

Quick Start Guide

Development

```
cd /home/ubuntu/code_artifacts/esgpro  
npm run dev
```

Then visit <http://localhost:3000>

Production Build

```
npm run build  
npm start
```

Project Features

Complete Chat Interface

- Natural conversational flow
- Real-time message updates
- Typing indicators
- Mobile-responsive design

Question Sequence

Collects:

1. Contact Name
2. Company Name
3. Company Size (small/medium/large)
4. Tender Days (urgency)
5. Industry
6. Current ESG Status
7. Pain Points

Results Display

Shows:

- Lead Score (0-100)
- Urgency Level (low/medium/high)
- Recommended Action (BOOK_FREE_CONSULT / BUY_50_ASSESSMENT / NURTURE)
- Focus Points
- Discussion Questions
- Pricing Recommendations

Full Bin Branding

- Header with Full Bin logo (text-based)
- Gradient color scheme (purple/blue)

- “Powered by Full Bin” footer
- Professional, clean design

File Structure

```

esgpro/
  └── components/
    ├── Layout.js           ✓ Header + Footer
    ├── ChatInterface.js    ✓ Main chat container
    ├── Message.js          ✓ Message bubbles
    ├── ResultsPanel.js     ✓ Results display
    └── TypingIndicator.js  ✓ Animated dots
  └── context/
    └── InterviewContext.js ✓ State management
  └── lib/
    ├── conversationFlow.js ✓ Question logic
    └── apiClient.js        ✓ API calls
  └── pages/
    ├── api/
      └── interview.js       ✓ API endpoint
    ├── _app.js
    ├── _document.js
    └── index.js             ✓ Home page
  └── styles/
    ├── globals.css          ✓ Global styles
    ├── Layout.module.css    ✓ Layout styles
    ├── ChatInterface.module.css
    ├── Message.module.css
    ├── TypingIndicator.module.css
    ├── ResultsPanel.module.css
    └── Home.module.css

```

API Testing

Test the interview API directly:

```

curl -X POST http://localhost:3000/api/interview \
-H "Content-Type: application/json" \
-d '{
  "contact_name": "John Smith",
  "company_name": "Acme Corp",
  "size": "medium",
  "tender_days": 30,
  "industry": "Manufacturing",
  "current_esg_status": "Just starting",
  "pain_points": "Need help with reporting"
}'

```

Expected response includes: action, urgency, score, lead_brief, focus_points, opening_questions, pricing_recommendation

Deployment

Vercel (Recommended)

1. Push to GitHub

2. Import on Vercel

3. Deploy

Environment Variables

Optional for development, required for production features:

- SUPABASE_URL
- SUPABASE_KEY
- STRIPE_SECRET_KEY
- SENDGRID_API_KEY (optional)
- TWILIO_SID (optional)
- TWILIO_TOKEN (optional)

Customization

Change Colors

Edit `styles/globals.css` :

```
background: linear-gradient(135deg, #YOUR_COLOR_1 0%, #YOUR_COLOR_2 100%);
```

Modify Questions

Edit `lib/conversationFlow.js` - add/remove/modify steps in the STEPS array

Adjust Scoring

Edit `pages/api/interview.js` - modify score calculation logic

Troubleshooting

Build Errors

```
rm -rf .next node_modules
npm install
npm run build
```

Port Already in Use

```
PORT=3001 npm run dev
```

What's Next?

1. **Database Integration:** Connect Supabase to store leads
2. **Notifications:** Set up SendGrid/Twilio for alerts
3. **Payments:** Integrate Stripe for £50 assessment
4. **Analytics:** Add tracking for conversions
5. **LLM Enhancement:** Use GPT for natural responses