



SpeedFanatic - O Portal da velocidade.

SpeedFanatic é um portal intuitivo para os amantes de Fórmula 1. Estamos aqui para oferecer facilidade de acesso às informações sobre o esporte, permitindo que até mesmo os novatos se sintam à vontade em mergulhar nesse mundo.

Integrantes:

Diego Polanski;

Guilherme de Almeida;

Pedro Marçal;

Pedro Ribeiro;







CONSIDERAÇÕES INICIAIS

The background of the slide is a dark, futuristic digital environment. It features a grid of glowing blue lines that create a sense of depth and connectivity. In the foreground and midground, there are several server racks or data storage units, each composed of multiple horizontal layers. These units are illuminated with a bright blue light, giving them a metallic, high-tech appearance. The overall aesthetic is clean, modern, and tech-oriented, typical of a presentation about software development or IT infrastructure.




- Back-end e Front-end;
- Pacotes APP, DAO, MODEL e SERVICE;
- CRUDs

ESTRUTURA DAS PASTAS








SERVICE

 CircuitoService.j...	13.702	2.510	Arquivo JAVA	26/10/2023 00:52	DAA46A86
 EquipeService.ja...	10.118	2.227	Arquivo JAVA	26/10/2023 18:26	B713A802
 PilotoService.java	11.154	2.306	Arquivo JAVA	26/10/2023 19:27	9D87F37C
 PostService.java	31.738	5.294	Arquivo JAVA	26/10/2023 19:41	AEA015ED
 RespostaService....	18.000	3.220	Arquivo JAVA	26/10/2023 19:39	362C6FAE
 UserService.java	13.615	2.688	Arquivo JAVA	26/10/2023 19:44	E82754E8

MODEL

 Circuito.java	2.048	508	Arquivo JAVA	25/10/2023 20:42	F0BC32EB
 Equipe.java	1.176	464	Arquivo JAVA	26/10/2023 17:47	C42E37CB
 Piloto.java	1.912	527	Arquivo JAVA	26/10/2023 19:20	0DCDF84A
 Post.java	1.689	635	Arquivo JAVA	26/10/2023 12:38	4B9B509E
 Resposta.java	2.237	622	Arquivo JAVA	26/10/2023 14:03	F0314D58
 User.java	2.185	603	Arquivo JAVA	19/10/2023 18:08	1A209B28

DAO

 CircuitoDAO.java	5.208	1.022	Arquivo JAVA	25/10/2023 21:47	3ECFD30C
 DAO.java	1.159	468	Arquivo JAVA	18/10/2023 16:19	D94ABC84
 EquipeDAO.java	4.540	1.220	Arquivo JAVA	26/10/2023 19:19	E37FD192
 PilotoDAO.java	5.857	1.170	Arquivo JAVA	26/10/2023 19:32	5033FDB1
 PostDAO.java	6.102	1.351	Arquivo JAVA	26/10/2023 12:43	0F564FB1
 RespostaDAO.java	8.028	1.484	Arquivo JAVA	26/10/2023 17:06	373D73EA
 UserDAO.java	8.784	1.556	Arquivo JAVA	26/10/2023 12:39	B7F5F74A

SCRIPT POSTGRESQL

```
-- Name: resposta_resposta_id; Type: SEQUENCE; Schema: public; Owner: ti2cc
--

CREATE SEQUENCE public.resposta_resposta_id
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;

ALTER TABLE public.resposta_resposta_id OWNER TO ti2cc;

--
-- Name: resposta; Type: TABLE; Schema: public; Owner: ti2cc
--

CREATE TABLE public.resposta (
  resposta_id integer DEFAULT nextval('public.resposta_resposta_id'::regclass) NOT NULL,
  resposta_conteudo text,
  resposta_data timestamp with time zone DEFAULT CURRENT_TIMESTAMP,
  resposta_post integer NOT NULL,
  resposta_pai_id integer,
  resposta_usuario text,
  resposta_categoria integer
);

ALTER TABLE public.resposta OWNER TO ti2cc;

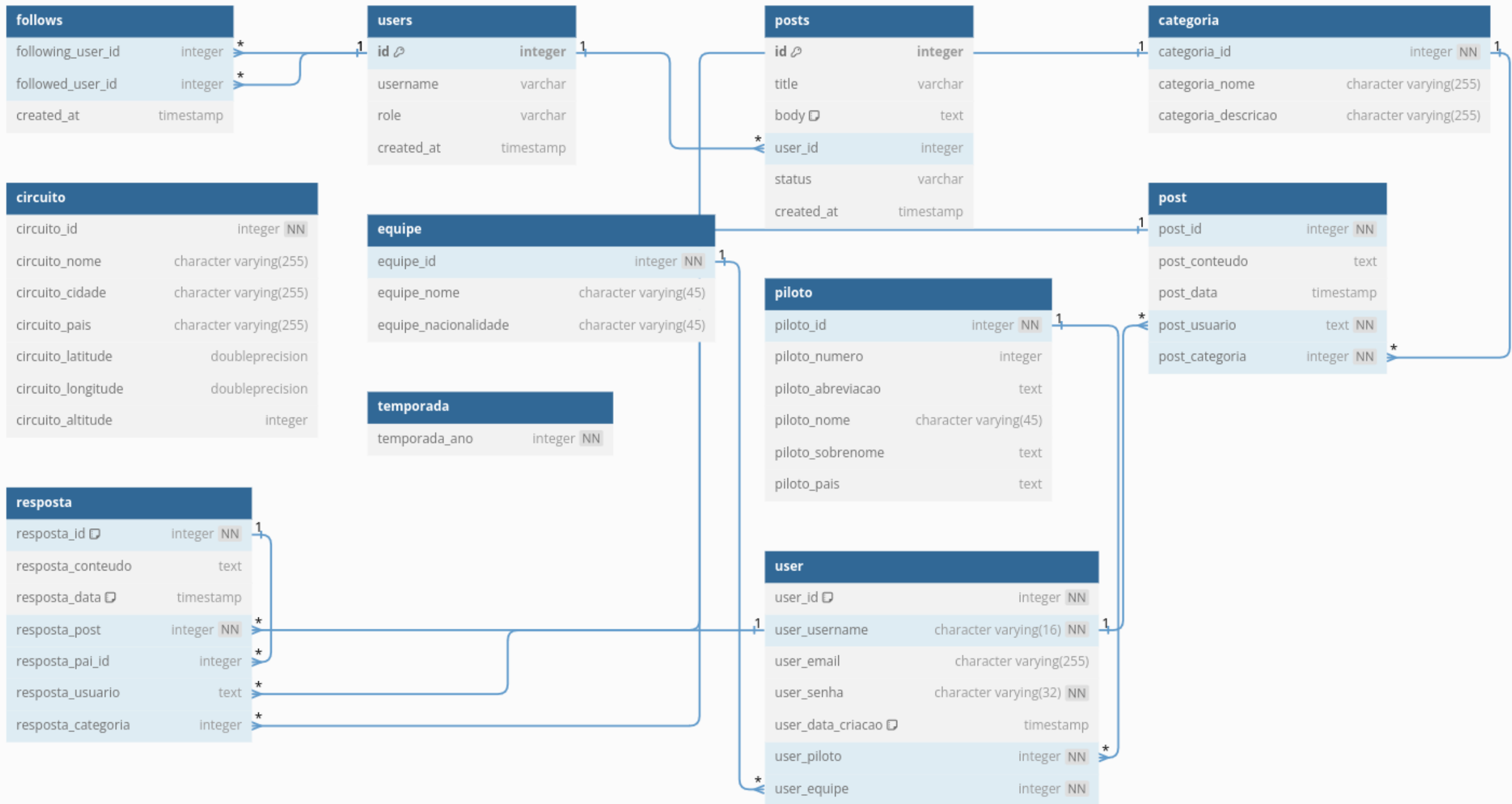
--
-- Name: resposta_reposta_pai_id_seq; Type: SEQUENCE; Schema: public; Owner: ti2cc
--

CREATE SEQUENCE public.resposta_reposta_pai_id_seq
  AS integer
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;

ALTER TABLE public.resposta_reposta_pai_id_seq OWNER TO ti2cc;

--
-- Name: resposta_reposta_pai_id_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: ti2cc
--

ALTER SEQUENCE public.resposta_reposta_pai_id_seq OWNED BY public.resposta.resposta_pai_id;
```



PACKAGE MODEL

```
public class Resposta {
    private int id;
    private String conteudo;
    private LocalDate data;
    private String usuario;
    private int post;
    private int respostaPaiId;
    private String chain;
    private int categoria;

    public Resposta() {
        id = 1;
        conteudo = "";
        data = LocalDate.now();
        usuario = "";
        post = 1;
        respostaPaiId = 0;
        categoria = 0;
    }

    public Resposta(int id, String conteudo, int post, String usuario, int respostaPaiId, LocalDate data, int categoria) {
        setID(id);
        setConteudo(conteudo);
        setPost(post);
        setUsuario(usuario);
        setRespostaPaiId(respostaPaiId);
        setData(data);
        setCategoria(categoria);
    }
}
```

PACKAGE APLICAÇÃO

```
post("/piloto/insert", (request, response) -> pilotoService.insert(request, response));

get("/piloto/:id", (request, response) -> pilotoService.get(request, response));

get("/piloto/list/:pagina", (request, response) -> pilotoService.getAll(request, response));

get("/piloto/update/:id", (request, response) -> pilotoService.getToUpdate(request, response));

post("/piloto/update/:id", (request, response) -> pilotoService.update(request, response));

get("/piloto/delete/:id", (request, response) -> pilotoService.delete(request, response));

post("/equipe/insert", (request, response) -> equipeService.insert(request, response));

get("/equipe/:id", (request, response) -> equipeService.get(request, response));

get("/equipe/list/:orderby", (request, response) -> equipeService.getAll    (request, response));

get("/equipe/update/:id", (request, response) -> equipeService.getToUpdate(request, response));

post("/equipe/update/:id", (request, response) -> equipeService.update(request, response));

get("/equipe/delete/:id", (request, response) -> equipeService.delete(request, response));


before("/post/insert", (request, response) -> {
    if (!usuarioEstaAutenticado(request)) {
        response.redirect("/user/pagina/1"); // Redireciona para a página de login se o usuário não estiver autenticado
    }
});
before("/resposta/insert", (request, response) -> {
    if (!usuarioEstaAutenticado(request)) {
        response.redirect("/user/pagina/1"); // Redireciona para a página de login se o usuário não estiver autenticado
    }
});
before("/post/list:orderby", (request, response) -> {
    if (!usuarioEstaAutenticado(request)) {
        response.redirect("/user/pagina/1"); // Redireciona para a página de login se o usuário não estiver autenticado
    }
});
```

PACKAGE DAO

```
public boolean insert(Resposta resposta) {
    boolean status = false;
    try {
        String sql = "INSERT INTO resposta (resposta_conteudo, resposta_data, resposta_usuario, resposta_post, resposta_pai_id, resposta_categoria) " +
            "VALUES (?, ?, ?, ?, ?, ?)";
        PreparedStatement st = conexao.prepareStatement(sql);
        st.setString(1, resposta.getConteudo());
        st.setTimestamp(2, Timestamp.valueOf(resposta.getData().atStartOfDay()));
        st.setString(3, resposta.getUsuario());
        st.setInt(4, resposta.getPost());
        if(resposta.getRespostaPaiId()==0)
        {
            st.setNull(5, resposta.getRespostaPaiId());
        }
        else
        st.setInt(5, resposta.getRespostaPaiId());
        st.setInt(6, resposta.getCategoria());
        st.executeUpdate();
        st.close();
        status = true;
    } catch (SQLException u) {
        throw new RuntimeException(u);
    }
    return status;
}
```


PACKAGE SERVICE

```
public Object insert(Request request, Response response) {
    String nome = request.queryParams("piloto_nome");
    String sobrenome = request.queryParams("piloto_sobrenome");
    String pais = request.queryParams("piloto_pais");
    int numero = Integer.parseInt(request.queryParams("piloto_numero"));
    String abreviacao = request.queryParams("piloto_abreviacao");

    String resp = "";
    if (nome == null) {
        resp = "Nome vazio";
        response.status(204);
    } else {
        Piloto piloto = new Piloto();
        piloto.setNome(nome);
        piloto.setSobrenome(sobrenome);
        piloto.setPais(pais);
        piloto.setNumero(numero);
        piloto.setAbreviacao(abreviacao);

        if (pilotoDAO.insert(piloto)) {
            resp = "Piloto (" + nome + ") inserido!";
            response.status(201); // 201 Created
        } else {
            resp = "Piloto (" + nome + ") não inserido!";
            response.status(404); // 404 Not Found
        }
    }
    makeForm();
    return form.replaceFirst("<input type=\"hidden\" id=\"msg\" name=\"msg\" value=\"\">", "<input type=\"hidden\" id=\"msg\" name=\"msg\" value=\"" + resp +
}
```

DEPENDÊNCIAS MAVEN

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>ti2cc</groupId>
  <artifactId>SpeedFanatic</artifactId>
  <version>2.0</version>
  <dependencies>
    <dependency>
      <groupId>com.sparkjava</groupId>
      <artifactId>spark-core</artifactId>
      <version>2.9.3</version>
    </dependency>
    <dependency>
      <groupId>org.slf4j</groupId>
      <artifactId>slf4j-simple</artifactId>
      <version>1.7.21</version>
    </dependency>
    <dependency>
      <groupId>org.postgresql</groupId>
      <artifactId>postgresql</artifactId>
      <version>42.3.3</version>
    </dependency>
    <dependency>
      <groupId>io.jsonwebtoken</groupId>
      <artifactId>jjwt-impl</artifactId>
      <version>0.11.5</version>
    </dependency>
    <dependency>
      <groupId>io.jsonwebtoken</groupId>
      <artifactId>jjwt-api</artifactId>
      <version>0.11.5</version>
    </dependency>
    <dependency>
      <groupId>io.jsonwebtoken</groupId>
      <artifactId>jjwt-jackson</artifactId>
      <version>0.11.5</version>
    </dependency>
  </dependencies>
</project>
```

PÁGINA FÓRUM

Speed Fanatic

Equipes

Pilotos

Circuitos

Fórum

Perfil

Inserir Novo Tópico

Geral



Assuntos diversos

Descrição Conteúdo: Local para bate papo sobre assuntos diversos, fora do mundo da F1.

2 Posts

Último post by Pedro
em 2023-10-26

Pilotos



Estreantes promissores

Descrição do Conteúdo: Discuta sobre os pilotos mais jovens e talentosos que estão fazendo suas estreias na Fórmula 1 e as expectativas que cercam suas carreiras.

3 Posts

Último post by Pedro
em 2023-10-26



Pilotos Lendários

Descrição do Conteúdo: Discuta pilotos lendários da Fórmula 1, como Michael Schumacher, Ayrton Senna, Juan Manuel Fangio e outros que deixaram uma marca duradoura no esporte.

0 Posts

Último post by
em 2023-10-26



Técnica de pilotagem

Discuta as diferentes técnicas de pilotagem usadas por pilotos de F1, como frenagem, curvas, estratégias de ultrapassagem e como elas evoluíram ao longo dos anos.

0 Posts

Último post by
em 2023-10-26



Rivalidades memoráveis

Descrição do Conteúdo: Analise rivalidades históricas entre pilotos, como a rivalidade entre Senna e Prost ou entre Hamilton e Rosberg, e como essas rivalidades moldaram o esporte.

0 Posts

Último post by
em 2023-10-26



Pilotos contemporâneos

Descrição do Conteúdo: Aborde os pilotos atuais da Fórmula 1, como Lewis Hamilton, Max Verstappen, Charles Leclerc e outros, e como eles estão moldando o cenário competitivo atual da categoria.

0 Posts

Último post by
em 2023-10-26

TELA POSTS/POST

Speed Fanatic Equipes Pilotos Circuitos Fórum Perfil

Status

Tema



//asada

Iniciado por: Pedro .



//Esse posto e sobre os melhores corredores

Iniciado por: Pedro .

© SpeedFanatic

Speed Fanatic Equipes Pilotos Circuitos Fórum Perfil

Post: Esse posto e sobre os melhores corredores

Feito por: Pedro

Data: 2023-10-26

Comentar

Autor

Tópico:

pedrinho

Mensagem:

eu concordo com o que voce disse

Comentar

Enviar

TELA DE LOGIN/PERFIL

Speed Fanatic Equipes Pilotos Circuitos Fórum Perfil

Entrar

Não tem conta?

[Cadastre-se](#)

© 2023 SpeedFanatic

Speed Fanatic Equipes Pilotos Circuitos Fórum Perfil

Perfil

Nome: Pedro

Data de Criação: 2023-10-26

Email: novoemail@gmail.com

Editar

EXEMPLO LISTAGEM

Speed Fanatic [Equipes](#) [Pilotos](#) [Circuitos](#) [Fórum](#) [Perfil](#)

Circuitos

Hungaroring	Budapest	Hungary	Latitude: 47.5789	Longitude: 19.2486	Altitude: 264.0
Valencia Street Circuit	Valencia	Spain	Latitude: 39.4589	Longitude: -0.331667	Altitude: 4.0
Circuit de Spa-Francorchamps	Spa	Belgium	Latitude: 50.4372	Longitude: 5.97139	Altitude: 401.0
Autodromo Nazionale di Monza	Monza	Italy	Latitude: 45.6156	Longitude: 9.28111	Altitude: 162.0
Marina Bay Street Circuit	Marina Bay	Singapore	Latitude: 1.2914	Longitude: 103.864.0	Altitude: 18.0

<--

-->

SISTEMAS INTELIGENTES



RECURSOS E
TECNICAS



PROPOSIÇÕES DE
VALOR



APRENDIZADO DA IA

SISTEMAS INTELIGENTES

IS CANVAS - IA para Moderação de Fórum

Ferramenta IA

Banco de dados;
Machine Learning;
APIs principalmente do Microsoft Azure.

Entradas

Perguntas dos usuários no fórum;
Respostas dos usuários do fórum;
Base de dados com palavras ofensivas.

Proposição de Valor

Maior segurança;
Facilidade de moderação;
Melhor experiência do usuário;
Redução de problemas legais;
Conformidade com as diretrizes.

Equipe

A equipe raramente entrará em contato com o usuário, já que a IA removerá o conteúdo ofensivo. Caso algo passe despercebido, o usuário poderá entrar em contato conosco.

Clientes

Fãs de Formula 1 de todas as idades que querem se inteirar em uma comunidade apaixonada pelo esporte.

Saídas

Exclusão de conteúdo;
Feedback para usuários.

Stakeholders Chaves

Usuários do fórum;
Desenvolvedores;
Moderadores;

Custos

Custo operacional da API no Microsoft Azure.

Receitas

Não haverá receitas diretamente envolvidas no sistema inteligente, apenas com propagandas e patrocínios de terceiros.