QueryStream Chatbot Integration Guide

🚀 Quick Setup (3 Easy Steps)

Step 1: Download the Files

Save these three files to your project directory:

- `chatbot.css` (styles)
- `chatbot.js` (functionality)
- `chatbot.html` (HTML widget)

Step 2: Add to Your Pages

Add this code to **EVERY page** where you want the chatbot (before the closing `</body>` tag):

```html
<!-- QueryStream AI Chatbot -->
<link rel="stylesheet" href="chatbot.css">

<!-- Copy and paste the entire content from chatbot.html here -->
<div class="chatbot-widget">
  <!-- All the chatbot HTML goes here -->
</div>

<script src="chatbot.js"></script>
```

Step 3: Set Your API Key

In `chatbot.js`, replace this line:

```javascript
const GEMINI_API_KEY = 'YOUR_GEMINI_API_KEY';
```

With your actual Gemini API key:

```javascript
const GEMINI_API_KEY = 'your_actual_api_key_here';
```

📝 Complete Integration for Each Page

For index.html:

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- your existing head content -->
  <link rel="stylesheet" href="chatbot.css">
</head>
<body>
  <!-- your existing page content -->

  <!-- Add chatbot before closing body tag -->
  <!-- Copy the entire chatbot.html content here -->

  <script src="chatbot.js"></script>
</body>
</html>
```

For about.html, faq.html, contact.html, services.html, industries.html:

Same process - just add the CSS link, HTML content, and JavaScript file to each page.

🔐 Secure API Key Setup (Recommended)

Option A: Environment Variables (Vercel)

1. In your Vercel dashboard, go to Settings > Environment Variables
2. Add: `GEMINI_API_KEY` = `your_actual_api_key`
3. Update `chatbot.js`:

```javascript
const GEMINI_API_KEY = process.env.GEMINI_API_KEY || 'fallback_key';
```

Option B: Server-Side API Route (Most Secure)

Create `api/chat.js`:

```javascript
```

```
export default async function handler(req, res) {
  if (req.method !== 'POST') {
    return res.status(405).json({ message: 'Method not allowed' });
  }

  const GEMINI_API_KEY = process.env.GEMINI_API_KEY;
  const { message } = req.body;

  try {
    const response = await fetch('https://generativelanguage.googleapis.com/v1beta/models/gemini-2.0-flash:genera
      method: 'POST',
      headers: {
        'Content-Type': 'application/json',
        'X-goog-api-key': GEMINI_API_KEY
      },
      body: JSON.stringify({
        contents: [{ parts: [{ text: message }] }]
      })
    });

    const data = await response.json();
    res.status(200).json({ response: data.candidates[0].content.parts[0].text });
  } catch (error) {
    res.status(500).json({ error: 'Failed to get response' });
  }
}
```

Then update the `callGeminiAPI` function in `chatbot.js`:

```javascript
async function callGeminiAPI(userMessage) {
  const response = await fetch('/api/chat', {
    method: 'POST',
    headers: { 'Content-Type': 'application/json' },
    body: JSON.stringify({ message: userMessage })
  });

  const data = await response.json();
  return data.response;
}
```

🎯 Features Included

- ✅ **AI Integration**: Google Gemini 2.0 Flash
- ✅ **QueryStream-Only Responses**: Refuses unrelated questions
- ✅ **15-Minute Idle Timeout**: With warning message
- ✅ **GDPR Disclaimer**: "By chatting, you agree to our disclaimer"
- ✅ **Mobile Responsive**: Works on all devices
- ✅ **Anthropic Design**: Matches your website perfectly
- ✅ **Quick Actions**: Pre-built service questions
- ✅ **Typing Indicators**: Professional chat experience
- ✅ **Notification Badge**: Shows when chat is closed

🔧 Customization

Change Colors:

Edit the CSS variables in `chatbot.css`:

```css
:root {
    --royal-blue: #2563eb;    /* Change main blue color */
    --royal-green: #059669;   /* Change green accent */
    --cream: #f8f6f0;         /* Change background */
}
```

Modify Welcome Message:

Edit the welcome message in `chatbot.html`:

```html
<h4>👋 Welcome to QueryStream!</h4>
<p>Your custom welcome message here...</p>
```

Update Knowledge Base:

Modify the `querystreamContext` variable in `chatbot.js` to add or change business information.

📱 Testing

1. **Open any page** with the chatbot
2. **Click the blue chat bubble** in bottom-right corner
3. **Try these test messages**:
    - "What services do you offer?"
    - "How much does it cost?"

    - "What's the weather like?" (should refuse and redirect)

4. **Test mobile responsiveness** on different screen sizes
5. **Test idle timeout** by waiting 15+ minutes

🚀 Deployment to Vercel

1. **Upload all files** to your Vercel project
2. **Set environment variable**: `GEMINI_API_KEY` in Vercel dashboard
3. **Deploy**: `vercel --prod`
4. **Test live**: Check that chatbot appears and responds correctly

---

📞 Support

If you need help with integration:

- **Email**: hello@querystream.co.uk
- **Response**: Within 4 hours
- **Documentation**: This guide covers all setup steps

🎉 You're Done!

Your QueryStream AI chatbot is now ready to help customers 24/7 with:

- Service information
- Pricing details
- Business hours
- Contact information
- Lead qualification
- Professional responses

The chatbot will appear on every page where you add the code and provide intelligent, QueryStream-specific assistance to your website visitors!