**Dissertação do Projeto "SeaGo"**

**Introdução**

No contexto atual da tecnologia de software, a eficiência no desenvolvimento, implantação e escalabilidade de aplicações web se tornou fundamental. O projeto "SeaGo" visa abordar esses desafios utilizando tecnologias modernas como .NET para desenvolvimento back-end e Docker para contêinerização, garantindo uma solução robusta, escalável e fácil de manter.

**Objetivos**

O projeto "SeaGo" tem como principais objetivos:

1. **Desenvolvimento Eficiente**: Utilizar o framework .NET para desenvolver uma aplicação web que seja eficiente em termos de performance e segurança.
2. **Facilidade de Implantação**: Implementar Docker e Docker Compose para simplificar a implantação da aplicação em qualquer ambiente, seja desenvolvimento, teste ou produção.
3. **Escalabilidade**: Garantir que a aplicação possa ser facilmente escalada para lidar com aumentos no volume de usuários e dados sem degradação significativa no desempenho.
4. **Manutenção Simplificada**: Utilizar contêineres para encapsular a aplicação e suas dependências, tornando a manutenção e atualizações mais gerenciáveis.

**Solução Proposta**

**Arquitetura da Aplicação**

A aplicação "SeaGo" é construída utilizando o ASP.NET Core como o principal framework de desenvolvimento. O ASP.NET Core foi escolhido por sua performance, segurança e por ser amplamente suportado na comunidade. A aplicação segue uma arquitetura de microserviços, com cada serviço executando uma função específica, facilitando assim a manutenção e a escalabilidade.

**Contêinerização com Docker**

Docker é utilizado para contêinerizar a aplicação, o que permite que ela seja executada de forma consistente em qualquer ambiente. O Docker Compose é empregado para definir e executar múltiplos contêineres como um único serviço. No Dockerfile, a aplicação é configurada para rodar com um usuário não privilegiado para aumentar a segurança, e o ambiente é definido para produção, garantindo otimizações de desempenho.

**Persistência de Dados**

A persistência de dados é tratada através de volumes Docker, que garantem que os dados sejam mantidos mesmo quando os contêineres são reiniciados ou destruídos. Isso é crucial para aplicações em produção onde a perda de dados pode ter consequências significativas.

**Demonstração e Documentação**

O projeto inclui um vídeo de demonstração que mostra o processo de clone do repositório, construção do contêiner, inicialização da aplicação e persistência de dados. A documentação completa está disponível no repositório GitHub, facilitando a compreensão e replicação do projeto por outros desenvolvedores.

**Conclusão**

O projeto "SeaGo" demonstra como tecnologias modernas como .NET e Docker podem ser utilizadas para criar aplicações web robustas, escaláveis e fáceis de manter. Através da contêinerização, a aplicação pode ser facilmente implantada em qualquer ambiente, reduzindo significativamente os problemas comuns associados à disparidade de ambientes entre desenvolvimento e produção.