**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA**

**Faculdade de Tecnologia de Jundiaí – “Deputado Ary Fossen”**

**Curso Superior de Tecnologia em ADS**

Fábio Santoro Furquim

Guilherme Massao Imada Sakaguti

Gléber Michel Alves Schiavo

Rodrigo Soares de Oliveira

Matheus Bio Salles

**Compartilhamento e Divulgação de Eventos – Event Hub**

**Jundiaí**

**2019**

Fábio Santoro Furquim

Guilherme Massao Imada Sakaguti

Gléber Michel Alves Schiavo

Rodrigo Soares de Oliveira

Matheus Bio Salles

**Compartilhamento e Divulgação de Eventos**

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Jundiaí - “Deputado Ary Fossen” como requisito parcial para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, sob a orientação do Mestre Professor Cláudio Luís Vieira Oliveira.

**Jundiaí**

**2019**

Este trabalho é

dedicado aos professores e

alunos da Fatec – Jundiaí.

**AGRADECIMENTOS**

Agradecemos imensamente a Fatec Jundiaí, ao Professor e orientador Cláudio Luís e aos alunos da instituição pelo apoio e orientação durante o desenvolvimento e elaboração do trabalho aqui apresentado.

Só é verdadeiramente digno da liberdade, bem como da vida, aquele que se empenha em conquistá-la.

Johann Goethe

SOARES, Rodrigo; SCHIAVO, Gléber; SAKAGUTI, Guilherme; SALLES, Matheus; FURQUIM, Fábio. **Event Hub, Compartilhamento e Divulgação de Eventos**. 34 f. Trabalho de Conclusão de Curso de Tecnólogo em Análise e Desenvolvimento de Sistemas. Faculdade de Tecnologia de Jundiaí - “Deputado Ary Fossen”. Centro Estadual de Educação Tecnológica Paula Souza. Jundiaí. 2019.

**RESUMO**

O “Event Hub” é uma aplicação web criada e desenvolvida com o objetivo de tornar o compartilhamento e a divulgação de eventos em mídias sociais de forma automática, otimizada e rápida, de tal forma que não haja a necessidade de fazer individualmente. Através dela é possível criar um evento no Google Agenda com todos os dados preenchidos pelo usuário e divulgá-lo nas redes sociais, no caso o Twitter. Dentro do sistema também é possível visualizar eventos criados por outras pessoas, com isso o sistema se torna uma corrente de compartilhamentos, em que pessoas podem se inscrever nos eventos criados pelo usuário e também se inscrever em eventos criados por outras pessoas. Podendo realizar todas as ações descritas sem sair da aplicação. O principal público alvo são as pessoas que desejam divulgar ou criar eventos de forma facilitada e integrada, seguido pelas pessoas que estão a procura de eventos que lhes interessem.

**Palavras-chave:** Compartilhar.Social.Web.Evento.Divulgação.

SOARES, Rodrigo; SCHIAVO, Gléber; SAKAGUTI, Guilherme; SALLES, Matheus; FURQUIM, Fábio. **Event Hub, event sharing and disclosure**. 34 p. End-of-course paper in Technologist Degree in Analysis and systems development. Faculdade de Tecnologia de Jundiaí - “Deputado Ary Fossen”. Centro Estadual de Educação Tecnológica Paula Souza. Jundiaí. 2019.

**abstract**

“Event Hub” is a web application designed and developed to make sharing and publicizing events on social media. Making it automatic, optimized and fast, in such a way that there is no need to do it individually. Through it is possible to create an event in Google Calendar with all data filled in by the user and posted on social networks, in this case Twitter. Within the system it is also possible to view events created by other people, so the system becomes a sharing stream where people can register for user created events and also subscribe to events created by other people. Being able to perform all the described actions without leaving the application. The main audience is people who sights to promote or create events in an easy and integrated way, followed by people who are looking for events of their interests.

**Keywords:** Share.Social.Web.Event.Divulgation.

**Lista de Ilustrações**

[Figura 1 - Diagrama de Casos de Uso 18](#_Toc25211297)

[Figura 2 - Diagrama de classe - User 21](#_Toc25211298)

[Figura 3 - Diagrama de classe - Event 23](#_Toc25211299)

[Figura 4 - Diagrama de Classe - Social 24](#_Toc25211300)

[Figura 5 – Diagrama de classe da biblioteca SocialConnection 25](#_Toc25211301)

[Figura 6 - Diagrama de Pacotes 26](#_Toc25211302)

[Figura 7 - Diagrama Entidade Relacionamento 27](#_Toc25211303)

**LISTA DE TABELAS**

[Tabela 1 - Requisitos Funcionais 14](#_Toc25211286)

[Tabela 2 - Requisitos não Funcionais 16](#_Toc25211287)

[Tabela 3 - Requisitos de Domínio 17](#_Toc25211288)

[Tabela 4 - Casos de Uso 19](#_Toc25211289)

[Tabela 5 - Entidade Usuário 27](#_Toc25211290)

[Tabela 6 - Entidade Public Place 28](#_Toc25211291)

[Tabela 7 - Entidade Adress 28](#_Toc25211292)

[Tabela 8 - Entidade Event 28](#_Toc25211293)

[Tabela 9 - Entidade EventSubscriptions 29](#_Toc25211294)

[Tabela 10 - Entidade GoogleCalendarSocialMarketing 29](#_Toc25211295)

[Tabela 11 - Entidade TwitterSocialMarketing 29](#_Toc25211296)

**Sumário**

[1 Introdução 12](#_Toc25211309)

[1.1 Escopo 13](#_Toc25211310)

[1.2 Clientes do software 13](#_Toc25211311)

[2 Requisitos do Sistema 14](#_Toc25211312)

[2.1 Requisitos Funcionais 14](#_Toc25211313)

[2.2 Requisitos não funcionais 16](#_Toc25211314)

[2.3 Requisitos do Domínio 17](#_Toc25211315)

[3 Definição do Projeto (caso: orientado objeto) 18](#_Toc25211316)

[3.1 Casos de Uso 18](#_Toc25211317)

[3.2 Diagramas de Classe 21](#_Toc25211318)

[3.3 Diagramas de Pacotes 26](#_Toc25211319)

[3.4 Diagrama Entidade Relacionamento 26](#_Toc25211320)

[3.5 Dicionário de Dados 27](#_Toc25211321)

[4 Escolha da Arquitetura da Solução 31](#_Toc25211322)

[5 Considerações Finais 32](#_Toc25211323)

[Referências 33](#_Toc25211324)

[APÊNDICE A - Manual do Usuário 34](#_Toc25211325)

# Introdução

O Event Hub foi desenvolvido, assim como muitas aplicações vistas hoje em dia, para facilitar as operações que o usuário deve fazer e também diminuir o número de ações necessárias para realizá-la.

A motivação de seu desenvolvimento veio da necessidade de criar uma opção de fácil acesso e entendimento de seus usuários de forma satisfatória, quando se trata de criação e divulgação de eventos.

Seu objetivo final é fazer com que as pessoas gostem e queiram usá-lo como sua primeira opção para atrair e se encontrar com seus amigos, familiares, colegas ou o público no geral.

Especificação do Programa

A seguir são apresentados os itens Escopo, que define de forma objetiva e precisa quais serão as funcionalidades do sistema assim como suas restrições e limites, e Clientes do Software.

## Escopo

O “Event Hub” é uma aplicação web onde se cria um evento no Google Agenda com todos os dados preenchidos pelo usuário para que seja divulgado nas redes sociais, podendo-se inscrever ou visualizar os eventos criados pelo usuário e também por outras pessoas. A aplicação não permite a criação de duas agendas para o mesmo evento para fins de evitar a redundância.

## Clientes do software

O software se trata de um sistema voltado para o âmbito tanto social, organizacional ou acadêmico. Portanto o público alvo do software se destina às pessoas que desejam utilizar do sistema para fins de divulgação, além daqueles que desejam procurar por atividades de seu interesse.

# Requisitos do Sistema

Qualquer tipo de aparelho eletrônico que tenha acesso à internet através de um navegador e ser possuidor de contas da Google e Twitter para explorar todo potencial do software.

## Requisitos Funcionais

Segue abaixo a tabela 1 contendo os requisitos funcionais do projeto:

Tabela - Requisitos Funcionais

|  |  |  |
| --- | --- | --- |
| Prioridade | Requisito | Descrição |
| Alta | Criação da conta do usuário | O usuário poderá criar um nova conta para utilizar das funções do sistema. |
| Alta | Edição da conta do usuário | O usuário poderá mudar as informações referente aos dados pessoais da sua conta. |
| Média | Exclusão da conta do usuário | O usuário poderá excluir sua conta, caso deseje. |
| Alta | Login do usuário | O usuário poderá fazer login no sistema, com uma conta já criada, para utilizar suas funcionalidades. |
| Alta | Redirecionar para a tela de login | Caso o usuário não esteja logado o sistema deve redirecionar para a tela de login, independentemente da página acessada. |
| Alta | Criação de um novo evento | O usuário poderá criar um evento após preencher um formulário com os dados desse evento. |
| Alta | Edição de um evento já criado | O usuário poderá editar o evento previamente criado |
| Alta | Exclusão de um evento já criado | O usuário poderá excluir um evento previamente criado, caso deseje. |
| Alta | Listagem e busca dos evento ativos | O sistema deverá listar apenas os eventos ativos para que os usuários possam buscar o evento desejado e se inscrever. |
| Alta | Inscrição de um evento já existente por outro usuário, diferente do criador | O usuário poderá se inscrever em um evento criado por outro usuário. |
| Média | Vínculo com a conta do Twitter do usuário | O usuário poderá vincular a sua conta do EventHub com a do Twitter, para acessar as funcionalidades do tweet. |
| Média | Vínculo com a conta Google do usuário | O usuário poderá vincular a sua conta do EventHub com a do Google, para acessar as funcionalidades do Google Agenda. |
| Média | Atualização das informações do usuário referente às API’s das redes sociais | O sistema deverá atualizar os tokens de acesso para que o usuário tenha acesso às funcionalidades das API’s das redes sociais. |
| Média | Postagem de um tweet com as informações ao criar o evento | O sistema deverá postar um novo tweet após o usuário criar um novo evento. |
| Média | Postagem pelo usuário de um tweet com as informações do evento já criado. | O usuário poderá postar quantos tweets desejar com as informações do evento após sua criação. |
| Baixa | Edição das informações do tweet a ser enviado | O usuário poderá editar as informações do tweet que será postado. |
| Média | Criação de um evento no Google Agenda com as informações ao criar o evento | O sistema deverá criar um novo evento no Google Agenda após o usuário criar um novo evento. |
| Baixa | Edição das informações do evento no Google Agenda | O sistema deverá permitir que o usuário edite as informações do evento criado no Google Agenda após a edição do evento no EventHub. |
| Média | Exclusão do evento no Google Agenda após o usuário excluir o evento no sistema. | O sistema deverá excluir o evento criado no Google Agenda após o usuário excluir o evento no EventHub. |
| Média | Inscrição na agenda do evento ao se inscrever | O usuário poderá se inscrever no evento do Google agenda após se inscrever no evento do EventHub. |

## Requisitos não funcionais

Segue abaixo a tabela 2 contendo os requisitos não funcionais do projeto:

Tabela - Requisitos não Funcionais

|  |  |
| --- | --- |
| Nome | Descrição |
| Portabilidade | O sistema deverá executar em dispositivos que possuem acesso à internet em qualquer navegador. |
| Interoperabilidade | O sistema deverá se comunicar com o banco SQL Server. |
| Implementação | A aplicação será desenvolvida utilizando o framework Angular (front-end) e .NET (back-end). |
| Confiabilidade | O sistema deverá garantir a total segurança dos dados do usuário. |
| Usabilidade | O sistema deverá ser amigável e possuir uma interface simples e de fácil manuseio para novos usuários. |
| Reusabilidade | O sistema deverá ser construído a partir de módulos para uma melhor reusabilidade. |
| Segurança | O sistema deverá gerar um backup de seus dados periodicamente. |

## Requisitos do Domínio

Segue abaixo a tabela 3 contendo os requisitos do domínio do projeto:

Tabela - Requisitos de Domínio

|  |
| --- |
| Requisitos de Domínio |
| O sistema deverá permitir compartilhar informações sobre perfis de usuários entre si. |

# Definição do Projeto

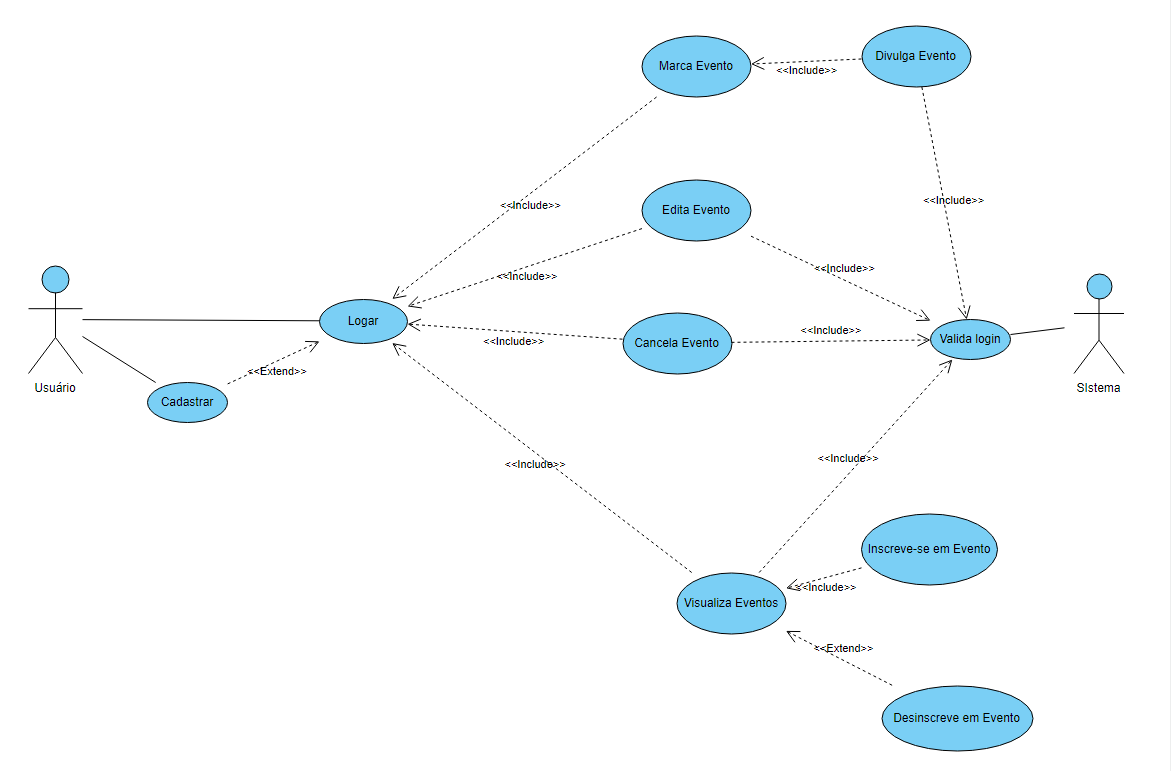
A API (application programming interface) e o website presentes na plataforma Event Hub foram modelados utilizando o paradigma orientado à objetos, devido a sua simples abstração, modularização e flexibilidade de manutenção. Além de facilitar a implementação dos seus módulos e funções, fazendo um paralelo ao mundo real, tradando-se de um processo muito mais natural a linguagem humana.

Já o banco de dados foi estruturado de acordo com o modelo relacional, por proporcionar robustez e garantir persistências dos dados da aplicação. Além de ser de comum conhecimento dentre os presentes envolvidos no projeto.

## Casos de Uso

Segue abaixo a Figura 1 contendo os Casos de uso do projeto:

Figura - Diagrama de Casos de Uso



Segue abaixo a Tabela 4 contendo os Casos de uso do projeto:

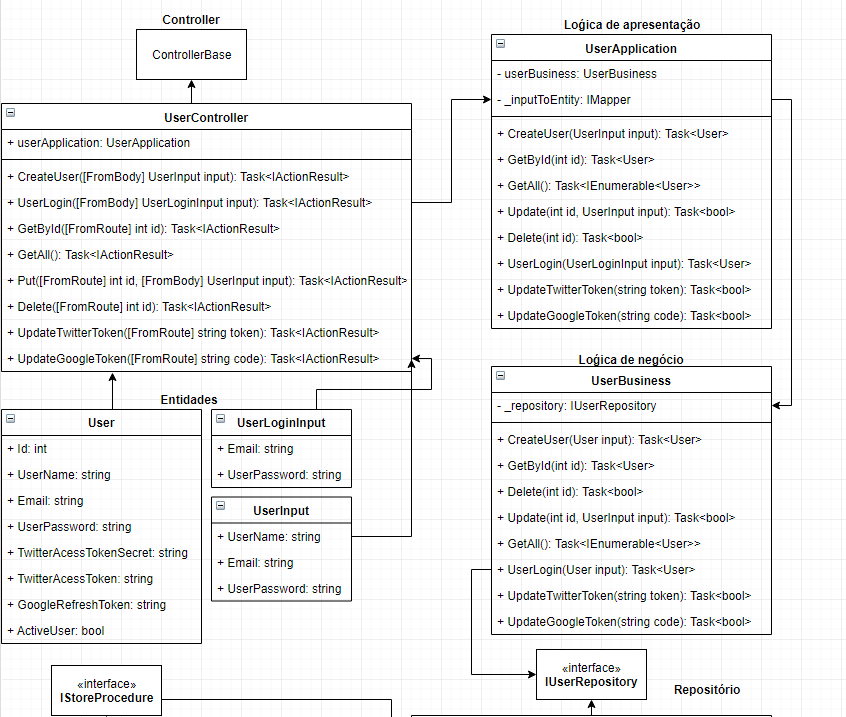
Tabela - Casos de Uso

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Caso de uso | Descrição | Atores Envolvidos | Pré-Condição | Pós-Condição | Fluxo de tarefas |
| UC01- Cadastrar | Cadastrar um novo usuário | Usuário | - | Logar | Usuário clica no botão cadastrar,  sistema mostra os dados de cadastro,  usuário preenche os dados de cadastro e clicar em cadastrar. |
| UC02- Logar | Realiza login do usuário. | Usuário | Possuir um cadastro. | Direciona à página principal. | Usuário clica no botão login, sistema mostra os dados de realização de login, usuário preenche os dados de login e clicar em logar. |
| UC03-Marca Evento | Cria um evento novo. | Usuário | Estar logado na conta. | Divulga Evento. | Usuário clica em marcar evento, exibe dados de criação de eventos, usuário preenche os dados. |
| UC04-Divulga  Evento | Divulga o evento criado | Usuário | Marca Evento | O evento é divulgado e valida login. | Usuário clica em divulgar evento. |
| UC05-Edita evento | Edita o evento criado | Usuário | Evento estar divulgado. | O evento é alterado, sobrescrevendo o anterior. | Usuário clica em editar evento, dados de alteração do evento são exibidos, usuário preenche os dados que deseja alterar e clicar em alterar evento. |
| UC06-Cancela Evento | Exclui o evento criado | Usuário | Evento estar divulgado. | o evento deixa de existir. | Usuário clica no botão cancelar evento. |
| UC07-Visualiza Evento | Visualizar eventos de outros usuários. | Usuário | Estar logado na conta. | Inscreve  -se em evento. | Usuário clica em visualizar eventos e escolhe algum evento de seu interesse. |
| UC08- Inscreve-se em evento | faz inscrição em algum evento. | Usuário | escolher o evento. | A inscrição no evento é executada | Usuário clica em realizar inscrição no evento desejado. |
| UC09- Desinscreve em Evento | desfaz inscrição em algum evento | Usuário | estar inscrito em algum evento | A inscrição no evento é desfeita. | Usuário clicar em desfazer inscrição no evento desejado. |
| UC10-Valida Login | valida as ações do usuário em relação aos eventos. | Sistema | Usuário deve divulgar,alterar,excluir,visualizar,inscrever ou desinscrever de algum evento. | a ação do usuário é validada pelo sistema e salva. | Sistema executa validação em todas as situações onde o usuário realiza alguma ação a qual esteja vinculado seu login. |

## Diagramas de Classe

Segue abaixo as Figuras 2, 3, 4 e 5 contendo os Diagramas de Classe do projeto:

Figura - Diagrama de classe - User



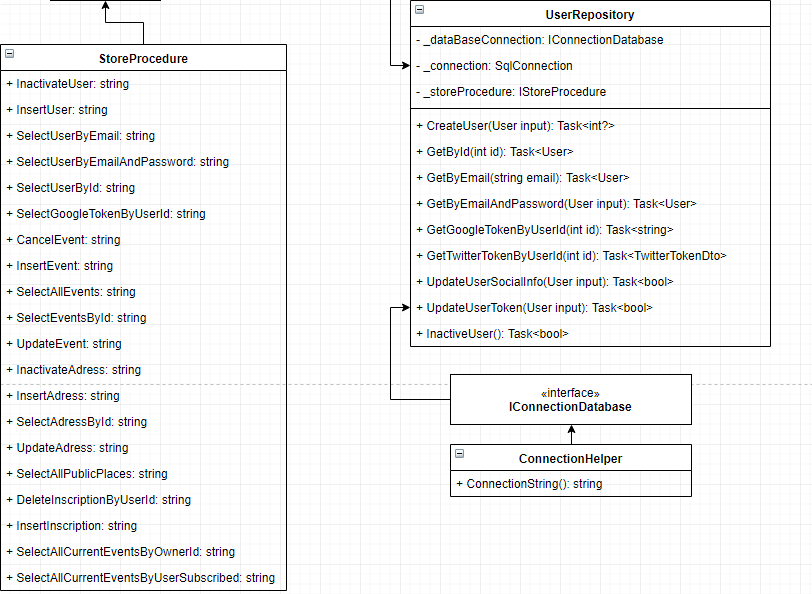
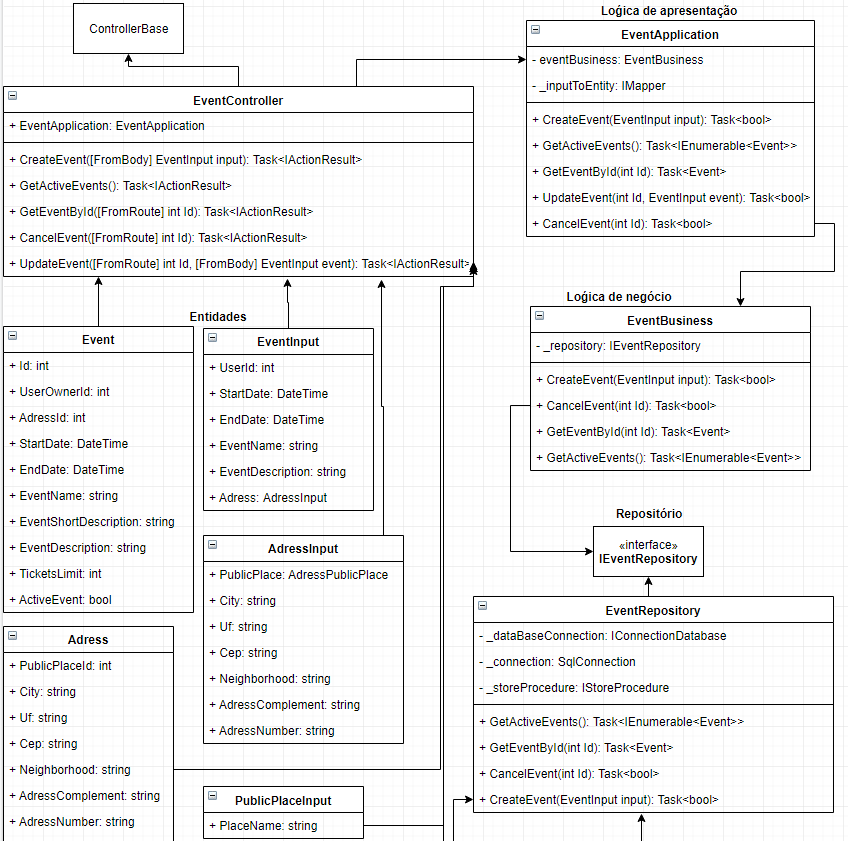


Figura - Diagrama de classe - Event



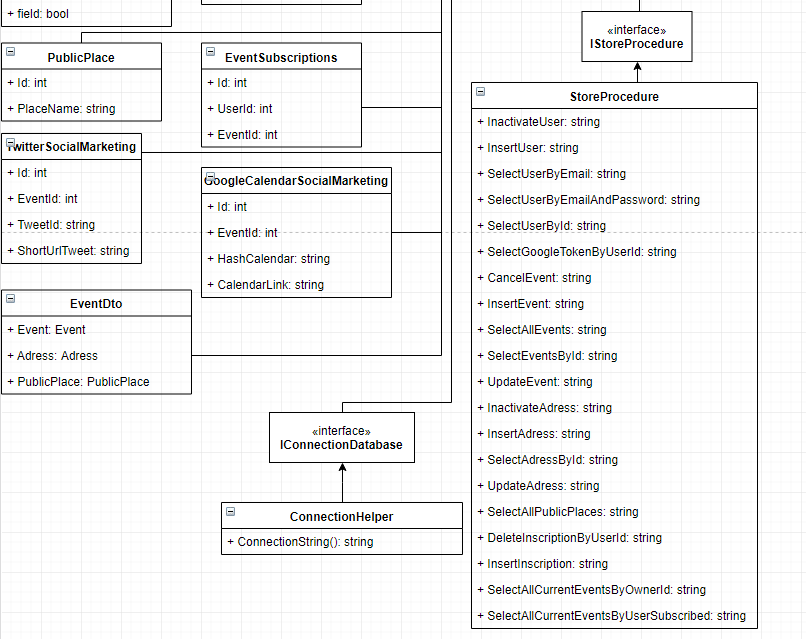


Figura - Diagrama de Classe - Social

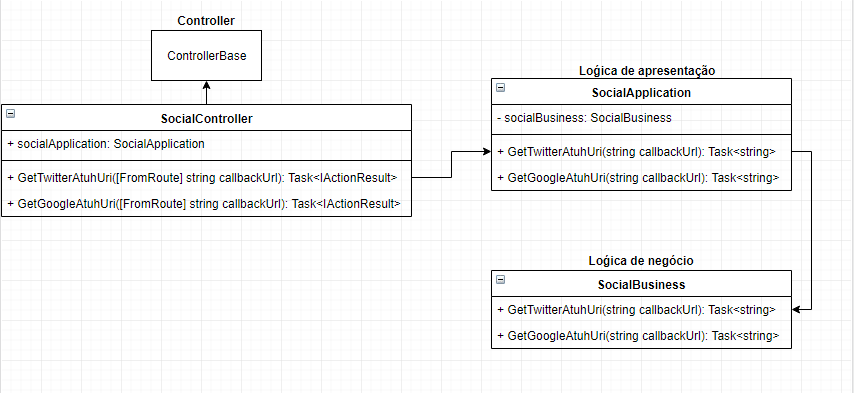
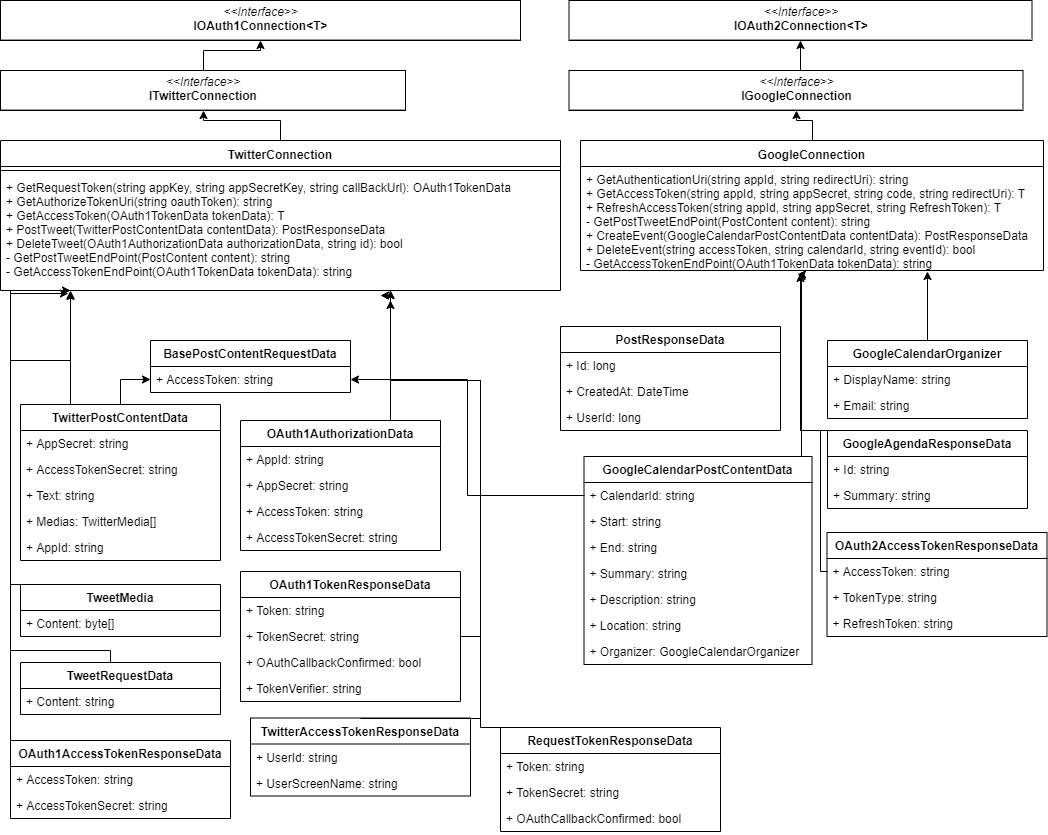


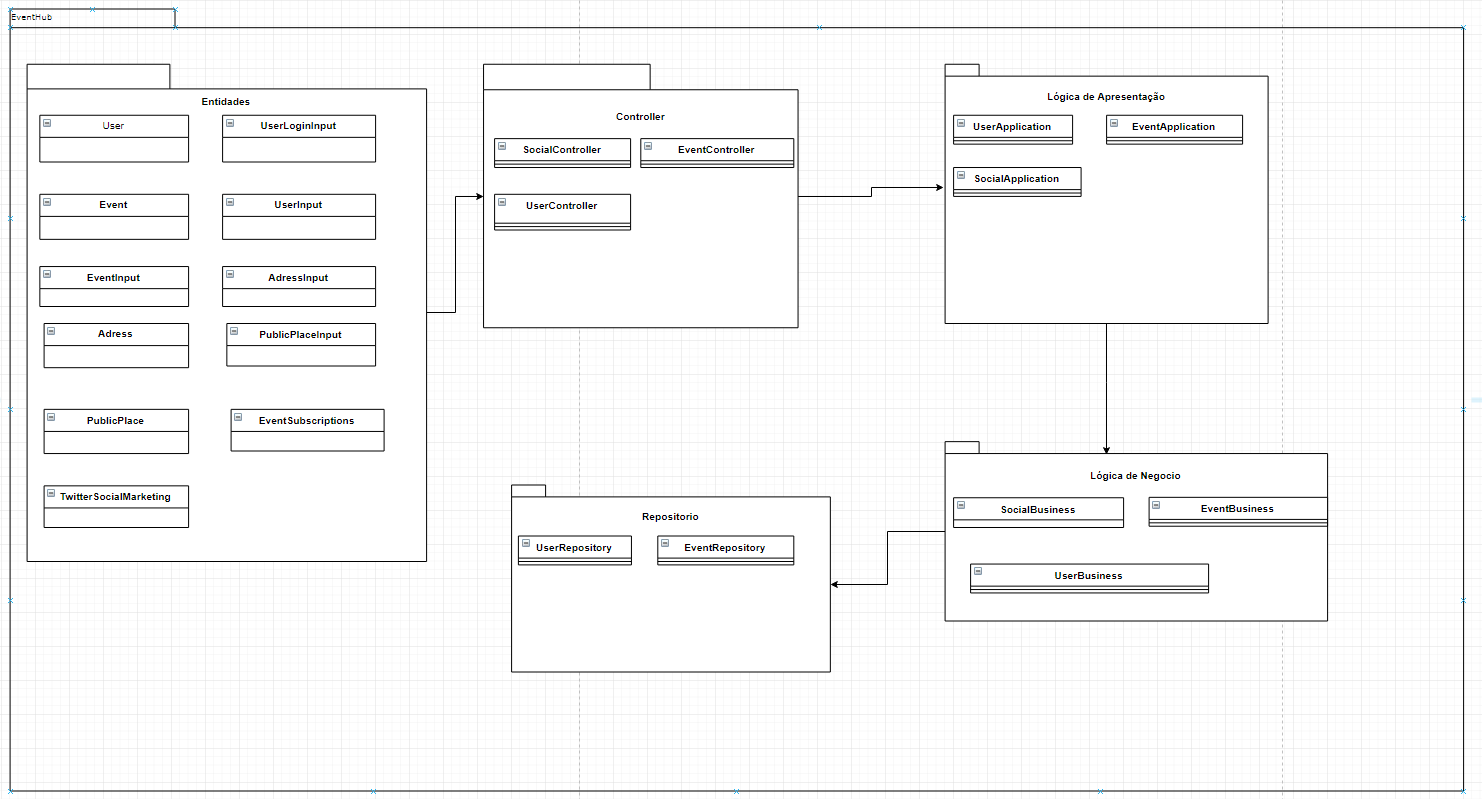
Figura – Diagrama de classe da biblioteca SocialConnection



## Diagramas de Pacotes

Segue abaixo a Figura 6 contendo o Diagrama de Pacotes do projeto:

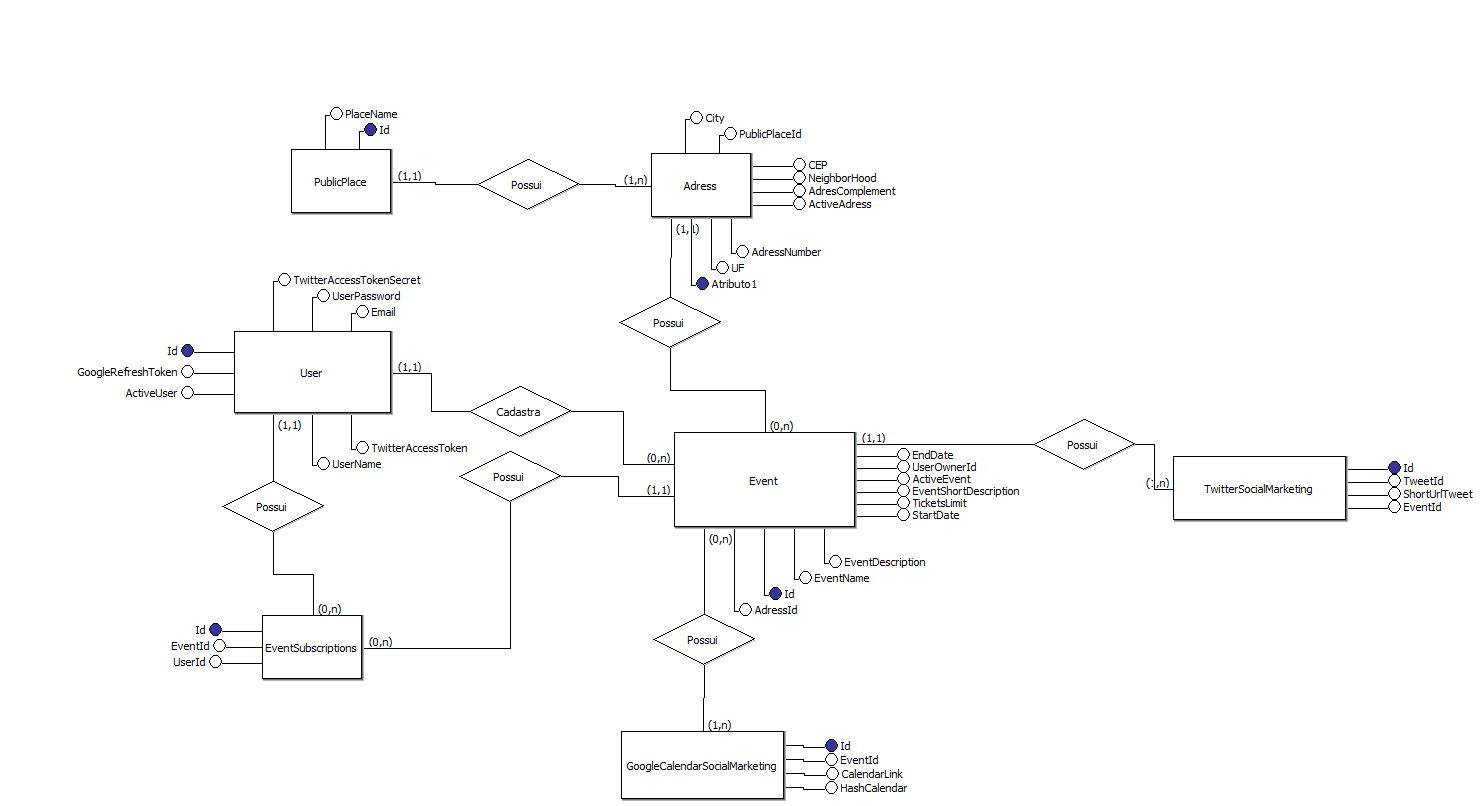
Figura - Diagrama de Pacotes



## Diagrama Entidade Relacionamento

Segue abaixo a Figura 7 contendo o Diagrama de Entidade Relacionamento do projeto:

Figura - Diagrama Entidade Relacionamento



## Dicionário de Dados

Segue abaixo as Tabelas 5, 6, 7, 8, 9, 10 e 11 contendo o Dicionário de Dados do Projeto:

Tabela - Entidade Usuário

|  |  |  |  |
| --- | --- | --- | --- |
| Entidade: Usuário | | | |
| Atributo | Classe | Domínio | Tamanho |
| ID | Determinante | Numérico |  |
| UserName | Simples | Texto | 50 |
| Email | Simples | Texto | 50 |
| UserPassword | Simples | Texto | 15 |
| TwitterAcessTokenSecret | Simples | Texto | 10 |
| TwitterAcessToken | Simples | Texto | 200 |
| GoogleRefreshToken | Simples | Texto | 10 |
| ActiveUser | Simples | Booleano |  |

Tabela - Entidade Public Place

|  |  |  |  |
| --- | --- | --- | --- |
| Entidade: PublicPlace | | | |
| Atributo | Classe | Domínio | Tamanho |
| ID | Determinante | Numérico |  |
| PlaceName | Simples | Texto | 20 |

Tabela - Entidade Adress

|  |  |  |  |
| --- | --- | --- | --- |
| Entidade: Adress | | | |
| Atributo | Classe | Domínio | Tamanho |
| ID | Determinante | Numérico |  |
| PublicPlaceID | Determinante | Numérico |  |
| City | Simples | Texto | 50 |
| UF | SImples | Texto | 2 |
| CEP | Simples | Texto | 10 |
| Neighborhood | Simples | Texto | 50 |
| AdressComplement | Simples | Texto | 50 |
| AdressNumer | Simples | Texto | 5 |
| ActiveAdress | Simples | Booleano |  |

Tabela - Entidade Event

|  |  |  |  |
| --- | --- | --- | --- |
| Entidade: Event | | | |
| Atributo | Classe | Domínio | Tamanho |
| ID | Determinante | Numérico |  |
| UserOwnerId | Determinante | Numérico |  |
| AdressId | Determinante | Numérico |  |
| StartDate | SImples | Data e Hora | dd/mm/aaaa hh:mm:ss |
| EndDate | Simples | Data e Hora | dd/mm/aaaa hh:mm:ss |
| EventName | Simples | Texto | 80 |
| EventShortDescription | Simples | Texto | 50 |
| EventDescription | Simples | Texto | 1000 |
| TicketsLimit | Simples | Numérico |  |
| ActiveEvent | Simples | Boleano |  |

Tabela - Entidade EventSubscriptions

|  |  |  |  |
| --- | --- | --- | --- |
| Entidade: EventSubscriptions | | | |
| Atributo | Classe | Domínio | Tamanho |
| ID | Determinante | Numérico |  |
| UserId | Determinante | Numérico |  |
| EventId | Determinante | Numérico |  |

Tabela - Entidade GoogleCalendarSocialMarketing

|  |  |  |  |
| --- | --- | --- | --- |
| Entidade: GoogleCalendarSocialMarketing | | | |
| Atributo | Classe | Domínio | Tamanho |
| HashCalendar | Simples | Texto | 30 |
| CalendarLink | Simples | Texto | 10 |
| ID | Determinante | Numérico |  |
| EventId | Determinante | Numérico |  |

Tabela - Entidade TwitterSocialMarketing

|  |  |  |  |
| --- | --- | --- | --- |
| Entidade: TwitterSocialMarketing | | | |
| Atributo | Classe | Domínio | Tamanho |
| ID | Determinante | Numérico |  |
| EventId | Determinante | Numérico |  |
| TweetID | Determinante | Texto | 20 |
| ShortUrlTweet | Simples | Texto | 30 |

# Escolha da Arquitetura da Solução

O sistema EventHub é composto por três partes, o FrontEnd, que foi criado utilizando a plataforma Angular 8 que utiliza a linguagem typescript, o BackEnd, que por sua vez é subdividido em algumas camadas de acordo com o padrão DDD (Domain-Driven Design), sendo que a WebApi está utilizando o .NET Core e as Class Libraries o .NET Framework 4.7.2 ambas utilizando a linguagem C#, e o Banco de dados, que foi gerado utilizando o SQL Server.

Para o BackEnd foi escolhido o padrão de arquitetura DDD, que consiste em dividir o sistema em camadas: a camada Application, responsável por conter a WebApi e as classes de apresentação, a Domain, responsável por conter as entidades e as classes POCOs (Plain old CLR objects) auxiliares, a Infrastructure, responsável por realizar a persistência no banco de dados e a Service, que contém a lógica de negócio. A arquitetura DDD consiste em modularizar o sistema para separar as responsabilidades das camadas do software.

Para o FrontEnd foi desenvolvido utilizando a biblioteca Angular 8, uma ferramenta desenvolvida pela Google, que consiste em dividir a aplicação em pequenas camadas, chamadas componentes, para que seja possível a reutilização de um mesmo component em várias partes do projeto, para maximizar a reutilização do código.

O banco de dados foi moldado utilizando o SQL Server, já que possuí uma boa compatibilidade com o .NET Framework e é um banco relacional, onde é possível moldar nosso banco de forma consistente.

Todas as funcionalidades do sistema seguem um ciclo base que começa no FrontEnd, onde o usuário insere os dados pela interface. Após realizada a ação pelo FrontEnd é feita uma requisição à WebApi do BackEnd, onde esses dados seguirão pelo fluxo do padrão DDD. Os dados passarão pela camada Application, que fará a validação inicial do payload recebido, irá para a camada de Service, onde está implementado as lógicas de negócio da aplicação, após isso irá para a Infrastructure, onde as classes de repositório serão responsáveis por se comunicar com o banco e realizar as operações necessárias. Após o método do repositório ser executado é enviado a resposta para o Controller correspondente na WebApi que irá devolver o response para o Front.

# Considerações Finais

O desenvolvimento do trabalho presente possibilitou uma análise de como um software feito sob uma visão específica pode melhorar a apuração dos resultados para quem busca divulgar e compartilhar eventos como foco principal. Nesse sentido, a utilização de tais recursos digitais permite aos seus usuários realizarem seu trabalho de forma mais rápida e eficiente.

Referências

TWITTER (Estados Unidos). **Docs**: Twitter Developers. 1.0. [*S. l.*], 2019. Disponível em: <https://developer.twitter.com/en/docs>. Acesso em: 20 nov. 2019.

GOOGLE (Estados Unidos) *et al*. **Calendar**: Developers. 1.0. [*S. l.*], 2019. Disponível em: <https://developers.google.com/calendar>. Acesso em: 20 nov. 2019.

NASCIMENTO , Wellington *et al*. Não lance Exceptions em seu Domínio: Use Notifications!. **Wellingtonjhn**, Brasil, p. 1-10, 25 nov. 2018. Disponível em: <https://www.wellingtonjhn.com/posts/não-lance-exceptions-em-seu-domínio-use-notifications/>. Acesso em: 20 nov. 2019.

APÊNDICE A - Manual do Usuário

O sistema EventHub é um sistema web, dessa forma para a sua utilização é necessário possuir conexão com a Internet e um navegador web (Google Chrome, Mozilla Firefox, Safari, entre outros).

Para usufruir das funções fornecidas pelo sistema é necessário criar uma nova conta, caso não possua. O sistema fornece um sistema simples e rápido de cadastro para que o usuário possa começar a utilizar o site imediatamente. O usuário caso não esteja logado no site será redirecionado automaticamente para a página de login. Dessa tela é possível selecionar a opção de “Ainda não é cadastrado? Clique aqui!”, onde será redirecionado para a tela de cadastro. Nessa tela, ao terminar de preencher as informações pedidas pelo formulário e clicar no botão “Cadastrar” o usuário será redirecionado para a tela principal do aplicativo.

Caso o usuário já possua uma conta no site, será possível entrar utilizando o e-mail e a senha informadas no cadastro, após isso o sistema irá redirecionar para a tela principal.

A tela principal do aplicativo o usuário terá visão dos eventos que ainda estão ativos para que possa se inscrever. Ao clicar no evento desejado será redirecionado para a tela com as informações do evento. Se o usuário se interessar pelo evento é possível se inscrever no evento clicando no botão “Inscrever”.

O usuário poderá gerenciar os eventos inscritos através de uma página específica, que possui uma lista com os detalhes dos eventos inscritos e um botão de ação para se desinscrever do evento.

Para editar as informações do usuário haverá a tela do perfil do usuário. Nesta tela é exibida as informações do usuário que foram cadastradas onde será possível editá-las, além da possibilidade de se conectar com as contas do Twitter e do Google.

Em todas as telas, exceto às de login e cadastro, será mostrado o menu superior, que irá auxiliar o usuário a navegar pelas funções do site.