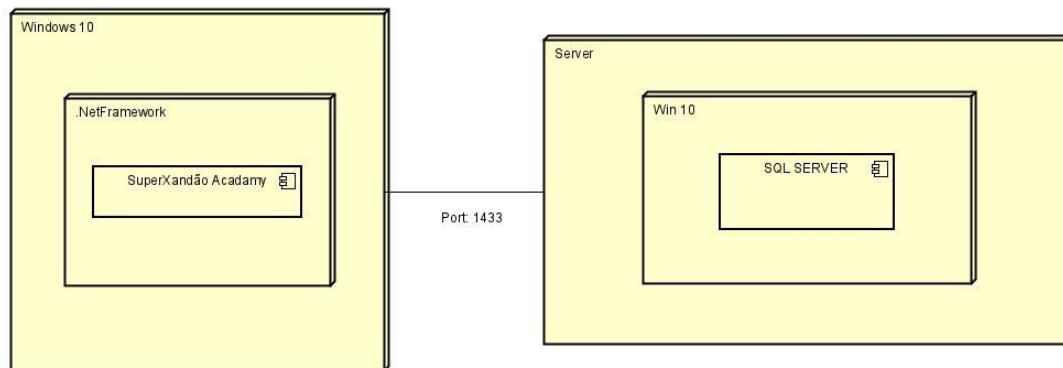


SuperXandão Academy

Descrição:

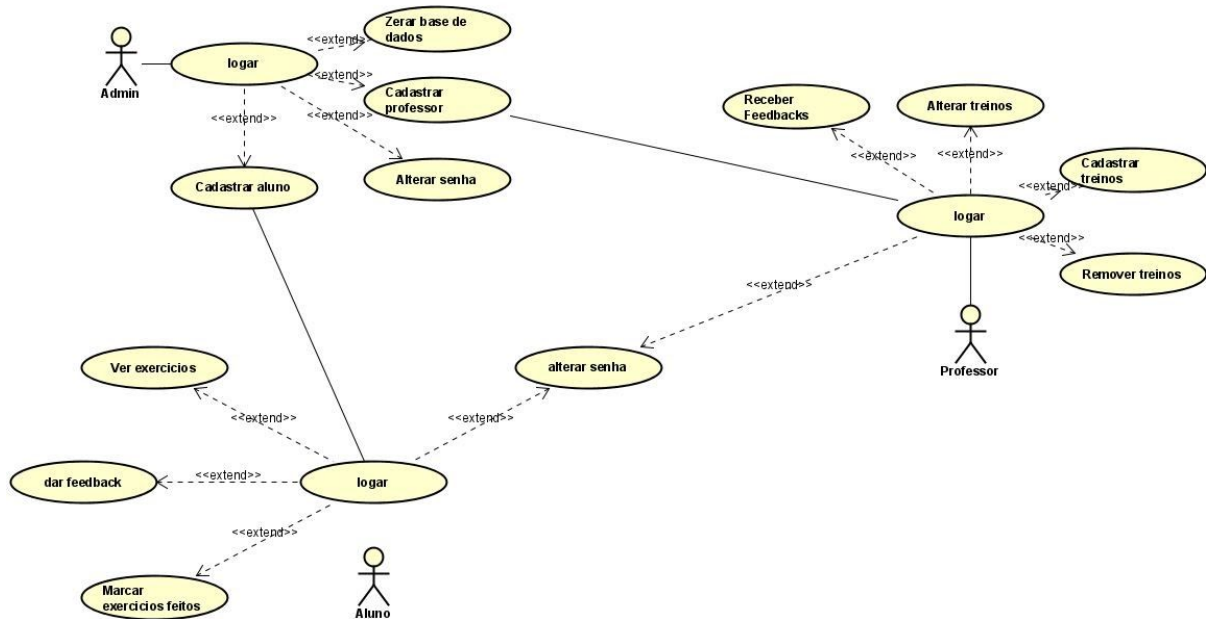
o SuperXandão Academy é um software para gerenciamento de exercícios entre professor e aluno, onde o professor deve cadastrar exercícios para alunos, informando o nome do exercício, a data de início, a data de término, os dias das semanas e a quantidade que deve ser feita. Já os alunos devem visualizar os exercícios, marcá-los como feitos e enviar feedbacks, tudo isso sendo gerenciado pelo administrador como: cadastro de professores e alunos, alteração de senha e limpeza do banco de dados.

Diagrama de implementação:



O sistema SuperXandão Academy deve ser executado em qualquer computador com acesso a internet banda larga e um sistema operacional windows 10, fazendo requisição a um servidor SQL Server 2019 Express através da porta 1433.

Diagrama de caso de uso:



Requisitos funcionais

RF01 – Solicitar login:

O sistema deverá solicitar o login para o aluno, professor e administrador.

RF02 – Trocar senha admin:

O sistema deve exigir a troca da senha do administrador logo no primeiro acesso.

RF03 – Cadastro:

O sistema deve permitir que o administrador cadastre novos alunos e professores.

RF04 – Deletar tudo:

O sistema deve dar a possibilidade do administrador zerar a base de dados.

RF05 – Trocar senha:

Caso algum aluno ou professor esqueça a senha o sistema deve dar a possibilidade do administrador alterar a senha.

RF06 – Cadastro de exercícios:

O sistema deve permitir que o professor possa cadastrar novos exercícios para seus respectivos alunos.

RF07 – Vincular professor e alunos:

O sistema deve possibilitar ao professor a vinculação de si mesmo com um novo aluno.

RF08 – Ver feedbacks:

Caso o professor queira o sistema deve mostrar os feedbacks dos alunos.

RF09 – Remover treinos:

Caso o professor necessite o sistema deve possibilitar a exclusão de um treino já cadastrado.

RF10 – Alterar treinos:

Caso o professor necessite o sistema deve possibilitar a alteração de um treino já cadastrado.

RF11 – Ver treinos:

O aluno deve conseguir visualizar seus treinos como sua data, dia da semana e a quantidade de repetições.

RF12 – Marcar treinos como feito:

Ao terminar um treino o aluno deve conseguir marcar o treino como feito.

RF13 – Enviar feedbacks:

O aluno deve enviar feedbacks sobre os exercícios para seu professor.

RF14 – Alteração de senhas:

Caso o usuário lembre sua o sistema deve permitir alteração da senha desde que ele insira a senha antiga.

Requisitos não funcionais**RNF01 - Compatibilidade:**

O sistema deve ser executado sobre o sistema operacional Microsoft Windows 10 e seu código será desenvolvido utilizando a linguagem de programação C#.

RNF02 - Disponibilidade:

O programa deve estar disponível sempre que um usuário tentar efetuar o login durante os horários atendidos pela própria academia.

RNF03 - Usabilidade:

O usuário terá possibilidades diferentes de escolhas a partir do seu login, sendo diferenciado por administrador, professor e aluno.

RNF04 - Segurança:

O programa deve conter métodos para garantir o sigilo das credenciais dos usuários, contendo uma hash sha256 para criptografia das senhas.

RNF05 - Padrão:

O programa deve seguir o padrão MVC (Model-View-Controller).

RNF06 - Desempenho:

O programa deve ser executado visando a melhor experiência de uso ao usuário.

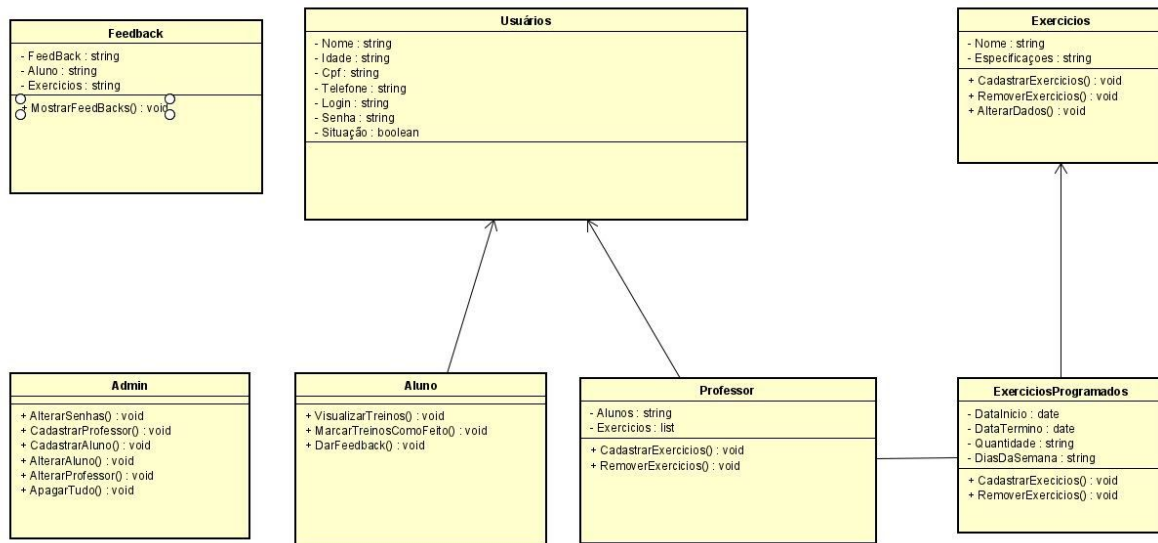
RNF07 - Facilidade de uso:

O programa deve possuir uma interface intuitiva e de fácil aprendizado, de modo que o usuário consiga compreender após uma breve explicação sobre o funcionamento do software.

RNF08 - Velocidade:

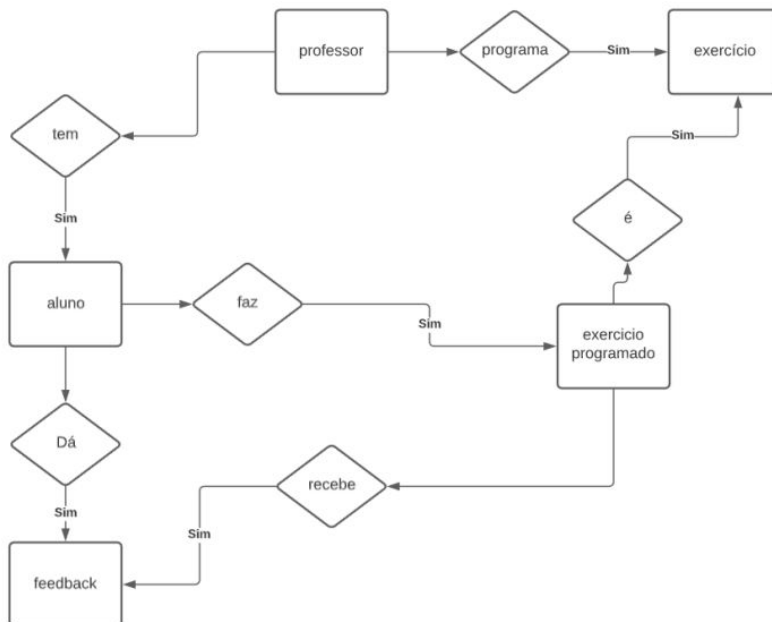
As operações realizadas no banco de dados são efetuadas em tempo real e atualizando o programa.

Diagrama de classe:



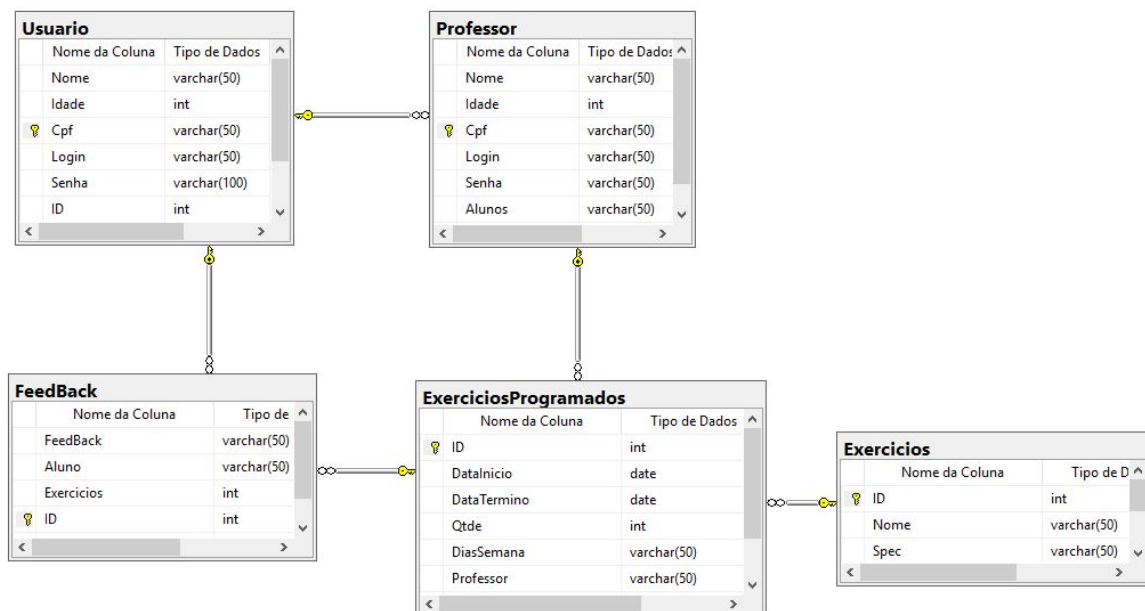
A construção das classes devem seguirem o padrão MVC (Model, View, Controller), onde Aluno e Professor herdam da classe Usuário, o professor pode programar um exercício programado que herda de Exercício, e o aluno da feedbacks para o professor através da classe Feedback.

Diagrama entidade relacionamento:



- Os alunos devem fazer os exercícios programados;
- O exercício programado é um exercício;
- O professor programa um exercício;
- O professor tem um aluno;
- O aluno dá um feedback;
- O exercício programado recebe um feedback.

Diagrama relacional:



O banco de dados deve possuir 5 tabelas (Usuario, Professor, FeedBack, ExerciciosProgramados, exercicios), sendo a tabela usuário responsável por armazenar os dados do administrador e do aluno, a tabela Exercícios contém todos os exercícios que estão disponíveis na academia, a tabela ExercíciosProgramados deverá conter os exercícios que já foram programados pelo professor e tabela FeedBack armazena os feedbacks do aluno.

Conclusão:

O sistema foi desenvolvido utilizando a linguagem de programação c# com windows forms acessando um banco de dados Sql Server 2019 express hospedado em um computador windows 10 o prazo para execução do mesmo foi de duas semanas.