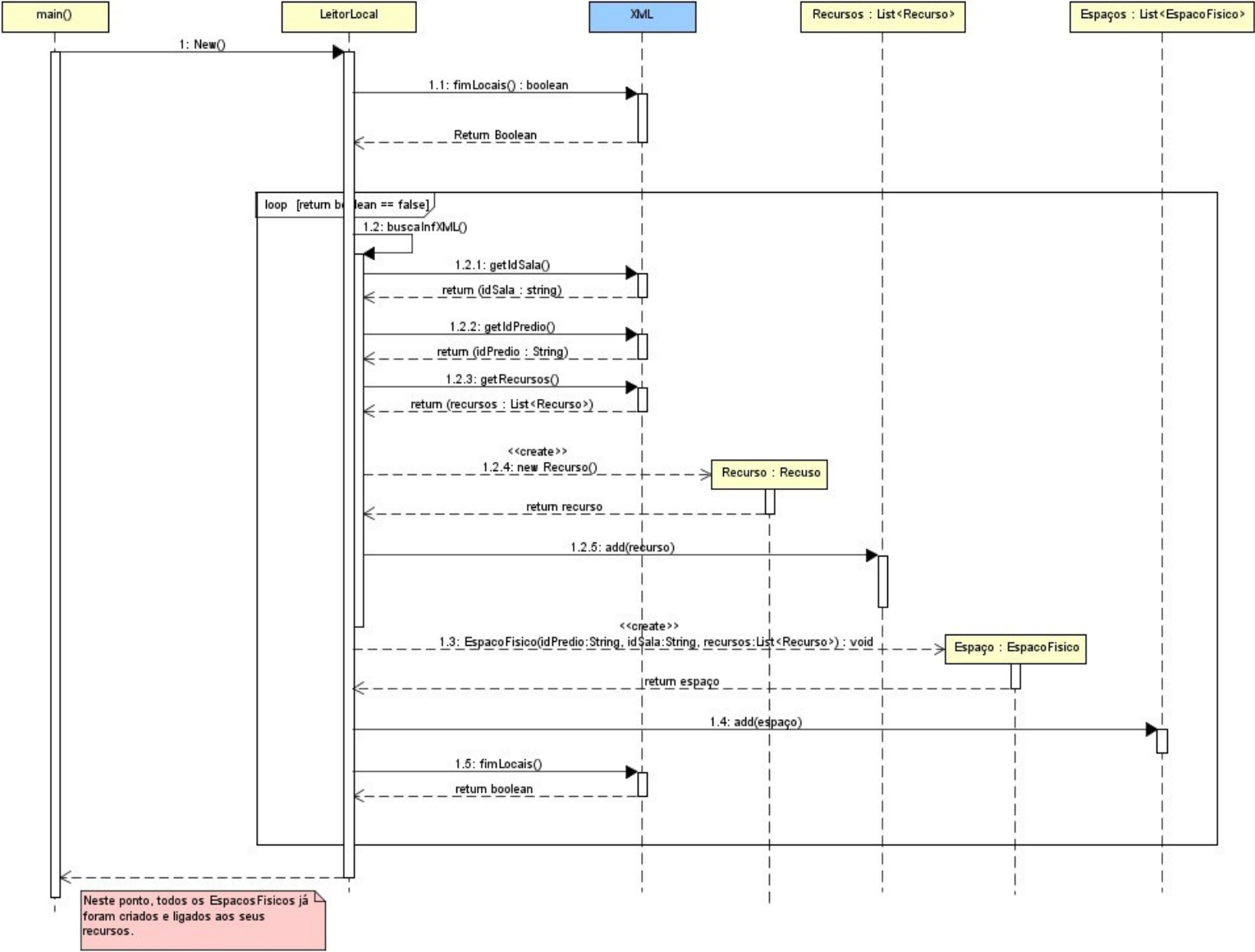
Diagramas

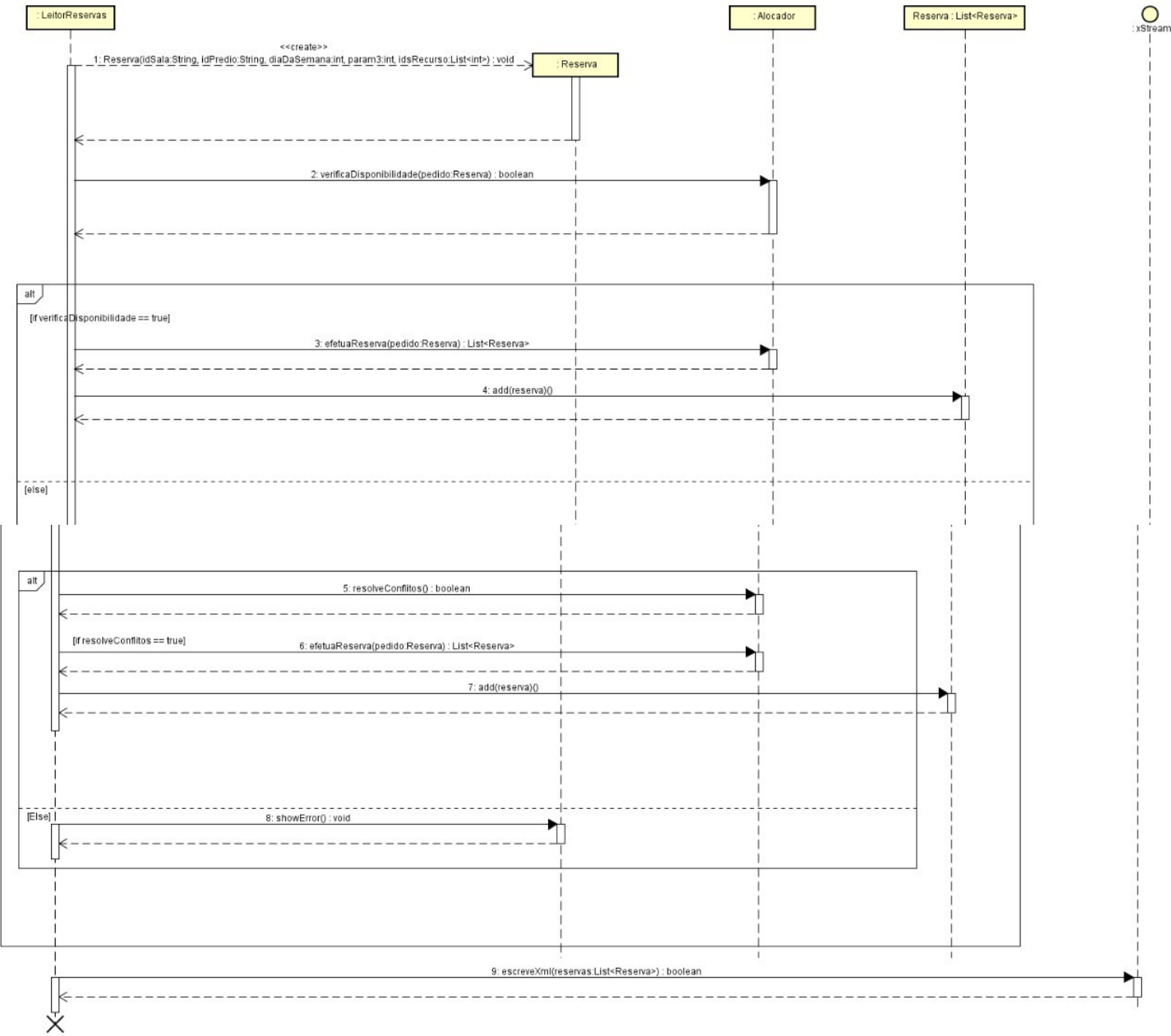


idTempo = 1 = 8:30 - 9:20
idTempo = 2 = 9:20 - 10:10
idTempo = 3 = 10:30 - 11:20
idTempo = 4 = 11:20 - 12:10
E assim por diante...

"Laboratório de ensino" id="1"
"Laboratório de ensino de hardware" id="2"
"Som e microfone" id="3"
"Reuniões" id="4"
"Apresentações" id="5"
"Tela motorizada" id="6"
"Eventos e Palestras" id="7"
"Sala de aula" id="8"
"Laboratório de ensino para cadeiras de introdução" id="9"
"Video conferencia" id="10"







Detalhamento do Projeto

Como se pode observar pelos diagramas de sequência, o programa inicia sua execução lendo o arquivo de entrada, do tipo XML, buscando os espaços físicos que estão disponíveis para alocação das turmas. Estes espaços estão divididos em Laboratórios (de Ensino, de Hardware e de Introdução) e Salas Teóricas (de Aula, Apresentação, Eventos, Reuniões e Videoconferência).

Após preencher uma lista com todos eles, ele começa a ler os pedidos de reserva e vai os alocando, um por um, verificando a disponibilidade das salas de acordo com os recursos necessários (por exemplo Som e Microfone e Telas Motorizadas). Quando uma sala já está reservada para outra cadeira, resolvemos esse problema através do “*resolveConflitos*”, que busca a reserva já feita, a analisa e vê se tem algum requerimento de sala/prédio/recurso/horário, movendo se necessário

Durante nossas discussões acerca do algoritmo, optamos por sempre buscar a menor sala com capacidade maior ao número de alunos inscritos, sempre respeitando os recursos requisitados. Fazemos isto para que, no caso de sobram salas de aula, sobrem as maiores para que nas atividades que venham a acontecer o tamanho da sala não seja um fator limitante, ou até mesmo para uma possível realocação de turma após matrículas posteriores por processo.

Além disto, também buscamos fazer uma interface que permite a evolução do software, podendo ele ser utilizado também para alocação de alunos nas disciplinas (vide item *Qualidade Interna*).

Qualidade

Interna

A qualidade interna de um software é aquele invisível a usuários, apenas vista por programadores e projetistas da aplicação. Neste primeiro momento, foi o que mais observamos e buscamos, por ser a fase de planejamento da implementação que é sequente a isto.

Assim, acabamos por buscar projetar um sistema bem modularizado, com uma média coesão e médio acoplamento. Com um sistema bem modularizado, é facilitada sua manutenção, sem que a mudança em um módulo afete muito os outros módulos que dele dependem.

Um ponto que observamos é a reusabilidade do software. Num primeiro momento nos questionamos se isso seria apenas destinado ao Instituto de Informática para alocação de turmas ou estendido à outras necessidades, como por exemplo, alocar alunos, reserva de salas para eventos ou até mesmo usado por outros departamentos. Analisando nosso programa nota-se que, com poucas mudanças, é possível fazer isso.

Outro ponto a ser observado é a extensibilidade do software, ou seja, a facilidade com que ele pode evoluir em versões seguintes. Durante nossas discussões, houve a dúvida de se era necessário também alocar os alunos na turma. Com a disponibilização de um exemplo de entrada, vimos que não, mas deixamos algumas pontas abertas para que, se em um segundo momento, quisermos isto, seja possível implementar.

Dentro dessa linha, também nos questionamos se todos os recursos para espaço físico já estavam listados, então tentamos planejar algo que, caso se fizessem necessárias algumas mudanças como inserção de lugares diferentes, elas não afetassem outros módulos da aplicação.

Externa

A qualidade externa depende intrinsecamente da qualidade interna. Esta é qualidade que é notada pelo usuário do programa, como, por exemplo, uma interface intuitiva e fácil de ser usada e a velocidade com que o programa dá respostas às entradas do usuário.

Durante esta primeira parte do trabalho, acabamos por não dedicar muito tempo a planejar a qualidade externa do software, focando na qualidade interna, da qual ela deriva. Assim, não pensamos em nada de interface ou interação com o usuário, utilizando-nos apenas do arquivo de entrada padronizado.

Por recebermos um arquivo XML padrão, acabamos por, num primeiro momento, não nos preocupando muito com a robustez do programa, o que pode levar a erros em caso de um usuário “desligado”, entretanto, este formato padrão de entrada também acaba por ajudar o usuário, já que é bem intuitivo a forma de serem inseridos dados para análise.

Análise Final

Como já comentamos anteriormente, por ser apenas um planejamento inicial e aberto a mudanças para evoluções do software, esta falta de cuidado com a qualidade externa pode ter um melhor cuidado na implementação do código na fase seguinte deste Projeto Final, não esquecendo de sua importância já que ela é a cara que o usuário tem do trabalho realizado.

Quanto a questão da qualidade interna, foi claramente nosso principal ponto forte. Por estarmos na posição de programadores e organizadores, nossos esforços buscam ajudar a nós mesmos, tanto na implementação quanto na futura manutenção e evolução da aplicação.