

Manual Técnico - EasyFitHub

Objetivo

O EasyFitHub procura fornecer uma solução abrangente para a gestão de ginásios, oferecendo um ambiente online onde os ginásios, seus clientes e funcionários possam interagir e realizar várias operações de forma eficiente. Os principais objetivos da plataforma incluem:

Facilitar a gestão dos ginásios: Permitir que os proprietários de ginásios gerenciem facilmente suas instalações, incluindo a configuração de informações do ginásio, atribuição de funcionários e gestão de inventário. Simplificar o processo de reserva e pagamento: Oferecer aos clientes uma experiência conveniente ao consultar planos de associação e efetuar pagamentos online de forma segura. Fornecer análises e estatísticas úteis: Apresentar métricas e relatórios detalhados que ajudem os ginásios a entender melhor o desempenho do negócio e tomar decisões informadas para otimização. Melhorar a experiência do utilizador: Garantir uma interface amigável e intuitiva para todos os utilizadores, tornando a navegação e o uso da plataforma fácil e agradável.

Índice

- Introdução
- Arquitetura do Sistema
- Tecnologias Utilizadas
- Estrutura de Diretórios e Organização do Código
- Base de Dados
- Integrações Externas
- Testes e Qualidade de Código

Introdução

O manual técnico do projeto EasyFitHub oferece uma visão detalhada da arquitetura, tecnologias utilizadas e outros aspectos técnicos relevantes do sistema. Este documento destina-se a desenvolvedores, administradores de sistemas e outros profissionais de tecnologia envolvidos no desenvolvimento, implantação e manutenção da plataforma EasyFitHub.

Arquitetura do Sistema

A arquitetura do sistema EasyFitHub é baseada em uma abordagem moderna e escalável, projetada para atender às demandas de um ambiente de ginásios em constante evolução. A seguir estão os principais componentes da arquitetura:

- **Frontend:** Desenvolvido utilizando ASP.NET para a criação da interface do utilizador.
- **Backend:** Implementado em ASP.NET, responsável pelo processamento lógico e comunicação com o base de dados.
- **Base de Dados:** Utiliza o Microsoft SQL Server para armazenamento de dados, garantindo segurança e desempenho.
- **Azure Cloud:** Implantação na plataforma Azure Cloud para garantir escalabilidade, disponibilidade e segurança.

A arquitetura segue um modelo de cliente-servidor, onde o cliente interage com o frontend para acessar as funcionalidades oferecidas pela plataforma, e o backend realiza o processamento e a comunicação com o base de dados.

Tecnologias Utilizadas

O desenvolvimento do projeto EasyFitHub envolveu a utilização das seguintes tecnologias e ferramentas:

- **ASP.NET:** Framework utilizado tanto para o frontend quanto para o backend do sistema, oferecendo uma solução completa para o desenvolvimento web.
- **Microsoft SQL Server:** Base de dados relacional utilizado para armazenamento seguro e eficiente de dados.
- **Azure Cloud:** Plataforma de nuvem utilizada para implantação e hospedagem do sistema, oferecendo escalabilidade e disponibilidade.
- **C#:** Linguagem de programação utilizada para o desenvolvimento do backend do sistema, fornecendo recursos avançados de programação orientada a objetos.
- **HTML/CSS/JavaScript:** Tecnologias web utilizadas para a criação da interface do utilizador e aprimoramento da experiência do utilizador.
- **Visual Studio:** IDE (Integrated Development Environment) utilizada para desenvolvimento, depuração e teste do projeto.
- **BitBucket:** Sistema de controle de versão utilizado para gerenciamento do código-fonte e colaboração entre os membros da equipe de desenvolvimento.

Essas tecnologias foram escolhidas com base em sua robustez, desempenho e suporte à escalabilidade, garantindo uma experiência confiável e eficiente para os utilizadores da plataforma EasyFitHub.

Estrutura das Pastas e Organização do Código

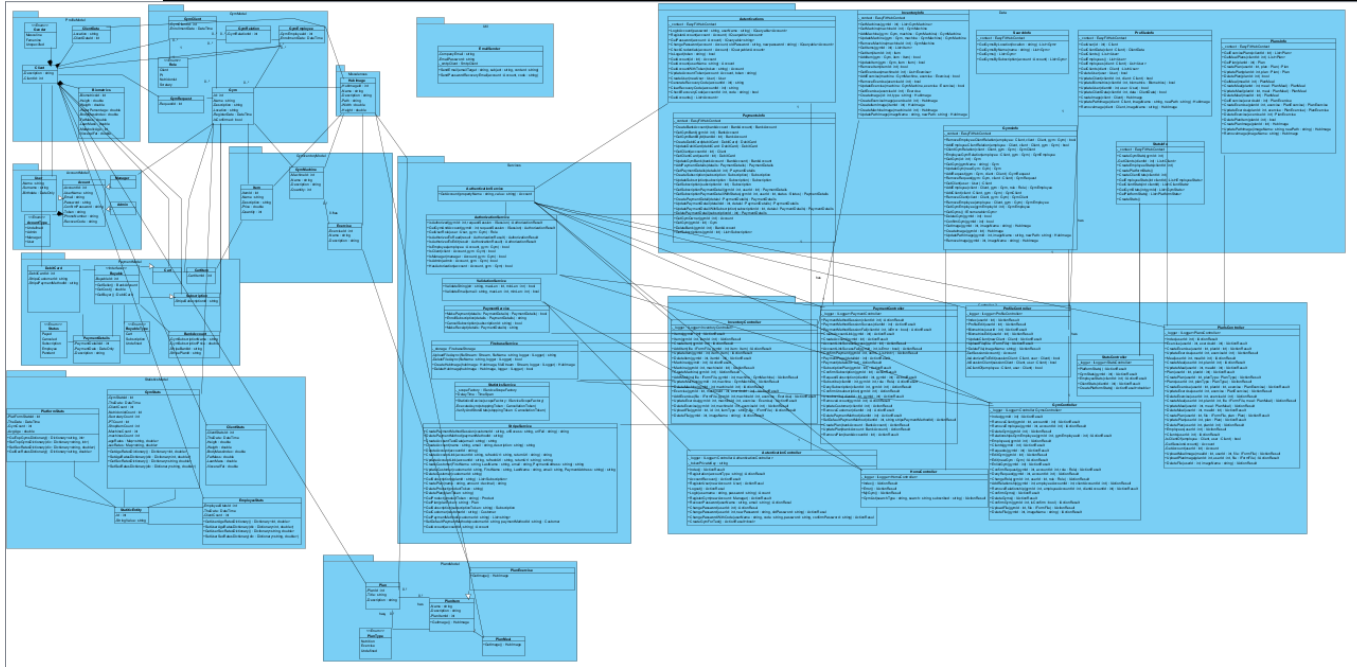
A estrutura das pastas e organização do código do projeto EasyFitHub segue as melhores práticas de desenvolvimento, mantendo uma separação clara das camadas do padrão MVC (Model-View-Controller), bem como uma pasta para os serviços, uma para a comunicação com a base de dados e outro projeto para testes. A seguir está a descrição da estrutura:

- **MVC:**
 - **Models:** Contém as classes que representam os modelos de dados do sistema, incluindo entidades da base de dados, DTOs (Data Transfer Objects), etc.
 - **Views:** Responsável pela apresentação e renderização das interfaces do utilizador, implementadas em arquivos .cshtml.
 - **Controllers:** Responsável pelo controle do fluxo de dados entre o modelo e a visualização, implementando a lógica de negócios e manipulando requisições HTTP.
- **Data:**
 - Contém as classes responsáveis pela ligação e manipulação de dados com a base de dados.
- **Serviços:**
 - Contém as classes responsáveis pela implementação de serviços do sistema que podem ser partilhados por varios componentes, como autenticação e ligação com os APIs.

- **Projeto de Testes:**

- Projeto separado dedicado à implementação de testes automatizados para garantir a qualidade e a robustez do código-fonte.

Diagramas de classes



Base de Dados

O EasyFitHub utiliza o Microsoft SQL Server como sistema para gerir a base de dados. O acesso e manipulação dos dados são realizados através do Entity Framework, uma tecnologia de mapeamento objeto-relacional (ORM) que simplifica o acesso e manipulação dos dados no base de dados por meio de objetos .NET.

Integrações Externas

O EasyFitHub faz uso de integrações externas para estender sua funcionalidade e oferecer recursos adicionais aos utilizadores. Duas das principais integrações externas utilizadas são:

Stripe (API de Pagamentos)

O Stripe é uma plataforma de pagamentos online que permite às empresas aceitar pagamentos pela internet. O EasyFitHub integra a API do Stripe para processar transações de pagamento de forma segura e conveniente. Com a integração do Stripe, os clientes podem realizar pagamentos de subscrição e pagamentos na loja do ginásio diretamente através da plataforma EasyFitHub.

Firebase Storage (Armazenamento de Imagens)

O Firebase Storage é um serviço de armazenamento de objetos baseado em nuvem oferecido pelo Google Firebase. O EasyFitHub utiliza o Firebase Storage para armazenar e gerenciar imagens e outros arquivos de mídia associados aos ginásios, clientes e outros recursos do sistema. Isso permite que o EasyFitHub ofereça suporte para o armazenamento seguro e eficiente de imagens de perfil de utilizador, fotos de ginásios, etc.

Essas integrações externas adicionam valor à plataforma EasyFitHub, proporcionando uma experiência mais completa e abrangente para os utilizadores, ao mesmo tempo em que simplificam o processo de desenvolvimento e implementação de recursos específicos.

Testes e Qualidade de Código

A qualidade do código fonte e a confiabilidade do sistema são aspectos essenciais para o sucesso do projeto EasyFitHub. Para garantir isso, adotamos práticas sólidas de testes e garantia de qualidade durante todo o ciclo de vida do desenvolvimento de software.

Testes Automatizados

O EasyFitHub utiliza testes automatizados para validar o funcionamento correto das funcionalidades do sistema. Isso inclui testes unitários, testes de integração e testes de aceitação, implementados usando NUnit. Esses testes garantem que cada componente do sistema funcione conforme o esperado, isoladamente e em conjunto com outros componentes.