



Engenharia de Software 2018/2019

Especificação do Sistema – alto nível

Turma: 3
Grupo nº 4
nº170221080 César Nero
nº1702210881 David Afonso
nº170221085 Rúben Ferreira

Versões do Trabalho

Versão	Data	Autor	Descrição
1	05/11/2018	César, David, Rúben	Elaboração do Documento
2	10/02/2019	César, David, Rúben	Correção Final

1	SUMARIO EXECUTIVO	4
2	INTRODUÇÃO	4
3	DESENHO DE ALTO NÍVEL	4
3.1	ARQUITETURA GERAL	4
3.2	ARQUITETURA LÓGICA	4
3.3	DIAGRAMA DE CLASSES DE DESENHO	6
3.4	INTERFACE COM O UTILIZADOR	7
3.4.1	Introdução	7
3.4.2	Protótipo (caso exista) ou <i>mock-up</i> em alternativa	7
3.4.3	Normas	17
3.4.4	Diagrama Geral de Navegação	18
3.4.5	Matriz de acessos	19
3.5	PERSISTÊNCIA	19
3.5.1	Introdução	21
3.5.2	Modelo Relacional.....	21
3.6	ARQUITETURA FÍSICA	22
3.6.1	Introdução	22
3.6.2	Diagrama de Componentes	22
3.6.3	Diagrama de Instalação	23
3.7	NORMAS DE CODIFICAÇÃO DA APLICAÇÃO	23

1 SUMÁRIO EXECUTIVO

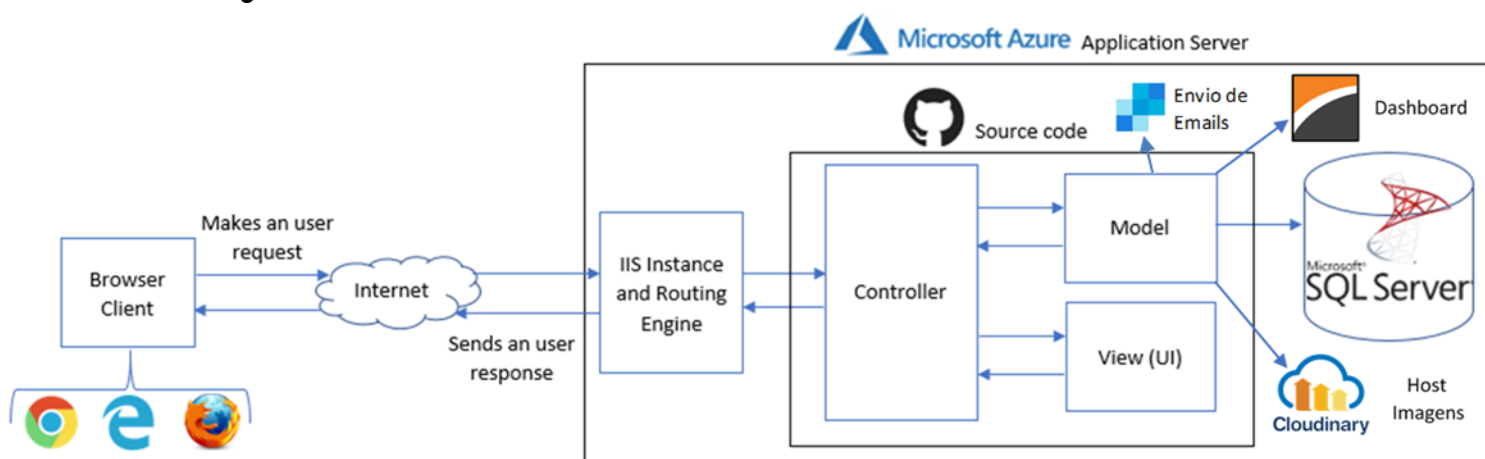
Este projeto tem como objetivo o desenvolvimento de um sistema de informação para um albergue animal, tendo como finalidade a gestão dos recursos internos, os animais e o acesso ao público de modo a promover as adoções e acolhimentos temporários.

2 INTRODUÇÃO

Este projeto foi realizado no âmbito da unidade curricular de Engenharia de Software, tendo como objetivo a necessidade de uma gestão e armazenamento de dados eficiente por parte do albergue animais.

3 DESENHO DE ALTO NÍVEL

3.1 ARQUITETURA GERAL

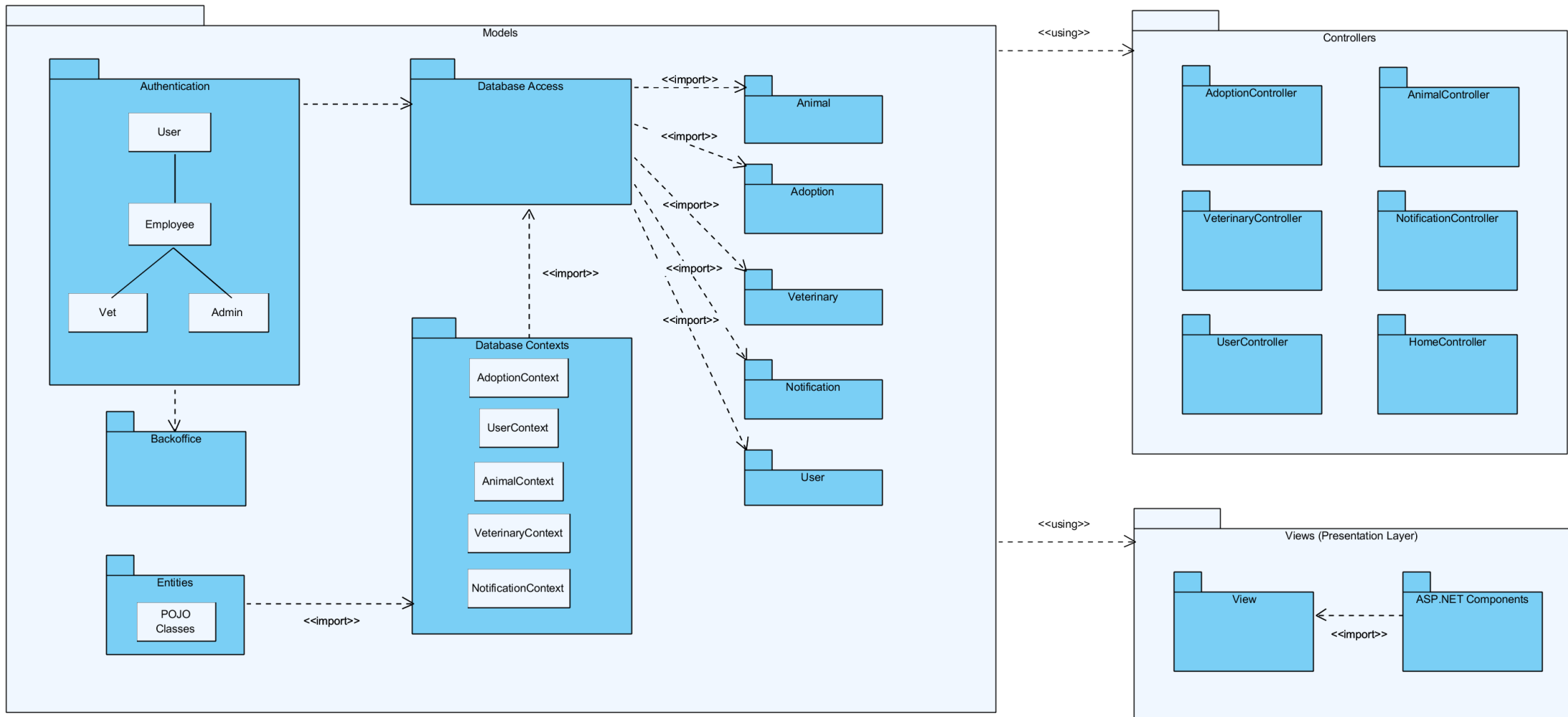


A aplicação tem como base uma arquitetura ASP.net Core com MVC com suporte a persistência de dados através de uma base de dados MS SQL. Todavia é necessário recorrer a outras tecnologias e/ou serviços de modo a reutilizar código/funcionalidades e diminuir custos. O “host” de imagens é realizado no serviço cloud chamado “Cloudinary” que permite a integração da sua API com ASP.net Core através da instalação de pacotes tipo NuGet.

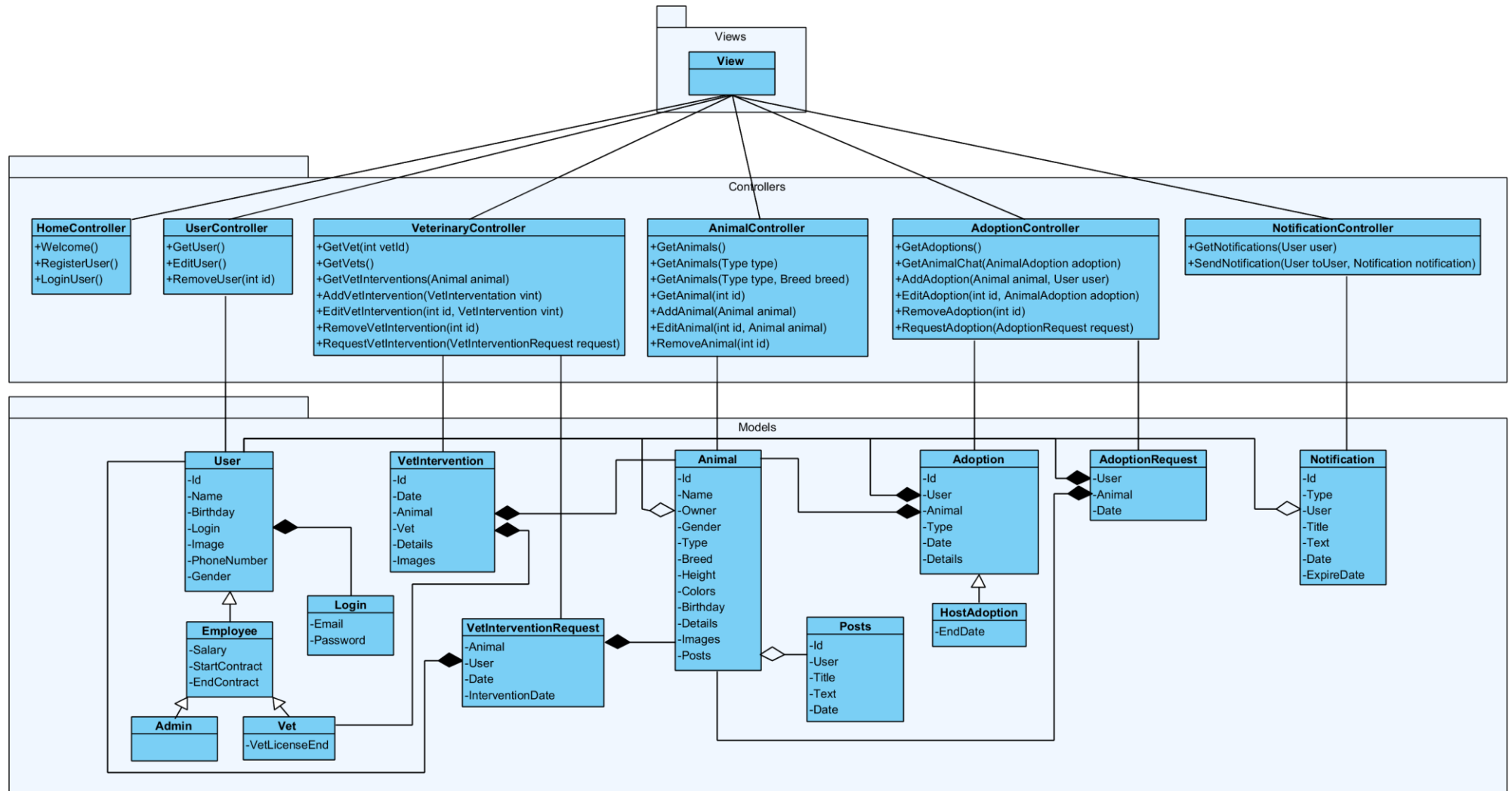
O envio de e-mails é realizado através da plataforma SendGrid, que permite enviá-los gratuitamente através dos seus servidores de e-mail. Já para a realização de estatísticas recorreu-se a uma ferramenta que permite a criação de *dashboards* da empresa DevExpress. O código fonte é alojado no GitHub de modo a facilitar o controlo de versões de software por vários membros. A aplicação estará em execução num serviço de publicação de software do tipo cloud, o Azure.

A aplicação então será consumida por browsers, tais como o Google Chrome, Edge e Firefox, de modo a chegar aos clientes finais.

3.2 ARQUITETURA LÓGICA



3.3 DIAGRAMA DE CLASSES DE DESENHO



3.4 INTERFACE COM O UTILIZADOR

3.4.1 Introdução

A interface desenvolvida procura ser o mais simples e acessível possível, não estando a descartar a qualidade visual em qualquer momento. Uma plataforma na qual qualquer utilizador consegue aceder ao pretendido de uma forma eficiente, cómoda e rápida.

3.4.2 Protótipo (caso exista) ou *mock-up* em alternativa

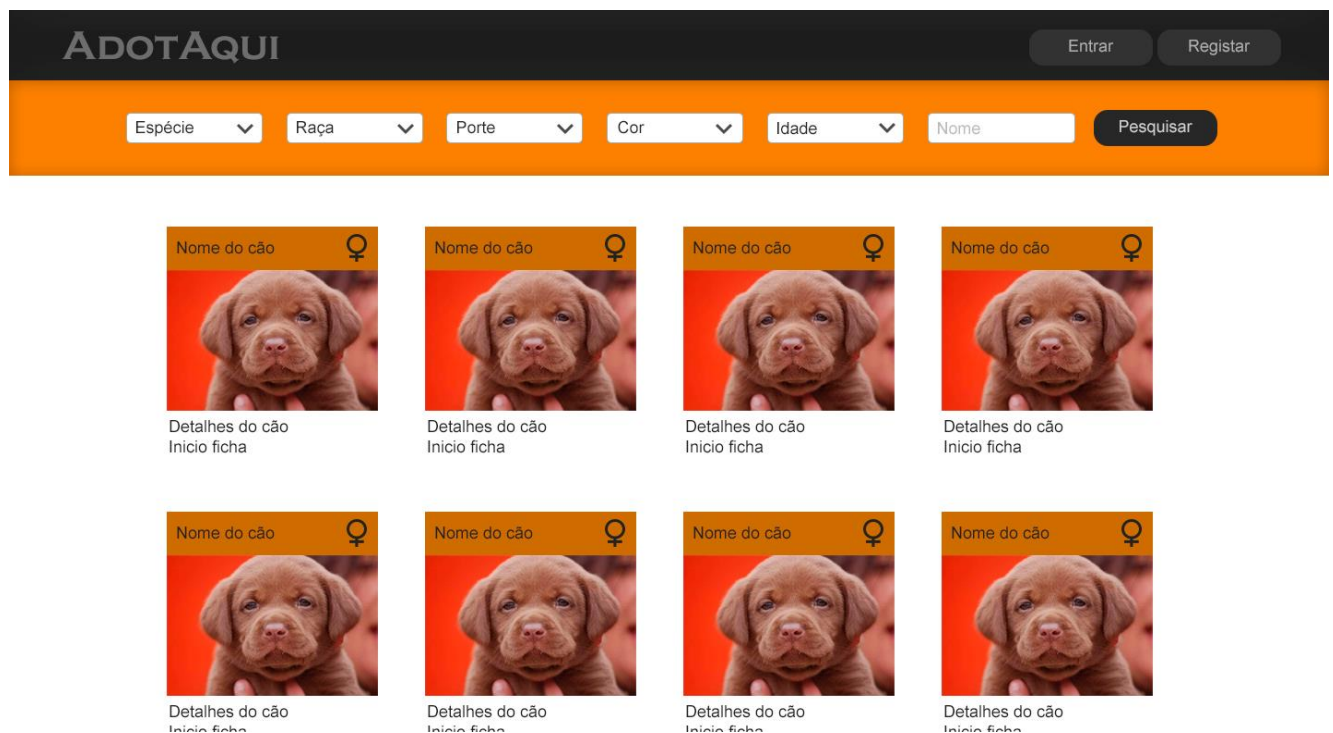


Figura 1 – Página inicial



ADOTAQUI

[Esqueceu-se da password?](#)

Figura 2 – Página de autenticação



ADOTAQUI

Figura 3 – Página de recuperação senha



ADOTAQUI

Dados pessoais

Registrar

Figura 4 – Página de registo



ADOTAQUI

Dados pessoais

Salvar Alterações

Retornar



Upload de foto

Utilizador exemplo

Ocultar no perfil

Figura 5 – Página de perfil



Figura 6 – Página de pesquisa utilizadores

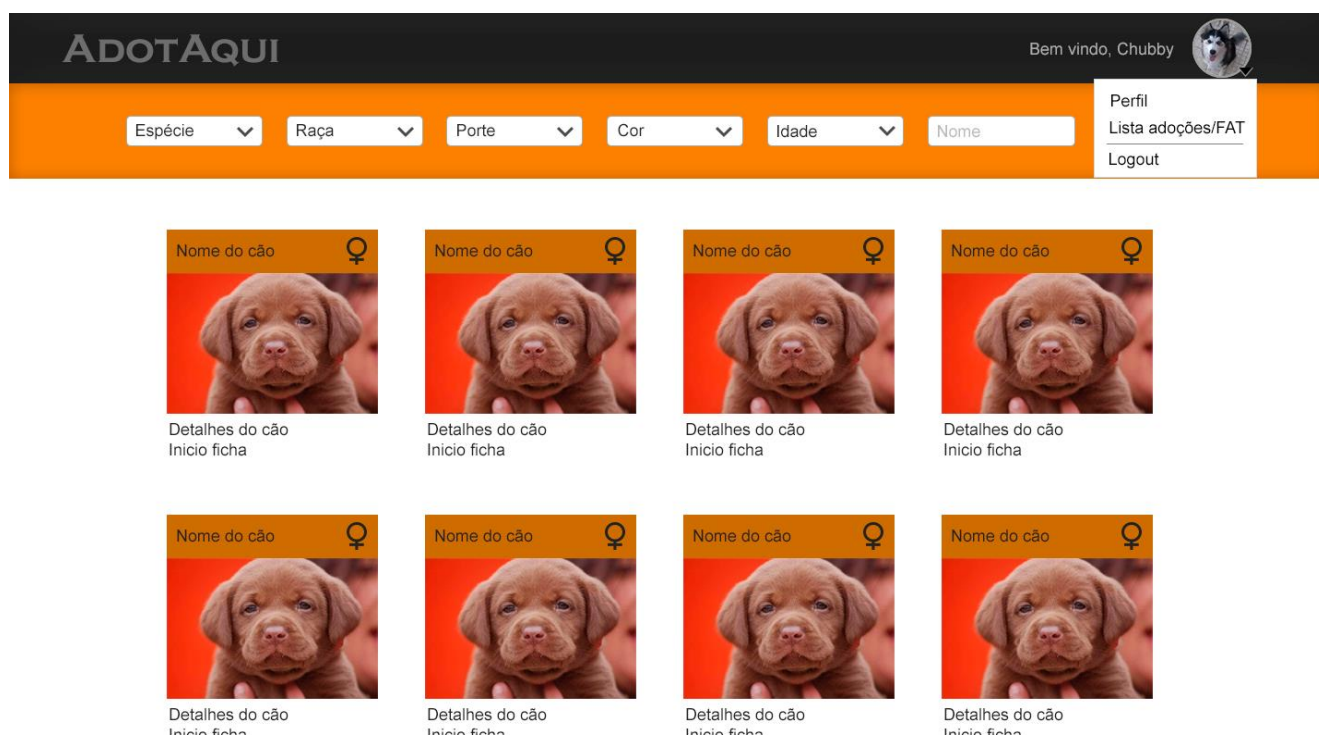


Figura 7 – Página de pesquisa animais sem ser administrador

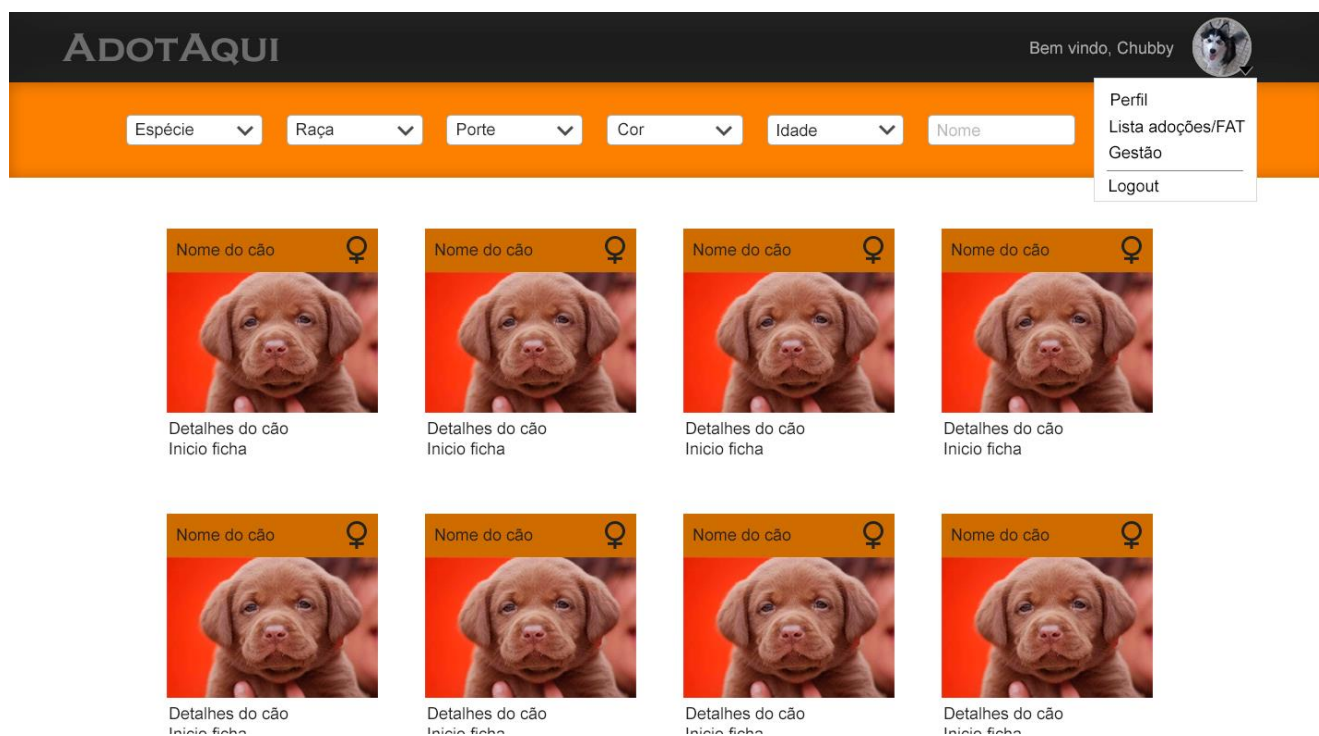


Figura 8 – Página com utilizador Administrador



Figura 9 – Página para ver adoções





ADOTAQUI

Nome

Telefone


Data nascimento

[Retomar](#)

[Editar informação](#)

Utilizador exemplo

Figura 10 – Página para alteração da informação perfil



Nome do cão ♀

Raça


Porte

Peso

Idade

Registrar


Consultar anexos



Candidatar a adoção

Sobre mim


Seu cachorrinho já lhe terá proporcionado muitas alegrias. Cuide para que ele tenha um final de vida feliz. Sempre que for possível, deixe que ele permaneça ao seu lado, pois este será, realmente, um dos poucos prazeres que lhe restarão na velhice. A grande despedida está próxima, e ele por instinto sabe disso. É natural que deseje a companhia daquele que aprendeu a amar e respeitar durante a sua vida. Não o abandone agora. Ele já não será aquele animal bonito de antes. Seu pelo começa a cair, seu caminho perdeu a elegância, a sua cabeça penderá cansada sobre suas patas. Somente seu olhar acompanhará os passos de seu dono.



Utilizador exemplo


Escrever comentário

Submeter



Utilizador exemplo

Este cão é o melhor cão de todos os cães
Parece-se mesmo com um cão!



Utilizador exemplo

Woof woof!


Editar

Apagar

Figura 11 – Página para inserção de opiniões



Figura 12 – Página administração aceitar adoção



Nome do cão ♀


Raça

Porte

Peso

Idade


Registrar

Consultar anexos 

Candidatar a adoção

Sobre mim


Seu cachorrinho já lhe terá proporcionado muitas alegrias. Cuide para que ele tenha um final de vida feliz. Sempre que for possível, deixe que ele permaneça ao seu lado, pois este será, realmente, um dos poucos prazeres que lhe restarão na velhice. A grande despedida está próxima, e ele por instinto sabe disso. É natural que deseje a companhia daquele que aprendeu a amar e respeitar durante a sua vida. Não o abandone agora. Ele já não será aquele animal bonito de antes. Seu pelo começa a cair, seu caminho perdeu a elegância, a sua cabeça penderá cansada sobre suas patas. Somente seu olhar acompanhará os passos de seu dono.



Utilizador exemplo


Escrever comentário

Submeter



Utilizador exemplo

Este cão é o melhor cão de todos os cães
Parece-se mesmo com um cão!



Utilizador exemplo

Woof woof!

Editar

Apagar

Figura 13 – Página ficha de um animal



Figura 14 – Página de intervenções médicas



Figura 15 – Página de notificações

3.4.3 Normas

A interface procura desenvolver um ambiente acolhedor perante o utilizador. Para tal, é utilizado um leque de cores quentes, desde tons alaranjados a cores mais escuras de modo a causar um contraste agradável à leitura por parte do utilizador.

- Header – #222222 com gradiente alvo a #000000
- Letras no header – #535353
- Barra de menu e botões vários – #e27f00
- Background – #ffffff
- Borders – #a4aaad

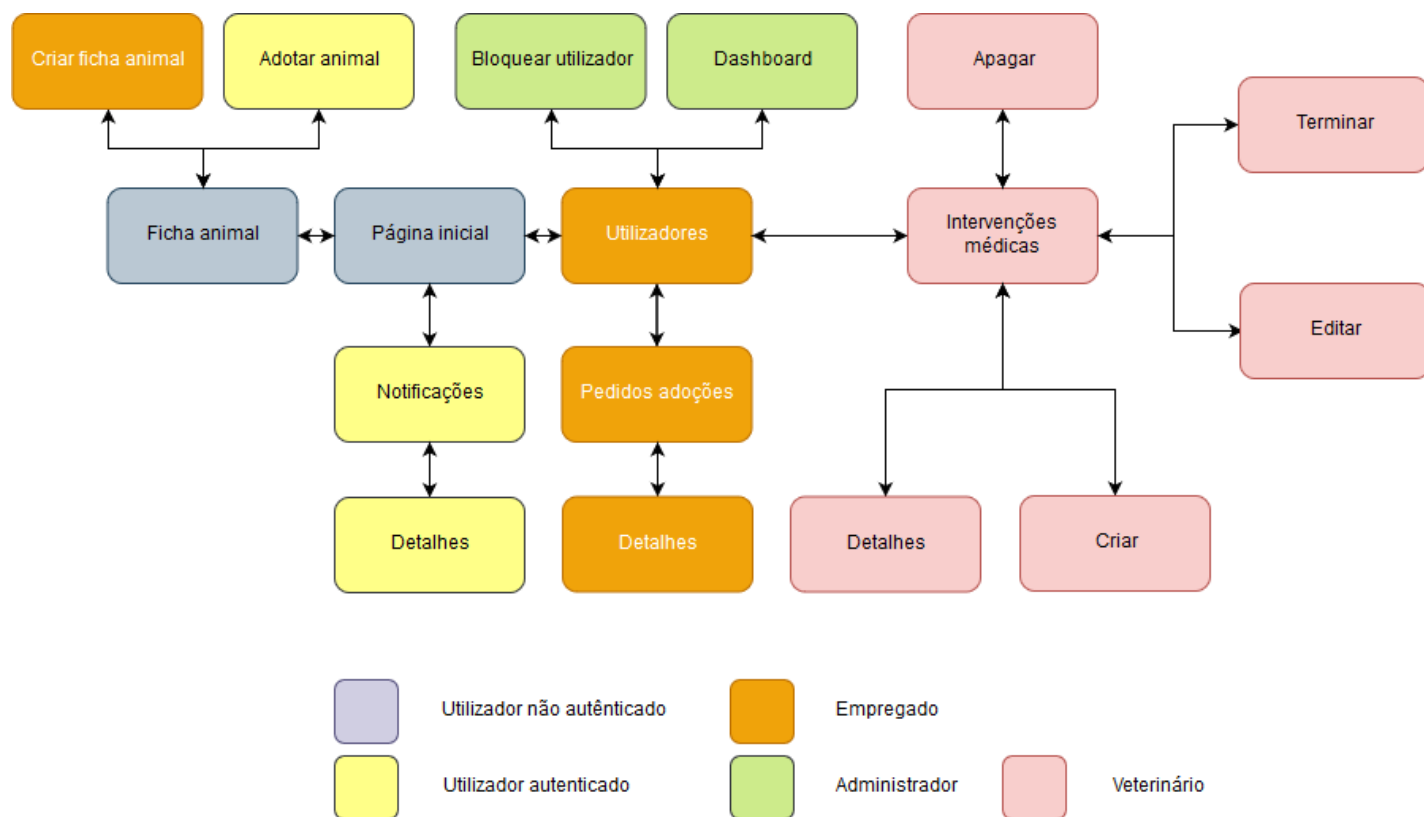
O texto da interface é constituído por:

- Texto conteúdo – Arial Regular 18px, #000000
- Texto máscara – Arial Regular 18px, #808080
- Texto de títulos – Arial Black 24px. #000000

O logótipo é construído num estilo simples, pouco detalhe e uso serifas de facilitar a leitura ao utilizador:

- Copperplate Gothic Bold Regular, #000000

3.4.4 Diagrama Geral de Navegação



3.4.5 Matriz de acessos

	Registo	Login	Recuperar Conta	Consultar Perfil	Alterar Perfil	Pesquisar Perfis	Consultar Adoções do Utilizador	Banir Utilizador	Enviar Notificação
Convidado	x	x	x						
Utilizador		x	x	x	x				
Empregado		x	x	x	x	x	x		x
Veterinário		x	x	x	x	x	x		x
Administrador		x	x	x	x	x	x	x	x

	Alterar Tipo Utilizador	Registar Animal	Consultar Ficha Animal	Modificar Ficha Animal	Carregar Anexos à Ficha Animal	Pesquisar Animais Adoção	Realizar Pedido Adoção	Aceitar Pedido Adoção
Convidado			x			x		
Utilizador			x			x	x	
Empregado		x	x	x	x	x	x	x
Veterinário		x	x	x	x	x	x	x
Administrador	x	x	x	x	x	x	x	x

Especificação do Sistema – Engenharia de Software

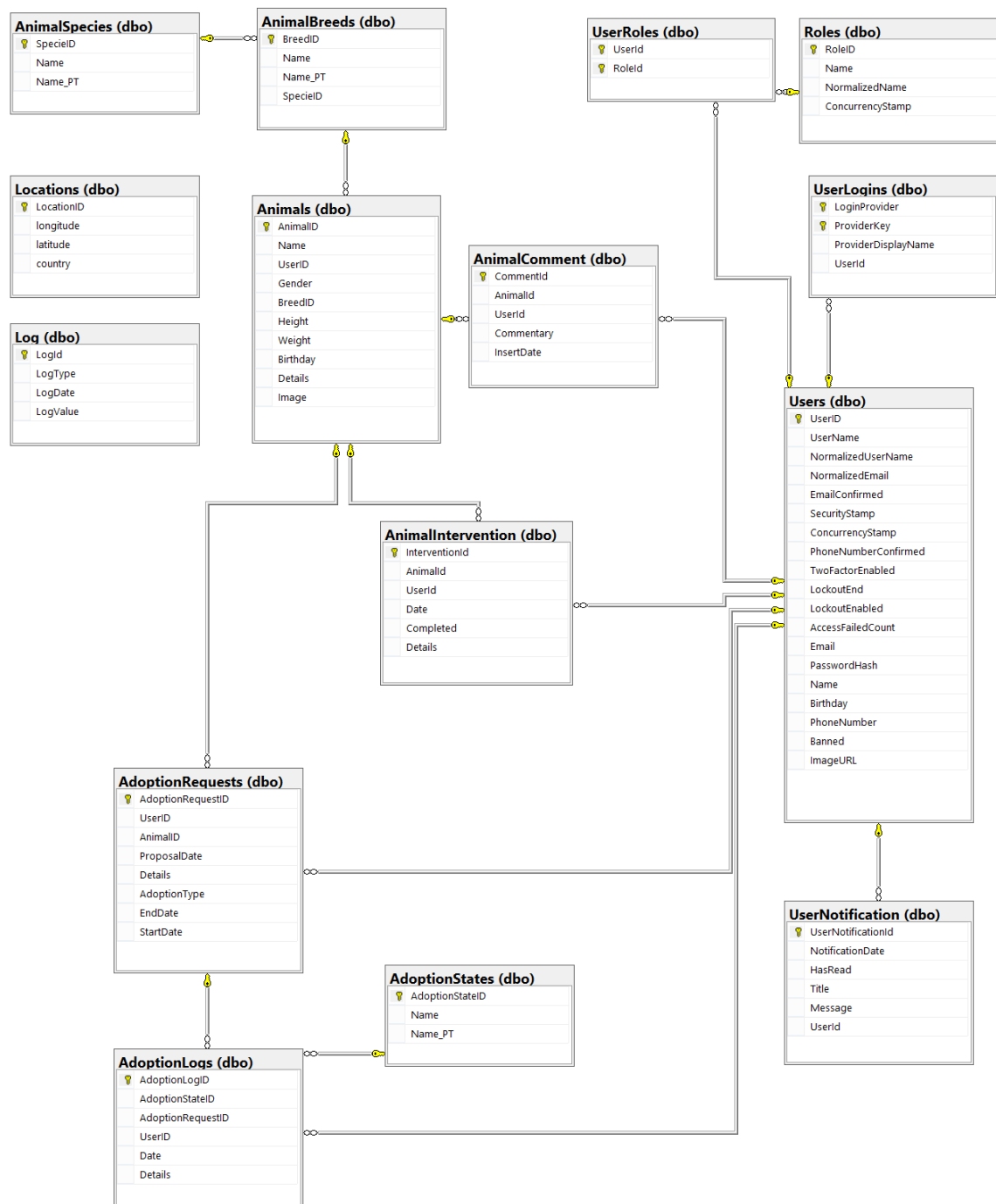
	Registrar Comentário sobre Animal	Consultar Comentários sobre Animal	Agendar Intervenção Médica	Registrar Intervenção Médica	Agendar Notificação
Convidado	x				
Utilizador	x				x
Empregado	x	x	x	x	x
Veterinário	x	x	x	x	x
Administrador	x	x	x	x	x

3.5 PERSISTÊNCIA

3.5.1 Introdução

Em relação à persistência dos dados, foi optado pela utilização do MSSQL 2017. Uma base de dados que dá suporte a toda a infraestrutura da plataforma de forma rápida e organizada. Através disto é possível guardar toda a informação relacionada com os utilizadores e animais.

3.5.2 Modelo Relacional



3.6 ARQUITETURA FÍSICA

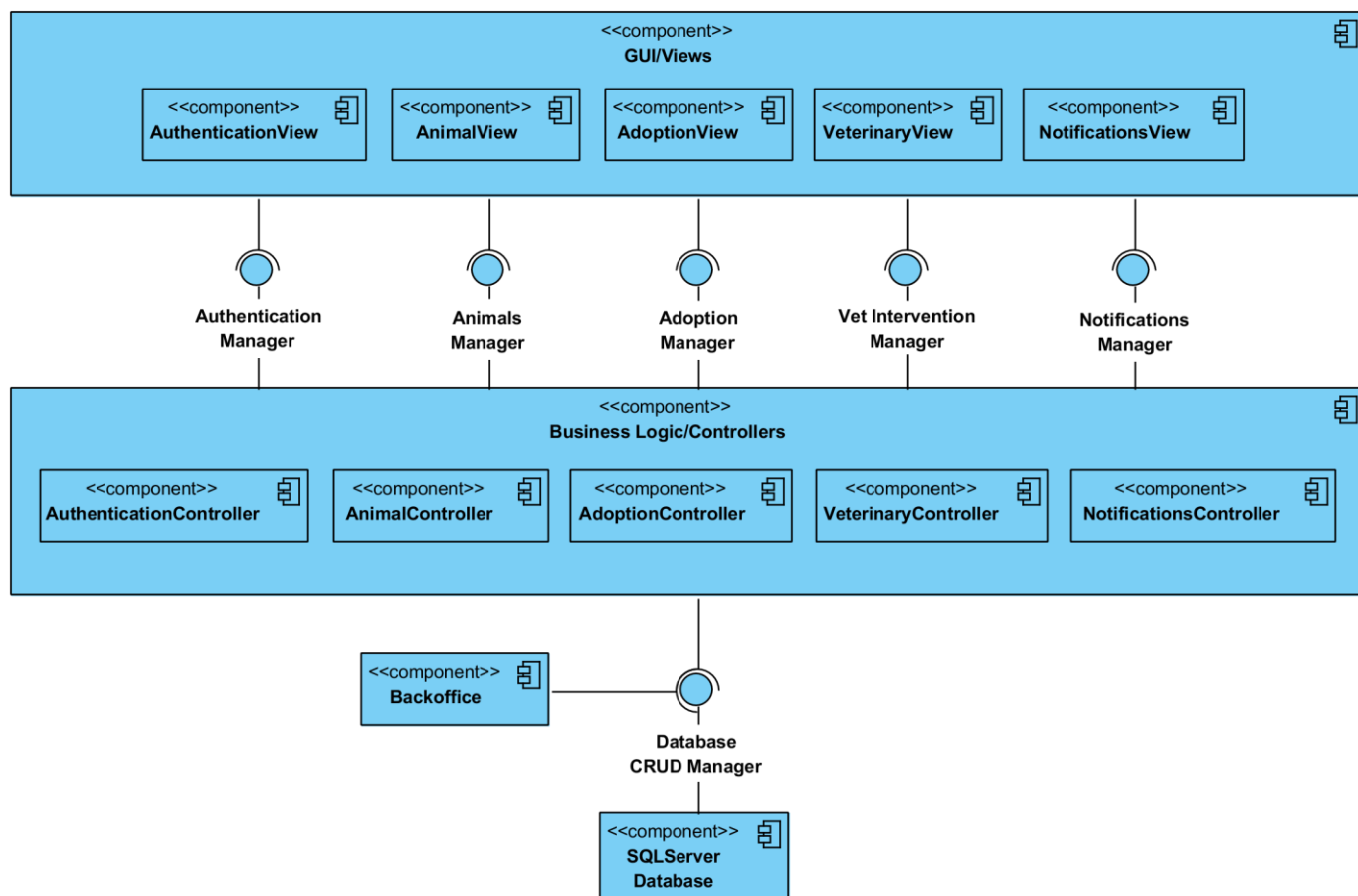
3.6.1 Introdução

De modo a desenvolver a estrutura de dados do sistema começou-se por idealizar os “use-cases” e relações das entidades da aplicação. Esta análise permitiu-nos a construção do modelo entidade-relação. A partir deste modelo inicial mais a atribuição de campos às entidades constituintes, procedemos à normalização da estrutura de dados para evitar redundância de dados, obter um desempenho aceitável e permitir maior escalabilidade do sistema.

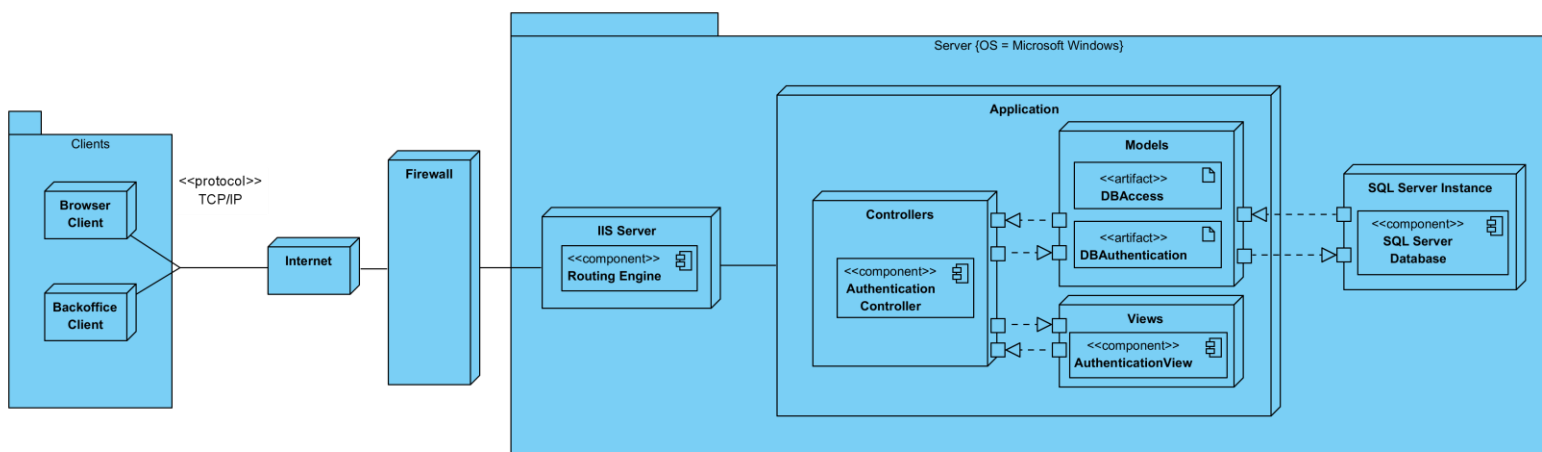
Com a estrutura de dados já normalizada existe a facilidade de implementá-la em qualquer sistema de gerenciamento de base de dados. De modo a corresponder aos requisitos ambientais da aplicação será utilizado o SGBD MS SQL Server 2017, com normas de codificação consideradas boas práticas de modo a maximizar o desempenho, tais como: utilização índices, utilização da cláusula *join* em alternativa a *sub-queries*, utilização de *schemas*, entre outras...

A nível físico da camada de rede também é necessária a configuração de uma *firewall* de forma a manter a aplicação segura e minimizar o tráfego de dados assim como as latências de *responses* e *requests*.

3.6.2 Diagrama de Componentes



3.6.3 Diagrama de Instalação



3.7 NORMAS DE CODIFICAÇÃO DA APLICAÇÃO

Como normas de codificação da aplicação foi utilizada a convenção de codificação em C# disponibilizada pela Microsoft em ["https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/inside-a-program/coding-conventions"](https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/inside-a-program/coding-conventions) tendo como finalidade uma aparência consistente para o código de modo a que qualquer utilizador consiga facilmente entender o código e a sua alteração/manutenção.

Ainda aliada a estas convenções disponibilizadas pela Microsoft, disponibilizadas em ["https://blogs.msdn.microsoft.com/aspnetue/2010/09/17/best-practices-for-asp-net-mvc/"](https://blogs.msdn.microsoft.com/aspnetue/2010/09/17/best-practices-for-asp-net-mvc/), também é utilizada as melhores práticas de desenvolvimento para ASP.NET MVC nas quais referem algumas coisas como a colocação de todo o modelo de negócio e validações nos *Models*, a colocação de todo o HTML nas *Views* ou a consideração de chamadas assíncronas nos *Controllers* para pedidos de longa duração.