**Tecnologias Usadas**

**HTML**

Criada por Tim Berners Lee na década de 90, HTML é uma linguagem de marcação, e está presente em praticamente todas as páginas da *web.* Se baseia em *tags* escritas de forma estruturada e hierárquica, interpretada pelos *browsers* modernos (Carey Wodehouse, 2015).

O HTML é a linguagem responsável por dar estrutura a uma página *web*. Está atualmente em sua versão HTML5, contando com suporte a *tags* multimídia e melhor especificação de separações de seções na página (w3schools.com, 2015). As especificações da linguagem são controladas pela W3C (World Wide Web Consortium), um consórcio internacional responsável por estabelecer os padrões de conteúdo para a internet.

No projeto Jarbas, o HTML é responsável pela estruturação do conteúdo na aplicação *front-end.* É utilizado juntamente com a tecnologia Ionic (vide abaixo).

**CSS**

Cascading Style Sheets, ou CSS, é uma linguagem *web* com finalidade de adicionar estilos em páginas HTML. É um conjunto de regras composta por seletores e blocos de seleção, por qual são declarados uma variedade de estilos definidos. Com CSS é possível controlar a posição de elementos HTML na página, definir *backgrounds*, fontes, alinhamentos, etc (MATERA Systems, 2012).

Suas especificações são definidas pela W3C (World Wide Web Consortium) e está atualmente em sua terceira versão, CSS3.

No projeto Jarbas, o CSS é responsável pela estilização do conteúdo na aplicação front-end. É utilizado juntamente com a tecnologia Ionic (vide abaixo).

**JavaScript**

JavaScript é uma linguagem interpretada, de tipagem fraca e dinâmica, com suporte a diversos paradigmas de programação - orientação a objetos, protótipo funcional, entre outras. Implementada por todos navegadores *web* modernos, considerado a *língua franca* do desenvolvimento *client-side*. É baseada na especificação ECMAScript, padronizada pela Ecma International (MDN web docs, 2017).

Desenvolvida inicialmente na Sun Microsystems por Brendan Eich, é hoje uma das linguagens de programação mais utilizadas no mundo, com suporte de corporações como Google (desenvolvedora da engine V8 para JavaScript) e Fundação Mozilla (Eduardo Mustafa, 2016).

É responsável por dar funcionalidade e dinamismo às páginas *web*. No projeto Jarbas, o JavaScript é utilizado em sua forma TypeScript (um *subset* JavaScript desenvolvido pela Microsoft) em conjunto com a *framework* AngularJS (vide abaixo) para realizar requisições HTTP ao servidor, trabalhar com dados e controlar aspectos da interface gráfica.

**AngularJS**

O AngularJS é um *framework* *open-source*, mantido pela Google, que auxilia no desenvolvimento de *single-page applications* e *web apps*. É considerado pela Google como seguindo o padrão MVW (*Model View Whatever*). O *Whatever* significa que o AngularJS pode ser enquadrado nos seguintes padrões, MVC (*Model View Controller*), MVVM (*Model View View Model*) e MVP (*Model View Presenter*) (Guilherme Vicente, 2014).

O Angular encontra-se atualmente na versão Angular 4, entretanto ainda contando com suporte às versões legadas AngularJS.

No projeto Jarbas, a versão utilizada é o AngularJS (versão 1) – por razões de familiaridade da equipe de desenvolvimento e estabilidade do *framework*. É responsável pela camada de exibição e controle das *views* da aplicação cliente.

**Ionic**

O Ionic é um *framework* destinado ao desenvolvimento de aplicações *mobile* híbridas. Construído em cima das plataformas AngularJS e Apache Cordova (com uso do Android SDK), o *framework* prove diversos componentes e serviços para a construção de aplicações *mobile* que utilizam tecnologias *web*, como HTML5 e CSS (Ionic, 2017).

No projeto Jarbas, Ionic é responsável pelo auxílio e construção das interfaces da aplicação cliente em conjunto com AngularJS.

**C# (C Sharp)**

C# (lê-se CSharp) é uma linguagem de programação de alto-nível, compilada, de tipagem estática e/ou dinâmica forte, parcialmente inferida e com suporte multi-paradigmas de programação – orientação a objetos, funcional, baseada em eventos, estruturada e concorrente (Microsoft, 2015).

É baseada em um tempo de execução em máquina virtual. Um código C# é compilado e otimizado para um código de máquina especificado, chamado *Common Language Infrastructure*, ou CLI abreviado, que é então executado em uma implementação da máquina virtual *Common Language Runtime*, ou CLR abreviado (Microsoft, 2015).

A linguagem é utilizada no projeto Jarbas como base linguagem base das tecnologias do servidor – *back-end*.

**.NET Core e ASP.NET Core Framework**

.NET Core é a mais nova *framework* de código aberto livre da Microsoft e comunidade .NET. É uma completa reescrita e reestruturação da antiga .NET Framework. Modular e multi-plataforma, atualizada para as novas tendências do mercado de desenvolvimento *web*, *mobile*, *IoT*, etc (Microsoft, 2016).

O ASP.NET Core é o *framework* da plataforma .NET Core para desenvolvimento *web*. Originalmente chamado de ASP.NET MVC 5, a Microsoft decidiu alterar seu nome para não confundir o novo framework com uma simples atualização do ASP.NET MVC 4 (Scott Hanselman, 2016).

No projeto Jarbas o ASP.NET Core é responsável pelo funcionamento do servidor *back-end* da aplicação. A escolha pela plataforma se deve ao fato de permitir o desenvolvimento e *deploy* da aplicação em múltiplas plataformas, performance excepcional e suporte extenso por parte da comunidade e Microsoft.

**Entity Framework Core**

Componente do ASP.NET Core, o Entity Framework permite ao desenvolvedor conectar sua aplicação com banco de dados de forma simples e fácil. O EF, sigla do *framework*, trabalha como ORM (*object-relational mapping*), possibilitando que um desenvolvedor programe sua aplicação pensando apenas nas relações entre classes da programação, deixando todo fardo de tratamento, mapeamento e transcrição de dados de objetos para tabelas relacionais de bancos de dados (Microsoft, 2016).

O Entity Framework foi escolhido para o projeto não apenas por ser considerado o *framework* padrão da plataforma .NET – de fato, existem diversos outros *frameworks* como o Entity, mas também pela agilidade que conseguiu trazer para os desenvolvedores do projeto acoplado com sua leve curva de aprendizado.

**Identity Core**

“Identidade do ASP.NET Core é um sistema de associação que permite que você adicionar a funcionalidade de logon ao seu aplicativo. Os usuários podem criar uma conta e logon com um nome de usuário e senha ou eles podem usar um provedor de logon externo, como Facebook, Google, Microsoft Account, Twitter ou outras pessoas” (Microsoft, 2017).

O Identity Core foi escolhido por ser o mais simples e eficiente *framework* para o desenvolvimento da camada de autenticação para o projeto Jarbas.

**Python**

Python é uma linguagem de programação de alto nível, interpretada, de tipagem dinâmica e forte, e com suporte multi-paradigmas de programação – orientação a objetos, funcional, imperativa, estruturada e concorrente (Downey, 2002).

Sobre o uso de linguagens de alto nível, nas palavras de Downey (2002, tradução do autor): “as vantagens são inúmeras. Primeiro, é muito mais fácil de programar um programa em uma linguagem de alto nível. Programas escritos nestas levam menos tempo para serem escritos, são menores e muito mais certos de estarem corretos. Em segundo, linguagens de alto nível são portáveis, significa que podem ser executadas em diferentes tipos de computadores, com poucas ou nenhuma modificação.”

Criada por Guido van Rossum, possui a filosofia de prezar por legibilidade de código e simplicidade em alta e baixa escala. Sua principal implementação, o interpretador CPython, é mantido pela comunidade Python e pela Python Software Foundation, uma organização sem fins lucrativos.

Python foi escolhida para o projeto Jarbas como complementar a tarefas simples onde não fazia sentido executar toda uma aplicação .NET para sua realização ou como suplementar a casos de uso fora do escopo da plataforma principal do servidor.

**PostgreSQL**

PostgreSQL, ou simplesmente Postgres, é um sistema de gerenciamento de banco de dados relacional de código aberto livre, com um grande foco em extensibilidade e acordo com padrões de mercado. É desenvolvido pelo PostgreSQL Global Development Group, um órgão internacional composto por diversas empresas da indústria de software. (PostgreSQL, 2017)

O Postgres foi escolhido como gerenciador de banco de dados relacional do projeto Jarbas pela sua facilidade de uso, por ser software livre e multiplataforma.

**Referências**

HANSELMAN, Scott. **ASP.NET 5 is dead:** Introducing ASP.NET Core 1.0 and .NET Core 1.0. Disponível em: <https://www.hanselman.com/blog/ASPNET5IsDeadIntroducingASPNETCore10AndNETCore10.aspx>. Acesso em 18 dez. 2017.

MATERA Systems. **O que é CSS e qual sua importância.** Disponível em:

<http://www.matera.com/br/2012/07/25/o-que-e-css-e-qual-sua-importancia/>. Acesso em 11 out. 2017.

MDN web docs. JavaScript. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>. Acesso em 02 dez. 2017.

Microsoft. **Entity Framework Core Quick Overview.** Disponível em: <https://docs.microsoft.com/en-us/ef/core/>. Acesso em 18 dez. 2017.

Microsoft. **Guia do .NET Core.** Disponível em: <https://docs.microsoft.com/pt-br/dotnet/core/>. Acesso em 18 dez. 2017.

Microsoft. **Introdução à identidade no núcleo do ASP.NET.** Disponível em: <https://docs.microsoft.com/pt-br/aspnet/core/security/authentication/identity>. Acesso em 18 dez. 2017.

Microsoft. **Introdução à linguagem C# e ao .NET Framework.** Disponível em: <https://docs.microsoft.com/pt-br/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework>. Acesso em 18 dez. 2017.

MUSTAFA, Eduardo. **JavaScript:** 20 anos de história e construção da web. Disponível em: <https://imasters.com.br/front-end/javascript/javascript-20-anos-de-historia-e-construcao-da-web>. Acesso em 01 dez. 2017.

PostegreSQL. **About**. Disponível em: <https://www.postgresql.org/about/>. Acesso em 18 out. 2017.

VICENTE, Guilherme. **Apresentando o Angular.JS.** Disponível em: <http://dextra.com.br/pt/blog/apresentando-o-angular-js-4/>. Acesso em 18 nov. 2017.

w3schools. **HTML Introduction**. Disponível em: <https://www.w3schools.com/html/html\_intro.asp>. Acesso em: 18 nov. 2017.

WODEHOUSE, Carey. **The Basics of Web Development**: Markup Languages HTML, XML & XHTML. Disponível em:

<https://www.upwork.com/hiring/development/the-basics-of-web-development/>. Acesso em: 02 dez. 2017.